

# Multi-Cloud Engineering Program

Parte 14 — Plataformas avanzadas, capstones y carrera

12 clases · 60 horas

Cuaderno de una sola parte: su introducción, sus clases completas y sus evaluaciones.

Extracto del manual integral, generado desde las mismas fuentes versionadas.

## Alcance de este cuaderno

| Componente      | Cobertura |
|-----------------|-----------|
| Parte           | 14 de 23  |
| Clases          | 169–180   |
| Evaluaciones    | 12        |
| Horas estimadas | 60        |

# Índice

|                                                                  |          |
|------------------------------------------------------------------|----------|
| <b>Alcance de este cuaderno</b>                                  | <b>2</b> |
| <b>Índice</b>                                                    | <b>3</b> |
| <b>Parte 14 — Plataformas avanzadas, capstones y carrera</b>     | <b>4</b> |
| 169 — Landing zones empresariales y vending de cuentas . . . . . | 6        |
| 170 — Gobierno federado y policy as code a escala . . . . .      | 15       |
| 171 — Platform as a Product y roadmap de capacidades . . . . .   | 25       |
| 172 — Modelo operativo FinOps y economía unitaria . . . . .      | 35       |
| 173 — Madurez SRE y confiabilidad organizacional . . . . .       | 45       |
| 174 — Arquitectura de seguridad cloud empresarial . . . . .      | 55       |
| 175 — Workloads de IA, GPU, datos y MLOps multi-cloud . . . . .  | 65       |
| 176 — Edge, IoT y procesamiento desconectado . . . . .           | 76       |
| 177 — Soberanía digital y confidential computing . . . . .       | 86       |
| 178 — Capstone: descubrimiento y diseño . . . . .                | 97       |
| 179 — Capstone: implementación y operación . . . . .             | 107      |
| 180 — Capstone: defensa, portafolio y plan profesional . . . . . | 118      |

# Parte 14 — Plataformas avanzadas, capstones y carrera

← Parte 13 · Índice completo · Parte 15 →

Descargar: Esta parte en PDF · Manual integral

Nivel: experto-frontera · Clases: 12 · Duración sugerida: 6–8 semanas

Integrar tecnología, operación, gobierno y comunicación en evidencia profesional.

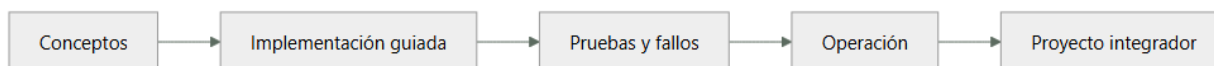
## Resultados de la parte

Al completar esta parte podrás explicar el dominio con lenguaje independiente del proveedor, implementar sus mecanismos esenciales, diagnosticar fallos y defender decisiones mediante evidencia, costo, seguridad y objetivos de confiabilidad.

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

## Secuencia



## Clases

| ID  | Clase                                            | Laboratorio | Horas |
|-----|--------------------------------------------------|-------------|-------|
| 169 | Landing zones empresariales y vending de cuentas | governance  | 4     |
| 170 | Gobierno federado y policy as code a escala      | governance  | 4     |
| 171 | Platform as a Product y roadmap de capacidades   | platform    | 4     |
| 172 | Modelo operativo FinOps y economía unitaria      | finops      | 4     |
| 173 | Madurez SRE y confiabilidad organizacional       | sre         | 4     |
| 174 | Arquitectura de seguridad cloud empresarial      | security    | 4     |
| 175 | Workloads de IA, GPU, datos y MLOps multi-cloud  | ai          | 4     |
| 176 | Edge, IoT y procesamiento desconectado           | edge        | 4     |
| 177 | Soberanía digital y confidential computing       | security    | 4     |
| 178 | Capstone: descubrimiento y diseño                | capstone    | 8     |
| 179 | Capstone: implementación y operación             | capstone    | 8     |
| 180 | Capstone: defensa, portafolio y plan profesional | capstone    | 8     |

## Evaluación de la parte

- 35 % laboratorios y evidencia reproducible.
- 25 % retos verificables.
- 20 % decisiones y ADR.
- 10 % seguridad, confiabilidad y costo.

- 10 % proyecto integrador de la parte.

Se aprueba con 80 % de clases en nivel B o superior y el proyecto integrador reproducible.

## **Pauta bibliográfica**

- Team Topologies — Skelton y Pais.
- Accelerate — Forsgren, Humble y Kim.
- Fundamentals of Software Architecture — Richards y Ford.

## 169 — Landing zones empresariales y vending de cuentas

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: governance · Estado: EXECUTABLECORE

### Propósito

Llevar la base de la clase 144 a una escala donde ya no se puede crear una cuenta a mano. La clase convierte la creación en un servicio de autoservicio que entrega una cuenta completa en minutos, y sostiene la condición sin la cual nada de esto existe: que sea la única forma de crear una cuenta. Después trata los dos problemas que solo aparecen con muchos equipos: un control correcto para el 90 % bloquea al 10 % restante, y las cuentas creadas hace dos años no tienen la base de hoy, que no se arregla con un proyecto de migración sino con un bucle.

### Resultados de aprendizaje

Al finalizar podrás:

1. Convertir la creación de cuentas en un servicio de autoservicio.
2. Impedir que exista otra forma de crear una cuenta.
3. Diseñar la jerarquía sabiendo que reorganizarla después duele.
4. Gestionar las excepciones para que nadie tenga que rodear un control.
5. Mantener al día las cuentas antiguas sin migraciones puntuales.

### Conceptos centrales

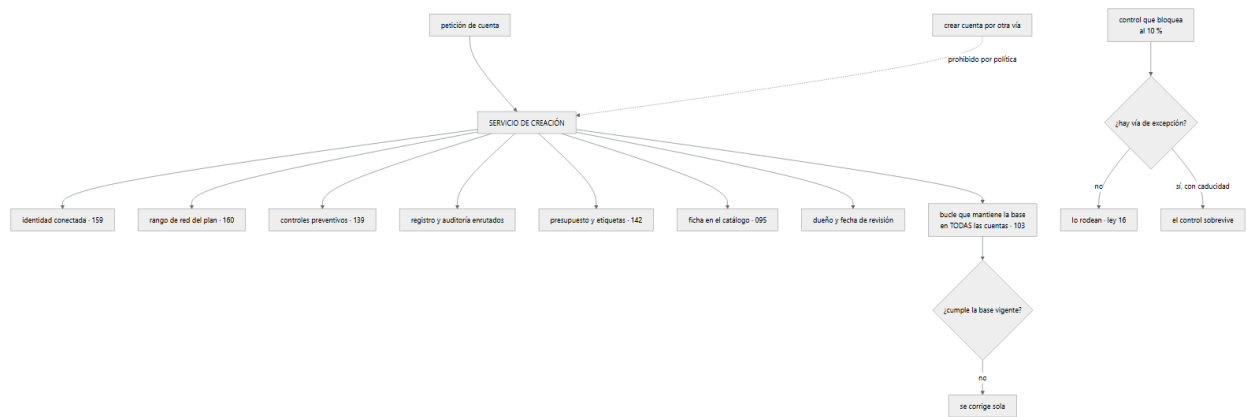
| Concepto                | Comprensión verificable                                                                                                              |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| servicio de creación    | Autoservicio que entrega una cuenta configurada: identidad, red, controles, registro, presupuesto, etiquetas y ficha en el catálogo. |
| vía única               | Propiedad de que no exista otra forma de crear una cuenta. Sin ella, la base no cubre lo que no pasó por ahí.                        |
| jerarquía               | Árbol de la organización donde se anclan los controles. Es una decisión de creación y reorganizarla mueve todo lo que cuelga.        |
| excepción con caducidad | Permiso explícito para saltarse un control, con motivo, responsable y fecha que revoca.                                              |
| deriva de base          | Diferencia entre la configuración con la que se creó una cuenta y la vigente. Crece con el tiempo y con el número de cuentas.        |
| cierre de cuenta        | Proceso de retirar una cuenta que ya no hace falta. Sin él, el inventario solo crece.                                                |

### Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

### ■ Flujo de razonamiento



## Desarrollo

### 1. La cuenta como unidad, y su servicio

A escala, la cuenta o proyecto deja de ser un contenedor y pasa a ser la unidad de aislamiento y de trabajo:

|                                               |           |
|-----------------------------------------------|-----------|
| es la frontera más fuerte que existe          | clase 133 |
| es donde se ancla el presupuesto              | clase 142 |
| es donde se aplican los controles preventivos | clase 139 |
| y es el radio de daño de casi cualquier error |           |

Y con decenas de equipos, crearlas a mano deja de ser viable:

60 equipos × 3 entornos 180 cuentas  
 cada una con identidad, red, controles, registro, presupuesto  
 y etiquetas  
 → a mano, cada una es medio día y ninguna queda igual que otra

De ahí el servicio de creación, que debe entregar una cuenta completa:

|                                                           |           |
|-----------------------------------------------------------|-----------|
| IDENTIDAD conectada al proveedor corporativo              | clase 159 |
| con los grupos del equipo ya asignados                    |           |
| RANGO DE RED asignado del plan central                    | clase 160 |
| y sin posibilidad de elegirlo                             |           |
| CONTROLES PREVENTIVOS de la organización                  | clase 139 |
| regiones, registro que no se desactiva, borrado de copias |           |
| REGISTRO Y AUDITORÍA enrutados a la cuenta central        | clase 141 |
| PRESUPUESTO con aviso por previsión                       | clase 142 |
| ETIQUETAS obligatorias ya puestas                         |           |
| FICHA EN EL CATÁLOGO con dueño y equipo                   | clase 095 |
| Y FECHA DE REVISIÓN                                       |           |

Y la propiedad que decide si todo lo anterior sirve para algo:

QUE SEA LA ÚNICA FORMA DE CREAR UNA CUENTA  
 → si alguien puede crear una desde la consola, esa cuenta no tiene nada de lo anterior  
 → y será una de las nueve que la clase 139 encontró fuera de todo inventario

Y se consigue con un control preventivo, no con una norma:

solo la identidad del servicio de creación puede crear cuentas  
 y esa identidad la usa el flujo, no una persona clase 098

Y el tiempo de entrega, que decide la adopción según la clase 106:

|                          |                                         |
|--------------------------|-----------------------------------------|
| por debajo de 30 minutos | se usa                                  |
| varios días              | la gente pide favores y aparecen atajos |

### 2. La jerarquía, que se decide una vez

Los controles se anclan en un árbol, y la forma de ese árbol es una decisión de creación —ley 14—: reorganizarla mueve todo lo que cuelga, con sus políticas y sus permisos.

Los criterios que compiten:

POR ENTORNO producción arriba, el resto abajo  
 + los controles duros se aplican a una rama entera

- los equipos quedan repartidos

#### POR UNIDAD DE NEGOCIO

- + el presupuesto y la responsabilidad coinciden con el árbol
- y las reorganizaciones de la empresa obligan a mover ramas

#### POR NIVEL DE EXIGENCIA

- regulado, interno, experimentación
- + los controles corresponden al riesgo
- hay que decidir dónde encaja cada carga

Y el diseño que suele aguantar mejor combina dos niveles:

- nivel 1 por exigencia: regulado / normal / experimentación
- nivel 2 por entorno: producción / preproducción / desarrollo
- nivel 3 por equipo o dominio

Y tres reglas prácticas:

#### DEJAR HUECO

- ramas vacías previstas para lo que vendrá: adquisiciones, una unidad nueva, un nivel de exigencia distinto

#### NO REFLEJAR EL ORGANIGRAMA

- cambia más deprisa que la infraestructura
- mejor por dominio o por exigencia, que son más estables

#### POCOS NIVELES

- cada nivel es un sitio más donde puede haber una política
- y calcular el efecto combinado se vuelve difícil clase 159

Y una cuenta especial que conviene prever desde el principio:

#### CUENTA DE SEGURIDAD Y AUDITORÍA

- recibe los registros de todas las demás
- y nadie de las demás puede escribir en ella clase 134
- con su propio acceso de emergencia

Y el tamaño de cuenta, que es la otra decisión:

- una por equipo y entorno lo habitual; equilibrio razonable
- una por carga y entorno aislamiento máximo, muchas cuentas
- una por inquilino solo si el aislamiento lo exige
- clase 154

Y el compromiso, que es el de la clase 148 aplicado aquí: más cuentas es menos radio de daño y más cosas que operar.

### 3. Controles y excepciones

Aquí aparece el problema propio de la escala, y es la ley 16:

- un control correcto para el 90 % de los equipos
- bloquea al 10 % restante
- y ese 10 % tiene un motivo legítimo

Y sin salida, ocurre lo previsible:

- piden una cuenta «especial» fuera del servicio de creación
- o consiguen que alguien les quite el control «temporalmente»
- o montan lo que necesitan en una cuenta que ya tenían
- y el control deja de existir para ellos, sin que nadie lo sepa

Y la corrección es tener vía de excepción, con la disciplina que este programa lleva usando desde la clase 046:

- se pide con un motivo escrito
- la aprueba alguien con autoridad para asumir ese riesgo
- tiene CADUCIDAD, y la caducidad revoca de verdad
- queda registrada y es visible
- y se revisa: si la misma excepción se pide diez veces,
- el control está mal diseñado, no los equipos

Y la última línea es la que hace que el sistema mejore:

- excepciones por control, ordenadas
- las tres primeras dicen qué hay que cambiar

Y el reparto entre preventivo y detectivo a escala, que es el de la clase 139:

PREVENTIVO lo que nunca debe ocurrir, y son POCAS reglas  
→ cuantas más haya, más excepciones y más rodeos  
→ cinco o diez, no cien

DETECTIVO todo lo demás, con línea base y prioridad por exposición

Y una advertencia sobre la tentación de la escala:

con sesenta equipos, la reacción natural es añadir controles  
→ y cada control nuevo tiene un coste que se multiplica por sesenta  
→ la pregunta antes de añadir uno: ¿qué incidente concreto evita?

Y el gobierno del propio conjunto de controles:

cada control tiene dueño, motivo y fecha de revisión  
y se retira el que no haya detectado ni impedido nada en un año  
→ es la misma higiene que las reglas de la clase 139

#### 4. Cuentas viejas y cuentas muertas

La deriva de base es el problema que aparece al segundo año:

las cuentas creadas en enero tienen la base de enero  
las de hoy tienen la de hoy  
y entre medias hay quince cambios de política  
→ tres generaciones de cuentas conviviendo

Y la respuesta equivocada es un proyecto de actualización:

«vamos a actualizar las 180 cuentas»  
→ dura meses, y al terminar la base ya ha cambiado otra vez

Y la correcta es la de la clase 103, aplicada a las cuentas:

UN BUCLE QUE APLICA LA BASE VIGENTE A TODAS, CONTINUAMENTE  
la base se declara en un repositorio  
y un proceso la reconcilia en cada cuenta  
→ una cuenta creada hace dos años converge sola  
→ y un cambio de base llega a las 180 sin proyecto

Y con ello vienen las mismas obligaciones que la clase 103 impuso:

alerta por ANTIGÜEDAD de la última reconciliación, por cuenta      ley 13  
comprobación del RESULTADO, no solo del estado del bucle      clase 164  
y campos que el bucle no debe tocar, declarados

El cierre de cuentas, que es la ley 20 a escala:

sin proceso de cierre, el inventario solo crece  
y aparecen las cuentas sin dueño que la clase 139 encontró

Y lo que hace falta:

cada cuenta tiene dueño y fecha de revisión desde su creación  
revisión periódica: ¿sigue haciendo falta? ¿quién responde por ella?  
y un proceso de cierre con pasos claros  
avisar, marcar, esperar, exportar lo que haya que conservar,  
y cerrar

Y las cifras que se vigilan en una organización con muchas cuentas:

cuentas creadas por el servicio frente a cuentas existentes  
cuentas sin dueño o con dueño inexistente  
cuentas sin actividad en 90 días  
cuentas con la base desactualizada, y desde cuándo  
excepciones vivas, por control y por antigüedad  
tiempo de entrega de una cuenta nueva  
y cuentas cerradas al año

La última suele ser cero, y ese es el problema: una organización que nunca cierra cuentas está acumulando riesgo y coste a la vez.

Y la lista de comprobación de la clase:

- existe un servicio de creación que entrega la cuenta completa
- es la única forma de crear una cuenta, impedida por política
- entrega en menos de 30 minutos
- la jerarquía tiene pocos niveles y hueco previsto
- hay cuenta de seguridad y auditoría a la que nadie más escribe

- los controles preventivos son pocos y cada uno tiene motivo
- existe vía de excepción con caducidad que revoca
- se revisan las excepciones más pedidas para corregir el control
- un bucle mantiene la base vigente en todas las cuentas
- hay alerta por antigüedad de reconciliación por cuenta
- se comprueba el resultado, no solo el estado del bucle
- cada cuenta tiene dueño y fecha de revisión
- existe proceso de cierre y se ejecuta

Y el cierre que enlaza con la clase siguiente: con cientos de cuentas y decenas de equipos, las políticas dejan de poder decidirse en un solo sitio. Cómo se reparte esa autoridad sin perder las garantías, y cómo se comprueba a esa escala, es la materia de la clase 170.

## Ejemplo trabajado

CloudShop pasa de 9 equipos a 60 tras dos adquisiciones. La base de la clase 144 funcionaba con 21 cuentas; con 180 deja de funcionar, y el ejercicio consiste en descubrir por dónde se rompe.

La situación a los seis meses del crecimiento.

|                                                    |           |
|----------------------------------------------------|-----------|
| cuentas                                            | 214       |
| creadas por el proceso establecido                 | 132       |
| creadas por otra vía                               | 82        |
| → adquisiciones                                    | 47        |
| → creadas desde la consola por equipos con permiso | 35        |
| tiempo de entrega de una cuenta                    | 6 días    |
| peticiones pendientes                              | 31        |
| cuentas sin dueño identificable                    | 29        |
| cuentas con la base completa                       | 81 de 214 |

Ochenta y dos cuentas fuera del proceso, y treinta y cinco de ellas creadas por gente que simplemente tenía el permiso.

Corrección 1: la vía única.

|                                    |                |                    |
|------------------------------------|----------------|--------------------|
| quién puede crear cuentas          | 35 identidades | 1 (el servicio)    |
| cómo se impide                     | norma escrita  | control preventivo |
| cuentas creadas fuera del servicio | 35 en 6 meses  | 0                  |

Y el efecto secundario que hubo que atender de inmediato:

|                                                                    |    |
|--------------------------------------------------------------------|----|
| al cortar la vía alternativa, las peticiones se acumularon         |    |
| peticiones pendientes la primera semana                            | 44 |
| → y esa cola es exactamente lo que hace que la gente busque atajos |    |

Corrección 2: el tiempo de entrega.

|                                    |                   |                                               |
|------------------------------------|-------------------|-----------------------------------------------|
| desglose de los 6 días             |                   |                                               |
| aprobación del responsable de área |                   | 2,5 días                                      |
| asignación de rango de red, a mano |                   | 1,5 días                                      |
| aplicación de la base, a mano      |                   | 1 día                                         |
| alta en el catálogo y presupuesto  |                   | 1 día                                         |
| aprobación                         | persona, 2,5 días | automática si el equipo existe en el catálogo |
| rango de red                       | a mano            | del plan, automático                          |
| base                               | a mano            | declarada y aplicada                          |
| catálogo y presupuesto             | a mano            | creados por el servicio                       |
| tiempo de entrega                  | 6 días            | 18 min                                        |
| peticiones pendientes              | 31                | 0                                             |
| cuentas creadas fuera del servicio | 35 / 6 meses      | 0                                             |

Y la aprobación humana se conservó solo donde aporta:

|                                      |                               |
|--------------------------------------|-------------------------------|
| cuenta de desarrollo o preproducción | automática                    |
| cuenta de producción                 | aprobación del dueño del área |
| cuenta en la rama regulada           | aprobación adicional          |

Corrección 3: las 82 cuentas que ya existían.

El primer plan fue un proyecto de actualización:

|                    |                                                  |
|--------------------|--------------------------------------------------|
| estimación         | 4 meses                                          |
| cuentas por semana | 5                                                |
| problema           | al terminar, la base habría cambiado 6 veces más |

Y se sustituyó por el bucle del apartado cuarto:

la base se declaró en un repositorio  
y un proceso la reconcilia en todas las cuentas cada 6 h

|                             |                                        |           |
|-----------------------------|----------------------------------------|-----------|
| semana 1                    | se aplicó en modo aviso                |           |
|                             | diferencias detectadas                 | 3.140     |
|                             | de ellas, en cuentas heredadas         | 2.610     |
| semana 2                    | se corrigieron las que no rompían nada |           |
| semana 4                    | se pasó a aplicar                      |           |
| cuentas con la base vigente | 81 de 214 → 214 de 214                 |           |
| tiempo total                |                                        | 5 semanas |

Y los quince cambios de base posteriores llegaron solos:

|                                                      |       |
|------------------------------------------------------|-------|
| cambios de base en 8 meses                           | 15    |
| proyectos de actualización                           | 0     |
| tiempo hasta que un cambio llega a todas las cuentas | < 6 h |

Y la alerta obligatoria, que encontró lo previsible:

|                                                     |   |
|-----------------------------------------------------|---|
| cuentas sin reconciliar en más de 24 h, detectadas  | 7 |
| de ellas, por permisos caducados                    | 4 |
| de ellas, por una cuenta suspendida por facturación | 2 |
| de ellas, por una cuenta que ya no existía          | 1 |

Las excepciones, y el control mal diseñado.

|                                    |    |           |
|------------------------------------|----|-----------|
| excepciones solicitadas en 8 meses | 94 |           |
| por control                        |    |           |
| regiones autorizadas               | 41 | ← destaca |
| tipos de instancia permitidos      | 22 |           |
| acceso público de almacenes        | 9  |           |
| otros                              | 22 |           |

Y las cuarenta y una de regiones se investigaron según la regla del apartado tercero:

|                                    |                                                                                                       |
|------------------------------------|-------------------------------------------------------------------------------------------------------|
| motivo real                        | 38 de las 41 eran equipos que necesitaban una región concreta para latencia, ya autorizada para otros |
| causa                              | la lista de regiones se había fijado hace dos años y no se había revisado                             |
| corrección                         | se ampliaron las regiones autorizadas de 4 a 7                                                        |
| excepciones de ese control después | 0-1 / mes                                                                                             |

El control estaba mal, no los equipos, y la cifra de excepciones lo dijo.

|                                               |          |          |
|-----------------------------------------------|----------|----------|
| excepciones vivas                             | 94       | 19       |
| con motivo, responsable y caducidad           | 41 de 94 | 19 de 19 |
| caducadas y aún activas                       | 23       | 0        |
| controles revisados por exceso de excepciones | 0        | 3        |
| controles preventivos totales                 | 31       | 12       |

→ se retiraron 19 que no habían impedido nada en un año

La jerarquía, rehecha una vez.

|                   |                                                                                                                                                                                           |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| jerarquía inicial | por unidad de negocio, 5 niveles                                                                                                                                                          |
| problema          | dos reorganizaciones de la empresa en 18 meses obligaron a mover ramas enteras cada movimiento cambiaba qué políticas aplicaban y hubo 2 incidentes por permisos que dejaron de heredarse |
| jerarquía nueva   | nivel 1: exigencia (regulado / normal / pruebas)<br>nivel 2: entorno<br>nivel 3: dominio<br>3 niveles, con 4 ramas vacías previstas                                                       |

|                                                                  |   |
|------------------------------------------------------------------|---|
| movimientos de rama tras el cambio, en 12 meses                  | 0 |
| → dos reorganizaciones más de la empresa, sin efecto en el árbol |   |

Y el coste de haberla rehecho:

|                 |           |
|-----------------|-----------|
| cuentas movidas | 214       |
| duración        | 3 semanas |

incidentes durante el cambio 2  
→ y es la ley 14: se hizo una vez y dolió; hacerlo cada año  
no sería viable

El cierre de cuentas.

|                                  |            |            |
|----------------------------------|------------|------------|
| cuentas con dueño                | 185 de 214 | 214 de 214 |
| con fecha de revisión            | 0          | 214        |
| revisión anual                   | no         | sí         |
| cuentas sin actividad en 90 días | 31         | 4          |
| cuentas cerradas en 12 meses     | 0          | 38         |
| coste mensual liberado           | —          | 2.900 €    |

Y el proceso de cierre, que hubo que escribir:

|                                                       |           |
|-------------------------------------------------------|-----------|
| avisar al dueño y al área                             | 30 días   |
| marcar como pendiente de cierre, sin crear nada nuevo |           |
| exportar lo que haya que conservar                    | clase 141 |
| suspender                                             | 14 días   |
| cerrar                                                |           |

|                                                    |         |
|----------------------------------------------------|---------|
| cuentas reactivadas tras el aviso                  | 6 de 44 |
| → las 6 tenían uso real que nadie había registrado |         |

A los doce meses.

|                                            |           |            |
|--------------------------------------------|-----------|------------|
| cuentas                                    | 214       | 176        |
| creadas fuera del servicio                 | 82        | 0          |
| tiempo de entrega                          | 6 días    | 18 min     |
| peticiones pendientes                      | 31        | 0          |
| cuentas con la base vigente                | 81 de 214 | 176 de 176 |
| tiempo de propagación de un cambio de base | proyecto  | < 6 h      |
| controles preventivos                      | 31        | 12         |
| excepciones vivas                          | 94        | 19         |
| caducadas y activas                        | 23        | 0          |
| cuentas con dueño y fecha de revisión      | 0 %       | 100 %      |
| cuentas cerradas                           | 0         | 38         |
| niveles de jerarquía                       | 5         | 3          |

La lección que esta clase abre para la parte 14: treinta y cinco cuentas se habían creado fuera del proceso no por rebeldía, sino porque el proceso tardaba seis días, y al cortar la vía alternativa sin arreglar el plazo la cola creció de inmediato. Y cuarenta y una de las noventa y cuatro excepciones pedían lo mismo: el control de regiones llevaba dos años sin revisarse y bloqueaba a treinta y ocho equipos con un motivo legítimo. A escala, un control mal calibrado no se discute: se rodea.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/169-landing-zones-empresariales-y-vending-de-cuentas/lab.py
```

El laboratorio selecciona el motor de práctica governance y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa landing-zone-empresarial en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

## Evidencia esperada

El artefacto principal es una jerarquía de política, ownership y evidencia. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye landing-zone-empresarial para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

### ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

### ■ Errores frecuentes

| Síntoma                                                      | Causa probable                                              | Corrección                                                                                                               |
|--------------------------------------------------------------|-------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| Aparecen cuentas que no tienen ninguno de los controles      | El proceso de creación no es la única vía                   | Impide por política que nadie salvo el servicio de creación pueda crear cuentas, y arregla a la vez el plazo de entrega. |
| La gente busca atajos para conseguir una cuenta              | Ley 16: el camino oficial tarda días                        | Entrega en menos de 30 minutos, con aprobación humana solo donde aporta.                                                 |
| Conviven varias generaciones de cuentas con bases distintas  | La base se aplicó al crear y nunca más                      | Un bucle que reconcilia la base vigente en todas las cuentas, con alerta por antigüedad y comprobación del resultado.    |
| Un control acumula decenas de excepciones                    | El control está mal calibrado, no los equipos               | Ordena las excepciones por control y corrige los tres primeros; y revisa las caducidades para que revoquen de verdad.    |
| Reorganizar la empresa obliga a mover ramas y rompe permisos | La jerarquía refleja el organigrama, que cambia más deprisa | Organiza por exigencia, entorno y dominio, con pocos niveles y ramas vacías previstas.                                   |
| El número de cuentas solo crece                              | Ley 20: no hay dueño, ni revisión, ni proceso de cierre     | Dueño y fecha de revisión desde la creación, revisión periódica y un proceso de cierre que se ejecute.                   |

### ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

### ■ Preguntas de comprobación

1. ¿Qué debe entregar un servicio de creación de cuentas y cuál es su propiedad imprescindible?
2. ¿Por qué la jerarquía no debe reflejar el organigrama?
3. ¿Qué ocurre cuando un control correcto para la mayoría bloquea a una minoría con motivo?
4. ¿Por qué actualizar las cuentas antiguas con un proyecto no funciona?
5. ¿Qué indica que una organización nunca cierra cuentas?

## Referencias

- AWS (2025). Account vending and Control Tower — creación automatizada con base aplicada.  
<<https://docs.aws.amazon.com/controltower/latest/userguide/what-is-control-tower.html>>

- Google Cloud (2025). Resource hierarchy and project factory — jerarquía, herencia de políticas y creación automatizada. <<https://cloud.google.com/resource-manager/docs/cloud-platform-resource-hierarchy>>
- Microsoft (2025). Enterprise-scale landing zones — áreas de diseño y jerarquía de grupos de administración. <<https://learn.microsoft.com/azure/cloud-adoption-framework/ready/enterprise-scale/>>
- CIS (2025). Foundations benchmarks — controles mínimos aplicables a toda cuenta. <<https://www.cisecurity.org/cis-benchmarks>>
- FinOps Foundation (2025). Account structure and allocation — la cuenta como unidad de atribución. <<https://www.finops.org/framework/capabilities/allocation/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                                | Índice              | Siguiente                                           |
|---------------------------------------------------------|---------------------|-----------------------------------------------------|
| ← 168 · Proyecto: continuidad activa-pasiva entre nubes | Parte 14 · Programa | 170 · Gobierno federado y policy as code a escala → |

## Evaluación — 169 Landing zones empresariales y vending de cuentas

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega landing-zone-empresarial con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- [ ] Resultado reproducible por una segunda persona.
- [ ] Evidencia vinculada a cada afirmación técnica.
- [ ] Prueba negativa o de fallo incluida.
- [ ] Costos y permisos expresados con unidades y alcance.
- [ ] Limitaciones declaradas sin presentar la simulación como producción.

### Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 170 — Gobierno federado y policy as code a escala

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: governance · Estado: EXECUTABLECORE

## Propósito

Gobernar decenas de equipos sin que un grupo central revise todo ni cada equipo invente su propia respuesta. La clase propone una división que resuelve la mayor parte de las discusiones —tres niveles de política, y solo el primero se impone técnicamente— y sostiene una idea incómoda: lo que no está en el primer nivel se incumplirá en algún sitio, y eso no es un fracaso: es el diseño. Después da el ciclo de vida de una política, con el paso que casi nadie da y que evita bloquear a media organización: medir a cuántos recursos afectaría antes de activarla.

## Resultados de aprendizaje

Al finalizar podrás:

1. Repartir las políticas en tres niveles según cómo se hacen cumplir.
2. Mantener el primer nivel corto, con un coste explícito para ampliarlo.
3. Publicar el impacto de una política antes de activarla.
4. Organizar quién decide sin convertirlo en un cuello de botella.
5. Medir el gobierno por lo que llega a producción, no por el número de reglas.

## Conceptos centrales

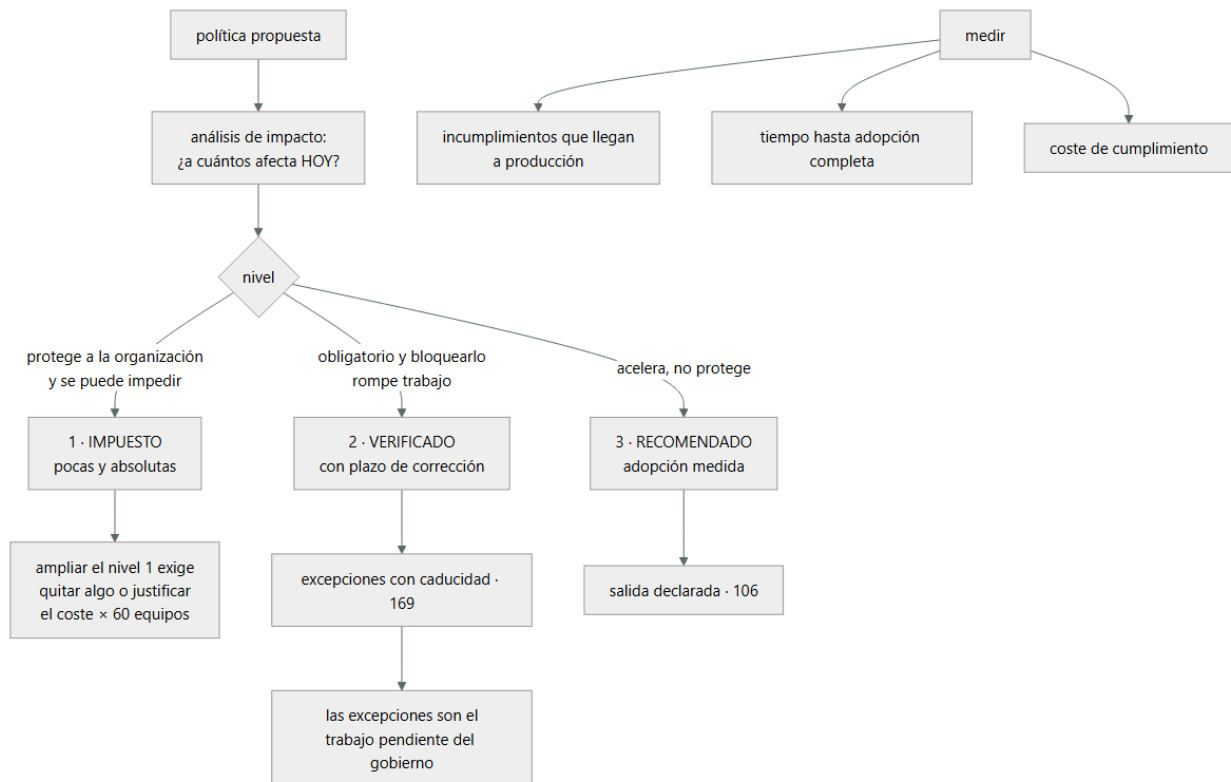
| Concepto                     | Comprensión verificable                                                                                                              |
|------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| nivel obligatorio impuesto   | Política que la plataforma impide incumplir. Son pocas y absolutas; ampliarlas tiene un coste multiplicado por el número de equipos. |
| nivel obligatorio verificado | Política que no se puede impedir sin bloquear trabajo legítimo. Se detecta y se corrige con plazo.                                   |
| nivel recomendado            | Camino asfaltado. Se mide la adopción y desviarse es legítimo si se declara.                                                         |
| análisis de impacto          | Cuántos recursos existentes incumplirían una política si se activara hoy. Es el paso que evita bloquear a media organización.        |
| órgano de gobierno           | Grupo pequeño que posee la lista del primer nivel, decide deprisa y se reúne poco.                                                   |
| coste de cumplimiento        | Tiempo que los equipos dedican a cumplir. Es la medida que falta en casi todos los programas de gobierno.                            |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. Tres niveles, y solo uno se impone

Con sesenta equipos, los dos extremos fallan:

**CENTRAL LO REVISA ABSOLUTAMENTE todo**

→ cuello de botella; el gobierno se convierte en la ventanilla que la clase 106 describió

**CADA EQUIPO DECIDE**

→ sesenta respuestas distintas a la misma pregunta  
→ y ninguna garantía transversal

Y el reparto que funciona es por cómo se hace cumplir cada cosa:

**NIVEL 1 · OBLIGATORIO E IMPUESTO**

la plataforma lo impide; no hay forma de incumplirlo  
protege a la organización entera de un daño grave  
→ regiones autorizadas, registro que no se desactiva,  
borrado de copias, acceso público, etiquetas de dueño  
→ POCAS: cinco a quince, no cien

**NIVEL 2 · OBLIGATORIO Y VERIFICADO**

hay que cumplirlo, y bloquearlo rompería trabajo legítimo  
se detecta y se corrige con plazo clase 139  
→ cifrado en tránsito interno, retención de copias,  
permisos sin usar, vulnerabilidades del embudo

**NIVEL 3 · RECOMENDADO**

acelera y no protege; es el camino asfaltado clase 106  
se mide la adopción y desviarse es legítimo si se declara  
→ plantilla de canalización, módulos, biblioteca de registro

Y la afirmación que ordena el resto:

lo que no está en el nivel 1 se incumplirá en algún sitio  
y eso NO es un fracaso: es el diseño  
→ por eso el nivel 2 tiene detección y plazo, y el 3 tiene medida de adopción y no una queja

Y el criterio para poner algo en el nivel 1, que debe ser exigente:

¿su incumplimiento puede causar un daño grave e irreversible?  
¿se puede impedir sin bloquear trabajo legítimo?  
¿hay una vía de excepción para los casos raros?  
→ tres síes, o no es nivel 1

Y el precio de ampliarlo, que hay que hacer visible:

cada regla del nivel 1 se multiplica por sesenta equipos  
→ sesenta veces la fricción, sesenta veces las excepciones  
→ y por eso ampliar el nivel 1 exige QUITAR algo  
o justificar el coste con un incidente concreto

Y el efecto de no hacerlo: una lista de cien reglas obligatorias es una lista que nadie conoce, y entonces la ley 15 y la ley 16 actúan a la vez.

## 2. El ciclo de vida de una política

Una política pasa por cinco fases, y la segunda es la que casi nadie ejecuta:

1. PROPUESTA  
qué problema resuelve, con un incidente o un requisito detrás  
quién la propone y quién la sufrirá
2. ANÁLISIS DE IMPACTO ← el paso decisivo  
¿cuántos recursos la incumplirían HOY?  
¿de cuántos equipos?  
¿cuánto costaría corregirlos?
3. MODO AVISO  
se detecta y no se bloquea; se publica la cifra  
y se acompaña a los equipos con más incumplimientos
4. APLICACIÓN  
cuando la lista de incumplimientos nuevos esté vacía  
y con la base histórica congelada si hace falta clase 101
5. REVISIÓN  
¿ha impedido algo? ¿cuántas excepciones acumula?  
¿sigue teniendo sentido?

Y el paso 2 merece detenerse porque cambia la conversación:

sin análisis de impacto  
«vamos a exigir X» → se activa → 340 recursos dejan de funcionar  
→ y el gobierno pierde la confianza de los equipos para siempre

con análisis de impacto  
«esta política afectaría hoy a 340 recursos de 14 equipos;  
corregirlos cuesta unas 6 semanas repartidas»  
→ y entonces se decide si compensa, y con qué plazo

Y lo que se publica junto a cada política, que es lo que la hace discutible:

qué problema resuelve  
cuántos recursos afecta hoy  
qué hay que hacer para cumplirla, con un ejemplo  
quién la aprueba y desde cuándo  
y cómo se pide una excepción

Y las políticas se tratan como código, con lo que eso implica desde la clase 139:

viven en un repositorio con revisión  
cada una tiene PRUEBA NEGATIVA que demuestra que detecta  
se despliegan por la canalización  
y se versionan: cambiar una política es un cambio

Y una comprobación que a escala es imprescindible:

las pruebas negativas se ejecutan en TODAS las cuentas y clústeres,  
no en uno clase 164  
→ una política puede estar activa en 170 cuentas y ausente en 6

## 3. Quién decide, sin ser un cuello de botella

El gobierno necesita un dueño, y ese dueño no puede revisar el trabajo de sesenta equipos.

LO QUE POSEE EL ÓRGANO CENTRAL

la lista del nivel 1, corta y con motivo  
el mecanismo: cómo se declara, se prueba y se aplica una política  
el proceso de excepción  
y las medidas del programa

#### LO QUE NO POSEE

las decisiones de arquitectura de cada equipo  
la revisión de cada cambio  
ni la elección de tecnologías dentro del marco

Y su forma, para que decida deprisa:

pequeño: cinco a siete personas  
con representación de los equipos, no solo de la plataforma  
se reúne poco y decide en la reunión  
y publica lo decidido, con el motivo

Y dos reglas que evitan que se convierta en un comité:

#### SILENCIO POSITIVO PARA EL NIVEL 3

una propuesta de camino asfaltado se adopta si nadie objeta  
en un plazo

#### DECISIÓN POR DEFECTO

si no hay acuerdo sobre subir algo al nivel 1, se queda en el 2  
→ el nivel 1 solo crece por decisión explícita

Las excepciones son la información más valiosa que produce este sistema:

cada excepción dice que un control no encaja en un caso real  
y la lista ordenada por control dice qué hay que arreglar  
→ las excepciones SON el trabajo pendiente del gobierno      clase 169

Y el reparto de quién aprueba qué:

nivel 1    el órgano central, y con caducidad corta  
nivel 2    el dueño del área, con caducidad y registro  
nivel 3    nadie: desviarse es legítimo; basta declararlo      clase 106

Y la participación de los equipos, que es lo que evita que el gobierno se perciba como algo ajeno:

cualquiera puede PROPONER una política  
y quien la propone participa en su análisis de impacto  
→ y varias de las mejores políticas salen de un equipo que sufrió  
un incidente y no quiere que le pase a nadie más

## 4. Medir el gobierno

La medida habitual —número de políticas— es exactamente la que la ley 17 desaconseja:

mal    «tenemos 140 políticas»  
→ sube sola, no dice nada y empeora el sistema

Lo que sí mide:

#### INCUMPLIMIENTOS QUE LLEGAN A PRODUCCIÓN

del nivel 1, deberían ser cero por construcción  
del nivel 2, cuántos y cuánto tardan en corregirse

#### TIEMPO DESDE QUE SE PUBLICA UNA POLÍTICA HASTA QUE SE CUMPLE

en toda la organización  
→ si son meses, el mecanismo es lento

#### EXCEPCIONES POR POLÍTICA

y su tendencia: las tres primeras señalan qué corregir

#### ADOPCIÓN DEL NIVEL 3

qué proporción usa el camino asfaltado, sin obligación  
→ es la medida honesta de la clase 106

#### COSTE DE CUMPLIMIENTO

horas al mes que los equipos dedican a cumplir  
→ la medida que falta en casi todos los programas

#### POLÍTICAS QUE NO HAN IMPEDIDO NI DETECTADO NADA EN UN AÑO

→ candidatas a retirarse      clase 139

Y la quinta merece detalle porque es la que evita que el gobierno crezca sin freno:

si cumplir cuesta a cada equipo 6 horas al mes  
con 60 equipos son 360 horas al mes  
→ dos personas y media a tiempo completo, repartidas  
→ y esa cifra hay que comparar con lo que el gobierno evita

Y el equilibrio que hay que vigilar, que es el traslado de la clase 155:

más gobierno      menos riesgo y menos velocidad  
menos gobierno    más velocidad y más incidentes  
→ y las dos columnas se publican juntas, como las cuatro medidas  
de la clase 107

Y una comprobación anual sana:

recorrer la lista del nivel 1 y preguntar por cada una  
¿qué incidente evita?  
¿ha intentado incumplirla alguien este año?  
¿cuántas excepciones tiene?  
→ y retirar o bajar de nivel lo que no se sostenga

Y la lista de comprobación de la clase:

- las políticas están repartidas en tres niveles, con criterio escrito
- el nivel 1 tiene menos de quince reglas y cada una su motivo
- ampliar el nivel 1 exige quitar algo o justificar el coste
- toda política nueva pasa por análisis de impacto antes de activarse
- se publica a cuántos recursos y equipos afectaría
- hay modo aviso antes de aplicar
- cada política tiene prueba negativa, ejecutada en todas las cuentas
- existe un órgano pequeño que posee el nivel 1 y decide deprisa
- cualquiera puede proponer una política
- las excepciones se ordenan por control y son el trabajo pendiente
- se mide lo que llega a producción, no el número de políticas
- se mide el coste de cumplimiento en horas de los equipos
- se retira lo que no ha impedido ni detectado nada en un año

Y el cierre que enlaza con la clase siguiente: gobernar bien no basta si lo que la plataforma ofrece no es lo que los equipos necesitan. Tratar la plataforma como un producto con hoja de ruta, y decidir qué capacidad se construye antes que otra, es la materia de la clase 171.

## Ejemplo trabajado

CloudShop tiene sesenta equipos y ciento cuarenta políticas, de las que ciento diez son obligatorias. El ejercicio empieza contando cuántas conoce la gente y termina con un nivel 1 de doce reglas.

El punto de partida.

|                                                     |                                   |
|-----------------------------------------------------|-----------------------------------|
| políticas declaradas                                | 140                               |
| declaradas obligatorias                             | 110                               |
| aplicadas técnicamente                              | 31                                |
| el resto                                            | documentos que alguien debía leer |
| encuesta a 24 personas de 12 equipos                |                                   |
| políticas obligatorias que sabían citar, mediana    | 4                                 |
| personas que sabían dónde estaba la lista           | 9 de 24                           |
| personas que habían pedido una excepción alguna vez | 3                                 |

Cuatro de ciento diez. Es la ley 15 aplicada al gobierno: una lista que nadie puede conocer no gobierna nada.

La reclasificación en tres niveles.

|                                |    |    |
|--------------------------------|----|----|
| nivel 1, impuesto técnicamente | 31 | 12 |
| nivel 2, verificado con plazo  | —  | 34 |
| nivel 3, recomendado           | —  | 41 |
| retiradas                      | —  | 53 |

Y las cincuenta y tres retiradas, por qué:

|                                                |    |         |
|------------------------------------------------|----|---------|
| no habían detectado ni impedido nada en 2 años | 29 |         |
| duplicaban otra                                | 14 |         |
| describían una preferencia, no un riesgo       | 7  |         |
| contradecían a otra                            | 3  | ← grave |

Las tres contradictorias eran el hallazgo incómodo: dos políticas obligatorias pedían cosas incompatibles, y los equipos habían estado incumpliendo una u otra sin saberlo.

Y el nivel 1 resultante, con su criterio:

|                                                   |                 |
|---------------------------------------------------|-----------------|
| regiones autorizadas                              |                 |
| registro de auditoría no desactivable             |                 |
| borrado de copias prohibido                       |                 |
| sin acceso público en almacenes                   |                 |
| etiquetas de dueño y entorno obligatorias         |                 |
| sin claves de larga duración                      | clase 137       |
| firma verificada en admisión                      | clases 067, 101 |
| sin usuarios locales en las nubes                 | clase 159       |
| sin acceso desde entornos inferiores a producción | clase 133       |
| rangos de red del plan central                    | clase 160       |
| sin recursos fuera del servicio de creación       | clase 169       |
| cifrado en reposo con claves gestionadas          |                 |

■■■■■

12

Y la regla para ampliarla, acordada por escrito:

añadir una regla al nivel 1 exige  
un incidente concreto que la habría evitado, o un requisito legal  
y quitar otra, o justificar el coste ante el órgano

|                                        |   |
|----------------------------------------|---|
| reglas añadidas al nivel 1 en 12 meses | 2 |
| reglas retiradas del nivel 1           | 1 |

El análisis de impacto, y la política que se detuvo a tiempo.

propuesta «todo almacén de objetos debe tener versionado activo»  
motivo un borrado accidental en un equipo

|                                                   |              |
|---------------------------------------------------|--------------|
| análisis de impacto, ejecutado antes de activarla |              |
| almacenes existentes                              | 910          |
| que incumplirían                                  | 341          |
| equipos afectados                                 | 14           |
| coste estimado de corregir                        | ~6 semanas   |
| coste adicional de almacenamiento                 | ~1.900 €/mes |

Y la conversación que ese dato permitió:

sin el análisis se habría activado y 341 recursos habrían quedado  
marcados como incumplidores de un día para otro  
con el análisis se decidió  
nivel 2, no nivel 1  
aplicable solo a almacenes con datos clasificados  
como confidenciales o personales clase 141  
plazo de 90 días  
→ recursos afectados: de 341 a 62  
→ coste adicional: de 1.900 € a 310 €/mes

Y el resultado a los 90 días:

|                                       |          |
|---------------------------------------|----------|
| recursos corregidos                   | 59 de 62 |
| excepciones concedidas                | 3        |
| → todos con motivo y caducidad        |          |
| incumplimientos nuevos desde entonces | 0        |

Las excepciones como trabajo pendiente.

|                               |    |
|-------------------------------|----|
| excepciones vivas al empezar  | 94 |
| tras la reclasificación       | 19 |
| después de 12 meses           | 23 |
| ordenadas por política        |    |
| tipos de instancia permitidos | 9  |
| retención mínima de copias    | 6  |
| cifrado en tránsito interno   | 4  |
| otras                         | 4  |

Y las nueve primeras se investigaron:

motivo real equipos que necesitaban instancias con acelerador  
para cargas de aprendizaje clase 175  
causa la lista se escribió antes de que existieran esas cargas

corrección se ampliaron los tipos permitidos, con límite de coste  
 excepciones de esa política después 0-1

#### El órgano de gobierno.

|                                                  |                         |                                             |
|--------------------------------------------------|-------------------------|---------------------------------------------|
| quién decidía                                    | el equipo de plataforma | órgano de 6                                 |
| composición                                      | —                       | 3 de plataforma,<br>3 de equipos rotatorios |
| frecuencia                                       | reuniones semanales     | mensual                                     |
| decisiones pendientes al terminar cada reunión   | varias                  | 0                                           |
| tiempo medio de una decisión                     | 3 semanas               | 11 días                                     |
| propuestas hechas por equipos, no por plataforma | 0                       | 14                                          |
| de ellas, adoptadas                              | —                       | 9                                           |

Y nueve de las catorce propuestas de los equipos se adoptaron; cuatro venían de un incidente que ese equipo había sufrido.

#### Las medidas.

|                                                |                  |           |
|------------------------------------------------|------------------|-----------|
| políticas totales                              | 140              | 87        |
| nivel 1                                        | 31               | 12        |
| incumplimientos del nivel 1 en producción      | no se sabía      | 0         |
| incumplimientos del nivel 2 abiertos           | no se sabía      | 41        |
| tiempo medio de corrección del nivel 2         | —                | 19 días   |
| tiempo desde publicar hasta cumplimiento total | —                | 62 días   |
| adopción del nivel 3, sin obligación           | —                | 78 %      |
| coste de cumplimiento, horas/equipo/mes        | no se medía      | 4,1 h     |
| coste total de cumplimiento                    | —                | 246 h/mes |
| políticas con prueba negativa                  | 0 de 140         | 87 de 87  |
| políticas activas en todas las cuentas         | no se comprobaba | 87 de 87  |

Y la comprobación de la última fila encontró lo previsible:

|                                                      |   |
|------------------------------------------------------|---|
| al ejecutar las pruebas negativas en las 176 cuentas |   |
| políticas ausentes en alguna cuenta                  | 6 |
| cuentas donde faltaba alguna                         | 9 |
| → todas, cuentas heredadas de adquisiciones          |   |
| → corregidas por el bucle de la clase 169            |   |

El coste de cumplimiento, comparado con lo que evita.

|                                              |           |
|----------------------------------------------|-----------|
| coste de cumplimiento                        | 246 h/mes |
| ≈ 1,5 personas a tiempo completo, repartidas |           |

|                                                       |     |
|-------------------------------------------------------|-----|
| lo que el gobierno evitó, medido en 12 meses          |     |
| intentos de crear recursos en regiones no autorizadas | 118 |
| intentos de desactivar el registro de auditoría       | 4   |
| intentos de hacer público un almacén con datos        | 11  |
| intentos de crear claves de larga duración            | 87  |
| borrados de copias impedidos                          | 6   |

Y el dato que cerró la discusión sobre si el gobierno «frenaba»:

|                                                                                       |     |
|---------------------------------------------------------------------------------------|-----|
| de los 226 intentos impedidos, ¿cuántos eran errores y cuántos necesidades legítimas? |     |
| errores o desconocimiento                                                             | 211 |
| necesidades legítimas                                                                 | 15  |
| → las 15 se resolvieron con excepción o cambiando la política                         |     |

A los doce meses.

|                                                |                  |           |
|------------------------------------------------|------------------|-----------|
| políticas                                      | 140              | 87        |
| obligatorias impuestas                         | 31               | 12        |
| políticas que nadie conocía                    | ~106             | —         |
| políticas contradictorias                      | 3                | 0         |
| con prueba negativa                            | 0                | 87        |
| activas en todas las cuentas                   | no se comprobaba | 87 de 87  |
| análisis de impacto antes de activar           | no               | sí        |
| políticas activadas sin él                     | —                | 0         |
| excepciones vivas                              | 94               | 23        |
| controles corregidos por exceso de excepciones | 0                | 3         |
| propuestas de los equipos                      | 0                | 14        |
| tiempo de una decisión de gobierno             | 3 semanas        | 11 días   |
| coste de cumplimiento medido                   | no               | 246 h/mes |

La lección que esta clase traslada a la parte 14: había ciento diez políticas obligatorias y la gente podía citar cuatro; tres de ellas se contradecían entre sí. Y la política que estuvo a punto de bloquear trescientos cuarenta y un recursos de catorce equipos se detuvo por un paso que costó una tarde: medir a cuántos afectaría antes de activarla. Con ese dato, la misma política se aplicó a sesenta y dos recursos, con plazo, y se cumplió en tres meses sin que nadie tuviera que rodearla.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/170-gobierno-federado-y-policy-as-code-a-escala/lab.py
```

El laboratorio selecciona el motor de práctica governance y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa modelo-gobierno en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

### Evidencia esperada

El artefacto principal es una jerarquía de política, ownership y evidencia. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye modelo-gobierno para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

### ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■ Errores frecuentes

| Síntoma                                                                | Causa probable                                                                                     | Corrección                                                                                                                  |
|------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| Hay decenas de políticas obligatorias y nadie las conoce               | Ley 15: una lista que no se puede conocer no gobierna                                              | Reparte en tres niveles, deja menos de quince reglas impuestas y retira lo que no haya detectado nada.                      |
| Activar una política deja a media organización incumpliendo            | No se midió el impacto antes                                                                       | Análisis de impacto obligatorio: cuántos recursos y equipos, y qué cuesta corregirlo; después, modo aviso antes de aplicar. |
| Dos políticas obligatorias piden cosas incompatibles                   | Se acumularon sin revisión ni dueño                                                                | Revisión anual de la lista completa, con motivo por regla, y retirada de lo duplicado y lo contradictorio.                  |
| El gobierno se convierte en un cuello de botella                       | El órgano central revisa el trabajo de los equipos en vez de poseer solo el nivel 1 y el mecanismo | Grupo pequeño con representación, que decide en la reunión, con silencio positivo para lo recomendado.                      |
| Una política está activa en la mayoría de cuentas y ausente en algunas | No se comprueba el resultado en todas                                                              | Prueba negativa por política, ejecutada en todas las cuentas y clústeres.                                                   |
| Se discute si el gobierno frena sin datos                              | Se mide el número de políticas y no el coste ni lo que evita                                       | Publica juntos el coste de cumplimiento en horas y los intentos impedidos, distinguiendo errores de necesidades legítimas.  |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Qué distingue los tres niveles de política y cuál se impone técnicamente?
2. ¿Por qué ampliar el nivel 1 debe tener un coste explícito?
3. ¿Qué aporta el análisis de impacto y qué evita?
4. ¿Qué posee el órgano de gobierno y qué no?
5. ¿Qué se mide en lugar del número de políticas?

## Referencias

- Open Policy Agent (2025). Policy as code at scale: bundles and testing — declarar, probar y distribuir políticas.  
<<https://www.openpolicyagent.org/docs/latest/management-bundles/>>
- AWS (2025). Guardrails: preventive and detective controls — reparto entre lo que se impide y lo que se detecta.  
<<https://docs.aws.amazon.com/controltower/latest/userguide/controls.html>>
- Google Cloud (2025). Organization policy and custom constraints — políticas heredadas por la jerarquía.  
<<https://cloud.google.com/resource-manager/docs/organization-policy/creating-managing-custom-constraints>>
- Microsoft (2025). Azure Policy: initiatives, compliance and remediation — agrupación, cumplimiento y corrección.  
<<https://learn.microsoft.com/azure/governance/policy/overview>>
- Skelton, M. y Pais, M. (2019). Team Topologies, cap. 4 — gobierno federado y carga cognitiva de los equipos.  
<<https://teamtopologies.com/book>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                                 | Índice              | Siguiente                                              |
|----------------------------------------------------------|---------------------|--------------------------------------------------------|
| ← 169 · Landing zones empresariales y vending de cuentas | Parte 14 · Programa | 171 · Platform as a Product y roadmap de capacidades → |

## Evaluación — 170 Gobierno federado y policy as code a escala

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega modelo-gobierno con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- ☐ Resultado reproducible por una segunda persona.
- ☐ Evidencia vinculada a cada afirmación técnica.
- ☐ Prueba negativa o de fallo incluida.
- ☐ Costos y permisos expresados con unidades y alcance.
- ☐ Limitaciones declaradas sin presentar la simulación como producción.

### Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 171 — Platform as a Product y roadmap de capacidades

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: platform · Estado: EXECUTABLECORE

## Propósito

Decidir qué construye la plataforma a continuación, que a escala es la pregunta que más dinero mueve y la que peor se responde. La clase sustituye la lista de funciones por un mapa de capacidades con niveles, que permite decir dónde se está y no solo qué falta; ordena el trabajo con señales de demanda observadas en vez de opiniones; y sostiene la mitad que nadie pone en una hoja de ruta: lo que se va a retirar, porque la capacidad de un equipo de plataforma la limita lo que mantiene, no lo que construye.

## Resultados de aprendizaje

Al finalizar podrás:

1. Describir la plataforma como capacidades con niveles, no como funciones.
2. Ordenar el trabajo con señales de demanda medidas.
3. Decidir construir, comprar o adoptar cada capacidad.
4. Incluir retiradas en la hoja de ruta, con su mecánica.
5. Comunicar direcciones en vez de fechas, y medir el producto.

## Conceptos centrales

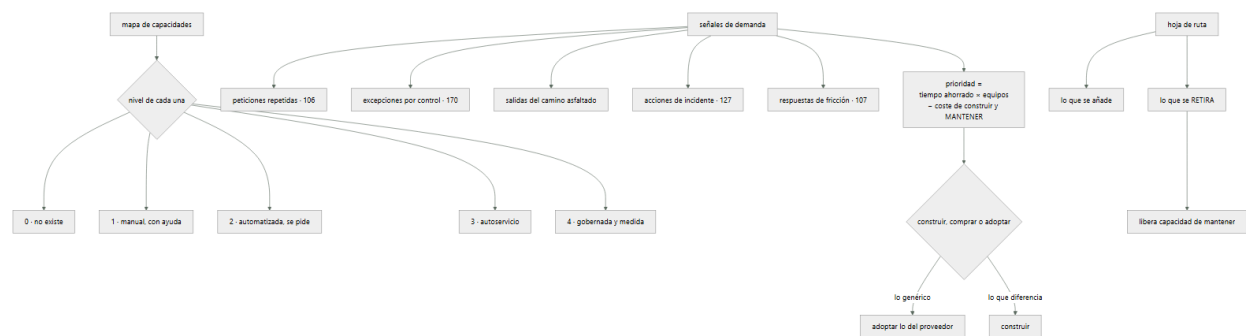
| Concepto                 | Comprensión verificable                                                                                                                         |
|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| capacidad                | Algo que la plataforma permite hacer a un equipo, expresado desde su punto de vista: «desplegar sin parada», «saber cuánto cuesta mi servicio». |
| nivel de madurez         | Grado en que una capacidad está resuelta: inexistente, manual, automatizada, autoservicio, gobernada.                                           |
| señal de demanda         | Necesidad observada: una petición, una excepción, una salida del camino asfaltado, una acción de incidente o una respuesta de fricción.         |
| retirada                 | Eliminación de una capacidad que ya no se usa o que el proveedor resuelve. Libera la capacidad de mantener.                                     |
| dirección frente a fecha | Compromiso sobre qué se va a resolver y en qué orden, sin prometer días concretos para trabajo con incertidumbre alta.                          |
| capacidad de mantener    | Límite real del equipo: lo que ya sostiene. Cada capacidad nueva la consume permanentemente.                                                    |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. Capacidades, no funciones

Una hoja de ruta hecha de funciones —«un portal nuevo», «integrar tal herramienta»— no permite decidir. Una hecha de capacidades sí, porque se expresa desde quien la usa:

mal «integrar el gestor de secretos»  
 bien «un equipo puede obtener credenciales sin guardar ningún secreto»

mal «desplegar la malla de servicio»  
 bien «un equipo puede limitar quién llama a su servicio, sin escribir código»

Y cada capacidad tiene un nivel, que es lo que permite decir dónde se está:

|   |                    |                                                                                 |
|---|--------------------|---------------------------------------------------------------------------------|
| 0 | NO EXISTE          | nadie puede hacerlo                                                             |
| 1 | MANUAL             | se puede, pidiéndolo a alguien                                                  |
| 2 | AUTOMATIZADA       | hay una herramienta, y hay que pedir acceso o ayuda                             |
| 3 | AUTOSERVICIO       | el equipo lo hace solo, en minutos <span style="float: right;">clase 106</span> |
| 4 | GOBERNADA Y MEDIDA | además está medida, con objetivos y coste conocido                              |

Y con esa escala se puede describir el estado de la plataforma en una tabla, que es lo que hace posible priorizar:

| capacidad                                | nivel |           |
|------------------------------------------|-------|-----------|
| crear un servicio nuevo con todo montado | 3     | clase 106 |
| desplegar sin parada                     | 3     | clase 102 |
| obtener credenciales sin secretos        | 3     | clase 137 |
| saber cuánto cuesta mi servicio          | 1     | clase 142 |
| probar en un entorno con datos realistas | 2     | clase 104 |
| recuperar mi servicio en otra región     | 1     | clase 166 |
| saber qué políticas me aplican           | 2     | clase 170 |
| aislar a un cliente concreto             | 0     | clase 154 |

Y el catálogo de capacidades sale de lo que este programa ha ido construyendo:

|              |                                           |          |
|--------------|-------------------------------------------|----------|
| entrega      | construir, desplegar, revertir, activar   | parte 08 |
| datos        | almacenar, replicar, migrar esquemas      | parte 09 |
| operación    | observar, alertar, responder, ensayar     | parte 10 |
| seguridad    | identidad, secretos, políticas, evidencia | parte 11 |
| arquitectura | contratos, aislamiento, límites           | parte 12 |
| continuidad  | copias, recuperación, portabilidad        | parte 13 |
| gobierno     | cuentas, excepciones, atribución          | parte 14 |

Y la ventaja de esta forma sobre una lista de herramientas:

permite decir «estamos en nivel 1 en coste y en nivel 3 en despliegue»  
 permite comparar el hueco con el impacto  
 y sobrevive a cambiar de herramienta: la capacidad no cambia

### 2. Qué construir a continuación

La forma habitual de decidir —lo que le parece más importante a quien dirige la plataforma— produce plataformas vacías, que es el tercer modo de fallo de la clase 106.

Y las señales que sí son necesidades observadas:

PETICIONES REPETIDAS  
 las que llegan una y otra vez clase 106  
 → el 83 % eran las mismas tres cosas

EXCEPCIONES POR CONTROL

|                                                                                                                                                                           |           |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| cada una dice que algo no encaja                                                                                                                                          | clase 170 |
| SALIDAS DEL CAMINO ASFALTADO<br>quién no usa la plataforma y por qué                                                                                                      | clase 106 |
| ACCIONES DE REVISIÓN DE INCIDENTES<br>«esto habría hecho falta»                                                                                                           | clase 127 |
| RESPUESTAS A «¿QUÉ TE HA FRENADO?»<br>la pregunta abierta y periódica                                                                                                     | clase 107 |
| Y LO QUE LA ORGANIZACIÓN VA A NECESITAR<br>una norma que entra en vigor, un cliente que exige algo<br>→ la única señal que no es observada, y hay que declararla como tal |           |

Y la fórmula de prioridad, que hay que escribir aunque sea aproximada:

prioridad ≈ tiempo ahorrado por equipo × equipos afectados  
 – coste de construirlo  
 – COSTE DE MANTENERLO, para siempre

Y el tercer sumando es el que casi nadie incluye y el que decide a largo plazo:

cada capacidad consume capacidad de mantener  
 actualizaciones, incidencias, documentación, formación,  
 y compatibilidad cuando cambie lo de debajo  
 → un equipo de plataforma no está limitado por lo que puede construir,  
 sino por lo que ya sostiene

Y de ahí la regla que gobierna la hoja de ruta:

añadir una capacidad exige tener capacidad de mantener  
 → y si no la hay, hay que RETIRAR algo primero

Construir, comprar o adoptar se decide con la clasificación de la clase 147, aplicada a la plataforma:

|                                                                   |                                                    |           |
|-------------------------------------------------------------------|----------------------------------------------------|-----------|
| GENÉRICO                                                          | lo resuelve el proveedor o el mercado              |           |
| → adoptar tal cual; no envolver                                   |                                                    | clase 158 |
| → y el error frecuente es construir lo que el proveedor ya ofrece |                                                    |           |
| DE APOYO                                                          | hace falta y no diferencia                         |           |
| → lo más simple que funcione, o comprar                           |                                                    |           |
| DIFERENCIAL                                                       | la forma concreta en que esta organización trabaja |           |
| → construir; suele ser pegamento, no producto                     |                                                    |           |

Y lo que casi siempre resulta ser diferencial en una plataforma interna:

|                                                                |                 |
|----------------------------------------------------------------|-----------------|
| la forma de crear un servicio nuevo con CUANTO exige esta casa |                 |
| el catálogo y la atribución                                    | clases 095, 142 |
| la integración entre piezas de proveedores distintos           | clase 162       |
| y los caminos asfaltados propios                               | clase 106       |
| → es decir, la COMPOSICIÓN, no las piezas                      |                 |

### 3. La mitad que falta: retirar

Una hoja de ruta que solo añade es una hoja de ruta que se detiene sola en tres años, cuando el mantenimiento consume al equipo.

Lo que hay que retirar:

|                                                          |           |
|----------------------------------------------------------|-----------|
| capacidades que nadie usa                                |           |
| capacidades que el proveedor ha empezado a ofrecer mejor |           |
| herramientas duplicadas: dos formas de hacer lo mismo    |           |
| versiones antiguas de lo propio                          | clase 106 |
| y caminos asfaltados que ya no son el camino             |           |

Y para saber qué retirar hace falta medir el uso por capacidad:

equipos que la usan, y con qué frecuencia  
 cuándo fue la última vez  
 y cuánto cuesta mantenerla

Y el proceso, que es el contrato de versiones de la clase 106 aplicado a una capacidad entera:

1. anunciar con fecha, y decir qué la sustituye
2. dejar de admitir usos nuevos
3. acompañar a quien la use, con herramienta de migración

- si la plataforma pide migrar, la plataforma escribe la migración
- 4. cortes de prueba anunciados clase 118
- 5. retirar

Y una advertencia que evita el peor final:

- retirar sin sustituto obliga a cada equipo a resolverlo por su cuenta
- y entonces hay sesenta soluciones distintas, que es peor que la capacidad que se retiró
- se retira lo que no se usa, o lo que tiene un sustituto mejor

Y una categoría especial que conviene vigilar:

- LO QUE EL PROVEEDOR EMPIEZA A OFRECER
- cada año, algo que se construyó a mano pasa a estar disponible
- conviene revisar el catálogo propio contra lo que existe, una vez al año
- y retirar lo propio cuando lo del proveedor sea suficiente

Y el coste de no hacerlo:

- se mantiene una herramienta propia que hace peor lo que el proveedor hace bien, y se sigue pagando su mantenimiento indefinidamente
- es la ley 20 aplicada a la plataforma: existe porque nadie decidió que dejara de existir

#### 4. Prometer, financiar y medir

Qué se promete a los equipos, que es donde las plataformas pierden credibilidad:

- mal fechas concretas para trabajo con incertidumbre alta
- se incumplen, y a la tercera nadie cree la hoja de ruta
- bien DIRECCIONES Y ORDEN
- «lo siguiente que resolvemos es la atribución de coste, después el aislamiento por cliente»
- + fecha solo para lo que ya está empezado y acotado
- + y una fecha firme para lo que tiene una obligación externa

Y lo que sí conviene comprometer siempre:

- las RETIRADAS, con fecha
- porque afectan a otros y necesitan planificarse clase 106
- los cambios incompatibles, con aviso
- y los plazos de las obligaciones normativas

La financiación, que distorsiona más de lo que parece:

- PRESUPUESTO CENTRAL
- + puede invertir en lo que beneficia a todos
- y puede construir lo que nadie pidió
- COBRO A LOS EQUIPOS
- + solo sobrevive lo que alguien valora
- nadie paga lo transversal: seguridad, continuidad, gobierno
- y aparecen optimizaciones locales que empeoran el total ley 17
- MIXTO
- central para lo transversal y lo obligatorio
- cobrado para lo opcional y lo específico
- es lo que suele funcionar

Y el efecto de la segunda, que conviene anticipar:

- una plataforma financiada solo por cobro tiende a NO invertir en lo que beneficia a todos y nadie quiere pagar
- y eso es justamente lo transversal: identidad, continuidad, observabilidad común, gobierno

Las medidas del producto, que amplían las de la clase 106:

- ADOPCIÓN SIN OBLIGACIÓN, por capacidad
- NIVEL DE MADUREZ, y su evolución
- TIEMPO AHORRADO, estimado y publicado
- PROPORCIÓN DEL TIEMPO DEL EQUIPO EN PETICIONES
- por encima de un tercio, se está volviendo ventanilla clase 106
- CAPACIDADES RETIRADAS AL AÑO
- si es cero, la capacidad de mantener se está agotando

Y la penúltima es la que más dice sobre la salud del producto a largo plazo.

Y una comprobación anual que conviene hacer con los equipos, no sobre ellos:

recorrer el mapa de capacidades y preguntar por cada una  
¿la usáis? ¿os sirve? ¿qué le falta? ¿qué haríais si desapareciera?  
→ la última pregunta separa lo imprescindible de lo cómodo

Y la lista de comprobación de la clase:

- la plataforma se describe como capacidades, no como herramientas
- cada capacidad tiene un nivel de madurez declarado
- la prioridad sale de señales de demanda observadas
- la fórmula de prioridad incluye el coste de MANTENER
- añadir una capacidad exige tener capacidad de mantener
- está clasificado qué es genérico, de apoyo y diferencial
- no se construye lo que el proveedor ya ofrece bien
- se mide el uso por capacidad
- la hoja de ruta incluye retiradas, con fecha y sustituto
- se prometen direcciones y orden, no fechas para lo incierto
- las retiradas y los cambios incompatibles sí llevan fecha
- la financiación cubre lo transversal, no solo lo que alguien paga
- se mide adopción sin obligación, tiempo en peticiones y retiradas al año

Y el cierre que enlaza con la clase siguiente: decidir qué construir exige saber qué cuesta cada cosa y qué produce, y a esta escala eso ya no es una hoja de cálculo: es un modelo operativo con su gente y sus rutinas. Es la materia de la clase 172.

## Ejemplo trabajado

La plataforma de CloudShop sirve a sesenta equipos con seis personas. El ejercicio consiste en dibujar el mapa de capacidades, priorizar con señales medidas y descubrir qué hay que retirar para poder añadir.

El mapa, con niveles.

| capacidad                                | nivel |
|------------------------------------------|-------|
| crear un servicio nuevo con todo montado | 3     |
| desplegar sin parada y revertir          | 3     |
| obtener credenciales sin secretos        | 3     |
| entorno de pruebas con datos             | 2     |
| observabilidad estándar                  | 3     |
| objetivos y presupuesto de error         | 2     |
| saber cuánto cuesta mi servicio          | 1     |
| recuperar mi servicio en otra región     | 1     |
| aislar a un cliente concreto             | 0     |
| saber qué políticas me aplican           | 2     |
| migrar un esquema sin parada             | 1     |
| crear una cuenta nueva                   | 3     |

Y el hueco medio ponderado por uso:

|                                   |         |
|-----------------------------------|---------|
| capacidades en nivel 3 o superior | 5 de 12 |
| capacidades en nivel 0 o 1        | 4 de 12 |

Las señales de demanda, contadas durante un trimestre.

| señal                                                                  | veces | capacidad implicada   |
|------------------------------------------------------------------------|-------|-----------------------|
| peticiones «¿cuánto cuesta esto?»                                      | 41    | coste por servicio    |
| excepciones al control de tipos de instancia                           | 22    | cargas con acelerador |
| peticiones de entorno con datos reales                                 | 19    | entorno de pruebas    |
| acciones de incidente sobre recuperación                               | 14    | recuperación          |
| respuestas «me frena no poder probar con datos parecidos a producción» | 12    | entorno de pruebas    |
| peticiones de aislamiento para un cliente                              | 6     | aislamiento           |
| salidas del camino asfaltado                                           | 4     | varias                |

Y la prioridad calculada, con el coste de mantener incluido:

| capacidad                 | ahorro/equipo/mes | equipos | construir | mantener/año |
|---------------------------|-------------------|---------|-----------|--------------|
| coste por servicio        | 2,5 h             | 60      | 6 sem     | 0,3 pers     |
| entorno con datos         | 4,0 h             | 31      | 8 sem     | 0,5 pers     |
| recuperación autoservicio | 1,0 h             | 60      | 12 sem    | 0,4 pers     |

|                         |       |   |        |          |
|-------------------------|-------|---|--------|----------|
| aislamiento por cliente | 6,0 h | 4 | 10 sem | 0,4 pers |
|-------------------------|-------|---|--------|----------|

Y el orden que sale:

1. coste por servicio 150 h/mes ahorradas, 6 semanas de trabajo
2. entorno con datos 124 h/mes, 8 semanas
3. recuperación 60 h/mes, 12 semanas
4. aislamiento 24 h/mes, 10 semanas → se pospone

Y el aislamiento, que era el nivel 0 y parecía urgente, quedó el último porque afecta a cuatro equipos. Se resolvió a mano para esos cuatro mientras tanto.

La capacidad de mantener, y lo que hubo que retirar.

|                                  |                |           |
|----------------------------------|----------------|-----------|
| personas en la plataforma        | 6              |           |
| tiempo en peticiones             | 21 %           | clase 106 |
| tiempo en mantener lo existente  | 54 %           |           |
| tiempo disponible para construir | 25 %           |           |
|                                  | ≈ 1,5 personas |           |

Y con un cuarto del tiempo disponible, las tres capacidades prioritarias sumaban veintiséis semanas: más de un año.

La conversación pasó a ser qué retirar.

|                                  |           |
|----------------------------------|-----------|
| inventario de lo que se mantiene | 31 piezas |
| uso medido                       |           |
| usadas por más de 10 equipos     | 12        |
| usadas por 2 a 10                | 9         |
| usadas por 1                     | 6         |
| sin uso en 6 meses               | 4         |

Y las decisiones:

RETIRAR SIN SUSTITUTO (4 piezas sin uso)  
 un panel antiguo, un generador de informes, dos guiones  
 liberado 0,3 personas

RETIRAR PORQUE EL PROVEEDOR YA LO OFRECE (3 piezas)  
 un sistema propio de rotación de certificados  
 un recolector de métricas hecho a medida  
 y una herramienta de inventario  
 → los tres existían desde antes de que el proveedor los tuviera  
 liberado 0,7 personas

UNIFICAR DUPLICADOS (2 pares)  
 dos formas de crear un servicio, heredadas de una adquisición  
 dos bibliotecas de registro  
 liberado 0,4 personas

MANTENER LAS 6 DE UN SOLO EQUIPO  
 → se ofrecieron al equipo que las usa; 4 las asumieron,  
 2 se retiraron con su acuerdo  
 liberado 0,3 personas

|                                  |               |               |
|----------------------------------|---------------|---------------|
| piezas mantenidas                | 31            | 20            |
| tiempo en mantener               | 54 %          | 33 %          |
| tiempo disponible para construir | 25 %          | 46 %          |
|                                  | ≈1,5 personas | ≈2,8 personas |

Casi el doble de capacidad de construir sin contratar a nadie, y el trabajo priorizado pasó de más de un año a unos siete meses.

La retirada del sistema propio de certificados, en detalle.

|                                                            |              |
|------------------------------------------------------------|--------------|
| construido hacía 3 años, cuando el proveedor no lo ofrecía |              |
| equipos que lo usaban                                      | 38           |
| coste de mantenerlo                                        | 0,3 personas |
| incidencias causadas por él en 12 meses                    | 4            |

proceso

|                                                        |           |
|--------------------------------------------------------|-----------|
| se anunció con 4 meses y con el sustituto identificado |           |
| se escribió la herramienta de migración                | clase 106 |
| equipos migrados en el primer mes                      | 9         |
| tras el primer corte de prueba anunciado               | 24        |
| tras el segundo                                        | 38        |
| incidencias durante la migración                       | 1         |

Y la comprobación que lo motivó, que ahora es anual:

|                                                              |   |
|--------------------------------------------------------------|---|
| revisar el catálogo propio contra lo que ofrece el proveedor |   |
| piezas candidatas encontradas el primer año                  | 3 |
| el segundo año                                               | 2 |

Lo que se prometió, y lo que no.

|                                                        |                    |
|--------------------------------------------------------|--------------------|
| se comprometió con fecha                               |                    |
| la retirada del sistema de certificados                | 4 meses            |
| el cambio incompatible de la plantilla de canalización | 6 semanas de aviso |
| el cumplimiento de un requisito normativo              | fecha legal        |

se comprometió como dirección, sin fecha  
coste por servicio, entorno con datos y recuperación,  
en ese orden

|                                                      |   |
|------------------------------------------------------|---|
| fechas incumplidas en 12 meses                       | 0 |
| → antes de este cambio: 4 de 7 compromisos con fecha |   |

La financiación.

|                                                                 |                     |                   |
|-----------------------------------------------------------------|---------------------|-------------------|
| modelo                                                          | presupuesto central | mixto             |
| lo transversal                                                  | central             | central           |
| identidad, continuidad, observabilidad común, gobierno          |                     |                   |
| lo opcional y específico                                        | —                   | cobrado al equipo |
| entornos adicionales, capacidad reservada, herramientas propias |                     |                   |

|                                                  |        |
|--------------------------------------------------|--------|
| efecto observado en 12 meses                     |        |
| entornos de pruebas sin uso                      | 14 → 2 |
| peticiones de capacidad reservada «por si acaso» | 9 → 1  |

Y lo que se decidió NO cobrar, y por qué:

la observabilidad común y la continuidad  
→ si se cobraran, los equipos con menos presupuesto las reducirían  
→ y son justamente las que protegen a toda la organización

Las medidas del producto, a los doce meses.

|                                           |         |            |
|-------------------------------------------|---------|------------|
| capacidades en nivel 3 o superior         | 5 de 12 | 8 de 12    |
| capacidades en nivel 0 o 1                | 4 de 12 | 1 de 12    |
| adopción sin obligación, media            | 71 %    | 89 %       |
| tiempo del equipo en peticiones           | 21 %    | 14 %       |
| tiempo en mantener                        | 54 %    | 33 %       |
| tiempo en construir                       | 25 %    | 53 %       |
| piezas mantenidas                         | 31      | 20         |
| capacidades retiradas                     | 0       | 9          |
| tiempo ahorrado a los equipos, estimado   | —       | ~290 h/mes |
| compromisos con fecha incumplidos         | 4 de 7  | 0 de 3     |
| coste de la plataforma por equipo servido | —       | medido     |

La lección que esta clase traslada a la parte 14: la plataforma no estaba limitada por lo que sabía construir, sino por lo que ya sostenía: el 54 % del tiempo se iba en mantener treinta y una piezas, cuatro de ellas sin ningún uso y tres que el proveedor ya ofrecía mejor. Retirar nueve capacidades dobló la capacidad de construir sin contratar a nadie, y el trabajo prioritario pasó de más de un año a siete meses. Y la capacidad que parecía más urgente —el aislamiento por cliente, único nivel 0— quedó la última al multiplicar el ahorro por el número de equipos afectados: cuatro.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/171-platform-as-a-product-y-roadmap-de-capacidades/lab.py
```

El laboratorio selecciona el motor de práctica platform y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.

4. Materializa roadmap-plataforma en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

## Evidencia esperada

El artefacto principal es una capacidad autoservicio con contrato y golden path. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye roadmap-plataforma para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

## ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■ Errores frecuentes

| Síntoma                                                             | Causa probable                                                                             | Corrección                                                                                                               |
|---------------------------------------------------------------------|--------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| La hoja de ruta es una lista de herramientas y no permite priorizar | Está escrita como funciones y no como capacidades desde el punto de vista de quien las usa | Describe capacidades con nivel de madurez y compara el hueco con el impacto.                                             |
| Se construye lo que nadie pidió                                     | La prioridad la fija quien dirige la plataforma en vez de señales observadas               | Cuenta peticiones repetidas, excepciones, salidas del camino, acciones de incidente y respuestas de fricción.            |
| El equipo no tiene tiempo de construir nada nuevo                   | El mantenimiento de lo existente consume la mayor parte                                    | Mide el uso por capacidad y retira lo que no se usa, lo duplicado y lo que el proveedor ya ofrece mejor.                 |
| Se mantiene una herramienta propia peor que la del proveedor        | Ley 20: existe porque nadie decidió que dejara de existir                                  | Revisa una vez al año el catálogo propio contra lo que ofrece el proveedor y retira lo que sobre, con migración escrita. |
| Nadie cree la hoja de ruta                                          | Se prometieron fechas para trabajo con incertidumbre alta                                  | Compromete direcciones y orden; deja las fechas para retiradas, cambios incompatibles y obligaciones externas.           |
| No se invierte en seguridad, continuidad ni observabilidad comunes  | La plataforma se financia solo por cobro y nadie paga lo transversal                       | Modelo mixto: central para lo transversal y obligatorio, cobrado para lo opcional y específico.                          |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Qué diferencia hay entre describir la plataforma por funciones y por capacidades?
2. ¿Qué señales de demanda son observaciones y cuál no lo es?

3. ¿Por qué la fórmula de prioridad debe incluir el coste de mantener?
4. ¿Qué se retira y qué proceso necesita una retirada?
5. ¿Qué distorsiona financiar la plataforma solo con cobro a los equipos?

## Referencias

- CNCF (2025). Platform engineering maturity model — capacidades y niveles de madurez.  
<<https://tag-app-delivery.cncf.io/whitepapers/platform-eng-maturity-model/>>
- CNCF (2025). Platforms white paper — la plataforma como producto y sus capacidades.  
<<https://tag-app-delivery.cncf.io/whitepapers/platforms/>>
- Skelton, M. y Pais, M. (2019). Team Topologies, cap. 5 — capacidad de mantener y carga del equipo de plataforma.  
<<https://teamtopologies.com/book>>
- Cagan, M. (2017). Inspired — hoja de ruta por resultados en vez de por funciones y fechas.  
<<https://www.svpg.com/inspired-how-to-create-products-customers-love/>>
- Thoughtworks (2025). Platform funding models — presupuesto central, cobro y modelos mixtos.  
<<https://www.thoughtworks.com/insights/blog/platforms>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                            | Índice              | Siguiente                                           |
|-----------------------------------------------------|---------------------|-----------------------------------------------------|
| ← 170 · Gobierno federado y policy as code a escala | Parte 14 · Programa | 172 · Modelo operativo FinOps y economía unitaria → |

## Evaluación — 171 Platform as a Product y roadmap de capacidades

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega roadmap-plataforma con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- ☐ Resultado reproducible por una segunda persona.
- ☐ Evidencia vinculada a cada afirmación técnica.
- ☐ Prueba negativa o de fallo incluida.
- ☐ Costos y permisos expresados con unidades y alcance.
- ☐ Limitaciones declaradas sin presentar la simulación como producción.

## Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 172 — Modelo operativo FinOps y economía unitaria

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: finops · Estado: EXECUTABLECORE

## Propósito

Convertir lo de las clases 142 y 143 en algo que funcione con sesenta equipos y no dependa de una persona. La clase reparte los tres papeles que hacen falta, fija el ritmo —qué se mira a diario, a la semana, al mes y al trimestre—, y desarrolla el lenguaje que permite hablar de coste con quien no es ingeniero: la economía unitaria, con su parte difícil, que es repartir lo compartido de forma que alguien pueda entenderlo. Y termina con los dos finales conocidos de estos programas: la policía del gasto y el panel que nadie mira.

## Resultados de aprendizaje

Al finalizar podrás:

1. Repartir los tres papeles del modelo operativo y saber qué decide cada uno.
2. Establecer un ritmo con cadencias distintas, en vez de un proyecto.
3. Construir el coste por unidad, incluidos los compartidos.
4. Prever el gasto a partir de la actividad del negocio, no de la factura anterior.
5. Evitar que el programa acabe siendo policía o decoración.

## Conceptos centrales

| Concepto                  | Comprensión verificable                                                                                                          |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| modelo operativo          | Reparto de papeles, decisiones y cadencias que hace que el gobierno del coste ocurra sin depender de una persona.                |
| coste por unidad          | Coste dividido entre una magnitud del negocio. Es el único lenguaje común entre ingeniería, finanzas y negocio.                  |
| coste de servir           | Todo lo que cuesta atender a un cliente: infraestructura, plataforma, soporte y licencias. Va más allá de la factura de la nube. |
| previsión por generadores | Prever la actividad del negocio y multiplicarla por el coste unitario, en vez de extrapolar la factura.                          |
| policía del gasto         | Modo de fallo en que el programa se dedica a señalar culpables. Produce rodeos, no ahorro.                                       |
| coste en la decisión      | Poner la cifra donde se decide —el cambio propuesto, la revisión de diseño— en vez de en un informe posterior.                   |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## Flujo de razonamiento



## Desarrollo

### 1. Tres papeles, y por qué no basta una persona

El error inicial de casi todas las organizaciones es nombrar a alguien responsable del coste de la nube. Y falla porque las decisiones que cuestan dinero están repartidas:

- quien elige una réplica de más
- quien fija una retención de 90 días
- quien deja un entorno encendido
- quien decide la clave de partición
- y quien firma un compromiso a tres años
- son decenas de personas, y ninguna es esa

Y los tres papeles que sí hacen falta:

#### INGENIERÍA

- toma las decisiones que generan coste
- necesita: ver el coste de lo suyo, en su unidad, y en el momento de decidir
- decide: arquitectura, dimensionado, retención, aprovechamiento

#### FINANZAS

- posee el presupuesto, la previsión y los contratos
- necesita: atribución fiable y una previsión creíble
- decide: compromisos, presupuestos por área, negociación

#### NEGOCIO

- posee la unidad que da sentido al coste
- necesita: coste por pedido, por cliente, por transacción
- decide: precios, qué clientes se atienden y con qué nivel

Y el papel que a veces existe y conviene entender bien:

- un equipo o persona que FACILITA
- construye la atribución, los paneles y las herramientas
- mantiene el ritmo
- y traduce entre los tres
- facilita; NO decide por ellos
- si decide, se convierte en la policía del apartado cuarto

Y la condición previa de todo, que la clase 142 ya estableció:

- sin atribución fiable no hay modelo operativo
- y la atribución se impone en la creación, no se pide
- clases 142, 169

Y una cifra que conviene tener para dimensionar el esfuerzo:

- el gobierno del coste no es gratis
- facilitación: 0,5-1 persona por cada ~50 equipos
- tiempo de los equipos: 1-2 h al mes cada uno
- y eso hay que compararlo con lo que ahorra, igual que en la clase 170

### 2. El ritmo

Sin cadencias, esto es un proyecto que termina y una factura que vuelve a crecer. Lo que se mira, y cuándo:

#### DIARIO – automático

- desviaciones por servicio, comparadas consigo mismas      clase 142
- con aviso al equipo dueño y enlace a la línea de cambios
- detecta en 1-2 días lo que antes se veía en la factura

#### SEMANAL – cada equipo, 10 minutos

- su gasto, su tendencia y su coste por unidad
- y las tres partidas que más han subido
- en la misma reunión donde ya miran sus objetivos      clase 126

#### MENSUAL – la organización

- coste por unidad, global y por servicio
- previsión frente a real, y por qué difieren
- bolsa sin asignar y su tendencia
- y las tres oportunidades mayores del trimestre

#### TRIMESTRAL – decisiones que cuestan tiempo

compromisos: renovar, ampliar o dejar vencer                  clase 143  
dimensionado y capacidad  
y revisión de lo que sobra: huérfanos, entornos, cuentas  
                                              clases 169, 171

ANUAL – decisiones grandes  
negociación de contratos  
decisiones de arquitectura con impacto de coste                      clase 155  
y revisión de la unidad elegida: ¿sigue siendo la buena?

Y las dos cadencias que más rinden son las dos primeras, por motivos distintos:

la diaria porque reduce el tiempo de detección de semanas a un día  
la semanal porque pone la cifra delante de quien decide, cada semana,  
sin que nadie tenga que pedirlo

Y una regla sobre la reunión mensual, que es donde estos programas mueren:

- si la reunión mensual repasa números sin tomar decisiones, se convierte en un informe y deja de venir gente
- cada reunión sale con acciones, con dueño y con fecha
- y empieza revisando las de la anterior

Y lo que no debe estar en el ritmo:

- revisión individual de equipos «que gastan mucho»
  - gastar mucho no es un problema si el coste por unidad es bueno
  - y esa reunión es el primer paso hacia la policía del gasto

### 3. La economía unitaria

Es lo que permite hablar con quien no es ingeniero, porque el coste absoluto sube cuando el negocio va bien y no dice nada.

Elegir la unidad, que es la primera decisión:

- buena unidad
  - la entiende el negocio sin traducción
  - crece con la actividad
  - y se puede medir con fiabilidad

ejemplos por pedido, por transacción, por cliente activo,  
por documento procesado, por hora de vídeo, por consulta

mala unidad  
«por usuario registrado», si muchos no usan nada  
«por servidor», que es una medida de la solución, no del negocio

Y conviene tener dos o tres, no una:

- una global para el conjunto
- y una por línea de negocio o por tipo de cliente
- porque un coste medio esconde clientes que cuestan diez veces más

Las tres capas del coste, que hay que separar:

1. DIRECTO ATRIBUIBLE  
lo que se puede asignar sin discusión: su cómputo, su base
2. COMPARTIDO REPARTIDO  
red, clúster, observabilidad, plataforma  
→ con un método explicable, no perfecto clase 142
3. COSTE DE SERVIR COMPLETO  
lo anterior más soporte, licencias por cliente y el tiempo de la gente que lo atiende  
→ es el que necesita negocio para decidir precios

Y la tercera es la que casi nunca se calcula y la que cambia decisiones:

un cliente puede tener un coste de infraestructura pequeño  
y consumir cuatro horas de soporte al mes  
→ y entonces no es rentable, aunque su factura de nube sea baja

La previsión, que es lo que finanzas necesita y donde más se falla:

- mal extrapolar la factura del mes anterior  
→ no distingue crecimiento de ineficiencia

→ y falla en cuanto hay una campaña o una migración

bien POR GENERADORES

negocio prevé la actividad: pedidos, clientes, transacciones

ingeniería aporta el coste por unidad y su tendencia

previsión = actividad prevista × coste unitario

+ cambios conocidos (migraciones, compromisos)

Y su ventaja, que es la que convence a finanzas:

cuando la previsión falla, se puede saber POR QUÉ

¿la actividad fue distinta a lo previsto?

¿o el coste por unidad se movió?

→ y son dos conversaciones distintas, con dueños distintos

Y una recomendación práctica: publicar la previsión con su banda, no con una cifra. Una previsión sin incertidumbre declarada se trata como un compromiso y se incumple.

#### 4. Los dos finales conocidos

La policía del gasto. El programa empieza a señalar equipos que gastan mucho.

síntomas

reuniones donde se explica por qué se ha gastado

comparaciones entre equipos con contextos distintos

y objetivos de reducción impuestos sin contexto

efecto

se optimiza lo visible y se esconde lo demás

ley 17

se dejan de pedir recursos que hacían falta

y aparecen decisiones peores que cuestan menos en la nube

y más en tiempo de personas

Y lo que lo evita:

hablar SIEMPRE en coste por unidad, no en coste absoluto

comparar cada equipo consigo mismo, no con otros

clase 107

y recordar que el objetivo no es gastar menos: es gastar bien

Y la formulación que funciona:

mal «tu equipo gastó 12.000 € este mes»

bien «tu coste por pedido subió de 0,038 € a 0,052 € desde el día 12;

coincide con este despliegue»

El panel que nadie mira. El otro final:

síntomas

un panel completo, con todo desglosado

y ninguna decisión tomada a partir de él en seis meses

Y lo que lo evita es lo que la clase 142 ya señaló:

poner la cifra DONDE SE DECIDE

en el cambio propuesto: «esto añade 310 €/mes»

en la revisión de diseño: coste por unidad esperado

en el panel del servicio, junto al objetivo

clase 125

y en la ficha del catálogo

clase 095

→ el informe mensual es para la organización; la decisión

ocurre en otro sitio

La madurez, para saber por dónde se va:

PRIMERO atribución y visibilidad

etiquetas impuestas, gasto por equipo, desviaciones diarias

DESPUÉS economía unitaria y optimización

coste por unidad, escalera de la clase 143, compromisos

AL FINAL coste como entrada de diseño

una de las cuatro columnas de toda decisión

clase 155

y previsión por generadores

Y el error de saltarse el primero:

optimizar sin atribuir produce ahorros que nadie sostiene

→ y a los seis meses el gasto ha vuelto

Y las medidas del propio modelo operativo:

proporción del gasto atribuida  
 coste por unidad, y su tendencia  
 error de la previsión mensual, y su descomposición  
 tiempo desde que empieza una desviación hasta que se actúa  
 decisiones tomadas a partir de datos de coste, contadas  
 horas que los equipos dedican a esto  
 y cuántos equipos miran su coste sin que se lo pidan

La última es la que dice si el modelo está vivo.

Y la lista de comprobación de la clase:

- los tres papeles están asignados, con lo que decide cada uno
- quien facilita no decide por los demás
- la atribución está impuesta en la creación
- hay detección diaria de desviaciones con aviso al dueño
- cada equipo ve su gasto en su reunión semanal
- la reunión mensual sale con acciones, dueño y fecha
- hay revisión trimestral de compromisos y de lo que sobra
- existen dos o tres unidades de negocio elegidas y medidas
- el coste de servir incluye soporte, licencias y personas
- la previsión se hace por generadores y con banda
- el coste aparece en el cambio propuesto y en el panel del servicio
- se habla en coste por unidad, y cada equipo se compara consigo mismo
- se cuentan las decisiones tomadas a partir de datos de coste

Y el cierre que enlaza con la clase siguiente: el coste es una de las cuatro columnas; la fiabilidad es otra, y a esta escala también necesita un modelo operativo y una forma de saber en qué punto está la organización. Es la materia de la clase 173.

## Ejemplo trabajado

CloudShop tiene sesenta equipos, la atribución resuelta y una persona dedicada al coste que se ha convertido en el cuello de botella. El ejercicio monta el modelo operativo y descubre dos cosas: que hay clientes que cuestan más de lo que pagan y que la previsión fallaba por el motivo equivocado.

El punto de partida.

|                                                     |         |
|-----------------------------------------------------|---------|
| personas dedicadas al coste                         | 1       |
| peticiones que recibía al mes                       | 41      |
| decisiones que tomaba por otros                     | 31      |
| equipos que miraban su gasto sin que se lo pidieran | 4 de 60 |
| error de la previsión mensual, media                | ±19 %   |
| reuniones mensuales de coste                        | sí      |
| acciones salidas de esas reuniones en 6 meses       | 3       |

Una persona decidiendo por sesenta equipos, y una reunión mensual que producía media acción al mes.

El reparto de papeles.

|                                       |                   |                                                      |
|---------------------------------------|-------------------|------------------------------------------------------|
| quién decide dimensionado y retención | la persona        | cada equipo                                          |
| quién decide compromisos              | la persona        | finanzas, con datos de ingeniería                    |
| quién decide precios y niveles        | nadie             | negocio                                              |
| qué hace la persona dedicada          | decidir por otros | facilitar: atribución, paneles, herramientas y ritmo |
| peticiones que recibe al mes          | 41                | 6                                                    |
| decisiones que toma por otros         | 31                | 0                                                    |

El ritmo, y el efecto de cada cadencia.

### DIARIO

ya existía desde la clase 142  
 tiempo de detección de una desviación 1,2 días

### SEMANAL

se añadió al orden del día de la reunión de cada equipo,  
 junto a sus objetivos  
 duración 10 min  
 equipos que la mantienen a los 6 meses 54 de 60  
 decisiones tomadas en esas reuniones, 6 meses 71

#### MENSUAL

cambio de formato: empieza revisando las acciones anteriores  
acciones por reunión 0,5 → 4,2  
asistencia cayendo → estable

#### TRIMESTRAL

compromisos, dimensionado y limpieza  
ahorro atribuible en 12 meses 6.900 €/mes

#### ANUAL

negociación con dos proveedores y revisión de la unidad

Y las setenta y una decisiones de las reuniones semanales son la cifra que más dice: decisiones pequeñas, tomadas por quien puede tomarlas, cerca del momento.

La unidad, y los tres intentos.

intento 1 «coste por servidor»  
→ nadie del negocio la entendía; media la solución

intento 2 «coste por usuario registrado»  
→ 340.000 registrados, 31.000 activos  
→ la cifra bajaba sola al registrar gente que no usaba nada

intento 3 «coste por pedido» y «coste por cliente activo al mes»  
→ las dos las entiende el negocio y crecen con la actividad

Y las tres capas, calculadas:

|                          |          |          |
|--------------------------|----------|----------|
| directo atribuible       | 0,029 €  | 1,90 €   |
| compartido repartido     | 0,009 €  | 0,60 €   |
| plataforma               | 0,003 €  | 0,20 €   |
|                          | ■■■■■■■■ | ■■■■■■■■ |
| coste de infraestructura | 0,041 €  | 2,70 €   |
| soporte humano           | —        | 1,10 €   |
| licencias por cliente    | —        | 0,80 €   |
|                          |          | ■■■■■■■■ |
| coste de servir          |          | 4,60 €   |

Los clientes que no eran rentables.

Con el coste de servir por cliente, y cruzándolo con lo que paga cada uno:

|                                          |             |           |
|------------------------------------------|-------------|-----------|
| clientes empresariales                   | 190         |           |
| coste de servir medio                    | 4,60 €/mes  |           |
| coste de servir, percentil 90            | 18,40 €/mes |           |
| clientes que cuestan más de lo que pagan | 14          |           |
| por consumo desproporcionado             | 9           | clase 154 |
| por soporte: más de 3 h/mes              | 3           |           |
| por precio mal fijado desde 2022         | 2           |           |

Y las tres decisiones que se tomaron, ninguna técnica:

|                  |                                            |           |
|------------------|--------------------------------------------|-----------|
| los 9 de consumo | cuotas y conversación comercial            | clase 154 |
| los 3 de soporte | se investigó por qué llamaban tanto        |           |
|                  | → dos usaban mal una función; se rehízo    |           |
|                  | su interfaz y las llamadas bajaron un 80 % |           |
| los 2 de precio  | renegociados al renovar                    |           |

Y el hallazgo del segundo grupo es el más interesante: el coste de soporte era un síntoma de un problema de producto, y solo se vio al incluirlo en el coste de servir.

La previsión, y por qué fallaba.

|                 |                                        |       |
|-----------------|----------------------------------------|-------|
| método anterior | extrapolar la factura del mes anterior |       |
| error medio     |                                        | ±19 % |

descomposición de un mes con error del 24 %

|                                                               |                                           |         |
|---------------------------------------------------------------|-------------------------------------------|---------|
| ¿fue la actividad?                                            | pedidos previstos 290.000, reales 310.000 | (+7 %)  |
| ¿fue el coste unitario?                                       | previsto 0,041 €, real 0,047 €            | (+15 %) |
| → dos causas distintas, y el método antiguo no las distinguía |                                           |         |

Y con previsión por generadores:

| método                        | extrapolación     | actividad × unitario<br>+ cambios conocidos                       |
|-------------------------------|-------------------|-------------------------------------------------------------------|
| error medio                   | ±19 %             | ±6 %                                                              |
| se publica con banda          | no                | sí                                                                |
| se puede descomponer el error | no                | sí                                                                |
| conversaciones sobre el error | «gastamos de más» | «la actividad subió<br>un 7 % y el unitario<br>un 15 %, por esto» |

Y el 15 % del coste unitario de ese mes se rastreó en cuarenta minutos hasta un despliegue concreto, gracias a la línea de cambios de la clase 121.

Los dos finales evitados.

#### POLICÍA DEL GASTO

la reunión mensual incluía una tabla de «los cinco que más gastan»  
→ se retiró: el que más gastaba tenía el mejor coste por pedido  
→ sustituida por «los cinco cuyo coste unitario más ha subido»

efecto en 6 meses

|                                                                                                 |       |
|-------------------------------------------------------------------------------------------------|-------|
| equipos que dejaron de pedir recursos por miedo                                                 | 3 → 0 |
| decisiones peores para reducir factura                                                          | 2 → 0 |
| → un equipo había quitado una réplica de lectura y<br>había empeorado su latencia y su objetivo |       |

#### PANEL QUE NADIE MIRA

|                                                        |   |
|--------------------------------------------------------|---|
| paneles de coste existentes                            | 9 |
| abiertos alguna vez en 90 días                         | 2 |
| → se retiraron 7 y la cifra se llevó a donde se decide |   |

|                                                   |           |
|---------------------------------------------------|-----------|
| coste en el cambio propuesto                      | clase 142 |
| cambios con estimación                            | 211       |
| cambios modificados tras verla                    | 19        |
| coste en el panel del servicio, junto al objetivo | sí        |
| coste en la ficha del catálogo                    | sí        |

A los doce meses.

|                                            |             |                  |
|--------------------------------------------|-------------|------------------|
| papeles asignados                          | 1           | 3 + facilitación |
| decisiones tomadas por la persona dedicada | 31          | 0                |
| equipos que miran su gasto sin pedirselo   | 4 de 60     | 54 de 60         |
| acciones por reunión mensual               | 0,5         | 4,2              |
| unidades de negocio medidas                | 0           | 2                |
| coste de servir calculado                  | no          | sí               |
| clientes que cuestan más de lo que pagan   | no se sabía | 14 → 3           |
| error de la previsión mensual              | ±19 %       | ±6 %             |
| previsión descomponible                    | no          | sí               |
| paneles de coste                           | 9           | 2                |
| decisiones tomadas con datos de coste      | 3 / 6 meses | 71 / 6 meses     |
| coste por pedido                           | 0,057 €     | 0,038 €          |
| horas de los equipos dedicadas a esto      | no se medía | 1,3 h/mes        |

La lección que esta clase traslada a la parte 14: una persona tomaba treinta y una decisiones al mes por sesenta equipos, y la reunión mensual producía media acción. Al repartir los papeles y añadir diez minutos semanales en las reuniones que ya existían, las decisiones pasaron a setenta y una en seis meses, tomadas por quien podía tomarlas. Y el hallazgo que cambió el negocio no vino de la factura de la nube: al incluir el soporte en el coste de servir aparecieron tres clientes cuyo problema no era el consumo sino una función mal diseñada que les hacía llamar cuatro veces al mes.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/172-modelo-operativo-finops-y-economia-unitaria/lab.py
```

El laboratorio selecciona el motor de práctica finops y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.

3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa operating-model-finops en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

## Evidencia esperada

El artefacto principal es un cálculo trazable con unidad, supuesto y sensibilidad. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye operating-model-finops para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

## ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■ ■ Errores frecuentes

| Síntoma                                                         | Causa probable                                                                | Corrección                                                                                                                                         |
|-----------------------------------------------------------------|-------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------|
| Una sola persona decide sobre el coste de toda la organización  | Las decisiones que cuestan dinero están repartidas y no se pueden centralizar | Reparte los tres papeles; quien facilita construye herramientas y mantiene el ritmo, pero no decide por los equipos.                               |
| El programa de coste se apaga a los seis meses                  | Era un proyecto, no un ritmo                                                  | Cadencias distintas: diaria automática, semanal en la reunión que ya existe, mensual con acciones, trimestral de compromisos y anual de contratos. |
| La reunión mensual repasa números y no cambia nada              | No sale con acciones ni revisa las anteriores                                 | Empieza revisando las acciones previas y termina con dueño y fecha para cada una nueva.                                                            |
| La previsión falla y no se sabe por qué                         | Se extrapola la factura anterior, que mezcla crecimiento con eficiencia       | Prevé la actividad del negocio y multiplícala por el coste unitario; publica con banda y descompón el error.                                       |
| Los equipos dejan de pedir recursos que necesitan               | El programa se ha convertido en policía del gasto                             | Habla en coste por unidad, compara cada equipo consigo mismo y retira las tablas de quién gasta más.                                               |
| Hay paneles de coste completos y ninguna decisión sale de ellos | La cifra está en un informe y no donde se decide                              | Lleva el coste al cambio propuesto, a la revisión de diseño, al panel del servicio y a la ficha del catálogo.                                      |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Por qué no basta con nombrar a un responsable del coste?
2. ¿Qué se mira en cada cadencia y cuáles dos rinden más?
3. ¿Qué tres capas componen el coste por unidad y cuál se olvida?
4. ¿Qué ventaja tiene prever por generadores frente a extrapolar la factura?
5. ¿Cuáles son los dos finales conocidos de estos programas y qué los evita?

## Referencias

- FinOps Foundation (2025). FinOps framework: personas, phases and capabilities — papeles y ciclo del modelo operativo. <<https://www.finops.org/framework/>>
- FinOps Foundation (2025). Unit economics and forecasting — coste por unidad y previsión por generadores. <<https://www.finops.org/framework/capabilities/forecasting/>>
- Storment, J. R. y Fuller, M. (2023). Cloud FinOps, caps. 6-9 — cadencias, informar frente a cobrar y madurez. <<https://www.oreilly.com/library/view/cloud-finops-2nd/9781492098348/>>
- AWS (2025). Cost management operating model — atribución, previsión y gobierno del gasto. <<https://docs.aws.amazon.com/whitepapers/latest/how-aws-pricing-works/aws-cost-management-tools.html>>
- Google Cloud (2025). Cost governance and unit cost metrics — métricas unitarias y su uso en decisiones. <<https://cloud.google.com/architecture/framework/cost-optimization>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                               | Índice              | Siguiente                                          |
|--------------------------------------------------------|---------------------|----------------------------------------------------|
| ← 171 · Platform as a Product y roadmap de capacidades | Parte 14 · Programa | 173 · Madurez SRE y confiabilidad organizacional → |

## Evaluación — 172 Modelo operativo FinOps y economía unitaria

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega operating-model-finops con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- [ ] Resultado reproducible por una segunda persona.
- [ ] Evidencia vinculada a cada afirmación técnica.
- [ ] Prueba negativa o de fallo incluida.
- [ ] Costos y permisos expresados con unidades y alcance.

- [ ] Limitaciones declaradas sin presentar la simulación como producción.

## Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 173 — Madurez SRE y confiabilidad organizacional

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: sre · Estado: EXECUTABLECORE

## Propósito

Organizar la fiabilidad cuando hay sesenta equipos, que es una pregunta distinta de la de la parte 10: no cómo se opera un sistema, sino quién responde por su fiabilidad y cómo decide la organización. La clase compara los tres modelos con sus modos de fallo, se detiene en el más común —el equipo central que acaba operando los sistemas de todos— y desarrolla dos cosas que solo existen a esta escala: el análisis entre incidentes, que encuentra la causa que aparece en once revisiones y que ninguna revisión individual puede ver, y la conversación con el negocio sobre fiabilidad frente a velocidad, que sin datos siempre la gana quien habla más alto.

## Resultados de aprendizaje

Al finalizar podrás:

1. Elegir el modelo organizativo de fiabilidad y reconocer su modo de fallo.
2. Impedir que un equipo central se convierta en la operación de los demás.
3. Situar la madurez por servicio, y usarla para decidir dónde invertir.
4. Aplicar la regla del presupuesto de error cuando alguien quiere saltársela.
5. Encontrar las causas que solo se ven mirando muchos incidentes juntos.

## Conceptos centrales

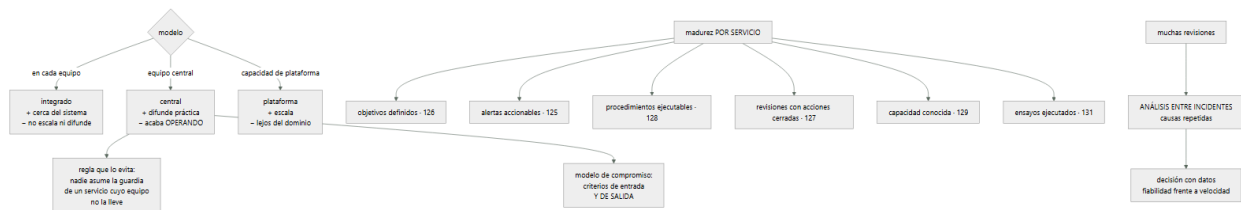
| Concepto                    | Comprensión verificable                                                                                                       |
|-----------------------------|-------------------------------------------------------------------------------------------------------------------------------|
| modelo de compromiso        | Reglas escritas de cuándo un equipo central acompaña a otro, con criterios de entrada y de salida.                            |
| guardia compartida          | Principio de que quien construye un servicio responde de él; nadie asume su guardia sin que el equipo dueño la lleve también. |
| madurez por servicio        | Nivel de las capacidades de fiabilidad de un servicio concreto. Sirve para priorizar, no para calificar equipos.              |
| regla de parada             | Compromiso previo de qué ocurre al agotar el presupuesto de error, y quién puede levantarla.                                  |
| análisis entre incidentes   | Estudio de muchas revisiones juntas para encontrar causas repetidas que ninguna revisión individual muestra.                  |
| carga de guardia sostenible | Avisos por turno e interrupciones del sueño que una persona puede soportar de forma continuada.                               |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. Tres modelos y sus modos de fallo

#### INTEGRADO EN CADA EQUIPO

- una o dos personas con foco en fiabilidad dentro del equipo
- + conocen el dominio y el sistema
- + la fiabilidad se decide donde se construye
- no escala: hacen falta muchas personas
- y la práctica no se difunde: cada equipo aprende lo mismo aparte

#### EQUIPO CENTRAL

- un grupo que acompaña, forma y construye práctica común
- + difunde lo aprendido; ve patrones entre equipos
- + puede sostener lo transversal: ensayos, análisis, herramientas
- MODO DE FALLO: acaba operando los sistemas de los demás

#### CAPACIDAD DE PLATAFORMA

- la fiabilidad se ofrece como capacidades: objetivos, alertas, procedimientos, ensayos, todo listo para usar clase 171
- + escala con el número de equipos
- lejos del dominio: la plataforma no sabe qué significa «el pedido está mal»

Y lo que suele funcionar es una combinación, con papeles claros:

la PLATAFORMA da las capacidades y los valores por defecto  
el EQUIPO CENTRAL difunde, acompaña y hace el análisis transversal  
y CADA EQUIPO responde de la fiabilidad de lo suyo

El modo de fallo del equipo central merece detalle porque es el más común:

empieza acompañando  
luego hace la guardia «temporalmente» de un servicio crítico  
luego de dos  
y al año lleva la operación de nueve sistemas que no construyó  
→ es la ventanilla de la clase 106, en versión de fiabilidad  
→ y los equipos dejan de aprender, porque alguien lo hace por ellos

Y las dos reglas que lo impiden:

- GUARDIA COMPARTIDA**  
nadie asume la guardia de un servicio cuyo equipo dueño no la lleve también  
→ si el equipo no puede llevarla, el problema es de plantilla o de complejidad, y hay que resolver ESO
- MODELO DE COMPROMISO CON SALIDA**  
se acompaña a un equipo durante un tiempo, con criterios de entrada y, sobre todo, de SALIDA  
→ «cuando estas cinco capacidades estén en nivel 3, terminamos»  
→ y la salida se ejecuta, no se pospone

Y la segunda es la que casi nunca se escribe: un acompañamiento sin criterio de salida se convierte en permanente.

### 2. Madurez por servicio, no por equipo

Con muchos servicios, hace falta saber dónde invertir. Y la unidad correcta es el servicio, no el equipo ni la organización:

| capacidad                                 | niveles | clase |
|-------------------------------------------|---------|-------|
| objetivos definidos y medidos en el borde | 0-4     | 126   |
| alertas accionables, con procedimiento    | 0-4     | 125   |
| procedimientos ejecutables por otro       | 0-4     | 128   |
| revisiones con acciones cerradas en plazo | 0-4     | 127   |

|                                       |     |     |
|---------------------------------------|-----|-----|
| capacidad y codo conocidos            | 0-4 | 129 |
| mecanismos de resiliencia verificados | 0-4 | 130 |
| ensayos ejecutados                    | 0-4 | 131 |
| continuidad probada                   | 0-4 | 166 |

Y su uso correcto, que hay que declarar para que no se malinterprete:

sirve para decidir dónde invertir y qué acompañar  
 no sirve para comparar equipos ni para evaluar a nadie  
 → y en cuanto se usa para lo segundo, las cifras dejan de ser fiables  
 ley 17

Y el nivel exigible no es el mismo para todos:

servicio del flujo principal se espera nivel 3-4 en todo  
 servicio interno de apoyo nivel 2 basta  
 herramienta con dos usuarios nivel 1  
 → y eso se declara: «este servicio es de nivel esperado 2»

Y lo que se hace con el mapa:

servicios críticos por debajo de su nivel esperado  
 → ahí va el acompañamiento del equipo central  
 huecos que aparecen en MUCHOS servicios  
 → ahí va una capacidad de plataforma, no un acompañamiento

Y la segunda línea es la que convierte el mapa en algo útil:

si cuarenta servicios están en nivel 1 en «continuidad probada»,  
 el problema no es de esos cuarenta equipos:  
 es que falta una capacidad de plataforma clase 171

Y una advertencia sobre cómo se recoge:

medido automáticamente donde se pueda  
 ¿tiene objetivos declarados? ¿tiene alertas con procedimiento?  
 ¿cuándo fue su último ensayo?  
 y declarado por el equipo donde no  
 → pero nunca solo declarado: una autoevaluación sin verificación  
 mide optimismo

### 3. Cuando alguien quiere saltarse la regla

La clase 126 estableció la regla de parada: al agotar el presupuesto de error, se para la funcionalidad nueva. A escala, esa regla se pone a prueba de forma distinta:

no la discute el equipo, que la aceptó  
 la discute alguien con más autoridad que el equipo y una fecha  
 comercial encima

Y lo que hace falta para que sobreviva:

ACORDADA ANTES, CON NOMBRE  
 quién la firmó y en qué fecha clase 126  
 → sin eso, cada agotamiento es una negociación nueva

UN CAMINO DE EXCEPCIÓN EXPLÍCITO  
 alguien PUEDE levantarla, y está escrito quién  
 con motivo, alcance y duración  
 y la excepción se publica  
 → una regla sin camino de excepción se rompe en silencio, que es peor

Y UNA CONSECUENCIA DE LEVANTARLA  
 el trabajo de fiabilidad no desaparece: se aplaza y queda anotado  
 → y si se levanta tres trimestres seguidos, eso es el dato

Y la conversación de fondo, que no es técnica:

la fiabilidad compite con la funcionalidad por el mismo tiempo  
 → decidir el reparto es una decisión de negocio  
 → y el papel de ingeniería es que se tome CON DATOS, no evitar  
 que se tome

Y los datos que la hacen posible son los que este programa ya produce:

las cuatro medidas de entrega, publicadas juntas clase 107  
 el presupuesto de error consumido y su ritmo clase 126  
 el trabajo no planificado como proporción del total clase 107

el coste de los incidentes, en euros y en horas  
y el trabajo repetitivo

clase 128

Y lo que ocurre cuando se tiene sin datos:

gana quien habla más alto o quien tiene el cargo mayor  
y la decisión no se puede revisar después, porque no hay referencia

Y una forma concreta de tenerla bien, que funciona:

cada trimestre, una revisión con negocio de una hora  
las cuatro medidas y el presupuesto de error, por producto  
el coste de los incidentes del trimestre  
y la propuesta de reparto para el siguiente  
→ y se decide ahí, con el reparto escrito

#### 4. Aprender entre incidentes

Una revisión de incidente encuentra lo que pasó en ese incidente. A escala falta lo otro: qué se repite.

revisiones al trimestre ~30  
cada una con su causa y sus acciones clase 127  
y nadie mira las treinta juntas

Y lo que aparece al mirarlas juntas es lo que ninguna revisión individual puede ver:

la misma causa en once revisiones distintas  
el mismo tipo de acción propuesta y nunca completada  
el mismo tramo del tiempo de respuesta que domina siempre  
y el mismo servicio apareciendo como origen o como víctima

El método, que cabe en media jornada al trimestre:

1. CLASIFICAR cada incidente con un vocabulario cerrado  
causa próxima: cambio, capacidad, dependencia, datos,  
configuración, dependencia externa  
factores que lo alargaron: detección, decisión, procedimiento,  
acceso, herramienta  
y qué lo habría evitado
2. CONTAR y ordenar
3. ATACAR las tres primeras, con dueño y fecha
4. Y COMPROBAR el trimestre siguiente si han bajado

Y el vocabulario cerrado es lo que hace posible contar: si cada revisión describe su causa con palabras propias, no se puede agregar nada.

Y las medidas que se derivan y que dicen si la organización aprende:

proporción de incidentes cuya causa ya había aparecido antes  
→ si es alta, se revisa mucho y se corrige poco  
acciones de revisión completadas en plazo clase 127  
incidentes por servicio y por trimestre  
y tiempo medio hasta detectar, por tipo de causa

La primera es la más reveladora de todas.

La guardia, que es la otra cuestión organizativa:

avisos por turno clase 125  
0-1 sostenible; más de 5, la gente deja de leerlos  
proporción de turnos con interrupción del sueño  
→ la cifra que decide si la guardia es sostenible  
compensación o descanso equivalente  
→ si no existe, la guardia la acaban llevando siempre los mismos  
personas elegibles por servicio clase 162  
→ menos de tres, es una dependencia de personas concretas

Y la opción de repartir por husos horarios, con su coste honesto:

+ nadie hace guardia de noche  
– exige equipos en varias zonas y traspasos disciplinados  
– y el conocimiento del sistema debe estar repartido de verdad  
→ solo compensa a partir de cierto tamaño

Y la lista de comprobación de la clase:

- está declarado quién responde de la fiabilidad de cada servicio
- nadie lleva la guardia de un servicio cuyo equipo no la lleve
- los acompañamientos tienen criterio de entrada y de SALIDA
- hay mapa de madurez por servicio, con nivel esperado declarado
- el mapa se usa para invertir, no para comparar equipos
- los huecos comunes se resuelven con capacidad de plataforma
- la regla de parada está acordada, con nombre y fecha
- existe camino de excepción explícito y publicado
- hay revisión trimestral con negocio, con las cuatro medidas
- los incidentes se clasifican con vocabulario cerrado
- hay análisis entre incidentes trimestral, con acciones
- se mide qué proporción de incidentes repite una causa conocida
- se miden avisos por turno e interrupciones del sueño
- hay al menos tres personas elegibles por servicio

Y el cierre que enlaza con la clase siguiente: fiabilidad y coste ya tienen modelo operativo. Falta el tercero de los grandes: la seguridad, que a escala organizativa deja de ser un conjunto de controles y pasa a ser una arquitectura con sus propias decisiones. Es la materia de la clase 174.

## Ejemplo trabajado

CloudShop tiene un equipo de fiabilidad de cinco personas para sesenta equipos. El ejercicio empieza contando en qué emplean el tiempo y termina encontrando una causa que aparecía en once revisiones de incidente sin que nadie lo hubiera notado.

En qué se iba el tiempo.

|                                          |      |
|------------------------------------------|------|
| guardia de servicios que no construyeron | 41 % |
| resolver incidencias de otros equipos    | 19 % |
| acompañamiento y formación               | 16 % |
| construir capacidades comunes            | 14 % |
| análisis y mejora                        | 10 % |

El 60 % operando sistemas ajenos. Y el origen de esos nueve servicios:

|                                                  |          |
|--------------------------------------------------|----------|
| servicios cuya guardia llevaba el equipo central | 9        |
| asumidos «temporalmente» durante un incidente    | 5        |
| asumidos porque el equipo dueño se disolvió      | 2        |
| asumidos porque el equipo dueño no daba abasto   | 2        |
| antigüedad media de esa situación                | 14 meses |
| equipos dueños que también llevaban la guardia   | 0        |

Ninguno de los nueve equipos dueños llevaba la guardia de su propio servicio.

La regla de guardia compartida, aplicada.

|                                                                                  |           |      |
|----------------------------------------------------------------------------------|-----------|------|
| los 5 asumidos en un incidente                                                   |           |      |
| → se devolvieron con un acompañamiento de 8 semanas y criterio de salida escrito |           |      |
| → los 5 volvieron a su equipo en 3 meses                                         |           |      |
| los 2 sin equipo dueño                                                           |           |      |
| → se asignó dueño; uno se retiró por no tener uso                                | ley 20    |      |
| los 2 cuyo equipo no daba abasto                                                 |           |      |
| → el problema era de plantilla y de complejidad                                  |           |      |
| → uno se simplificó (2 servicios unidos en 1)                                    | clase 148 |      |
| → el otro recibió una persona más                                                |           |      |
| servicios con guardia del equipo central                                         | 9         | 0    |
| tiempo en guardia ajena                                                          | 41 %      | 0 %  |
| tiempo en construir capacidades                                                  | 14 %      | 38 % |
| tiempo en análisis y mejora                                                      | 10 %      | 27 % |

Y el criterio de salida que se escribió para los acompañamientos:

|                                                       |         |
|-------------------------------------------------------|---------|
| se termina cuando el servicio alcanza                 |         |
| objetivos definidos y medidos                         | nivel 3 |
| alertas accionables con procedimiento                 | nivel 3 |
| procedimientos ejecutables por otro                   | nivel 3 |
| revisiones con acciones cerradas                      | nivel 3 |
| y el equipo ha llevado la guardia 4 semanas sin apoyo |         |

|                                       |    |
|---------------------------------------|----|
| acompañamientos iniciados en 12 meses | 11 |
|---------------------------------------|----|

|                                |   |
|--------------------------------|---|
| terminados según criterio      | 9 |
| prorrogados con motivo escrito | 2 |
| prorrogados sin motivo         | 0 |

El mapa de madurez.

|                                                                 |    |
|-----------------------------------------------------------------|----|
| servicios evaluados                                             | 87 |
| capacidades por servicio                                        | 8  |
| recogida 6 automáticas, 2 declaradas y verificadas por muestreo |    |

Y el resultado agregado, que orientó la inversión:

| capacidad                        | media | servicios en 0-1 |               |
|----------------------------------|-------|------------------|---------------|
| objetivos definidos              | 3,1   | 6                |               |
| alertas accionables              | 3,0   | 9                |               |
| procedimientos ejecutables       | 2,4   | 18               |               |
| revisiones con acciones cerradas | 2,8   | 11               |               |
| capacidad y codo conocidos       | 1,6   | 41               | ← hueco común |
| resiliencia verificada           | 1,9   | 34               | ← hueco común |
| ensayos ejecutados               | 1,2   | 52               | ← hueco común |
| continuidad probada              | 0,9   | 61               | ← hueco común |

Y la lectura del apartado segundo, aplicada:

|                                                         |           |
|---------------------------------------------------------|-----------|
| servicios críticos por debajo de su nivel esperado      | 7         |
| → acompañamiento individual                             |           |
| huecos que aparecen en más de la mitad de los servicios | 4         |
| → NO son problemas de esos equipos                      |           |
| → son capacidades de plataforma que faltan              | clase 171 |

Y las cuatro pasaron a la hoja de ruta de la plataforma:

|                                       |                                          |
|---------------------------------------|------------------------------------------|
| ensayos como servicio                 | un catálogo que un equipo ejecuta solo   |
| continuidad como capacidad            | plantilla, replicación y ensayo guiado   |
| prueba de capacidad                   | entorno y herramienta para medir el codo |
| resiliencia verificada                | batería de comprobaciones automáticas    |
| ensayos ejecutados, media             | 1,2 → 2,8                                |
| continuidad probada, media            | 0,9 → 2,4                                |
| servicios en nivel 0-1 en esas cuatro | 188 → 47                                 |

El análisis entre incidentes.

Se clasificaron las últimas cuatro tandas de revisiones con vocabulario cerrado:

|                                        |              |
|----------------------------------------|--------------|
| incidentes revisados, 12 meses         | 118          |
| causa próxima                          |              |
| cambio (despliegue o configuración)    | 41           |
| dependencia externa                    | 22           |
| capacidad                              | 19           |
| datos                                  | 14           |
| dependencia interna                    | 13           |
| otros                                  | 9            |
| factores que lo alargaron              |              |
| detección                              | 28           |
| procedimiento inexistente o inservible | 31 ← destaca |
| acceso o permisos                      | 24 ← destaca |
| decisión                               | 18           |
| herramienta                            | 11           |

Y el hallazgo que ninguna revisión individual mostraba:

en 11 revisiones distintas, de 7 equipos distintos,  
el factor que más alargó el incidente fue el mismo:

«quien estaba de guardia no tenía permiso para ejecutar  
el paso de mitigación y hubo que localizar a alguien»

|                        |        |
|------------------------|--------|
| tiempo añadido, sumado | 6 h 40 |
| en 12 meses            |        |

Y la corrección fue una sola, y no técnica:

el acceso temporal de la clase 134 existía  
y los procedimientos no decían cómo pedirlo

y quien estaba de guardia no estaba en el grupo que podía pedirlo

corrección

todo procedimiento incluye el comando exacto de solicitud

quien está de guardia entra automáticamente en el grupo

durante su turno

concesión automática para las acciones de mitigación

catalogadas, con registro

clase 134

incidentes con ese factor, trimestre siguiente

0

Y las otras dos primeras causas se atacaron igual:

PROCEDIMIENTO INSERVIBLE (31)

→ prueba de ejecución por otra persona, obligatoria

clase 128

→ 87 procedimientos revisados; 41 corregidos

→ factor en el trimestre siguiente: 31 → 9

DETECCIÓN (28)

→ la pregunta «¿qué alerta faltaba?» ya existía, y sus acciones

se completaban al 61 %

→ se cambió: la acción de detección es requisito para cerrar

la revisión

→ factor en el trimestre siguiente: 28 → 12

Y la medida que dice si la organización aprende:

incidentes cuya causa ya había aparecido antes

71 %

38 %

acciones de revisión completadas en plazo

61 %

88 %

incidentes por trimestre

30

19

La regla de parada, puesta a prueba.

trimestre 3 un producto agota su presupuesto de error el día 19

hay un lanzamiento comprometido con un cliente el día 28

sin camino de excepción

la regla se habría roto en silencio, o habría habido una discusión

que la dejaría sin valor para siempre

con camino de excepción

la levantó quien estaba autorizado, por escrito

alcance: solo ese lanzamiento

duración: hasta el día 30

contrapartida: el trabajo de fiabilidad aplazado quedó anotado

y se ejecutó en el trimestre 4

publicado en el mismo sitio que los objetivos

levantamientos en 12 meses

2

en el mismo producto

1

→ si hubieran sido tres seguidos, ese habría sido el dato a discutir

La guardia.

avisos por turno

1,4

0,9

turnos con interrupción del sueño

31 %

9 %

personas elegibles por servicio, mediana

2

4

servicios con menos de 3 elegibles

34 de 87

6 de 87

compensación o descanso equivalente

informal

declarada

A los doce meses.

servicios con guardia del equipo central

9

0

tiempo del equipo central en guardia ajena

41 %

0 %

tiempo en construir y en analizar

24 %

65 %

acompañamientos con criterio de salida

0

11

mapa de madurez

no

87 servicios

huecos comunes convertidos en capacidad

0

4

análisis entre incidentes

no

trimestral

incidentes que repiten una causa conocida

71 %

38 %

acciones de revisión completadas

61 %

88 %

incidentes por trimestre

30

19

turnos con interrupción del sueño

31 %

9 %

revisión trimestral con negocio

no

sí

La lección que esta clase traslada a la parte 14: el equipo de fiabilidad dedicaba el 60 % de su tiempo a operar nueve sistemas que no había construido, y ninguno de los nueve equipos dueños llevaba su propia guardia; devolver esos servicios con acompañamiento y criterio de salida liberó dos tercios de su capacidad. Y el hallazgo mayor del año no salió de ninguna revisión de incidente: salió de mirar ciento dieciocho juntas y descubrir que en once de ellas, de siete equipos distintos, lo que más alargó el incidente fue que quien estaba de guardia no tenía permiso para mitigar.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/173-madurez-sre-y-confiabilidad-organizacional/lab.py
```

El laboratorio selecciona el motor de práctica sre y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa assessment-sre en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

### Evidencia esperada

El artefacto principal es un SLO con presupuesto de error y política de acción. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye assessment-sre para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

### ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■ Errores frecuentes

| Síntoma                                                                                | Causa probable                                                     | Corrección                                                                                                                               |
|----------------------------------------------------------------------------------------|--------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| El equipo de fiabilidad acaba operando sistemas que no construyó                       | Se asumieron guardias temporalmente y nunca se devolvieron         | Nadie lleva la guardia de un servicio cuyo equipo dueño no la lleve, y todo acompañamiento tiene criterio de salida escrito y ejecutado. |
| El mapa de madurez se convierte en una comparación entre equipos                       | Se usó para calificar en vez de para priorizar; ley 17             | Declara el nivel esperado por servicio, úsalo para decidir dónde invertir y verifica lo declarado con muestreo.                          |
| Muchos servicios están mal en la misma capacidad                                       | Se trata como un problema de esos equipos                          | Un hueco común es una capacidad de plataforma que falta, no una carencia de sesenta equipos.                                             |
| La regla del presupuesto de error se rompe en silencio                                 | No hay camino de excepción, así que la única salida es incumplirla | Camino explícito, con quién puede levantarla, alcance, duración, contrapartida y publicación.                                            |
| Se revisan muchos incidentes y se repiten las mismas causas                            | Cada revisión mira su caso y nadie mira el conjunto                | Clasifica con vocabulario cerrado, cuenta, ataca las tres primeras causas y comprueba el trimestre siguiente.                            |
| La conversación sobre fiabilidad frente a velocidad la gana quien tiene el cargo mayor | Se tiene sin datos                                                 | Revisión trimestral con las cuatro medidas de entrega, el presupuesto de error y el coste de los incidentes, con el reparto escrito.     |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Cuál es el modo de fallo del equipo central de fiabilidad y qué dos reglas lo evitan?
2. ¿Por qué la madurez se mide por servicio y no por equipo?
3. ¿Qué significa que muchos servicios estén mal en la misma capacidad?
4. ¿Por qué una regla de parada necesita camino de excepción?
5. ¿Qué encuentra un análisis entre incidentes que ninguna revisión individual puede ver?

## Referencias

- Beyer, B. y otros (2018). The Site Reliability Workbook, caps. 1 y 20 — modelos organizativos y compromiso con equipos. <<https://sre.google/workbook/table-of-contents/>>
- Google SRE (2025). SRE engagement model — criterios de entrada y de salida de un acompañamiento. <<https://sre.google/sre-book/evolving-sre-engagement-model/>>
- Skelton, M. y Pais, M. (2019). Team Topologies, cap. 5 — equipos capacitadores frente a equipos que operan por otros. <<https://teamtopologies.com/book>>
- Allspaw, J. (2025). Learning from incidents — análisis entre incidentes y vocabulario común. <<https://www.learningfromincidents.io/>>
- PagerDuty (2025). On-call sustainability — carga por turno, interrupciones del sueño y elegibilidad. <<https://response.pagerduty.com/oncall/beingoncall/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                            | Índice              | Siguiente                                           |
|-----------------------------------------------------|---------------------|-----------------------------------------------------|
| ← 172 · Modelo operativo FinOps y economía unitaria | Parte 14 · Programa | 174 · Arquitectura de seguridad cloud empresarial → |

## Evaluación — 173 Madurez SRE y confiabilidad organizacional

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega assessment-sre con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- ☐ Resultado reproducible por una segunda persona.
- ☐ Evidencia vinculada a cada afirmación técnica.
- ☐ Prueba negativa o de fallo incluida.
- ☐ Costos y permisos expresados con unidades y alcance.
- ☐ Limitaciones declaradas sin presentar la simulación como producción.

### Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 174 — Arquitectura de seguridad cloud empresarial

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: security · Estado: EXECUTABLECORE

## Propósito

Organizar la seguridad cuando deja de ser un conjunto de controles y pasa a ser una arquitectura con decisiones propias, unas cuantas de ellas irreversibles. La clase ordena las capacidades como hizo la 171 con la plataforma, desarrolla lo que la parte 11 no cubrió —la función de detección y respuesta, con su propio problema de ruido— y trata el incidente de seguridad como lo que es: distinto de uno de disponibilidad, porque hay que preservar evidencia, hay obligaciones que cumplir y quien está dentro puede estar leyendo el canal donde se coordina la respuesta.

## Resultados de aprendizaje

Al finalizar podrás:

1. Ordenar la seguridad como capacidades con nivel, y distinguir huecos de equipo de huecos de plataforma.
2. Montar la detección sin repetir el problema de ruido de la parte 11.
3. Gestionar un incidente de seguridad con sus diferencias respecto de uno de disponibilidad.
4. Fijar las decisiones de arquitectura de seguridad que no se pueden deshacer.
5. Medir la seguridad por alcance y por tiempos, no por recuento de hallazgos.

## Conceptos centrales

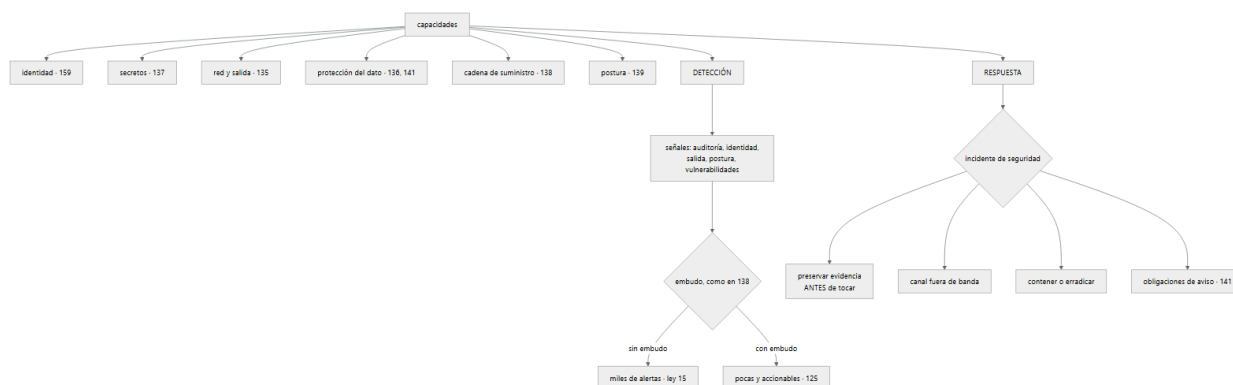
| Concepto                           | Comprensión verificable                                                                                                                |
|------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| mapa de capacidades de seguridad   | Identidad, secretos, red, protección del dato, cadena de suministro, postura, detección y respuesta, con su nivel por servicio.        |
| detección                          | Función que observa señales y decide qué merece respuesta. Su fallo característico es el mismo de la parte 10: demasiadas alertas.     |
| contención frente a erradicación   | Cortar el daño ahora frente a expulsar al atacante del todo. Actuar pronto puede avisarle; esperar cuesta datos.                       |
| canal fuera de banda               | Medio de coordinación independiente de los sistemas posiblemente comprometidos. Sin él, la respuesta se coordina delante del atacante. |
| preservación de evidencia          | Conservar el estado antes de tocarlo. Reiniciar destruye lo que permite entender el alcance.                                           |
| decisión irreversible de seguridad | Dónde vive la identidad, dónde están las fronteras y cómo se cifra. Cambiarlas después es un proyecto.                                 |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. Capacidades, y dónde están los huecos

A escala, la seguridad se gobierna igual que la plataforma: como capacidades con nivel, no como una lista de herramientas.

| capacidad                      | de dónde viene      | nivel típico |
|--------------------------------|---------------------|--------------|
| identidad de personas y cargas | clases 133, 159     |              |
| secretos y credenciales        | clase 137           |              |
| red, segmentación y salida     | clase 135           |              |
| protección del dato            | clases 136, 141     |              |
| cadena de suministro           | clases 067, 138     |              |
| postura y política             | clases 139, 170     |              |
| DETECCIÓN                      | esta clase          |              |
| RESPUESTA                      | esta clase y la 127 |              |

Y el mismo hallazgo que la clase 173:

|                             |                                                  |
|-----------------------------|--------------------------------------------------|
| un hueco en pocos servicios | → acompañamiento a esos equipos                  |
| un hueco en la mayoría      | → falta una capacidad de plataforma<br>clase 171 |

Y las dos últimas son las que la parte 11 no desarrolló, porque a pequeña escala se resuelven con la operación normal y a esta ya no.

Las decisiones de arquitectura, que conviene tomar una vez y bien, porque son irreversibles —ley 14—:

#### DÓNDE VIVE LA IDENTIDAD

|                                                  |           |
|--------------------------------------------------|-----------|
| un proveedor de identidad, y quién confía en él  | clase 159 |
| → cambiarlo después toca a todo lo que autentica |           |

#### DÓNDE ESTÁN LAS FRONTERAS DE CONFIANZA

|                                               |                 |
|-----------------------------------------------|-----------------|
| cuentas, entornos, redes, inquilinos          | clases 140, 169 |
| → mover una frontera es mover lo que contiene |                 |

#### EL MODELO DE CIFRADO

|                                                                          |           |
|--------------------------------------------------------------------------|-----------|
| qué se cifra, con qué claves, quién puede usarlas y quién administrarlas | clase 136 |
| → y el ámbito de las claves: global, por región, por cliente             |           |

#### EL MODELO DE REGISTRO Y EVIDENCIA

|                                                                                            |           |
|--------------------------------------------------------------------------------------------|-----------|
| qué se registra, dónde se guarda, quién no puede tocarlo y cuánto se conserva              | clase 141 |
| → si se decide tarde, faltará justo lo que haría falta para investigar el primer incidente |           |

#### EL MODELO DE SEGMENTACIÓN

|                                                             |           |
|-------------------------------------------------------------|-----------|
| denegación por defecto, control de salida y sus excepciones | clase 135 |
|-------------------------------------------------------------|-----------|

Y las cinco tienen la misma propiedad: decidir las mal cuesta un proyecto; no decidir las cuesta más, porque cada equipo improvisa la suya.

Y el reparto entre lo central y lo de cada equipo:

|         |                                 |
|---------|---------------------------------|
| CENTRAL | las cinco decisiones anteriores |
|---------|---------------------------------|

DE CADA EQUIPO

### 3. El incidente de seguridad es distinto

Comparte el proceso de la clase 127 —declarar, papeles, un cambio cada vez, comunicación— y tiene cuatro diferencias que cambian las decisiones:

1. HAY QUE PRESERVAR EVIDENCIA ANTES DE TOCAR  
reiniciar destruye la memoria; borrar destruye el rastro  
→ capturar primero: memoria, discos, registros, estado de red  
→ y anotar quién tomó qué y cuándo
2. QUIEN ESTÁ DENTRO PUEDE ESTAR LEYENDO  
el canal de incidentes, el correo, el repositorio, el sistema de tareas  
→ hace falta un canal FUERA DE BANDA acordado de antemano  
→ y no se discute el alcance en el canal habitual hasta saber que no está comprometido
3. CONTENER PRONTO PUEDE AVISARLE  
si se corta su acceso, sabe que se le ha visto y puede acelerar o borrar rastros  
→ y esperar cuesta datos  
→ la decisión es de negocio y de riesgo, no técnica, y se toma con quien pueda tomarla
4. HAY OBLIGACIONES CON PLAZO  
avisos a autoridades y a clientes, con plazos legales clase 141  
→ y el reloj empieza cuando se conoce, no cuando se termina de investigar

Y la segunda merece énfasis porque casi nadie la prevé:

el canal fuera de banda  
acordado ANTES, con instrucciones de cómo llegar a él  
independiente del proveedor de identidad corporativo  
y con la lista de quién debe estar  
→ y probado, porque si se estrena durante el incidente,  
no funcionará ley 22

Y los papeles, que amplían los de la clase 127:

|                                               |                     |
|-----------------------------------------------|---------------------|
| mando del incidente                           | igual que en la 127 |
| responsable técnico de la investigación       |                     |
| responsable de comunicación interna y externa |                     |
| responsable legal y de cumplimiento           | ← nuevo             |
| responsable de evidencia                      | ← nuevo             |

Y el orden de trabajo, que conviene tener escrito:

1. confirmar que es un incidente de seguridad
2. abrir canal fuera de banda
3. preservar evidencia
4. acotar el alcance: ¿qué credenciales, qué datos, desde cuándo?
5. decidir contención, con quien deba decidirlo
6. contener: rotar, revocar, aislar clases 133, 137
7. erradicar y recuperar
8. cumplir las obligaciones de aviso
9. revisar, con la disciplina de la clase 127

Y el paso 4 es el que decide todo lo demás, y el que exige haber preparado el registro de auditoría:

¿qué pudo hacer con esa credencial? clase 133  
¿qué hizo, según el registro?  
¿pudo obtener otra?  
→ y si el registro no cubre el periodo, esa pregunta no tiene respuesta

### 4. Prioridad, medida y ejercicios

La priorización de lo que se atiende, con honestidad:

|      |                                                         |           |
|------|---------------------------------------------------------|-----------|
| mal  | por lo que aparece en las noticias                      |           |
|      | → produce trabajo urgente sobre amenazas que no aplican |           |
| bien | por lo que se explota de verdad Y aplica a este entorno |           |
|      | catálogos de explotación observada                      | clase 138 |
|      | técnicas conocidas de movimiento lateral                | clase 133 |

Y el filtro que más reduce:

¿esta técnica es posible en NUESTRO entorno?

- muchas no lo son porque una decisión de arquitectura las impide
- y saberlo evita trabajo, y también evita creerse a salvo de las que sí lo son

Las medidas, que a escala hay que elegir con cuidado porque la ley 17 acecha:

|      |                                                                                               |           |
|------|-----------------------------------------------------------------------------------------------|-----------|
| mal  | número de vulnerabilidades, número de alertas, número de políticas, número de certificaciones |           |
|      | → suben o bajan sin que el riesgo cambie                                                      |           |
| bien | ALCANCE desde cada punto de entrada                                                           | clase 133 |
|      | accesos permanentes a lo sensible                                                             | clase 134 |
|      | credenciales de larga duración que quedan                                                     | clase 137 |
|      | antigüedad del hallazgo más viejo que supera el embudo                                        | clase 138 |
|      | caminos hasta objetivos críticos                                                              | clase 140 |
|      | tiempo hasta detectar y hasta contener                                                        |           |
|      | proporción de alertas accionables                                                             | clase 125 |
|      | cobertura del mapa de capacidades                                                             |           |
|      | y ejercicios ejecutados                                                                       | ley 22    |

Y las dos de tiempo son las que mejor resumen la función de detección y respuesta:

- tiempo hasta detectar de la acción del atacante al aviso
- tiempo hasta contener del aviso a cortar su acceso
- y las dos se miden en los ejercicios, porque en incidentes reales hay muy pocos datos

Los ejercicios, que son la aplicación de la ley 22 a esta materia:

DE MESA

- se plantea un escenario y el equipo dice qué haría
- barato, y encuentra huecos de proceso y de responsabilidad

TÉCNICO CONTROLADO

- se ejecuta una técnica concreta en un entorno acotado
- y se comprueba si se detecta y en cuánto

DE EQUIPO CONTRARIO

- alguien intenta alcanzar un objetivo declarado, con reglas
- y con alguien que sabe que ocurre
- el más caro y el que más enseña
- y su valor está en el informe conjunto, no en si «ganó» alguien

Y lo que hay que exigir a cualquiera de los tres:

- objetivo declarado, reglas escritas y alguien que puede pararlo
- y un informe que produzca acciones con dueño y fecha clase 127

Y la lista de comprobación de la clase:

- las cinco decisiones de arquitectura están tomadas y escritas
- hay mapa de capacidades con nivel por servicio
- los huecos comunes se resuelven con capacidad de plataforma
- las alertas de seguridad pasan por un embudo y tienen dueño
- se mide la proporción accionable, no el número de reglas
- hay línea base antes de alertar por lo anómalo
- se simula trimestralmente lo que se quiere detectar
- existe canal fuera de banda acordado y probado
- hay procedimiento de preservación de evidencia
- los papeles incluyen evidencia y cumplimiento
- la decisión de contener frente a observar la toma quien debe
- el registro de auditoría cubre el periodo necesario para investigar
- se prioriza por explotación real y por aplicabilidad
- se miden alcance y tiempos, no recuentos
- hay ejercicios de los tres tipos, con acciones

Y el cierre que enlaza con la clase siguiente: todo lo anterior vale para cualquier carga. Las de inteligencia artificial añaden exigencias propias —volumenes de datos, hardware escaso y caro, y preguntas sobre de dónde salen los datos— y son la materia de la clase 175.

## Ejemplo trabajado

CloudShop monta la función de detección y respuesta con sesenta equipos. El ejercicio empieza con doce mil alertas al mes y termina con un incidente real en el que lo primero que hizo el equipo fue abrir un canal que no había usado nunca.

El punto de partida de la detección.

|                                   |        |
|-----------------------------------|--------|
| fuentes conectadas                | 14     |
| reglas activas                    | 780    |
| alertas al mes                    | 12.400 |
| alertas atendidas                 | 310    |
| alertas que produjeron una acción | 19     |
| proporción accionable             | 0,15 % |
| personas dedicadas                | 2      |

Diecinueve acciones de doce mil cuatrocientas alertas. Es la ley 15, y con dos personas la única salida era ignorar el 97,5 %.

El embudo, aplicado.

|                                                                                                                                                  |        |
|--------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| alertas al mes                                                                                                                                   | 12.400 |
| plausibles en este entorno                                                                                                                       | 2.100  |
| → 10.300 eran de técnicas imposibles aquí: sistemas que no usamos, protocolos desactivados, servicios inexistentes sobre algo expuesto o crítico |        |
| con acción concreta asociada                                                                                                                     | 640    |
| con dueño identificable                                                                                                                          | 210    |
|                                                                                                                                                  | 190    |

Y las reglas correspondientes:

|                                                            |        |      |
|------------------------------------------------------------|--------|------|
| reglas activas                                             | 780    | 240  |
| fuentes conectadas                                         | 14     | 9    |
| → 5 no habían producido ninguna detección útil en 12 meses |        |      |
| alertas al mes                                             | 12.400 | 190  |
| proporción accionable                                      | 0,15 % | 41 % |
| acciones al mes                                            | 19     | 78   |

Y las fuentes que quedaron, ordenadas por lo que aportaron:

|                      |                                |
|----------------------|--------------------------------|
| identidad            | 51 % de las detecciones útiles |
| salida de datos      | 19 %                           |
| auditoría de la nube | 14 %                           |
| postura              | 9 %                            |
| cadena de suministro | 4 %                            |
| otras                | 3 %                            |

La mitad de las detecciones útiles venían de identidad, que es también la fuente más barata de conectar.

Las líneas base, y las seis semanas de ruido.

|                                             |                                       |   |
|---------------------------------------------|---------------------------------------|---|
| al activar la detección de accesos anómalos |                                       |   |
| semana 1                                    | 410 alertas; todas normales           |   |
| → nadie sabía qué era normal                |                                       |   |
| semanas 2-6                                 | periodo de observación, sin alertar   |   |
| semana 7                                    | se activó con la línea base aprendida |   |
|                                             | alertas semanales                     | 6 |
|                                             | de ellas, reales                      | 2 |

Y una comprobación que se añadió tras eso:

cuando se despliega un servicio nuevo, su detección entra en observación 4 semanas antes de alertar  
→ y mientras tanto, sus señales se registran pero no despiertan a nadie

La comprobación de que la detección detecta.

simulaciones trimestrales, ejecutadas en las 176 cuentas

|                                          |              |           |
|------------------------------------------|--------------|-----------|
| crear una identidad con permisos amplios | detectada en | 3 min     |
| usar una credencial dormida 6 meses      | detectada en | 8 min     |
| sacar 2 GB a un destino nuevo            | detectada en | 14 min    |
| desactivar un registro de auditoría      | NO DETECTADA | ← 1.ª vez |
| usar el acceso de emergencia             | detectada en | 1 min     |
| crear un almacén público                 | detectada en | 6 min     |

Y la que falló:

desactivar el registro de auditoría estaba impedido por política  
 en el nivel 1 clase 170  
 pero en 6 cuentas heredadas la política no estaba activa clase 169  
 → y en esas 6, la acción era posible y NO se detectaba  
 → corregido, y añadido a las pruebas negativas de todas las cuentas

El incidente real.

14:02 alerta: una credencial de una carga se usa desde una dirección  
 nunca vista, fuera del horario de esa carga  
 14:04 se confirma que es un incidente de seguridad  
 14:05 se abre el canal fuera de banda  
 → primera vez que se usaba; dos personas no sabían cómo entrar  
 → 11 minutos de retraso  
 14:16 preservación de evidencia: instantáneas de los discos,  
 volcado de memoria del proceso, exportación de registros  
 14:31 acotación del alcance con el registro de auditoría  
 ¿qué podía hacer? leer un almacén de documentos  
 ¿qué hizo? listó y descargó 41 objetos  
 ¿pudo obtener otra credencial? NO ← clase 133  
 14:48 decisión de contener, tomada por quien correspondía  
 14:52 credencial revocada; identidad rotada  
 15:10 origen encontrado: la credencial estaba en un registro de  
 una herramienta interna que la mostraba clase 137  
 16:30 aviso a los clientes cuyos documentos se habían descargado  
 → 3 clientes; dentro del plazo legal clase 141

tiempo hasta detectar ~7 min  
 tiempo hasta contener 50 min  
 datos afectados 41 documentos, 3 clientes  
 credenciales adicionales obtenidas 0

Y las tres cosas que funcionaron y las tres que no:

FUNCIONÓ

la detección: 7 minutos desde el primer uso  
 la respuesta a «¿pudo obtener otra credencial?»: no,  
 gracias al trabajo de la clase 133  
 y el registro de auditoría cubría el periodo

NO FUNCIONÓ

el canal fuera de banda: nunca se había usado ley 22  
 la preservación de evidencia: no había procedimiento y  
 se improvisó; se perdió el estado de red  
 y la credencial estaba visible en una herramienta interna  
 que ya se había señalado en la clase 137 y no se había corregido  
 en ese componente

|                                          |         |            |
|------------------------------------------|---------|------------|
| canal fuera de banda probado             | no      | trimestral |
| personas que saben llegar a él           | 4 de 11 | 11 de 11   |
| procedimiento de evidencia               | no      | sí         |
| con lista de qué capturar y en qué orden |         |            |
| componentes que muestran credenciales    | 1       | 0          |

Los ejercicios.

|                                           |                                                                         |
|-------------------------------------------|-------------------------------------------------------------------------|
| de mesa, trimestrales                     | 4                                                                       |
| hallazgos                                 | 23                                                                      |
| de ellos, de proceso o de responsabilidad | 19                                                                      |
| técnicos controlados, trimestrales        | 4                                                                       |
| técnicas probadas                         | 18                                                                      |
| detectadas                                | 14                                                                      |
| no detectadas                             | 4 → reglas nuevas                                                       |
| de equipo contrario, anual                | 1                                                                       |
| objetivo declarado                        | «llegar a los datos de un cliente<br>desde una posición de empleado»    |
| resultado                                 | no lo consiguió; llegó hasta un almacén<br>interno sin datos personales |
| hallazgos                                 | 11                                                                      |
| de ellos, corregidos en el trimestre      | 10                                                                      |

Y el hallazgo del ejercicio anual que más valió:

el equipo defensor tardó 2 h 40 en darse cuenta  
y la señal que lo delató no fue ninguna de las reglas:  
fue un aumento del volumen de salida de una carga      clase 135  
→ se convirtió en regla, con línea base por carga

Las medidas, cambiadas.

| lo que se publicaba                          | nº de vulnerabilidades, nº de alertas |
|----------------------------------------------|---------------------------------------|
| lo que se publica ahora                      |                                       |
| alcance desde cada punto de entrada          | 2 de 6 puntos → 0                     |
| accesos permanentes a lo sensible            | 19 → 0                                |
| credenciales de larga duración               | 14 → 0                                |
| antigüedad del hallazgo más viejo del embudo | 19 meses → 5 días                     |
| camino hasta objetivos críticos              | 14 → 2                                |
| tiempo hasta detectar (ejercicios)           | no se medía → 9 min                   |
| tiempo hasta contener (ejercicios)           | no se medía → 24 min                  |
| proporción de alertas accionables            | 0,15 % → 41 %                         |
| ejercicios ejecutados al año                 | 0 → 9                                 |

A los doce meses.

|                                               |          |             |
|-----------------------------------------------|----------|-------------|
| reglas de detección                           | 780      | 240         |
| alertas al mes                                | 12.400   | 190         |
| proporción accionable                         | 0,15 %   | 41 %        |
| acciones al mes                               | 19       | 78          |
| fuentes conectadas                            | 14       | 9           |
| simulaciones de detección                     | no había | trimestral  |
| técnicas no detectadas al simular             | —        | 4 de 18 → 0 |
| canal fuera de banda probado                  | no       | trimestral  |
| procedimiento de evidencia                    | no       | sí          |
| ejercicios al año                             | 0        | 9           |
| cuentas donde la política del nivel 1 faltaba | 6        | 0           |

La lección que esta clase traslada a la parte 14: de doce mil cuatrocientas alertas al mes, diez mil trescientas eran de técnicas imposibles en este entorno, y el embudo que las eliminó multiplicó por cuatro las acciones reales con las mismas dos personas. Y en el incidente real la detección funcionó en siete minutos y lo que falló fue lo que nunca se había ejecutado: el canal fuera de banda tardó once minutos en abrirse porque dos de las once personas no sabían cómo entrar, que es la ley 22 en el peor momento posible.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/174-arquitectura-de-seguridad-cloud-empresarial/lab.py
```

El laboratorio selecciona el motor de práctica security y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa security-reference-architecture en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

## Evidencia esperada

El artefacto principal es un control con amenaza, mitigación y evidencia. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye security-reference-architecture para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

## ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■ Errores frecuentes

| Síntoma                                                                      | Causa probable                                                                               | Corrección                                                                                                                   |
|------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------|
| Miles de alertas de seguridad al mes y casi ninguna acción                   | Se activaron todas las reglas del catálogo sin filtrar por lo que es posible en este entorno | Embudo: plausible aquí, sobre algo expuesto, con acción concreta y con dueño; y mide la proporción accionable.               |
| Una detección nueva produce ruido durante semanas                            | No hay línea base: nadie sabe qué es normal                                                  | Periodo de observación antes de alertar, también para cada servicio nuevo.                                                   |
| Se descubre que una técnica no se detectaba en algunas cuentas               | La política o la regla no estaban activas en todas                                           | Simula trimestralmente lo que quieres detectar, en todas las cuentas y clústeres.                                            |
| Durante un incidente se coordina la respuesta donde el atacante puede leerla | No hay canal fuera de banda, o nunca se ha usado                                             | Acuérdalo antes, con instrucciones y lista de participantes, y pruébalo cada trimestre.                                      |
| Se reinicia el sistema y se pierde la posibilidad de saber el alcance        | No hay procedimiento de preservación de evidencia                                            | Captura memoria, discos, registros y estado de red antes de tocar nada, con registro de quién tomó qué.                      |
| Se publica el número de vulnerabilidades como medida de seguridad            | Ley 17: sube y baja sin que el riesgo cambie                                                 | Publica alcance por punto de entrada, accesos permanentes, caminos a objetivos críticos y tiempos hasta detectar y contener. |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Cuáles son las cinco decisiones de arquitectura de seguridad que no se pueden deshacer?
2. ¿Qué fuente de señales aporta más detecciones útiles y por qué?
3. ¿Cuáles son las cuatro diferencias de un incidente de seguridad respecto de uno de disponibilidad?
4. ¿Por qué contener pronto no siempre es la mejor decisión, y quién decide?
5. ¿Qué se mide en lugar del número de vulnerabilidades o de alertas?

## Referencias

- NIST (2012). SP 800-61: computer security incident handling guide — fases, evidencia y comunicación.  
<<https://csrc.nist.gov/pubs/sp/800/61/r2/final>>
- MITRE (2025). ATT&CK and detection engineering — priorizar por técnicas reales y aplicables.  
<<https://attack.mitre.org/>>
- Google (2025). Autonomic security operations — detección con embudo y medida de la proporción accionable.  
<<https://cloud.google.com/blog/products/identity-security>>

- SANS (2025). Incident response and out-of-band communications — canal fuera de banda y preservación de evidencia. <<https://www.sans.org/white-papers/>>
- CISA (2025). Cloud security technical reference architecture — decisiones de arquitectura y capacidades. <<https://www.cisa.gov/resources-tools/resources/cloud-security-technical-reference-architecture>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                           | Índice              | Siguiente                                                  |
|----------------------------------------------------|---------------------|------------------------------------------------------------|
| ← 173 · Madurez SRE y confiabilidad organizacional | Parte 14 · Programa | 175 · Workloads de IA, GPU, datos y MLOps multi-cloud<br>→ |

## Evaluación — 174 Arquitectura de seguridad cloud empresarial

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega security-reference-architecture con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- ☐ Resultado reproducible por una segunda persona.
- ☐ Evidencia vinculada a cada afirmación técnica.
- ☐ Prueba negativa o de fallo incluida.
- ☐ Costos y permisos expresados con unidades y alcance.
- ☐ Limitaciones declaradas sin presentar la simulación como producción.

### Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 175 — Workloads de IA, GPU, datos y MLOps multi-cloud

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: ai · Estado: EXECUTABLECORE

## Propósito

Tratar las cargas de aprendizaje automático como lo que son en términos de ingeniería: una canalización de datos con un paso de cómputo caro pegado. Casi todos sus problemas ya están resueltos en las partes 09 a 12; lo genuinamente nuevo son dos cosas: la economía de un hardware escaso donde el tiempo ocioso es pérdida pura, y las preguntas sobre de dónde salen los datos, que no tienen respuesta técnica. Y una tercera que conviene decir pronto: un modelo que responde mal tiene todas sus señales en verde.

## Resultados de aprendizaje

Al finalizar podrás:

1. Situar qué es nuevo y qué es la parte 09 con otro nombre.
2. Aprovechar un hardware escaso, donde el tiempo ocioso no se recupera.
3. Responder de dónde salen los datos y qué se puede hacer con ellos.
4. Reconocer las decisiones irreversibles y los dobles escritores propios de esta materia.
5. Vigilar la calidad de las respuestas, que ninguna señal técnica muestra.

## Conceptos centrales

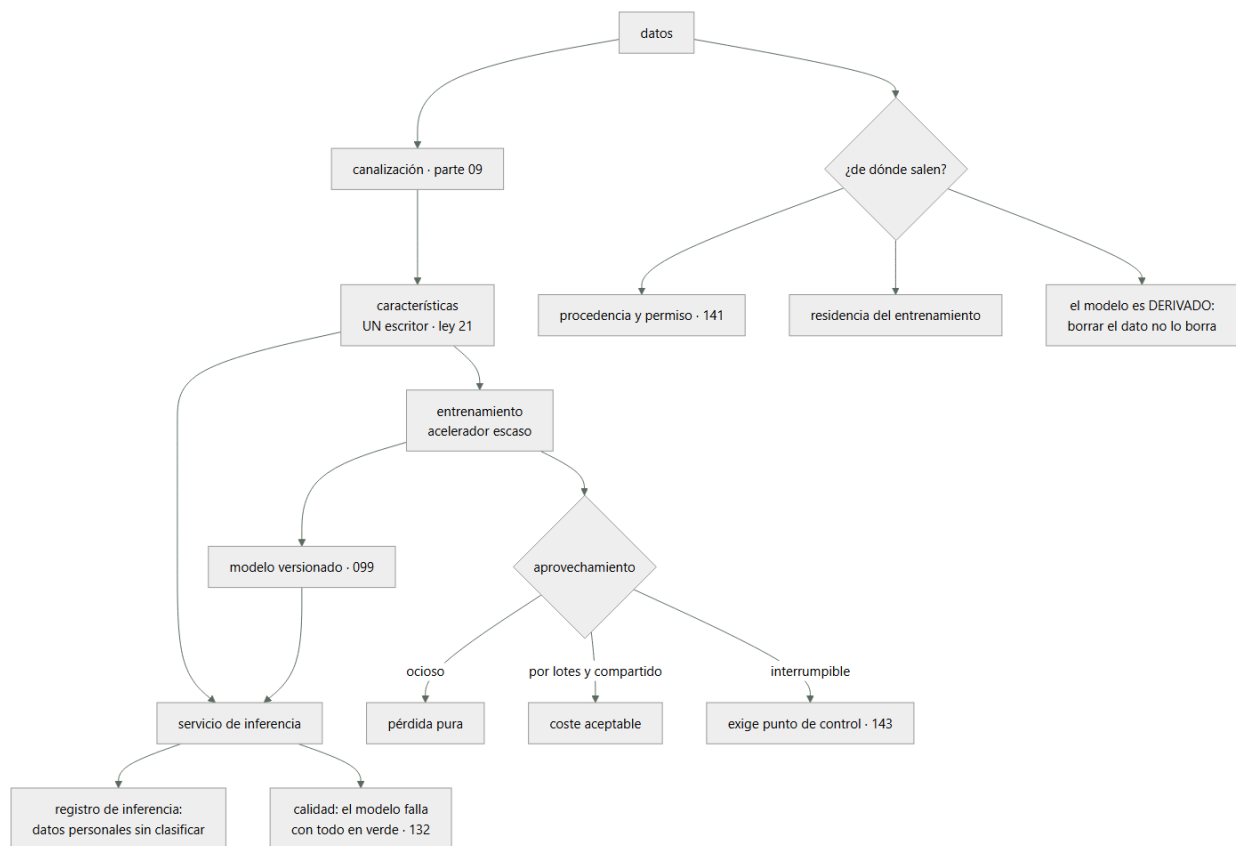
| Concepto                                  | Comprensión verificable                                                                                                  |
|-------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| aprovechamiento del acelerador            | Proporción del tiempo contratado que el acelerador está calculando. Es la magnitud que decide el coste de estas cargas.  |
| punto de control                          | Estado guardado que permite reanudar un entrenamiento. Sin él no se puede usar capacidad interrumpible.                  |
| procedencia de los datos                  | De dónde salió cada dato de entrenamiento y con qué permiso se usa. No es una propiedad técnica: es contractual y legal. |
| desviación entre entrenamiento y servicio | Que el cálculo de una característica difiera entre el entrenamiento y la inferencia. Es un problema de dos escritores.   |
| deriva                                    | Cambio en los datos de entrada o en la relación que el modelo aprendió. El sistema sigue funcionando y responde peor.    |
| registro de inferencia                    | Lo que se guarda de cada petición al modelo. Suele contener datos personales que nadie clasificó.                        |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. Qué es nuevo y qué no

Conviene separar antes de empezar, porque la mayor parte del trabajo ya está hecha:

LO QUE ES LA PARTE 09 CON OTRO NOMBRE  
 ingesta y transformación de datos clase 112  
 formatos columnares y particionado clase 112  
 colas, reintentos e idempotencia clases 113, 116  
 orquestación de procesos largos clase 119  
 y el lago como sitio de las consultas no previstas

LO QUE ES LA PARTE 08 CON OTRO NOMBRE  
 artefacto inmutable y versionado: el modelo lo es clase 099  
 despliegue escalonado y reversión clase 102  
 y puertas en la canalización clase 100

LO QUE ES LA PARTE 12 CON OTRO NOMBRE  
 contratos: el esquema de las características es uno clase 153  
 y la propiedad del dato ley 21

LO GENUINAMENTE NUEVO  
 1. la economía de un hardware escaso  
 2. las preguntas sobre de dónde salen los datos  
 3. que la calidad de la respuesta no se ve en ninguna señal técnica

Y conviene decir con claridad qué no cambia:

un modelo desplegado es un artefacto: se versiona, se firma,  
 se despliega escalonado y se revierte parte 08  
 una canalización de entrenamiento es un proceso por lotes  
 y un servicio de inferencia es un servicio: tiene objetivos,  
 latencia, capacidad y guardia partes 10, 12

Y el error de tratarlo como algo aparte:

equipos de datos con su propia canalización, su propia forma de  
 desplegar y su propia observabilidad  
 → es la proliferación que la clase 148 desaconsejaba, con otro nombre

→ y produce modelos en producción sin objetivos, sin reversión  
y sin nadie de guardia

Y la observación de escala, que aquí es especialmente cierta:

la parte cara y difícil de estas cargas no es el modelo:  
son los datos y su movimiento clase 161  
→ y por eso el cómputo va donde están los datos, y no al revés

## 2. Un hardware que no espera

El acelerador es escaso, caro y se paga por tiempo contratado, no por tiempo útil:

el tiempo ocioso no se recupera  
un acelerador reservado y sin trabajo cuesta lo mismo que uno  
al 100 %

Y de ahí que el aprovechamiento sea la magnitud que decide el coste:

aprovechamiento típico sin gestión 20-40 %  
con cola y agrupación 70-85 %  
→ la diferencia es un factor de dos o tres en la factura

Y lo que baja el aprovechamiento, por frecuencia:

esperar datos: el acelerador calcula más rápido de lo que se le  
alimenta ← la primera causa  
trabajos que reservan y no empiezan  
reservas «por si acaso» durante el día  
experimentos manuales que dejan sesiones abiertas  
y tamaños de lote que no llenan la memoria disponible

Y la primera merece énfasis: muy a menudo el cuello de botella es el almacenamiento o la red, no el acelerador, y se resuelve con formato, prelectura y colocación —clases 112 y 161—, no comprando más.

Las técnicas que lo suben, en orden:

1. COLA DE TRABAJOS  
una cola con prioridades en vez de reservas por persona  
→ nadie tiene un acelerador «suyo»
2. AGRUPACIÓN  
juntar peticiones o muestras para llenar el cálculo  
→ en inferencia, es un traslado: más caudal y más latencia
3. COMPARTIR EL DISPOSITIVO  
repartir un acelerador entre varias cargas pequeñas  
→ útil para inferencia y para desarrollo, no para entrenamientos grandes
4. CAPACIDAD INTERRUMPIBLE clase 143  
mucho más barata, y EXIGE punto de control  
→ un entrenamiento que no puede reanudar no puede usarla
5. ALIMENTAR MEJOR clases 112, 161  
formato columnar, prelectura, datos cerca

Y la cuarta tiene una consecuencia de diseño concreta:

guardar estado cada N pasos, y poder reanudar desde ahí  
→ y probarlo de verdad: reanudar sin haberlo ensayado no funciona  
ley 22

Y la capacidad, que es la clase 129 en su forma más aguda:

la cuota de aceleradores es escasa y por región  
conseguir más tarda semanas, y a veces no se consigue  
→ es de lo que «no escala solo» clase 129  
→ y por eso es uno de los motivos legítimos para usar varios  
proveedores: capacidad exclusiva clase 157

Y el coste unitario, que es el lenguaje de la clase 172:

coste por entrenamiento completo  
coste por mil inferencias  
→ y esta segunda suele dominar el coste unitario del producto

### 3. De dónde salen los datos

Aquí está lo que no tiene respuesta técnica y decide si una carga se puede poner en producción.

Procedencia y permiso:

- ¿de dónde salió cada conjunto de datos?
- ¿con qué permiso se usa: contrato, consentimiento, licencia?
- ¿permite ese permiso entrenar un modelo, o solo prestar el servicio?
- ¿incluye datos de terceros que a su vez tienen sus propias condiciones?
- ¿y datos de clientes que no lo autorizaron?

Y la forma de responderlas es un inventario, no una intuición:

- por conjunto de datos
  - origen, fecha, permiso que lo ampara y su vigencia
  - categorías que contiene clase 141
  - y qué modelos se han entrenado con él
- la última columna es la que casi nadie mantiene y la que hace falta cuando alguien pregunta

La consecuencia incómoda, que conviene entender bien:

- un modelo entrenado con unos datos es un DERIVADO de esos datos
- borrar los datos de una persona no borra su influencia en el modelo
- y volver a entrenar sin ellos cuesta lo que cuesta entrenar

Y las salidas prácticas, con su compromiso:

- no usar datos personales para entrenar, y usar agregados
  - lo más limpio, y a veces reduce la calidad
- entrenar con datos seudonimizados y sin atributos identificativos
- reentrenar con cadencia, de modo que la exclusión surta efecto en el siguiente ciclo, y decirlo así
- y llevar el registro de qué modelo se entrenó con qué

La residencia, que es la clase 141 con un matiz nuevo:

- los datos de entrenamiento tienen residencia
- y el modelo resultante, ¿también?
- depende del marco y del contrato, y hay que preguntarlo, no suponerlo
- y los registros de inferencia son datos nuevos, generados en el sitio donde se sirve

Los registros de inferencia, que son el problema silencioso:

- se guardan las entradas y las salidas «para mejorar el modelo»
- y esas entradas contienen lo que el usuario escribió
- que puede ser cualquier cosa: datos personales, secretos, información de terceros
- y nadie los clasificó clase 141

Y lo mínimo exigible:

- clasificarlos como cualquier otro dato
- decidir retención, y que sea corta por defecto
- depurar lo identificable antes de guardarlos clase 122
- y decir al usuario qué se guarda

### 4. Lo irreversible, los dos escritores y la calidad

Decisiones que se toman al crear —ley 14— y que después cuestan un reentrenamiento o una migración:

- qué características usa el modelo, y cómo se calculan
- el esquema de los datos de entrenamiento
- la partición de los conjuntos: entrenamiento, validación, prueba
  - si se hace mal, el modelo parece mejor de lo que es y no se sabrá hasta producción
- la región donde vive el conjunto y donde se entrena
- y el formato en que se guardan los datos históricos clase 112

El problema de los dos escritores —ley 21— tiene aquí una forma propia y muy conocida:

- la característica «gasto medio de los últimos 30 días»
  - se calcula de una forma en la canalización de entrenamiento
  - y de otra en el servicio de inferencia
- dos escritores del mismo concepto

- el modelo aprende con una definición y responde con otra
- y el efecto es una pérdida de calidad que no produce ningún error

Y la corrección es la misma que la clase 147 dio para los datos:

UN SOLO SITIO QUE CALCULA CADA CARACTERÍSTICA  
y las dos partes lo consumen

- es lo que hace un almacén de características, y su valor no está en el almacenamiento: está en tener un único escritor

y si no lo hay, la comprobación mínima:

calcular la característica por los dos caminos sobre los mismos datos y comparar, de forma automática

La calidad, que es lo que ninguna señal técnica muestra:

|            |        |
|------------|--------|
| latencia   | bien   |
| errores    | cero   |
| saturación | normal |

y el modelo lleva tres semanas recomendando mal

- es la limitación que la clase 132 dejó escrita: el sistema funciona correctamente según todas sus señales y está mal

Y lo que sí lo detecta:

DERIVA DE ENTRADA

las distribuciones de entrada se alejan de las del entrenamiento  
→ se detecta sin conocer el resultado, y es la primera línea

MÉTRICAS DE NEGOCIO

proporción de recomendaciones aceptadas, de conversiones,  
de correcciones humanas  
→ es la señal verdadera, y llega con retraso

MUESTREO Y REVISIÓN HUMANA

un porcentaje de respuestas revisadas por personas  
→ caro y es lo único que detecta ciertos fallos

COMPARACIÓN CON UNA REFERENCIA

un modelo anterior o una regla sencilla ejecutados en paralelo  
→ y comparar; es el tráfico en espejo de la clase 167

Y el despliegue, que es el de la parte 08 sin cambios:

|                                                             |           |
|-------------------------------------------------------------|-----------|
| modelo versionado y firmado                                 | clase 099 |
| escalonado con análisis, comparando contra el modelo actual | clase 102 |
| interruptor para volver al anterior o a una regla           | clase 105 |
| y reversión que no requiera reentrenar                      |           |

Y la lista de comprobación de la clase:

- la canalización, el despliegue y la observabilidad son los comunes
- el modelo se versiona, se firma y se puede revertir
- el servicio de inferencia tiene objetivos, capacidad y guardia
- se mide el aprovechamiento del acelerador
- se ha comprobado si el cuello de botella son los datos y no el cálculo
- hay cola de trabajos en vez de reservas por persona
- los entrenamientos guardan punto de control y se ha probado reanudar
- existe inventario de conjuntos con origen, permiso y modelos derivados
- está decidido qué ocurre con un modelo cuando se borran datos
- los registros de inferencia están clasificados, depurados y con retención
- cada característica tiene un solo sitio donde se calcula
- hay comprobación de que entrenamiento y servicio coinciden
- se vigila deriva de entrada y al menos una métrica de negocio
- hay comparación contra una referencia o revisión humana muestreada

Y el cierre que enlaza con la clase siguiente: estas cargas suelen tener su origen en sensores y dispositivos que están lejos, con conectividad mala y volúmenes que no caben en ninguna red. Procesar donde se genera el dato y decidir qué viaja es la materia de la clase 176.

## Ejemplo trabajado

CloudShop tiene un modelo de recomendaciones en producción y dos equipos de datos que trabajan por su cuenta. El ejercicio empieza midiendo el aprovechamiento del hardware y termina con una pregunta legal que estuvo a punto de parar el producto.

El aprovechamiento.

|                          |         |
|--------------------------|---------|
| aceleradores contratados | 8       |
| coste mensual            | 9.400 € |
| aprovechamiento medido   | 27 %    |

Y el desglose de lo que consumía el tiempo:

|                                          |      |              |
|------------------------------------------|------|--------------|
| calculando                               | 27 % |              |
| esperando datos                          | 41 % | ← la primera |
| reservados sin trabajo                   | 22 % |              |
| sesiones interactivas abiertas y ociosas | 10 % |              |

Y las tres correcciones:

|                                                    |                 |       |
|----------------------------------------------------|-----------------|-------|
| ESPERANDO DATOS                                    |                 |       |
| los datos se leían de ficheros JSON en otra región | clases 112, 161 |       |
| → columnar, comprimido y en la misma región        |                 |       |
| tiempo de espera                                   | 41 %            | → 6 % |

|                                                  |      |       |
|--------------------------------------------------|------|-------|
| RESERVADOS SIN TRABAJO                           |      |       |
| cada persona tenía «su» acelerador               | 22 % | → 3 % |
| → cola con prioridades; nadie tiene uno asignado |      |       |

|                                                   |               |              |
|---------------------------------------------------|---------------|--------------|
| SESIONES OCIOSAS                                  |               |              |
| → caducidad automática a los 60 min sin actividad | 10 %          | → 1 %        |
| aprovechamiento                                   | 27 %          | 78 %         |
| aceleradores necesarios                           | 8             | 3            |
| coste mensual                                     | 9.400 €       | 3.500 €      |
| tiempo de espera en cola, mediana                 | 0 (reservado) | 11 min       |
| quejas por esperar                                | –             | 2 en 6 meses |

Tres aceleradores en lugar de ocho, sin reducir el trabajo hecho.

Y la capacidad interrumpible, que exigió trabajo previo:

|                                                        |                                                        |                           |
|--------------------------------------------------------|--------------------------------------------------------|---------------------------|
| entrenamientos que podían usar capacidad interrumpible |                                                        |                           |
| antes                                                  | 0 de 4                                                 | → ninguno guardaba estado |
| después                                                | 4 de 4                                                 |                           |
| trabajo añadir punto de control cada 15 minutos        |                                                        |                           |
| ensayo                                                 | retirar la capacidad a propósito                       | ley 22                    |
| primera vez                                            | el punto de control se guardaba y no se sabía reanudar |                           |
|                                                        | → 6 h de entrenamiento perdidas en la prueba           |                           |
|                                                        | tras corregir reanudación correcta, +9 min de coste    |                           |
| coste de los entrenamientos                            | 3.500 €                                                | → 1.400 €                 |

La pregunta legal.

Durante una revisión de la clase 141 alguien preguntó de dónde salían los datos de entrenamiento:

|                                     |                         |
|-------------------------------------|-------------------------|
| conjuntos usados                    | 6                       |
| con origen documentado              | 2                       |
| con permiso comprobado              | 1                       |
| al reconstruirlo                    |                         |
| histórico de navegación de clientes | contrato lo permite ✓   |
| histórico de compras                | contrato lo permite ✓   |
| catálogo y descripciones            | propio ✓                |
| valoraciones de clientes            | contrato lo permite ✓   |
| datos de un proveedor de terceros   | LICENCIA NO PERMITE     |
|                                     | entrenar modelos ✗      |
| datos de una empresa adquirida      | consentimiento anterior |
|                                     | no cubría este uso ✗    |

Dos de seis no se podían usar para entrenar.

acciones

se retiraron los dos conjuntos  
se reentrenó sin ellos  
pérdida de calidad medida -3,1 % en aceptación  
→ aceptada, frente al riesgo de usarlos  
y se renegoció la licencia del proveedor: ahora sí lo permite  
→ se reincorporó 4 meses después

y lo que se montó para que no vuelva a ocurrir  
inventario por conjunto: origen, permiso, vigencia, categorías  
y qué modelos se entrenaron con él  
puerta en la canalización de entrenamiento: no se entrena con  
un conjunto sin permiso vigente registrado  
conjuntos bloqueados por esa puerta en 12 meses 3

#### Los registros de inferencia.

qué se guardaba la petición completa y la respuesta,  
«para mejorar el modelo»  
retención indefinida  
clasificación ninguna  
volumen acumulado 14 TB

al analizar una muestra clase 122  
peticiones con datos personales 31 %  
peticiones con texto libre escrito por el usuario 19 %  
→ y en ese texto: nombres, correos, en un caso un número  
de tarjeta

clasificación ninguna personal  
retención indefinida 30 días  
depuración antes de guardar no sí  
texto libre se guardaba se guarda solo  
si el usuario lo  
autoriza

volumen 14 TB 410 GB  
sujeto al borrado por solicitud no sí clase 141

#### La desviación entre entrenamiento y servicio.

##### Al comparar cómo se calculaban las características:

características usadas por el modelo 31  
calculadas en dos sitios distintos 31  
con definición idéntica comprobada 0

comparación automática sobre los mismos datos  
coinciden 24  
DIFIEREN 7  
→ 3 por ventana temporal distinta (30 días naturales frente  
a 30 días móviles)  
→ 2 por tratamiento de valores ausentes  
→ 2 por redondeo

##### Y el efecto medido:

al corregir las 7 y reentrenar  
aceptación de recomendaciones +4,8 %

Casi cinco puntos de mejora sin tocar el modelo, solo porque entrenamiento y servicio calculaban lo mismo de la misma manera.

sitios donde se calcula cada característica 2 1  
comprobación automática de coincidencia no sí  
diferencias detectadas después - 2 en 8 meses

#### La calidad, y el mes que nadie vio.

mes 4 la aceptación de recomendaciones cae del 11,2 % al 7,4 %  
latencia normal  
errores cero  
saturación normal  
alertas disparadas 0  
tiempo hasta que alguien lo notó 23 días  
cómo se notó una caída en las ventas cruzadas

causa un cambio en el catálogo modificó la distribución de una

característica; el modelo seguía respondiendo con confianza

Y lo que se montó después:

|                                                                                            |                                     |
|--------------------------------------------------------------------------------------------|-------------------------------------|
| deriva de entrada, por característica, con alerta                                          | diaria                              |
| métrica de negocio en el panel del servicio, junto al objetivo                             | clase 125                           |
| modelo de referencia sencillo, ejecutado en paralelo sobre el 1 % del tráfico, y comparado |                                     |
| revisión humana muestreada                                                                 | 50 respuestas/semana                |
| repetición del mismo escenario, provocada                                                  |                                     |
| tiempo hasta detectar                                                                      | 23 días → 2 días                    |
| qué lo detectó                                                                             | la deriva de entrada, no el negocio |

El despliegue, unificado con el resto.

|                                      |                   |                   |           |
|--------------------------------------|-------------------|-------------------|-----------|
| canalización                         | propia del equipo | la común          | clase 106 |
| modelo versionado y firmado          | no                | sí                |           |
| despliegue escalonado                | todo de una vez   | 1-10-50-100 %     |           |
| comparación contra el modelo actual  | no                | sí                |           |
| interruptor para volver              | no                | sí                | clase 105 |
| objetivos e indicadores del servicio | no                | sí                | clase 126 |
| alguien de guardia                   | el autor          | equipo, rotatorio |           |

A los doce meses.

|                                            |         |         |
|--------------------------------------------|---------|---------|
| aprovechamiento del acelerador             | 27 %    | 78 %    |
| aceleradores                               | 8       | 3       |
| coste mensual de entrenamiento             | 9.400 € | 1.400 € |
| entrenamientos con punto de control        | 0 de 4  | 4 de 4  |
| conjuntos con permiso documentado          | 1 de 6  | 6 de 6  |
| conjuntos usados sin permiso               | 2       | 0       |
| registros de inferencia                    | 14 TB   | 410 GB  |
| características con dos definiciones       | 31      | 0       |
| aceptación de recomendaciones              | 11,2 %  | 16,4 %  |
| tiempo hasta detectar una caída de calidad | 23 días | 2 días  |
| coste por mil inferencias                  | 0,84 €  | 0,31 €  |

La lección que esta clase traslada a la parte 14: el hardware caro estaba el 41 % del tiempo esperando datos —el cuello de botella no era el acelerador—, y arreglar el formato y la colocación permitió pasar de ocho aceleradores a tres. Y las dos mejoras de calidad no vinieron del modelo: casi cinco puntos por hacer que entrenamiento y servicio calcularan las características de la misma forma, y la detección de una caída de calidad que había durado veintitrés días con todas las señales técnicas en verde. Lo único que estuvo a punto de parar el producto no fue técnico: dos de seis conjuntos de datos no se podían usar para entrenar.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/175-workloads-de-ia-gpu-datos-y-mlops-multi-cloud/lab.py
```

El laboratorio selecciona el motor de práctica ai y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa arquitectura-ia-cloud en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

## Evidencia esperada

El artefacto principal es una arquitectura de IA con datos, cómputo, costo y seguridad. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye arquitectura-ia-cloud para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

### ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

### ■ Errores frecuentes

| Síntoma                                                                  | Causa probable                                                                        | Corrección                                                                                                                             |
|--------------------------------------------------------------------------|---------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| El acelerador cuesta mucho y está la mayor parte del tiempo sin calcular | Espera datos, o está reservado por personas que no lo usan                            | Mide el desglose del tiempo; corrige formato y colocación de los datos, sustituye reservas por una cola y caduca las sesiones ociosas. |
| No se puede usar capacidad interrumpible para entrenar                   | El entrenamiento no guarda estado ni sabe reanudar                                    | Punto de control periódico y ensayo real de retirada de capacidad.                                                                     |
| Nadie sabe si se pueden usar los datos con los que se entrena            | No hay inventario de origen y permiso                                                 | Inventario por conjunto con origen, permiso, vigencia y modelos derivados, y una puerta que impida entrenar sin él.                    |
| El modelo responde peor que en las pruebas                               | Desviación entre entrenamiento y servicio: la característica se calcula de dos formas | Un solo sitio que calcula cada característica, y comprobación automática de que ambos caminos coinciden.                               |
| La calidad cae durante semanas con todas las señales en verde            | Ninguna señal técnica mide si la respuesta es buena                                   | Deriva de entrada, métrica de negocio en el panel, modelo de referencia en paralelo y revisión humana muestreada.                      |
| Se acumulan terabytes de peticiones con datos personales                 | Los registros de inferencia se guardan sin clasificar ni depurar                      | Clasifícalos, depura lo identificable, fija retención corta y sújélos al proceso de borrado.                                           |

### ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

### ■ Preguntas de comprobación

1. ¿Qué parte de estas cargas es nueva y cuál son las partes 08 a 12 con otro nombre?
2. ¿Cuál suele ser la primera causa de bajo aprovechamiento del acelerador?
3. ¿Por qué borrar los datos de una persona no elimina su influencia en un modelo?
4. ¿Qué es la desviación entre entrenamiento y servicio y por qué es un problema de dos escritores?
5. ¿Qué señales detectan una caída de calidad que la latencia y los errores no muestran?

## Referencias

- Sculley, D. y otros (2015). Hidden technical debt in machine learning systems — la mayor parte del sistema no es el modelo. <<https://papers.nips.cc/paper/2015/hash/86df7dcfd896fcdf2674f757a2463eba-Abstract.html>>
- Google Cloud (2025). MLOps: continuous delivery and automation pipelines — versionado de datos, características y modelos. <<https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning>>
- Feast (2025). Feature store: training and serving consistency — un solo cálculo por característica. <<https://docs.feast.dev/>>
- NVIDIA (2025). GPU utilization, sharing and checkpointing — aprovechamiento, reparto del dispositivo y reanudación. <<https://docs.nvidia.com/datacenter/cloud-native/>>
- ICO (2025). AI and data protection: lawful basis and training data — procedencia, permiso y derechos sobre datos usados para entrenar. <<https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/artificial-intelligence/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                            | Índice              | Siguiente                                      |
|-----------------------------------------------------|---------------------|------------------------------------------------|
| ← 174 · Arquitectura de seguridad cloud empresarial | Parte 14 · Programa | 176 · Edge, IoT y procesamiento desconectado → |

## Evaluación — 175 Workloads de IA, GPU, datos y MLOps multi-cloud

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega arquitectura-ia-cloud con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- ☐ Resultado reproducible por una segunda persona.
- ☐ Evidencia vinculada a cada afirmación técnica.
- ☐ Prueba negativa o de fallo incluida.
- ☐ Costos y permisos expresados con unidades y alcance.
- ☐ Limitaciones declaradas sin presentar la simulación como producción.

## Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 176 — Edge, IoT y procesamiento desconectado

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: edge · Estado: EXECUTABLECORE

## Propósito

Pasar de cientos de emplazamientos —la clase 165— a cientos de miles de dispositivos, donde atender uno por uno deja de ser posible y todo tiene que ser automático y estadístico. La clase se centra en lo que cambia con esa escala: cómo se le da identidad a un millón de cosas y cómo se le quita a una sola; el embudo de datos, porque el volumen se genera en el borde y casi todo sobra; la actualización por cohortes con parada automática, porque un fallo puede dejar dispositivos inalcanzables para siempre; y una aritmética que engaña, la de los costes por dispositivo que parecen ridículos hasta que se multiplican.

## Resultados de aprendizaje

Al finalizar podrás:

1. Reconocer qué cambia al pasar de cientos de sitios a cientos de miles de dispositivos.
2. Dar identidad a una flota y poder revocar la de uno solo.
3. Diseñar el embudo de datos sabiendo que descartar es irreversible.
4. Actualizar por cohortes, con parada automática y sin poder probarlo todo.
5. Calcular los costes que se multiplican por el número de dispositivos.

## Conceptos centrales

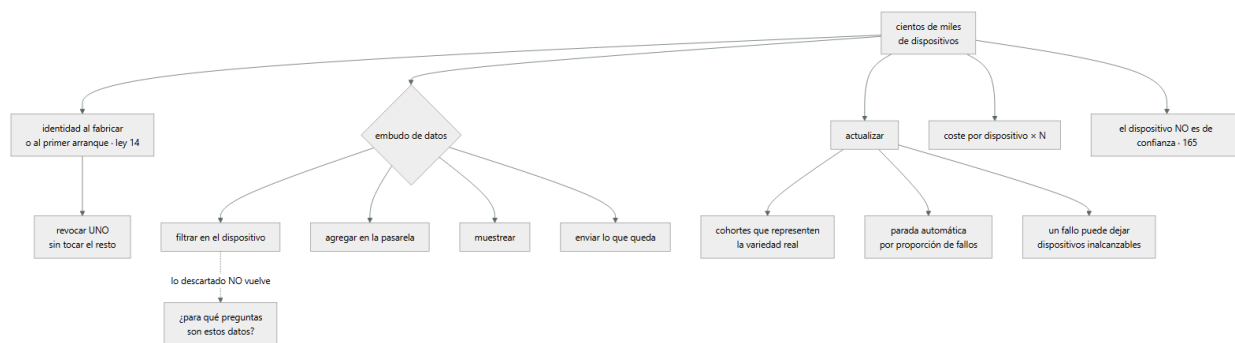
| Concepto                 | Comprensión verificable                                                                                                          |
|--------------------------|----------------------------------------------------------------------------------------------------------------------------------|
| escala estadística       | Régimen en el que las decisiones se toman sobre proporciones de la flota, porque atender a un dispositivo concreto no es viable. |
| identidad de fabricación | Credencial grabada en el dispositivo al fabricarlo. Es una decisión irreversible: dura lo que dura el dispositivo.               |
| embudo de datos          | Filtrado, agregación y muestreo antes de enviar. Lo descartado no vuelve, así que depende de qué preguntas se quieran responder. |
| cohorte                  | Subconjunto de la flota que recibe una actualización antes que el resto, elegido para que represente la variedad real.           |
| parada automática        | Detención del despliegue cuando la proporción de fallos supera un umbral, sin intervención humana.                               |
| coste por dispositivo    | Céntimos que se multiplican por el tamaño de la flota. Es donde se pierde el control del gasto en estas cargas.                  |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. Qué cambia con la escala

La clase 165 trataba trescientas cuarenta tiendas. Con cuatrocientos mil dispositivos, cambian cinco cosas:

#### ATENDER UNO DEJA DE SER VIABLE

- no hay nadie que mire un dispositivo concreto
- todo se decide sobre PROPORCIONES de la flota
- «el 0,3 % no reporta» es normal; «el 4 %» es un incidente

#### EL DISPOSITIVO ES BARATO Y LIMITADO

- poca memoria, poco cómputo, a veces sin sistema operativo completo
- no cabe un agente de observabilidad ni casi nada de lo de la parte 10

#### DURA MUCHOS AÑOS

- cinco, diez, quince
- sobrevive a los algoritmos de firma, a las versiones de protocolo y a las interfaces del proveedor

#### ES FÍSICAMENTE ACCESIBLE

- y en cantidades que hacen probable que alguien lo abra

#### Y EL COSTE SE MULTIPLICA

- céntimos por dispositivo y por mes son decenas de miles al año

Y la consecuencia de la primera, que reordena toda la operación:

- las alertas no son «este dispositivo falla»
- sino «la proporción de dispositivos que no reportan ha subido»
- y eso exige tener una línea base de lo normal clase 174

Y la de la tercera, que es una decisión de creación y por tanto irreversible —ley 14—:

- lo que se decida ahora sobre identidad, protocolo y formato seguirá vivo dentro de diez años, en dispositivos que nadie puede sustituir
- y hay que dejar sitio para cambiarlo: versión en el mensaje, posibilidad de rotar credenciales, y un mecanismo de actualización que funcione

Y la comprobación que conviene hacerse antes de nada, que es la de la clase 157 aplicada aquí:

- ¿qué dispositivos necesitan de verdad estar conectados?
- ¿qué decisiones tienen que tomarse en el dispositivo?
- ¿qué datos hacen falta en el centro, y para responder qué?
- y casi siempre la respuesta reduce mucho el problema

### 2. Identidad para un millón de cosas

Cada dispositivo necesita una identidad propia, y eso plantea tres preguntas distintas:

#### 1. ¿CUÁNDO SE LE DA?

##### AL FABRICAR

- una credencial única grabada en el dispositivo
- + no hay ventana de dispositivos sin identidad
- depende del fabricante y de su seguridad
- y dura lo que dure el dispositivo

ley 14

##### AL PRIMER ARRANQUE

el dispositivo se presenta con algo compartido y recibe la suya  
+ independiente del fabricante  
– ese «algo compartido» está en todos: si se filtra, sirve para registrar dispositivos falsos  
– y hay que acotar la ventana: registro solo permitido una vez por número de serie

## 2. ¿CÓMO SE REVOKA UNA SOLA?

un dispositivo robado o comprometido tiene que dejar de servir sin tocar a los demás  
→ lista de revocación, o credenciales de vida corta que dejan de renovarse clase 137  
→ y la segunda escala mucho mejor

## 3. ¿QUÉ PUEDE HACER?

un dispositivo debe poder escribir SUS datos y nada más  
→ y eso lo comprueba el centro, no el dispositivo clase 165  
→ si un dispositivo comprometido puede escribir datos de otro, el diseño está mal

Y la segunda pregunta merece énfasis porque a esta escala tiene una forma propia:

una lista de revocación con 400.000 entradas potenciales es un problema en sí misma  
→ por eso lo habitual es credencial de vida corta, renovada contra el centro  
→ y revocar es dejar de renovar  
→ con la consecuencia: un dispositivo desconectado mucho tiempo se queda sin credencial y hay que poder recuperarlo

Y una precaución sobre la duración, que es la tercera propiedad del apartado anterior:

el dispositivo durará más que el algoritmo con el que se firmó  
→ hay que poder cambiar de algoritmo y de credencial en campo  
→ y si no se puede, la vida del dispositivo la fija la criptografía, no el hardware

Y el modelo de confianza, que es el de la clase 165 sin cambios:

EL DISPOSITIVO NO ES DE CONFIANZA  
se valida en el centro lo que envía  
se acota lo que puede tocar  
y comprometer uno debe alcanzar solo a ese uno

## 3. El embudo de datos

El volumen se genera en el borde y casi todo sobra:

400.000 dispositivos × 1 medición por segundo  
= 400.000 mensajes/s  
= 34.500 millones al día

Y enviarlo todo es inviable por tres motivos a la vez: enlace, coste de ingesta y coste de almacenar.

El embudo, con sus cuatro pasos y su efecto típico:

1. FILTRAR EN EL DISPOSITIVO  
enviar solo cuando cambia, o cuando supera un umbral  
→ reducción de 10 a 100 veces, y es la mayor con diferencia
2. AGREGAR EN UNA PASARELA  
una pasarela por tienda o por planta resume lo de sus dispositivos  
→ medias, máximos, conteos por minuto  
→ reducción adicional de 10 a 60 veces
3. MUESTREAR  
conservar el detalle de una parte y el agregado de todo  
→ y conservar SIEMPRE lo anómalo clase 121
4. ENVIAR LO QUE QUEDA

Y la decisión que hay que tomar antes, porque lo descartado no vuelve:

¿PARA QUÉ PREGUNTAS SON ESTOS DATOS?

|                              |                                    |
|------------------------------|------------------------------------|
| vigilancia en tiempo real    | basta el agregado por minuto       |
| diagnóstico de una avería    | hace falta detalle, de ese momento |
| análisis histórico y modelos | hace falta muestra representativa  |

```

class 175

```

hace falta el dato original

Y la técnica que reconcilia lo irreconciliable:

GUARDAR EL DETALLE EN EL DISPOSITIVO O EN LA PASARELA, UNOS DÍAS  
y enviarlo solo bajo demanda cuando haga falta investigar  
→ el detalle existe donde se genera y no viaja salvo que se pida  
→ es el equivalente de «no mover los datos» de la clase 161

Y dos precauciones del transporte, que son la parte 09 con peores redes:

la cola local debe estar acotada y saber qué descartar      clase 165  
los mensajes llevan número de secuencia del propio dispositivo  
→ porque su reloj no es de fiar      clase 149  
→ y el orden se reconstruye con ese número, no con la hora  
y la reconexión masiva necesita variación aleatoria      clase 165

Y el coste, que es donde estas cargas se descontrolan:

coste por mensaje ingerido, por dispositivo y mes, y por gigabyte  
parecen céntimos  
× 400.000 dispositivos  
→ y una decisión de enviar una medición más por minuto puede  
costar decenas de miles al año

Y de ahí la regla: cada campo que se envía tiene un motivo escrito, y se revisa.

#### 4. Actualizar sin poder probarlo todo

La clase 165 estableció el modelo: el dispositivo tira, hay vigilante de arranque y se despliega por grupos. A esta escala se añaden tres cosas:

NO SE PUEDE PROBAR TODA LA VARIEDAD  
versiones de hardware, revisiones de fabricante, versiones  
de firmware anteriores, condiciones de red y de temperatura  
→ 400.000 dispositivos son decenas de combinaciones

LAS COHORTES DEBEN REPRESENTAR ESA VARIEDAD  
no vale «los primeros mil»: vale «mil elegidos para cubrir  
todas las combinaciones conocidas»

Y LA PARADA DEBE SER AUTOMÁTICA  
nadie puede vigilar un despliegue de tres días  
→ «si la proporción de dispositivos que no completan supera  
el 1 %, se detiene»

Y el escalonado típico:

|           |                                                      |      |
|-----------|------------------------------------------------------|------|
| cohorte 0 | dispositivos de laboratorio, con todas las variantes |      |
| cohorte 1 | 0,1 % de la flota, representativa                    | 24 h |
| cohorte 2 | 1 %                                                  | 24 h |
| cohorte 3 | 10 %                                                 | 48 h |
| cohorte 4 | el resto, por lotes                                  |      |

Y las comprobaciones que decide cada etapa:

proporción que completa la actualización  
proporción que vuelve a reportar después  
proporción que revierte por el vigilante  
y las métricas propias del dispositivo: consumo, memoria, errores

Y el riesgo que no existe en la nube y que hay que aceptar por escrito:

un dispositivo que no arranca puede quedar inalcanzable PARA SIEMPRE  
si no hay quien vaya, o si el coste de ir supera al del dispositivo  
→ por eso el vigilante de arranque no es opcional  
→ y por eso conviene mantener dos particiones: la nueva y la anterior

Y una precaución sobre el propio mecanismo de actualización:

- lo único que NUNCA debe romperse es la capacidad de actualizar
- el componente que la implementa se toca lo mínimo posible
- y sus cambios se despliegan con más cuidado que ningún otro
- si se rompe, la flota queda congelada en la versión actual

Lo que se vigila de una flota, todo en proporciones:

dispositivos que reportan, frente a los esperados  
distribución de versiones, y cuántos llevan más de N meses sin actualizar  
proporción que ha revertido en el último despliegue  
latencia de mensajes, por percentil y por región  
volumen enviado por dispositivo, y su tendencia  
dispositivos que envían datos que no les corresponden  
y coste por dispositivo y mes

Y la alerta que la ley 13 impone, en su forma estadística:

no «este dispositivo dejó de reportar»  
sino «la proporción que no reporta ha subido por encima de lo normal»  
→ y el desglose por versión, región y modelo, que suele señalar  
la causa de inmediato

Y la lista de comprobación de la clase:

- está escrito qué dispositivos necesitan estar conectados y para qué
- cada dispositivo tiene identidad propia, y está decidido cuándo se le da
- se puede revocar uno solo sin tocar a los demás
- un dispositivo solo puede escribir sus datos, y lo comprueba el centro
- se puede cambiar de algoritmo y de credencial en campo
- existe embudo de datos, y está escrito para qué preguntas sirve cada capa
- el detalle se conserva cerca del origen y viaja solo bajo demanda
- los mensajes llevan número de secuencia propio
- las colas locales están acotadas, con criterio de descarte
- la reconexión masiva tiene variación aleatoria y límite de caudal
- las cohortes representan la variedad real, no las primeras unidades
- hay parada automática por proporción de fallos
- hay vigilante de arranque y dos particiones
- el mecanismo de actualización se toca lo mínimo y con más cuidado
- las alertas son sobre proporciones, con línea base
- está calculado el coste por dispositivo y mes
- cada campo enviado tiene un motivo escrito

Y el cierre que enlaza con la clase siguiente: buena parte de las decisiones de esta clase y de la 165 —dónde vive el dato, quién puede verlo, quién opera el sistema— dejan de ser técnicas cuando aparecen exigencias de soberanía. Qué resuelven de verdad esas exigencias, y qué se vende bajo ese nombre, es la materia de la clase 177.

## Ejemplo trabajado

CloudShop instala 410.000 sensores en tiendas y almacenes: temperatura, presencia, peso en estantería y estado de los equipos de frío. El ejercicio empieza con una factura de ingesta que nadie esperaba y termina con una actualización detenida automáticamente en el 2 %.

El primer diseño, y su factura.

|                          |                 |
|--------------------------|-----------------|
| dispositivos             | 410.000         |
| mensajes por dispositivo | 1 por segundo   |
| mensajes al día          | 35.400 millones |
| tamaño medio del mensaje | 180 B           |
| coste mensual            |                 |
| ingesta                  | 41.000 €        |
| almacenamiento (90 días) | 6.200 €         |
| conectividad             | 8.900 €         |
|                          | ■■■■■■■■■■      |
|                          | 56.100 €        |

Y la pregunta del apartado tercero, hecha por primera vez:

|                                           |                         |
|-------------------------------------------|-------------------------|
| ¿para qué preguntas son estos datos?      |                         |
| vigilar que la cadena de frío no se rompa | agregado por minuto     |
| avisar de una avería                      | evento, no serie        |
| analizar consumo y planificar             | serie por hora          |
| demostrar cumplimiento sanitario          | serie por 5 minutos,    |
|                                           | conservada 2 años       |
| diagnosticar una avería concreta          | detalle por segundo,    |
|                                           | de ESE equipo y ESE día |

Y ninguna de las cinco necesitaba un mensaje por segundo de todos los dispositivos en el centro.

El embudo, aplicado.

|                                                                  |                            |              |
|------------------------------------------------------------------|----------------------------|--------------|
| 1. FILTRAR EN EL DISPOSITIVO                                     |                            |              |
| enviar cuando el valor cambia más de un umbral, o cada 60 s      |                            |              |
| mensajes                                                         | 35.400 M/día → 1.180 M/día |              |
| reducción                                                        |                            | x30          |
| 2. AGREGAR EN PASARELA (una por tienda o planta)                 |                            |              |
| media, mínimo, máximo y conteo por minuto, por grupo de sensores |                            |              |
| mensajes                                                         | 1.180 M/día → 42 M/día     |              |
| reducción adicional                                              |                            | x28          |
| 3. EVENTOS SIN AGREGAR                                           |                            |              |
| cualquier lectura fuera de rango se envía entera y de inmediato  |                            |              |
| mensajes                                                         |                            | ~180.000/día |
| 4. DETALLE, EN LA PASARELA                                       |                            |              |
| 7 días de detalle por segundo, en disco local                    |                            |              |
| se sube solo bajo demanda cuando hay que investigar              |                            |              |
| peticiones de detalle al mes                                     |                            | ~40          |
| volumen subido por esas peticiones                               |                            | ~11 GB/mes   |
| mensajes al día                                                  | 35.400 M                   | 42 M         |
| coste de ingesta                                                 | 41.000 €                   | 1.100 €      |
| almacenamiento                                                   | 6.200 €                    | 890 €        |
| conectividad                                                     | 8.900 €                    | 1.400 €      |
|                                                                  | ██████████                 | ██████████   |
|                                                                  | 56.100 €                   | 3.390 €      |

Y lo que se conserva para cumplimiento, decidido aparte:

|                                                    |           |
|----------------------------------------------------|-----------|
| serie por 5 minutos de los equipos de frío, 2 años |           |
| volumen                                            | 1,4 TB    |
| coste                                              | 31 €/mes  |
| en formato columnar y particionado                 | clase 112 |

La identidad.

primer diseño una credencial compartida por modelo de dispositivo  
→ 4 credenciales para 410.000 dispositivos

lo que eso permitía  
un dispositivo comprometido podía suplantar a cualquier otro  
del mismo modelo  
no se podía revocar uno solo  
y no se podía saber qué dispositivo envió qué

|                                             |            |                                                       |
|---------------------------------------------|------------|-------------------------------------------------------|
| credenciales                                | 4          | 410.000                                               |
| cuándo se asigna                            | –          | al primer arranque,<br>una vez por número<br>de serie |
| vida de la credencial                       | indefinida | 30 días,<br>renovada al conectar<br>dejar de renovar  |
| revocar uno                                 | imposible  |                                                       |
| un dispositivo puede escribir datos de otro | sí         | no                                                    |
| → comprobado en el centro                   |            | clase 165                                             |

Y el problema de los desconectados, que apareció al mes:

dispositivos que no conectaron en 30 días 1.180  
→ credencial caducada; no podían volver  
causa almacenes de temporada, cerrados fuera de campaña

corrección

ventana de gracia: un dispositivo con credencial caducada  
menos de 180 días puede renovar presentando su identidad  
de fabricación, con registro y alerta  
dispositivos recuperados así en 12 meses 1.940  
intentos rechazados por exceder los 180 días 11  
→ investigados uno a uno: 9 eran dispositivos retirados

La actualización que se detuvo sola.

|                                |    |
|--------------------------------|----|
| variedad de la flota           |    |
| modelos de hardware            | 6  |
| revisiones de fabricante       | 14 |
| versiones de firmware en campo | 9  |
| combinaciones conocidas        | 31 |

|           |                                                    |      |
|-----------|----------------------------------------------------|------|
| cohorte 0 | laboratorio, 62 dispositivos, las 31 combinaciones |      |
| cohorte 1 | 410 dispositivos (0,1 %), representativos          | 24 h |
| cohorte 2 | 4.100 (1 %)                                        | 24 h |
| cohorte 3 | 41.000 (10 %)                                      | 48 h |
| cohorte 4 | el resto                                           |      |

Y lo que ocurrió en el segundo despliegue del año:

|           |                                         |        |
|-----------|-----------------------------------------|--------|
| cohorte 0 | correcta                                |        |
| cohorte 1 | correcta; 410 de 410 completan          |        |
| cohorte 2 | a las 6 h, 89 dispositivos no completan |        |
|           | proporción                              | 2,17 % |
|           | umbral de parada                        | 1,00 % |
|           | → DETENIDO AUTOMÁTICAMENTE              |        |

#### investigación

los 89 eran de una revisión de fabricante concreta  
presente en el 0,4 % de la flota  
y solo 2 de esa revisión estaban en la cohorte 1  
→ la cohorte 1 no la representaba bien

#### corrección

las cohortes pasan a garantizar un mínimo de 20 unidades  
por combinación conocida  
y se corrigió el firmware para esa revisión

dispositivos que quedaron inservibles 0  
→ el vigilante de arranque revirtió los 89

Ochenta y nueve dispositivos revertidos automáticamente en vez de ocho mil ochocientos, que es lo que habría llegado si el despliegue hubiera seguido a la cohorte 3.

|                                                       |                   |                           |
|-------------------------------------------------------|-------------------|---------------------------|
| parada automática                                     | no había          | 1 % por cohorte           |
| cohortes representativas                              | primeras unidades | mínimo 20 por combinación |
| vigilante de arranque                                 | sí                | sí                        |
| dispositivos inservibles por actualización, histórico | 41                | → 0                       |

Y una decisión que se tomó tras esto:

el componente que gestiona la actualización se congeló  
cambios en 12 meses 1  
ese cambio se desplegó con cohortes de 10 unidades y 7 días  
entre etapas  
motivo si se rompe, la flota queda congelada y no hay forma  
de arreglarla a distancia

#### La vigilancia en proporciones.

|                                                    |           |
|----------------------------------------------------|-----------|
| línea base, medida durante 6 semanas               |           |
| dispositivos que no reportan en 1 h                | 0,4 %     |
| variación normal                                   | ±0,15 %   |
| alertas definidas                                  |           |
| proporción sin reportar > 1 %                      | aviso     |
| proporción sin reportar > 3 %                      | incidente |
| y desglose automático por versión, modelo y región |           |

Y las tres veces que saltó en un año:

vez 1 4,1 % sin reportar; desglose: todos de una región  
→ corte del operador de conectividad; nada que hacer  
→ detectado 40 min antes de que el operador lo comunicara

vez 2 2,2 %; desglose: todos de una versión de firmware  
→ una fuga de memoria que reiniciaba el dispositivo  
→ corregida y desplegada por cohortes

vez 3 1,4 %; desglose: sin patrón  
→ falsa alarma: coincidió con el cierre estacional de  
almacenes; la línea base no lo contemplaba  
→ línea base ajustada por temporada

#### El coste por dispositivo, vigilado.

|                     |          |         |
|---------------------|----------|---------|
| coste mensual total | 56.100 € | 3.390 € |
|---------------------|----------|---------|

|                                                   |          |           |
|---------------------------------------------------|----------|-----------|
| coste por dispositivo y mes                       | 0,137 €  | 0,0083 €  |
| campos enviados por mensaje                       | 14       | 6         |
| → 8 se retiraron: nadie los consultaba en 90 días |          |           |
| revisión de campos enviados                       | no había | semestral |

Y el ejercicio de retirar campos dio la cifra más ilustrativa:

un campo adicional de 8 bytes por mensaje  
 × 42 millones de mensajes al día  
 × 30 días  
 = 10 GB al mes  
 → parece nada, y ocho campos así eran el 57 % del volumen

A los doce meses.

|                                            |                 |              |
|--------------------------------------------|-----------------|--------------|
| dispositivos                               | 410.000         | 410.000      |
| mensajes al día                            | 35.400 M        | 42 M         |
| coste mensual                              | 56.100 €        | 3.390 €      |
| coste por dispositivo y mes                | 0,137 €         | 0,0083 €     |
| credenciales                               | 4               | 410.000      |
| revocar un dispositivo                     | imposible       | inmediato    |
| dispositivos que pueden suplantar a otros  | todos           | 0            |
| cohortes representativas                   | no              | sí           |
| parada automática                          | no              | sí           |
| dispositivos inservibles por actualización | 41              | 0            |
| alertas por proporción con línea base      | no              | sí           |
| detección de un corte regional             | por el operador | 40 min antes |

La lección que esta clase traslada a la parte 14: la factura inicial era de cincuenta y seis mil euros al mes porque nadie había preguntado para qué preguntas servían los datos; al responderlo, el 99,9 % de los mensajes dejó de viajar y el detalle se quedó a siete días de distancia, en la pasarela, disponible bajo demanda. Y la actualización que se detuvo sola en el 2 % lo hizo porque la cohorte de prueba contenía dos unidades de una revisión que representaba al 0,4 % de la flota: a esta escala no se puede probar todo, y por eso las cohortes se construyen para cubrir la variedad y la parada es automática.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/176-edge-iot-y-procesamiento-desconectado/lab.py
```

El laboratorio selecciona el motor de práctica edge y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa arquitectura-edge en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

### Evidencia esperada

El artefacto principal es un flujo edge que tolera desconexión y sincroniza estado. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye arquitectura-edge para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

### ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.

- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■ Errores frecuentes

| Síntoma                                                                    | Causa probable                                                               | Corrección                                                                                                                              |
|----------------------------------------------------------------------------|------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| La factura de ingesta es enorme desde el primer mes                        | Se envía todo lo que los dispositivos generan sin preguntarse para qué sirve | Escribe las preguntas que deben responderse y construye el embudo: filtrar, agregar, muestrear y conservar el detalle cerca del origen. |
| Un dispositivo comprometido puede suplantar a cualquier otro               | Credenciales compartidas por modelo                                          | Identidad propia por dispositivo, de vida corta y renovable, con comprobación en el centro de qué puede escribir cada uno.              |
| Dispositivos que estuvieron mucho tiempo apagados no pueden volver         | Su credencial caducó y no hay forma de renovarla                             | Ventana de gracia con la identidad de fabricación, con registro y alerta, y límite de tiempo.                                           |
| Una actualización deja miles de dispositivos inservibles                   | Se desplegó sin cohortes representativas ni parada automática                | Cohortes con un mínimo por combinación conocida, parada por proporción de fallos y vigilante de arranque con dos particiones.           |
| El mecanismo de actualización deja de funcionar y la flota queda congelada | Se cambió como cualquier otro componente                                     | Tócalo lo mínimo y despliega sus cambios con cohortes más pequeñas y más tiempo entre etapas que ningún otro.                           |
| Un campo pequeño añadido al mensaje dispara el coste                       | Se multiplica por el número de dispositivos y de mensajes                    | Cada campo con motivo escrito, revisión periódica de lo enviado y vigilancia del coste por dispositivo y mes.                           |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Qué cinco cosas cambian al pasar de cientos de sitios a cientos de miles de dispositivos?
2. ¿Qué compromiso tiene dar la identidad al fabricar frente a darla al primer arranque?
3. ¿Por qué hay que decidir para qué preguntas sirven los datos antes de construir el embudo?
4. ¿Cómo se construye una cohorte y por qué no valen las primeras unidades?
5. ¿Por qué el componente de actualización merece un trato distinto a los demás?

## Referencias

- AWS (2025). IoT device provisioning and fleet management — identidad por dispositivo y despliegue por cohortes. <<https://docs.aws.amazon.com/iot/latest/developerguide/iot-provision.html>>
- Azure (2025). IoT Hub device provisioning service and device update — registro, revocación y actualización escalonada. <<https://learn.microsoft.com/azure/iot-dps/about-iot-dps>>
- Google Cloud (2025). Edge data processing patterns — filtrar y agregar antes de enviar. <<https://cloud.google.com/architecture/connected-devices>>

- IETF (2025). MQTT y CoAP: protocolos para dispositivos limitados — transporte con recursos escasos. <<https://datatracker.ietf.org/doc/html/rfc7252>>
- NIST (2022). IoT device cybersecurity guidance — identidad, actualización y acceso físico. <<https://csrc.nist.gov/pubs/ir/8259/final>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                                | Índice              | Siguiente                                          |
|---------------------------------------------------------|---------------------|----------------------------------------------------|
| ← 175 · Workloads de IA, GPU, datos y MLOps multi-cloud | Parte 14 · Programa | 177 · Soberanía digital y confidential computing → |

## Evaluación — 176 Edge, IoT y procesamiento desconectado

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega arquitectura-edge con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- ☐ Resultado reproducible por una segunda persona.
- ☐ Evidencia vinculada a cada afirmación técnica.
- ☐ Prueba negativa o de fallo incluida.
- ☐ Costos y permisos expresados con unidades y alcance.
- ☐ Limitaciones declaradas sin presentar la simulación como producción.

### Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 177 — Soberanía digital y confidential computing

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 4 Laboratorio: security · Estado: EXECUTABLECORE

## Propósito

Separar las cuatro exigencias distintas que se venden bajo una sola palabra —soberanía— y responder a cada una con lo que de verdad la resuelve. La clase sostiene que la mayoría de estas conversaciones confunden un requisito de residencia con uno jurisdiccional, que solo los dos primeros tienen respuesta técnica limpia, y que el último no la tiene en absoluto. Después trata con honestidad la tecnología que más se cita en estos casos —el cómputo con memoria cifrada y atestación—, diciendo qué protege, qué cuesta y contra qué no sirve.

## Resultados de aprendizaje

Al finalizar podrás:

1. Separar residencia, control operativo, independencia técnica y jurisdicción.
2. Preguntar contra quién se protege, antes de elegir ninguna solución.
3. Situar el cómputo confidencial por lo que hace y por lo que no hace.
4. Ordenar las opciones de gestión de claves por lo que dejan de poder hacer al proveedor.
5. Responder a un requisito de soberanía con evidencia, que es lo que casi siempre se pide de verdad.

## Conceptos centrales

| Concepto              | Comprensión verificable                                                                                                                |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| residencia            | Los datos permanecen en una geografía. Es la exigencia más citada y la que tiene respuesta técnica más clara.                          |
| control operativo     | Quién puede acceder al sistema y desde dónde, incluido el personal del proveedor.                                                      |
| independencia técnica | Poder seguir funcionando sin el proveedor. Se mide como coste de salida, no como una propiedad binaria.                                |
| jurisdicción          | Qué derecho puede obligar a entregar datos. Ningún control técnico la elimina; se acota con dónde vive la clave y quién la controla.   |
| cómputo confidencial  | Ejecución con la memoria cifrada y aislada del sistema anfitrión, con una prueba verificable de qué código se está ejecutando.         |
| atestación            | Prueba criptográfica de qué código y qué configuración se están ejecutando. Es la parte que aporta la garantía, y la que hay que usar. |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. Cuatro exigencias, no una

«Soberanía» se usa para cuatro cosas distintas, y confundirlas produce compras caras que no resuelven lo que preocupaba:

#### 1. RESIDENCIA

los datos permanecen en una geografía

→ respuesta técnica clara: regiones, políticas, y comprobar registros, telemetría, copias y soporte clase 141

→ es lo que más se pide y lo que más veces se incumple sin saberlo

#### 2. CONTROL OPERATIVO

quién puede acceder al sistema, incluido el personal del proveedor, desde dónde y con qué registro

→ respuesta técnica PARCIAL: aprobación de accesos, transparencia de acceso, personal local, y claves fuera del alcance del proveedor

#### 3. INDEPENDENCIA TÉCNICA

poder seguir funcionando sin el proveedor

→ no es binaria: es el coste de salida de la clase 158

→ y se responde con una cifra en semanas, no con una promesa

#### 4. JURISDICCIÓN

qué derecho puede obligar a entregar datos, y a quién

→ NO tiene respuesta técnica

→ se acota con dónde está constituida la entidad que opera, con el contrato y con quién controla la clave

Y la observación que ordena la clase:

casi todas las conversaciones empiezan hablando de la 1

y lo que preocupa de verdad es la 4

→ o al revés: se compra por la 4 lo que solo resuelve la 1 y la 2

Y las preguntas que separan las cuatro, y que conviene hacer al principio:

¿qué pasa si estos datos salen de este territorio? → residencia

¿te preocupa que alguien del proveedor pueda leerlos? → operativo

¿qué pasa si mañana no puedes usar este proveedor? → independencia

¿te preocupa que un tribunal extranjero pueda exigirlos? → jurisdicción

Y una precisión importante sobre la cuarta:

si la entidad que opera está sujeta a un derecho, ese derecho puede alcanzarla, esté donde estén los servidores

→ y por eso las ofertas «soberanas» se construyen sobre entidades locales, personal local y separación societaria

→ que son respuestas jurídicas y organizativas, no técnicas

Y lo único que sí acota técnicamente el alcance de un requerimiento:

que el proveedor no pueda descifrar

→ y eso exige que la clave esté fuera de su alcance

→ que es el apartado tercero

### 2. Contra quién se protege

Antes de elegir tecnología, la pregunta de la clase 140:

#### OTRO CLIENTE DEL PROVEEDOR

→ aislamiento normal; ya está resuelto clase 154

→ y es una preocupación legítima que casi nunca es la real

#### UN ATACANTE EXTERNO

- toda la parte 11
- el cómputo confidencial ayuda poco aquí: si el atacante compromete tu aplicación, esta descifra por él

#### UN ATACANTE QUE COMPROMETE AL PROVEEDOR

- aquí sí ayudan las claves fuera de su alcance

#### PERSONAL DEL PROVEEDOR

- administradores, soporte, mantenimiento
- controles operativos: aprobación de accesos, registro visible para el cliente, personal en una geografía concreta
- y claves que el proveedor no puede usar

#### UN REQUERIMIENTO LEGAL AL PROVEEDOR

- contrato, jurisdicción de la entidad, y de nuevo la clave
- si el proveedor no puede descifrar, lo que entregue no sirve

### Y la comprobación honesta que conviene hacerse:

- ¿ha habido algún incidente real de la amenaza que nos preocupa?
- ¿qué probabilidad tiene frente a las que ya nos han ocurrido?
- y comparar el coste de la solución con el de los riesgos que sí se han materializado clase 133

### Las opciones de gestión de claves, ordenadas por lo que dejan de poder hacer al proveedor:

#### CLAVES DEL PROVEEDOR

- el proveedor genera, guarda y usa la clave sí
- puede descifrar

#### CLAVES GESTIONADAS POR EL CLIENTE

- el cliente controla la política y puede revocar clase 136
- la clave sigue en el servicio del proveedor
- puede descifrar mientras la política lo permita sí
- aporta: control, auditoría y capacidad de revocar

#### CLAVE EXTERNA, FUERA DEL PROVEEDOR

- la clave vive en un sistema del cliente; el proveedor la pide para cada operación
- puede descifrar sin permiso del cliente no
- coste: latencia, disponibilidad del sistema de claves y un punto de fallo nuevo clase 136

#### CIFRADO EN EL CLIENTE

- el proveedor nunca ve el contenido
- puede descifrar no
- coste: no se puede buscar, ordenar ni procesar en el proveedor

### Y la observación práctica que suele decidir:

- las dos últimas resuelven la amenaza «el proveedor puede leer»
- y hacen inviable casi todo lo que hace útil una nube: bases gestionadas, análisis, búsqueda clase 158
- por eso se aplican a CAMPOS o a CONJUNTOS concretos, no a todo

## 3. Cómputo confidencial, sin exageraciones

### Qué hace, con precisión:

- la memoria del proceso está cifrada por el hardware
- el sistema anfitrión y quien lo administra no pueden leerla
- y la ATESTACIÓN produce una prueba criptográfica de qué código y qué configuración se están ejecutando

### Y la segunda mitad es la que aporta la garantía y la que más se ignora:

- sin verificar la atestación, se está confiando igual que antes
- la garantía no es «está cifrado»: es «puedo comprobar QUÉ se ejecuta»
- y esa comprobación la tiene que hacer alguien, con una política de qué medidas acepta

### Qué no hace, que es lo que hay que decir en voz alta:

- no protege de fallos de tu propio código
- una inyección o un permiso mal puesto funcionan igual dentro
- no protege si tu aplicación descifra y luego lo registra clase 122

no elimina la confianza: la traslada al fabricante del hardware y a su cadena de atestación

no resuelve la jurisdicción por sí solo  
si la clave la controla el proveedor, puede haber una vía

y no es gratis  
penalización de rendimiento, tipos de instancia limitados, disponibilidad desigual por región, y complejidad de operación

Y los usos donde sí resuelve algo que ninguna otra cosa resuelve:

VARIAS PARTES QUE NO SE FÍAN ENTRE SÍ  
dos empresas quieren calcular algo sobre sus datos combinados sin enseñárselos  
→ cada una comprueba la atestación antes de aportar su clave

PROCESAR DATOS QUE NO SE PUEDEN VER  
un tercero procesa datos del cliente sin poder leerlos

EXIGENCIA NORMATIVA EXPLÍCITA  
cuando una norma o un contrato lo pide, aunque el riesgo sea discutible

Y REDUCIR EL ALCANCE DE UN COMPROMISO DEL ANFITRIÓN  
en entornos compartidos con requisitos altos

Y el coste de adoptarlo, que hay que estimar antes:

|                                                                    |                                                             |
|--------------------------------------------------------------------|-------------------------------------------------------------|
| rendimiento                                                        | penalización variable, medible                              |
| tipos de instancia disponibles                                     | menos, y no en todas las regiones                           |
| servicios gestionados compatibles                                  | pocos: se pierde parte de lo que hace útil la nube          |
| atestación                                                         | hay que verificarla, con política y con rotación de medidas |
| depuración y observabilidad                                        | más difícil por diseño                                      |
| y el modelo de amenazas debe ESCRIBIRSE, o no se sabrá qué se ganó |                                                             |

Y la comprobación que decide si merece la pena:

¿qué amenaza concreta elimina, que no elimine algo más barato?  
→ si la respuesta es «que el personal del proveedor lea la memoria», hay que compararla con las amenazas que sí se han materializado  
→ y si la respuesta es «lo exige el contrato», es una respuesta válida y conviene decirlo así

## 4. Lo que casi siempre se pide de verdad

Al desmontar un requisito de soberanía, lo que suele quedar es esto:

saber DÓNDE está cada dato, y poder demostrarlo  
saber QUIÉN ha accedido, y poder demostrarlo  
saber QUÉ pasaría si hubiera un requerimiento legal  
y poder salir si hiciera falta, con una cifra

Y las cuatro son la clase 141 y la 158, no una tecnología nueva:

|                                                             |                 |
|-------------------------------------------------------------|-----------------|
| DÓNDE                                                       | clase 141       |
| inventario por categoría y por cliente                      |                 |
| incluidos registros, telemetría, copias y soporte           |                 |
| política que restringe regiones                             | clase 139       |
| y comprobación automática, no una declaración               |                 |
| QUIÉN                                                       | clases 134, 141 |
| registro de auditoría inalterable y consultable             |                 |
| incluido el acceso del personal del proveedor, si lo ofrece |                 |
| y correlación de identidades entre proveedores              | clase 159       |
| QUÉ PASARÍA                                                 |                 |
| qué entidad opera el servicio y bajo qué derecho            |                 |
| qué se comprometería a notificar y en qué plazo             |                 |
| y si el proveedor puede o no descifrar                      |                 |
| SALIR                                                       |                 |
| coste de salida por carga, en semanas                       | clase 158       |

Y la forma de responder que funciona con un cliente o un auditor:

no «cumplimos con la soberanía»  
sino  
«estos datos viven aquí, y esta consulta lo demuestra para cualquier fecha del último año»  
«estas personas y estos sistemas han accedido, y aquí está el registro»  
«esto es lo que el proveedor puede y no puede descifrar»  
«y salir de aquí nos costaría estas semanas»

Y las decisiones que sí conviene tomar, ordenadas por lo que aportan frente a lo que cuestan:

1. RESIDENCIA COMPROBADA, incluidos registros y telemetría barato, y es lo que más veces se incumple sin saberlo
2. CLAVES GESTIONADAS POR EL CLIENTE, con revocación barato, y da control y auditoría clase 136
3. REGISTRO DE ACCESO DEL PROVEEDOR, si se ofrece barato, y responde a «quién»
4. CIFRADO EN EL CLIENTE PARA CAMPOS CONCRETOS caro en funcionalidad; se aplica a pocos campos
5. CLAVE EXTERNA caro en disponibilidad y latencia; resuelve la amenaza del proveedor que descifra
6. CÓMPUTO CONFIDENCIAL caro y limitado; resuelve casos concretos
7. OFERTA SOBERANA COMPLETA la más cara: menos servicios, más latencia y coste de salida mayor; se elige cuando lo exige una norma o un contrato

Y las preguntas que hay que hacerle a cualquier oferta soberana antes de firmarla:

¿quién tiene acceso administrativo, y desde qué país?  
¿qué ocurre ante un requerimiento legal, y a quién se notifica?  
¿qué servicios NO están disponibles respecto de la región normal?  
¿qué diferencia de latencia y de coste hay?  
¿quién opera el hardware y bajo qué contrato?  
¿y cuál es el coste de salida desde ahí? clase 158

Y la lista de comprobación de la clase:

- están separadas las cuatro exigencias y se sabe cuál se pide
- está escrito contra quién se protege
- se ha comparado con las amenazas que sí se han materializado
- la residencia está comprobada, incluidos registros y telemetría
- hay claves gestionadas por el cliente con revocación
- se sabe qué puede y qué no puede descifrar el proveedor
- hay registro de acceso del proveedor, si se ofrece
- si hay cómputo confidencial, se verifica la atestación con política
- está escrito qué amenaza elimina y cuál no
- está estimado el coste de salida de la opción soberana
- la respuesta a un auditor es una consulta, no una afirmación

Y el cierre que enlaza con la clase siguiente: con esto termina el material de la parte 14. Las tres clases restantes son el proyecto final del programa: descubrir y diseñar, implementar y operar, y defender lo hecho. La primera es la materia de la clase 178.

## Ejemplo trabajado

Un cliente del sector público exige a CloudShop «una solución soberana». El ejercicio consiste en averiguar qué pide exactamente, y termina con tres decisiones baratas y una compra descartada.

La conversación inicial y las cuatro preguntas.

lo que decía el pliego  
«los datos deberán alojarse en infraestructura soberana, con garantías de que no puedan ser accedidos por terceros ni por autoridades extranjeras»

las cuatro preguntas, hechas en una reunión de una hora

¿qué pasa si estos datos salen del país?

→ incumplimiento del pliego; el contrato se anula

→ RESIDENCIA: exigencia real y dura

¿os preocupa que alguien del proveedor pueda leerlos?

→ «no habíamos pensado en eso, pero sí»

→ CONTROL OPERATIVO: preocupación real, sin requisito escrito

¿qué pasa si mañana no podemos usar este proveedor?

→ «tendríais que seguir prestando el servicio»

→ INDEPENDENCIA: requisito de continuidad, no de tecnología

¿os preocupa un requerimiento de un tribunal extranjero?

→ «es lo que motivó el párrafo»

→ JURISDICCIÓN: la exigencia de fondo

La cuarta era la que importaba, y el pliego la había escrito como si fuera la primera.

Lo que se ofreció, por capas.

#### RESIDENCIA

región del país, ya disponible

comprobación de los nueve destinos clase 141

→ 3 incumplían: telemetría, registros y el proveedor de correo

corregidos: coste adicional +180 €/mes

y comprobación automática semanal, con evidencia consultable

#### CONTROL OPERATIVO

claves gestionadas por el cliente, con revocación clase 136

registro de acceso del personal del proveedor, activado

→ y una consulta que el cliente puede ejecutar por su cuenta

coste adicional +40 €/mes

#### INDEPENDENCIA

coste de salida de esa carga, estimado clase 158

9 semanas y ~1.200 € de transferencia

documentado en el contrato, con revisión anual

#### JURISDICCIÓN

→ aquí no había respuesta técnica completa

La cuarta, tratada con honestidad.

lo que se puede afirmar

el proveedor no puede descifrar si la clave está fuera de su alcance

la entidad que opera está sujeta a un derecho concreto

y el contrato fija qué se notifica y en qué plazo

lo que NO se puede afirmar

que ningún derecho pueda alcanzar nunca a ninguna de las partes

Y las tres opciones evaluadas para esa exigencia:

#### A · CLAVE EXTERNA, en un sistema del cliente

el proveedor no puede descifrar sin una llamada al cliente

coste +2.100 €/mes y un punto de fallo nuevo

latencia +8 ms por operación con datos cifrados

disponibilidad la del sistema de claves del cliente

→ si cae, el servicio no funciona

#### B · CIFRADO EN EL CLIENTE, campos concretos

5 campos identificativos, cifrados antes de salir

coste desarrollo, 3 semanas

consecuencia esos campos no se pueden buscar ni ordenar

→ 2 informes hubo que rehacerlos

#### C · OFERTA SOBERANA COMPLETA del proveedor

entidad local, personal local, separación

coste +180 % sobre el precio de la región normal

servicios no disponibles 14 de los 31 que usábamos

latencia +22 ms

coste de salida desde ahí estimado 6 meses

Y la decisión, tomada con el cliente delante:

se eligieron A parcial + B

A solo para el conjunto de datos identificativos, no para todo

B para los 5 campos que el pliego menciona expresamente

se descartó C

motivo escrito: los 14 servicios ausentes obligaban a construir a mano lo que ya funcionaba, con más riesgo operativo que el que se pretendía evitar

y se documentó lo que NO se garantiza

→ y el cliente lo aceptó por escrito, que es lo que permitió firmar

El cómputo confidencial, evaluado y descartado.

El equipo comercial lo propuso como argumento. La evaluación:

¿qué amenaza elimina?

que el personal del proveedor lea la memoria del proceso

¿está esa amenaza en el modelo del cliente?

sí, era la preocupación número dos

¿qué la elimina más barato?

el cifrado en el cliente para los campos sensibles ya la elimina para esos datos (opción B)

¿qué añadiría el cómputo confidencial?

proteger también los datos NO cifrados en cliente mientras se procesan

¿qué cuesta?

|                                              |                        |
|----------------------------------------------|------------------------|
| penalización de rendimiento medida           | -14 %                  |
| tipos de instancia disponibles en esa región | 2 de 9                 |
| servicios gestionados compatibles            | 3 de 11                |
| verificación de atestación                   | a construir, 4 semanas |
| y rotación de medidas al actualizar          | trabajo permanente     |

Y la decisión:

no se adoptó

motivo escrito la amenaza que elimina de más está cubierta por B para los datos que la exigencia menciona

revisar si aparece un requisito explícito, o si se procesan datos que no se pueden cifrar en cliente porque hay que buscarlos o agregarlos

Y un uso donde sí se adoptó, dos años después:

tres empresas del sector quisieron calcular una estadística conjunta sin enseñarse los datos

→ cada una verifica la atestación antes de aportar su clave

→ es el caso en que ninguna otra tecnología lo resuelve

La respuesta al auditor.

La auditoría del cliente llegó nueve meses después:

pregunta «demuestren que los datos no han salido del país en los últimos 12 meses»

respuesta antes de este trabajo

una declaración firmada y capturas de la consola

respuesta ahora

una consulta sobre el registro de configuración y el de auditoría ejecutada delante del auditor

cubriendo los nueve destinos, día a día

tiempo empleado 6 min

observaciones del auditor 0

Y la segunda pregunta, que fue la difícil:

pregunta «¿puede el proveedor acceder a estos datos?»

respuesta

«a los cinco campos que su pliego señala, no: están cifrados antes de salir de nuestros sistemas y la clave no sale de aquí»  
 «al resto, técnicamente sí podría con una orden judicial dirigida a la entidad que opera, que está sujeta a este derecho; el contrato obliga a notificárnoslo salvo prohibición legal»  
 «y aquí está el registro de accesos de su personal de los últimos 12 meses: 4 accesos, todos con aprobación previa nuestra»

Y la valoración que el auditor anotó fue que la respuesta era verificable, que era lo que el pliego perseguía sin saber expresarlo.

Los tres incumplimientos de residencia que nadie sabía.

| al comprobar los nueve destinos |                      | clase 141 |
|---------------------------------|----------------------|-----------|
| telemetría                      | región distinta      | X         |
| registros de aplicación         | región distinta      | X         |
| proveedor de correo             | procesaba fuera      | X         |
| datos principales               | correcto             | ✓         |
| copias                          | correcto             | ✓         |
| réplicas                        | correcto             | ✓         |
| entorno de pruebas              | correcto             | ✓         |
| caché de borde                  | sin datos de cliente | ✓         |
| soporte del proveedor           | documentado          | ✓         |

Y el coste de corregir los tres: ciento ochenta euros al mes, frente a los más de veinte mil que habría costado la opción soberana completa.

El recuento.

|                                       |                 |               |
|---------------------------------------|-----------------|---------------|
| exigencias separadas                  | 0               | 4             |
| contra quién se protege, escrito      | no              | sí            |
| incumplimientos de residencia         | 3               | 0             |
| comprobación automática de residencia | no              | semanal       |
| claves gestionadas por el cliente     | no              | sí            |
| registro de acceso del proveedor      | no              | sí            |
| campos cifrados en cliente            | 0               | 5             |
| coste adicional mensual               | —               | 2.320 €       |
| coste de la opción soberana completa  | —               | ~21.000 €/mes |
| tiempo de responder a una auditoría   | días de trabajo | 6 min         |
| lo que no se garantiza, escrito       | no              | sí            |

La lección que esta clase traslada a la parte 14: el pliego pedía «infraestructura soberana» y lo que de verdad preocupaba era la jurisdicción, que no tiene respuesta técnica; separarlo en cuatro exigencias permitió responder a tres con dos mil trescientos euros al mes en lugar de veintiún mil. Y de todo el trabajo, lo que satisfizo al auditor no fue ninguna tecnología: fue poder ejecutar delante de él una consulta que demostraba dónde había estado cada dato cada día del último año, y decir con precisión qué no se garantizaba.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/177-soberania-digital-y-confidential-computing/lab.py
```

El laboratorio selecciona el motor de práctica security y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa decision-soberania en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

## Evidencia esperada

El artefacto principal es un control con amenaza, mitigación y evidencia. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye decision-soberania para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

### ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

### ■ Errores frecuentes

| Síntoma                                                           | Causa probable                                                          | Corrección                                                                                                                                     |
|-------------------------------------------------------------------|-------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Se compra una oferta soberana y el requisito seguía sin cumplirse | Se confundieron residencia y jurisdicción, que son exigencias distintas | Separa las cuatro con las cuatro preguntas y responde a cada una con lo que la resuelve.                                                       |
| Se adopta cómputo confidencial sin saber qué amenaza elimina      | No se escribió el modelo de amenazas                                    | Escribe contra quién se protege, comprueba si algo más barato ya lo cubre y estima el coste en rendimiento, servicios disponibles y operación. |
| Se usa cómputo confidencial y no se verifica la atestación        | Se supone que basta con que la memoria esté cifrada                     | La garantía es poder comprobar qué código se ejecuta: verifica la atestación con una política de medidas aceptadas y róta la al actualizar.    |
| Los datos cumplen residencia y los registros y la telemetría no   | Solo se comprobó el almacenamiento principal                            | Comprueba los nueve destinos y automatiza la comprobación con evidencia consultable.                                                           |
| Cifrar en el cliente rompe informes y búsquedas                   | Se aplicó a más campos de los necesarios                                | Aplicarlo a los campos que la exigencia menciona y deja el resto con claves gestionadas por el cliente.                                        |
| No se puede responder a un auditor sin días de trabajo            | La respuesta es una declaración y no una consulta                       | Construye la evidencia como subproducto de la operación y ejecútala delante; y escribe también lo que no se garantiza.                         |

### ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

### ■ Preguntas de comprobación

1. ¿Cuáles son las cuatro exigencias que se agrupan bajo la palabra soberanía y cuál no tiene respuesta técnica?
2. ¿Qué opciones de gestión de claves impiden que el proveedor descifre, y qué cuestan?
3. ¿Qué aporta la atestación y por qué sin ella el cómputo confidencial cambia poco?
4. ¿Contra qué amenazas no protege el cómputo confidencial?
5. ¿Qué se pide realmente en la mayoría de los requisitos de soberanía?

## Referencias

- Confidential Computing Consortium (2025). A technical analysis of confidential computing — qué protege y qué no. <<https://confidentialcomputing.io/resources/white-papers-reports/>>
- IETF (2023). RFC 9334: Remote attestation procedures architecture — verificación de qué código se ejecuta. <<https://www.rfc-editor.org/rfc/rfc9334.html>>
- Google Cloud (2025). Sovereign controls and external key management — residencia, control operativo y claves externas. <<https://cloud.google.com/sovereign-cloud>>
- Microsoft (2025). Sovereignty and access transparency — registro de acceso del personal del proveedor. <<https://learn.microsoft.com/azure/cloud-adoption-framework/scenarios/cloud-for-sovereignty/>>
- ENISA (2025). Cloud sovereignty: technical and legal dimensions — separación de exigencias técnicas y jurídicas. <<https://www.enisa.europa.eu/publications>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                       | Índice              | Siguiente                                 |
|------------------------------------------------|---------------------|-------------------------------------------|
| ← 176 · Edge, IoT y procesamiento desconectado | Parte 14 · Programa | 178 · Capstone: descubrimiento y diseño → |

## Evaluación — 177 Soberanía digital y confidential computing

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega decision-soberania con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- ☐ Resultado reproducible por una segunda persona.
- ☐ Evidencia vinculada a cada afirmación técnica.
- ☐ Prueba negativa o de fallo incluida.
- ☐ Costos y permisos expresados con unidades y alcance.
- ☐ Limitaciones declaradas sin presentar la simulación como producción.

## Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 178 — Capstone: descubrimiento y diseño

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

## Propósito

Empezar el proyecto que cierra el programa, y empezarlo por donde se empieza de verdad: descubriendo lo que hay y escribiendo las decisiones antes de construir nada. La clase da el encargo, el método de descubrimiento en el orden que evita rehacer trabajo, la lista de lo que hay que entregar como diseño y —lo más importante— los criterios con los que se va a juzgar, que se publican ahora y no al final, porque un diseño que no sabe cómo lo van a evaluar se defiende mal y se hace peor.

## Resultados de aprendizaje

Al finalizar podrás:

1. Recoger lo que existe con datos y no con documentación.
2. Escribir escenarios de calidad con medida antes de decidir nada.
3. Producir las decisiones de diseño con su traslado y sus premisas.
4. Reconocer cuáles de esas decisiones serán irreversibles.
5. Conocer los criterios de evaluación desde el principio.

## Conceptos centrales

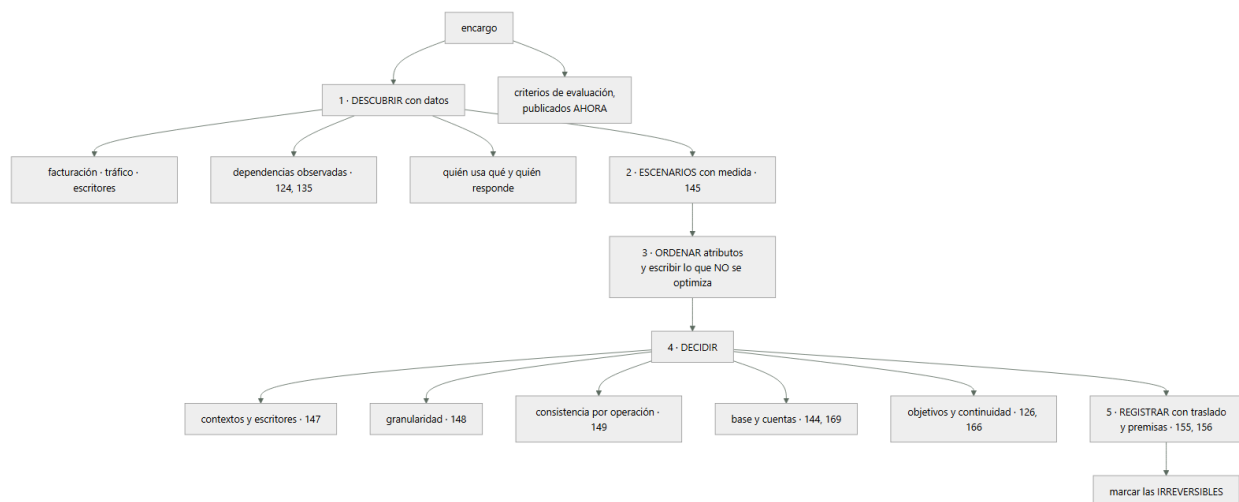
| Concepto               | Comprensión verificable                                                                                       |
|------------------------|---------------------------------------------------------------------------------------------------------------|
| descubrimiento         | Averiguar qué existe realmente, con facturación, tráfico y registros, no con inventarios ni diagramas.        |
| escenario de calidad   | Exigencia escrita con origen, estímulo, entorno, respuesta y medida. Es lo que permite decidir.               |
| registro de decisión   | Documento corto con opciones, traslado, prioridad, premisas y qué haría revisarla.                            |
| decisión irreversible  | La que después cuesta una migración: propiedad de los datos, particionado, identidad, jerarquía y colocación. |
| criterio de evaluación | Lo que se va a mirar al juzgar el trabajo. Publicado antes de empezar, cambia cómo se hace.                   |
| alcance declarado      | Lo que el proyecto sí hace y lo que explícitamente no hace, escrito antes de construir.                       |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. El encargo

El proyecto consiste en tomar un sistema real —el propio, uno de trabajo o el caso que este programa ha usado— y llevarlo desde donde está hasta un estado que se pueda defender. Y el encargo se enuncia así:

un sistema con usuarios reales o realistas  
 con una exigencia de negocio concreta que justifique el trabajo  
 con restricciones de presupuesto y de equipo declaradas  
 y con un plazo

Y lo que hay que producir en esta primera fase:

1. el retrato de lo que existe, con datos
2. los escenarios de calidad, con medida
3. el orden de atributos y lo que NO se optimiza
4. las decisiones de diseño, con su traslado
5. la lista de las que serán irreversibles
6. y el alcance declarado: qué hace y qué no hace este proyecto

Y una advertencia sobre el tamaño, porque es el error más común en un proyecto de cierre:

un alcance grande produce un trabajo superficial  
 un alcance pequeño y llevado hasta el final produce un trabajo  
 que se puede defender  
 → mejor un flujo de negocio completo, de extremo a extremo,  
 que quince a medias

Y el criterio para elegir el alcance:

que incluya al menos un dato con más de un consumidor ley 21  
 que tenga una operación que no se pueda perder  
 que cruce al menos una frontera de servicio  
 que tenga usuarios cuya experiencia se pueda medir  
 y que quepa en el plazo

Los criterios de evaluación, publicados ahora. Se juzgará:

1. ¿Está el descubrimiento hecho con datos, no con documentación?
2. ¿Hay escenarios con medida, y se usan para decidir?
3. ¿Cada decisión dice qué mejora y qué empeora, con cifras?
4. ¿Están identificadas las decisiones irreversibles?
5. ¿Está escrito lo que NO se hace, y por qué?
6. ¿Se han ejecutado las pruebas negativas, y qué encontraron?
7. ¿Las cifras del antes y el después son medidas o estimadas?
8. ¿Qué se descubrió que estaba mal y no se sabía?

Y el octavo es el que más peso tiene, por un motivo que este programa ha demostrado catorce veces: casi todo el valor de estos ejercicios está en lo que aparece al mirar, no en lo que se construye después.

### 2. Descubrir con datos

La primera regla es la que este programa ha comprobado tres veces:

Y el orden en que hay que recoger, porque cada paso depende del anterior:

- Y las cifras que conviene tener antes de diseñar nada:

Y una regla sobre el tiempo: el descubrimiento tarda más de lo que nadie planifica, y recortarlo produce las sorpresas de la clase 167. Conviene reservarle una parte real del plazo y observar las dependencias durante semanas, no días.

Con el retrato hecho, el orden de las decisiones importa porque unas condicionan a otras:

- Multi-Cloud Engineering Program · Parte 14 · Plataformas avanzadas, capstones y carrera

indicadores, objetivos por escenario y patrón de recuperación

- 7. COSTE clases 142, 172  
unidad de negocio y coste esperado por unidad

Y el formato de cada decisión, que es el de la clase 156:

- qué se decidió y qué opciones había
- qué mejora y qué empeora, con cifras clase 155
- qué prioridad lo decidió
- qué premisas se dan por ciertas
- y qué haría revisarla

Las decisiones irreversibles, que hay que marcar explícitamente porque después cuestan una migración —ley 14—:

- quién escribe cada dato ley 21
- la clave de partición y el número de particiones clases 110, 150
- la estructura de cuentas y el dominio de identidad clases 144, 169
- el esquema de etiquetado clase 142
- la colocación de los datos y su región clases 141, 161
- el tipo de la clave primaria clase 109
- y el nivel de aislamiento por cliente, si no hay camino clase 154

Y la disciplina que conviene aplicarles:

- cada irreversible se decide con más cuidado que las demás
- se escribe qué costaría cambiarla, en semanas
- y se comprueba si hay forma de dejar la puerta abierta
  - un camino de migración entre niveles, un número de particiones con margen, un esquema de etiquetas con hueco

Y el alcance declarado, que cierra la fase:

- lo que este proyecto SÍ hace
- lo que NO hace, y por qué
- lo que se deja para después, con fecha
- y lo que se acepta como riesgo, con nombre

La segunda lista es la que más discusiones evita durante la defensa.

## 4. Cómo se va a juzgar

Los ocho criterios del primer apartado, desarrollados, para que se puedan usar mientras se trabaja:

1. DESCUBRIMIENTO CON DATOS  
se espera ver cifras de facturación, tráfico y registros  
y la comparación entre lo documentado y lo observado  
→ un descubrimiento basado en el diagrama existente no cuenta
2. ESCENARIOS QUE DECIDEN  
no basta con escribirlos: hay que poder señalar qué decisión tomó cada uno  
→ si ninguna decisión se apoya en ellos, sobran
3. TRASLADO CON CIFRAS  
cada decisión dice qué empeora y cuánto clase 155  
→ «mejora todo» significa que no se ha medido
4. IRREVERSIBLES IDENTIFICADAS  
con su coste de cambio estimado
5. LO QUE NO SE HACE  
escrito, con motivo  
→ un proyecto sin límites declarados no se puede evaluar
6. PRUEBAS NEGATIVAS EJECUTADAS ley 22  
y lo que encontraron, incluidas las que fallaron  
→ una prueba que no se ejecutó no cuenta como resuelta
7. CIFRAS MEDIDAS, NO ESTIMADAS  
y las estimadas, marcadas como tales
8. LO QUE SE DESCUBRIÓ QUE ESTABA MAL  
→ el criterio con más peso

Y sobre el octavo, la evidencia de este programa:

casi todos los hallazgos importantes aparecieron al MIRAR

|                                          |           |
|------------------------------------------|-----------|
| 9 cuentas fuera de todo inventario       | clase 139 |
| 41 tablas con más de un escritor         | clase 147 |
| 85 % de permisos nunca usados            | clase 134 |
| un plan de 4 h que tardaba 11            | clase 166 |
| 10.300 alertas de técnicas imposibles    | clase 174 |
| y un compromiso de 9.800 €/mes sin dueño | clase 142 |

→ ninguno se construyó: todos estaban ahí

Y dos cosas que no se van a valorar, y conviene decirlo:

el número de tecnologías empleadas  
 → usar cinco cosas nuevas no es mejor que usar dos conocidas  
 y la sofisticación de la solución  
 → si un problema se resuelve con configuración, eso es la respuesta correcta clase 152

Y el consejo práctico para trabajar con estos criterios delante:

anotar los hallazgos según aparecen, con su fecha y su cifra  
 → porque al final se olvidan, y son lo que más vale

Y la lista de comprobación de esta fase:

- el alcance está declarado, con lo que no se hace
- el descubrimiento sale de facturación, tráfico y registros
- se ha comparado lo documentado con lo observado
- se sabe quién escribe cada dato
- hay entre 5 y 12 escenarios con medida
- hay tres atributos ordenados y una lista de lo que no se optimiza
- las decisiones siguen el orden: contextos, consistencia, granularidad
- cada decisión tiene traslado con cifras y premisas
- las irreversibles están marcadas con su coste de cambio
- están anotados los hallazgos del descubrimiento, con fecha
- los criterios de evaluación están a la vista mientras se trabaja

Y el cierre que enlaza con la clase siguiente: con el diseño escrito y las decisiones registradas, queda construirlo, operarlo y comprobarlo con las pruebas negativas de todo el programa. Es la materia de la clase 179.

## Ejemplo trabajado

Un equipo aplica esta fase a un sistema real: una plataforma de reservas con 40.000 usuarios diarios, once servicios y dos personas de guardia. Lo que sigue es el descubrimiento y el diseño, con lo que aparecieron por el camino.

El alcance declarado.

|    |                                                                                                                               |
|----|-------------------------------------------------------------------------------------------------------------------------------|
| SÍ | el flujo completo de reserva: buscar, reservar, pagar, confirmar y la operación de ese flujo: objetivos, alertas, continuidad |
| NO | el panel de administración interno                                                                                            |
| NO | la migración del sistema de facturación heredado                                                                              |
| NO | la aplicación móvil, que consume la misma API                                                                                 |

después, con fecha el panel de administración, en el trimestre siguiente  
 riesgo aceptado el sistema de facturación heredado sigue siendo un punto único; asumido por el responsable de área

El descubrimiento, y lo que no coincidía.

|                                 |    |    |
|---------------------------------|----|----|
| servicios                       | 11 | 11 |
| dependencias entre ellos        | 14 | 27 |
| servicios que llaman a terceros | 3  | 7  |
| tablas compartidas              | 0  | 19 |
| cuentas de nube                 | 2  | 5  |
| trabajos programados            | 6  | 14 |
| almacenes de objetos            | 4  | 9  |

Y los hallazgos anotados según aparecían:

|    |                                                                                                    |        |
|----|----------------------------------------------------------------------------------------------------|--------|
| H1 | 19 tablas con más de un escritor<br>la peor: «reservas», escrita por 5 servicios                   | ley 21 |
| H2 | 3 cuentas de nube que nadie recordaba<br>una con un almacén público con volcados de 2023           | ley 20 |
| H3 | 8 trabajos programados sin dueño identificable<br>2 de ellos llevaban 7 meses fallando en silencio | ley 13 |

- H4 el 100 % de las lecturas iban al principal «por seguridad»  
clase 149
- H5 4 servicios llamaban a un proveedor de pago con la misma  
clave estática, de hace 3 años clase 137
- H6 la disponibilidad observada era del 99,1 % y el techo por  
dependencias, del 99,05 % clase 126  
→ el sistema ya estaba en su límite teórico
- H7 de los 9 incidentes del último semestre, 2 se detectaron  
por alerta clase 120
- H8 no había ninguna copia probada; la última restauración  
era de hacía 14 meses ley 22

Ocho hallazgos, ninguno construido.

Los escenarios.

- E1 un usuario busca disponibilidad en hora punta, 900 peticiones/s  
→ resultados en menos de 400 ms, percentil 99
- E2 el proveedor de pago no responde durante 20 minutos  
→ se aceptan reservas y se cobran después; ninguna se pierde
- E3 dos usuarios reservan la última plaza con 100 ms de diferencia  
→ uno la consigue; nunca se reserva dos veces
- E4 se pierde la región principal  
→ el servicio vuelve en menos de 1 h, con menos de 5 min  
de datos perdidos
- E5 añadir un método de pago nuevo  
→ sin tocar el servicio de reservas, en menos de una semana
- E6 una dependencia se degrada  
→ alguien se entera en menos de 3 minutos
- E7 el volumen se multiplica por 3 en 12 meses  
→ sin rediseño y con coste por reserva que no suba
- E8 un cliente corporativo exige que sus datos no salgan del país  
→ incluidos registros y telemetría clase 141

Y el orden de atributos, tras el conflicto entre E2 y E3, que se resolvió como en la clase 145:

1. corrección de los datos no reservar dos veces, no cobrar dos veces
2. disponibilidad del flujo de reserva
3. evolución añadir integraciones sin tocar el núcleo

no se optimiza  
latencia por debajo de 200 ms fuera del país  
más de 5.000 peticiones/s  
despliegues sin ninguna parada en migraciones mayores  
ni multi-proveedor activo-activo

Las decisiones, en orden.

- D1 CONTEXTOS Y ESCRITORES clase 147  
5 contextos: búsqueda, reservas, pagos, clientes, notificación  
19 tablas compartidas → un escritor cada una  
traslado mejora: cambios de esquema de 4 semanas a días  
empeora: 3 consultas que cruzaban contextos pasan  
a copia por evento, con ~2 s de desfase  
irreversible Sí; coste de cambiarla: ~10 semanas
- D2 CONSISTENCIA POR OPERACIÓN clase 149  
17 operaciones clasificadas  
11 eventual, 4 con garantías de sesión, 2 fuertes  
traslado mejora: carga sobre el principal de 9.400/s a 1.100/s  
empeora: hay que implantar la garantía de sesión  
irreversible no
- D3 GRANULARIDAD clase 148

11 servicios → 5 unidades desplegadas  
 motivo por frontera, escrito; 2 equipos de 4 personas  
 traslado mejora: latencia p99 de 780 ms a ~220 ms estimado  
                   5 canalizaciones en vez de 11  
                   empeora: búsqueda y reservas escalan juntas  
 irreversible   parcialmente: unir es fácil, separar cuesta

D4 BASE Y CUENTAS clases 144, 169  
 3 cuentas nuevas por entorno; las 3 heredadas se cierran  
 12 controles preventivos; identidad federada  
 irreversible   SÍ (estructura e identidad)

D5 OBJETIVOS Y CONTINUIDAD clases 126, 166  
 2 indicadores, objetivo 99,5 %, y techo elevado a 99,74 %  
   haciendo opcionales 2 dependencias  
 patrón: mínimo encendido en segunda región  
 traslado mejora: E4 pasa a ser alcanzable  
                   empeora: +410 €/mes

D6 COSTE clase 142  
 unidad: coste por reserva  
 actual 0,061 €; objetivo tras el proyecto: por debajo de 0,045 €

Y las premisas anotadas, con quién las verifica:

|                                                       |                              |
|-------------------------------------------------------|------------------------------|
| «el proveedor de pago admite clave de idempotencia»   | verificado ✓                 |
| «la región secundaria tiene los servicios que usamos» | verificado ✓                 |
| «el volumen crecerá ×3 y no ×10»                      | producto, revisar en 6 meses |
| «el equipo seguirá siendo de 8 personas»              | dirección                    |

Y la tercera resultó falsa nueve meses después —el volumen creció ×1,4—, lo que hizo revisable la decisión de dimensionado y ahorró dinero: es el caso de la clase 156.

Lo que se marcó como irreversible.

|                                              |                    |
|----------------------------------------------|--------------------|
| decisión                                     | coste de cambiarla |
| quién escribe cada dato                      | ~10 semanas        |
| estructura de cuentas e identidad            | ~6 semanas         |
| esquema de etiquetado                        | ~2 semanas         |
| colocación de datos por región               | ~8 semanas         |
| clave de partición del histórico de reservas | ~5 semanas         |

Y en dos de ellas se dejó la puerta abierta a propósito:

|                       |                                    |           |
|-----------------------|------------------------------------|-----------|
| número de particiones | 24 en vez de 8, con margen para ×3 | clase 114 |
| nivel de aislamiento  | con camino de migración escrito    | clase 154 |

El estado al terminar la fase de diseño.

|                                      |                       |
|--------------------------------------|-----------------------|
| alcance declarado                    | sí                    |
| hallazgos anotados                   | 8                     |
| escenarios con medida                | 8                     |
| atributos ordenados                  | 3                     |
| lista de lo que no se optimiza       | 4 puntos              |
| decisiones registradas               | 6                     |
| con traslado y cifras                | 6 de 6                |
| con premisas y condición de revisión | 6 de 6                |
| irreversibles marcadas               | 5                     |
| con coste de cambio estimado         | 5 de 5                |
| cifras del descubrimiento            | medidas, no estimadas |

La lección que esta clase abre para el proyecto: la fase de diseño no produjo ni una línea de código y produjo ocho hallazgos que ya estaban ahí, entre ellos diecinueve tablas con varios escritores, tres cuentas olvidadas con datos de clientes expuestos y un sistema que ya operaba en su techo teórico de disponibilidad. Y las seis decisiones que se tomaron después se apoyan todas en un escenario concreto con su medida, lo que significa que dentro de dos años se podrá comprobar si seguían teniendo sentido.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/178-capstone-descubrimiento-y-diseno/lab.py
```

El laboratorio selecciona el motor de práctica capstone y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa dossier-diseno-final en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

## Evidencia esperada

El artefacto principal es un incremento integrado, demostrable y documentado. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye dossier-diseno-final para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

## ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■■ Errores frecuentes

| Síntoma                                                             | Causa probable                                                | Corrección                                                                                                     |
|---------------------------------------------------------------------|---------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------|
| El diseño se apoya en un inventario que no coincide con la realidad | El descubrimiento salió de documentación en vez de datos      | Usa facturación, tráfico observado, trazas y registro de auditoría; y compara lo documentado con lo observado. |
| El proyecto abarca demasiado y queda superficial                    | Alcance sin declarar y sin límites                            | Un flujo completo de extremo a extremo, con lo que no se hace escrito antes de empezar.                        |
| Las decisiones no se pueden discutir ni revisar                     | No dicen qué empeora ni de qué premisas dependen              | Registro con opciones, traslado con cifras, prioridad que lo decidió, premisas y qué haría revisarla.          |
| Una decisión tomada en media hora cuesta meses de deshacer          | No se marcó como irreversible                                 | Identifica las irreversibles, estima su coste de cambio y deja la puerta abierta donde se pueda.               |
| Se escriben escenarios y ninguna decisión se apoya en ellos         | Se hicieron como trámite                                      | Para cada decisión, señala qué escenario la motivó; si un escenario no decide nada, sobra.                     |
| El trabajo se juzga por la cantidad de tecnología empleada          | No se publicaron los criterios de evaluación antes de empezar | Publica los criterios al inicio, y valora sobre todo lo que se descubrió que estaba mal y no se sabía.         |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Por qué el descubrimiento debe salir de datos y no de documentación?
2. ¿En qué orden se toman las decisiones de diseño y por qué?
3. ¿Qué decisiones hay que marcar como irreversibles?
4. ¿Qué debe contener un registro de decisión para poder revisarse en dos años?
5. ¿Cuál de los ocho criterios de evaluación pesa más y por qué?

## Referencias

- Bass, L. y otros (2021). Software Architecture in Practice, caps. 3-4 — escenarios y decisiones dirigidas por atributos. <<https://www.oreilly.com/library/view/software-architecture-in/9780136886051/>>
- Nygard, M. (2011). Documenting architecture decisions — formato del registro de decisión. <<https://cognitect.com/blog/2011/11/15/documenting-architecture-decisions>>
- Ford, N. y otros (2021). Software Architecture: The Hard Parts — compromisos explícitos y análisis de alternativas. <<https://www.oreilly.com/library/view/software-architecture-the/9781492086888/>>
- Google Cloud (2025). Migration assessment and discovery — inventario a partir de datos reales. <<https://cloud.google.com/architecture/migration-to-gcp-getting-started>>
- Brandolini, A. (2025). Event storming — descubrir contextos a partir de los hechos del negocio. <<https://www.eventstorming.com/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                           | Índice              | Siguiente                                    |
|----------------------------------------------------|---------------------|----------------------------------------------|
| ← 177 · Soberanía digital y confidential computing | Parte 14 · Programa | 179 · Capstone: implementación y operación → |

## Evaluación — 178 Capstone: descubrimiento y diseño

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega dossier-diseno-final con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

## Criterio de aceptación

- ☐ Resultado reproducible por una segunda persona.
- ☐ Evidencia vinculada a cada afirmación técnica.
- ☐ Prueba negativa o de fallo incluida.
- ☐ Costos y permisos expresados con unidades y alcance.
- ☐ Limitaciones declaradas sin presentar la simulación como producción.

## Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 179 — Capstone: implementación y operación

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

## Propósito

Construir lo diseñado en la clase 178, ponerlo a funcionar y —lo que de verdad distingue un proyecto terminado de uno presentado— ejecutar las pruebas negativas de todo el programa y publicar lo que encuentren, incluidas las que fallen. La clase da el orden de construcción que evita rehacer trabajo, la lista completa de comprobaciones, el modo de medir el antes y el después sin engañarse, y la disciplina de anotar lo que se decidió no hacer y lo que salió mal.

## Resultados de aprendizaje

Al finalizar podrás:

1. Construir en el orden que evita rehacer: base, datos, entrega, operación.
2. Ejecutar las pruebas negativas del programa y registrar sus resultados.
3. Medir el antes y el después con cifras comparables.
4. Operar el sistema el tiempo suficiente para que los datos signifiquen algo.
5. Registrar lo que salió mal y lo que se dejó sin hacer.

## Conceptos centrales

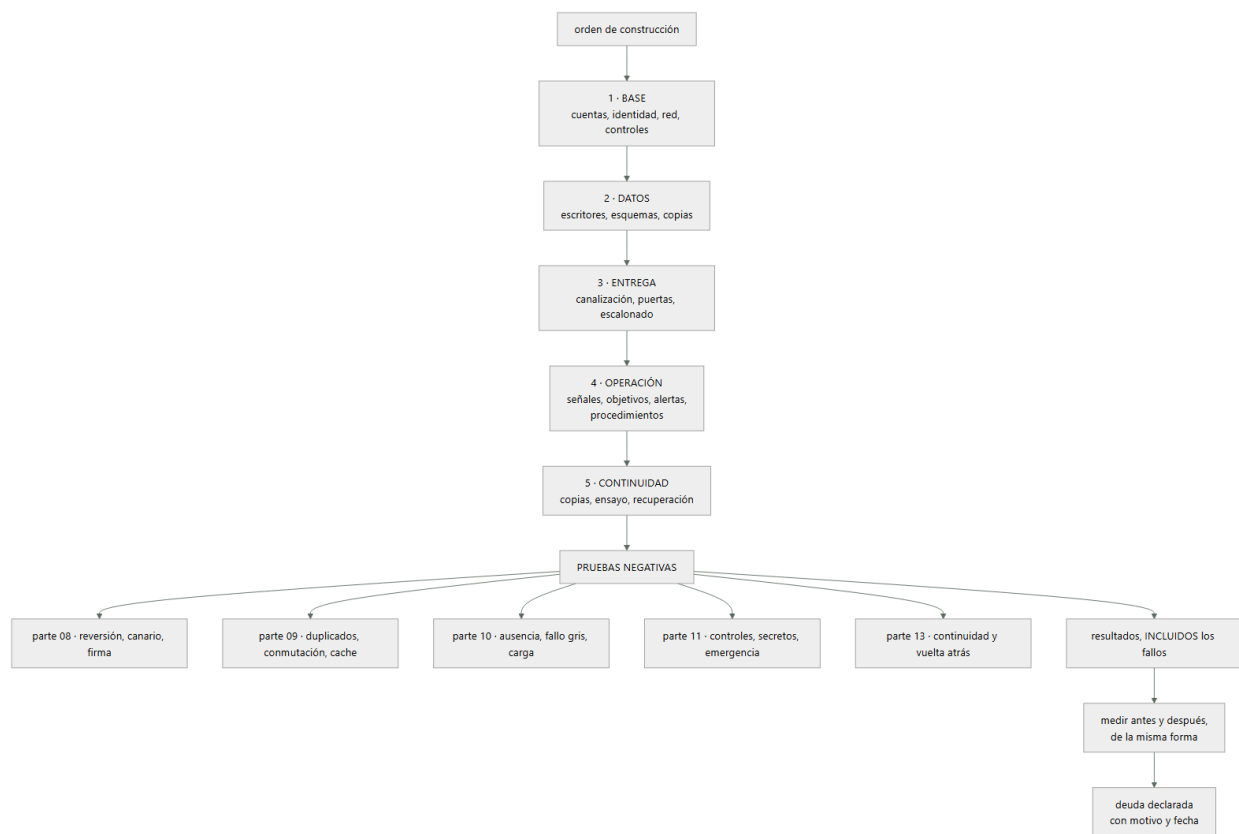
| Concepto                 | Comprensión verificable                                                                                  |
|--------------------------|----------------------------------------------------------------------------------------------------------|
| orden de construcción    | Secuencia en que se monta: lo que condiciona a lo demás primero, para no rehacerlo.                      |
| prueba negativa          | Comprobación que provoca el fallo a propósito y verifica que el sistema responde como se dijo.           |
| medida comparable        | Cifra tomada de la misma forma antes y después. Sin ella, la mejora no se puede afirmar.                 |
| periodo de operación     | Tiempo en que el sistema funciona de verdad antes de dar el proyecto por terminado. Sin él no hay datos. |
| registro de lo que falló | Anotación de las pruebas que no pasaron y de lo que se descubrió tarde. Es lo que hace creíble el resto. |
| deuda declarada          | Lo que se decidió no hacer, con motivo y fecha. Distinto de lo que se olvidó.                            |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## ■ Flujo de razonamiento



## Desarrollo

### 1. El orden de construcción

Construir en el orden equivocado obliga a rehacer, y el criterio es el mismo de la clase 169: primero lo que condiciona a lo demás.

1. BASE clases 144, 169  
 cuentas o proyectos, jerarquía, controles preventivos  
 identidad federada, sin usuarios locales clase 159  
 red con rangos del plan y denegación por defecto clases 135, 160  
 registro y auditoría enrutados clase 141  
 etiquetas obligatorias y presupuesto clase 142  
 → si esto se hace después, hay que migrar todo lo que se creó antes
2. DATOS clases 147, 149, 150  
 un escritor por dato, esquemas separados  
 consistencia decidida por operación  
 claves de partición con margen  
 copias inmutables en cuenta separada clase 166  
 → es lo más caro de cambiar; se decide con cuidado y se hace pronto
3. ENTREGA parte 08  
 canalización con puertas, artefacto inmutable y firmado  
 repositorio de entorno y bucle de reconciliación clase 103  
 despliegue escalonado con reversión clase 102
4. OPERACIÓN parte 10  
 las cuatro señales correlacionadas y la línea de cambios  
 indicadores y objetivos, medidos en el borde  
 alertas accionables con procedimiento  
 y proceso de incidentes
5. CONTINUIDAD clase 166  
 copias probadas, objetivos por escenario  
 y el ensayo de recuperación

Y dos reglas sobre el orden:

la base y los datos condicionan todo lo demás  
la operación se monta ANTES de que haga falta, no después  
del primer incidente

Y una advertencia sobre el paso 4, que es el que más se pospone:

montar la operación al final significa que durante toda la  
construcción no se sabe qué está pasando  
→ y los problemas de diseño aparecen sin señales que los expliquen  
→ conviene tener las señales y la línea de cambios desde el día 1

Y una nota sobre la marcha del trabajo:

cada paso termina cuando sus pruebas negativas pasan  
no cuando «está montado»  
→ y si alguna falla, ese paso no está terminado                      ley 22

## 2. Las pruebas negativas del programa

Esta es la lista completa, reunida de las partes anteriores. Ejecutarlas es el trabajo; el resultado es la entrega.

ENTREGA                                              parte 08

- desplegar una versión rota y comprobar que el canario la para
- revertir un despliegue y cronometrar
- desplegar un artefacto sin firmar y ver que la admisión lo rechaza
- enviar un secreto al repositorio y ver que se rechaza
- parar el bucle de reconciliación y esperar la alerta de antigüedad

DATOS                                              parte 09

- matar un consumidor a mitad de proceso
- entregar el mismo mensaje dos veces a propósito
- provocar una conmutación de la base y medir la pérdida
- vaciar el caché en hora punta
- parar el publicador de la tabla de salida
- enviar un mensaje con un campo renombrado

OPERACIÓN                                              parte 10

- parar un servicio y comprobar que salta la alerta de ausencia
- desplegar una versión con errores y medir el ritmo de consumo
- inyectar latencia en una dependencia opcional
- ralentizar una instancia y ver si el reparto la evita
- ejecutar un procedimiento con alguien que no lo escribió
- tirar el recolector de telemetría
- medir el codo con modelo abierto y datos realistas

SEGURIDAD                                              parte 11

- crear un recurso sin etiquetas de dueño
- crear en una región no permitida
- desactivar el registro de auditoría
- borrar una copia de seguridad
- hacer público un almacén
- alcanzar producción desde un entorno inferior
- sacar datos a un destino externo no declarado
- pedir un acceso temporal y cronometrarlo
- usar el acceso de emergencia
- simular lo que se quiere detectar                      clase 174

CONTINUIDAD                                              parte 13

- restaurar una copia y cronometrar
- simular un borrado y recuperar de copia inmutable
- conmutar a la segunda región y medir los cinco tramos
- volver atrás y comprobar que no hay dos escritores

Y la disciplina que las hace valer:

se ejecutan de verdad, no se razonan  
se anota el resultado y el tiempo  
las que fallan se corrigen y SE VUELVEN A EJECUTAR  
y las que fallaron se publican en el informe

Y la evidencia de este programa sobre qué esperar:

en la clase 144, 3 de 11 fallaron la primera vez  
en la clase 168, 5 de 11  
→ que fallen es lo normal; que no se ejecuten es el problema

Y una regla que evita el autoengaño más común:

una prueba que se ejecuta en preproducción y no en producción  
comprueba preproducción  
→ y hay que decir en cuál se ejecutó cada una

### 3. Medir sin engañarse

La comparación entre el antes y el después es donde estos proyectos se estropean, por tres motivos evitables:

1. LA CIFRA DE ANTES NO SE TOMÓ  
y se estima a posteriori, siempre a favor  
→ hay que tomarla ANTES de tocar nada clase 178
2. SE MIDE DE FORMA DISTINTA  
antes en el servidor y después en el borde clase 126  
antes con la media y después con el percentil  
→ misma forma, mismo punto, misma ventana
3. CAMBIÓ OTRA COSA  
el tráfico, la temporada, el catálogo  
→ por eso el coste se compara por unidad de negocio clase 142

Y las cifras que conviene tener en las dos columnas:

#### RENDIMIENTO

latencia por percentil, medida en el borde  
llamadas de red por operación clase 152  
distancia al codo clase 129

#### FIABILIDAD

disponibilidad observada, y techo por dependencias  
incidentes por trimestre y tiempo hasta mitigar  
proporción de problemas detectados por alerta clase 120

#### COSTE

coste por unidad de negocio clase 142  
y proporción atribuida

#### SEGURIDAD

alcance desde cada punto de entrada clase 133  
permisos concedidos y no usados clase 134  
credenciales de larga duración

#### CONTINUIDAD

plazo de recuperación MEDIDO y pérdida medida clase 166

#### OPERACIÓN

alertas por turno y proporción accionable clase 125  
trabajo repetitivo clase 128

Y una honestidad que hay que mantener en el informe:

lo medido se marca como medido  
lo estimado se marca como estimado, y se dice cómo se estimó  
y lo que no se pudo medir se dice que no se midió

El periodo de operación, que es lo que separa un proyecto de una demostración:

el sistema tiene que funcionar de verdad durante un tiempo  
→ con tráfico real o realista  
→ con despliegues reales  
→ y pasando al menos un ciclo completo de negocio clase 167

un mes es el mínimo razonable; tres son mejores  
→ porque los procesos mensuales, los picos y las sorpresas  
no ocurren en una semana

Y lo que se recoge durante ese periodo:

incidentes, con su revisión y sus acciones clase 127  
despliegues, y cuántos se revirtieron  
alertas disparadas y cuántas fueron accionables  
desviaciones de coste clase 142  
y lo que alguien tuvo que hacer a mano

Y la última es la más informativa: lo que se hizo a mano señala lo que falta automatizar.

#### 4. Lo que salió mal y lo que no se hizo

Un informe que solo cuenta lo que funcionó no se puede creer, y además pierde su parte más útil.

Lo que salió mal se registra con esta forma:

qué falló  
cuándo se descubrió, y cómo  
qué se hizo  
qué se cambió para que no vuelva  
y si se descubrió tarde, POR QUÉ no antes

Y la última pregunta es la que produce las mejoras de método:

|                                      |        |
|--------------------------------------|--------|
| «¿por qué no lo vimos antes?»        |        |
| → no había prueba negativa para eso  | ley 22 |
| → la señal existía y nadie la miraba | ley 15 |
| → algo lo compensaba automáticamente | ley 19 |
| → no tenía dueño                     | ley 20 |

La deuda declarada, que es distinta de lo que se olvidó:

qué se decidió NO hacer  
por qué  
qué riesgo se acepta mientras tanto  
quién lo acepta, con nombre  
y en qué fecha se revisa

Y la diferencia práctica:

|                                                                 |                                             |
|-----------------------------------------------------------------|---------------------------------------------|
| deuda declarada                                                 | alguien la conoce, la aceptó y hay fecha    |
| lo que se olvidó                                                | aparece en el peor momento y nadie responde |
| → el informe debe distinguirlas, y decir cuánta hay de cada una |                                             |

Y conviene incluir también lo que se probó y se descartó, que es información valiosa y suele perderse:

qué se intentó  
por qué no funcionó o no compensó  
y qué se hizo en su lugar  
→ evita que la siguiente persona lo intente otra vez

Y el cierre de esta fase, antes de la defensa:

|                                                         |           |
|---------------------------------------------------------|-----------|
| un documento con                                        |           |
| el antes y el después, con cifras y su origen           |           |
| los resultados de las pruebas negativas, con los fallos |           |
| los hallazgos del descubrimiento                        | clase 178 |
| las decisiones registradas, con sus premisas            |           |
| lo que salió mal y por qué no se vio antes              |           |
| la deuda declarada, con fechas y nombres                |           |
| y lo que se probó y se descartó                         |           |

Y la lista de comprobación de esta fase:

- se construyó en orden: base, datos, entrega, operación, continuidad
- la operación estaba montada antes de necesitarla
- cada paso terminó cuando sus pruebas negativas pasaron
- se ejecutaron TODAS las pruebas negativas de la lista
- está anotado en qué entorno se ejecutó cada una
- las que fallaron están publicadas, con lo que se corrigió
- las cifras de antes se tomaron antes de tocar nada
- antes y después se miden de la misma forma y en el mismo punto
- el coste se compara por unidad de negocio
- hubo un periodo de operación real de al menos un mes
- está registrado lo que hubo que hacer a mano
- lo que salió mal está escrito, con por qué no se vio antes
- la deuda declarada tiene motivo, riesgo, nombre y fecha
- está anotado lo que se probó y se descartó

Y el cierre que enlaza con la clase siguiente: con el sistema construido, operado y medido, queda defenderlo ante alguien que pregunte, convertirlo en algo que se pueda enseñar y decidir qué hacer después. Es la materia de la clase 180, que además cierra la parte 14.

## Ejemplo trabajado

El equipo de la clase 178 construye lo diseñado y lo opera tres meses. Lo que sigue es el resultado de las pruebas negativas —incluidas las nueve que fallaron— y la comparación entre el antes y el después.

La construcción, en orden.

### semanas 1-3 BASE

3 cuentas nuevas, jerarquía, 12 controles preventivos  
identidad federada; los 4 usuarios locales retirados  
rangos del plan central; denegación por defecto  
hallazgo al aplicar la denegación por defecto aparecieron  
11 conexiones que nadie había declarado

### semanas 3-7 DATOS

19 tablas con varios escritores → un escritor cada una  
7 consultas cruzadas sustituidas por copia por evento  
copias inmutables en cuenta separada  
hallazgo 2 de las 19 tablas resultaron no usarse en absoluto

### semanas 6-9 ENTREGA

canalización con puertas, artefacto firmado  
repositorio de entorno y bucle  
despliegue escalonado con reversión automática

### semanas 2-10 OPERACIÓN ← desde el principio, en paralelo

cuatro señales, línea de cambios, objetivos y alertas  
procedimientos escritos y probados por otra persona

### semanas 9-11 CONTINUIDAD

segunda región con lo mínimo encendido  
ensayo de conmutación

Y la decisión de montar la operación desde la semana 2 se justificó sola:

|                                                              |   |
|--------------------------------------------------------------|---|
| problemas de diseño detectados durante la construcción       | 6 |
| de ellos, gracias a las señales ya montadas                  | 5 |
| → una consulta por elemento en el nuevo servicio de búsqueda |   |
| → una conexión por petición en el de reservas                |   |
| → dos plazos ausentes en llamadas nuevas                     |   |
| → un trabajo programado que se ejecutaba dos veces           |   |

Las pruebas negativas: 31 ejecutadas, 9 fallaron.

### ENTREGA

|                                      |   |                                                   |
|--------------------------------------|---|---------------------------------------------------|
| versión rota → canario la para       | ✓ | parada en 9 min                                   |
| reversión cronometrada               | ✓ | 4 min                                             |
| artefacto sin firmar → rechazado     | ✓ |                                                   |
| secreto al repositorio → rechazado   | ✓ |                                                   |
| bucle parado → alerta por antigüedad | ✗ | la alerta existía y apuntaba a un canal sin nadie |

### DATOS

|                                     |   |                                                        |
|-------------------------------------|---|--------------------------------------------------------|
| matar consumidor a mitad            | ✓ | sin efecto duplicado                                   |
| entregar dos veces el mismo mensaje | ✗ | el consumidor de notificaciones envió dos correos      |
| conmutación de la base              | ✓ | 52 s, 0 pérdidas                                       |
| vaciar el caché en hora punta       | ✗ | caída de 4 min<br>→ faltaba limitación hacia el origen |
| parar el publicador                 | ✓ | alerta a los 4 min                                     |
| campo renombrado                    | ✓ | rechazado al producir                                  |

### OPERACIÓN

|                                        |   |                                          |
|----------------------------------------|---|------------------------------------------|
| parar un servicio → alerta de ausencia | ✓ | 70 s                                     |
| versión con errores → ritmo de consumo | ✓ | aviso a los 6 min                        |
| latencia en dependencia opcional       | ✗ | no era opcional:<br>el catálogo era duro |
| instancia lenta → reparto la evita     | ✗ | reparto rotatorio                        |
| procedimiento por otra persona         | ✗ | 9 de 14 con preguntas                    |
| tirar el recolector                    | ✓ | la aplicación siguió                     |
| medir el codo, modelo abierto          | ✓ | 1.850/s, no 6.000                        |

## SEGURIDAD

|                                   |   |                             |
|-----------------------------------|---|-----------------------------|
| recurso sin etiquetas             | ✓ | rechazado                   |
| región no permitida               | ✓ | rechazado                   |
| desactivar auditoría              | ✓ | rechazado                   |
| borrar una copia                  | ✓ | rechazado                   |
| almacén público                   | ✓ | rechazado                   |
| producción desde entorno inferior | ✓ | bloqueado                   |
| destino externo no declarado      | X | no había control de salida  |
| acceso temporal cronometrado      | ✓ | 84 s                        |
| acceso de emergencia              | X | la credencial no funcionaba |
| simular lo que se quiere detectar | X | 2 de 6 no se detectaban     |

## CONTINUIDAD

|                                   |   |                               |
|-----------------------------------|---|-------------------------------|
| restaurar copia y cronometrar     | ✓ | 41 min                        |
| simular borrado y recuperar       | ✓ | pérdida de 6 min              |
| conmutar y medir los cinco tramos | X | 2 h 10 frente a 1 h declarada |
| volver atrás sin dos escritores   | ✓ | tras corregir lo anterior     |

Nueve de treinta y una. Y lo que enseñó cada una:

|                                |                                               |
|--------------------------------|-----------------------------------------------|
| canal sin nadie                | el mismo hallazgo de las clases 131 y 159     |
| correos duplicados             | faltaba tabla de entrada en un consumidor     |
| caché sin limitación           | el caché era portante y no se sabía clase 111 |
| catálogo como dependencia dura | estaba declarada blanda y no lo era           |
| reparto rotatorio              | valor por defecto clase 152                   |
| procedimientos                 | 9 de 14 suponían contexto clase 128           |
| sin control de salida          | no se había montado clase 135                 |
| acceso de emergencia           | creado y nunca probado ley 22                 |
| detección incompleta           | 2 técnicas sin regla clase 174                |
| conmutación 2 h 10             | decidir y redirigir dominaban clase 166       |

Y tras corregirlas y volver a ejecutar:

|                            |          |   |
|----------------------------|----------|---|
| segunda ejecución completa | 31 de 31 | ✓ |
| conmutación                | 38 min   |   |

El periodo de operación: tres meses.

|                                                    |           |
|----------------------------------------------------|-----------|
| despliegues                                        | 214       |
| revertidos                                         | 7         |
| de ellos, por el canario, automáticamente          | 5         |
| incidentes                                         | 4         |
| detectados por alerta                              | 4         |
| tiempo medio hasta mitigar                         | 11 min    |
| alertas disparadas                                 | 41        |
| accionables                                        | 34 (83 %) |
| acciones manuales que hubo que hacer               | 19        |
| de ellas, repetidas más de 3 veces                 | 2         |
| → reprocesar mensajes fallidos y ampliar una cuota |           |
| → las dos se automatizaron                         |           |
| desviaciones de coste detectadas                   | 2         |
| → una real: un índice que faltaba                  | clase 142 |

Y el ciclo completo de negocio, que ocurrió en el mes 2:

|                                                                 |                   |
|-----------------------------------------------------------------|-------------------|
| cierre mensual                                                  | reveló 1 problema |
| → un proceso que asumía que podía leer del principal sin límite |                   |
| → corregido; no se habría visto en un periodo de una semana     |                   |

El antes y el después.

|                                        |           |         |           |
|----------------------------------------|-----------|---------|-----------|
| latencia p99 del flujo (borde)         | 780 ms    | 210 ms  | medida    |
| llamadas de red por reserva            | 9         | 3       | medida    |
| distancia al codo en el pico           | 88 %      | 41 %    | medida    |
| disponibilidad observada               | 99,1 %    | 99,7 %  | medida    |
| techo por dependencias                 | 99,05 %   | 99,74 % | calculado |
| incidentes por trimestre               | 9         | 4       | medida    |
| tiempo hasta mitigar                   | 1 h 40    | 11 min  | medida    |
| problemas detectados por alerta        | 2 de 9    | 4 de 4  | medida    |
| coste por reserva                      | 0,061 €   | 0,039 € | medida    |
| gasto atribuido                        | 38 %      | 96 %    | medida    |
| alcance desde un servicio comprometido | 10 de 11  | 2 de 5  | medida    |
| permisos concedidos y no usados        | 79 %      | 11 %    | medida    |
| credenciales de larga duración         | 6         | 0       | medida    |
| plazo de recuperación                  | no medido | 38 min  | MEDIDO    |

|                         |           |      |          |
|-------------------------|-----------|------|----------|
| pérdida en recuperación | no medido | 31 s | medida   |
| alertas por turno       | 6,1       | 0,8  | medida   |
| proporción accionable   | 14 %      | 83 % | medida   |
| trabajo repetitivo      | 29 %      | 12 % | estimado |

Y las dos filas marcadas como estimadas o calculadas lo están a propósito: el trabajo repetitivo se estimó con un registro de dos semanas, no con un año de datos, y el techo por dependencias es una multiplicación, no una medición.

Lo que salió mal, y por qué no se vio antes.

- la separación de escritores rompió 2 informes internos descubierto por un usuario, 9 días después  
por qué no antes nadie sabía que esos informes existían; no aparecían en el tráfico observado porque se ejecutaban una vez al mes ley 20  
cambio inventario de consumidores por consulta, no solo por servicio
- el coste subió un 21 % durante 3 semanas  
causa la segunda región encendida antes de retirar recursos antiguos  
por qué no antes la detección diaria avisó el día 2 y nadie la miró: iba a un canal nuevo sin nadie ley 15  
cambio la alerta de coste va al mismo canal que las demás
- una migración de esquema bloqueó la base 6 minutos  
por qué no antes se probó en un entorno con 1.000 filas  
cambio entorno de pruebas con volumen realista clase 104

La deuda declarada.

|                                                                 |                          |                     |                            |
|-----------------------------------------------------------------|--------------------------|---------------------|----------------------------|
| qué el sistema de facturación heredado sigue siendo punto único | motivo fuera del alcance | riesgo punto único  | revisa 6 meses (dirección) |
| la aplicación móvil usa la API v1                               | no se tocó               | contrato a mantener | 3 meses (producto)         |
| el aislamiento por cliente sigue siendo manual                  | solo 2 clientes lo piden | manual              | 12 meses (plataforma)      |
| el informe de cumplimiento se genera a mano                     | no había requisito aún   | ninguno             | cuando lo haya             |

Y lo que se probó y se descartó, que evita repetir el trabajo:

|                            |                                                        |           |
|----------------------------|--------------------------------------------------------|-----------|
| mallá de servicio          | 4 de 5 capacidades ya resueltas                        | clase 152 |
| capa de abstracción propia | cubría el 21 % del coste de salida                     | clase 158 |
| segundo proveedor activo   | ninguno de los motivos resistía                        | clase 157 |
| reescribir el buscador     | el problema era una consulta por elemento, no el motor | clase 124 |

La lección que esta clase abre para la defensa: se ejecutaron treinta y una pruebas negativas y fallaron nueve, entre ellas un acceso de emergencia que nunca se había usado y una dependencia declarada opcional que no lo era. Ninguna de las nueve se habría descubierto revisando la configuración, y las nueve estaban documentadas como resueltas. Y de los tres problemas que salieron mal durante la operación, dos tenían la misma causa que este programa lleva nombrando desde la parte 10: una señal que existía y que iba a un sitio donde nadie miraba.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/179-capstone-implementacion-y-operacion/lab.py
```

El laboratorio selecciona el motor de práctica capstone y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

- Lee exercise.steps y formula una predicción antes de ejecutar.
- Ejecuta la práctica y verifica todos los elementos de checks.

3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa evidencia-implementacion-final en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

## Evidencia esperada

El artefacto principal es un incremento integrado, demostrable y documentado. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye evidencia-implementacion-final para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

## ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.
- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■ Errores frecuentes

| Síntoma                                                           | Causa probable                                                    | Corrección                                                                                                                          |
|-------------------------------------------------------------------|-------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| Hay que rehacer trabajo porque la base llegó tarde                | Se construyó en el orden equivocado                               | Base, datos, entrega, operación y continuidad; lo que condiciona a lo demás, primero.                                               |
| Durante la construcción no se sabe qué está pasando               | La operación se dejó para el final                                | Monta señales, línea de cambios y objetivos desde las primeras semanas.                                                             |
| Se declara terminado un paso que no lo está                       | Se dio por bueno al montarlo, sin ejecutar sus pruebas negativas  | Un paso termina cuando sus pruebas negativas pasan, y las que fallan se corrigen y se repiten.                                      |
| La mejora no se puede afirmar                                     | La cifra de antes se estimó a posteriori o se midió de otra forma | Toma las cifras antes de tocar nada, mide igual en los dos momentos y compara el coste por unidad de negocio.                       |
| Aparecen sorpresas justo después de dar el proyecto por terminado | No hubo periodo de operación ni pasó un ciclo completo de negocio | Opera de verdad al menos un mes, y registra lo que hubo que hacer a mano.                                                           |
| El informe solo cuenta lo que funcionó y no se cree               | No se publicaron los fallos ni lo que se dejó sin hacer           | Registra las pruebas que fallaron, por qué no se vio antes, la deuda declarada con nombre y fecha, y lo que se probó y se descartó. |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Por qué la base y los datos se construyen antes que todo lo demás?
2. ¿Por qué la operación se monta antes de necesitarla?

3. ¿Qué hace que una prueba negativa cuente como ejecutada?
4. ¿Qué tres errores estropean la comparación entre el antes y el después?
5. ¿Qué diferencia hay entre deuda declarada y lo que se olvidó?

## Referencias

- Beyer, B. y otros (2018). The Site Reliability Workbook — puesta en producción y verificación con pruebas reales.  
<<https://sre.google/workbook/table-of-contents/>>
- Basiri, A. y otros (2016). Chaos engineering — provocar el fallo como método de verificación.  
<<https://ieeexplore.ieee.org/document/7503833>>
- Humble, J. y Farley, D. (2010). Continuous Delivery — orden de construcción de la canalización y verificación.  
<<https://www.oreilly.com/library/view/continuous-delivery-reliable/9780321670250/>>
- AWS (2025). Well-Architected review process — evaluación con evidencia y registro de deuda.  
<<https://docs.aws.amazon.com/wellarchitected/latest/framework/welcome.html>>
- Forsgren, N. y otros (2018). Accelerate — medir antes y después con las mismas definiciones.  
<<https://itrevolution.com/product/accelerate/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                  | Índice              | Siguiente                                                |
|-------------------------------------------|---------------------|----------------------------------------------------------|
| ← 178 · Capstone: descubrimiento y diseño | Parte 14 · Programa | 180 · Capstone: defensa, portafolio y plan profesional → |

## Evaluación — 179 Capstone: implementación y operación

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega evidencia-implementacion-final con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- [ ] Resultado reproducible por una segunda persona.
- [ ] Evidencia vinculada a cada afirmación técnica.
- [ ] Prueba negativa o de fallo incluida.
- [ ] Costos y permisos expresados con unidades y alcance.
- [ ] Limitaciones declaradas sin presentar la simulación como producción.

## Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |

# 180 — Capstone: defensa, portafolio y plan profesional

Parte: 14 — Plataformas avanzadas, capstones y carrera Nivel: experto-frontera · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

## Propósito

Cerrar el proyecto final defendiéndolo ante alguien que pregunte de verdad, convertirlo en un portafolio que demuestre criterio y no solo herramientas, y decidir qué hacer después con honestidad sobre lo que aún no se sabe. La clase cierra además la parte 14: corrige las cinco predicciones de la clase 168 —dos acertadas, una acertada de más, dos a medias—, actualiza el recuento de leyes, añade la ley 23 y escribe la hipótesis de la parte 15.

## Resultados de aprendizaje

Al finalizar podrás:

1. Defender el proyecto ante preguntas duras, incluidas las que no tienen buena respuesta.
2. Construir un portafolio que demuestre criterio, no catálogo de herramientas.
3. Decidir el siguiente paso profesional con datos y no con etiquetas.
4. Corregir las cinco predicciones de la clase 168 con la evidencia de la parte 14.
5. Escribir la hipótesis de la parte 15 en forma refutable.

## Conceptos centrales

| Concepto                     | Comprensión verificable                                                                                            |
|------------------------------|--------------------------------------------------------------------------------------------------------------------|
| defensa                      | Exposición del proyecto ante alguien que pregunta con intención de encontrar el punto débil.                       |
| pregunta sin buena respuesta | Aquella cuya respuesta honesta es «no lo medimos» o «se aceptó el riesgo». Contestarla bien vale más que evitarla. |
| portafolio                   | Conjunto de evidencias que demuestran cómo se decide, no qué herramientas se han tocado.                           |
| criterio demostrable         | Decisión registrada con premisas, alternativas descartadas y su coste de cambio.                                   |
| hipótesis de parte           | Afirmación refutable escrita antes de estudiar, que la parte siguiente corrige con evidencia.                      |
| ley 23                       | La capacidad de un equipo la limita lo que ya mantiene, no lo que sabe construir.                                  |

## Modelo mental

El nivel experto no consiste en conocer más productos, sino en formular mejores preguntas, validar supuestos y sostener decisiones frente a costo, riesgo y operación.

Aplicado a esta clase, separa siempre cuatro planos: intención (qué necesita el usuario), configuración (qué declaramos), estado observado (qué existe de verdad) y evidencia (cómo sabemos que cumple). Confundirlos produce diseños que se ven correctos en un diagrama pero fallan al operar.

## Flujo de razonamiento



## Desarrollo

### 1. La defensa

Una defensa útil no es una presentación: es alguien buscando el punto débil. Y las preguntas que de verdad distinguen son estas:

1. ¿QUÉ DESCUBRISTE QUE ESTABA MAL Y NADIE SABÍA?  
→ la respuesta débil habla de lo que se construyó  
→ la fuerte habla de lo que se encontró clase 178
2. ¿CUÁL FUE LA DECISIÓN MÁS DIFÍCIL Y QUÉ DESCARTASTE?  
→ si no hay alternativas descartadas, no hubo decisión  
→ y hay que saber decir el coste de cambiarla
3. ¿QUÉ PRUEBA NEGATIVA FALLÓ?  
→ «ninguna» significa que no se ejecutaron ley 22
4. ¿QUÉ NO MEDISTE?  
→ todo proyecto tiene huecos; saber cuáles es la señal
5. ¿QUÉ HARÍAS DISTINTO?  
→ y en qué momento se supo que había que hacerlo así

Y las preguntas que un revisor experimentado añade, que son incómodas a propósito:

¿qué parte de la mejora se debe al cambio y qué parte a otra cosa?  
¿qué se rompió y quién lo notó primero, tú o un usuario?  
¿esta cifra está medida o estimada?  
¿qué pasa si mañana se va la persona que montó esto?  
¿cuánto cuesta al mes y por qué esa cifra y no la mitad?  
¿qué se sigue haciendo a mano?

Y cómo contestar las que no tienen buena respuesta, que es donde se gana o se pierde la defensa:

decir «no lo medimos» y decir por qué  
decir «se aceptó el riesgo, lo aceptó X, se revisa en Y»  
decir «no lo sé» y decir cómo se averiguaría  
→ inventar una respuesta se detecta en la siguiente pregunta

Y el error más frecuente en estas defensas:

defender la decisión en lugar de defender el razonamiento  
→ una decisión puede ser incorrecta con un razonamiento correcto  
si las premisas cambiaron; eso se explica  
→ una decisión correcta sin razonamiento fue suerte, y se nota

### 2. El portafolio

Un portafolio de herramientas dice qué se ha tocado; uno de criterio dice cómo se decide. El segundo es el que resiste una entrevista técnica.

LO QUE NO DEMUESTRA CASI NADA  
una lista de tecnologías  
un diagrama bonito sin decisiones  
un repositorio que despliega algo  
un certificado ley 17

LO QUE SÍ DEMUESTRA  
una decisión registrada con premisas y alternativas  
una prueba negativa que falló, y qué se cambió  
una cifra de antes y una de después, con su origen  
un problema que se diagnosticó mal, y cómo se corrigió  
algo que se retiró clase 171

Y la pieza más difícil de encontrar en cualquier portafolio, que por eso vale tanto:

«esto lo diagnosticué mal»  
qué se creyó al principio  
qué evidencia lo desmintió  
y qué se cambió en el método, no solo en el sistema  
→ el programa entero está construido sobre esto: cada parte  
corrige por escrito las predicciones de la anterior

Y la forma práctica del portafolio:

un documento corto por proyecto, no un repositorio enorme  
 problema, con la cifra que lo demostraba  
 qué se descubrió que no se sabía  
 tres o cuatro decisiones, con alternativas y coste de cambio  
 pruebas negativas, con las que fallaron  
 antes y después, con origen de cada cifra  
 deuda declarada  
 y qué se haría distinto

Y una nota sobre lo que se puede publicar:

|                              |                              |
|------------------------------|------------------------------|
| cifras absolutas del negocio | normalmente no               |
| proporciones y relaciones    | casi siempre sí              |
| nombres de clientes          | no                           |
| el razonamiento              | siempre, y es lo que importa |

### 3. El siguiente paso, sin etiquetas

Al terminar un programa como este la pregunta habitual es «¿qué certificación saco ahora?», y casi siempre es la pregunta equivocada.

lo que una certificación demuestra  
 que se conoce el catálogo de un proveedor  
 y que se sabe contestar a preguntas de examen

lo que NO demuestra  
 criterio, que es lo que se paga  
 ni haber operado nada bajo presión

cuándo compensa de verdad  
 cuando un contrato o un socio la exigen  
 cuando se cambia de área y hace falta cubrir un hueco  
 cuando la empresa la paga y no cuesta tiempo propio

Y una forma más útil de decidir el siguiente paso, con los datos del propio proyecto:

1. mira las pruebas negativas que fallaron  
 → señalan lo que no se domina, con evidencia
2. mira lo que se hizo a mano y se repitió  
 → señala una capacidad que falta clase 128
3. mira lo que se dejó como deuda por no saber hacerlo  
 → distinto de lo que se dejó por prioridad
4. mira qué preguntas de la defensa no supiste contestar

Y sobre los caminos posibles, con lo que cada uno exige de verdad:

|                                                  |                                                                                                                                                                   |
|--------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| INGENIERÍA DE PLATAFORMA<br>exige<br>fracasa por | producto interno, no infraestructura<br>saber decir que no y medir adopción<br>construir capacidades sin usuarios                                                 |
| FIABILIDAD<br>exige<br>fracasa por               | objetivos, presupuesto de error, incidentes<br>disciplina de medida y de revisión<br>convertirse en soporte de otros <span style="float: right;">clase 173</span> |
| ARQUITECTURA<br>exige<br>fracasa por             | decisiones y sus consecuencias<br>decir el coste de cambio antes de decidir<br>diseñar sin operar nunca                                                           |
| DATOS<br>exige<br>fracasa por                    | escritores, contratos, linaje<br>entender consistencia de verdad<br>tratar el dato como fichero                                                                   |
| SEGURIDAD<br>exige<br>fracasa por                | alcance, identidad, detección<br>medir alcance, no contar controles<br>policía en vez de camino fácil <span style="float: right;">ley 16</span>                   |

Y el consejo que este programa sostiene por encima de las etiquetas:

lo que hace a alguien difícil de sustituir no es la herramienta  
 es haber operado algo, haberlo roto a propósito, haber medido  
 el resultado y haber escrito lo que salió mal  
 → y eso se puede hacer en cualquiera de los cinco caminos

## 4. Cierre de la parte 14: corrección, recuento e hipótesis

Las cinco predicciones de la clase 168, corregidas con la evidencia de las clases 169 a 179.

1. «la escala organizativa no traerá problemas técnicos nuevos: los mismos con más gente; la ley 16 dominará»

PRIMERA MITAD CORRECTA. Lo genuinamente nuevo de la parte 14 fue organizativo –modelo operativo, hoja de ruta, órganos de decisión, retirada– y no técnico.

SEGUNDA MITAD A MEDIAS. La ley 16 apareció con fuerza (35 cuentas creadas por fuera porque el proceso tardaba 6 días; 41 excepciones a un control; la policía del coste), pero NO dominó: empató con la 15 (110 políticas obligatorias que nadie sabía citar; 12.400 alertas) y con la 20 (cuentas, trabajos y servicios sin dueño). Tres leyes a la par, no una dominante.

2. «en cargas de IA el problema dominante no será el modelo ni las GPU: será el dato; leyes 14 y 21, sin leyes nuevas»

CORRECTA, y en los tres términos. El 41 % del tiempo de GPU esperando datos, dos conjuntos usados sin permiso, y el desvío entre entrenamiento y servicio –ley 21 en su forma más cara– valía 4,8 puntos de precisión. Las definiciones de atributo resultaron ser decisiones de creación, ley 14. Ninguna ley nueva hizo falta.

3. «en soberanía, la mayor parte de lo que se vende resuelve una amenaza que casi nadie tiene; lo que resuelve un problema real es saber dónde está cada dato y poder demostrarlo»

CORRECTA. La oferta soberana costaba 21.000 €/mes y se descartó; lo que satisfizo al auditor fue una consulta que respondía dónde estaba cada dato, y costó 2.320 €/mes.

4. «el proyecto final encontrará que lo que falta no es tecnología: la mayoría de los hallazgos serán leyes 20 y 22»

CORRECTA, y con más margen del previsto. En el descubrimiento: 8 trabajos programados sin dueño, 2 fallando en silencio 7 meses, 3 cuentas olvidadas, copias sin probar en 14 meses. Y en la ejecución: 9 de 31 pruebas negativas fallaron, y las 9 estaban documentadas como resueltas. Todas son ley 22.

5. «las leyes que más aparecen no serán las más útiles: la 13 y la 15 dominarán el recuento, pero las que más decisiones cambiaron serán la 14 y la 21»

A MEDIAS, y el fallo es instructivo. La primera mitad se cumple: 13 y 15 dominan el recuento. La segunda es incompleta: la 14 y la 21 cambiaron las decisiones de DISEÑO –claves de partición, jerarquía de cuentas, dominio de identidad, división de datos–, pero la ley que más problemas REALES destapó en todo el programa fue la 22: 5 de 11 pruebas fallidas en la parte 13 y 9 de 31 en el proyecto final. Pruejamos bien el mecanismo (frecuencia ≠ utilidad) y mal cuál gana: no es la que cambia el diseño, es la que descubre que lo escrito no era cierto.

Marcador: dos correctas, una correcta y subestimada, dos a medias. Peor que el de la clase 156, y por un motivo identificable: la parte 14 era la primera cuyo contenido dependía de personas y no de sistemas, y ahí las predicciones acertaron el tipo de problema y fallaron el reparto.

El recuento de leyes, cerrada la parte 14.

|        |                                                  |    |
|--------|--------------------------------------------------|----|
| ley 13 | lo que no se mira deja de funcionar en silencio  | 30 |
| ley 15 | la señal existe y nadie la mira                  | 24 |
| ley 14 | el coste se decide al crear, no al pagar         | 19 |
| ley 16 | un control que estorba se rodea                  | 17 |
| ley 22 | un procedimiento nunca ejecutado no funciona     | 15 |
| ley 20 | lo que no tiene dueño se filtra y se desperdicia | 13 |
| ley 21 | el acoplamiento vive en quién escribe            | 12 |
| ley 19 | la compensación hace invisible el fallo          | 9  |
| ley 18 | lo asíncrono traslada la garantía, no la elimina | 8  |

Y la parte 14 obliga a escribir una ley nueva, la primera que no habla de sistemas:

LEY 23

la capacidad de un equipo la limita lo que ya mantiene,  
no lo que sabe construir

apariciones en esta parte

4

clase 169 214 cuentas, ninguna retirada  
clase 170 140 políticas que había que mantener  
clase 171 54 % de la capacidad en mantener 31 piezas  
clase 173 60 % del tiempo operando sistemas de otros

consecuencia sin retirada, la capacidad de construir  
tiende a cero, y ninguna contratación lo arregla

La hipótesis de la parte 15 (clases 181 a 192, arquitectura de sistemas), escrita antes de estudiarla y para que la clase 192 la corrija:

1. la parte 15 va a formalizar con vocabulario lo que este programa ya obtuvo por evidencia; el valor no estará en los nombres sino en poder discutir una decisión antes de tomarla
2. de los atributos de calidad, el que más decisiones cambiará no será rendimiento ni disponibilidad: será modificabilidad, porque es el único que se paga todos los meses
3. la frontera correcta entre módulos coincidirá con quién escribe cada dato en más del 70 % de los casos: la ley 21 predice las fronteras mejor que cualquier criterio funcional
4. el análisis de puntos de fallo encontrará que la mayoría de los puntos únicos no son de infraestructura sino de conocimiento y de procedimiento: leyes 22 y 23
5. las decisiones registradas y las funciones de aptitud fracasarán por la misma razón por la que fracasan los controles: si estorban más de lo que ayudan, se rodean; y el registro se rellenará después de decidir, para cumplir

ley 16

Y el cierre de la parte 14, que es también el de la primera mitad del programa: de ciento ochenta clases, lo que más problemas reales destapó no fue ningún diseño ni ninguna herramienta, sino ejecutar a propósito lo que estaba escrito como resuelto. La parte 15 empieza por el otro extremo —los requisitos y los atributos de calidad, antes de construir nada— y su primera clase, la 181, distingue lo que el sistema debe hacer de lo que lo condiciona y de lo bien que debe hacerlo.

## Ejemplo trabajado

Defensa del proyecto de las clases 178 y 179 ante un comité de tres personas: una de arquitectura, una de operación y una de finanzas. Transcripción de las nueve preguntas y las respuestas que se dieron, incluidas las tres que no tenían buena respuesta.

P1 (arquitectura)

¿Qué descubriste que estaba mal y nadie sabía?

R Ocho cosas. La que más me sorprendió: ocho trabajos programados sin dueño, y dos llevaban siete meses fallando en silencio sin que nadie lo notara. Uno de ellos calculaba las comisiones de los socios. Nadie se quejó porque el sistema seguía usando el último valor correcto.

← respuesta fuerte: habla de un hallazgo, no de una construcción

P2 (arquitectura)

¿Cuál fue la decisión más difícil y qué descartaste?

R Separar los escritores de las diecinueve tablas compartidas. La alternativa era mantener el acceso compartido y añadir validación; la descarté porque no resuelve el acoplamiento, solo lo hace más lento de descubrir. Coste de cambio si me equivoqué: alto, unos tres meses, porque siete consultas

cruzadas se convirtieron en copias por evento y volver atrás exige rehacerlas.

← alternativa descartada y coste de cambio dicho sin que lo pregunten

P3 (operación)

¿Qué prueba negativa falló?

R Nueve de treinta y una. La peor: el acceso de emergencia no funcionaba. Estaba creado, documentado y nunca usado. Si hubiéramos tenido un incidente de identidad, no habríamos podido entrar. También falló la conmutación: dos horas diez frente a la hora declarada, y los dos tramos que dominaban eran decidir y redirigir, no arrancar nada.

← «ninguna» habría terminado la defensa aquí

P4 (finanzas)

Dices que el coste por reserva bajó de 0,061 a 0,039.

¿Cuánto de eso es tu cambio y cuánto es que el tráfico subió?

R El tráfico subió un 14 % en el trimestre, y por eso comparo por reserva y no en total. Del descenso, la parte medible es el índice que faltaba y las siete consultas cruzadas retiradas. Lo que NO puedo separar limpiamente es el efecto de la reserva de capacidad que compró finanzas en el mes 2: eso son unos 0,004 € que no son míos.

← reconoce la parte de la mejora que no le corresponde

P5 (operación)

¿Qué se rompió, y lo notaste tú o un usuario?

R Un usuario. Dos informes internos dejaron de funcionar al separar los escritores, y lo supimos nueve días después porque se ejecutaban una vez al mes. No aparecían en el tráfico observado. Cambiamos el método: ahora el inventario de consumidores se hace por consulta, no por servicio.

← incluye el cambio de método, no solo el arreglo

P6 (finanzas)

¿Esta cifra de trabajo repetitivo está medida o estimada?

R Estimada. La medimos con un registro de dos semanas y la extrapolamos. La de antes es peor: es una encuesta al equipo. Las dos están marcadas como estimadas en el informe y no definiendo la diferencia exacta, solo la dirección.

← pregunta sin buena respuesta, contestada bien

P7 (arquitectura)

¿Qué pasa si mañana te vas?

R Catorce procedimientos escritos, de los cuales nueve fallaron la prueba de que los ejecutara otra persona y se corrigieron. Lo que sigue dependiendo de mí es la relación entre el inventario de consumidores y los informes de negocio: eso está en mi cabeza y no está escrito. Es la deuda que más me preocupa y no está en la lista formal.

← el candidato añade una deuda que no había declarado

P8 (operación)

¿Qué se sigue haciendo a mano?

R Diecinueve acciones en tres meses. Dos se repitieron más de tres veces y se automatizaron. Las otras diecisiete fueron únicas. Y el informe de cumplimiento se sigue generando a mano, declarado como deuda, porque no hay requisito todavía.

P9 (comité)

¿Qué harías distinto?

R Tres cosas. Ejecutaría las pruebas negativas de continuidad

la primera semana y no la novena: la conmutación tardaba el doble de lo declarado y eso condicionaba el diseño. Probaría las migraciones de esquema contra volumen realista desde el principio. Y mandaría la alerta de coste al mismo canal que las demás: existía desde el día uno, avisó el día dos y nadie la miró durante tres semanas.

La evaluación del comité, con los ocho criterios publicados en la clase 178:

|   |                                               |   |       |
|---|-----------------------------------------------|---|-------|
| 1 | problema real, demostrado con una cifra       | 1 | alto  |
| 2 | descubrimiento antes que diseño               | 2 | alto  |
| 3 | decisiones registradas con alternativas       | 2 | alto  |
| 4 | pruebas negativas ejecutadas y publicadas     | 3 | alto  |
| 5 | antes y después medidos igual                 | 2 | medio |
| 6 | deuda declarada con nombre y fecha            | 1 | alto  |
| 7 | operación real y sostenida                    | 2 | alto  |
| 8 | qué se descubrió que estaba mal y no se sabía | 4 | alto  |

Y el comentario que el comité escribió, que resume lo que esta parte del programa quería enseñar:

«lo que sostiene esta defensa no es el diseño, que es correcto y poco original. Es que nueve pruebas fallaron, están publicadas, y una de ellas —el acceso de emergencia— habría convertido un incidente normal en uno sin salida. Y que a la pregunta sobre el trabajo repetitivo contestó <estimada> en lugar de defender la cifra.»

Y el paso siguiente que el candidato eligió, con el método de esta clase y no con una certificación:

|                                 |                            |
|---------------------------------|----------------------------|
| pruebas negativas que fallaron  | → identidad y continuidad  |
| trabajo manual repetido         | → reprocesado de mensajes  |
| deuda por no saber hacerlo      | → aislamiento por cliente  |
| preguntas que no supo contestar | → separar efectos en coste |

|          |                                                                                                           |
|----------|-----------------------------------------------------------------------------------------------------------|
| decisión | seis meses en fiabilidad, no en arquitectura                                                              |
| motivo   | tres de los cuatro huecos son de operación medida,<br>y el cuarto se aprende antes operando que diseñando |

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-14-advanced-platform-capstones-career/180-capstone-defensa-portafolio-y-plan-profesional/lab.py
```

El laboratorio selecciona el motor de práctica capstone y produce labresult.json. El escenario, sus comprobaciones y el artefacto esperado corresponden a esta clase; no requiere credenciales y deja explícito qué debe revalidarse en un sandbox real.

1. Lee exercise.steps y formula una predicción antes de ejecutar.
2. Ejecuta la práctica y verifica todos los elementos de checks.
3. Provoca el caso de negativetest y explica la señal observada.
4. Materializa defensa-final en evidence/ usando la plantilla indicada.
5. Para proveedor real, sigue sandbox.requires, registra costo y ejecuta destroy.

### Evidencia esperada

El artefacto principal es un incremento integrado, demostrable y documentado. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

## Reto verificable

Construye defensa-final para el caso CloudShop. Incluye una alternativa descartada, un supuesto que pueda falsarse, una prueba de fallo y una decisión de rollback.

### ■ Criterio de aceptación

- [ ] lab.py termina con código 0 y genera JSON válido.

- [ ] La entrega conecta al menos tres requisitos con mecanismos verificables.
- [ ] Existe una prueba positiva y una prueba negativa con evidencia.
- [ ] Seguridad, costo y operación aparecen como decisiones, no como anexos.
- [ ] Se declara una limitación y una condición que obligaría a revisar el diseño.
- [ ] Otra persona puede repetir el recorrido sin conocimiento tácito.

## ■ Errores frecuentes

| Síntoma                                                                   | Causa probable                                                              | Corrección                                                                                                                 |
|---------------------------------------------------------------------------|-----------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| La defensa se cae en la segunda pregunta                                  | Se defendió la decisión en vez del razonamiento, o se inventó una respuesta | Explica premisas y alternativas; di «no lo medimos» o «no lo sé, se averiguaría así» cuando sea el caso.                   |
| Un revisor pregunta qué prueba negativa falló y la respuesta es «ninguna» | No se ejecutaron de verdad                                                  | Ejecútalas y publica las que fallen; un proyecto sin fallos publicados no se cree.                                         |
| El portafolio impresiona poco pese a incluir muchas tecnologías           | Demuestra catálogo, no criterio                                             | Sustituye la lista por decisiones con alternativas descartadas, pruebas fallidas, cifras con origen y algo que se retiró.  |
| Se atribuye toda la mejora al propio trabajo                              | No se separó lo que cambió por otras causas                                 | Compara por unidad de negocio y di explícitamente qué parte de la mejora no te corresponde.                                |
| El siguiente paso profesional se decide por la certificación de moda      | Se confunde la medida con el objetivo                                       | Elige con los datos del proyecto: pruebas fallidas, trabajo manual repetido, deuda por no saber y preguntas sin contestar. |
| El equipo no avanza aunque se contrate más gente                          | La capacidad está consumida por lo que ya se mantiene                       | Retira antes de añadir; sin retirada, la capacidad de construir tiende a cero.                                             |

## ■ Seguridad, ética y costo

Trabaja con cuentas propias o sandboxes autorizados. No publiques secretos, identificadores reales ni datos personales. Antes de crear recursos pagos define presupuesto, etiquetas y comando de destrucción; después verifica que no queden recursos huérfanos. Los ejemplos locales enseñan contratos, pero no certifican cumplimiento ni disponibilidad de producción.

## ■ Preguntas de comprobación

1. ¿Cuál es la pregunta de defensa que más pesa, y por qué?
2. ¿Qué distingue un portafolio de criterio de uno de herramientas?
3. ¿Cuál de las cinco predicciones de la clase 168 falló, y en qué mitad?
4. ¿Qué dice la ley 23 y qué consecuencia práctica tiene?
5. ¿Qué cuatro datos del propio proyecto sirven para elegir el siguiente paso?

## Referencias

- Kruchten, P. y otros (2012). Technical debt: from metaphor to theory and practice — deuda declarada frente a deuda accidental. <<https://ieeexplore.ieee.org/document/6336722>>
- Nygard, M. (2011). Documenting architecture decisions — registro con premisas y alternativas. <<https://cognitect.com/blog/2011/11/15/documenting-architecture-decisions>>
- Forsgren, N. y otros (2018). Accelerate — capacidades medibles frente a etiquetas de rol. <<https://itrevolution.com/product/accelerate/>>
- Skelton, M. y Pais, M. (2019). Team Topologies — carga cognitiva y límite de lo que un equipo puede mantener. <<https://teamtopologies.com/book>>

- Google (2025). Cloud Architecture Framework: review and iterate — evaluación con evidencia y revisión periódica. <<https://cloud.google.com/architecture/framework>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 14 en PDF · Manual integral

| Anterior                                     | Índice              | Siguiente                                                            |
|----------------------------------------------|---------------------|----------------------------------------------------------------------|
| ← 179 · Capstone: implementación y operación | Parte 14 · Programa | 181 · Requisitos funcionales, restricciones y atributos de calidad → |

## Evaluación — 180 Capstone: defensa, portafolio y plan profesional

### Preguntas

1. Explica el problema que resuelve esta capacidad sin mencionar un proveedor.
2. Identifica una frontera de responsabilidad y su propietario.
3. Compara dos alternativas mediante confiabilidad, seguridad, costo y operación.
4. ¿Qué demuestra el laboratorio y qué no puede demostrar?
5. Propón un caso límite o una prueba de fallo.

### Reto verificable

Entrega defensa-final con diagrama o configuración, evidencia de ejecución, alternativa descartada, riesgo residual y criterio de rollback.

### Criterio de aceptación

- [ ] Resultado reproducible por una segunda persona.
- [ ] Evidencia vinculada a cada afirmación técnica.
- [ ] Prueba negativa o de fallo incluida.
- [ ] Costos y permisos expresados con unidades y alcance.
- [ ] Limitaciones declaradas sin presentar la simulación como producción.

### Escala

| Nivel             | Evidencia                                                |
|-------------------|----------------------------------------------------------|
| A — competente    | Cumple todo y defiende trade-offs con datos.             |
| B — en desarrollo | Cumple lo esencial; falta profundidad en una dimensión.  |
| C — inicial       | Artefacto parcial o conclusión sin evidencia suficiente. |
| 0                 | Sin entrega reproducible.                                |