

Multi-Cloud Engineering Program

Parte 23 — Capstones por industria y defensa final

12 clases · 84 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	23 de 23
Clases	277–288
Evaluaciones	12
Horas estimadas	84

Índice

Alcance de este cuaderno	2
Índice	3
Parte 23 — Capstones por industria y defensa final	4
277 — Capstone retail: comercio multi-región	6
278 — Capstone financiero: pagos, auditoría y recuperación	16
279 — Capstone salud: privacidad e interoperabilidad	27
280 — Capstone sector público: soberanía y continuidad	37
281 — Capstone media: streaming y distribución global	49
282 — Capstone industria: IoT, edge y operación desconectada	60
283 — Capstone SaaS: multi-tenancy y unit economics	71
284 — Capstone datos e IA: plataforma gobernada	82
285 — Game day integrado y respuesta a incidentes	93
286 — Revisión Well-Architected multi-proveedor	104
287 — Paquete de evidencia, costos y riesgos residuales	114
288 — Defensa final, retrospectiva y plan de 12 meses	124

Parte 23 — Capstones por industria y defensa final

← Parte 22 · Índice completo · Fin

Descargar: Esta parte en PDF · Manual integral

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

Integrar arquitectura, implementación, operación, costo y comunicación en casos industriales completos.

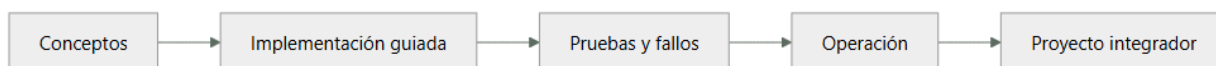
Resultados de la parte

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 capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

Secuencia



Clases

ID	Clase	Laboratorio	Horas
277	Capstone retail: comercio multi-región	capstone	8
278	Capstone financiero: pagos, auditoría y recuperación	capstone	8
279	Capstone salud: privacidad e interoperabilidad	capstone	8
280	Capstone sector público: soberanía y continuidad	capstone	8
281	Capstone media: streaming y distribución global	capstone	8
282	Capstone industria: IoT, edge y operación desconectada	capstone	8
283	Capstone SaaS: multi-tenancy y unit economics	capstone	8
284	Capstone datos e IA: plataforma gobernada	capstone	8
285	Game day integrado y respuesta a incidentes	chaos	4
286	Revisión Well-Architected multi-proveedor	architecture	4
287	Paquete de evidencia, costos y riesgos residuales	assessment	4
288	Defensa final, retrospectiva y plan de 12 meses	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

- Building Evolutionary Architectures — Ford, Parsons y Kua.
- Release It! — Michael Nygard.
- Accelerate — Forsgren, Humble y Kim.

277 — Capstone retail: comercio multi-región

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

Propósito

Primer capstone sectorial: comercio en varias regiones. La clase da el encargo completo, la restricción que manda en este sector —el pico estacional concentrado y la asimetría entre leer y comprar—, las decisiones por capa con sus alternativas, lo que se acepta perder cuando algo falla, y las pruebas negativas que verifican el resultado.

Resultados de aprendizaje

Al finalizar podrás:

1. Identificar la restricción dominante del comercio y qué decide.
2. Separar los recorridos por criticidad y tratarlos distinto.
3. Decidir la estrategia entre regiones con coste y con cifras.
4. Definir qué se degrada y en qué orden bajo carga.
5. Verificar el diseño con las pruebas negativas del sector.

Conceptos centrales

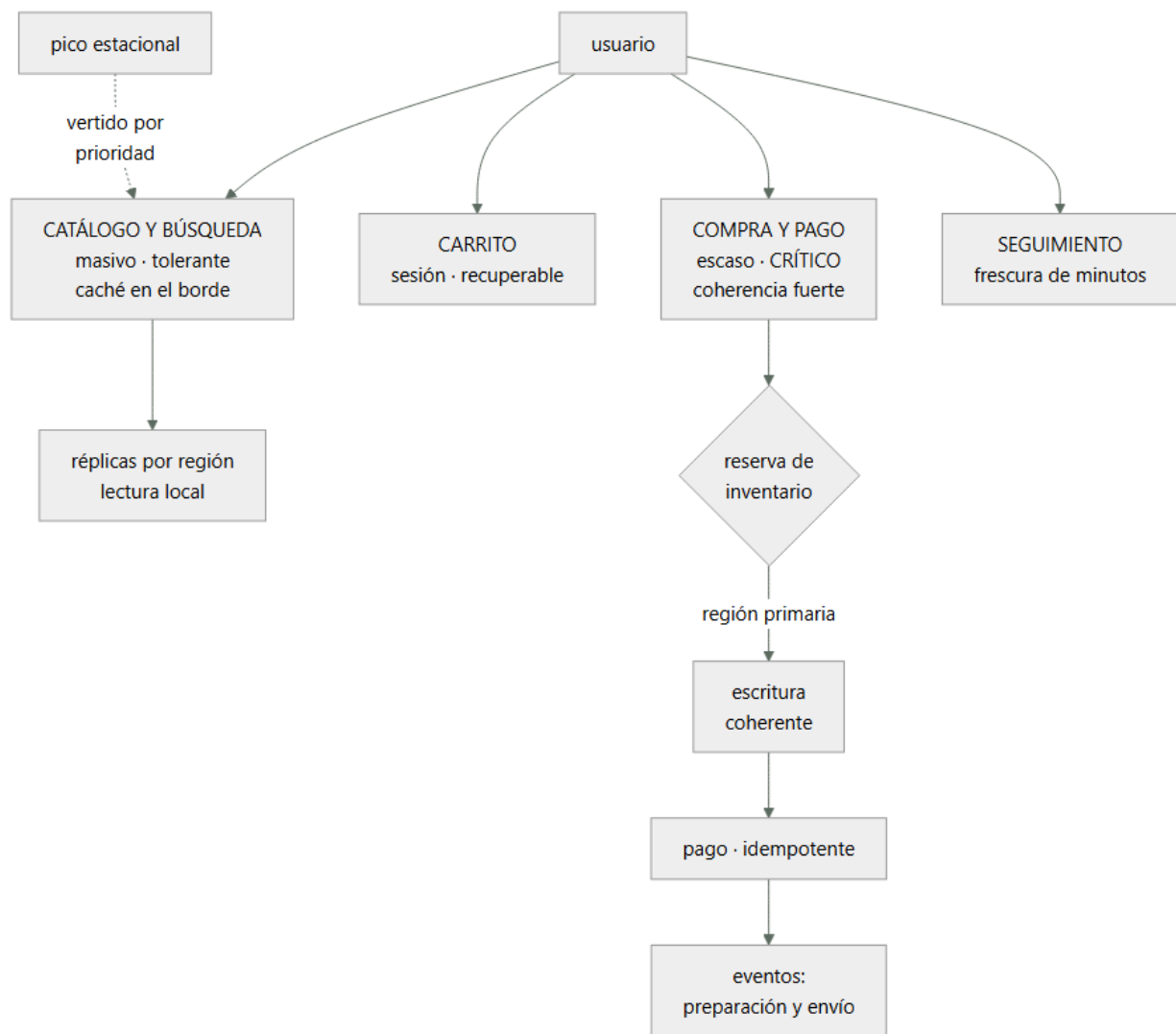
Concepto	Comprensión verificable
asimetría de recorridos	Leer catálogo es masivo y tolerante; comprar es escaso y crítico. Se diseñan por separado.
pico estacional	Concentración de demanda en horas concretas del año. Determina capacidad, cambio y guardia.
reserva de inventario	El punto donde el comercio necesita coherencia fuerte. Casi el único.
sobreventa	Vender más unidades de las existentes. Riesgo aceptable con compensación, o no, según el producto.
degradación por prioridad	Apagar lo accesorio para que lo esencial siga. Decidido antes, con negocio.
residencia por mercado	Obligación de que ciertos datos permanezcan en una región concreta.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 y la restricción que manda

El encargo. CloudShop opera en tres mercados —Europa, Norteamérica y Latinoamérica— con catálogo compartido, inventario por almacén y pago por pasarela local. Hay que diseñar y defender la arquitectura completa, con cifras.

CIFRAS DE PARTIDA

visitas/mes	41 M
búsquedas y vistas de catálogo/día	18 M
pedidos/día	38.000
pico de temporada alta	×3,1 sobre un martes normal
y el evento de campaña	×1,6 durante 40 minutos
referencias en catálogo	2,1 M
almacenes	7
mercados con residencia obligatoria	1 (Europa, datos personales)

Y la restricción que manda, que no es técnica:

EL NEGOCIO SE JUEGA EN HORAS CONCRETAS DEL AÑO

el 31 % de los ingresos anuales ocurre en 6 semanas
y dentro de esas, el 9 % en 3 días

→ consecuencias directas

la capacidad se dimensiona para el pico y se paga todo el año, salvo que sea elástica de verdad

clase 262
la ventana de cambio de esas semanas vale más que
cualquier otra decisión clase 260
y la guardia de esos días necesita gente, no
procedimientos clase 257

→ y en un sector donde una caída de una hora en el día
malo cuesta más que un mes de infraestructura, la
conversación de coste cambia de sentido

Y la segunda restricción, la que ordena el diseño:

LA ASIMETRÍA ENTRE LEER Y COMPRAR
18 M vistas/día frente a 38.000 pedidos/día
→ 474 lecturas por cada compra

y sus exigencias son opuestas
catálogo masivo · tolerante a datos algo viejos ·
caché sirve · latencia importa mucho
compra escaso · exige coherencia · idempotencia ·
latencia importa menos que la corrección

→ tratarlos igual es el error de partida del sector
→ y separarlos permite que el 99,8 % del tráfico se sirva
desde caché regional y el 0,2 % desde la región donde
vive la verdad

2. Las decisiones por capa

Cada una con su alternativa y con lo que empeora.

CATÁLOGO Y BÚSQUEDA

decisión réplicas de solo lectura por región, con
caché en el borde y validez de 5 minutos
alternativa servir todo desde una región
por qué 474 lecturas por compra; la latencia de
catálogo mueve la conversión
empeora un cambio de precio tarda hasta 5 minutos en
verse en todas partes
mitigación invalidación explícita para precio y para
disponibilidad; los demás campos, por caducidad

INVENTARIO Y RESERVA

decisión la reserva se hace en la región del ALMACÉN
que sirve, con coherencia fuerte
alternativa inventario replicado con resolución de
conflictos
por qué la sobreventa de un producto único no se
compensa; la de un producto con reposición, sí
empeora una compra que cruza almacenes de dos regiones
tarda más
y se acepta porque son el 3 % de los pedidos

CARRITO

decisión persistente y asociado a la identidad, no a
la sesión ni a la instancia
alternativa en memoria de la instancia
por qué la clase 261 midió lo que cuesta: el 9 % de
usuarios perdía la compra al retirar una
instancia
empeora una escritura por acción del usuario

PAGO

decisión pasarela local por mercado, con clave de
idempotencia obligatoria y máquina de estados
explícita clase 210
alternativa una pasarela global
por qué las tasas de aprobación locales son entre 4 y
11 puntos mejores, y eso domina cualquier
consideración técnica
empeora tres integraciones que mantener y tres modos
de fallo distintos

PEDIDO Y CUMPLIMIENTO

decisión pedido confirmado = evento; el resto es
asíncrono clase 210
alternativa todo síncrono hasta el almacén
por qué el almacén no puede tumbar la compra
empeora el usuario ve «confirmado» antes de que el
almacén lo sepa
y se acepta con seguimiento por frescura clase 275

DATOS PERSONALES

decisión los de clientes europeos residen en Europa;
el catálogo y los datos agregados, no
alternativa todo en una región
por qué obligación legal del mercado
empeora dos almacenes analíticos y una vista unificada
que hay que construir

Y la estrategia entre regiones:

decisión una región activa por mercado, con la
secundaria en frío y datos replicados
objetivo de recuperación: 30 minutos en
temporada alta, 4 horas el resto del año
→ y esa diferencia es lo interesante
→ durante 6 semanas se mantiene capacidad caliente en la
secundaria; el resto del año no
coste +9.000 USD/mes base; +41.000 en temporada
alternativa activo-activo permanente: +112.000/mes

3. Qué se acepta perder

El diseño no se define por lo que aguanta: se define por lo que se decide sacrificar y en qué orden.

EL ORDEN DE DEGRADACIÓN, acordado con negocio

PRIORIDAD 1 . nunca se apaga
completar una compra ya iniciada
pago y confirmación
→ si esto cae, no hay negocio

PRIORIDAD 2 . se degrada, no se apaga
catálogo y búsqueda
→ si la búsqueda falla, se sirve el catálogo por
categorías desde caché
→ peor experiencia, negocio vivo

PRIORIDAD 3 . se apaga bajo carga
recomendaciones
valoraciones y reseñas
disponibilidad exacta por tienda física
→ y esto se decidió ANTES, no durante clase 262

PRIORIDAD 4 . se vierte
histórico de pedidos
exportaciones y paneles de comerciantes

Y los sacrificios explícitos, que son la parte que cuesta acordar:

SE ACEPTA

que un precio tarde hasta 5 minutos en propagarse
que el seguimiento muestre «actualizado hace X» en vez
de un estado falso
que un pedido aparezca en el almacén con hasta 2 minutos
de retraso
que el histórico no esté disponible en el pico
y sobreventa de hasta el 0,3 % en productos con
reposición, con compensación automática al cliente

NO SE ACEPTA

cobrar dos veces clase 210
perder un pedido confirmado
sobrevender un producto único o personalizado
mostrar el pedido de otro cliente
y que un dato personal europeo salga de Europa

→ y esta lista es el documento más útil del capstone

→ porque durante un incidente nadie tiene tiempo de decidirlo
clase 257

Y las señales que verifican esas promesas:

éxito de compra, medido en el cliente	> 99,9 %
latencia del catálogo, percentil 95	< 800 ms
cobros duplicados	0, alerta inmediata
sobreventa por categoría de producto	panel diario
frescura del seguimiento por transportista	p95 < 10 min
retraso pedido → almacén	p95 < 2 min
y datos personales fuera de región	0, control automático

4. Las pruebas negativas del capstone

Lo que hay que ejecutar para saber si el diseño funciona.

DE CARGA Y CAPACIDAD clase 262

- ¿cuál es el recurso limitante de la compra y dónde está su codo?
- ¿el vertido por prioridad está probado con carga real?
- ¿escalar el servicio empeora algún limitante?
- ¿las cuotas de la región secundaria dan para el 100 %?

DE FALLO clase 261

- retirar una instancia: ¿alguien pierde el carrito?
- +200 ms en inventario: ¿cuánto se degrada la compra?
- la pasarela de un mercado no responde: ¿qué ve el usuario?
- el almacén no consume eventos durante 2 horas: ¿se pierde algún pedido?
- caída de una zona en pico: ¿qué se degrada y en qué orden?

DE CORRECCIÓN

- reenviar el mismo pago tres veces: ¿un solo cobro?
- dos compras simultáneas de la última unidad: ¿una sola gana?
- un evento de estado llega desordenado: ¿se aplica el viejo?
- un evento llega duplicado: ¿se duplica el pedido?

DE DATOS Y NORMATIVA

- ¿algún dato personal europeo sale de Europa? control automático clase 251
- ¿un cliente puede ver el pedido de otro cambiando un identificador?
- ¿la exportación analítica contiene datos personales?

DE OPERACIÓN parte 21

- ¿cuánto tarda restaurar el pedido más reciente?
- ¿la conmutación de región se ha ensayado con carga?
- ¿el procedimiento de sobreventa existe y se ha ejecutado?
- ¿la guardia de temporada alta está dimensionada?

El entregable del capstone:

- 1 diagrama por recorrido, no por servicio
- 2 registro de decisiones con alternativas y coste clase 272
- 3 la lista de lo que se acepta perder, firmada con negocio
- 4 el orden de degradación
- 5 los indicadores por recorrido y sus objetivos clase 268
- 6 el plan de temporada alta: capacidad, congelación, guardia y ensayos
- 7 el coste mensual estimado, base y pico
- 8 y el resultado de las pruebas negativas, con lo que falló

Y el cierre que enlaza con la clase siguiente: el comercio acepta perder cosas y compensar. El siguiente sector no puede: cuando el dato es dinero y hay que probar ante un tercero qué ocurrió, el diseño cambia. Pagos, auditoría y recuperación es la materia de la clase 278.

Ejemplo trabajado

El capstone resuelto, con las cifras de las pruebas negativas. Lo que sigue son las tres decisiones que costaron discusión, lo que falló al ejecutar las pruebas, y el resultado de la temporada alta.

Discusión 1 · Sobreventa: dónde está la coherencia fuerte.

```
posturas
ingeniería  «reservar con coherencia fuerte cuesta
             latencia y limita el caudal»
negocio     «vender algo que no existe destruye la
             confianza»
```

lo que resolvió la discusión: separar por tipo de producto

```
producto con reposición (94 % del catálogo)
→ reserva optimista con reconciliación
→ sobreventa tolerada hasta el 0,3 %
→ compensación automática: cupón y aviso en 2 horas
→ coste de la compensación ~11.000 USD/año
```

```
producto único o personalizado (6 %)
→ reserva con coherencia fuerte en la región del
  almacén
→ cero sobreventa
→ y esos productos son el 23 % del margen
```

→ el coste de la coherencia fuerte solo se paga donde hace falta

→ y la cifra que cerró la discusión fue el coste de las compensaciones frente al de aplicar coherencia fuerte a todo: 11.000 al año frente a 214.000

Discusión 2 · La región activa por mercado.

```
propuesta inicial  una sola región global
latencia de catálogo desde Latinoamérica    310 ms
conversión medida en ese mercado            1,7 %
```

```
prueba controlada
se sirvió el catálogo desde una réplica regional a la
mitad del tráfico durante 3 semanas
latencia          310 ms → 84 ms
conversión         1,7 % → 2,3 %
→ +35 % relativo
```

```
ingresos adicionales anualizados    1,9 M USD
coste de la réplica regional         148.000 USD/año
```

→ decisión tomada con datos, no con principios

→ y el registro guarda la prueba, no solo la conclusión
clase 272

Y lo que empeoró, dicho en el registro:

```
un cambio de precio tarda hasta 5 minutos en verse en
todas las regiones
→ y en una campaña relámpago eso importaba
→ mitigación: invalidación explícita para precio y
  disponibilidad, propagada en < 20 s
→ y los demás campos por caducidad, que es lo barato
```

Las pruebas negativas, ejecutadas.

```
CARGA
codo de la compra          4.100 pet/s
recurso limitante          conexiones a
                             inventario
vertido por prioridad probado sí
```

escalar empeora el limitante	sí, corregido con intermediario
cuotas de la secundaria	3,2 % → 100 %
FALLO	
retirar instancia: ¿se pierde carrito?	no
+200 ms en inventario	compra 1.240 ms (antes 4.100)
pasarela caída de un mercado	se ofrece método alternativo en 1,4 s
almacén sin consumir 2 h	0 pedidos perdidos; cola al día en 9 min
caída de zona en pico	recomendaciones y reseñas apagadas; compra intacta
CORRECCIÓN	
pago reenviado 3 veces	1 cobro
última unidad, 2 compras simultáneas	1 gana, 1 avisa
evento de estado desordenado	descartado
evento duplicado	sin efecto
DATOS	
dato personal europeo fuera de Europa	1 HALLAZGO
cliente ve pedido ajeno	no
exportación con datos personales	1 HALLAZGO
OPERACIÓN	
restaurar el pedido más reciente	4 min 12 s
conmutación ensayada con carga	sí, 22 min
procedimiento de sobreventa	sí, usado 3 veces

Y los dos hallazgos de datos, que eran los graves:

HALLAZGO 1
 el servicio de recomendaciones enviaba el identificador de cliente a un proveedor externo alojado fuera de Europa
 → llevaba 14 meses
 → y el contrato con el proveedor no cubría transferencia
 corrección se sustituyó por un identificador seudonimizado y rotatorio; y el control automático se añadió a la cadena clase 251

HALLAZGO 2
 la exportación analítica nocturna incluía correo electrónico y dirección, y se copiaba a un almacén en otra región
 → la exportación existía desde antes de la separación por mercados ley 25
 corrección seudonimización en el origen y control de esquema que detiene la publicación clase 243

→ los dos son de datos, ninguno da error, y los dos llevaban más de un año ley 29

La temporada alta, con el plan aplicado.

plan
 capacidad para 9.200 pet/s, con vertido preparado
 cuotas ampliadas en ambas regiones de cada mercado
 objetivo de recuperación bajado a 30 min con secundaria caliente durante 6 semanas
 congelación parcial: estándar sí, alto impacto no clase 260
 guardia reforzada: 8 personas, turnos de 8 horas
 y ensayo de conmutación 2 semanas antes

lo que ocurrió
 pico real 10.870 pet/s
 vertido activo 49 min
 peticiones rechazadas 8,4 %

compras completadas	100 % de las intentadas
incidentes de gravedad alta (una pasarela local degradada 26 min; método alternativo ofrecido automáticamente)	1
ingresos de las 6 semanas	+37 % interanual

Y la comparación con el año anterior:

pico soportado	3.800 pet/s	9.200 pet/s
caídas del flujo de compra	2 h 40 min	0
compras completadas en el pico	41 %	100 %
incidentes graves	6	1
cambios desplegados en las 6 semanas	0	214
incidentes causados por ellos	-	0
coste de infraestructura de las 6 semanas	189.000	241.000
ingresos de las 6 semanas	base	+37 %

Y la lectura que el equipo llevó a la defensa:

se gastaron 52.000 USD más en infraestructura durante la temporada
y se desplegaron 214 cambios donde antes 0

→ y no hubo ni una caída del flujo de compra
→ la congelación total del año anterior no había evitado las 2 h 40 de caída: las había concentrado
clase 260

La lección que este capstone deja: la coherencia fuerte se aplicó al 6 % del catálogo —los productos únicos, que son el 23 % del margen— y compensar la sobreventa del resto costó 11.000 USD al año frente a los 214.000 de aplicarla a todo. Y las dos pruebas que fallaron no eran de carga ni de fallo: eran datos personales saliendo de Europa por dos caminos que llevaban más de un año abiertos y que no producían ningún error.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/277-capstone-retail-comercio-multi-region/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 retail-capstone 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 retail-capstone 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 catálogo y la compra se diseñan igual y ninguno queda bien	Se ignoró la asimetría: cientos de lecturas por cada compra, con exigencias opuestas	Separa los recorridos: catálogo con réplicas y caché regional, compra con coherencia donde hace falta.
Aplicar coherencia fuerte a todo el inventario limita el caudal y encarece	No se distinguió el producto con reposición del producto único	Reserva optimista con reconciliación y compensación donde la sobreventa se puede compensar; coherencia fuerte solo donde no.
Durante el pico se decide sobre la marcha qué apagar	El orden de degradación no estaba acordado antes con negocio	Escribe y firma el orden de prioridad y la lista de lo que se acepta perder; en un incidente nadie tiene tiempo de decidirlo.
Se congela todo en temporada alta y aun así hay caídas	La congelación concentra el riesgo en vez de eliminarlo y retiene arreglos de capacidad	Congela solo el alto impacto, manten los cambios estándar con canario más lento y nunca congeles capacidad ni seguridad.
Un dato personal sale de su región sin que nadie lo note	Integraciones y exportaciones anteriores a la separación por mercados siguen activas	Añade control automático de residencia en la cadena y comprobación de esquema que detenga exportaciones con datos personales.
El plan de continuidad no funciona justo en temporada alta	Las cuotas de la región secundaria estaban en valores por defecto y el ensayo nunca levantó carga real	Amplía cuotas en ambas regiones y ensaya la conmutación con carga antes de que empiece la temporada.

■ 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 restricción dominante del comercio y qué tres cosas decide?
2. ¿Por qué se separan catálogo y compra y qué exige cada uno?
3. ¿Dónde hace falta coherencia fuerte en este sector y dónde no?
4. ¿Qué contiene la lista de lo que se acepta perder y por qué se escribe antes?
5. ¿Qué pruebas negativas revelaron los fallos más graves y por qué no daban error?

Referencias

- AWS (2024). Retail reference architectures and Well-Architected retail lens.
<<https://docs.aws.amazon.com/wellarchitected/latest/retail-lens/retail-lens.html>>
- Google Cloud (2024). Retail and commerce architectures.
<<https://cloud.google.com/architecture/ecommerce-web-application>>
- Microsoft (2024). E-commerce architecture on Azure.
<<https://learn.microsoft.com/azure/architecture/industries/retail/>>
- Helland, P. (2007). Life beyond distributed transactions — reservas y compensación.
<<https://queue.acm.org/detail.cfm?id=3025012>>

- Reglamento (UE) 2016/679, RGPD — residencia y transferencias de datos personales.
<<https://eur-lex.europa.eu/eli/reg/2016/679/oj>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 23 en PDF · Manual integral

Anterior	Índice	Siguiente
← 276 · Proyecto: defensa técnica ante panel	Parte 23 · Programa	278 · Capstone financiero: pagos, auditoría y recuperación →

Evaluación — 277 Capstone retail: comercio multi-región

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 retail-capstone 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.

278 — Capstone financiero: pagos, auditoría y recuperación

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

Propósito

Capstone financiero: pagos, auditoría y recuperación. La clase da el encargo, la restricción que manda en este sector —hay que poder demostrar ante un tercero qué ocurrió, y el dinero no se puede perder ni duplicar—, las decisiones que de ahí se derivan, y las pruebas negativas que un auditor y un incidente real ejecutan de verdad.

Resultados de aprendizaje

Al finalizar podrás:

1. Identificar la restricción dominante del sector financiero.
2. Diseñar un flujo de pago idempotente y reconciliable.
3. Construir un rastro de auditoría que sirva como prueba.
4. Definir objetivos de recuperación que la normativa exige y probarlos.
5. Verificar el diseño con las pruebas negativas del sector.

Conceptos centrales

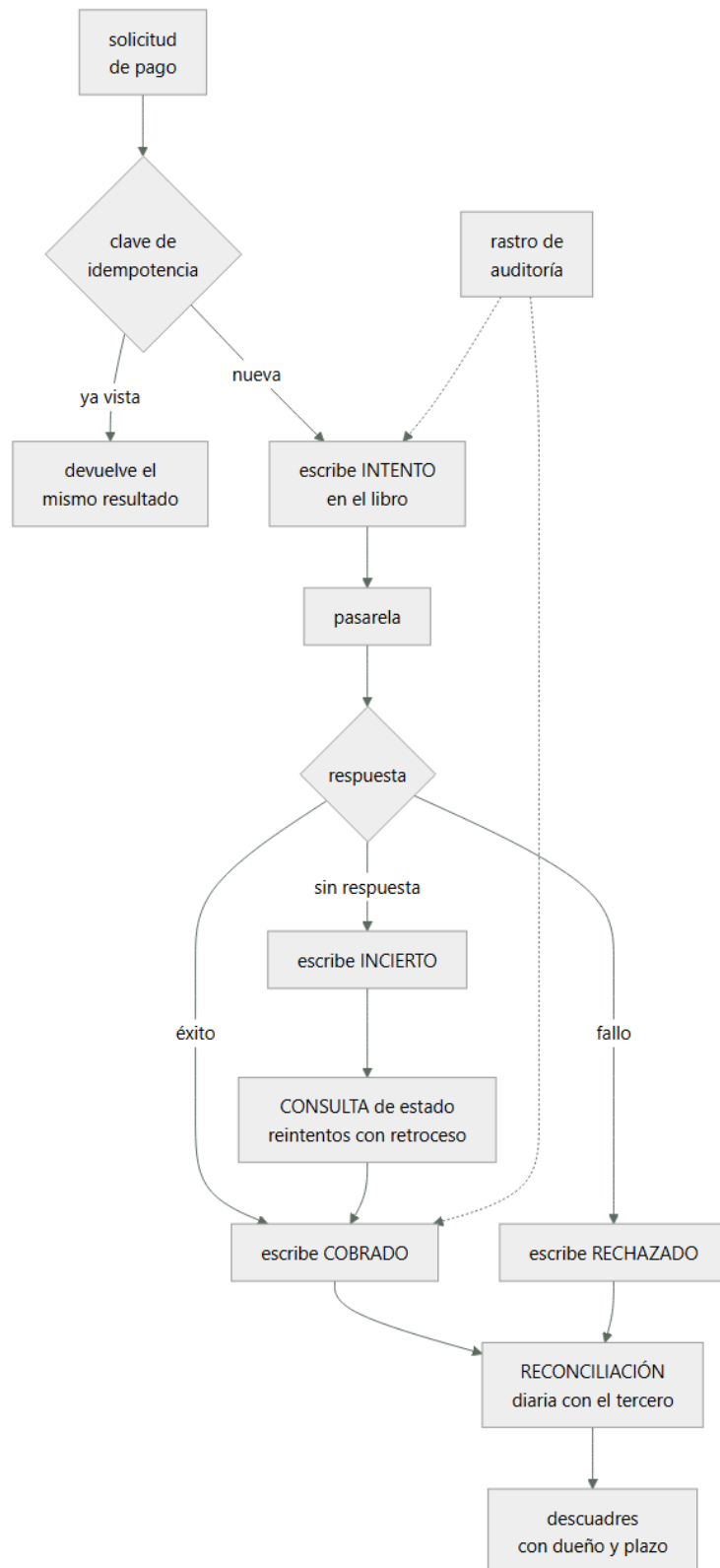
Concepto	Comprensión verificable
libro mayor de eventos	Registro inmutable y ordenado de lo que ocurrió. El saldo es un cálculo, no un dato que se edita.
partida doble	Todo movimiento tiene origen y destino. Hace que el descuadre sea detectable por construcción.
reconciliación	Comparar lo propio con lo del tercero. En pagos, la verdad está fuera.
idempotencia de cobro	Reintentar un pago no cobra dos veces. Requisito absoluto, no deseable.
rastro de auditoría	Registro que permite reconstruir quién hizo qué, cuándo y con qué autorización, ante un tercero.
objetivo de recuperación regulado	Tiempo y pérdida de datos máximos exigidos por la norma, no elegidos por ingeniería.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 y la restricción que manda

El encargo. El brazo de pagos de CloudShop: cobros con tarjeta y transferencia en tres mercados, monedero de cliente con saldo, devoluciones y liquidación diaria a comerciantes. Sujeto a auditoría externa anual y a requisitos de continuidad del regulador.

CIFRAS DE PARTIDA

transacciones/día	41.000
importe medio	68 USD
monederos de cliente con saldo	340.000
comerciantes a liquidar	1.900
pasarelas integradas	3
auditoría externa	anual, con muestreo
y el requisito del regulador	
recuperación	< 2 horas
pérdida de datos admisible	0

Y la restricción que manda, que es doble y no técnica:

- 1 HAY QUE PODER DEMOSTRARLO ANTE UN TERCERO
no basta con que el sistema sea correcto
→ hay que poder reconstruir, meses después, qué pasó
con una transacción concreta, quién la autorizó y con
qué datos
→ y con evidencia que el auditor acepte: inmutable,
completa y con tiempo fiable
 - 2 EL DINERO NO SE PIERDE NI SE DUPLICA
perder un pedido se compensa; perder un cobro, no
→ y cobrar dos veces cuesta más que no cobrar: genera
reclamación, coste de gestión y daño de marca
- y de estas dos salen casi todas las decisiones del
capstone
→ nótese que ninguna es un requisito de rendimiento

Y la consecuencia que ordena el diseño:

- EL SALDO NO ES UN DATO: ES UN CÁLCULO
no se guarda «saldo = 412,30» y se actualiza
se guardan los MOVIMIENTOS y el saldo se deriva
- por qué
un saldo editado no tiene historia y no se puede
auditar
y una escritura perdida deja el saldo mal para siempre
sin dejar rastro ley 29
- y qué se hace por rendimiento
saldos materializados con la posición del último
movimiento aplicado
→ y recalculables desde el origen en cualquier momento
→ esa recalculabilidad es lo que se enseña al auditor

Y la segunda pieza estructural:

- PARTIDA DOBLE
todo movimiento tiene origen y destino, y la suma
cuadra
→ un descuadre se detecta por construcción, no por
casualidad
→ y la comprobación «la suma de todos los saldos es
igual a la suma de todas las entradas menos las
salidas» se ejecuta cada hora
- es la comprobación de calidad más barata y más potente
de todo el capstone clase 243

2. El flujo de pago: idempotencia e incertidumbre

El problema central de pagos no es el éxito ni el fallo: es no saber.

- LOS TRES RESULTADOS POSIBLES
éxito cobrado
fallo rechazado
y el tercero, que es el importante
SIN RESPUESTA
la pasarela no contestó, o contestó tarde, o la red
se cortó
→ puede haber cobrado o no
→ y reintentar sin más puede cobrar dos veces

→ un sistema de pagos se juzga por cómo trata el tercer caso
clase 201

Y el diseño que lo resuelve:

- 1 CLAVE DE IDEMPOTENCIA
la genera el cliente, no el servidor
→ misma clave, mismo resultado, siempre
→ y se conserva al menos 90 días
- 2 SE ESCRIBE ANTES DE LLAMAR
estado INTENTO en el libro, con la clave
→ si el sistema muere entre escribir y llamar, al recuperarse sabe que hay un intento en el aire
- 3 EL ESTADO INCIERTO ES UN ESTADO DE PRIMERA CLASE
no un error
→ y tiene su propio proceso: consultar a la pasarela con reintentos y retroceso hasta resolverlo
→ nunca se reintenta el COBRO: se consulta el ESTADO
- 4 MÁQUINA DE ESTADOS EXPLÍCITA
intento → cobrado | rechazado | incierto
incierto → cobrado | rechazado
cobrado → devuelto (parcial o total)
→ y las transiciones no permitidas se rechazan y se alertan
- 5 Y RECONCILIACIÓN DIARIA
porque la verdad está en el tercero, no en nosotros

Y la reconciliación, que es la red de seguridad real:

CADA DÍA, CONTRA CADA PASARELA
nuestras transacciones frente a las suyas
y se clasifican los descuadres
en nosotros y no en ellos → posible cobro fantasma
en ellos y no en nosotros → cobro no registrado:
el más grave
importes distintos → error de conversión o de comisión
estados distintos → incierto sin resolver

cada descuadre con DUEÑO y PLAZO
y un panel con la antigüedad del descuadre más viejo

→ un sistema de pagos sin reconciliación diaria no es un sistema de pagos
→ y el número que importa no es «cuántos descuadres»:
es «cuánto tarda en cerrarse el más viejo»

3. Auditoría y recuperación

Lo que distingue este sector de los demás: hay que probarlo ante alguien de fuera.

QUÉ HACE ÚTIL UN RASTRO DE AUDITORÍA
INMUTABLE
escritura única; nadie con permisos de administrador puede alterarlo ni borrarlo
clase 255
COMPLETO
quién, qué, cuándo, desde dónde, con qué autorización y con qué valores antes y después
CON TIEMPO FIABLE
reloj sincronizado y desviación vigilada
→ una discrepancia de relojes invalida una secuencia
ENLAZADO
la transacción, su intento, su reconciliación y su liquidación, unidos por un identificador
Y CONSULTABLE
→ un auditor pide «enséñame esta transacción de hace 14 meses»
→ y si tarda tres días en salir, el control se considera inefectivo

→ y el error más común: guardar registros y no poder

RECONSTRUIR una transacción concreta

Y el acceso a datos de pago, que también se audita:

nadie accede a datos de tarjeta en claro
→ tokenización desde el borde; el sistema propio nunca
los ve clase 251
todo acceso a datos de cliente queda registrado con
motivo
acceso de personas mediante elevación temporal
clase 256
y separación de funciones: quien despliega no aprueba
liquidaciones

Y la recuperación, que aquí no la elige ingeniería:

EL REGULADOR EXIGE
recuperación < 2 horas
pérdida de datos admisible: 0

→ y «0 de pérdida» cambia el diseño entero
replicación síncrona para el libro de movimientos
→ con el coste de latencia que eso implica
y confirmación al cliente solo tras la escritura
replicada

→ lo que NO exige cero pérdida
paneles, informes, catálogos de comerciantes
→ y separarlo permite pagar la replicación síncrona solo
donde hace falta

Y lo que la norma obliga a demostrar, no solo a tener:

restauración probada, con acta y con reloj
conmutación ensayada, al menos anualmente clase 261
procedimientos ejecutables y con fecha de última
ejecución clase 259
y registro de quién puede autorizar qué, revisado

→ y aquí la ley 22 deja de ser una observación y pasa a
ser un hallazgo de auditoría

4. Las pruebas negativas del capstone

Lo que hay que ejecutar. Varias de estas las ejecuta un auditor tal cual.

DE CORRECCIÓN DEL DINERO

- enviar el mismo pago 5 veces con la misma clave:
¿un solo cobro?
- enviar el mismo pago con claves distintas: ¿se detecta
el duplicado por otro medio?
- cortar la red justo tras llamar a la pasarela: ¿queda
en incierto y se resuelve solo?
- dos devoluciones simultáneas de la misma transacción:
¿una sola?
- devolver más de lo cobrado: ¿se rechaza?
- ¿la suma de saldos cuadra con los movimientos, ahora?
- recalcular todos los saldos desde el origen: ¿coincide
con lo materializado?

DE RECONCILIACIÓN

- ¿cuántos descuadros abiertos hay y cuál es el más
viejo?
- inyectar una transacción en la pasarela que no exista
en el sistema: ¿la detecta la reconciliación del día
siguiente?
- ¿qué pasa si una pasarela envía el fichero con 6 horas
de retraso?

DE AUDITORÍA

- pedir la reconstrucción de una transacción de hace 14
meses: ¿cuánto se tarda?
- intentar modificar un registro de auditoría con
permisos de administrador: ¿se puede?
- ¿qué desviación tienen los relojes de los servicios?

- ¿queda registrado quién consultó datos de un cliente concreto y por qué?

DE RECUPERACIÓN

- restaurar el libro de movimientos: ¿cuánto tarda y cuánto se pierde?
- conmutar de región con carga: ¿se cumplen las 2 horas?
- ¿la réplica síncrona está realmente síncrona? mácala y comprueba
- ¿las copias sobreviven a un administrador comprometido?

DE OPERACIÓN Y ACCESO

- ¿alguien puede ver datos de tarjeta en claro?
- ¿quien despliega puede aprobar una liquidación?
- ¿hay credenciales permanentes con acceso al libro?

El entregable del capstone:

- 1 el modelo de datos del libro, con partida doble
- 2 la máquina de estados del pago, con el estado incierto
- 3 el diseño de reconciliación y su panel de antigüedad
- 4 el rastro de auditoría y una reconstrucción de ejemplo
- 5 el plan de continuidad con los números del regulador y su prueba
- 6 la matriz de separación de funciones
- 7 el coste, incluido el de la replicación síncrona
- 8 y el resultado de las pruebas negativas, con lo que falló

Y el cierre que enlaza con la clase siguiente: aquí el dato es dinero y hay que probar ante un tercero. En el siguiente sector el dato es una persona, el daño de una filtración no se compensa con dinero, y hay que compartirlo igualmente con otros. Salud, privacidad e interoperabilidad es la materia de la clase 279.

Ejemplo trabajado

El capstone resuelto. Lo que sigue es el estado incierto que apareció 41 veces al mes, la reconciliación que encontró 19 cobros no registrados, y la prueba de auditoría que suspendió la primera vez.

El estado incierto, medido.

transacciones/mes	1.240.000
éxito directo	1.238.900
rechazo	...
SIN RESPUESTA de la pasarela	41

- 41 de 1,24 millones parece despreciable
- y son 41 casos al mes de «puede que hayamos cobrado o puede que no»

- y lo que pasaba antes del rediseño
- el sistema reintentaba el cobro
- cobros duplicados detectados por reclamación 9/mes
- coste de gestión por reclamación ~40 USD
- y 4 de esos 9 acababan en incidencia con el banco

Y el rediseño:

estado INCIERTO explícito, con su proceso
 consulta de estado a la pasarela, con retroceso
 a los 5 s, 15 s, 60 s, 5 min, 30 min, 2 h, 6 h
 y nunca se reintentaba el cobro

resultado a los 6 meses	
casos sin respuesta	43/mes
resueltos automáticamente en < 5 min	38
resueltos en < 6 h	4
escalados a persona	1
COBROS DUPLICADOS	0

- y el usuario ve «estamos confirmando tu pago» durante esos segundos, en vez de un error que le invita a reintentar
- ese texto, que parece cosmético, eliminó el 71 % de los

reintentos manuales del usuario

La reconciliación, primer mes.

primera ejecución contra las tres pasarelas, con 90 días de histórico

transacciones nuestras	3.690.000
transacciones de las pasarelas	3.690.019
descuadres	61
en nosotros y no en ellos	9
→ intentos que nunca llegaron; sin impacto	
EN ELLOS Y NO EN NOSOTROS	19
→ cobros reales no registrados	
→ importe total	12.400 USD
→ antigüedad media	47 días
importes distintos	28
→ comisiones aplicadas distinto en conversión de moneda	
estados distintos	5

Y el análisis de los 19 graves:

17 de los 19 venían del mismo escenario
el proceso moría entre recibir la respuesta de la pasarela y escribir en el libro
→ la pasarela había cobrado; nosotros no lo sabíamos
→ y el cliente había recibido el cargo sin ver el pedido

→ y ninguno había generado una alerta
→ los 17 se conocieron por reclamación del cliente, uno a uno, o no se conocieron ley 29

corrección estructural
escribir el INTENTO antes de llamar
→ así el proceso de recuperación encuentra los intentos en el aire al arrancar
y la reconciliación diaria como red final

resultado a los 12 meses
descuadres del tipo «en ellos y no en nosotros» 0
descuadres totales por mes 3-9
antigüedad del descuadre más viejo < 26 horas

Y la métrica que el equipo puso en el panel principal:

no «número de descuadres»
sino ANTIGÜEDAD DEL DESCUADRE MÁS VIEJO
→ porque un descuadre nuevo es normal
→ uno de 47 días es un fallo de proceso

La prueba de auditoría, que suspendió.

simulacro interno, antes de la auditoría real
el equipo de cumplimiento hizo de auditor

PETICIÓN 1

«reconstruye esta transacción de hace 14 meses:
quién la inició, con qué datos, qué autorizó la pasarela, cuándo se liquidó al comerciante y quién ha consultado esos datos desde entonces»

resultado
 la transacción 9 minutos
 la respuesta de la pasarela 2 días
 → estaba en registros archivados sin índice
 la liquidación 4 horas
 → en otro sistema, sin identificador común
 quién la consultó NO DISPONIBLE
 → no se registraban las consultas de lectura

→ SUSPENSO
→ y el hallazgo «no se puede demostrar quién accedió a los datos de un cliente» es de los que cierran una

auditoría en mal lugar

Y las correcciones:

identificador de correlación único, presente en
transacción, respuesta de pasarela, liquidación y
registro de auditoría clase 211
respuestas de pasarela guardadas en almacén indexado y
consultable, 7 años, inmutable clase 255
registro de LECTURAS de datos de cliente, con motivo
→ y esto generó discusión: añade escrituras y volumen
→ se resolvió registrando por sesión y por cliente
consultado, no por consulta
y una consulta guardada que reconstruye una transacción
entera

resultado del segundo simulacro
reconstrucción completa 4 minutos
quién la consultó disponible
→ APROBADO

y la auditoría real, 4 meses después
hallazgos 2, menores
hallazgos del año anterior 11

La continuidad exigida por el regulador.

exigencia < 2 horas de recuperación, 0 de pérdida

diseño
libro de movimientos con replicación síncrona a una
segunda región
confirmación al cliente solo tras la escritura replicada
todo lo demás, asíncrono

coste de esa decisión
latencia añadida a la confirmación de pago +34 ms
coste de la replicación síncrona +19.000 USD/mes
→ y aplicarla a todo el sistema habría costado 71.000

la prueba, con reloj y con acta
conmutación completa con carga real
tiempo hasta servir tráfico 41 minutos
transacciones perdidas 0
descuadres tras la conmutación 0
→ y el acta se entregó al auditor

y el hallazgo del primer ensayo, 9 meses antes
la réplica «síncrona» estaba configurada como asíncrona
desde una migración
→ se descubrió matándola: se perdieron 4 transacciones
en el ensayo
→ en producción habrían sido reclamaciones sin
explicación ley 15

Las cifras finales del capstone.

cobros duplicados	9/mes	0
cobros no registrados	19/90 días	0
antigüedad del descuadre más viejo	47 días	26 h
reclamaciones por pago	41/mes	6/mes
reconstrucción de una transacción		
antigua	2 días	4 min
registro de lecturas de datos de cliente	no	sí
hallazgos de auditoría	11	2
conmutación de región probada, con acta	no	41 min
pérdida en la conmutación	n/d	0
replicación síncrona verificada	no (era asíncrona)	sí
coste mensual adicional por continuidad	-	19.000 USD
coste de las reclamaciones evitadas (estimado)	-	16.800 USD

La lección que este capstone deja: 41 transacciones al mes sin respuesta parecen despreciables sobre 1,24 millones, y eran 9 cobros duplicados al mes hasta que el estado incierto dejó de tratarse como un error y pasó a ser un estado con su propio proceso. Y la réplica que garantizaba cero pérdida de datos ante el regulador llevaba meses configurada como asíncrona: solo se supo al matarla en un ensayo.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/278-capstone-financiero-pagos-auditoria-y-recuperacion/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 finance-capstone 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 finance-capstone 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 cobros duplicados en las reclamaciones de clientes	Se reintenta el cobro cuando la pasarela no responde	Trata la ausencia de respuesta como estado incierto de primera clase: nunca reintentes el cobro, consulta el estado con retroceso hasta resolverlo.
Hay cobros del banco que el sistema no tiene registrados	El proceso muere entre recibir la respuesta y escribir, y no hay reconciliación	Escribe el intento antes de llamar y reconcilia a diario contra el tercero; vigila la antigüedad del descuadre más viejo, no su número.
El auditor pide reconstruir una transacción antigua y se tarda días	Los registros existen en sistemas distintos sin identificador común ni índice	Usa un identificador de correlación en toda la cadena, guarda las respuestas del tercero en almacén indexado e inmutable y deja la consulta preparada.
No se puede demostrar quién consultó los datos de un cliente	Solo se registran escrituras, no lecturas	Registra los accesos de lectura con motivo, agregando por sesión y cliente consultado para que el volumen sea manejable.
El saldo de un cliente quedó mal y no hay forma de saber por qué	El saldo se guarda y se edita en vez de derivarse de los movimientos	Guarda movimientos con partida doble y materializa el saldo con la posición aplicada; comprueba cada hora que la suma cuadra.
La replicación que garantizaba cero pérdida no era síncrona	Una migración cambió la configuración y nadie lo comprobó	Mata la réplica en un ensayo y mide la pérdida real; una garantía no verificada no es una garantía.

■ 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 dos restricciones que mandan en este sector?
2. ¿Por qué el saldo debe ser un cálculo y no un dato?
3. ¿Cómo se trata la ausencia de respuesta de una pasarela?
4. ¿Qué hace que un rastro de auditoría sirva ante un tercero?
5. ¿Qué prueba revela que una réplica síncrona no lo es?

Referencias

- PCI Security Standards Council (2024). PCI DSS v4.0. <<https://www.pcisecuritystandards.org/documentlibrary/>>
- AWS (2024). Financial Services Industry Lens, Well-Architected Framework. <<https://docs.aws.amazon.com/wellarchitected/latest/financial-services-industry-lens/financial-services-industry-lens.html>>
- Microsoft (2024). Azure financial services regulatory compliance. <<https://learn.microsoft.com/azure/industry/financial/>>
- Helland, P. (2007). Life beyond distributed transactions. <<https://queue.acm.org/detail.cfm?id=3025012>>
- Banco de Pagos Internacionales (2021). Principles for operational resilience. <<https://www.bis.org/bcbs/publ/d516.htm>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 23 en PDF · Manual integral

Anterior	Índice	Siguiente
← 277 · Capstone retail: comercio multi-región	Parte 23 · Programa	279 · Capstone salud: privacidad e interoperabilidad →

Evaluación — 278 Capstone financiero: pagos, auditoría y recuperación

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 finance-capstone 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.

279 — Capstone salud: privacidad e interoperabilidad

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

Propósito

Capstone de salud: privacidad e interoperabilidad. La clase da el encargo y la restricción que manda en este sector —el daño de una filtración no se compensa con dinero, y aun así hay que compartir el dato con otros para que el paciente reciba atención—, la tensión entre esas dos exigencias, y las pruebas negativas que la revelan.

Resultados de aprendizaje

Al finalizar podrás:

1. Identificar la tensión entre privacidad e interoperabilidad y resolverla por capas.
2. Diseñar control de acceso por relación asistencial, no por rol.
3. Aplicar minimización, seudonimización y separación de identidad.
4. Tratar el acceso de emergencia sin abrir una puerta permanente.
5. Verificar el diseño con las pruebas negativas del sector.

Conceptos centrales

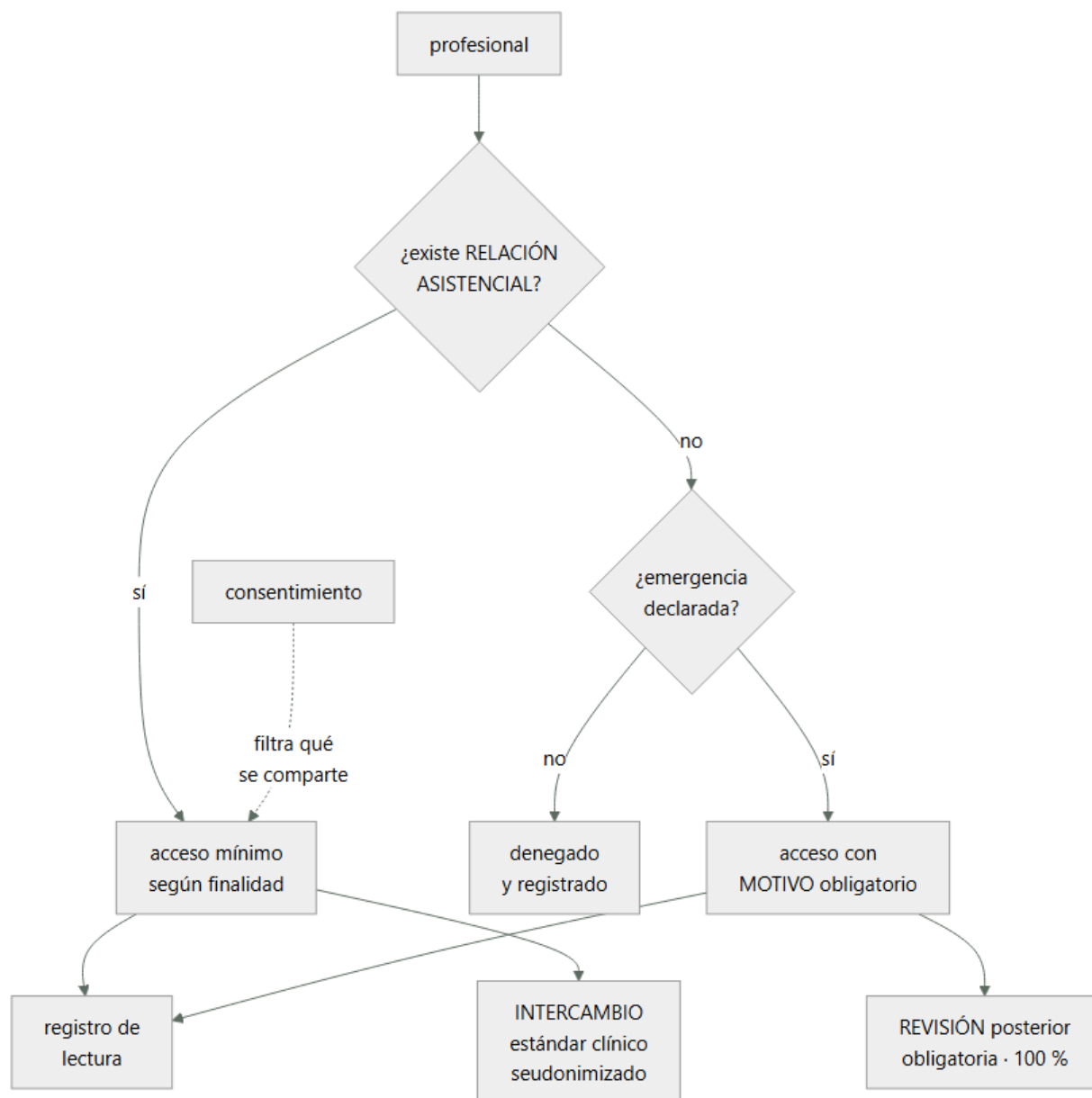
Concepto	Comprensión verificable
relación asistencial	Vínculo entre profesional y paciente que justifica el acceso. Es lo que autoriza, no el cargo.
acceso de emergencia	Mecanismo que permite saltarse el control en urgencias, dejando rastro y con revisión posterior obligatoria.
minimización	Recoger y mostrar solo lo necesario para la finalidad concreta.
seudonimización	Separar los identificadores directos del dato clínico, con la correspondencia guardada aparte.
interoperabilidad	Que otro sistema pueda usar el dato con su significado intacto. Exige estándar, no solo formato.
consentimiento	Permiso del paciente sobre qué se comparte, con quién y para qué. Debe poder retirarse y aplicarse.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 y la tensión que lo define

El encargo. Una plataforma que da soporte a una red de clínicas: historia clínica, resultados de laboratorio, imagen médica, citas y facturación. Debe intercambiar información con hospitales públicos, con laboratorios externos y con la aseguradora.

CIFRAS DE PARTIDA

pacientes	1,4 M
profesionales con acceso	4.100
centros	62
estudios de imagen/mes	41.000
intercambios con terceros/día	18.000
y el requisito legal	
datos de salud: categoría especial	
consentimiento explícito para usos secundarios	
derecho de acceso y de supresión	
y notificación de brecha en 72 horas	

Y la restricción que manda:

EL DAÑO DE UNA FILTRACIÓN NO SE COMPENSA	
un cobro duplicado se devuelve	clase 278

un pedido perdido se repone clase 277
un diagnóstico filtrado no se puede deshacer
→ afecta al empleo, al seguro, a la familia y a la
intimidad, de forma permanente

→ y de ahí sale la primera consecuencia de diseño: el radio de explosión de cualquier credencial de este sistema tiene que ser mínimo

Y la tensión, que es lo que hace difícil el capstone:

Y AUN ASÍ HAY QUE COMPARTIRLO

un paciente que llega a urgencias necesita que su
alergia se conozca
un laboratorio externo tiene que devolver el resultado
el hospital público necesita el informe previo
y la aseguradora necesita justificar el pago

- la privacidad absoluta mata la atención
- y la interoperabilidad sin control mata la privacidad

- el diseño no elige entre las dos: las separa por CAPAS
 - qué se comparte (mínimo por finalidad)
 - con quién (relación y consentimiento)
 - con qué significado (estándar clínico)
 - y con qué rastro (todo acceso registrado)

2. Acceso por relación, no por rol

El error clásico del sector: dar acceso por cargo. Un médico no debe poder ver a cualquier paciente por ser médico.

EL MODELO POR ROL, que falla

- «rol: médico» → acceso a la historia clínica
- 4.100 profesionales pueden ver 1,4 M de historias
- radio de explosión: total
- y es indistinguible un acceso legítimo de una curiosidad

EL MODELO POR RELACIÓN ASISTENCIAL

el acceso lo autoriza el VÍNCULO, no el cargo profesional asignado a una cita con ese paciente profesional del servicio donde el paciente está ingresado profesional al que se le ha derivado el caso o profesional del equipo del anterior, durante la atención

y la relación tiene VIGENCIA

- termina cuando termina el episodio
- y con un margen definido, no indefinido

- pasa de «4.100 pueden ver a 1,4 M» a «cada profesional ve, de media, a 340 pacientes activos»
- tres órdenes de magnitud de radio de explosión

Y las dimensiones que se combinan:

RELACIÓN	¿hay vínculo asistencial vigente?
FINALIDAD	¿para qué? asistencia · facturación · investigación · calidad → y cada finalidad ve un subconjunto distinto
CONSENTIMIENTO	¿el paciente permite ese uso? → y puede excluir apartados concretos
SENSIBILIDAD	hay datos con protección reforzada → salud mental, adicciones, reproductiva, genética → y estos exigen consentimiento explícito adicional

→ y las cuatro se evalúan en el momento del acceso, no al crear la cuenta

Y el caso que rompe cualquier control: la urgencia.

EL ACCESO DE EMERGENCIA

un paciente inconsciente en urgencias, sin relación
previa
→ si el control lo impide, el sistema se rodea o alguien
muere ley 16

el diseño correcto
existe, y es UN CLIC
exige MOTIVO escrito, no una casilla
concede acceso limitado en alcance y en tiempo
genera un aviso inmediato al responsable de privacidad
y su revisión posterior es OBLIGATORIA, del 100 %

- la clave no es dificultar el acceso: es que sea imposible que pase desapercibido
- y el número que se vigila es cuántos accesos de emergencia hay por profesional y por mes
- uno es normal; treinta es un patrón

3. Minimización, separación e interoperabilidad

Cómo se reduce el daño posible antes de que ocurra.

SEPARACIÓN DE IDENTIDAD Y CLÍNICA

los identificadores directos –nombre, documento, dirección, teléfono– en un almacén
el dato clínico en otro, con un identificador interno
y la correspondencia, con acceso propio y registrado

- una filtración del almacén clínico no identifica a nadie por sí sola
- y una del de identidad no dice nada clínico
- el coste: una unión más en cada consulta y disciplina para no desnormalizar «por comodidad»

MINIMIZACIÓN POR FINALIDAD

facturación no necesita el diagnóstico completo: necesita el código de procedimiento
investigación no necesita identidad: necesita cohortes
calidad no necesita el texto libre

- y el texto libre es el mayor problema de la minimización: contiene de todo y no se puede filtrar por campo
- se trata como el dato más sensible del sistema

Y EL BORRADO

el derecho de supresión choca con la obligación de conservar la historia clínica durante años
→ se resuelve distinguiendo: lo asistencial se conserva por obligación legal; lo secundario –marketing, analítica– se borra
→ y esa distinción tiene que estar en el modelo de datos desde el principio, no añadirse después

Y la interoperabilidad, que es más que un formato:

COMPARTIR NO ES ENVIAR UN FICHERO

hace falta que el receptor entienda lo mismo
identificadores de paciente que se puedan conciliar
vocabularios clínicos codificados, no texto
unidades explícitas
y contexto: quién lo midió, cuándo y con qué método

- un resultado de laboratorio sin unidad ni método no es interoperable: es peligroso
- y aquí el contrato de datos de la parte 20 deja de ser buena práctica y pasa a ser seguridad del paciente
clase 241

Y LA CONCILIACIÓN DE PACIENTES

el mismo paciente con identificadores distintos en dos sistemas
→ unirlos mal mezcla dos historias clínicas
→ y ese es el error más grave posible en este sector

→ por eso la conciliación automática exige umbral alto y revisión humana en la zona dudosa

4. Las pruebas negativas del capstone

Varias de estas las ejecuta una inspección tal cual.

DE ACCESO

- un profesional sin relación con un paciente, ¿puede verlo?
- ¿sigue viéndolo tres meses después del alta?
- ¿un administrativo de facturación ve el diagnóstico completo?
- ¿un profesional puede buscar por nombre a alguien con quien no tiene relación?
- ¿cuántos accesos de emergencia hubo el mes pasado y cuántos se revisaron?
- ¿alguien con permisos de administrador de base puede leer datos clínicos?

DE CONSENTIMIENTO

- retirar un consentimiento: ¿deja de compartirse desde ese momento?
- ¿y lo ya compartido con terceros?
- ¿los apartados con protección reforzada exigen consentimiento adicional?

DE MINIMIZACIÓN

- ¿qué campos viajan a la aseguradora? ¿alguno sobra?
- ¿la exportación para investigación permite reidentificar?
- ¿el texto libre sale del sistema en alguna integración?
- ¿los entornos de prueba tienen datos reales?

DE INTEROPERABILIDAD

- enviar un resultado sin unidad: ¿se rechaza?
- dos pacientes con nombre y fecha de nacimiento iguales: ¿se fusionan?
- ¿un código clínico desconocido se descarta en silencio o se detiene? clase 243
- una imagen médica con metadatos identificativos: ¿se limpian?

DE OPERACIÓN Y BRECHA

- simular una filtración: ¿en cuánto se detecta y se puede notificar en 72 horas?
- ¿se sabe exactamente qué registros se vieron afectados?
- ¿las copias contienen datos identificables y quién accede a ellas?
- restaurar la historia clínica: ¿cuánto tarda?

El entregable del capstone:

- 1 el modelo de acceso por relación, finalidad, consentimiento y sensibilidad
- 2 el diseño del acceso de emergencia y su revisión
- 3 la separación de identidad y clínica, con su coste
- 4 el mapa de qué campos salen a cada tercero y por qué
- 5 el diseño de conciliación de pacientes y su umbral
- 6 el plan de respuesta a brecha, con el ensayo hecho
- 7 la política de conservación y borrado, distinguiendo finalidad
- 8 y el resultado de las pruebas negativas, con lo que falló

Y el cierre que enlaza con la clase siguiente: aquí el límite lo pone la protección de una persona. En el siguiente sector lo pone la soberanía de un Estado sobre sus datos y su continuidad, con requisitos que no se negocian con ingeniería. Sector público es la materia de la clase 280.

Ejemplo trabajado

El capstone resuelto. Lo que sigue es el cambio de acceso por rol a acceso por relación con sus cifras, los accesos de emergencia que revelaron un patrón, y las cuatro pruebas que fallaron.

El cambio de modelo de acceso.

ANTES · por rol	
profesionales con acceso a historia clínica	4.100
pacientes visibles por cada uno	1,4 M
accesos/mes	890.000
accesos revisados	0

→ y una auditoría interna preguntó: «¿cuántos de esos 890.000 accesos eran legítimos?»

→ la respuesta honesta era «no se puede saber»

Y la medición previa al cambio, que fue lo que lo justificó:

se cruzaron los accesos con las relaciones asistenciales existentes en la agenda y en los ingresos

accesos con relación asistencial clara	847.100	95,2 %
accesos sin relación identificable	42.900	4,8 %

y de esos 42.900, muestreo manual de 400

cobertura de compañero (equipo real, relación no registrada)	291	72,8 %
continuidad asistencial fuera de episodio	74	18,5 %
gestión administrativa legítima	21	5,3 %
SIN JUSTIFICACIÓN APARENTE	14	3,5 %

→ extrapolado: ~1.500 accesos al mes sin justificación

→ incluidos 3 casos de un profesional consultando a personas de su municipio sin relación con la clínica

Y el modelo nuevo:

relación asistencial derivada automáticamente de
cita asignada · ingreso activo · derivación ·
pertenencia al equipo tratante
vigencia: episodio + 90 días
y para todo lo demás, acceso de emergencia con motivo

resultado a los 6 meses

pacientes visibles por profesional, media	340
accesos/mes	861.000
accesos denegados por falta de relación	4.100
de los cuales, convertidos en emergencia	2.900
y abandonados	1.200
accesos de emergencia/mes	2.900
revisados	100 %

Y lo que la revisión del 100 % encontró:

de 2.900 accesos de emergencia mensuales	
justificados por cobertura o continuidad	2.847
con motivo insuficiente, aclarados	41
ESCALADOS a privacidad	12

y el patrón que apareció al mirar por profesional

media de accesos de emergencia por profesional	0,7/mes
y un profesional con	31/mes

→ todos a pacientes sin ninguna relación con su servicio

→ investigación interna; caso confirmado de acceso indebido

→ con el modelo por rol, esos 31 accesos habrían sido

indistinguibles de los otros 890.000

→ el control no impidió el acceso: lo hizo VISIBLE

→ y eso es lo que se buscaba clase 269

Las cuatro pruebas que fallaron.

FALLO 1 · el entorno de prueba

- ¿los entornos de prueba tienen datos reales?
- Sí: una copia de producción de hacía 14 meses
- con 1,1 M de historias completas
- accesible por 34 personas, incluidos 6 proveedores externos
- y sin registro de acceso
- era, con diferencia, el mayor riesgo del sistema
- y no estaba en ningún inventario de riesgo ley 24

corrección

- generación de datos sintéticos con la misma forma y, donde hacía falta realismo, seudonimización irreversible con desplazamiento de fechas
- los entornos dejaron de tener datos reales en 7 semanas

FALLO 2 · el texto libre en la integración con la aseguradora

- el mensaje de facturación incluía el campo «observaciones clínicas» completo
- porque la especificación original lo pedía «por si acaso» ley 25
- 4 años enviándolo

muestreo de 200 mensajes

con información clínica sensible en el texto	87
con datos de terceros (familiares)	9

corrección

- el campo se retiró
- y se añadió control de esquema que detiene el envío si aparece un campo no declarado clase 243

FALLO 3 · retirada de consentimiento

- retirar el consentimiento de uso secundario dejaba de compartir a partir de ese momento
- y lo ya enviado al almacén de investigación seguía ahí
- y ese almacén permitía reidentificar por cruce de fecha de nacimiento, código postal y fecha de episodio

corrección

- el almacén de investigación pasó a guardar cohortes con generalización de fechas y de zona
- y la retirada de consentimiento dispara la eliminación del registro individual, con acta

FALLO 4 · conciliación de pacientes

- dos pacientes con el mismo nombre y la misma fecha de nacimiento
- el algoritmo los fusionó
- detectado en la prueba, no en producción
- si hubiera pasado en producción, dos historias clínicas mezcladas
- que es el peor error posible de este sector

corrección

- umbral de fusión automática elevado
- zona dudosa a revisión humana obligatoria
- y fusión REVERSIBLE, con registro de qué se unió

El ensayo de brecha.

escenario «una credencial de un profesional se usa desde una dirección desconocida y descarga 4.100 historias en 40 minutos»

lo que se midió

tiempo hasta detectar	6 minutos
(regla de volumen de lectura anómalo por profesional)	
tiempo hasta cortar el acceso	9 minutos

tiempo hasta saber QUÉ registros se vieron 3 h 20
→ y esto era lo que había que medir
tiempo hasta poder notificar con precisión 4 h 10

→ dentro de las 72 horas, con margen
→ y el tramo largo fue reconstruir el alcance exacto, no detectar

corrección

el registro de lecturas pasó a incluir el identificador de cada registro consultado, no solo la sesión
→ coste: +31 % de volumen de registro
→ beneficio: alcance exacto en 11 minutos en el segundo ensayo

Las cifras finales del capstone.

pacientes visibles por profesional	1,4 M	340
accesos sin justificación (estimado)	1.500/mes	~0/mes
accesos de emergencia revisados	0 %	100 %
casos de acceso indebido detectados	0	1
entornos de prueba con datos reales	sí	no
campos innecesarios enviados a terceros	1	0
reidentificación posible en investigación	sí	no
fusiones erróneas de pacientes	no medido	0 (con revisión)
detección de brecha simulada	n/d	6 min
alcance exacto de la brecha	n/d	11 min
notificación posible en 72 h	no	sí
coste añadido (separación, registro, datos sintéticos)	-	21.000 USD/mes

La lección que este capstone deja: el modelo por rol permitía a 4.100 profesionales ver 1,4 millones de historias, y el cambio a relación asistencial no impidió los accesos indebidos —los hizo visibles: un profesional con 31 accesos de emergencia al mes frente a una media de 0,7 apareció en la primera revisión. Y el mayor riesgo del sistema no estaba en producción: era una copia completa de 1,1 millones de historias en los entornos de prueba, accesible por 34 personas y sin registro.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/279-capstone-salud-privacidad-e-interoperabilidad/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 health-capstone 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 health-capstone 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
Cualquier profesional puede consultar cualquier historia clínica	El acceso se concede por rol y no por relación asistencial	Deriva la relación de citas, ingresos y derivaciones, dale vigencia limitada y evalúa relación, finalidad, consentimiento y sensibilidad en el momento del acceso.
El control de acceso se rodea en urgencias	No existe una vía de emergencia y bloquear pone en riesgo al paciente	Ofrece acceso de emergencia a un clic con motivo escrito, aviso inmediato y revisión posterior del 100 %; el objetivo es visibilidad, no fricción.
Hay datos reales en entornos de prueba	Se copió producción por comodidad y nunca se revisó	Genera datos sintéticos con la misma forma y, donde haga falta realismo, seudonimiza de forma irreversible desplazando fechas.
Se envían campos innecesarios a un tercero	La especificación original pedía el campo completo y nadie lo revisó	Mapea qué campo sale a cada tercero con su finalidad y añade control de esquema que detenga el envío de campos no declarados.
Retirar el consentimiento no afecta a lo ya compartido	El almacén secundario guarda registros individuales reidentificables	Guarda cohortes generalizadas y haz que la retirada dispare la eliminación del registro individual, con acta.
Se fusionan dos pacientes distintos	El umbral de conciliación automática es bajo y la fusión es irreversible	Eleva el umbral, manda la zona dudosa a revisión humana y haz la fusión reversible con registro de qué se unió.

■ 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 tensión que define este sector y cómo se resuelve por capas?
2. ¿Por qué el acceso por rol falla y qué lo sustituye?
3. ¿Cómo se diseña el acceso de emergencia sin abrir una puerta permanente?
4. ¿Qué aporta separar identidad y dato clínico y qué cuesta?
5. ¿Qué hace que un dato clínico sea interoperable y no solo transferible?

Referencias

- HL7 (2024). FHIR: Fast Healthcare Interoperability Resources. <<https://www.hl7.org/fhir/>>
- Reglamento (UE) 2016/679, RGPD — datos de salud como categoría especial. <<https://eur-lex.europa.eu/eli/reg/2016/679/oj>>

- AWS (2024). Healthcare Industry Lens, Well-Architected Framework. <<https://docs.aws.amazon.com/wellarchitected/latest/healthcare-industry-lens/healthcare-industry-lens.html>>
- Microsoft (2024). Azure for healthcare: compliance and data protection. <<https://learn.microsoft.com/azure/architecture/industries/healthcare/>>
- El Emam, K. (2013). Guide to the De-Identification of Personal Health Information. <<https://www.routledge.com/Guide-to-the-De-Identification-of-Personal-Health-Information/Emam/p/book/9781466579064>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 23 en PDF · Manual integral

Anterior	Índice	Siguiente
← 278 · Capstone financiero: pagos, auditoría y recuperación	Parte 23 · Programa	280 · Capstone sector público: soberanía y continuidad →

Evaluación — 279 Capstone salud: privacidad e interoperabilidad

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 health-capstone 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.

280 — Capstone sector público: soberanía y continuidad

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

Propósito

Capstone de sector público: soberanía y continuidad. La clase da el encargo y la restricción que manda —el servicio no se puede sustituir por otro y el Estado responde de que siga funcionando y de que sus datos no queden bajo jurisdicción ajena—, lo que eso obliga técnicamente, y las pruebas negativas que verifican que la soberanía declarada es real.

Resultados de aprendizaje

Al finalizar podrás:

1. Distinguir residencia, soberanía y control operativo efectivo.
2. Diseñar para continuidad cuando no existe un competidor al que acudir.
3. Planificar la reversibilidad de proveedor como requisito, no como discurso.
4. Atender a toda la población, incluida la que no tiene buena conexión.
5. Verificar el diseño con las pruebas negativas del sector.

Conceptos centrales

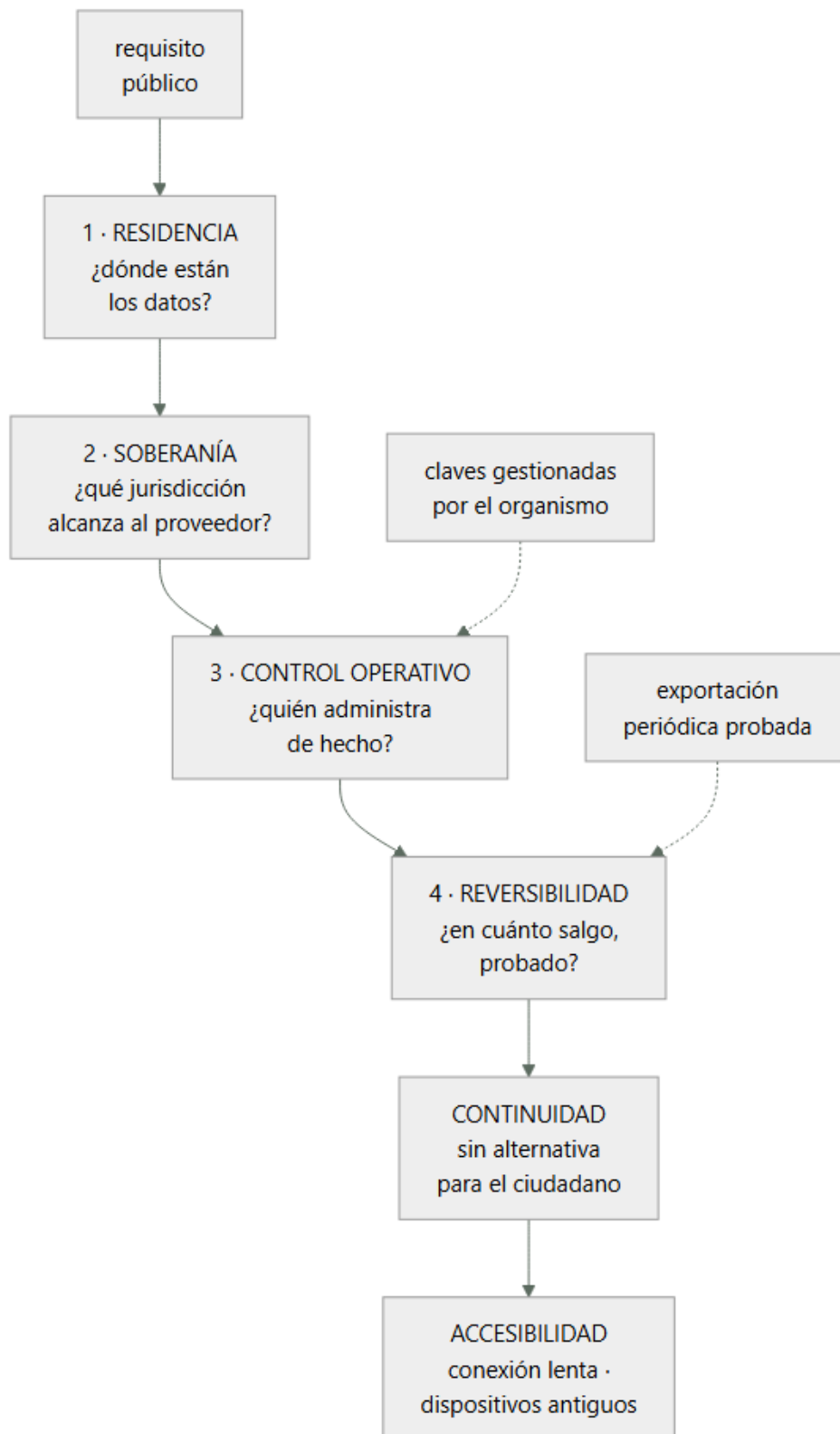
Concepto	Comprensión verificable
residencia de datos	Dónde están almacenados físicamente. Necesaria y muy insuficiente.
soberanía	Bajo qué jurisdicción y control quedan los datos y quién puede ser obligado a entregarlos.
control operativo	Quién puede acceder, administrar y detener el servicio en la práctica.
reversibilidad	Capacidad demostrada de recuperar datos y funciones y llevarlas a otro sitio en un plazo conocido.
servicio esencial	Aquel para el que el ciudadano no tiene alternativa. Cambia el cálculo de disponibilidad.
accesibilidad	Que el servicio sea usable por toda la población, incluidas conexiones lentas y capacidades diversas.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 y las tres capas de la soberanía

El encargo. La plataforma de trámites de una administración: identidad ciudadana, presentación de solicitudes, notificaciones con efectos legales, pago de tasas y consulta de expedientes. Servicio esencial, con obligación de continuidad y con requisitos de soberanía.

CIFRAS DE PARTIDA

ciudadanos con identidad digital	6,2 M
trámites/año	19 M
picos	fin de plazos fiscales: ×22 en 48 h
notificaciones con efecto legal/año	4,1 M
organismos que consumen datos	41
obligación de conservación	hasta 30 años
y disponibilidad exigida	99,9 % en horario, con plazos legales que no se suspenden

Y la restricción que manda:

EL CIUDADANO NO TIENE ALTERNATIVA

si el comercio cae, se compra en otro sitio clase 277
 si esto cae, el trámite no se puede hacer
 → y si cae el último día de plazo, se produce un daño
 jurídico

→ consecuencias

la disponibilidad no es una decisión de coste: es una
 obligación
 el sistema debe poder acreditar CUÁNDO estuvo caído,
 porque los plazos se amplían por ello
 y hace falta una vía alternativa: presencial o
 diferida, no como cortesía sino como parte del diseño

Y las tres capas de la soberanía, que se confunden constantemente:

1 RESIDENCIA

los datos están en centros situados en el territorio
 → necesaria y muy insuficiente
 → un dato en el territorio puede estar bajo control de
 una entidad sujeta a otra jurisdicción

2 SOBERANÍA JURÍDICA

¿qué legislación alcanza al proveedor y a su matriz?
 → ¿puede una autoridad extranjera obligarle a entregar
 datos alojados aquí?
 → esta pregunta es jurídica y determina decisiones
 técnicas

3 CONTROL OPERATIVO EFECTIVO

¿quién administra de hecho?
 → ¿el personal de soporte del proveedor puede acceder?
 → ¿desde dónde? ¿con qué registro? ¿con autorización de
 quién?
 → ¿quién tiene las claves de cifrado?

→ y la tercera es la que se puede verificar técnicamente
 → las tres juntas son lo que hace real una declaración de
 soberanía; una sola no basta

Y lo que de ahí se deriva:

CLAVES GESTIONADAS POR EL ORGANISMO

el cifrado con claves que el proveedor no puede usar
 → si el proveedor es obligado a entregar datos, entrega
 cifrado
 → coste: gestión de claves propia, con su recuperación
 y su ensayo clase 197

ACCESO DEL PROVEEDOR, CONTROLADO

cualquier acceso de soporte requiere aprobación explícita
 y queda registrado
 → y ese registro lo conserva el organismo, no el
 proveedor

Y REVERSIBILIDAD DEMOSTRADA

→ que es la siguiente sección

2. Reversibilidad: el requisito que nadie prueba

Todos los pliegos la exigen y casi nadie la comprueba. Y sin prueba, no existe.

LO QUE SUELE HABER

- una cláusula: «el adjudicatario entregará los datos al finalizar el contrato»
- sin formato definido
- sin plazo verificado
- sin haberlo hecho nunca ley 22

LO QUE HACE FALTA

- 1 EXPORTACIÓN PERIÓDICA REAL
 - datos y metadatos, en formato abierto y documentado, fuera del proveedor
 - mensual, no al final
- 2 QUE ESA EXPORTACIÓN SE PUEDA CARGAR EN OTRO SITIO
 - y probarlo, al menos una vez al año
 - cargar en un entorno alternativo y consultar
- 3 DEPENDENCIAS INVENTARIADAS Y CLASIFICADAS
 - qué usamos que solo existe en este proveedor
 - y qué coste tendría sustituirlo
- 4 Y UN PLAZO DE SALIDA ESTIMADO Y REVISADO
 - «podríamos operar en otro proveedor en N meses»
 - con N medido, no supuesto

- la reversibilidad no es no usar servicios gestionados
- es saber exactamente qué costaría dejar de usarlos clase 273

Y el equilibrio honesto:

RENUNCIAR A LOS SERVICIOS GESTIONADOS

- más coste, más operación, menos seguridad por defecto
- y con un equipo público, casi siempre peor resultado

USARLOS SIN NINGÚN INVENTARIO

- salida imposible en un plazo razonable

EL PUNTO INTERMEDIO

- gestionado para lo que tiene equivalente en otros proveedores
- cómputo, objetos, relacional, colas, identidad
- cuidado con lo que no lo tiene
- servicios propietarios sin equivalente, funciones muy específicas, modelos exclusivos
- y una capa fina propia en las fronteras que más costaría cambiar
- sin construir abstracciones grandes, que cuestan más de lo que ahorran clase 267

Y la continuidad, que aquí tiene una forma distinta:

NO BASTA CON OTRA REGIÓN DEL MISMO PROVEEDOR

- el escenario que preocupa al regulador no es una caída técnica
- es la indisponibilidad del proveedor por causa jurídica o comercial

- y de ahí sale una decisión cara: una capacidad mínima de servicio fuera del proveedor principal
- no todo el sistema: los trámites críticos y la consulta de expedientes
- y probada, no declarada

- es el único sector de este programa donde «multiproveedor» se justifica por sí mismo clase 273

3. Continuidad, plazos y accesibilidad

Lo que este sector exige y que en los demás no aparece.

LOS PLAZOS LEGALES NO SE SUSPENDEN SOLOS

- si el sistema cae el último día de un plazo, hay que

- acreditar la caída
- registro de disponibilidad con sellado de tiempo fiable y conservado
- y un procedimiento para ampliar plazos con base documental
- y esto convierte el registro de disponibilidad en un documento con efectos jurídicos
- que a su vez exige que sea inmutable y auditable
clase 278

LAS NOTIFICACIONES CON EFECTO LEGAL
 el momento de puesta a disposición importa
 la evidencia de entrega importa
 y el acceso del ciudadano a esa evidencia importa
 → idempotencia y no duplicación son requisitos legales aquí, no técnicos
 clase 210

Y el pico, que es de una forma peculiar:

- FIN DE PLAZO: x22 EN 48 HORAS
 no es estacional suave como el comercio
 → es una pared
- lo que funciona
 cola de presentación con acuse inmediato
 → «tu solicitud ha sido registrada a las 23:47:12» y el procesamiento va detrás
 → el registro es lo que tiene efecto jurídico, no el procesamiento
 vertido por prioridad: presentar antes que consultar
 clase 262
 y capacidad elástica reservada para esas ventanas
- y aquí el diseño asíncrono no es una preferencia técnica: es lo que permite cumplir el plazo del ciudadano
 ley 18

Y la accesibilidad, que en este sector es obligación legal:

- EL SERVICIO DEBE FUNCIONAR PARA TODA LA POBLACIÓN
 con conexiones lentas o intermitentes
 con dispositivos antiguos
 con lectores de pantalla y navegación por teclado
 con contraste suficiente
 y sin depender de que el usuario tenga un teléfono concreto
- consecuencias técnicas reales
 peso de página muy limitado
 funcionamiento sin ejecución de guiones para lo esencial
 formularios que se pueden guardar y retomar
 → porque una conexión que se cae a mitad de un trámite de 40 minutos es un problema de accesibilidad, no de experiencia
- y esto CONTRADICE muchas decisiones habituales de la ingeniería moderna
 → y en este sector, gana la accesibilidad

4. Las pruebas negativas del capstone

Varias de estas las ejecuta una intervención o una auditoría de la administración.

- DE SOBERANÍA Y CONTROL
- ¿algún dato sale del territorio? incluidos registros, métricas y copias
 - ¿el personal del proveedor puede acceder? ¿desde dónde? ¿queda registrado dónde?
 - ¿quién tiene las claves y puede el proveedor descifrar sin nosotros?
 - ¿el soporte técnico ve datos personales al depurar?
 - ¿los modelos o servicios de terceros procesan datos fuera?
clase 251

DE REVERSIBILIDAD

- ¿existe exportación mensual completa fuera del proveedor?
- ¿se ha CARGADO alguna vez en otro sitio? ¿cuándo?
- ¿qué servicios usamos sin equivalente en otro proveedor?
- ¿en cuántos meses podríamos operar fuera? ¿de dónde sale esa cifra?

DE CONTINUIDAD

- ¿los trámites críticos funcionan si el proveedor principal no está disponible?
- ¿se ha ensayado con carga?
- ¿el registro de disponibilidad es inmutable y sirve para acreditar una caída?
- ¿existe la vía alternativa y alguien la ha usado?

DE PICO Y PLAZOS

- simular ×22 durante 6 horas: ¿se registra todo?
- ¿el acuse de registro es inmediato e independiente del procesamiento?
- ¿una notificación puede duplicarse o perderse?
- ¿el sellado de tiempo es fiable y su desviación se vigila?

DE ACCESIBILIDAD

- ¿el trámite se completa con conexión de 1 Mbps y 400 ms de latencia?
- ¿se puede completar solo con teclado y con lector de pantalla?
- ¿un formulario largo se puede guardar y retomar?
- ¿funciona lo esencial sin ejecución de guiones?

El entregable del capstone:

- 1 el análisis de las tres capas de soberanía, con quién controla qué
- 2 el inventario de dependencias sin equivalente y su coste de sustitución
- 3 el plan de reversibilidad con la última prueba de carga externa
- 4 la capacidad mínima fuera del proveedor principal y su ensayo
- 5 el diseño del pico de fin de plazo, con acuse independiente
- 6 el registro de disponibilidad con valor probatorio
- 7 el informe de accesibilidad con pruebas reales
- 8 y el resultado de las pruebas negativas, con lo que falló

Y el cierre que enlaza con la clase siguiente: aquí la restricción es jurídica y el volumen llega a golpes. En el siguiente sector el volumen es continuo y enorme, el coste por byte decide la arquitectura y la calidad percibida se mide en el dispositivo del espectador. Medios y distribución global es la materia de la clase 281.

Ejemplo trabajado

El capstone resuelto. Lo que sigue es el análisis de soberanía que encontró tres fugas de datos fuera del territorio, la prueba de reversibilidad que nunca se había hecho, y el pico de fin de plazo con la cola de registro.

El análisis de las tres capas.

CAPA 1 · RESIDENCIA

declarada todos los datos en la región nacional
verificada barrido automático de recursos por región

hallazgos

- ✓ bases de datos, objetos y cómputo: en región
- ✗ REGISTROS de aplicación: enviados a un servicio de análisis alojado fuera
 - incluían identificadores de ciudadano en las trazas de error
- ✗ MÉTRICAS Y RASTREO: agregados fuera
 - sin datos personales, pero con patrones de uso

X COPIAS DE SEGURIDAD de un servicio secundario
replicadas a una región extranjera por una opción
por defecto ley 26

- tres fugas, ninguna intencionada, todas por valores por defecto o por integraciones antiguas
- y la declaración de residencia era falsa desde hacía años

CAPA 2 · SOBERANÍA JURÍDICA

análisis jurídico externo

- el proveedor tenía matriz sujeta a legislación extranjera con posibilidad de requerimiento
- conclusión: la residencia no basta

decisión técnica derivada

cifrado con claves gestionadas por el organismo, en
módulo propio clase 197

- el proveedor no puede descifrar
- coste: gestión, custodia y ensayo de recuperación de claves; 0,5 personas

CAPA 3 · CONTROL OPERATIVO

prueba: se pidió al proveedor la lista de accesos de su personal a los entornos del organismo en 12 meses

respuesta

accesos de soporte	41
con aprobación previa del organismo	6
registrados por el organismo	0

- los otros 35 se conocieron por esta petición

corrección

acceso del proveedor solo con aprobación explícita,
sesión grabada y registro conservado por el organismo

- contractual y técnicamente

accesos en los 12 meses siguientes	9
todos aprobados y registrados	9

La prueba de reversibilidad.

el pliego exigía reversibilidad

la cláusula existía desde hacía 6 años

y nunca se había ejecutado ley 22

el ejercicio

exportar todo y cargarlo en un entorno de otro
proveedor, y consultar

lo que pasó

exportación de bases relacionales	3 días
-----------------------------------	--------

exportación de objetos (41 TB)	9 días
--------------------------------	--------

- y con coste de salida de 3.900 USD

metadatos del catálogo	NO EXPORTABLE
------------------------	---------------

en formato abierto

configuración de identidad y permisos	NO EXPORTABLE
---------------------------------------	---------------

definiciones de flujos de trabajo	NO EXPORTABLE
-----------------------------------	---------------

(formato propietario)

carga en el entorno alternativo

datos	correcta
permisos	rehechos a mano, 4 semanas

flujos de trabajo	reescritos, 11 semanas
-------------------	---------------------------

tiempo total del ejercicio parcial	17 semanas
------------------------------------	------------

estimación de salida completa	14 meses
-------------------------------	----------

Y las decisiones que salieron:

los flujos de trabajo se reescribieron en un formato
abierto y ejecutable en cualquier proveedor

- coste: 9 semanas
- y la estimación de salida bajó de 14 a 5 meses

la configuración de identidad y permisos pasó a infraestructura como código, exportable clase 128

el catálogo de metadatos se replicó a un formato abierto mensualmente

- y se estableció un ejercicio de carga externa ANUAL
- tercer año: 6 semanas, con el 92 % automatizado

- y lo importante: la cifra de salida pasó a ser un dato medido que se revisa, no una cláusula

El pico de fin de plazo.

el escenario
último día de plazo fiscal
×22 sobre un día normal, concentrado entre las 18:00 y las 23:59

diseño anterior
el trámite se procesaba de forma síncrona
→ a las 21:40 el sistema se saturó
→ 41 minutos sin servicio
→ 19.000 ciudadanos sin poder presentar
→ ampliación de plazo por resolución, y 340 reclamaciones

diseño nuevo
presentación → validación mínima → REGISTRO con acuse y sello de tiempo → cola → procesamiento
→ el acuse llega en menos de 2 segundos
→ el procesamiento puede tardar horas y no afecta al plazo
y vertido por prioridad: presentar > consultar > descargar histórico

Y el resultado del ensayo con carga sintética:

carga simulada	×22 durante 6 horas
presentaciones	414.000
acuses emitidos	414.000
acuses con sello de tiempo válido	414.000
pérdidas	0
latencia del acuse, percentil 99	1,7 s
retraso máximo del procesamiento	3 h 40 min
consultas de expediente vertidas durante el pico	31 %

- y el ciudadano que consultaba veía «servicio de consulta no disponible por alta demanda; la presentación funciona con normalidad»
- mensaje acordado con el área jurídica antes

Y el día real, ese año:

pico real	×19
tiempo sin servicio de presentación	0
acuses emitidos	361.000
reclamaciones	4
ampliaciones de plazo por caída	0

- y el retraso del procesamiento 2 h 10
- sin efecto jurídico, porque el registro ya constaba

La accesibilidad, medida.

prueba con conexión de 1 Mbps y 400 ms de latencia	
antes trámite completo	no completable (el formulario tardaba 74 s en cargar)
después	completable en

4 min 20

peso de la página del trámite
antes 3,9 MB
después 412 KB

completable solo con teclado y lector de pantalla
antes no
después sí

formulario largo guardable y retomable
antes no
después sí
→ y esto solo redujo los abandonos del 34 % al 11 %

y lo que costó
se retiraron 3 bibliotecas del cliente y se sustituyó
la validación en el navegador por validación en el
servidor con recarga parcial
→ decisión que en otro sector habría sido un retroceso
→ aquí es cumplir la obligación legal

Las cifras finales del capstone.

datos fuera del territorio	3 fugas	0
accesos del proveedor registrados por el organismo	0 de 41	9 de 9
claves controladas por el organismo	no	sí
reversibilidad probada	no	sí, anual
salida estimada	14 meses	5 meses
servicios sin equivalente	7	2
caída en fin de plazo	41 min	0
ampliaciones de plazo por caída	1	0
acuse independiente del procesamiento	no	sí
trámite completable a 1 Mbps	no	sí
peso de la página	3,9 MB	412 KB
abandono de formularios largos	34 %	11 %

La lección que este capstone deja: la declaración de residencia nacional era falsa desde hacía años por tres caminos que nadie había elegido —registros con identificadores a un servicio externo, métricas fuera y copias replicadas por una opción por defecto—. Y la cláusula de reversibilidad llevaba seis años en el pliego sin ejecutarse nunca: al probarla, la salida real resultó ser de catorce meses, y bajó a cinco solo tras reescribir lo que estaba en formatos propietarios.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/280-capstone-sector-publico-soberania-y-continuidad/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 public-sector-capstone 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 public-sector-capstone 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 declara residencia nacional y hay datos fuera del territorio	Registros, métricas y copias salen por integraciones antiguas y opciones por defecto	Barre automáticamente todos los recursos por región, incluidos registros, métricas y copias, y controla los valores por defecto de replicación.
La residencia se cumple y la soberanía sigue sin estar garantizada	Se confunde dónde están los datos con quién puede ser obligado a entregarlos y quién los administra	Analiza las tres capas: residencia, jurisdicción del proveedor y control operativo; cifra con claves que el proveedor no pueda usar.
La cláusula de reversibilidad existe y nadie sabe cuánto costaría salir	Nunca se ha ejecutado una exportación ni una carga en otro sitio	Exporta mensualmente en formato abierto y carga en un entorno alternativo al menos una vez al año; la cifra de salida debe ser medida.
El sistema cae en fin de plazo y hay que ampliar plazos por resolución	El trámite se procesa de forma síncrona y la pared de demanda lo satura	Separa registro y procesamiento: acuse inmediato con sello de tiempo, cola detrás, y vertido por prioridad de presentar sobre consultar.
El servicio no se puede usar con conexión lenta o con lector de pantalla	Se aplicaron decisiones habituales de experiencia sin la restricción legal de accesibilidad	Limita el peso, haz que lo esencial funcione sin guiones y permite guardar y retomar formularios largos; aquí la accesibilidad gana.
No se puede acreditar cuándo estuvo caído el servicio	El registro de disponibilidad no es inmutable ni está sellado	Trata ese registro como documento con efectos jurídicos: inmutable, con sello de tiempo fiable y conservado por el organismo.

■ 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 residencia, soberanía jurídica y control operativo?
2. ¿Qué hace real un plan de reversibilidad y qué lo convierte en discurso?
3. ¿Por qué en este sector se justifica una capacidad fuera del proveedor principal?
4. ¿Cómo se diseña el pico de fin de plazo sin ampliar plazos por caída?
5. ¿Qué decisiones técnicas impone la accesibilidad y por qué aquí prevalecen?

Referencias

- ENISA (2024). European Cybersecurity Certification Scheme for Cloud Services. <<https://www.enisa.europa.eu/topics/certification>>
- Comisión Europea (2023). Data Act — cambio de proveedor y portabilidad en la nube. <<https://digital-strategy.ec.europa.eu/en/policies/data-act>>
- W3C (2023). Web Content Accessibility Guidelines (WCAG) 2.2. <<https://www.w3.org/TR/WCAG22/>>
- AWS (2024). Digital sovereignty and data residency guidance. <<https://aws.amazon.com/compliance/data-protection/>>
- Microsoft (2024). Azure sovereign cloud and data residency. <<https://learn.microsoft.com/azure/cloud-adoption-framework/scenarios/sovereignty/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 23 en PDF · Manual integral

Anterior	Índice	Siguiente
← 279 · Capstone salud: privacidad e interoperabilidad	Parte 23 · Programa	281 · Capstone media: streaming y distribución global →

Evaluación — 280 Capstone sector público: soberanía y continuidad

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 public-sector-capstone 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.

281 — Capstone media: streaming y distribución global

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

Propósito

Capstone de medios: streaming y distribución global. La clase da el encargo y la restricción que manda —el coste por byte servido y la calidad percibida en el dispositivo del espectador deciden todo lo demás—, por qué aquí el borde no es una optimización sino la arquitectura, y las pruebas negativas de un sector donde el pico se produce por un evento que se anuncia con meses de antelación.

Resultados de aprendizaje

Al finalizar podrás:

1. Explicar por qué el coste por byte reordena todas las decisiones.
2. Diseñar la cadena de vídeo desde la ingesta hasta el reproductor.
3. Medir calidad percibida y no disponibilidad de servidor.
4. Planificar un evento en directo con capacidad y con degradación.
5. Verificar el diseño con las pruebas negativas del sector.

Conceptos centrales

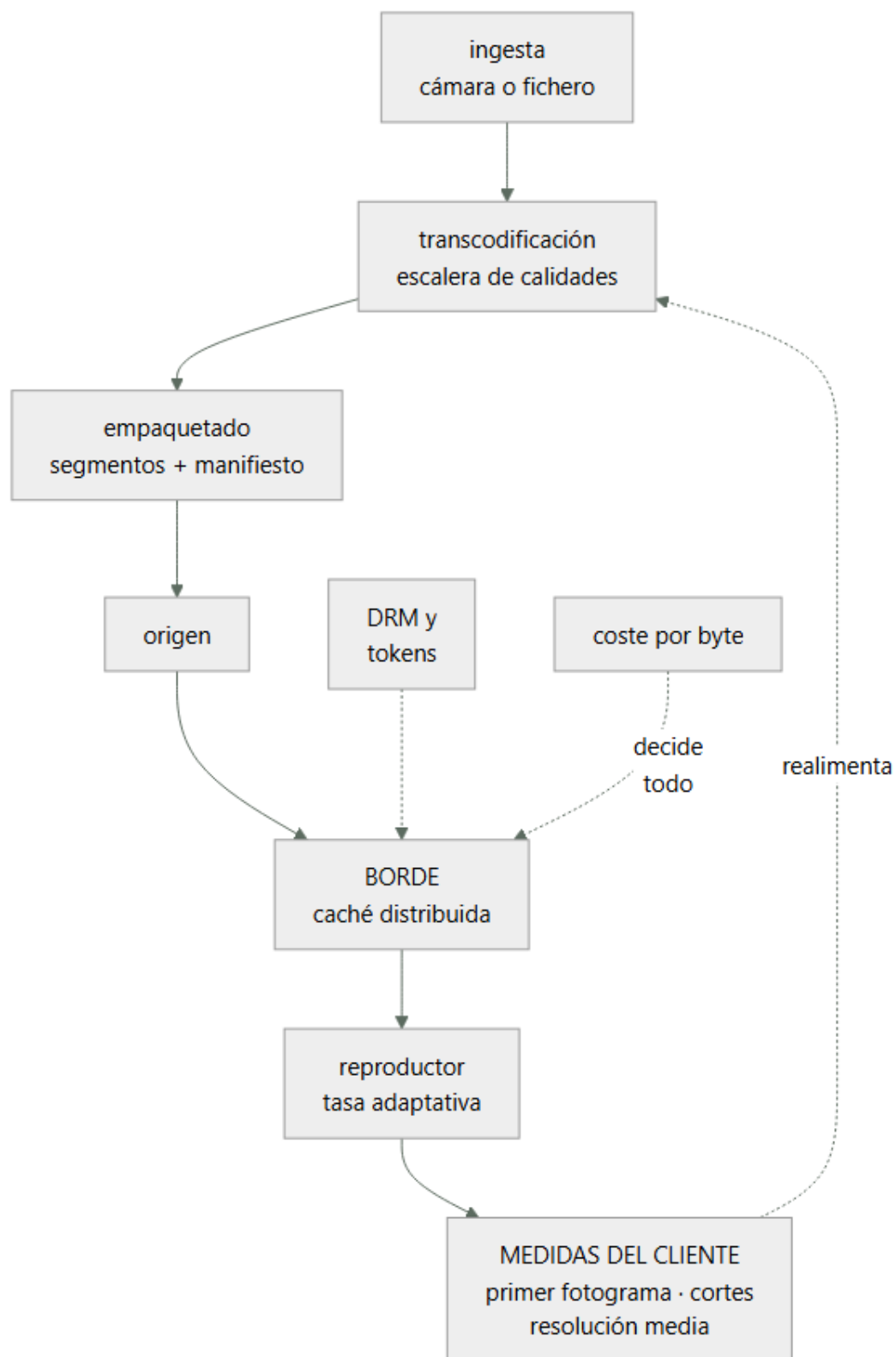
Concepto	Comprensión verificable
coste por byte servido	La unidad económica del sector. Domina sobre cómputo y almacenamiento.
tasa de acierto en el borde	Porcentaje de bytes servidos desde la caché. Un punto de mejora vale mucho dinero.
calidad percibida	Lo que experimenta el espectador: tiempo hasta el primer fotograma, cortes y resolución media.
tasa de bits adaptativa	El reproductor elige la calidad según la red. Traslada la decisión al dispositivo.
directo frente a bajo demanda	El directo no admite reproceso ni reintento largo: lo que se pierde, se pierde.
purga de caché	Invalidar contenido distribuido. Cara, lenta y necesaria; se diseña desde el principio.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 y la unidad que decide

El encargo. Una plataforma de vídeo: catálogo bajo demanda, canales en directo, un evento anual con audiencia masiva, y distribución en 40 países con derechos por territorio.

CIFRAS DE PARTIDA
horas de catálogo

41.000

espectadores simultáneos, día normal	180.000
espectadores simultáneos, evento anual	2,4 M
tráfico servido/mes	14 PB
países con derechos distintos	40
y el reparto del coste	
distribución de bytes	71 %
transcodificación	14 %
almacenamiento	9 %
resto	6 %

Y la restricción que manda:

EL COSTE POR BYTE SERVIDO MANDA SOBRE EL RESTO

el 71 % del coste es mover bytes
→ y de ahí sale que la arquitectura sea la red de distribución, y no los servicios de cómputo

y la aritmética que lo hace evidente

14 PB/mes
1 punto de tasa de acierto en el borde = 140 TB que no salen del origen
→ y cada punto de acierto vale decenas de miles al mes

→ por eso decisiones que en otros sectores son detalles aquí son estructurales
tamaño de segmento
cabeceras de caché
nombres de fichero y su variabilidad
y la escalera de calidades

Y la segunda restricción, que es la que define el producto:

LO QUE IMPORTA ES LA CALIDAD PERCIBIDA, NO LA DISPONIBILIDAD DEL SERVIDOR

un servidor al 100 % con vídeo que se corta es un fallo
→ y ninguna métrica de servidor lo detecta

las medidas que importan, todas en el CLIENTE

tiempo hasta el primer fotograma
porcentaje de sesiones con al menos un corte
tiempo total en pausa por almacenamiento intermedio
resolución media servida
y errores de inicio de reproducción

→ y este es el mismo principio de la clase 268 llevado al extremo: si no se mide en el dispositivo, no se mide

2. La cadena de vídeo

Del origen al ojo, con las decisiones de cada tramo.

INGESTA

bajo demanda: ficheros maestros a un almacén
directo: flujo redundante desde dos rutas distintas
→ y en directo, la redundancia de ingesta es lo único que salva un evento: si se pierde ahí, no hay reproceso

TRANSCODIFICACIÓN

una escalera de calidades por título
decisión escalera fija frente a escalera por contenido
→ una animación y un partido de fútbol no necesitan la misma tasa de bits para verse igual
→ la escalera por contenido reduce bytes entre un 15 % y un 30 % sin pérdida percibida
empeora más cómputo de análisis y más complejidad
y en directo no hay tiempo: escalera fija por tipo de canal

EMPAQUETADO

segmentos y manifiesto
decisión tamaño de segmento
segmentos cortos → menos retardo en directo, más peticiones, peor acierto de caché
segmentos largos → mejor acierto, más retardo

→ y esta decisión sola mueve el coste y la experiencia a la vez

DISTRIBUCIÓN

el borde es la arquitectura
claves de caché estables: sin parámetros variables en la ruta
→ un identificador de sesión en la ruta destruye el acierto de caché y multiplica la factura
y jerarquía de caché: borde → intermedio → origen
→ el intermedio protege el origen en un pico

REPRODUCTOR

tasa adaptativa: el cliente decide
→ y ahí está la mayor parte de la calidad percibida
buffer inicial frente a tiempo hasta el primer fotograma
→ más buffer, menos cortes, arranque más lento
y comunicación de medidas al servicio de análisis

DERECHOS Y ACCESO

tokens firmados con caducidad corta y con territorio
cifrado de contenido con licencias por dispositivo
→ y el bloqueo por territorio se comprueba en el borde, no en el origen

Y la purga, que hay que diseñar antes de necesitarla:

RETIRAR UN CONTENIDO DE 40 PAÍSES

por caducidad de derechos, por orden judicial o por error
→ y hacerlo en minutos, no en horas

lo que lo hace posible
versionado en la ruta en vez de purga masiva
tokens de acceso con caducidad corta
→ así retirar el permiso vale casi tanto como purgar y purga selectiva para lo urgente

→ y probarlo: la purga es una de esas capacidades que nadie ha ejercitado hasta que hace falta ley 22

3. El evento en directo

El sector tiene un pico peculiar: se conoce con meses de antelación y no admite fallo.

LO QUE LO DIFERENCIA

la fecha se sabe clase 262
la audiencia se puede estimar con histórico
y NO HAY SEGUNDA OPORTUNIDAD: el partido ocurre una vez

y las formas de fallar, en orden de gravedad

- | | | |
|---|-------------------------------|--------------------|
| 1 | perder la ingesta | → no hay contenido |
| 2 | saturar el origen | → todos afectados |
| 3 | agotar la capacidad del borde | → región afectada |
| 4 | fallar la autorización | → nadie entra |
| 5 | degradar la calidad | → aceptable |

Y el plan, que es sobre todo de reserva y de ensayo:

CAPACIDAD RESERVADA CON ANTELACIÓN

el borde se contrata; no se autoescala solo
→ y hay que avisar al proveedor de distribución con semanas

CALENTAMIENTO DE CACHÉ

el contenido estático y los manifiestos, precargados

AUTORIZACIÓN, QUE ES EL CUELLO REAL

2,4 M de espectadores entran en 4 minutos
→ el pico no es de vídeo: es de LOGIN y de emisión de tokens
→ y ahí se cae la mayoría de los eventos
mitigación: tokens emitidos con antelación, ventana de entrada escalonada y capacidad específica para identidad clase 209

DEGRADACIÓN DECIDIDA ANTES

- 1 se apagan recomendaciones y catálogo enriquecido
 - 2 se limita la calidad máxima
 - 3 se sirve una calidad única si hace falta
 - 4 y se vierte el acceso a bajo demanda
- todo antes que perder el directo clase 262

Y ENSAYO CON CARGA SINTÉTICA

- al menos dos, uno a 3 semanas y otro a 1
clase 261

4. Las pruebas negativas del capstone

Lo que hay que ejecutar para saber si el diseño aguanta.

DE COSTE Y CACHE

- ¿cuál es la tasa de acierto en el borde por tipo de contenido?
- ¿hay algún parámetro variable en las rutas que rompa la clave de caché?
- ¿cuánto tráfico llega al origen y por qué?
- ¿cuál es el coste por hora vista y su tendencia?

DE CALIDAD PERCIBIDA

- ¿qué porcentaje de sesiones tiene al menos un corte?
- ¿tiempo hasta el primer fotograma, percentil 95, por país?
- ¿resolución media servida por tipo de red?
- simular una red de 3 Mbps con pérdidas: ¿qué hace el reproductor?

DE FALLO

- caer el origen durante 10 minutos: ¿siguen las sesiones activas?
- caer una región de borde: ¿se redistribuye o falla?
- perder una de las dos rutas de ingesta en directo: ¿se nota?
- el servicio de licencias no responde: ¿qué ve el espectador?

DE DERECHOS Y ACCESO

- un espectador fuera de territorio con un enlace directo: ¿accede?
- un token caducado reutilizado: ¿funciona?
- retirar un título de 40 países: ¿cuánto tarda y se verifica?
- ¿un enlace de segmento se puede compartir y consumir sin autorización?

DEL EVENTO

- ensayo con carga sintética al 120 % de lo estimado
- 2,4 M de autorizaciones en 4 minutos: ¿aguanta identidad?
- ¿la degradación por prioridad se ejecuta sola o requiere una persona?
- ¿quién puede activar la calidad única y en cuánto tiempo?

El entregable del capstone:

- 1 la cadena completa con las decisiones de cada tramo
- 2 la política de caché y su tasa de acierto objetivo
- 3 las métricas de calidad percibida y sus objetivos
clase 268
- 4 el diseño de derechos por territorio y de retirada de contenido
- 5 el plan del evento: capacidad, autorización, degradación y ensayos
- 6 el coste por hora vista, base y evento
- 7 y el resultado de las pruebas negativas, con lo que falló

Y el cierre que enlaza con la clase siguiente: aquí la red es la arquitectura y el espectador siempre está conectado. En el siguiente sector el dispositivo está lejos, la conexión falla durante días y el sistema tiene que seguir funcionando sin

ella. Industria, dispositivos y operación desconectada es la materia de la clase 282.

Ejemplo trabajado

El capstone resuelto. Lo que sigue es el parámetro en la ruta que costaba 61.000 USD al mes, el evento que se cayó por identidad y no por vídeo, y las cifras de calidad percibida antes y después.

El parámetro que rompía la caché.

situación	
tasa de acierto en el borde	71 %
tráfico al origen	4,1 PB/mes
coste del origen y su salida	81.000 USD/mes

la investigación empezó por una pregunta simple
«¿por qué el 29 % de los bytes va al origen si el catálogo es estático?»

lo que se encontró, agrupando peticiones al origen

por tipo de recurso	
segmentos de vídeo	88 %
manifiestos	9 %
imágenes y otros	3 %

y dentro de los segmentos, por patrón de ruta
→ el reproductor añadía un identificador de sesión
como parámetro de consulta
→ y la configuración del borde incluía los parámetros
en la clave de caché

→ cada espectador pedía SU copia de cada segmento
→ la caché servía solo lo que un mismo espectador repetía

Y la corrección:

el identificador de sesión pasó a una cabecera que no
forma parte de la clave
y la clave de caché se normalizó explícitamente

resultado a las 2 semanas	
tasa de acierto	71 % → 96,4 %
tráfico al origen	4,1 PB → 0,5 PB/mes
coste del origen y su salida	81.000 → 20.000 USD/mes
ahorro	61.000 USD/mes
esfuerzo	9 horas

Y la lectura:

el parámetro llevaba 3 años
y nadie lo había mirado porque el sistema FUNCIONABA
→ la calidad percibida era buena; solo el coste era malo
→ y sin unidad económica –coste por hora vista– nada lo
señalaba
clase 270

coste por hora vista	
antes	0,0141 USD
después	0,0038 USD

El evento que se cayó por identidad.

el evento del año anterior	
espectadores esperados	1,9 M
capacidad de borde contratada	sí, al 130 %
ensayo de carga de vídeo	sí, superado

lo que pasó

20:57	se abre la entrada
20:58	340.000 peticiones de autorización en 60 segundos
20:58	el servicio de identidad se satura
20:59	los reintentos del cliente multiplican la carga por 3 clase 201
21:02	el 61 % de los espectadores no puede entrar
21:14	se restablece, con el evento empezado

vídeo servido durante toda la caída perfecto
→ la infraestructura de vídeo estaba ociosa

→ el cuello no era el vídeo: era la puerta

Y el rediseño para el año siguiente:

- 1 TOKENS EMITIDOS CON ANTELACIÓN
quien tenía la aplicación abierta recibía su token desde 2 horas antes, renovado en segundo plano
→ el 71 % de la audiencia entró sin pedir autorización en el minuto crítico
- 2 ENTRADA ESCALONADA
la sala se abre por lotes de usuarios, con 15 segundos de separación, indicado en la interfaz
→ «tu acceso se activa en 12 s»
- 3 CAPACIDAD ESPECÍFICA PARA IDENTIDAD
dimensionada para el pico de entrada, no para el uso medio clase 262
- 4 REINTENTOS DEL CLIENTE CON RETROCESO Y JITTER
→ y esto solo bajó la carga de pico un 43 %
- 5 Y ENSAYO ESPECÍFICO DE AUTORIZACIÓN
2,4 M de autorizaciones en 4 minutos, sintéticas

Y el evento del año siguiente:

espectadores simultáneos, pico	2,41 M
autorizaciones en el minuto de apertura	418.000
fallos de autorización	0,03 %
tiempo hasta el primer fotograma, p95	2,1 s
sesiones con al menos un corte	1,7 %
degradación activada	nivel 1
(recomendaciones y catálogo enriquecido apagados, 19 minutos)	
incidentes	0

y el coste del evento

distribución	214.000 USD
capacidad reservada de identidad	9.000 USD

→ y esos 9.000 eran lo que faltaba el año anterior

La escalera por contenido.

prueba con 200 títulos, comparando escalera fija y escalera calculada por contenido

	bytes	calidad percibida (evaluación ciega, 40 personas)
escalera fija	100 %	referencia
por contenido	78 %	indistinguible en 191 de 200 peor en 4 mejor en 5

→ -22 % de bytes en el catálogo
→ y de los 4 peores, 3 eran contenido con mucho grano y se corrigieron con un mínimo por perfil

efecto anual

bytes servidos	14 PB → 11,2 PB/mes
ahorro de distribución	~47.000 USD/mes
coste de transcodificación	+8.000 USD/mes
neto	~39.000 USD/mes

La retirada de contenido, probada.

escenario caducan los derechos de una serie en 12 países a medianoche

primer intento, con purga masiva

purga de 41.000 objetos en 12 territorios
tiempo hasta que dejó de servirse 3 h 20 min
→ y durante ese tiempo se sirvió contenido sin derechos
→ riesgo contractual real

rediseño

el acceso pasa por token firmado con territorio y con
caducidad de 5 minutos
→ retirar los derechos invalida la emisión de tokens
nuevos
→ el contenido deja de ser accesible en 5 minutos
→ la purga sigue existiendo, para limpiar, sin urgencia

segundo ensayo

tiempo hasta dejar de servirse 4 min 40 s
→ y verificado desde 12 puntos de medición

Las cifras finales del capstone.

tasa de acierto en el borde	71 %	96,4 %
tráfico al origen	4,1 PB/mes	0,5 PB/mes
bytes servidos	14 PB	11,2 PB
coste por hora vista	0,0141	0,0031
tiempo hasta el primer fotograma p95	4,8 s	1,9 s
sesiones con al menos un corte	6,4 %	1,7 %
resolución media servida	720p	1080p
fallos de autorización en el evento	61 %	0,03 %
incidentes en el evento	1	0
tiempo de retirada de contenido	3 h 20 min	4 min 40 s

Y la observación final del equipo:

de las tres mejoras grandes
la caché 9 horas de trabajo
la escalera por contenido 6 semanas
la entrada escalonada 4 semanas

→ la de 9 horas ahorró más que las otras dos juntas
→ y llevaba tres años delante de todo el mundo

La lección que este capstone deja: un identificador de sesión en la ruta de los segmentos hacía que cada espectador pidiera su propia copia de cada trozo de vídeo, y corregirlo costó nueve horas y 61.000 USD al mes; llevaba tres años sin que nadie lo notara porque la calidad percibida era buena y solo el coste era malo. Y el evento que falló no falló por vídeo: la infraestructura de vídeo estuvo ociosa mientras el 61 % de los espectadores no podía cruzar la puerta.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/281-capstone-media-streaming-y-distribucion-global/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 media-capstone 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 media-capstone 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 tasa de acierto en el borde es baja y el origen recibe mucho tráfico	Hay parámetros variables en la ruta que forman parte de la clave de caché	Normaliza explícitamente la clave de caché y mueve identificadores de sesión a cabeceras que no la compongan.
Los servidores están sanos y los espectadores se quejan de cortes	Se mide disponibilidad de servidor y no calidad percibida	Instrumenta el reproductor: tiempo hasta el primer fotograma, sesiones con corte, tiempo en pausa y resolución media, por país y tipo de red.
Un evento en directo falla al abrirse aunque el vídeo aguante	El pico real es de autorización y no de vídeo, y los reintentos lo multiplican	Emite tokens con antelación, escalona la entrada, dimensiona identidad para el pico y aplica retroceso con dispersión en el cliente.
Retirar un contenido por derechos tarda horas	Se depende de la purga masiva de la caché	Controla el acceso con tokens de caducidad corta por territorio: revocar la emisión corta el acceso en minutos y la purga limpia sin urgencia.
El coste sube con la audiencia y nadie sabe si es razonable	No hay unidad económica: se mira la factura y no el coste por hora vista	Define coste por hora vista y sigue su tendencia; es lo que hace visible un desperdicio que funciona bien.
El directo se pierde y no hay forma de recuperarlo	La ingesta no era redundante y el directo no admite reproceso	Duplica la ingesta por rutas distintas y ensáyalo cortando una; lo que se pierde en directo no se recupera.

■ 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 coste por byte reordena las decisiones de este sector?
2. ¿Qué decisiones de la cadena afectan a la vez al coste y a la experiencia?
3. ¿Qué métricas de calidad percibida hay que medir y dónde?
4. ¿Cuál es el cuello de botella real de un evento en directo masivo?
5. ¿Cómo se retira un contenido de muchos territorios en minutos?

Referencias

- Bitmovin (2024). Video Developer Report — métricas de calidad de experiencia.
<<https://bitmovin.com/video-developer-report/>>
- Netflix Technology Blog (2018). Dynamic optimizer: per-title and per-shot encoding.
<<https://netflixtechblog.com/dynamic-optimizer-a-perceptual-video-encoding-optimization-framework-e19f1e3a277f>>
- AWS (2024). Media and entertainment reference architectures. <<https://aws.amazon.com/media/resources/>>
- Google Cloud (2024). Live streaming and video-on-demand architectures.
<<https://cloud.google.com/architecture/live-streaming-and-video-on-demand>>
- CTA (2020). CTA-5004: Common Media Client Data — medición desde el reproductor.
<<https://cdn.cta.tech/cta/media/media/resources/standards/pdfs/cta-5004-final.pdf>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 23 en PDF · Manual integral

Anterior	Índice	Siguiente
← 280 · Capstone sector público: soberanía y continuidad	Parte 23 · Programa	282 · Capstone industria: IoT, edge y operación desconectada →

Evaluación — 281 Capstone media: streaming y distribución global

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 media-capstone 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.

282 — Capstone industria: IoT, edge y operación desconectada

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

Propósito

Capstone industrial: dispositivos, borde y operación desconectada. La clase da el encargo y la restricción que manda —la conexión no es fiable y el dispositivo tiene que seguir haciendo su trabajo sin ella—, lo que eso obliga en decisión local, sincronización y actualización remota, y las pruebas negativas de un sector donde una actualización mal hecha deja mil equipos inaccesibles.

Resultados de aprendizaje

Al finalizar podrás:

1. Decidir qué se resuelve en el dispositivo y qué en la nube.
2. Diseñar sincronización tolerante a desconexiones largas.
3. Actualizar flota remota sin dejar equipos inaccesibles.
4. Gestionar identidad y ciclo de vida de dispositivos a escala.
5. Verificar el diseño con las pruebas negativas del sector.

Conceptos centrales

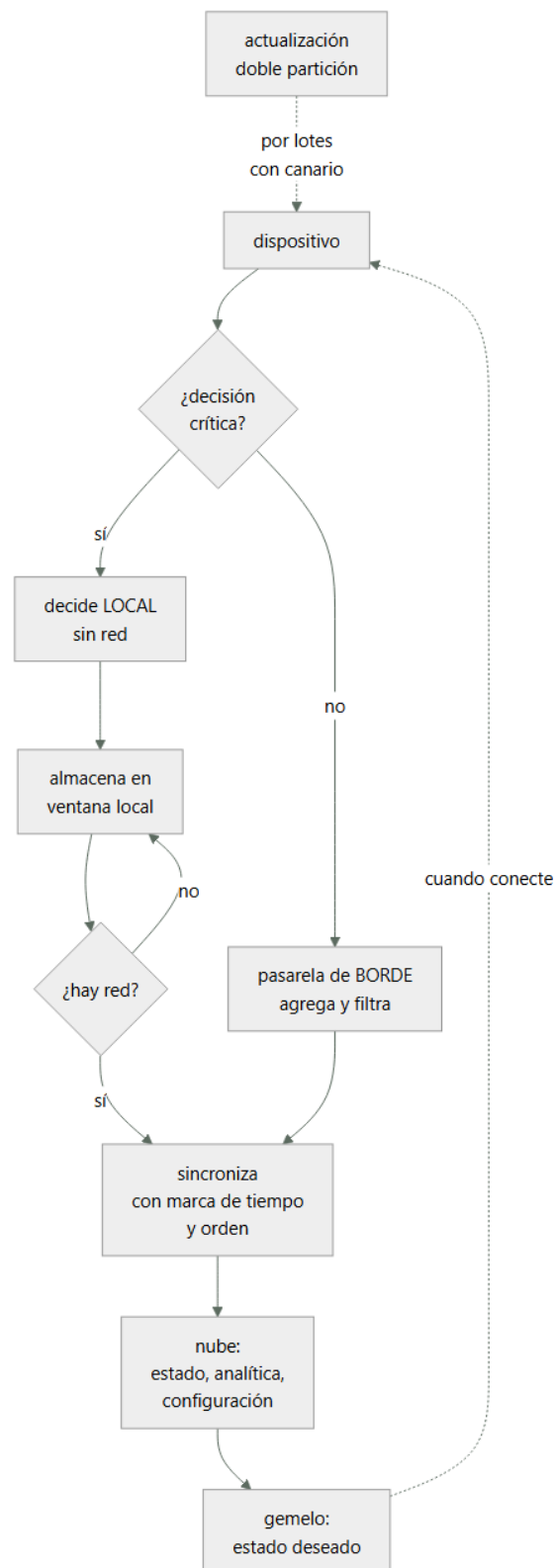
Concepto	Comprensión verificable
operación desconectada	El dispositivo cumple su función sin red, durante horas o días, y reconcilia al volver.
gemelo de dispositivo	Estado deseado y estado informado de cada equipo, sincronizados cuando hay conexión.
actualización con doble partición	Se escribe en la partición inactiva y se arranca en ella; si falla, vuelve sola a la anterior.
identidad de dispositivo	Credencial única por equipo, idealmente en hardware, revocable individualmente.
borde	Capacidad de cómputo cercana a los dispositivos que agrega, filtra y decide con baja latencia.
ventana de datos	Cuánto histórico guarda el dispositivo antes de sobrescribir. Define lo que se puede perder.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 y la restricción que manda

El encargo. Una plataforma para 41.000 equipos industriales instalados en plantas y en campo: sensores de proceso, controladores de línea y equipos móviles. Deben operar, informar y actualizarse.

CIFRAS DE PARTIDA
dispositivos

41.000

emplazamientos	380
conectividad	
fija fiable	42 %
fija intermitente	39 %
móvil o satélite	19 %
mediciones/segundo, agregadas	1,9 M
vida útil del equipo	8-12 años
y el requisito de negocio	
una línea parada cuesta	14.000 USD/hora

Y la restricción que manda:

LA CONEXIÓN NO ES FIABLE Y EL TRABAJO NO PUEDE ESPERAR
un controlador que detiene la línea porque no llega la
respuesta de la nube es un fallo de diseño, no de red

- y de ahí sale la primera regla del sector
TODA DECISIÓN CON CONSECUENCIA FÍSICA SE TOMA LOCAL
 - parar una máquina, abrir una válvula, disparar una alarma de seguridad
 - la nube observa, ajusta parámetros y aprende; no decide en el instante
- y la segunda
EL DISPOSITIVO ES UN SISTEMA COMPLETO, no un sensor
 - tiene almacenamiento, lógica, estado y actualizaciones
 - y una vida útil de una década, que es más de lo que dura cualquier decisión de arquitectura de nube

Y la consecuencia sobre el ciclo de vida:

- 8-12 AÑOS DE VIDA ÚTIL CAMBIAN LAS REGLAS
- el protocolo tiene que sobrevivir a varias generaciones de plataforma
 - la credencial tiene que poder rotarse sin visita física
 - el formato de datos tiene que ser compatible hacia atrás y hacia delante clase 106
 - y habrá dispositivos con versiones muy distintas conviviendo, siempre
- diseñar suponiendo que toda la flota está actualizada es el error estructural del sector
 - y en la práctica, un 8-15 % de la flota está siempre varias versiones por detrás

2. Reparto de responsabilidad y sincronización

Qué corre dónde, y qué pasa cuando vuelve la conexión.

EN EL DISPOSITIVO

- control en tiempo real y seguridad
- almacenamiento en ventana: horas o días de datos
- reglas locales de alarma
- y capacidad de operar desconectado el tiempo objetivo

EN LA PASARELA DE BORDE, por emplazamiento

- agregación y reducción: de 1.900 mediciones/s a 40
- y esto decide el coste de todo lo demás
- reglas que necesitan varias fuentes
- almacenamiento intermedio cuando cae el enlace
- y actualización local de la flota del emplazamiento

EN LA NUBE

- estado consolidado y gemelos
- analítica, comparación entre plantas y modelos
- configuración y campañas de actualización
- alertas de negocio y mantenimiento

- y la regla que ordena el reparto
si la respuesta debe llegar en menos de lo que tarda un viaje de ida y vuelta poco fiable, va abajo

Y la sincronización, que es el problema técnico central:

LO QUE HAY QUE RESOLVER

el dispositivo estuvo 3 días sin red
→ tiene 3 días de datos y ha tomado decisiones
→ y la nube, mientras, le cambió la configuración

LAS REGLAS QUE FUNCIONAN

- 1 CADA MEDICIÓN LLEVA SU MARCA DE TIEMPO DE ORIGEN
→ nunca la de llegada
→ y con reloj del dispositivo sincronizado y su desviación informada
- 2 LA SINCRONIZACIÓN ES IDEMPOTENTE Y REANUDABLE
→ identificador de secuencia por dispositivo y confirmación con marca de agua clase 242
→ si se corta a mitad, se reanuda donde iba
- 3 LO NUEVO PRIMERO, LO VIEJO DESPUÉS
→ al reconectar, primero el estado actual y las alarmas; el histórico va detrás, con menor prioridad
→ porque saber cómo está la máquina ahora vale más que los tres días anteriores
- 4 Y LA CONFIGURACIÓN SE APLICA CON VERSIÓN Y CONFIRMACIÓN
→ el gemelo tiene estado deseado y estado informado
→ y la diferencia entre ambos es la métrica que importa

→ y la ventana local define lo que se acepta perder
→ 72 horas de ventana significa que una desconexión de cinco días pierde dos
→ y eso se decide con negocio, no en ingeniería

Y el coste, que aquí se decide en el borde:

- 1,9 M mediciones/s enviadas en crudo son inasumibles
→ agregación en el borde: media, máximo, mínimo y desviación por ventana
→ y el dato crudo solo cuando hay anomalía o a petición
- reducción típica de 40 a 200 veces
→ y aquí ley 28 se ve con claridad: cada medición enviada es una factura, en transporte, en ingesta y en almacenamiento clase 270

3. Actualizar la flota sin perderla

El riesgo característico del sector: una actualización deja mil equipos sin acceso y hay que ir físicamente.

LO QUE LO EVITA

- 1 DOBLE PARTICIÓN
se escribe en la inactiva, se arranca en ella
→ si el arranque no confirma en N minutos, vuelve sola a la anterior
→ esta sola medida elimina la mayoría de las visitas
- 2 CONFIRMACIÓN DESDE EL DISPOSITIVO
no basta con que la descarga termine
→ el dispositivo debe informar de que arrancó, de que su función esencial pasa una comprobación y de que recupera conexión
- 3 DESPLIEGUE POR LOTES CON CANARIO
1 dispositivo → 10 → 1 % → 10 % → 100 % clase 102
y detención automática si el lote no confirma
- 4 VENTANA DE MANTENIMIENTO POR EMPLAZAMIENTO
no se actualiza una línea en producción
- 5 Y FIRMA Y VERIFICACIÓN DE LA IMAGEN
el dispositivo solo arranca lo que está firmado clase 216
→ porque un equipo en campo es físicamente accesible

→ y la métrica de la campaña
dispositivos actualizados, fallidos, revertidos y SIN

RESPUESTA

→ esa última es la que importa: son visitas

Y la identidad, que a esta escala es un problema en sí:

CADA DISPOSITIVO CON SU PROPIA CREDENCIAL

nunca una compartida por modelo o por lote

→ una credencial compartida convierte un equipo robado en acceso a la flota entera clase 269
idealmente en elemento seguro de hardware
y provisión en fábrica o en primera instalación

Y REVOCACIÓN INDIVIDUAL

un equipo retirado, robado o vendido deja de tener acceso

→ y esto requiere inventario de ciclo de vida clase 253

→ dispositivo dado de alta, activo, en mantenimiento, retirado

→ y el estado más peligroso es «instalado hace 7 años, nadie sabe dónde, sigue conectando»

Y la seguridad física, que en este sector es parte del modelo:

el atacante puede tener el dispositivo en la mano

→ asumir que el firmware se puede extraer
→ nada de secretos compartidos en la imagen
→ cifrado en reposo del almacenamiento local
→ y arranque verificado

→ y el modelo de amenazas debe incluir «un equipo comprometido informa datos falsos»

→ detectable comparando entre equipos del mismo emplazamiento

4. Las pruebas negativas del capstone

Lo que hay que ejecutar. Varias exigen un laboratorio con dispositivos reales.

DE DESCONEXIÓN

- cortar la red a un dispositivo 72 horas: ¿sigue haciendo su trabajo?
- al reconectar, ¿llega primero el estado actual o el histórico?
- ¿se duplica algún dato al reanudar una sincronización cortada?
- desconectar un emplazamiento entero 5 días: ¿qué se pierde?
- ¿qué pasa si el reloj del dispositivo se desvía 4 horas?

DE ACTUALIZACIÓN

- interrumpir la energía a mitad de una actualización: ¿arranca?
- instalar una imagen que no arranca: ¿vuelve sola?
- ¿cuántos dispositivos quedaron sin responder en la última campaña?
- actualizar un equipo con 4 versiones de retraso: ¿funciona?
- intentar instalar una imagen sin firmar: ¿la acepta?

DE IDENTIDAD Y CICLO DE VIDA

- ¿hay credenciales compartidas entre dispositivos?
- revocar un dispositivo: ¿deja de conectar de inmediato?
- ¿cuántos dispositivos conectan sin estar en el inventario?
- ¿cuántos llevan más de un año sin conectar y siguen autorizados?

DE DATOS Y COSTE

- ¿cuántas mediciones se envían y cuántas se usan?
- ¿la agregación del borde pierde información necesaria?

- un dispositivo que informa valores imposibles: ¿se detecta? clase 243
- ¿cuál es el coste por dispositivo y por mes?

DE OPERACIÓN

- ¿se puede diagnosticar un dispositivo sin ir físicamente?
- caer la nube 4 horas: ¿se para alguna línea?
- ¿existe procedimiento para un emplazamiento aislado y se ha usado?

El entregable del capstone:

- 1 el reparto de responsabilidad entre dispositivo, borde y nube, con el criterio
- 2 el diseño de sincronización y la ventana local acordada con negocio
- 3 el mecanismo de actualización y el resultado de la última campaña
- 4 el modelo de identidad y el inventario de ciclo de vida
- 5 la política de agregación y el coste por dispositivo
- 6 el modelo de amenazas con acceso físico
- 7 y el resultado de las pruebas negativas, con lo que falló

Y el cierre que enlaza con la clase siguiente: aquí el cliente es una planta y el dispositivo está lejos. En el siguiente, muchos clientes comparten la misma infraestructura y el reto es que ninguno se entere de los demás mientras se sabe cuánto cuesta cada uno. Multiinquilino y economía por cliente es la materia de la clase 283.

Ejemplo trabajado

El capstone resuelto. Lo que sigue es la campaña de actualización que dejó 1.140 equipos sin responder, la sincronización que duplicaba datos, y el reparto que evitó parar líneas cuando cayó la nube.

La campaña que salió mal, antes del rediseño.

campaña de actualización de firmware			
dispositivos objetivo	18.400		
método	envío directo, partición única		
despliegue	todos a la vez		
resultado			
actualizados y confirmados	16.980	92,3 %	
fallidos con reintento correcto	280		
SIN RESPONDER	1.140	6,2 %	
y el coste de esos 1.140			
visitas técnicas necesarias	1.140		
coste medio por visita	190 USD		
coste total	216.600 USD		
tiempo hasta recuperar la flota	11 semanas		
y 34 de ellos en emplazamientos remotos: 4 meses			

Y la causa:

- la nueva imagen fallaba al arrancar en dispositivos con una revisión de hardware concreta
 - esa revisión era el 6,1 % de la flota
 - y con partición única, un arranque fallido dejaba el equipo inutilizable
- un canario de 10 dispositivos elegidos por revisión de hardware lo habría detectado
- y la doble partición lo habría hecho irrelevante

Y el rediseño:

- doble partición con vuelta atrás automática a los 10 minutos sin confirmación
- confirmación en tres pasos: arranque, comprobación funcional y reconexión
- despliegue por lotes: 1 → 10 → 1 % → 10 % → 100 %
- con detención automática si un lote no confirma al 98 %

canario estratificado por revisión de hardware, modelo y tipo de conectividad y firma verificada en el arranque

campana siguiente, 19.100 dispositivos
actualizados y confirmados 18.874 98,8 %
revertidos automáticamente 214
→ todos de una revisión de hardware; detectados en el lote del 1 % y la campana se detuvo sola
SIN RESPONDER 12 0,06 %
visitas necesarias 12
coste 2.280 USD

La sincronización que duplicaba.

síntoma
los informes de producción de emplazamientos con conectividad intermitente daban entre un 3 % y un 11 % más de unidades producidas que el conteo físico

→ y llevaba dos años ley 29
→ nadie lo había investigado porque «los sensores a veces fallan»

Y la causa, encontrada agrupando por emplazamiento:

la sincronización enviaba un lote y esperaba confirmación
→ si el enlace caía tras enviar y antes de confirmar, el dispositivo reenviaba el lote entero
→ y la ingesta no deduplicaba

y el patrón
emplazamientos con enlace fiable 0 % de exceso
con enlace intermitente 3-11 %
con enlace satélite hasta 19 %

→ la correlación con la calidad del enlace fue lo que lo identificó clase 258

Y la corrección:

identificador de secuencia por dispositivo, monótono
confirmación con marca de agua: «he recibido hasta el número N»
deduplicación por (dispositivo, secuencia) en la ingesta y reanudación desde N+1, no reenvío del lote
clase 242

resultado
exceso frente al conteo físico 3-19 % → 0,02 %
y el 0,02 % restante se explicó por mediciones descartadas por calidad, documentadas

Y lo que el equipo anotó:

durante dos años, la planificación de producción usó cifras infladas entre un 3 % y un 19 % según el emplazamiento
→ y las decisiones de compra de materia prima se tomaron con ellas
→ el coste real de este fallo no fue técnico

El reparto de responsabilidad, puesto a prueba.

prueba «la nube no está disponible durante 4 horas»

antes del rediseño
las reglas de alarma se evaluaban en la nube
→ durante una caída de 40 minutos el año anterior:
3 líneas paradas por precaución
coste $14.000 \times 3 \times 0,67$
= 28.140 USD

después del rediseño
control y seguridad: local, siempre
reglas de alarma de proceso: local

reglas que necesitan varias fuentes del emplazamiento:
 pasarela de borde
 comparación entre plantas, modelos y mantenimiento
 predictivo: nube

resultado de la prueba de 4 horas
 líneas paradas 0
 alarmas locales disparadas y atendidas 7
 datos acumulados y sincronizados al volver 100 %
 tiempo de puesta al día tras reconectar 22 min
 y lo que se perdió ninguna
 medición

Identidad y ciclo de vida.

inventario inicial
 dispositivos en el sistema de gestión 41.000
 dispositivos que conectaron el último mes 43.180
 → 2.180 dispositivos conectando SIN estar en el
 inventario

investigación de esos 2.180
 instalados por integradores y nunca registrados 1.740
 equipos de laboratorio y pruebas 290
 equipos vendidos o retirados que seguían
 conectando 147
 sin identificar 3
 → los 3 usaban una credencial compartida de un lote
 de 2016

→ la credencial compartida del lote de 2016 daba acceso a
 4.100 dispositivos
 → llevaba 8 años activa ley 25

Y la corrección, que tardó:

provisión individual en la primera conexión para los
 equipos que lo soportaban 31.400
 visita técnica para los que no 4.100
 → escalonada en 14 meses, aprovechando mantenimientos
 revocación de la credencial compartida al terminar
 inventario obligatorio en el alta, con integradores

resultado a los 18 meses
 dispositivos fuera de inventario 2.180 → 0
 credenciales compartidas 1 → 0
 dispositivos autorizados sin conectar
 en más de 1 año 890 → 0

Las cifras finales del capstone.

ACTUALIZACIÓN		
dispositivos sin responder tras campaña	6,2 %	0,06 %
coste de visitas por campaña	216.600 USD	2.280 USD
tiempo de recuperación de flota	11 semanas	2 días

DATOS		
exceso de producción informado	3-19 %	0,02 %
mediciones enviadas/s	1,9 M	41.000
coste por dispositivo y mes	0,84 USD	0,19 USD

DESCONEXIÓN		
ventana local	4 horas	96 horas
líneas paradas por caída de nube (4 h)	3	0
datos perdidos en desconexión de 5 días	n/d	24 h del histórico

IDENTIDAD		
dispositivos fuera de inventario	2.180	0
credenciales compartidas	1	0
imágenes sin firmar aceptadas	sí	no

La lección que este capstone deja: una actualización sin doble partición dejó 1.140 equipos inaccesibles por una revisión de hardware que era el 6,1 % de la flota, y costó 216.600 USD en visitas y once semanas; el mismo fallo, con

doble partición y canario estratificado, produjo 214 vueltas atrás automáticas y doce visitas. Y durante dos años la planificación de producción usó cifras infladas hasta un 19 % porque una sincronización reenviaba lotes enteros al cortarse el enlace, sin que nada diera error.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/282-capstone-industria-iot-edge-y-operacion-desconectada/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 industry-capstone 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 industry-capstone 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 actualización deja cientos de equipos inaccesibles	Partición única y despliegue a toda la flota a la vez	Usa doble partición con vuelta atrás automática, confirmación funcional desde el dispositivo y lotes con canario estratificado por revisión de hardware.
Los informes cuentan más de lo que ocurrió y correlaciona con la calidad del enlace	La sincronización reenvía lotes al cortarse y la ingesta no deduplica	Numera secuencias por dispositivo, confirma con marca de agua, reanuda desde el siguiente y deduplica en la ingesta.
Una caída de la nube para líneas de producción	Decisiones con consecuencia física evaluadas en la nube	Toda decisión crítica va en el dispositivo o en la pasarela; la nube observa, ajusta y aprende, no decide en el instante.
Conectan dispositivos que no están en ningún inventario	Altas por terceros sin registro y credenciales compartidas por lote	Credencial individual por equipo, provisión en fábrica o primera conexión, inventario obligatorio en el alta y revocación individual.
El coste crece linealmente con el número de sensores	Se envían mediciones en crudo sin agregar en el borde	Agrega por ventana en la pasarela y envía crudo solo ante anomalía o a petición; cada medición enviada es una factura.
Al reconectar tras días sin red tarda mucho en saberse cómo está la máquina	Se sincroniza el histórico antes que el estado actual	Prioriza estado y alarmas al reconectar y manda el histórico detrás, con menor prioridad.

■ 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 decisiones deben tomarse en el dispositivo y por qué?
2. ¿Qué cuatro reglas hacen fiable una sincronización tras días sin red?
3. ¿Qué mecanismos evitan que una campaña de actualización obligue a visitas?
4. ¿Por qué una credencial compartida por lote es un fallo estructural?
5. ¿Qué define la ventana local y quién debe decidirla?

Referencias

- AWS (2024). IoT Lens, Well-Architected Framework.
<<https://docs.aws.amazon.com/wellarchitected/latest/iot-lens/iot-lens.html>>
- Microsoft (2024). Azure IoT reference architecture and device update.
<<https://learn.microsoft.com/azure/architecture/reference-architectures/iot>>
- IETF (2021). RFC 9019: A Firmware Update Architecture for Internet of Things.
<<https://www.rfc-editor.org/rfc/rfc9019.html>>
- IEC (2020). IEC 62443: seguridad en sistemas de automatización industrial.
<<https://www.iec.ch/blog/understanding-iec-62443>>
- Google Cloud (2024). Edge and industrial IoT architectures.
<<https://cloud.google.com/architecture/connected-devices>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Anterior	Índice	Siguiente
← 281 · Capstone media: streaming y distribución global	Parte 23 · Programa	283 · Capstone SaaS: multi-tenancy y unit economics →

Evaluación — 282 Capstone industria: IoT, edge y operación desconectada

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 industry-capstone 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.

283 — Capstone SaaS: multi-tenancy y unit economics

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

Propósito

Capstone de software como servicio: multiinquilino y economía por cliente. La clase da el encargo y la restricción que manda —hay que aislar a clientes que comparten infraestructura y saber cuánto cuesta cada uno—, el espectro de modelos de aislamiento con su coste real, el problema del vecino ruidoso, y las pruebas negativas de un sector donde el peor fallo posible es que un cliente vea los datos de otro.

Resultados de aprendizaje

Al finalizar podrás:

1. Situar un producto en el espectro de aislamiento con criterio.
2. Impedir el cruce de datos entre clientes por construcción.
3. Contener al vecino ruidoso sin penalizar a todos.
4. Calcular el coste y el margen por cliente y por plan.
5. Verificar el diseño con las pruebas negativas del sector.

Conceptos centrales

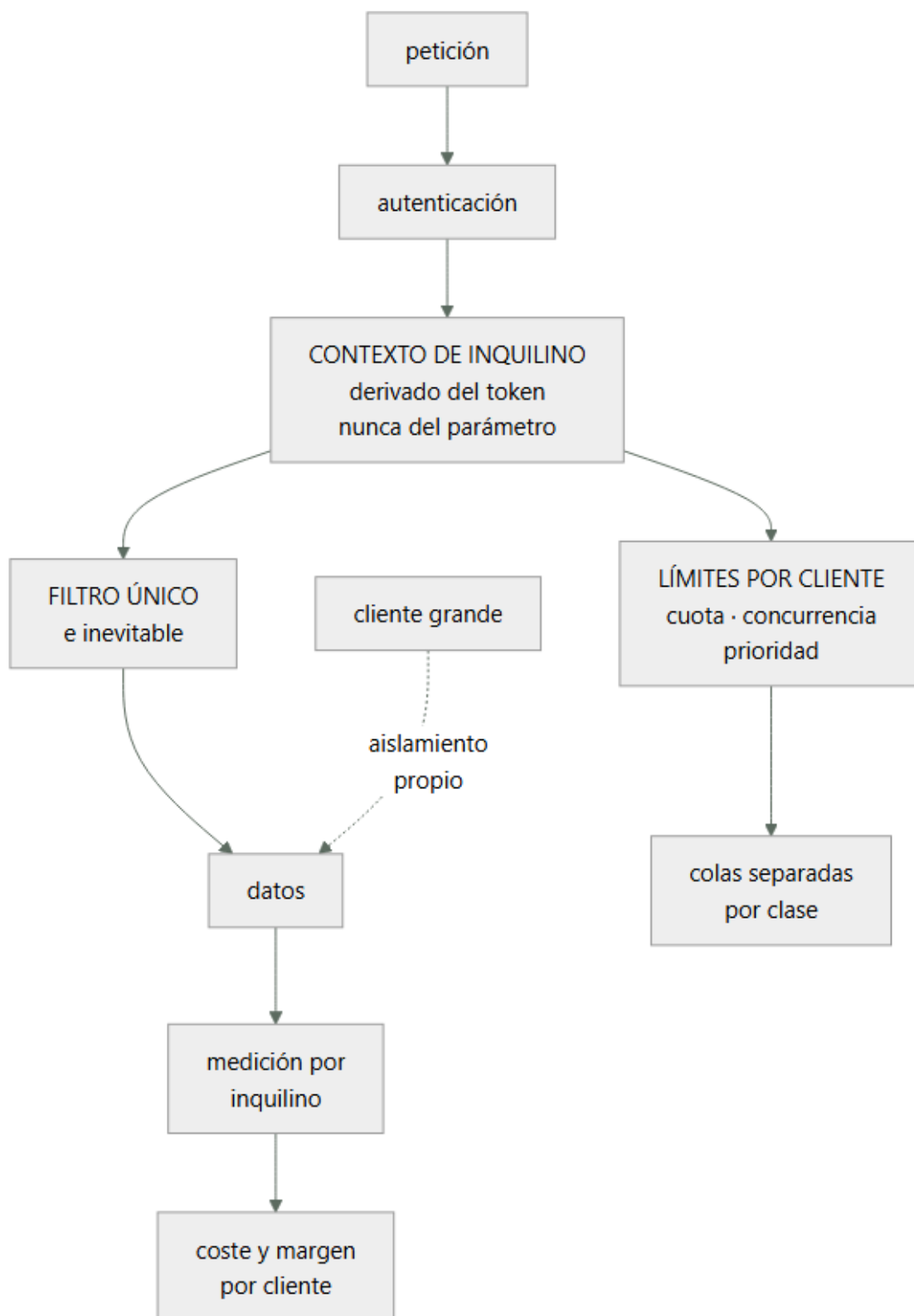
Concepto	Comprensión verificable
multiinquilino	Varios clientes comparten la misma infraestructura y código, con sus datos separados.
espectro de aislamiento	Desde todo compartido hasta una instalación por cliente. No es binario: se elige por capa.
vecino ruidoso	Un cliente cuyo consumo degrada a los demás. El fallo operativo característico del sector.
coste por cliente	Cuánto cuesta servir a cada uno. Sin él no hay precio defendible ni margen conocido.
cliente grande	El que rompe los supuestos: cien veces el volumen medio y exigencias propias.
filtro de inquilino	Restricción por cliente aplicada en un punto único e inevitable, no repetida en cada consulta.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 y el espectro de aislamiento

El encargo. Una plataforma de gestión para comercios: catálogo, pedidos, facturación e informes. 1.900 clientes, desde una tienda con dos empleados hasta una cadena con 400 locales.

CIFRAS DE PARTIDA
clientes

1.900

el mayor, en volumen	31 % del total
los 10 mayores	68 % del total
los 1.400 menores	6 % del total
planes	3

y el requisito comercial
 los grandes piden aislamiento y acuerdos propios
 los pequeños solo pueden pagar si el coste unitario es muy bajo

Y la restricción que manda:

DOS EXIGENCIAS QUE TIRAN EN SENTIDOS OPUESTOS

AISLAR que un cliente no pueda afectar ni ver a otro
 COMPARTIR que el coste por cliente permita el precio

→ y el sector entero es la gestión de esa tensión
 → resolverla con «una instalación por cliente» es fácil y hace inviable el negocio pequeño
 → resolverla con «todo compartido» es barato y produce el fallo que mata la empresa: un cliente viendo los datos de otro

Y el espectro, que se elige por capa y no de una vez:

CAPA	COMPARTIDO	SEPARADO
cómputo	mismo servicio	instancias dedicadas
datos	misma tabla con columna de inquilino	esquema por cliente · base por cliente
almacén de objetos	mismo contenedor con prefijo	contenedor por cliente
cifrado	clave común	clave por cliente
red	compartida	red por cliente
identidad	proveedor común	federación propia

→ y lo normal es una combinación
 cómputo compartido + datos compartidos para los pequeños
 cómputo compartido + base por cliente para los medianos
 todo separado para los tres o cuatro mayores

→ y lo que decide es el COSTE POR CLIENTE frente a lo que paga

Y el criterio para elegir la capa de datos, que es la decisión más cara de cambiar:

MISMA TABLA CON COLUMNA DE INQUILINO

- + coste mínimo, una migración de esquema para todos
- un error de filtro cruza datos
- un cliente grande genera particiones calientes
- clase 208
- restaurar un solo cliente es difícil

ESQUEMA O BASE POR CLIENTE

- + aislamiento fuerte y restauración individual
- + límites naturales por cliente
- miles de esquemas que migrar
- ley 23
- coste base por cliente que los pequeños no pagan

→ y el punto intermedio que funciona bien
 agrupar clientes en «lotes» de base compartida
 → 1.900 clientes en 40 bases
 → restauración por lote, migración por lote, y un cliente grande puede tener su lote propio

2. Que el cruce de datos sea imposible

El peor fallo del sector no es una caída: es que un cliente vea lo de otro. Y ocurre casi siempre por la misma causa.

LA CAUSA HABITUAL

el filtro por inquilino se aplica en cada consulta
 → y en algún sitio, alguien olvida ponerlo
 → una consulta de 340, en un informe poco usado

→ no se resuelve con disciplina ni con revisiones

→ se resuelve haciendo IMPOSIBLE consultar sin filtro

Y los mecanismos que lo hacen imposible:

- 1 EL CONTEXTO DE INQUILINO SE DERIVA DEL TOKEN
nunca de un parámetro de la petición
→ si viene en la ruta o en el cuerpo, un atacante lo cambia
- 2 UN ÚNICO PUNTO INEVITABLE
toda consulta pasa por una capa que añade el filtro
→ y consultar sin pasar por ella no compila o no arranca
o seguridad a nivel de fila en la base, con la identidad del inquilino en la sesión
→ así incluso una consulta mal escrita devuelve solo lo suyo
- 3 IDENTIFICADORES NO ADIVINABLES
→ y aun así, el filtro es lo que protege; los identificadores solo reducen el ruido
- 4 COMPROBACIÓN AUTOMÁTICA EN LA CADENA
una prueba que ejecuta cada consulta del sistema con dos inquilinos y verifica que ninguna devuelve datos del otro
clase 100
- 5 Y DETECCIÓN EN PRODUCCIÓN
toda respuesta lleva el identificador de inquilino de los datos; si no coincide con el del token, se corta y se alerta
→ esta comprobación cuesta poco y ha salvado a mucha gente

Y las mismas reglas en las demás capas:

almacén de objetos	prefijo por inquilino y política que lo impone; nunca listar el contenedor entero
cachés	la clave incluye el inquilino → el fallo clásico: una caché con clave sin inquilino sirve el dato de otro
colas y trabajos	el mensaje lleva el inquilino y el trabajador lo aplica
registros y trazas	el inquilino como etiqueta, y sin datos de cliente en el texto
e informes	los agregados no deben permitir deducir datos de otro cliente

3. El vecino ruidoso y el cliente grande

El fallo operativo característico: un cliente consume y todos lo notan.

CÓMO SE MANIFIESTA

una importación de 400.000 productos
un informe que recorre tres años
una integración que consulta cada segundo
o un cliente que crece diez veces en un mes

→ y sin defensas, la degradación es general y el soporte no sabe por qué

Y las defensas, por orden de eficacia:

- 1 LÍMITES POR CLIENTE, NO SOLO GLOBALES
peticiones por segundo, concurrencia, tamaño de consulta y trabajos simultáneos
→ y distintos por plan
- 2 SEPARAR LO INTERACTIVO DE LO PESADO
colas distintas para importaciones, informes y exportaciones
→ y una cola por clase, no una por cliente
→ así una importación gigante no bloquea la navegación

de nadie

3 JUSTICIA EN EL REPARTO

turno rotatorio entre clientes en las colas pesadas
→ un cliente con 10.000 trabajos no adelanta al que tiene 3

4 MEDICIÓN POR INQUILINO, SIEMPRE

→ sin ella, el diagnóstico de una degradación es imposible clase 258
→ y con ella, la pregunta «¿a qué clientes afecta?» se responde en segundos

5 Y AISLAMIENTO PARA LOS QUE LO JUSTIFIQUEN

→ cuando un cliente es el 31 % del volumen, tenerlo compartido es un riesgo, no un ahorro

Y el cliente grande, que rompe los supuestos:

LO QUE TRAE

volumen que satura particiones clase 208
exigencias de acuerdo de servicio propias
peticiones de configuración específica
auditorías y requisitos de seguridad
y a veces, residencia de datos propia

LO QUE NO HAY QUE HACER

añadir condicionales por cliente en el código
→ «si el cliente es X, entonces...»
→ esto es el principio del fin del multiinquilino

LO QUE SÍ

todo comportamiento distinto se expresa como CONFIGURACIÓN, no como código
y si necesita infraestructura propia, se despliega la MISMA versión en un lote dedicado
→ mismo código, distinto emplazamiento

→ y el día que haya dos versiones de código distintas por cliente, el producto se ha convertido en consultoría

4. Economía por cliente y pruebas negativas

Sin coste por cliente, el precio es una apuesta.

CÓMO SE MIDE

etiquetar por inquilino lo que se puede: almacenamiento, consultas, trabajos, tráfico
repartir lo compartido por una regla ACORDADA
→ por peticiones, por almacenamiento o por una mezcla
→ imperfecta y suficiente clase 270
y el resultado, por cliente y por plan

QUÉ REVELA, casi siempre

el plan más barato pierde dinero en la cola de clientes más activos
unos pocos clientes consumen desproporcionadamente
y algunas funciones cuestan mucho más de lo que se creía
→ informes largos, exportaciones, integraciones que consultan

QUÉ SE HACE CON ELLO

ajustar límites por plan
cobrar por uso lo que escala con el uso
poner al cliente grande donde su coste es visible
y retirar funciones cuyo coste no se recupera

Y las pruebas negativas del capstone:

DE AISLAMIENTO DE DATOS

- cambiar el identificador de inquilino en la petición: ¿devuelve datos de otro?
- ejecutar todas las consultas del sistema con dos inquilinos: ¿alguna cruza?
- ¿alguna caché tiene claves sin inquilino?

- ¿un trabajo en cola puede procesar datos del inquilino equivocado?
- ¿los informes agregados permiten deducir datos ajenos?
- ¿los registros contienen datos de cliente en texto?

DE VECINO RUIDOSO

- un cliente importa 400.000 registros: ¿lo notan los demás?
- un cliente lanza 50 informes a la vez: ¿se encolan justo?
- ¿existen límites por cliente y se han probado?
- ¿se puede responder en segundos a «a qué clientes afecta esto»?

DE CICLO DE VIDA

- restaurar los datos de UN cliente: ¿cuánto tarda?
- exportar todos los datos de un cliente que se va: ¿existe y se ha probado?
- borrar un cliente: ¿se borra de todas partes, incluidas copias y almacenes analíticos?
- migrar un cliente a un lote dedicado: ¿con qué parada?

DE ECONOMÍA

- ¿cuál es el coste del cliente más caro y del más barato?
- ¿algún plan pierde dinero y en qué tramo?
- ¿qué función tiene peor relación entre coste y valor?

El entregable del capstone:

- 1 la posición en el espectro por capa, con el motivo
- 2 el mecanismo que hace imposible el cruce de datos
- 3 los límites por plan y las clases de cola
- 4 el tratamiento del cliente grande, sin código específico
- 5 el coste y el margen por cliente y por plan
- 6 los procedimientos de restauración, exportación y borrado por cliente, probados
- 7 y el resultado de las pruebas negativas, con lo que falló

Y el cierre que enlaza con la clase siguiente: queda el último capstone sectorial, el que reúne datos e inteligencia artificial con gobierno, y que la hipótesis de la parte 23 señaló como el que más fallos silenciosos revelaría. Plataforma de datos e IA gobernada es la materia de la clase 284.

Ejemplo trabajado

El capstone resuelto. Lo que sigue es la consulta sin filtro que se encontró en una prueba automática, el vecino ruidoso que degradaba a 1.900 clientes, y la economía por cliente que reveló un plan con margen negativo.

La consulta sin filtro.

la prueba
se ejecutan todas las consultas del sistema con dos inquilinos preparados: A con 100 registros, B con 100 registros distintos
y se verifica que ninguna consulta ejecutada como A devuelve nada de B

consultas en el sistema	412
ejecutadas en la prueba	412

resultado de la primera ejecución	
consultas que cruzaban datos	3

Y las tres:

- 1 un informe de «productos más vendidos por categoría»
 - agregaba sin filtrar por inquilino
 - el resultado mezclaba ventas de todos los clientes
 - llevaba 19 meses en producción
 - lo usaban 41 clientes
- 2 una consulta de búsqueda por código de barras

- ```
3 la caché de configuración
→ clave: «config:{clave}» sin inquilino
→ el primer cliente que la pedía la dejaba cacheada
 para todos durante 5 minutos

→ ninguna había sido reportada por un cliente
→ y la 1 llevaba 19 meses dando datos de la competencia a
 41 comercios ley 29
```

seguridad a nivel de fila en la base, con el inquilino en la sesión  
→ una consulta sin filtro devuelve cero filas, no las de otro contexto de inquilino derivado del token, nunca de la petición  
claves de caché con inquilino obligatorio, verificado por el propio cliente de caché  
y comprobación en la respuesta: si un registro devuelto tiene un inquilino distinto al del token, se corta la respuesta y se alerta

cortes por la comprobación de respuesta, 12 meses  
2, ambos en desarrollo, ninguno en producción

síntoma  
dos o tres veces por semana, la aplicación iba lenta  
para todos durante 20-40 minutos  
sin patrón horario aparente

- la agrupación por inquilino durante un episodio
  - un cliente concentraba el 71 % de las consultas a la base
  - siempre el mismo tipo: exportación de catálogo completo
  - y su integración la lanzaba cada 20 minutos, no una vez al día como estaba documentado

- límites por cliente y por plan
  - plan básico 20 pet/s · 2 trabajos simultáneos
  - plan medio 80 pet/s · 6
  - plan avanzado 300 pet/s · 20→ y respuesta clara al superarlo, con cabecera de cuándo reintentar clase 201
- colas separadas
  - interactivo · importaciones · informes · exportaciones→ y turno rotatorio entre clientes dentro de cada cola
- exportación de catálogo completo con marca de cambios
  - el cliente solo se lleva lo modificado
  - su volumen bajó un 97 %

Multi-Cloud Engineering Program · Parte 23 · Capstones por industria y defensa final

clientes afectados por un vecino ruidoso 1.900 → 0  
quejas de soporte por lentitud 41/mes → 4/mes

Y lo que el equipo destacó:

la corrección más eficaz no fue el límite  
fue la exportación incremental: el cliente no quería  
consumir tanto, no tenía otra forma

→ el vecino ruidoso casi nunca es hostil  
→ suele ser un cliente al que le faltaba una función

La economía por cliente.

medición por inquilino durante 3 meses  
directo almacenamiento, consultas, trabajos,  
tráfico  
repartido plataforma y compartidos, por una mezcla  
acordada de peticiones y almacenamiento

resultado por plan

| plan     | clientes | precio  | coste medio | margen |
|----------|----------|---------|-------------|--------|
| básico   | 1.400    | 29 USD  | 11 USD      | 62 %   |
| medio    | 410      | 99 USD  | 38 USD      | 62 %   |
| avanzado | 90       | 390 USD | 201 USD     | 48 %   |

Y lo que apareció al mirar la distribución dentro de cada plan:

plan básico, por decil de consumo

| decil | coste medio | margen |
|-------|-------------|--------|
| 1-7   | 4 USD       | 86 %   |
| 8     | 9 USD       | 69 %   |
| 9     | 21 USD      | 28 %   |
| 10    | 67 USD      | -131 % |

→ el 10 % más activo del plan básico perdía 38 USD al mes  
cada uno  
→ 140 clientes × 38 = 5.320 USD/mes de pérdida  
→ y el plan en conjunto daba 62 % de margen, así que nadie  
lo había visto

Y las decisiones:

se identificó qué generaba ese consumo

|                                        |      |
|----------------------------------------|------|
| informes largos                        | 41 % |
| exportaciones completas                | 29 % |
| integraciones que consultaban en bucle | 22 % |
| resto                                  | 8 %  |

y se hicieron cuatro cosas

- 1 límites del plan básico ajustados a lo que el 90 %  
usaba, con mensaje claro al superarlos
- 2 informes largos: solo en plan medio o superior
- 3 exportación incremental para todos
- 4 y a los 140 clientes se les ofreció el plan medio con  
descuento el primer año

resultado a los 6 meses

|                                         |                |
|-----------------------------------------|----------------|
| clientes del básico con margen negativo | 140 → 11       |
| de los 140: pasaron a plan medio        | 79             |
| redujeron su uso                        | 50             |
| se dieron de baja                       | 11             |
| ingresos netos                          | +6.100 USD/mes |

El cliente grande.

el mayor cliente era el 31 % del volumen  
compartía lote con otros 46 clientes  
→ sus picos afectaban a esos 46  
→ y pedía un acuerdo de servicio propio

decisión

lote dedicado: misma versión de código, misma  
configuración de despliegue, base y cómputo propios

- NO una rama de código
- y las 3 diferencias de comportamiento que pedía se expresaron como configuración, disponible para todos

coste del lote dedicado                      2.900 USD/mes  
 lo que paga                                      14.200 USD/mes  
 y lo que se ganó además  
 los otros 46 clientes dejaron de sufrir sus picos  
 y la migración de ese cliente dejó de bloquear a los  
 demás                                              clase 260

Las cifras finales del capstone.

#### AISLAMIENTO

|                                        |          |    |
|----------------------------------------|----------|----|
| consultas que cruzan datos             | 3 de 412 | 0  |
| seguridad a nivel de fila              | no       | sí |
| cachés con clave sin inquilino         | 4        | 0  |
| comprobación de inquilino en respuesta | no       | sí |

#### VECINO RUIDOSO

|                                   |            |          |
|-----------------------------------|------------|----------|
| episodios de lentitud general     | 3/semana   | 0        |
| límites por cliente               | no         | sí       |
| colas separadas por clase         | no         | 4        |
| responder «a qué clientes afecta» | no medible | segundos |

#### CICLO DE VIDA

|                                  |                 |                    |
|----------------------------------|-----------------|--------------------|
| restaurar un solo cliente        | no posible      | 18 min             |
| exportar los datos de un cliente | manual,<br>días | 2 h,<br>automático |
| borrado completo verificado      | no              | sí                 |

#### ECONOMÍA

|                              |      |      |
|------------------------------|------|------|
| clientes con margen negativo | 140  | 11   |
| coste por cliente conocido   | no   | sí   |
| margen del plan básico       | 62 % | 74 % |

La lección que este capstone deja: un informe agregaba sin filtrar por inquilino y llevaba diecinueve meses enseñando ventas de la competencia a 41 comercios sin que nadie lo reportara; lo encontró una prueba automática que ejecuta las 412 consultas con dos clientes. Y el plan básico daba un 62 % de margen mientras su decil más activo perdía 38 USD al mes por cliente: la media ocultaba a 140 clientes que costaban más de lo que pagaban.

## Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/283-capstone-saas-multi-tenancy-y-unit-economics/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 saas-capstone 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 saas-capstone 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                                                                                                                                             |
|-----------------------------------------------------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| Un cliente ve datos de otro en un informe poco usado            | El filtro por inquilino se aplica consulta a consulta y en algún sitio falta        | Haz imposible consultar sin filtro: seguridad a nivel de fila, contexto derivado del token y comprobación del inquilino en la respuesta.               |
| Una caché devuelve la configuración de otro cliente             | La clave de caché no incluye el inquilino                                           | Exige inquilino en toda clave de caché desde el propio cliente de caché, y prueba cada consulta con dos inquilinos en la cadena.                       |
| La aplicación va lenta para todos varias veces por semana       | Un cliente consume desproporcionadamente y no hay límites ni medición por inquilino | Etiqueta todas las señales por inquilino, pon límites por plan y separa colas por clase con turno rotatorio.                                           |
| El plan barato parece rentable y la empresa no gana lo esperado | Se mira el margen medio del plan y no la distribución por consumo                   | Calcula coste por cliente y mira los deciles; la cola de clientes activos del plan barato suele tener margen negativo.                                 |
| Aparecen condicionales por cliente en el código                 | Se aceptaron exigencias de un cliente grande como excepciones de código             | Expresa toda diferencia como configuración disponible para todos, y si necesita infraestructura propia despliega la misma versión en un lote dedicado. |
| No se puede restaurar ni borrar los datos de un solo cliente    | Todo comparte una base sin agrupación por lotes                                     | Agrupar clientes en lotes de base compartida: permite restauración, migración y borrado por lote, y lote propio para el cliente grande.                |

## ■ 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 dos exigencias opuestas definen este sector?
2. ¿Por qué el espectro de aislamiento se elige por capa y no de una vez?
3. ¿Qué mecanismos hacen imposible el cruce de datos entre clientes?
4. ¿Cuáles son las defensas contra el vecino ruidoso y cuál suele ser su causa?
5. ¿Qué revela el coste por cliente que el margen medio del plan oculta?

## Referencias

- AWS (2024). SaaS Lens, Well-Architected Framework.  
<<https://docs.aws.amazon.com/wellarchitected/latest/saas-lens/saas-lens.html>>

- Microsoft (2024). Multitenant architectural guidance on Azure. <<https://learn.microsoft.com/azure/architecture/guide/multitenant/overview>>
- Google Cloud (2024). Multi-tenancy patterns. <<https://cloud.google.com/architecture/framework/multitenancy>>
- PostgreSQL (2024). Row Security Policies. <<https://www.postgresql.org/docs/current/ddl-rowsecurity.html>>
- FinOps Foundation (2024). Unit economics for SaaS providers. <<https://www.finops.org/framework/capabilities/unit-economics/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar:
 Parte 23 en PDF ·
 Manual integral

| Anterior                                                       | Índice              | Siguiente                                         |
|----------------------------------------------------------------|---------------------|---------------------------------------------------|
| ← 282 · Capstone industria: IoT, edge y operación desconectada | Parte 23 · Programa | 284 · Capstone datos e IA: plataforma gobernada → |

## Evaluación — 283 Capstone SaaS: multi-tenancy y unit economics

### 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 saas-capstone 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.                                |

## 284 — Capstone datos e IA: plataforma gobernada

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

### Propósito

Último capstone sectorial: plataforma de datos e inteligencia artificial gobernada. La clase da el encargo y la restricción que manda —el fallo no da error, el dato tiene dueño y finalidad, y el modelo hereda todos los defectos de lo que come—, el gobierno que lo hace sostenible, y las pruebas negativas que revelan lo que ninguna alerta detecta.

### Resultados de aprendizaje

Al finalizar podrás:

1. Diseñar la plataforma en el orden que evita rehacerla.
2. Gobernar el dato por dueño, finalidad y contrato, sin frenar el uso.
3. Detectar fallos que no producen error, con comparaciones y no con alertas.
4. Evaluar modelos con datos reales y decidir cuándo retirarlos.
5. Verificar el diseño con las pruebas negativas del sector.

### Conceptos centrales

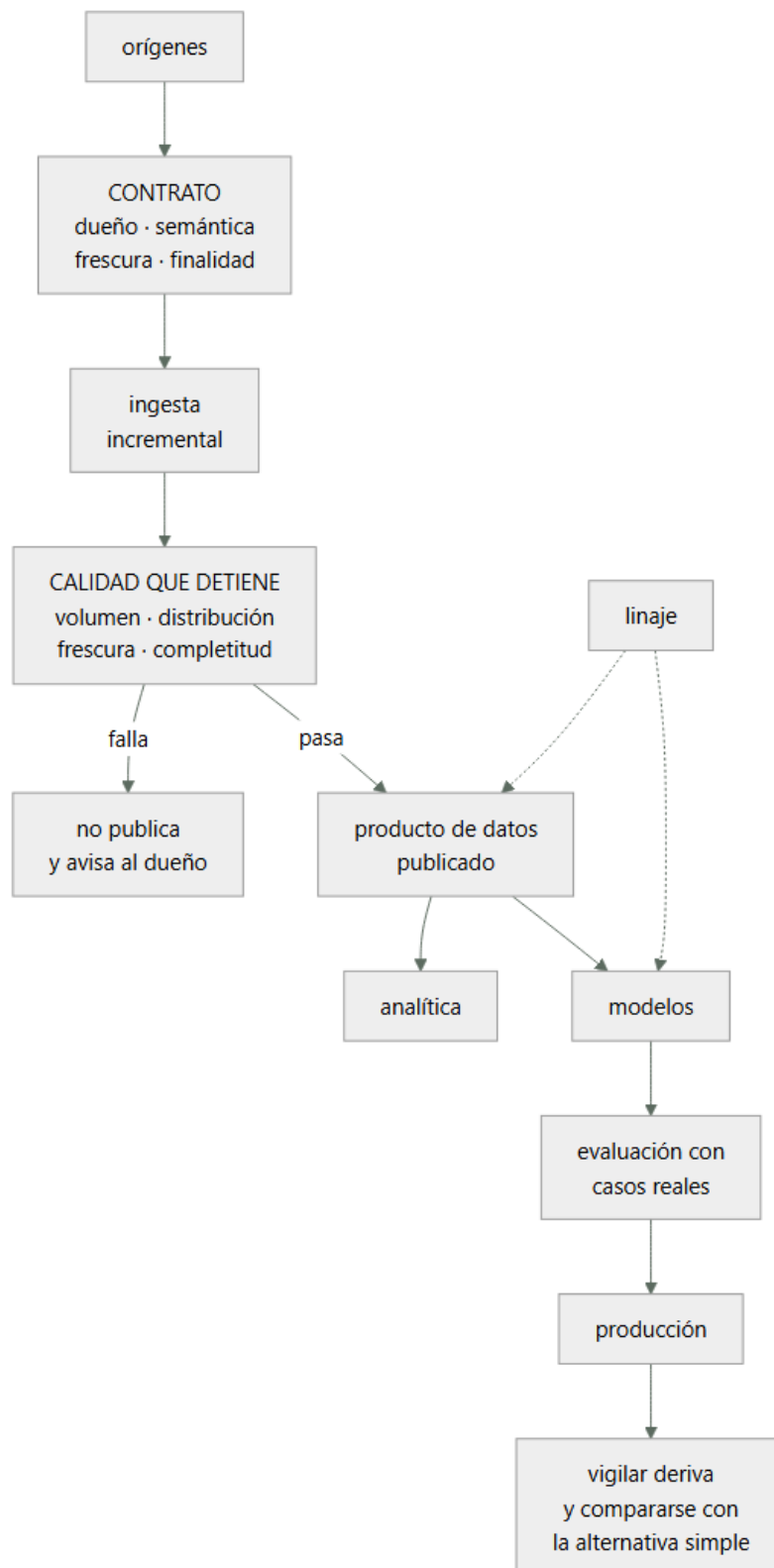
| Concepto                    | Comprensión verificable                                                                       |
|-----------------------------|-----------------------------------------------------------------------------------------------|
| producto de datos           | Conjunto publicado con dueño, contrato, semántica y compromiso de frescura.                   |
| gobierno útil               | El que hace más rápido el camino correcto en vez de prohibir los demás.                       |
| linaje                      | De dónde viene cada dato y qué depende de él. Lo que permite evaluar el impacto de un cambio. |
| comprobación que detiene    | Validación que impide publicar datos incorrectos, en vez de avisar después.                   |
| evaluación con casos reales | Medir con lo que ocurrió de verdad. Lo imaginado da números que engañan.                      |
| finalidad                   | Para qué se puede usar un dato. Determina quién accede y qué se permite derivar.              |

### Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 y la restricción que manda

El encargo. La plataforma de datos e inteligencia artificial de CloudShop: alimenta informes de negocio, recomendaciones, previsión de demanda, detección de fraude y el asistente operativo.

CIFRAS DE PARTIDA  
orígenes

|                                |            |
|--------------------------------|------------|
| conjuntos publicados           | 610        |
| consumidores internos          | 14 equipos |
| modelos en producción          | 9          |
| y el histórico                 |            |
| incidentes por dato incorrecto | 19/año     |
| detectados por una persona     | 17/19      |
| tiempo medio hasta detectarlos | 23 días    |

Y la restricción que manda:

EL FALLO NO DA ERROR ley 29  
 el canal se ejecuta, publica filas y devuelve un informe  
 → y todo es incorrecto  
 → y lo detecta alguien, semanas después, porque una  
 cifra le extraña

→ y de ahí sale toda la arquitectura de este capstone  
no sirve alertar sobre errores: hay que COMPARAR con lo  
de siempre  
volumen, distribución, frescura y completitud  
y detener antes de publicar, no avisar después

Y las otras dos restricciones, que son de gobierno:

EL DATO TIENE DUEÑO Y FINALIDAD

- no basta con que exista y sea correcto
- alguien responde de su significado
- y hay usos permitidos y usos que no
  - un dato recogido para facturar no vale para entrenar un modelo de segmentación

clase 251

EL MODELO HEREDA LOS DEFECTOS DE LO QUE COME  
un modelo sobre datos sin contrato reproduce sus sesgos,  
sus huecos y sus cambios de significado  
→ y los amplifica, porque decide muchas veces  
→ y su fallo también es silencioso: sigue prediciendo

clase 271

Y el orden que evita rehacerlo, que ya se demostró en la parte 20:

- ```
1 contratos y semántica           lo más caro de cambiar
2 ingesta y reproceso
3 calidad que detiene
4 atributos, calculados en un solo sitio
5 servicio y evaluación
6 el modelo                       lo más barato de cambiar
```

→ empezar por el 6 fue lo que costó siete semanas de rehacer en el caso de la clase 271

2. Gobierno que no frena

El gobierno de datos tiene mala fama merecida: comités, formularios y catálogos que nadie mantiene. Lo que funciona es otra cosa.

EL GOBIERNO QUE FRACASA
 aprobación manual para publicar
 catálogo mantenido a mano
 normas escritas sin comprobación automática
 → y el resultado medido: el 71 % del consumo lo rodea
 con exportaciones y copias ley 16

EL GOBIERNO QUE FUNCIONA

- publicar por el camino oficial es LO MÁS RÁPIDO
 - 2 días, no once semanas
- el contrato genera el resto
 - permisos derivados del contrato
 - documentación generada, no escrita
 - comprobaciones de calidad derivadas del esquema
 - y linaje capturado automáticamente
- y lo que se prohíbe, se prohíbe técnicamente
 - no con una norma que recordar

Y las piezas del gobierno, con lo que aporta cada una:

CONTRATO

esquema, semántica en palabras, dueño, frescura
comprometida, finalidad permitida y clasificación de
sensibilidad
→ y la semántica escrita es lo que evita el error más
caro: un campo llamado «importe» que significa tres
cosas clase 252

CATÁLOGO GENERADO

de los contratos, no a mano
→ un catálogo mantenido a mano está desactualizado
siempre ley 25

LINAJE AUTOMÁTICO

qué alimenta qué
→ y sirve para dos cosas concretas
evaluar el impacto de un cambio antes de hacerlo
y encontrar, tras un fallo, todo lo que quedó
contaminado
→ esta segunda es la que salva semanas de trabajo

CLASIFICACIÓN Y FINALIDAD

qué es personal, qué es sensible, para qué se puede usar
→ y controles automáticos que impiden que un dato
personal acabe en un conjunto de entrenamiento sin
autorización

Y RETIRADA

los conjuntos sin consumo se marcan y se retiran
→ porque un catálogo con 610 conjuntos de los que se
usan 180 es un catálogo inservible

Y la métrica que dice si el gobierno funciona:

no «cuántos conjuntos catalogados»
sino QUÉ PORCENTAJE DEL CONSUMO PASA POR LA PLATAFORMA
→ y si baja, el camino oficial se ha vuelto lento
→ es la misma señal de adopción de la clase 267

3. Detectar lo que no da error

El corazón técnico del capstone.

LAS CUATRO COMPARACIONES

VOLUMEN

¿cuántas filas frente a lo habitual para este día de
la semana?

DISTRIBUCIÓN

¿la media, los percentiles y la proporción de valores
nulos se parecen a las de siempre?

→ esto es lo que detecta el cambio de céntimos a
euros: todos los valores son posibles

FRESCURA

¿el dato más reciente es de cuándo?

COMPLETITUD

¿faltan orígenes, particiones o claves esperadas?

→ y las cuatro DETIENEN la publicación, no avisan
→ porque una vez publicado, el error se propaga a todo lo
que consume clase 243

Y las comprobaciones que la experiencia añade:

REFERENCIAS CRUZADAS

el total de ventas del almacén frente al del sistema
operacional

→ discrepancia mayor del X %, se detiene
→ es la comprobación más potente y la que menos se hace

BORRADOS

¿cuántos borrados trajo la ingesta incremental?
→ si lleva meses sin traer ninguno, algo está mal
clase 242

CONTINUIDAD TEMPORAL

¿hay huecos de horas o días sin datos?

Y VALORES IMPOSIBLES POR CONTEXTO

una fecha de entrega anterior al pedido

un descuento del 340 %

→ y estos sí dan error si se comprueban

Y el bucle de reproceso, que hay que tener antes de necesitarlo:

CUANDO SE DETECTA UN FALLO ANTIGUO

- 1 el linaje dice qué quedó contaminado
- 2 se marca lo afectado como no fiable
- 3 se reprocesa desde el origen, con la corrección
- 4 se recalculan los derivados en orden
- 5 y se avisa a los consumidores de qué cifras cambian

→ el paso 5 es el que casi nadie hace

→ y es el que evita que alguien siga decidiendo con la cifra vieja

Y los modelos, con lo que la parte 20 dejó medido:

ATRIBUTOS EN UN SOLO SITIO

el mismo cálculo al entrenar y al servir clase 244

→ 17 atributos calculados distinto costaron un error de 3,9 días frente a 1,4

EVALUACIÓN CON CASOS REALES

60 casos imaginados daban 94 %

los reales dieron 67 % clase 250

COMPARACIÓN CON LA ALTERNATIVA SIMPLE, SIEMPRE

una regla, una media móvil, un umbral

→ y si consigue el 90 % con el 3 % del coste, el modelo sobra

Y VIGILANCIA DE DERIVA

distribución de entradas y de salidas

→ un modelo se degradó de 1,4 a 3,9 días de error

durante catorce meses sin que nada avisara

clase 246

4. Las pruebas negativas del capstone

Lo que hay que ejecutar, y que revela lo que ninguna alerta muestra.

DE CALIDAD SILENCIOSA

- multiplicar por 100 los importes de un origen: ¿se detiene la publicación?
- cambiar la unidad de un campo sin cambiar el esquema: ¿se detecta?
- ejecutar un trabajo antes de que llegue un origen: ¿publica parcial?
- eliminar el 30 % de las filas de una partición: ¿se nota?
- ¿cuánto tarda la ingesta incremental en traer un borrado?
- ¿cuadran las ventas del almacén con las del sistema operacional, hoy?

DE GOBIERNO

- ¿cuánto tarda publicar un conjunto nuevo?
- ¿qué porcentaje del consumo pasa por la plataforma?
- ¿cuántos conjuntos publicados no tiene nadie consumiendo?
- ¿algún conjunto no tiene dueño identificable hoy?
- ¿un dato personal puede acabar en un entrenamiento sin autorización?
- dado un campo, ¿qué informes y modelos dependen de él?

DE MODELOS

- ¿los atributos se calculan igual al entrenar y al servir? demuéstalo
- ¿de dónde salieron los casos de evaluación?
- ¿cuál es el resultado de la alternativa simple para

- cada modelo?
- ¿qué modelo lleva más tiempo sin evaluarse contra datos recientes?
- ¿algún modelo usa un atributo que en producción no existe al predecir?

DE RECUPERACIÓN

- reprocesar tres meses desde el origen: ¿cuánto tarda y da lo mismo?
- ante un fallo detectado hoy con origen hace 40 días, ¿qué quedó contaminado?
- ¿se avisa a los consumidores de que sus cifras cambian?

El entregable del capstone:

- 1 el mapa de productos de datos con dueño, contrato y finalidad
- 2 el conjunto de comprobaciones que detienen, por producto
- 3 el linaje y una demostración de análisis de impacto
- 4 el inventario de modelos con su evaluación, su alternativa simple y su deriva
- 5 el procedimiento de reproceso y de aviso a consumidores
- 6 el coste por consulta y por predicción clase 270
- 7 y el resultado de las pruebas negativas, con lo que falló

Y el cierre que enlaza con la clase siguiente: con los ocho sectores resueltos, queda someterlos a la prueba que este programa considera definitiva: provocar los fallos y ver qué hace la organización. El ensayo integrado es la materia de la clase 285.

Ejemplo trabajado

El capstone resuelto. Lo que sigue son las seis pruebas de calidad silenciosa con su resultado, el linaje que convirtió una investigación de semanas en veinte minutos, y los tres modelos que se retiraron.

Las seis pruebas de calidad silenciosa.

- prueba 1 · multiplicar por 100 los importes de un origen
antes publicaba; el informe de ingresos daba $\times 100$ y alguien lo veía al día siguiente
después DETENIDO en 4 minutos
→ la comprobación de distribución detectó que la media se salía 41 desviaciones
- prueba 2 · cambiar céntimos a euros sin cambiar esquema
antes publicaba; todos los valores eran posibles
→ este fue el fallo real de la parte 20
después DETENIDO
→ media dividida por 100; distribución incompatible con el histórico
- prueba 3 · ejecutar un trabajo antes de que llegue un origen
antes publicaba con el 61 % de los pedidos, sin error
después DETENIDO por la comprobación de completitud
→ «faltan 3 de 8 orígenes esperados»
- prueba 4 · eliminar el 30 % de las filas de una partición
antes no se detectaba
después DETENIDO por volumen frente al histórico del mismo día de la semana
- prueba 5 · ¿cuánto tarda la ingesta en traer un borrado?
antes NUNCA: llevaba 2 años sin traer un solo borrado
→ registros borrados en el origen seguían vivos en el almacén clase 242
después detectado por la comprobación «borrados en los últimos 30 días = 0» y corregido
→ aparecieron 41.000 registros que debían haber desaparecido

prueba 6 · ¿cuadran las ventas con el sistema operacional?
 primera ejecución discrepancia 2,7 %
 → investigada: pedidos cancelados contados distinto en los dos sitios
 → definición acordada y escrita después discrepancia 0,02 %
 y comprobación diaria que detiene si supera el 0,5 %

Y el efecto agregado:

incidentes por dato incorrecto	19/año	3/año
detectados por una persona	17/19	0/3
tiempo medio hasta detectar	23 días	11 min
publicaciones detenidas por calidad	n/d	41/mes
de ellas, falsos positivos	n/d	6

Y lo que costaron los falsos positivos:

6 detenciones al mes eran cambios legítimos
 campañas, promociones, cambios de negocio reales

→ y cada una requería una persona para desbloquear
 → se añadió un mecanismo: el dueño del producto puede aprobar la publicación con un motivo escrito
 → y esas aprobaciones se revisan mensualmente

→ sin ese mecanismo, la calidad que detiene se habría desactivado en tres meses ley 16

El linaje, puesto a prueba.

el caso real
 se detecta que un campo de un origen cambió de significado hace 40 días

SIN LINAJE, lo que costaba antes
 buscar a mano qué consultas usan ese campo
 preguntar a los 14 equipos
 revisar informes uno a uno
 → estimación del equipo: 2-3 semanas
 → y con la certeza de olvidarse de alguno

CON LINAJE

consulta de impacto	20 minutos
resultado	
conjuntos derivados afectados	31
informes que los usan	84
modelos que los usan	3
equipos a avisar	9

y lo que se hizo
 los 31 conjuntos marcados como no fiables
 reproceso de 40 días desde el origen 6 horas
 recálculo de derivados en orden topológico 3 horas
 aviso a los 9 equipos con la lista de cifras que cambiaban y en cuánto
 y 2 de los 3 modelos reentrenados; el tercero no usaba ese atributo de forma significativa

tiempo total 2 días

Y el detalle que el equipo consideró más valioso:

el aviso a los consumidores incluía
 «el informe de margen por categoría cambia entre un 3 % y un 7 % en el periodo del 1 de marzo al 9 de abril»

→ porque tres de esos informes se habían usado en decisiones de compra
 → y sin el aviso, nadie habría revisado esas decisiones

Los modelos: evaluación y retirada.

inventario inicial: 9 modelos en producción

cada uno evaluado con

casos reales del último trimestre
y comparación con la alternativa simple

modelo	resultado	alternativa	coste/mes
previsión de demanda	1,6 d	media	
		móvil: 2,9 d	3.100
recomendación de producto	+4,1 %	más vendidos de categoría:	
		+3,7 %	8.400
detección de fraude	0,91	reglas: 0,74	2.200
previsión de entrega	1,4 d	histórico por transportista:	
		2,1 d	1.900
segmentación de clientes	n/d	no medible	900
predicción de abandono	0,68	antigüedad + actividad:	
		0,66	1.400
ordenación de búsqueda	+2,3 %	popularidad:	
		+2,1 %	2.800
clasificación de incidencias	0,88	palabras clave: 0,71	600
precio dinámico	+1,9 %	reglas: +1,8 %	4.200

Y las decisiones:

SE RETIRARON 3

- recomendación de producto
 - +4,1 % frente a +3,7 % de «más vendidos de la categoría»
 - 0,4 puntos por 8.400 USD/mes
 - y la diferencia no era significativa con el volumen disponible
- predicción de abandono
 - 0,68 frente a 0,66 de dos variables
 - retirado
- precio dinámico
 - +1,9 % frente a +1,8 % de las reglas
 - retirado, y además reducía la explicabilidad ante clientes

SE MANTUVIERON 5, con margen claro

Y 1 SE DEJÓ EN OBSERVACIÓN

- segmentación de clientes: no se pudo medir su efecto
 - se le dio un trimestre para definir una métrica
 - no se definió; se retiró

ahorro directo 14.500 USD/mes

y el ahorro indirecto

- 4 modelos menos que mantener, evaluar y vigilar
- aproximadamente 0,6 personas

Y la observación que el equipo escribió:

- ninguno de los cuatro retirados era un mal modelo
 - eran modelos que funcionaban y no aportaban lo bastante frente a algo mucho más simple
 - y ninguno se había comparado nunca con esa alternativa
 - porque nadie compara con la alternativa simple DESPUÉS de haber construido el modelo
- por eso la comparación tiene que ser un requisito de entrada a producción, no una revisión posterior

El gobierno, medido.

plazo de publicar un conjunto nuevo	11 semanas	2 días
consumo que pasa por la plataforma	29 %	96 %
conjuntos publicados	610	184
sin consumo en 90 días	426	0
conjuntos sin dueño identificable	171	0
catálogo mantenido a mano	sí	no
linaje disponible	no	sí
análisis de impacto	2-3 semanas	20 min
controles que impiden dato personal en entrenamiento	no	sí

La lección que este capstone deja: la ingesta incremental llevaba dos años sin traer un solo borrado y nadie lo sabía, porque un canal que se ejecuta bien y publica filas no da ningún error; lo encontró una comprobación que preguntaba lo obvio. Y de nueve modelos en producción, cuatro se retiraron no por ser malos sino porque una regla o una media móvil conseguía casi lo mismo por una fracción del coste: ninguno se había comparado nunca con esa alternativa.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/284-capstone-datos-e-ia-plataforma-gobernada/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 data-ai-capstone 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 data-ai-capstone 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
Un dato incorrecto se descubre semanas después por una persona	Se alerta sobre errores y un fallo de datos no produce error	Compara volumen, distribución, frescura y completitud contra el histórico, y detén la publicación en vez de avisar después.
La ingesta funciona y faltan registros borrados en el origen	La captura incremental no trae borrados y nadie lo comprueba	Añade una comprobación explícita de borrados por ventana; cero borrados durante meses es una anomalía, no una tranquilidad.
Las comprobaciones que detienen acaban desactivadas	Los cambios legítimos de negocio las disparan y desbloquear exige a otra persona	Permite que el dueño del producto apruebe la publicación con motivo escrito y revisa esas aprobaciones cada mes.
Tras detectar un fallo antiguo nadie sabe qué quedó contaminado	No hay linaje capturado automáticamente	Captura linaje en la ejecución y ten preparada la consulta de impacto; después avisa a los consumidores de qué cifras cambian y cuánto.
El catálogo tiene cientos de conjuntos y nadie encuentra nada	Se cataloga todo y no se retira lo que no se consume	Genera el catálogo de los contratos, marca lo que lleva 90 días sin consumo y retíralo.
Hay modelos en producción que nadie sabe si aportan	Nunca se compararon con una alternativa simple	Haz de esa comparación un requisito de entrada a producción y revísala periódicamente; muchos modelos no sobreviven a una media móvil.

■ 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 en esta plataforma no sirve alertar sobre errores?
2. ¿Qué distingue un gobierno de datos que funciona de uno que se rodea?
3. ¿Cuáles son las cuatro comparaciones y qué detecta cada una?
4. ¿Para qué sirve el linaje en los dos momentos en que importa?
5. ¿Por qué la comparación con la alternativa simple debe ser requisito de entrada?

Referencias

- Kleppmann, M. (2017). Designing Data-Intensive Applications. <<https://dataintensive.net/>>
- Sculley, D. y otros (2015). Hidden technical debt in machine learning systems. <<https://papers.nips.cc/paper/2015/hash/86df7dcfd896fc2674f757a2463eba-Abstract.html>>
- Google Cloud (2024). Data governance and lineage on BigQuery and Dataplex. <<https://cloud.google.com/dataplex/docs/data-lineage>>
- AWS (2024). Machine Learning Lens and data governance guidance. <<https://docs.aws.amazon.com/wellarchitected/latest/machine-learning-lens/machine-learning-lens.html>>
- Microsoft (2024). Data governance with Microsoft Purview. <<https://learn.microsoft.com/purview/governance-solutions-overview>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 23 en PDF · Manual integral

Anterior	Índice	Siguiente
← 283 · Capstone SaaS: multi-tenancy y unit economics	Parte 23 · Programa	285 · Game day integrado y respuesta a incidentes →

Evaluación — 284 Capstone datos e IA: plataforma gobernada

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 data-ai-capstone 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.

285 — Game day integrado y respuesta a incidentes

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 4 Laboratorio: chaos · Estado: EXECUTABLECORE

Propósito

Ensayo integrado: provocar fallos en varios sistemas a la vez y observar cómo responde la organización entera. La clase da el diseño de un ejercicio de varias horas con escenarios encadenados, cómo se observa sin intervenir, cómo se evalúa lo que ocurre, y por qué —según la hipótesis de la parte 23— la mayoría de los hallazgos volverán a ser organizativos.

Resultados de aprendizaje

Al finalizar podrás:

1. Diseñar un ejercicio integrado con escenarios encadenados y radio acotado.
2. Organizar roles, observación y criterios de parada para varias horas.
3. Provocar los fallos que cruzan equipos, que son los que no se ensayan.
4. Evaluar la respuesta con una rúbrica y extraer acciones cerradas.
5. Comparar los hallazgos con lo que la hipótesis de la parte 23 predijo.

Conceptos centrales

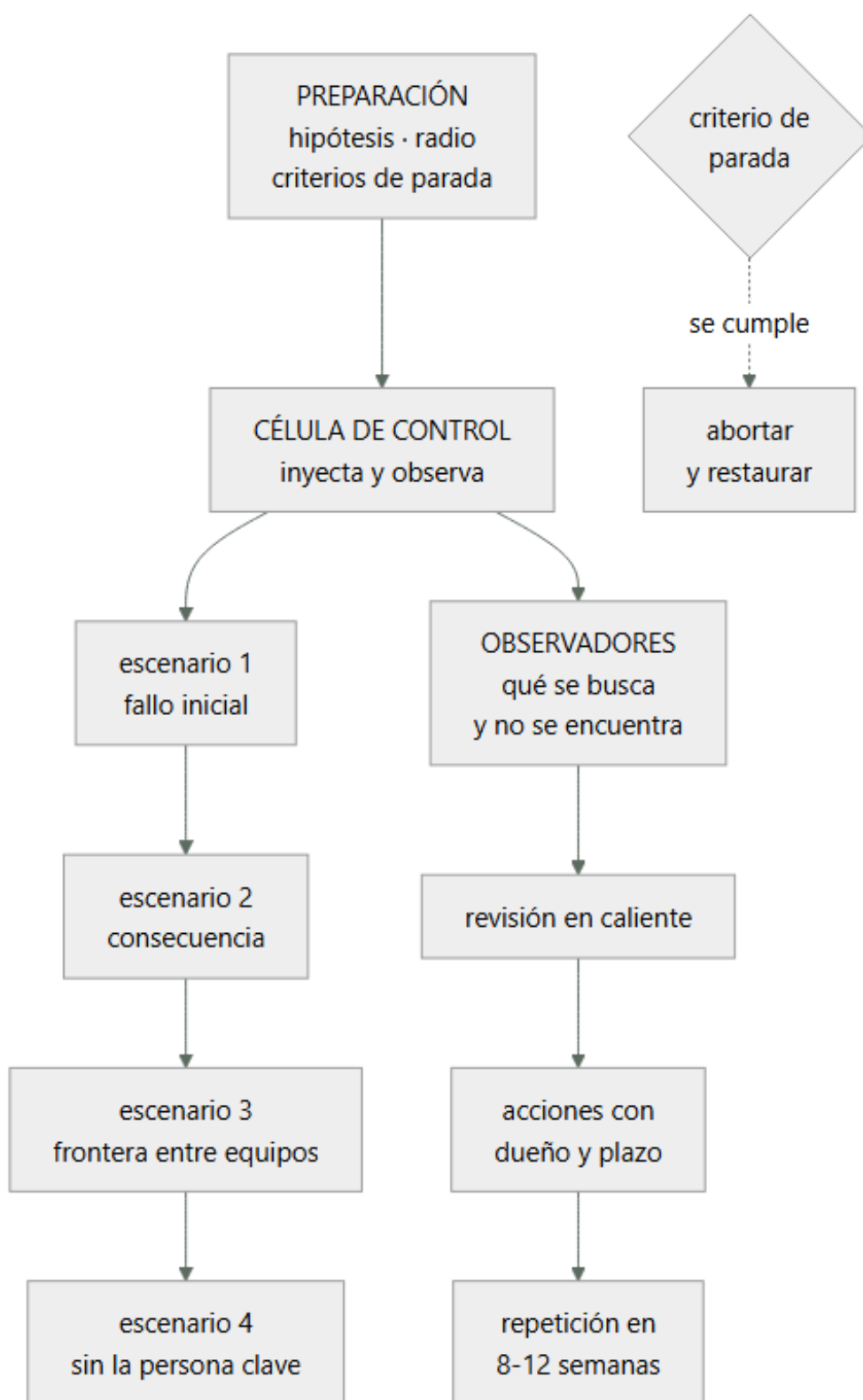
Concepto	Comprensión verificable
ensayo integrado	Ejercicio de varias horas con fallos encadenados que cruzan varios equipos y sistemas.
escenario encadenado	Un fallo que provoca otro, como ocurre de verdad. Lo que ningún ensayo aislado prueba.
célula de control	Grupo que dirige el ejercicio, inyecta los fallos y puede abortarlo.
observador	Quien anota qué ocurre y cuándo, incluido lo que se busca y no se encuentra.
fallo de frontera	El que ocurre entre equipos y por eso no tiene dueño hasta que alguien lo reclama.
acción cerrada	Hallazgo convertido en cambio con dueño, plazo y comprobación posterior.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 añade un ensayo integrado

Los ensayos por sistema encuentran lo que falla dentro de un equipo. El integrado encuentra lo que falla entre equipos, que es donde ocurren los incidentes graves.

LO QUE UN ENSAYO AISLADO NO ENCUENTRA
quién decide cuando el problema afecta a dos equipos

qué pasa cuando dos incidentes coinciden clase 258
cómo se comunica con un cliente externo mientras se
arregla
si el equipo A sabe que el equipo B ya está trabajando
en ello
y qué ocurre cuando la respuesta de uno empeora lo del
otro

- y estos son exactamente los fallos que alargan los incidentes reales
- porque no tienen dueño hasta que alguien los reclama

Y los escenarios que solo se pueden probar así:

FALLOS DE FRONTERA

una dependencia compartida se degrada y afecta a cuatro
servicios de tres equipos clase 185
→ ¿quién coordina?

CAUSAS SIMULTÁNEAS

dos incidentes distintos a la vez
→ y la correlación automática los presenta como uno
clase 263

EFFECTOS CRUZADOS

el equipo A escala su servicio y satura la base del B
clase 262
el equipo B activa el vertido de carga y el A ve errores
que no entiende

DECISIONES QUE NADIE PUEDE TOMAR SOLO

conmutar de región
desactivar una función que afecta a ingresos
comunicar públicamente
→ y aquí aparece el tramo que domina el tiempo, que la
parte 21 midió: decidir y comunicar clase 264

Y LA AUSENCIA DE PERSONAS

quien más sabe no participa clase 261

2. El diseño del ejercicio

Un ensayo integrado dura entre tres y seis horas y necesita estructura.

ANTES

- 1 HIPÓTESIS por escenario, escritas
incluida la de detección: ¿saltará la alerta y en
cuánto?
- 2 RADIO DE IMPACTO máximo, por escenario
- 3 CRITERIOS DE PARADA medibles, y quién puede
declararlos
→ cualquiera
- 4 RESTAURACIÓN probada de cada inyección
- 5 ANUNCIO a todos los equipos y a atención al cliente
→ y aviso a los proveedores externos si procede
- 6 Y REGLA DE ORO
si ocurre un incidente REAL, el ejercicio se detiene
de inmediato

DURANTE

CÉLULA DE CONTROL

inyecta, cronometra, decide si escalar el escenario y
puede abortar
→ no responde preguntas ni da pistas

PARTICIPANTES

los equipos de guardia reales, no los autores

OBSERVADORES

uno por equipo, que anota
qué se mira y en qué orden
qué se busca y no se encuentra
qué se supone sin comprobar
cuándo se pide ayuda y a quién
y los silencios: cuánto tiempo sin que nadie diga
nada

DESPUÉS

revisión en caliente de 45 minutos, ese mismo día
y acciones con dueño y plazo en 48 horas

Y el encadenamiento, que es lo que lo hace realista:

EL PATRÓN DE UN INCIDENTE REAL

algo se degrada

- la respuesta automática empeora otra cosa
- una alerta se dispara en otro equipo
- y mientras se investiga, aparece un segundo problema sin relación

- un ensayo con un solo fallo no reproduce nada de esto
- y por eso los escenarios se encadenan, con la célula de control decidiendo cuándo introducir el siguiente

Y LA REGLA DE ESCALADA

el escenario siguiente entra cuando el anterior está
contenido o cuando pasan N minutos
→ lo segundo es lo que crea presión realista

Y las dos cosas que hay que resistir:

AYUDAR

quien diseñó el ejercicio conoce la respuesta
→ y el impulso de dar una pista destruye el hallazgo
→ los observadores no hablan

Y AMPLIAR EL RADIO SOBRE LA MARCHA

«esto va bien, vamos a probar también...»
→ los criterios de parada y el radio se fijaron antes
por una razón clase 261

3. Qué se evalúa

La rúbrica del ejercicio, conocida por todos antes de empezar.

- 1 DETECCIÓN**
¿saltó la alerta? ¿en cuánto? ¿la recibió quien debía?
¿alguien se enteró por otro camino antes que por la alerta?
 - 2 COORDINACIÓN**
¿se declaró incidente y en cuánto?
¿hubo alguien coordinando o todos arreglando? clase 257
¿se supo quién hacía qué?
 - 3 DIAGNÓSTICO**
¿se consultó la línea de cambios primero? clase 258
¿se escribieron hipótesis alternativas?
¿se comprobó lo que se daba por sabido?
 - 4 DECISIÓN**
¿se mitigó antes de entender?
¿quién tomó las decisiones que cruzaban equipos?
¿hubo alguna decisión que nadie se atrevió a tomar?
 - 5 COMUNICACIÓN**
¿se comunicó a negocio y a clientes? ¿cuándo y qué?
¿la primera comunicación tardó más de lo acordado?
 - 6 Y HERRAMIENTAS**
¿los procedimientos funcionaron?
¿los permisos alcanzaron?
¿algún panel no mostraba lo necesario?
- y las seis se puntúan con lo observado, no con la impresión

Y las cifras que se extraen del ejercicio:

tiempo hasta detectar, por escenario
tiempo hasta declarar incidente

tiempo hasta la primera hipótesis correcta
tiempo hasta mitigar
tiempo hasta la primera comunicación
número de personas implicadas frente a las necesarias
procedimientos usados y cuántos funcionaron
hallazgos, clasificados en técnicos y organizativos
y decisiones bloqueadas por falta de autoridad

Y la clasificación de hallazgos, que la hipótesis de la parte 23 predijo:

la predicción escrita en la clase 276
«alrededor de dos tercios de los hallazgos volverán a
ser organizativos»

- y esta clase la pone a prueba
- el ejercicio de CloudShop lo comprueba en el ejemplo

4. De los hallazgos a las acciones

Un ejercicio sin acciones cerradas es una anécdota cara.

LA REVISIÓN EN CALIENTE, ese mismo día
¿qué esperábamos y qué pasó?
¿qué nos sorprendió?
¿qué buscamos y no encontramos?
¿qué decisión costó tomarse y por qué?
¿qué habría pasado si el radio hubiera sido mayor?
y ¿qué habría pasado a las 03:00?

→ las dos últimas producen la mitad de las acciones

Y las reglas de las acciones:

dueño con nombre, no un equipo
plazo, y no «cuando se pueda»
criterio de cierre comprobable
y priorizadas por lo que habrían costado en un incidente
real

- y una comprobación al mes siguiente: ¿cuántas se cerraron?
- si menos del 70 %, el ejercicio siguiente no se hace hasta cerrarlas
 - porque repetir un ejercicio con las acciones anteriores abiertas encuentra lo mismo

Y la repetición, que es lo que lo convierte en programa:

EL MISMO EJERCICIO, 8-12 SEMANAS DESPUÉS
y las hipótesis ahora deben cumplirse
→ y si no, las acciones no funcionaron

y la cadencia que funciona		
ensayos por sistema	mensuales	clase 261
ensayo integrado	trimestral	
ensayo integrado con ausencias	semestral	
y conmutación de región completa	anual, con	acta
		clase 278

Y los errores que arruinan un ensayo integrado:

- 1 HACERLO SIN AVISAR
→ produce un incidente real con gente confundida y quema la confianza
- 2 QUE LO RESPONDAN LOS AUTORES
→ sale bien y no enseña nada
- 3 NO PARAR CUANDO HAY UN INCIDENTE REAL
- 4 USAR LOS HALLAZGOS PARA SEÑALAR PERSONAS
→ y al siguiente nadie participa de verdad
- 5 NO CERRAR LAS ACCIONES
- 6 Y HACERLO DEMASIADO FÁCIL
→ si todo sale perfecto, el ejercicio era pequeño
→ un ensayo que no descubre nada es un ensayo mal diseñado

Y la lista de comprobación de la clase:

- hay hipótesis escritas por escenario, con detección
- hay radio máximo y criterios de parada por escenario
- la restauración de cada inyección está probada
- está anunciado a todos los equipos y a atención al cliente
- responden los equipos de guardia, no los autores
- hay célula de control que no da pistas
- hay un observador por equipo que anota los silencios
- los escenarios se encadenan y cruzan equipos
- se prueba al menos una decisión que nadie puede tomar solo
- se ensaya la ausencia de la persona clave
- el ejercicio se detiene si hay un incidente real
- hay revisión en caliente el mismo día
- las acciones tienen dueño con nombre y plazo
- se repite en 8-12 semanas y las hipótesis se cumplen

Y el cierre que enlaza con la clase siguiente: el ensayo dice qué hace la organización cuando algo falla. Queda revisar los ocho capstones con un método estructurado, en los tres proveedores, y anotar los riesgos que quedan. La revisión bien diseñada multiproveedor es la materia de la clase 286.

Ejemplo trabajado

El ensayo integrado de CloudShop: cinco horas, cuatro escenarios encadenados, cuatro equipos. Lo que sigue es la línea de tiempo, la decisión que nadie tomó durante 31 minutos, y la clasificación de los 47 hallazgos.

El diseño.

duración prevista	5 horas, un jueves de 10:00 a 15:00
equipos participantes	plataforma · pedidos · pagos · datos
célula de control	3 personas
observadores	4, uno por equipo
anunciado a	todos los equipos, atención al cliente y dirección
no anunciado	el contenido de los escenarios
criterio de parada global	errores de usuario > 1 % durante 3 minutos, o cualquier participante lo declara

Y las hipótesis escritas antes:

- ESCENARIO 1 · una zona del proveedor principal se degrada
esperamos detección en < 3 min · incidente declarado en < 6 min · tráfico redistribuido · sin errores de usuario
- ESCENARIO 2 · la respuesta al 1 satura la base de pedidos
esperamos el equipo de pedidos detecta y conecta ambos sucesos en < 10 min
- ESCENARIO 3 · en paralelo, la pasarela de pagos de un mercado empieza a responder lento
esperamos se trata como incidente SEPARADO y no se confunde con el primero
- ESCENARIO 4 · las dos personas que más saben de conmutación no están disponibles
esperamos la conmutación se ejecuta igual, en < 25 min

La línea de tiempo.

10:00	se degrada la zona	
10:02	alerta de instancias sanas	✓ 2 min
10:04	incidente declarado; coordina el turno de plataforma	✓ 4 min
10:06	tráfico redistribuido; sin errores de usuario	✓
10:14	ESCENARIO 2: la base de pedidos al límite de conexiones	
10:16	alerta de latencia en pedidos	

10:16 el equipo de pedidos abre SU PROPIO incidente
→ sin saber que había otro abierto

10:29 alguien de plataforma ve el segundo incidente y
conecta los dos X 15 min
→ la hipótesis decía < 10

10:35 ESCENARIO 3: la pasarela del mercado B se degrada

10:37 alerta de pagos

10:38 el sistema de correlación agrupa la alerta de
pagos con el incidente principal
→ «probablemente derivado de la degradación de
zona»

10:38-11:12 nadie investiga pagos por separado

11:12 atención al cliente informa de reclamaciones de
pago del mercado B

11:14 se separa el incidente X 39 min

11:20 se plantea conmutar la zona degradada por completo

11:20-11:51 NADIE TOMA LA DECISIÓN X 31 min

11:51 la célula de control fuerza el escenario 4
(«las dos personas de conmutación no están»)

11:53 el turno de guardia asume la decisión y ejecuta

12:09 conmutación completada ✓ 16 min

12:20 primera comunicación a clientes del mercado B
→ 1 h 45 después del inicio de su problema X

13:40 ejercicio finalizado y restaurado

Los tres fallos grandes, analizados.

FALLO 1 · dos incidentes abiertos sin saberlo
el equipo de pedidos abrió su incidente a las 10:16
el de plataforma llevaba uno abierto desde las 10:04
→ 13 minutos de trabajo duplicado y de confusión

causa

no había forma de ver «incidentes abiertos ahora
mismo» en un solo sitio
→ cada equipo usaba su propio canal

acción

un canal único de incidentes activos, con quien
coordina cada uno
y comprobación obligatoria al declarar: «hay 1
incidente abierto; ¿el tuyo es el mismo?»

FALLO 2 · la correlación agrupó dos incidentes distintos
exactamente el riesgo que la clase 263 anticipó
→ y el equipo confió en la agrupación
→ 39 minutos de un problema de pagos sin atender

causa

el agrupamiento presentaba una conclusión, no una
hipótesis
y separar no era visible

acción

botón de separar, visible y en primer plano
el agrupamiento se presenta como «posible relación»
y la instrucción para quien coordina: «si la evidencia
no encaja en una sola historia, sepáralo»
clase 258

FALLO 3 · 31 minutos sin que nadie decidiera
la conmutación de zona afectaba a pedidos, pagos y
datos
→ cada equipo esperaba que decidiera otro
→ y el turno de guardia creía que no tenía autoridad

causa

el procedimiento decía «coordinar con los equipos
afectados»

→ y no decía quién decide si no hay acuerdo

acción

quien coordina el incidente DECIDE, y esa autoridad
está escrita y comunicada clase 257
lista explícita de decisiones que puede tomar sin
consultar
y las que sí requieren negocio, con quién y en cuánto

→ y el equipo comentó que este fallo era el más caro y el
más barato de arreglar: una frase en un documento

La clasificación de los 47 hallazgos.

ORGANIZATIVOS	31	66 %
autoridad para decidir no definida	6	
información que no estaba donde se buscó	9	
procedimientos incompletos o rotos	7	
comunicación: cuándo, a quién y qué	5	
y coordinación entre equipos	4	
TÉCNICOS	16	34 %
paneles sin la señal necesaria	5	
permisos insuficientes	3	
correlación mal presentada	2	
límites de conexión mal dimensionados	3	
y alertas mal enrutadas	3	

Y la comparación con la predicción:

la hipótesis de la parte 23 decía
«alrededor de dos tercios de los hallazgos volverán a
ser organizativos»

resultado medido	66 %
la clase 261, con ensayos por sistema	68 %

→ la predicción se cumple
→ y con una precisión que sorprendió al propio equipo
→ y refuerza lo que la parte 22 había concluido: el 11 %
del contenido que no se predijo –lo organizativo– es
donde se concentran los fallos

La repetición, 10 semanas después.

acciones cerradas antes de repetir 41 de 47 87 %

misimos cuatro escenarios, mismas hipótesis

	primera	repetición
detección del escenario 1	2 min	2 min
incidente declarado	4 min	3 min
conectar escenarios 1 y 2	15 min	4 min ✓
separar el incidente de pagos	39 min	6 min ✓
decisión de conmutar	31 min	2 min ✓
conmutación ejecutada	16 min	14 min
primera comunicación a clientes	1 h 45 min	11 min ✓
hallazgos nuevos	47	14
organizativos	31	8
técnicos	16	6

Y la observación del equipo sobre la repetición:

la decisión que costó 31 minutos costó 2
→ y lo único que cambió fue una frase escrita: «quien
coordina decide»

y los 14 hallazgos nuevos eran de otra clase
más finos, más profundos
→ el ejercicio había subido de nivel

El coste del ejercicio.

preparación	62 horas
ejecución (16 personas × 5 h)	80 horas

revisión y acciones	94 horas
repetición	108 horas
	■■■■■■■■■■
	344 horas

y la comparación que se llevó a dirección
incidente grave medio en CloudShop ~40 horas de
equipo
hallazgos clasificados como «habría causado o alargado
un incidente grave» 19 de 47

→ y si solo 9 de esos 19 se hubieran materializado, el
ejercicio se paga

La lección que este ensayo deja: los tres fallos mayores no fueron técnicos —dos incidentes abiertos sin que ninguno supiera del otro, una correlación que ocultó un problema de pagos durante 39 minutos, y 31 minutos en que nadie tomó una decisión porque nadie sabía que podía—. El más caro de los tres se arregló con una frase escrita, y en la repetición esos 31 minutos fueron 2.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/285-game-day-integrado-y-respuesta-a-incidentes/lab.py
```

El laboratorio selecciona el motor de práctica chaos 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 final-gameday 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 hipótesis de resiliencia y criterio de abortar. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

Reto verificable

Construye final-gameday 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 ensayo sale bien y no enseña casi nada	Lo respondieron los autores, o el escenario era demasiado sencillo	Que respondan los equipos de guardia reales y encadena escenarios que crucen equipos; un ejercicio que no descubre nada está mal diseñado.
Dos equipos trabajan en el mismo incidente sin saberlo	No existe una vista única de incidentes activos y cada equipo usa su canal	Canal único de incidentes con quien coordina cada uno, y comprobación obligatoria al declarar uno nuevo.
Un problema queda sin atender porque se agrupó con otro	La correlación presentó una conclusión y separar no era visible	Presenta el agrupamiento como posible relación, pon el botón de separar en primer plano y forma a quien coordina para separar cuando la evidencia no encaja.
Una decisión importante queda en el aire mientras todos esperan	El procedimiento pide coordinar y no dice quién decide si no hay acuerdo	Escribe que quien coordina decide, con la lista de decisiones que puede tomar sin consultar y las que requieren negocio y en cuánto tiempo.
El ejercicio genera hallazgos y nada cambia	Las acciones no tienen dueño con nombre, plazo ni criterio de cierre	Asigna, fija plazo y comprueba al mes; si no se cierra el 70 %, no repitas el ejercicio hasta cerrarlas.
Tras el ensayo hay recelo y menos participación	Los hallazgos se usaron para señalar personas o el ejercicio no se anunció	Anuncia siempre el ejercicio, atribuye los hallazgos al sistema y detén el ensayo de inmediato si aparece un incidente real.

■ 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 encuentra un ensayo integrado que no encuentra uno por sistema?
2. ¿Qué papel tiene la célula de control y qué no debe hacer?
3. ¿Por qué se encadenan los escenarios y con qué regla se introduce el siguiente?
4. ¿Qué seis dimensiones evalúa la rúbrica del ejercicio?
5. ¿Qué proporción de hallazgos suele ser organizativa y qué implica?

Referencias

- Google (2018). The Site Reliability Workbook — DiRT y ejercicios de continuidad.
<<https://sre.google/workbook/reliable-product-launches/>>
- Rosenthal, C. y Jones, N. (2020). Chaos Engineering: system resiliency in practice.
<<https://www.oreilly.com/library/view/chaos-engineering/9781492043850/>>
- AWS (2024). Fault Injection Service: multi-account and multi-region experiments.
<<https://docs.aws.amazon.com/fis/latest/userguide/what-is.html>>
- Microsoft (2024). Azure Chaos Studio experiment design.
<<https://learn.microsoft.com/azure/chaos-studio/chaos-studio-overview>>
- Allspaw, J. (2015). Trade-offs under pressure — coordinación en incidentes reales.
<<https://www.researchgate.net/publication/282869185>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Anterior	Índice	Siguiente
← 284 · Capstone datos e IA: plataforma gobernada	Parte 23 · Programa	286 · Revisión Well-Architected multi-proveedor →

Evaluación — 285 Game day integrado y respuesta a incidentes

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 final-gameday 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.

286 — Revisión Well-Architected multi-proveedor

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 4 Laboratorio: architecture · Estado: EXECUTABLECORE

Propósito

Revisar un sistema con método estructurado en cualquier proveedor: los cinco pilares comunes, cómo se ejecuta una revisión que produzca acciones en vez de un informe, y —lo más importante— las preguntas que este programa ha demostrado que faltan en los marcos oficiales, con las cifras que lo justifican.

Resultados de aprendizaje

Al finalizar podrás:

1. Ejecutar una revisión estructurada de un sistema en medio día.
2. Aplicar los cinco pilares comunes con preguntas comprobables.
3. Añadir las preguntas que los marcos oficiales no incluyen.
4. Priorizar los hallazgos por riesgo real y no por cantidad.
5. Convertir la revisión en acciones con dueño, plazo y comprobación.

Conceptos centrales

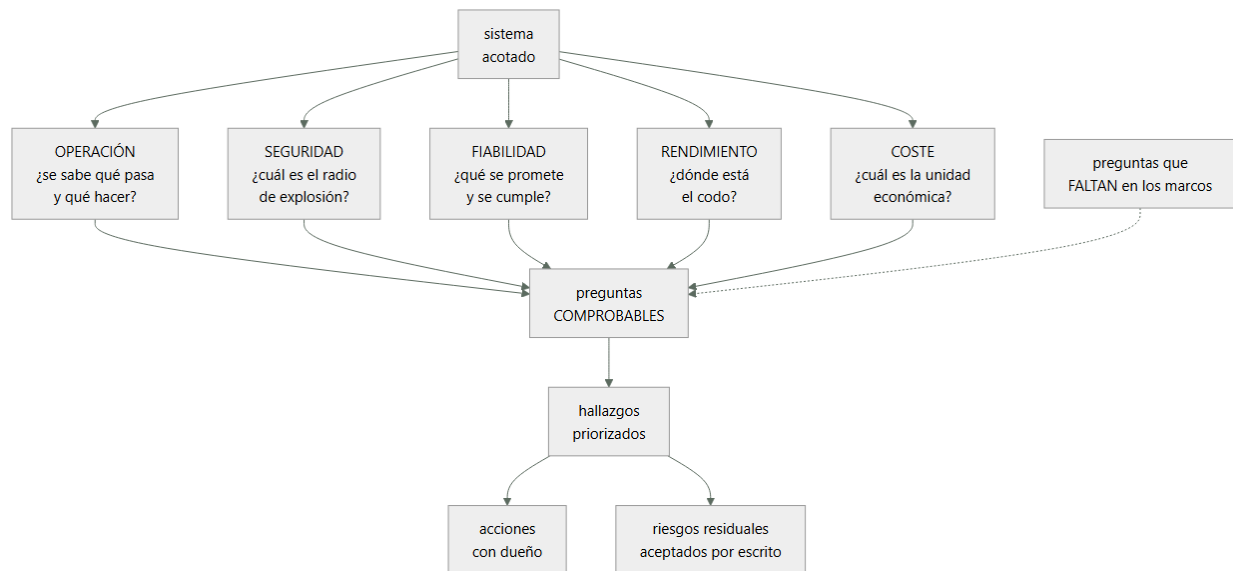
Concepto	Comprensión verificable
revisión estructurada	Recorrido guiado por preguntas sobre un sistema concreto, con participantes que lo conocen.
pilar	Dimensión de calidad que se revisa por separado: operación, seguridad, fiabilidad, rendimiento y coste.
pregunta comprobable	La que se responde con una evidencia, no con una opinión.
riesgo residual	El que se decide aceptar, con dueño, motivo y vigencia escritos.
hallazgo priorizado	El que se ordena por impacto y probabilidad reales, no por su gravedad nominal.
revisión que no cambia nada	Modo de fracaso: se genera un informe extenso y ninguna acción se cierra.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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. Cómo se ejecuta una revisión que sirve

Una revisión estructurada es media jornada, no un proyecto. Y su valor depende de tres decisiones de formato.

- 1 ACOTAR EL SISTEMA
un servicio o un recorrido completo, no «la plataforma»
→ «el flujo de compra, del navegador al almacén»
→ con sus dependencias dibujadas ley 24
- 2 LOS PARTICIPANTES SON QUIENES LO CONSTRUYEN Y LO OPERAN
→ no un comité externo
→ y quien facilita hace preguntas, no las responde
- 3 LAS PREGUNTAS SE RESPONDEN CON EVIDENCIA
«sí, tenemos copias» no vale
«restauramos el 14 de marzo en 52 minutos, aquí está el acta» sí
→ y esta regla sola cambia el resultado de la revisión entera

Y el formato que funciona:

media jornada, 4 horas
 0:00-0:20 el equipo presenta el sistema y su diagrama
 0:20-3:00 los cinco pilares, ~30 min cada uno
 3:00-3:30 priorización conjunta
 3:30-4:00 acciones, dueños y plazos

y lo que NO se hace

no se puntúa al equipo
 no se genera un informe de 60 páginas
 y no se revisa todo el catálogo de preguntas
 → 8-12 preguntas por pilar, las que apliquen

Y la priorización, que evita el modo de fracaso:

cada hallazgo con dos números
 IMPACTO si ocurre, ¿qué pasa?
 PROBABILIDAD ¿ha estado a punto de pasar?

y tres categorías

ACTUAR YA alto impacto, probable
 PLANIFICAR alto impacto, poco probable
 ACEPTAR y entonces se escribe como riesgo residual, con dueño y vigencia

→ y la tercera categoría es la que da credibilidad
 → una revisión donde todo hay que arreglarlo produce que

no se arregle nada

Y el criterio de éxito:

no «cuántos hallazgos»

sino CUÁNTAS ACCIONES SE CERRARON EN 90 DÍAS

→ y si es menos del 60 %, la revisión generó una lista de deseos, no un plan

2. Los cinco pilares, con preguntas comprobables

Las preguntas que valen son las que exigen enseñar algo.

OPERACIÓN

- enséñame el inventario de este sistema con su dueño
- ¿cuál fue la última alerta y qué hizo quien la recibió?
- coge un procedimiento y ejecútalo ahora
- ¿dónde está la línea de cambios de este sistema?
- ¿cuándo fue la última vuelta atrás?
- ¿cuánto tarda un cambio desde confirmar hasta producción?
- ¿cuántas interrupciones de guardia tuvo el último turno?

SEGURIDAD

- si esta credencial se filtra, ¿hasta dónde llega?
- ¿queda alguna credencial permanente?
- ¿quién puede borrar las copias?
- ¿quién puede desactivar el registro?
- haz algo que debería detectarse y cronométralo
- ¿los entornos de prueba tienen datos reales?
- ¿qué datos salen a terceros y con qué contrato?

FIABILIDAD

- ¿qué prometes al usuario y cómo lo mides?
- ¿ese objetivo se incumple alguna vez?
- restaura una copia ahora y cronométralo
- ¿qué pasa si cae una zona? ¿lo has probado?
- ¿qué dependencia te tumba si va LENTA en vez de fallar?
- ¿qué se degrada antes de caer y quién lo decidió?

RENDIMIENTO

- ¿qué recurso satura primero y dónde está su codo?
- si el tráfico se duplica, ¿qué se rompe y en cuánto?
- ¿escalar empeora algún limitante?
- ¿qué cuotas te limitan y a qué porcentaje estás?
- ¿la región secundaria tiene cuotas para el 100 %?

COSTE

- ¿cuál es tu unidad económica y su tendencia?
- ¿qué porcentaje del coste está atribuido?
- ¿qué parte del gasto es trabajo que no hace falta?
- ¿alguna decisión reciente llevó su coste estimado delante?

Y las preguntas transversales, que suelen dar los mejores hallazgos:

- ¿qué parte de este sistema no has probado nunca?
- ¿qué te quita el sueño de este sistema?
- si te fueras mañana, ¿qué no sabría nadie?
- y ¿qué decisión de hace dos años ya no tiene sentido?

→ la tercera es la que más deuda de conocimiento revela

→ y la cuarta, la que más simplificaciones produce

3. Las preguntas que faltan en los marcos

Los marcos oficiales de los proveedores son buenos y este programa ha encontrado, midiendo, huecos concretos.

Estas son las preguntas que hay que añadir, con lo que las justifica.

- 1 ¿CUÁNDO SE EJECUTÓ POR ÚLTIMA VEZ CADA PROCEDIMIENTO?
los marcos preguntan si existen procedimientos
→ 19 de 34 estaban rotos al ejecutarlos clase 259

→ existir y funcionar son cosas distintas ley 22

- 2 ¿QUÉ JUICIO APLICA LA PERSONA QUE HACE ESTO A MANO?
antes de automatizar algo
→ una remediación correcta convirtió una degradación en
caída total en 3 minutos por no saber cuándo NO
actuar ley 30
- 3 ¿CÓMO DETECTAS UN FALLO QUE NO DA ERROR?
los marcos preguntan por alertas de error
→ un fallo de datos da otro número ley 29
→ y 17 de 19 incidentes de datos los detectó una
persona, 23 días después clases 243, 284
- 4 ¿QUÉ CUOTAS TIENE LA REGIÓN DE CONMUTACIÓN?
los marcos preguntan por planes de continuidad
→ la región secundaria daba para el 3,2 % de la
capacidad prometida clase 262
- 5 ¿MIDES EN EL CLIENTE O EN EL SERVIDOR?
→ 1,23 puntos de diferencia en el éxito de compra,
unas 4.100 compras al mes invisibles clase 268
- 6 ¿QUÉ PASA SI UNA DEPENDENCIA VA LENTA EN VEZ DE
FALLAR?
→ 200 ms de latencia produjeron 4.100 ms de
degradación; el mismo servicio CAÍDO se comportaba
mejor clase 261
- 7 ¿CUÁNTOS EQUIPOS TOMAN BUENAS DECISIONES SIN
CONSULTARTE?
para funciones de apoyo
→ toda función de apoyo deriva hacia autorizar y se
queda ciega ley 31
- 8 ¿QUÉ HAS RETIRADO ESTE AÑO?
los marcos preguntan qué se ha añadido
→ 245 alertas de 412, 9 procedimientos, 426 conjuntos
de datos, 4 modelos, 61 paneles
→ y retirar fue, medido, de lo más rentable
- 9 ¿QUIÉN DECIDE CUANDO EL PROBLEMA CRUZA DOS EQUIPOS?
→ 31 minutos sin que nadie decidiera, arreglados con
una frase escrita clase 285
- 10 Y ¿QUÉ PARTE DE ESTE SISTEMA NO ESTÁ EN EL DIAGRAMA?
→ lo que no está en el diagrama no se analiza ley 24

Y el patrón que comparten las diez:

los marcos preguntan por lo que EXISTE
y estas preguntan por lo que se ha COMPROBADO

→ y la diferencia entre las dos cosas es donde han estado
casi todos los hallazgos caros de este programa

4. Revisar en tres proveedores

La revisión es la misma; lo que cambia es dónde se busca la evidencia.

LO QUE NO CAMBIA
las preguntas
los pilares
la exigencia de evidencia
y la priorización

LO QUE CAMBIA
el nombre de los servicios
dónde se consultan las cuotas
cómo se comprueban los permisos efectivos
qué informa el detalle de coste y con qué retraso
y qué hay activado por defecto ley 26

→ y por eso la rejilla de cinco capas de la clase 273 es

la herramienta de traducción de la revisión

Y los puntos donde una revisión multiproveedor encuentra más:

AISLAMIENTO

la unidad administrativa no es equivalente entre nubes
→ y una arquitectura copiada de una a otra suele quedar
peor aislada clase 219

PERMISOS

la semántica de herencia y denegación difiere
→ una política traducida puede conceder de más
clase 273
→ hay que comprobar permisos EFECTIVOS, no textos

CUOTAS

distintas por defecto y por región
→ y el inventario de cuotas es por nube y por cuenta

COSTE

distintos modelos y distinto retraso del detalle
→ la unidad económica debe ser comparable entre nubes
clase 270

Y OPERACIÓN

dos inventarios, dos catálogos de alertas, dos guardias
→ y el coste real de eso, medido: 2,5 personas y
492.000 USD al año en CloudShop clase 273

Y el cierre de la revisión, que es lo que la hace útil:

EL ACTA, en dos páginas

el sistema revisado y su alcance
los hallazgos priorizados, con impacto y probabilidad
las acciones con dueño y plazo
los riesgos residuales ACEPTADOS, con dueño y vigencia
y la fecha de la próxima revisión

→ y a los 90 días, una comprobación de cuántas acciones
se cerraron
→ que es el único número que dice si la revisión sirvió

Y la lista de comprobación de la clase:

- el sistema revisado está acotado y dibujado
- participan quienes lo construyen y lo operan
- cada respuesta va con evidencia enseñada
- se ejecuta un procedimiento durante la sesión
- se restaura una copia y se cronometra
- se hace algo que debería detectarse y se cronometra
- se preguntan las diez que faltan en los marcos
- los hallazgos se priorizan por impacto y probabilidad
- hay riesgos aceptados por escrito, con dueño y vigencia
- el acta cabe en dos páginas
- y a los 90 días se cuenta cuántas acciones se cerraron

Y el cierre que enlaza con la clase siguiente: la revisión produce hallazgos, acciones y riesgos aceptados. Reunir todo eso en un paquete que alguien de fuera pueda evaluar —con costes y con los riesgos que quedan— es la materia de la clase 287.

Ejemplo trabajado

Dos revisiones de CloudShop, la misma metodología en dos proveedores. Lo que sigue son los hallazgos que solo aparecen al exigir evidencia, la diferencia entre las dos nubes, y qué pasó a los 90 días.

Revisión 1 · El flujo de compra, proveedor principal.

alcance del navegador al almacén, incluidas pasarelas y eventos
duración 4 horas
participan 6 personas del equipo, 1 facilitando

Y lo que ocurrió al exigir evidencia:

PREGUNTA «coge un procedimiento y ejecútalo ahora»

RESPUESTA se eligió al azar «recuperar la cola de pedidos atascada»

RESULTADO falló en el paso 3: el nombre de la cola había cambiado en una migración de hacía 5 meses
→ hallazgo, en directo, en 6 minutos

PREGUNTA «restaure una copia ahora y cronométralo»

RESPUESTA «tarda unos 40 minutos»

RESULTADO 3 h 10 min
→ y el objetivo comprometido era de 1 hora
→ hallazgo de impacto alto

PREGUNTA «haz algo que debería detectarse y cronométralo»

RESPUESTA se creó una clave de acceso permanente

RESULTADO nadie la detectó en 45 minutos
→ la regla existía y apuntaba a un canal retirado ley 15

PREGUNTA «¿mides en el cliente o en el servidor?»

RESPUESTA en el servidor

RESULTADO al instrumentar el cliente: 1,23 puntos de diferencia
→ 4.100 compras al mes que nadie veía

PREGUNTA «¿qué pasa si el inventario va LENTO?»

RESPUESTA «se degradaría un poco»

RESULTADO ensayo posterior: 200 ms → 4.100 ms
→ hipótesis refutada

PREGUNTA «¿qué has retirado este año?»

RESPUESTA silencio, y después «nada»
→ y de ahí salió el trabajo de las 245 alertas

Y la priorización:

restauración 3x peor que el objetivo	alto	alto	YA
no se detecta credencial permanente	alto	medio	YA
degradación por dependencia lenta	alto	alto	YA
procedimiento roto	medio	alto	YA
medición solo en servidor	medio	alto	YA
cuotas de la región secundaria	alto	bajo	PLANIF
sin retirada de nada	medio	alto	PLANIF
registro de accesos incompleto	medio	bajo	ACEPTAR
sin cifrado propio en un almacén interno	bajo	bajo	ACEPTAR
hallazgos		19	
actuar ya		5	
planificar		9	
aceptar como riesgo residual		5	

Y los cinco riesgos aceptados, escritos:

cada uno con
qué es, qué pasaría, por qué se acepta, quién lo acepta
y hasta cuándo

ejemplo

«el registro de accesos al almacén interno no incluye lecturas. Impacto: no podríamos reconstruir quién consultó qué en caso de sospecha. Se acepta porque el almacén no contiene datos personales y añadirlo cuesta 6 semanas. Acepta: responsable de plataforma. Vigencia: hasta diciembre o hasta que el almacén reciba datos personales, lo que ocurra antes.»

→ y esa última cláusula es lo que evita que un riesgo aceptado se convierta en un riesgo olvidado

Revisión 2 · El mismo recorrido, segundo proveedor.

la arquitectura se había copiado del principal
las mismas preguntas, la misma exigencia de evidencia

hallazgos comunes	11
hallazgos EXCLUSIVOS del segundo proveedor	8

Y los ocho exclusivos:

- 1 la unidad de aislamiento no era equivalente: entornos que en el principal estaban en cuentas separadas aquí compartían contenedor administrativo clase 219
- 2 la política de permisos, traducida literalmente, concedía acceso a 214 conjuntos en vez de 1
→ detectado comparando permisos EFECTIVOS
clase 273
- 3 cuotas por defecto muy inferiores en 4 servicios
→ y sin alerta de proximidad
- 4 el registro de auditoría no estaba activado por defecto en 2 de 3 cuentas ley 26
- 5 la replicación de copias tenía un destino preconfigurado fuera de la región deseada
- 6 el detalle de coste llegaba con 26 horas de retraso frente a 8
→ y la alerta de anomalía se había calibrado con el retraso del otro proveedor
- 7 el balanceador no expulsaba atípicos con la misma configuración clase 196
- 8 y la herramienta de infraestructura como código se estrangulaba por cuota del plano de control
clase 262

Y la conclusión que el equipo escribió:

«copiamos la arquitectura y no copiamos las garantías»

- 8 de 19 hallazgos venían de suponer equivalencias
- y ninguno lo habría encontrado una revisión que preguntara «¿tenéis aislamiento por entorno?»
- los encontró preguntar «enséñame el aislamiento efectivo»

Los 90 días.

hallazgos	19	19
acciones asignadas	14	16
cerradas en 90 días	12	13
	86 %	81 %
riesgos residuales escritos	5	3
revisados al vencer la vigencia	5	3
de ellos, dejaron de aceptarse	2	1

y el efecto medido de las cerradas

restauración	3 h 10 → 52 min
detección de credencial permanente	no detectada → 4 min
degradación por latencia	4.100 ms → 1.240 ms
medición en cliente	no → sí
cuotas de la secundaria	3,2 % → 100 %
permisos efectivos del segundo proveedor	214 → 1 conjunto

Y la comparación con la revisión anterior, hecha un año antes con el formato clásico:

duración	3 días	4 horas
páginas del informe	61	2
hallazgos	94	19
acciones asignadas	12	14
cerradas en 90 días	3	12
	25 %	86 %
hallazgos encontrados EN LA SESIÓN, ejecutando algo	0	6

Y la lectura del equipo:

la revisión de 3 días produjo 94 hallazgos y 3 acciones
cerradas
la de 4 horas produjo 19 y 12

→ la diferencia no fue el rigor: fue exigir evidencia y
aceptar riesgos por escrito
→ una lista de 94 cosas por arreglar se ignora entera
→ y 5 riesgos aceptados con fecha se revisan

La lección que esta clase deja: la revisión de tres días con 94 hallazgos cerró 3 acciones en 90 días; la de cuatro horas con 19 hallazgos cerró 12, y seis de esos hallazgos aparecieron durante la propia sesión al ejecutar un procedimiento, restaurar una copia y provocar algo que debía detectarse. Y en el segundo proveedor, ocho de diecinueve hallazgos venían de haber copiado la arquitectura sin copiar las garantías.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/286-revision-well-architected-multi-proveedor/lab.py
```

El laboratorio selecciona el motor de práctica architecture 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 final-architecture-review 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 diagrama acompañado por decisiones y trade-offs. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

Reto verificable

Construye final-architecture-review 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 revisión produce decenas de hallazgos y no cambia nada	No se priorizó ni se aceptó ningún riesgo por escrito	Clasifica en actuar ya, planificar y aceptar; una lista donde todo hay que arreglarlo se ignora entera.
Todo se responde afirmativamente y luego falla en un incidente	Las preguntas se responden con opiniones en vez de con evidencia	Exige enseñar: ejecuta un procedimiento, restaura una copia y provoca algo que debería detectarse, todo durante la sesión.
Los marcos oficiales se cumplen y siguen apareciendo fallos caros	Preguntan por lo que existe y no por lo que se ha comprobado	Añade las diez preguntas que faltan: última ejecución, juicio implícito, fallos sin error, cuotas de la secundaria, medición en el cliente y qué se ha retirado.
Se copia una arquitectura a otro proveedor y queda peor	Se supusieron equivalentes la unidad de aislamiento, los permisos y las cuotas	Revisa el aislamiento y los permisos efectivos en la nube destino, no los textos ni los nombres de servicio.
Un riesgo aceptado hace dos años ya no debería aceptarse	Se aceptó sin vigencia ni condición de revisión	Escribe cada riesgo residual con dueño, motivo, vigencia y la condición que lo invalida.
La revisión dura días y agota al equipo	Se recorre todo el catálogo de preguntas en vez de las que aplican	Acota el sistema, usa entre ocho y doce preguntas por pilar y cierra en media jornada con acta de dos páginas.

■ 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 tres decisiones de formato determinan que una revisión sirva?
2. ¿Qué distingue una pregunta comprobable de una que no lo es?
3. ¿Cuáles son las preguntas que faltan en los marcos y qué las justifica?
4. ¿Dónde encuentra más hallazgos una revisión multiproveedor?
5. ¿Cuál es el único número que dice si la revisión funcionó?

Referencias

- AWS (2024). Well-Architected Framework and Reviews. <<https://docs.aws.amazon.com/wellarchitected/latest/framework/welcome.html>>
- Microsoft (2024). Azure Well-Architected Framework. <<https://learn.microsoft.com/azure/well-architected/>>
- Google Cloud (2024). Architecture Framework. <<https://cloud.google.com/architecture/framework>>
- Kim, G. y otros (2016). The DevOps Handbook — revisiones que producen cambio. <<https://itrevolution.com/product/the-devops-handbook-second-edition/>>
- Beyer, B. y otros (2018). The Site Reliability Workbook, cap. «Reliability reviews». <<https://sre.google/workbook/table-of-contents/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 23 en PDF · Manual integral

Anterior	Índice	Siguiente
← 285 · Game day integrado y respuesta a incidentes	Parte 23 · Programa	287 · Paquete de evidencia, costos y riesgos residuales →

Evaluación — 286 Revisión Well-Architected multi-proveedor

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 final-architecture-review 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.

287 — Paquete de evidencia, costos y riesgos residuales

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 4 Laboratorio: assessment · Estado: EXECUTABLECORE

Propósito

Reunir el trabajo en un paquete de evidencia que alguien de fuera pueda evaluar sin explicaciones: qué se construyó, qué se comprobó, cuánto cuesta y qué riesgos quedan. La clase da la estructura del paquete, cómo se presentan costes que resistan una revisión financiera, y cómo se declaran los riesgos residuales de forma que sean creíbles y revisables.

Resultados de aprendizaje

Al finalizar podrás:

1. Estructurar un paquete de evidencia que se evalúe sin acompañamiento.
2. Presentar costes con base, supuestos y rango, no con una cifra.
3. Declarar riesgos residuales con dueño, vigencia y condición de revisión.
4. Distinguir lo comprobado de lo supuesto en cada afirmación.
5. Adaptar el mismo paquete a ingeniería, a negocio y a auditoría.

Conceptos centrales

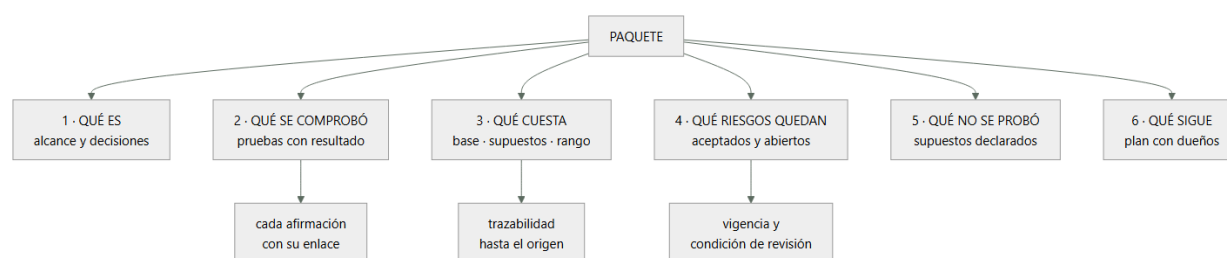
Concepto	Comprensión verificable
paquete de evidencia	Conjunto de documentos que permite a un tercero evaluar un sistema sin que nadie se lo explique.
afirmación con respaldo	La que enlaza a la prueba que la sostiene. Sin enlace, es una conjetura.
riesgo residual	El que se acepta conscientemente, con dueño, motivo, vigencia y condición de revisión.
supuesto declarado	Lo que se da por cierto sin haberlo comprobado. Se escribe, no se esconde.
coste con rango	Estimación con banda y con los factores que la mueven, en vez de un número único.
trazabilidad	Que cada cifra del paquete se pueda seguir hasta su origen.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 estructura del paquete

El criterio: alguien competente que no ha participado debe poder evaluarlo sin preguntar nada.

- 1 QUÉ ES
el alcance, en una página
el diagrama por recorrido, no por servicio
y las decisiones importantes con sus alternativas
→ y las decisiones van aquí, no en un anexo
clase 272
- 2 QUÉ SE COMPROBÓ
cada prueba negativa con su resultado y su fecha
→ incluidas las que fallaron y qué se hizo
→ y esta sección es la que da credibilidad al resto
- 3 QUÉ CUESTA
coste actual, coste proyectado, unidad económica
y los supuestos que lo mueven
- 4 QUÉ RIESGOS QUEDAN
aceptados, con dueño y vigencia
y abiertos, con plan
- 5 QUÉ NO SE PROBÓ
los supuestos declarados
→ y esta sección, que casi nadie escribe, es la que
más confianza genera
- 6 QUÉ SIGUE
plan con dueños, plazos y criterios de cierre

Y las reglas que hacen creíble el paquete:

- CADA AFIRMACIÓN CON SU RESPALDO
«la restauración tarda 52 minutos» → enlace al acta del
14 de marzo
→ sin enlace, se marca como estimación
- SE DISTINGUE COMPROBADO DE SUPUESTO
y se marca visualmente
→ «comprobado el 14/3» frente a «estimado a partir de
la prueba de carga de febrero»
- LO QUE FALLÓ SE CUENTA
→ un paquete donde todo salió bien no se cree, y con
razón
→ y este programa lleva 288 clases demostrándolo
- Y CABE EN LO QUE SE PUEDE LEER
20-30 páginas, con anexos aparte
→ un paquete de 200 páginas no se evalúa: se hojea

2. Costes que resisten una revisión

La sección donde más paquetes pierden credibilidad, por presentar certeza donde hay estimación.

- LO QUE NO FUNCIONA
«costará 41.000 USD al mes»
→ un número sin base ni supuestos
→ y la primera pregunta de finanzas lo desmonta
- LO QUE FUNCIONA
COSTE ACTUAL, medido y trazable
por componente, con la fuente
COSTE PROYECTADO, con rango
«entre 34.000 y 58.000 según el volumen; el factor
dominante es el tráfico entre zonas, que representa
el 31 %»
SUPUESTOS EXPLÍCITOS
volumen, crecimiento, retención, número de clientes
SENSIBILIDAD

«si el volumen se duplica, el coste sube un 60 %, no un 100 %, porque el almacenamiento no crece igual»

UNIDAD ECONÓMICA

coste por pedido, por cliente o por hora vista
clase 270

Y COSTE EVITADO

las alternativas descartadas y lo que habrían costado

Y los errores que se detectan de inmediato en una revisión financiera:

- 1 no incluir el coste de las personas
→ la operación de un sistema cuesta personas, y eso se cuenta
ley 23
- 2 no incluir transferencia de datos
→ suele ser el componente que más sorprende
- 3 usar precios de lista sin descuentos aplicables
o al revés, contar descuentos que no están firmados
- 4 ignorar el coste de los entornos que no son producción
→ suelen ser entre el 20 % y el 40 %
- 5 no decir qué pasa si los supuestos fallan
- 6 y presentar el ahorro sin la base
→ «ahorramos un 30 %» sin decir de qué ni si el sistema creció
clase 270

Y el formato que funciona con negocio:

una tabla con tres columnas
qué se consigue · qué cuesta · qué riesgo se evita o se asume

y la comparación con la alternativa
→ siempre, incluso con la de no hacer nada
→ porque decidir sin alternativa no es decidir
clase 272

3. Riesgos residuales y supuestos

La sección que distingue un paquete profesional de una presentación comercial.

UN RIESGO RESIDUAL BIEN ESCRITO

QUÉ ES en una frase, sin jerga
QUÉ PASARÍA impacto concreto, con cifra si se puede
PROBABILIDAD y en qué se basa esa apreciación
POR QUÉ SE ACEPTA coste o plazo de mitigarlo
QUIÉN LO ACEPTA nombre, no un equipo
VIGENCIA hasta cuándo
Y CONDICIÓN DE REVISIÓN
qué cambio lo invalida

→ las dos últimas son las que evitan que un riesgo aceptado se convierta en un riesgo olvidado
→ y son las que casi nunca se escriben

Y la sección de supuestos, que genera más confianza que ninguna otra:

QUÉ VA AQUÍ

lo que damos por cierto y no hemos comprobado
las pruebas que no hemos hecho y por qué
las cifras estimadas y su origen
y las dependencias de terceros que no controlamos

EJEMPLOS REALES DE ESTE PROGRAMA

«la conmutación de región se ha ensayado con 4 instancias, no con carga completa; es nuestro mayor riesgo abierto»
clase 276
«la proyección de temporada alta asume un crecimiento del 34 %; el año pasado la campaña la superó por 1,88 veces y el plan aguantó por el vertido de carga, no por la proyección»
clase 262
«suponemos que el proveedor de pagos mantiene su tasa de aprobación; no tenemos forma de verificarlo»

→ quien escribe esto se hace más creíble, no menos
→ y quien evalúa deja de buscar lo que se esconde

Y las tres audiencias, con el mismo contenido:

INGENIERÍA

mecanismos, compromisos y pruebas
→ y las secciones 2 y 5 son las que leen

NEGOCIO

qué se consigue, qué cuesta, qué riesgo se asume
→ y la tabla de tres columnas
→ y la unidad económica, no la factura

AUDITORÍA O CLIENTE

evidencia con fecha, trazabilidad y riesgos declarados
→ y la sección 2 con enlaces a las actas

→ el mismo paquete, distinto punto de entrada
→ y las cifras deben coincidir en las tres lecturas
→ si no coinciden, alguien lo notará

4. Errores que descartan un paquete

Lo que hace que un evaluador deje de confiar, en orden de gravedad.

- 1 UNA CIFRA QUE NO CUADRA CON OTRA DEL MISMO PAQUETE
→ y a partir de ahí se desconfía de todas
→ por eso la trazabilidad importa: cada cifra con su origen
- 2 NINGÚN FALLO
→ nadie construye algo sin que falle nada
→ y el evaluador asume que se ha ocultado
- 3 CERTEZA DONDE HAY ESTIMACIÓN
«costará exactamente X» o «la disponibilidad será del 99,99 %»
→ sin decir con qué base
- 4 NO DECIR LO QUE EMPEORA
→ el error 2 de la clase 272, en formato escrito
- 5 RIESGOS SIN DUEÑO
«se mitigará en el futuro»
→ sin nombre ni fecha, no es un plan
- 6 Y EVIDENCIA SIN FECHA
«la restauración tarda 52 minutos»
→ ¿medido cuándo? si fue hace 14 meses, no vale
ley 22

Y lo que un evaluador experimentado comprueba primero:

busca la sección de lo que no funcionó
→ si no existe, ya sabe qué clase de documento es
coge una cifra al azar y pide su origen
mira la fecha de la evidencia más antigua
busca un riesgo aceptado y comprueba si tiene vigencia
y pregunta por lo que NO está en el diagrama ley 24

→ cinco comprobaciones en diez minutos
→ y deciden cómo se lee el resto

Y la lista de comprobación de la clase:

- el paquete se puede evaluar sin que nadie lo explique
- cabe en 20-30 páginas, con anexos aparte
- las decisiones están en el cuerpo, no en un anexo
- cada afirmación enlaza a su respaldo
- se distingue lo comprobado de lo estimado, con fecha
- hay una sección de lo que falló y qué se hizo
- hay una sección de lo que NO se probó
- los costes llevan base, supuestos, rango y sensibilidad
- se incluyen personas, transferencia y entornos no productivos
- hay unidad económica y coste evitado
- cada riesgo aceptado tiene dueño, vigencia y condición

- las cifras cuadran entre secciones y son trazables
- y el plan tiene dueños con nombre y criterios de cierre

Y el cierre que enlaza con la clase siguiente: con el paquete escrito, queda defenderlo y cerrar el programa: corregir la última hipótesis, revisar las leyes acumuladas y escribir el plan de los doce meses siguientes. Es la clase 288, y cierra las 288.

Ejemplo trabajado

El paquete de evidencia de CloudShop, presentado a un cliente corporativo en una evaluación de proveedor. Lo que sigue es su estructura real, la sección que el evaluador leyó primero, y las dos cifras que no cuadraban.

La estructura entregada.

páginas del cuerpo	26
anexos	9 documentos
actas de pruebas enlazadas	31
1 Qué es	4 pág
alcance, diagrama por recorrido y 11 decisiones con alternativas	
2 Qué se comprobó	9 pág
47 pruebas negativas con resultado y fecha	
3 Qué cuesta	5 pág
4 Qué riesgos quedan	4 pág
5 Qué no se probó	2 pág
6 Qué sigue	2 pág

Y lo que el evaluador hizo en los primeros diez minutos:

- buscó la sección 5 («qué no se probó»)
 - existía; leyó los 6 supuestos declarados
 - comentario posterior: «esa sección decidió que me tomara en serio el resto»
- cogió una cifra al azar
 - «restauración de la base de pedidos: 52 minutos»
 - pidió el origen
 - acta del 14 de marzo, con hora de inicio, hora de fin, volumen restaurado y quién lo ejecutó
- miró la fecha de la evidencia más antigua
 - 11 meses (el ensayo de conmutación de región)
 - y estaba marcada como tal, con la fecha del siguiente ensayo ya fijada
- buscó un riesgo aceptado
 - los 7 tenían dueño con nombre, vigencia y condición
- preguntó qué no estaba en el diagrama
 - y aquí encontró algo

Lo que el evaluador encontró.

HALLAZGO 1 · una dependencia fuera del diagrama
el servicio de verificación de dirección postal, de un tercero
→ no aparecía en el diagrama por recorrido
→ y estaba en el camino crítico de la compra

la pregunta «¿qué pasa si no responde?»
la respuesta «se degrada; el pedido se acepta sin verificar»
la repregunta «¿lo habéis probado?»
la respuesta «no»

→ se añadió al diagrama, se probó dos días después
→ resultado: con el servicio lento (no caído), el flujo de compra se degradaba a 2.900 ms ley 24
→ y pasó a la sección 5 mientras se corregía

HALLAZGO 2 · dos cifras que no cuadraban
sección 3: «coste mensual 227.000 USD»

sección 3, tabla por componente: la suma daba 219.400

la diferencia: 7.600 USD del entorno de ensayo, incluido en el total y no desglosado

→ error de presentación, no de fondo

→ y aun así, el evaluador anotó: «revisar el resto de cifras con más cuidado»

→ y eso es exactamente lo que cuesta una cifra que no cuadra

La sección de costes, tal como se presentó.

COSTE ACTUAL, medido, últimos 3 meses

componente	USD/mes	fuelle
cóputo	71.400	facturación por etiqueta
base de datos	48.200	ídem
almacenamiento	19.100	ídem
transferencia de datos	31.900	ídem
distribución de contenido	14.600	ídem
servicios gestionados	22.800	ídem
entornos no productivos	11.400	ídem
observabilidad	7.600	ídem
	■■■■■■■■	
infraestructura	227.000	
personas de plataforma (4)	41.000	coste interno
	■■■■■■■■	
total	268.000	

Y la proyección con rango:

supuestos declarados	
crecimiento de pedidos	+34 % anual
crecimiento de catálogo	+12 % anual
retención de datos	sin cambios
y ningún mercado nuevo	

proyección a 12 meses

banda baja	291.000 USD/mes
banda central	318.000
banda alta	374.000

factores que la mueven, por orden

1 transferencia entre zonas	31 % de la variación
2 volumen de pedidos	27 %
3 retención de registros	19 %
4 el resto	23 %

sensibilidad

si los pedidos se duplican, el coste sube un 61 %, no un 100 %: cóputo y transferencia escalan, almacenamiento y servicios gestionados casi no

unidad económica

coste por pedido 0,41 → 0,17 USD en 24 meses
con pedidos creciendo un 34 % anual

coste evitado registrado, 14 meses

16 decisiones con la alternativa descartada
total anualizado de las alternativas 2.139.000 USD

Y el comentario del evaluador sobre esta sección:

«es la primera vez en esta evaluación que veo una sensibilidad y una banda. Todos los demás me han dado un número.»

→ y pidió el detalle del punto 3 (retención de registros)
→ que llevó a una conversación técnica de 40 minutos
→ y a una acción: bajar la retención en caliente de 90 a 30 días, -19.400 USD/mes

Los siete riesgos residuales.

ejemplo del que más se discutió

QUÉ ES

la conmutación de región se ha ensayado con 4 instancias; nunca con carga completa

QUÉ PASARÍA

en una caída real de región, el tiempo de conmutación podría superar el objetivo de 30 minutos; estimamos entre 30 y 90

PROBABILIDAD

caída de región completa: no ha ocurrido en 4 años en esta región según el histórico del proveedor

POR QUÉ SE ACEPTA

el ensayo con carga completa requiere duplicar capacidad durante 6 horas: 14.000 USD y una ventana coordinada con negocio

QUIÉN LO ACEPTA

el responsable de plataforma, con conocimiento del director de tecnología

VIGENCIA

hasta el 30 de noviembre

CONDICIÓN DE REVISIÓN

antes de la temporada alta, o si un cliente exige contractualmente el objetivo de 30 minutos

- el cliente exigía ese objetivo contractualmente
- la condición se activó en la propia reunión
- el ensayo se hizo 5 semanas después: 22 minutos con carga completa

Y lo que el cliente dijo al respecto:

«que lo tuvierais escrito como riesgo abierto, con el coste de cerrarlo y la condición exacta que lo activa, es lo que hizo que confiara en el resto del documento. Si me hubierais dicho que la conmutación estaba probada, lo habría comprobado y os habría pillado.»

Las tres versiones del paquete.

el mismo contenido, tres puntos de entrada

INGENIERÍA

entra por la sección 2 (qué se comprobó)
lee las 47 pruebas y los 6 supuestos
→ 45 minutos de lectura

NEGOCIO

entra por una tabla de 1 página
qué se consigue · qué cuesta · qué riesgo
→ 10 minutos
→ y la unidad económica, no la factura

AUDITORÍA

entra por el índice de actas con fechas
→ y de ahí a las pruebas que le interesan

y la comprobación que se hizo antes de entregar
las cifras de las tres vistas se cotejaron entre sí
→ y ahí se encontró la discrepancia de 7.600 USD
→ tarde: ya estaba impreso

El resultado.

evaluación de proveedor, 4 candidatos
CloudShop quedó primero

y lo que el comité escribió
«único candidato que declaró lo que no había probado»
«único que presentó rango y sensibilidad de coste»
«los riesgos aceptados tenían fecha de caducidad»
y «una discrepancia menor en la tabla de costes, ya aclarada»

- los tres primeros puntos eran secciones que los otros candidatos no tenían
- y el cuarto costó una explicación que no debería haber hecho falta

La lección que esta clase deja: el evaluador leyó primero la sección de lo que no se había probado, y fue la que le hizo tomarse en serio el resto; el riesgo declarado con su coste de cierre y su condición de activación fue lo que generó confianza, no la lista de logros. Y una discrepancia de 7.600 USD entre el total y la suma de componentes —un error de presentación, no de fondo— bastó para que el evaluador anotara que revisaría el resto de cifras con más cuidado.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/287-paquete-de-evidencia-costos-y-riesgos-residuales/lab.py
```

El laboratorio selecciona el motor de práctica assessment 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 final-evidence-pack 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 evaluación por escenarios con rúbrica y evidencia trazable. Además, la entrega debe incluir el comando ejecutado, la salida estructurada y una conclusión que no exceda lo observado.

Reto verificable

Construye final-evidence-pack 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 paquete necesita que alguien lo explique para entenderse	Está escrito para quien ya conoce el sistema	Escríbelo para un evaluador competente que no participó: alcance, diagrama por recorrido y decisiones en el cuerpo, no en anexos.
El evaluador desconfía de todas las cifras tras encontrar una	Una cifra no cuadra con otra del mismo documento	Coteja las cifras entre secciones y vistas antes de entregar, y haz que cada una sea trazable hasta su origen.
La sección de costes se desmonta con la primera pregunta	Se presentó un número único sin base, supuestos ni rango	Da coste actual medido, proyección con banda, supuestos, factores dominantes y sensibilidad; incluye personas, transferencia y entornos no productivos.
El paquete parece demasiado bueno y genera desconfianza	No cuenta ningún fallo ni ningún supuesto sin comprobar	Incluye una sección de lo que falló y otra de lo que no se probó; ambas aumentan la credibilidad del resto.
Un riesgo aceptado hace tiempo ya no debería aceptarse y nadie lo revisó	Se aceptó sin vigencia ni condición de revisión	Cada riesgo residual con dueño con nombre, vigencia y la condición concreta que lo invalida.
La evidencia existe pero no convence	No tiene fecha, o la más reciente es de hace más de un año	Fecha cada prueba y marca la evidencia antigua como tal, con la fecha de la siguiente ejecución ya fijada.

■ 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 seis secciones tiene el paquete y cuál genera más confianza?
2. ¿Qué hace que una sección de costes resista una revisión financiera?
3. ¿Qué siete elementos tiene un riesgo residual bien escrito?
4. ¿Qué comprueba primero un evaluador experimentado?
5. ¿Por qué contar lo que falló aumenta la credibilidad del paquete?

Referencias

- AWS (2024). Well-Architected Framework: documenting review outcomes.
<<https://docs.aws.amazon.com/wellarchitected/latest/framework/welcome.html>>
- Nygard, M. (2011). Documenting architecture decisions.
<<https://cognitect.com/blog/2011/11/15/documenting-architecture-decisions>>
- ISO/IEC (2022). ISO 31000: gestión del riesgo — riesgo residual y aceptación.
<<https://www.iso.org/iso-31000-risk-management.html>>
- FinOps Foundation (2024). Forecasting and unit economics.
<<https://www.finops.org/framework/capabilities/forecasting/>>
- Tufte, E. (2006). Beautiful Evidence — presentación de evidencia y trazabilidad.
<<https://www.edwardtufte.com/tufte/booksbe>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Anterior	Índice	Siguiente
← 286 · Revisión Well-Architected multi-proveedor	Parte 23 · Programa	288 · Defensa final, retrospectiva y plan de 12 meses →

Evaluación — 287 Paquete de evidencia, costos y riesgos residuales

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 final-evidence-pack 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.

288 — Defensa final, retrospectiva y plan de 12 meses

Parte: 23 — Capstones por industria y defensa final Nivel: experto · Horas estimadas: 8 Laboratorio: capstone · Estado: EXECUTABLECORE

Propósito

Cierre del programa. La clase da el formato de la defensa final, corrige las cinco predicciones de la clase 276, cierra el recuento de las leyes con la número 32, y deja lo único que importa a partir de aquí: un plan de doce meses que convierta 288 clases en capacidad demostrable, con hipótesis escritas y con fecha de revisión.

Resultados de aprendizaje

Al finalizar podrás:

1. Defender el trabajo completo ante un panel, con evidencia y compromisos.
2. Corregir las cinco predicciones de la clase 276 con evidencia medida.
3. Cerrar el recuento de leyes y entender qué las hace útiles.
4. Escribir un plan de doce meses con hipótesis y criterios de revisión.
5. Continuar el método del programa por cuenta propia.

Conceptos centrales

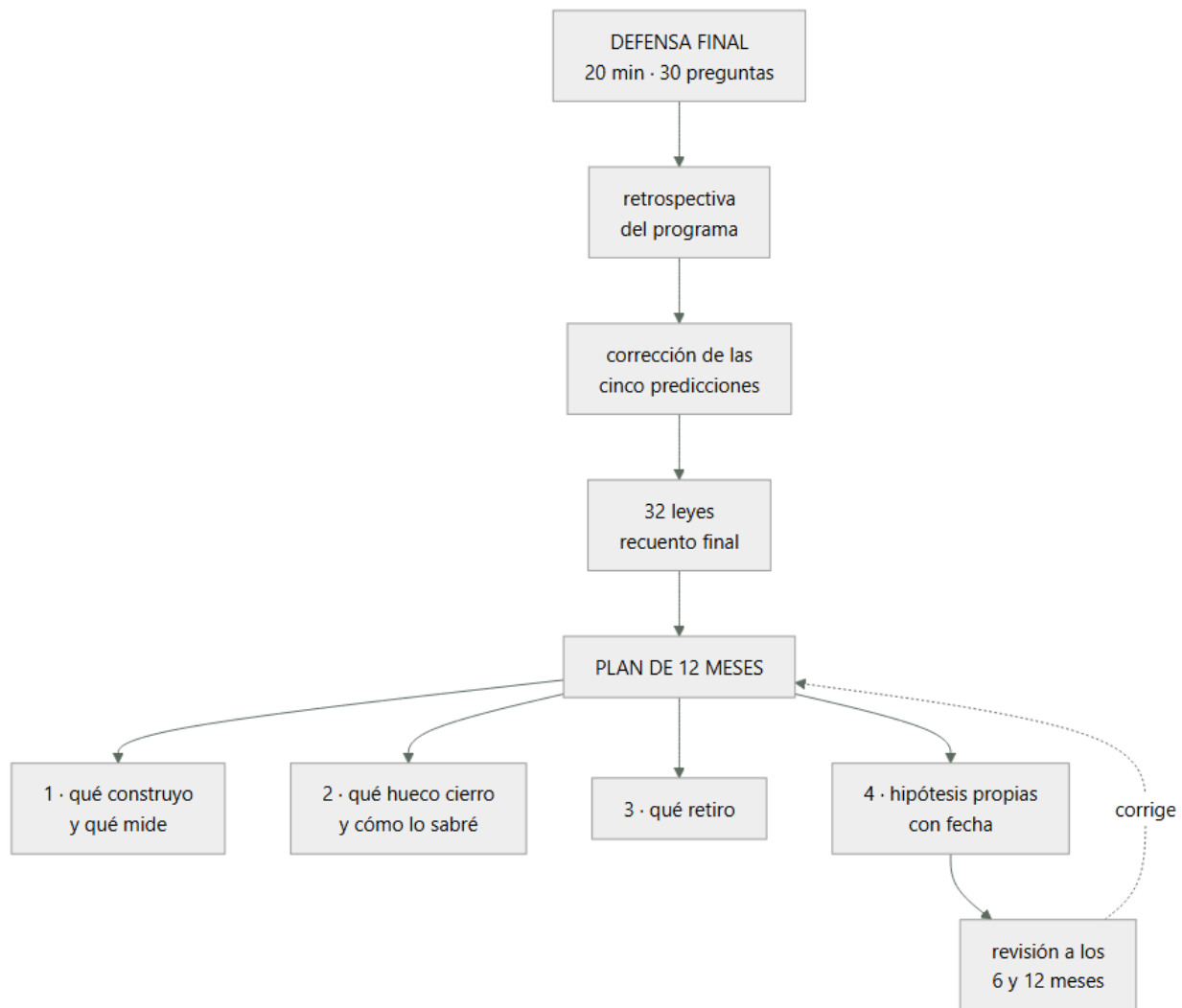
Concepto	Comprensión verificable
defensa final	Exposición del trabajo completo ante un panel que pregunta por lo que no se probó.
ley 32	Manda la restricción que el dinero no puede levantar; todas las demás son decisiones de coste.
plan de doce meses	Compromiso con hipótesis, medidas y fechas de revisión, no una lista de temas.
capacidad demostrable	Lo que se puede enseñar con efecto, mecanismo y cifra.
hipótesis propia	Afirmación refutable sobre el propio trabajo, escrita antes y corregida después.
método del programa	Predecir, medir, corregir en público y escribir la ley solo cuando la evidencia obliga.

Modelo mental

El capstone no premia cantidad de servicios, sino trazabilidad entre contexto, decisiones, implementación, fallos, evidencia y aprendizaje.

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 final

El encargo. Presentar el trabajo completo —los ocho capstones, el ensayo integrado, la revisión y el paquete de evidencia— ante un panel de tres personas, con el formato de la clase 276 ampliado.

EL FORMATO

- 20 minutos de exposición
- 30 minutos de preguntas
- 10 minutos de cierre: qué cambiarías y qué harás

LA EXPOSICIÓN, con la estructura que ya se conoce
 qué se construyó y para qué clase 287
 qué se comprobó, incluido lo que falló
 qué cuesta, con supuestos y rango
 qué riesgos quedan, con dueño y vigencia
 y qué no se probó

→ y el orden importa: lo que falló va en el minuto 8, no al final

Y las preguntas que el panel debe hacer, que son las de todo el programa:

- «¿qué parte de esto no has probado nunca?»
- «enséñame la última vez que ejecutaste ese procedimiento»
- «¿cómo detectarías un fallo que no da error?»
- «¿qué pasa si esa dependencia va LENTA en vez de fallar?»
- «¿quién decide cuando el problema cruza dos equipos?»
- «¿qué has retirado?»

«¿cuál es la alternativa más barata y por qué no la elegiste?»
«¿qué te haría cambiar de opinión?»
y «si te fueras mañana, ¿qué no sabría nadie?»

→ y ninguna de ellas se responde con conocimiento de
catálogo clase 274

Y la rúbrica, que suma a la de la clase 276 dos criterios:

acotación	10
alternativas	15
criterio	15
compromisos	15
respuesta a las preguntas	15
revisión	10
EVIDENCIA: ¿está fechada y enseñada?	10
Y LO QUE NO SE PROBÓ: ¿está declarado?	10

→ y los dos últimos son los que este programa ha
demostrado que más separan a quien sabe de quien lo
parece

2. Corrección de las cinco predicciones de la clase 276

La corrección, con evidencia y sin adornos. La última.

1. «los ocho sectores diferirán mucho menos en tecnología de lo que aparentan: lo que cambiará será QUÉ RESTRICCIÓN MANDA y qué se acepta perder»

CORRECTA Y MEDIDA. Los ocho capstones usaron las mismas piezas: identidad, aislamiento, colas, coherencia donde hace falta, señales, copias y despliegue progresivo. Lo que cambió fue el orden de prioridad y la lista de lo que se acepta perder. El comercio acepta sobreventa del 0,3 % y compensa; pagos no acepta un cobro duplicado; salud no acepta una filtración; el sector público no acepta que el ciudadano no pueda presentar en plazo. La misma caja de herramientas, ocho listas de sacrificios distintas.

2. «en todos ellos, la restricción dominante será NO TÉCNICA: normativa, contrato o coste unitario»

FALLIDA POR DECIR «EN TODOS». Se cumplió en siete de ocho: pico estacional, obligación de prueba ante terceros, protección de la persona, soberanía, coste por byte, aislamiento entre clientes y fiabilidad del dato. La excepción es el capstone industrial, donde la restricción dominante es FÍSICA: la conexión no es fiable y no hay contrato que lo cambie. Y esa excepción enseña más que los siete aciertos, porque señala qué tienen en común todas: se explica en la ley 32.

3. «el capstone que más fallos silenciosos revele será el de datos e inteligencia artificial»

CORRECTA EN EL ORDEN Y EQUIVOCADA EN LA EXCLUSIVIDAD. Si fue el que más reveló. Pero los fallos silenciosos aparecieron en los ocho: dos años de datos de producción inflados hasta un 19 % en el industrial, un informe que enseñó ventas de la competencia durante 19 meses en el multiinquilino, dos vías de datos personales fuera de Europa durante más de un año en el comercio, y 19 cobros no registrados en pagos. La ley 29 no es una ley del sector de datos: es una ley de los sistemas que producen números.

4. «en el ensayo integrado, alrededor de dos tercios de los hallazgos volverán a ser organizativos»

CORRECTA CON UNA PRECISIÓN INCÓMODA. 66 %, frente al 68 % de los ensayos por sistema de la clase 261. Y los

tres fallos mayores del ensayo integrado fueron: dos incidentes abiertos sin que ninguno supiera del otro, una correlación que ocultó un problema 39 minutos, y 31 minutos en que nadie decidió porque nadie sabía que podía. El más caro se arregló con una frase escrita.

5. «más del 70 % de las decisiones de los ocho capstones se podrán justificar con leyes ya escritas; y las que no, se concentrarán en sector público y salud»

CORRECTA EN LA CIFRA Y PARCIAL EN LA CONCENTRACIÓN. De 147 decisiones documentadas, 121 se justifican con leyes previas: el 82 %. De las 26 restantes, 11 están en sector público y salud, como predijimos, pero 7 están en medios –derechos por territorio, retirada de contenido, calidad percibida– que no habíamos anticipado. Acertamos la cifra y el motivo de la mitad.

Marcador final: dos correctas, dos parciales y una fallida. Y el fallo de la segunda es el que cierra el programa: acertamos que la restricción dominante casi nunca es técnica y nos equivocamos al decir que nunca lo es; lo que comparten todas no es su naturaleza, sino que ninguna se puede comprar.

3. El recuento final y la ley 32

Las 32 leyes, cerradas las 288 clases.

ley 13	lo que no se mira deja de funcionar en silencio	74
ley 15	la señal existe y nadie la mira	63
ley 22	un procedimiento nunca ejecutado no funciona	61
ley 16	un control que estorba se rodea	48
ley 14	el coste se decide al crear, no al pagar	46
ley 20	lo que no tiene dueño se filtra y se desperdicia	41
ley 25	lo provisional sobrevive a su motivo	38
ley 21	el acoplamiento vive en quién escribe	37
ley 29	un fallo de datos no da error: da otro número	31
ley 26	el valor por defecto sirve a la demostración	31
ley 23	la capacidad la limita lo que ya se mantiene	28
ley 17	se optimiza la medida, no el objetivo	24
ley 24	lo que no está en el diagrama no se analiza	23
ley 30	automatizar elimina un juicio que nadie escribió	17
ley 27	un control solo actúa sobre lo que cambia	15
ley 18	lo asíncrono traslada la garantía, no la elimina	14
ley 28	donde se paga por uso, cada defecto es una factura	14
ley 31	la función de apoyo acaba autorizando, y se queda ciega	12
ley 19	la compensación hace invisible el fallo	12
ley 32	manda la restricción que el dinero no puede levantar	8

Y la última ley, que la parte 23 obliga a escribir:

LEY 32
manda la restricción que el dinero no puede levantar;
todas las demás son decisiones de coste

apariciones en esta parte	8
clase 277	la capacidad del pico se compra; lo que no se compra es la confianza de quien no pudo comprar el último día
clase 278	un cobro duplicado se devuelve; lo que no se compra es poder demostrar ante un tercero qué ocurrió
clase 279	ningún presupuesto deshace una filtración de un diagnóstico
clase 280	ninguna cantidad cambia bajo qué jurisdicción queda un dato
clase 281	los bytes se pagan; la calidad percibida en el dispositivo del espectador, no
clase 282	no hay contrato que haga fiable un enlace satélite en una mina
clase 283	ningún descuento repara que un cliente haya visto los datos de otro
clase 284	ningún gasto detecta un fallo que no produce error, si nadie compara

y lo que la distingue de la ley 14
 la 14 dice CUÁNDO se decide el coste: al crear
 la 32 dice QUÉ NO ES una decisión de coste
 → y esa distinción es la que ordena una arquitectura
 → primero se identifica lo que el dinero no levanta, y
 todo lo demás se optimiza alrededor
 → hacerlo al revés produce sistemas caros que fallan en
 lo único que importaba

Y lo que hace útiles a estas 32:

NINGUNA SE ESCRIBIÓ ANTES DE LA EVIDENCIA
 cada una apareció cuando una predicción falló
 → 24 partes, 120 predicciones escritas antes
 → y las que fallaron fueron las productivas

Y NINGUNA ES UNA REGLA
 no dicen qué hacer: dicen qué OCURRE si no se hace nada
 → describen el estado por defecto de los sistemas y de
 las organizaciones
 → y por eso lo único que las contrarresta es
 comprobación automática y periódica

clases 228, 264

4. El plan de doce meses

Lo único que importa a partir de aquí. Y no es una lista de temas.

LO QUE NO FUNCIONA
 «voy a estudiar Kubernetes y sacarme dos
 certificaciones»
 → temas, sin efecto medible

clase 265

LO QUE FUNCIONA, cuatro compromisos

- 1 QUÉ CONSTRUYO, y qué mide que funcionó
 una cosa, pequeña y completa, con
 infraestructura como código, cadena de entrega,
 identidad sin credenciales permanentes, señales por
 recorrido, copias con restauración probada y coste
 medido
- 2 QUÉ HUECO CIERRO, y cómo lo sabré
 una RESTRICCIÓN, no un producto
 → «no sé razonar sobre coherencia entre regiones»
 → y el criterio: «lo sabré cuando pueda predecir el
 resultado de un ensayo y acertar»
- 3 QUÉ RETIRO
 → porque retirar fue, medido, de lo más rentable de
 todo el programa: 245 alertas, 9 procedimientos, 426
 conjuntos, 61 paneles, 4 modelos
 → y nadie lo pide nunca
- 4 QUÉ HIPÓTESIS ESCRIBO, con fecha de corrección
 dos o tres afirmaciones refutables sobre tu propio
 trabajo
 → «creo que el cuello de mi sistema es la base; predigo
 que al cargar hasta romper será el grupo de
 conexiones»
 → y una fecha para corregirlas en público

clase 275

Y el calendario mínimo, que es lo que lo hace real:

MENSUAL
 ejecutar un procedimiento al azar
 revisar qué se puede retirar

clase 259

TRIMESTRAL
 un ensayo con hipótesis escrita
 restaurar una copia con reloj
 revisar los riesgos aceptados que vencen

clase 261

clase 255

clase 287

SEMESTRAL

una revisión estructurada de un sistema clase 286
corregir las hipótesis propias en público

ANUAL
conmutación completa con carga y con acta clase 278
y reescribir el plan

Y las tres preguntas para elegir qué hacer primero:

«¿QUÉ ME QUITA EL SUEÑO DE MI SISTEMA?»
→ y comprobarlo esta semana

«¿QUÉ NO HE PROBADO NUNCA?»
→ y probar lo más barato de probar

«¿QUÉ SÉ QUE NADIE MÁS SABE?»
→ y escribirlo

Y el cierre del programa:

288 clases, 24 partes, 32 leyes y 120 predicciones
escritas antes de saber la respuesta.

De esas 120, unas cuarenta fallaron. Y todo lo que este
programa tiene de propio –las 32 leyes– salió de esas
cuarenta, no de las otras ochenta.

El método no era enseñar la nube: era predecir, medir y
corregirse en público. Y funciona igual en un sistema de
cuarenta servicios que en una carrera: escribe lo que
crees que va a pasar, mídelo, y cuando falles, escribe
por qué.

Lo demás –los productos, los nombres de servicio, las
consolas– cambiará. Las restricciones no.

Ejemplo trabajado

La retrospectiva completa del programa, con las cifras. Lo que sigue son las 120 predicciones y su marcador, las leyes
que más aparecieron y por qué, lo que el programa dejó fuera, y un plan de doce meses real.

El marcador de las 120 predicciones.

24 partes × 5 predicciones	120	
correctas y confirmadas	49	41 %
correctas y subestimadas o incompletas	31	26 %
parciales	23	19 %
fallidas	17	14 %
y las leyes escritas	32	
de ellas, nacidas de una predicción fallida	20	
de una predicción subestimada	9	
de una confirmación que reveló un mecanismo	3	

Y el patrón:

el 63 % de las leyes salió de haberse equivocado
→ y ninguna de las 49 predicciones plenamente correctas
produjo una ley por sí sola

→ acertar confirma lo que ya se sabía
→ fallar es lo que enseña
→ y por eso las predicciones se escribían ANTES, con
cifra, y se corregían en público

Las leyes que más aparecieron, y qué tienen en común.

las cinco primeras suman el 45 % de todas las apariciones

- 13 lo que no se mira deja de funcionar en silencio
- 15 la señal existe y nadie la mira
- 22 un procedimiento nunca ejecutado no funciona
- 16 un control que estorba se rodea
- 14 el coste se decide al crear, no al pagar

→ y las cinco describen lo mismo desde ángulos distintos
UN SISTEMA SIN COMPROBACIÓN PERIÓDICA SE DEGRADA HACIA
EL ESTADO EN QUE NADIE SE ENTERA

→ no son leyes de la nube: son leyes de las cosas que
cambian y de las personas que las olvidan
→ y por eso el programa acabó en operación, ensayos y
revisiones, y no en catálogos de servicios

Los hallazgos más caros del programa, ordenados.

posición	hallazgo	clase	coste o efecto
1	la réplica «síncrona» era asíncrona	278	0 pérdida prometida al regulador, falsa
2	19 de 34 procedimientos rotos	259	incluido el de conmutación
3	cuotas de la región secundaria al 3,2 %	262	plan de continuidad inviable
4	1,1 M de historias clínicas en entornos de prueba	279	mayor riesgo del sistema, fuera del inventario
5	un parámetro en la ruta de los segmentos	281	61.000 USD/mes, 3 años
6	un campo «importe» con tres significados	252	el error más caro de la parte 20
7	2 años de producción inflada hasta un 19 %	282	decisiones de compra con datos falsos
8	un informe cruzando datos entre clientes	283	19 meses, 41 comercios
9	el comité aprobó el 91 % de lo que causó incidentes	260	546 h/año y 4,2 días de espera
10	31 minutos sin que nadie decidiera	285	arreglado con una frase

→ de los diez, SIETE no producían ningún error
→ y NUEVE existían desde hacía más de un año
→ ninguno se encontró leyendo documentación: los diez
aparecieron al EJECUTAR algo

Lo que el programa dejó fuera, dicho explícitamente.

no enseña a programar
no cubre el detalle de configuración de ningún producto
→ deliberadamente: caduca clase 265
no entra en el diseño de interfaces de usuario
no cubre gestión de personas ni contratación
las cifras de CloudShop son un caso construido para
enseñar: los órdenes de magnitud son realistas, los
números concretos no son de una empresa real
y las 32 leyes son observaciones, no teoremas
→ cada una puede refutarse con evidencia
→ y eso sería el mejor uso posible de este programa

Un plan de doce meses, escrito de verdad.

DE UNA PERSONA CON 4 AÑOS DE EXPERIENCIA, NIVEL 2 REAL

- QUÉ CONSTRUYO
un servicio de seguimiento de envíos, pequeño y
completo
infraestructura como código, cadena con canario,
identidad sin credenciales permanentes, indicadores
por recorrido, copias restauradas cada mes y coste
por consulta medido
plazo meses 1 a 4
qué mide que funcionó
restauración probada < 15 min
y un ensayo mensual que no lo tumba

- 2 QUÉ HUECO CIERRO
el fallo parcial: no sé razonar sobre dependencias
lentas
cómo lo sabré
en el mes 6, inyecto 200 ms en una dependencia de mi
servicio
PREDIGO: la latencia total subirá menos de 400 ms
→ y si me equivoco por más del doble, no lo he
entendido
- 3 QUÉ RETIRO
las 14 alertas de mi equipo que no han hecho actuar a
nadie en 6 meses
plazo mes 2
y los 3 paneles que nadie abre
- 4 MIS HIPÓTESIS, con fecha de corrección: mes 12
a «el cuello de mi servicio será el grupo de
conexiones, no la CPU»
b «más de la mitad de mis procedimientos fallarán al
ejecutarlos»
c «el coste estará dominado por transferencia de datos
y no por cómputo»

Y la corrección de esa persona, doce meses después:

- a «el cuello será el grupo de conexiones»
CORRECTA. Codo a 1.840 pet/s con la CPU al 44 %.
- b «más de la mitad de mis procedimientos fallarán»
FALLIDA, Y POR UN MOTIVO QUE NO ESPERABA. Fallaron 2
de 9, no 5. Pero solo porque los escribí hace cuatro
meses. Los tres que tenían más de ocho meses fallaron
los tres. La ley 22 no depende del número: depende del
tiempo desde la última ejecución.
- c «el coste estará dominado por transferencia»
FALLIDA. Lo dominó el almacenamiento de registros:
el 51 %, con 400 días de retención que puse por
defecto y no revisé. Ley 26, en mi propio sistema, y
después de haberla estudiado.

y la predicción del hueco
inyecté 200 ms: la latencia total subió 310 ms
→ dentro de lo predicho
→ hueco cerrado, y con evidencia

Y lo que esa persona escribió al cerrar el año:

«de mis tres hipótesis, fallé dos. Y las dos que fallé me
enseñaron algo que no sabía: que la ley 22 se mide en
tiempo y no en cantidad, y que estudiar la ley 26 no me
protegió de ella.

Eso último es lo que más me sorprendió: sabía que los
valores por defecto sirven a la demostración, lo había
leído, lo había explicado, y aun así dejé 400 días de
retención en mi propio sistema durante un año.

Saber una ley no protege de ella. Solo la comprobación
periódica protege. Que es exactamente lo que el programa
decía y yo creía haber entendido.»

Y el cierre.

288 clases.
24 partes que empezaron escribiendo lo que creían y
terminaron corrigiéndolo con cifras.
120 predicciones, unas cuarenta fallidas.
32 leyes, veinte de ellas nacidas de un error propio.

Y una sola instrucción para lo que viene:

escribe lo que crees que va a pasar,
mídelo,

y cuando falles, escribe por qué.

Lo demás cambiará de nombre.

La lección que esta clase deja: de las 120 predicciones, las 49 plenamente correctas no produjeron ni una sola ley; las 32 salieron de las que fallaron o se quedaron cortas. Y la mejor demostración de que el método funciona no está en el programa: está en que alguien que había estudiado la ley 26 dejó 400 días de retención por defecto en su propio sistema durante un año —porque saber una ley no protege de ella, y solo la comprobación periódica lo hace.

Laboratorio guiado

Ejecuta desde la raíz:

```
python classes/part-23-industry-capstones/288-defensa-final-retrospectiva-y-plan-de-12-meses/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 final-defense 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 final-defense 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 plan de doce meses es una lista de tecnologías y certificaciones	Se planifican temas en vez de efectos medibles	Comprométete con qué construyes y qué lo mide, qué hueco de restricción cierras y cómo lo sabrás, qué retiras y qué hipótesis escribes con fecha.
Se estudia una ley y aun así se cae en ella	Saber una ley no protege de ella: describen el estado por defecto, no una regla que se recuerde	Convierte cada ley en una comprobación periódica automática; es lo único que la contrarresta.
La defensa final se centra en lo que salió bien	Se presenta el trabajo como una demostración comercial	Pon lo que falló en el minuto ocho y declara lo que no probaste; el panel evalúa esas dos secciones antes que ninguna.
Un año de trabajo no se puede demostrar en una conversación	Se acumuló actividad sin efecto, mecanismo ni cifra	Por cada trabajo, escribe qué cambió, por qué mecanismo y cuánto, en el momento de hacerlo, no meses después.
Las hipótesis propias nunca se corrigen	No tenían fecha ni criterio de refutación	Escribe dos o tres afirmaciones refutables con cifra y una fecha de corrección; sin fecha no se corrigen nunca.
Se optimiza el coste y el sistema falla en lo esencial	No se identificó primero la restricción que el dinero no levanta	Empieza por lo que ningún presupuesto resuelve y optimiza todo lo demás alrededor; hacerlo al revés produce sistemas caros que fallan donde importaba.

■ 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 dos criterios añade la rúbrica final y por qué separan más que los otros?
2. ¿Cuál de las cinco predicciones falló y qué ley obligó a escribir?
3. ¿Qué distingue la ley 32 de la ley 14 y cómo ordena una arquitectura?
4. ¿Qué proporción de las leyes nació de un error y qué implica eso sobre el método?
5. ¿Qué cuatro compromisos tiene un plan de doce meses que sirva?

Referencias

- Popper, K. (1963). Conjeturas y refutaciones — el método de predecir y corregir. <<https://www.routledge.com/Conjectures-and-Refutations/Popper/p/book/9780415285940>>
- Beyer, B. y otros (2018). The Site Reliability Workbook. <<https://sre.google/workbook/table-of-contents/>>
- Forsgren, N., Humble, J. y Kim, G. (2018). Accelerate. <<https://itrevolution.com/product/accelerate/>>
- Kleppmann, M. (2017). Designing Data-Intensive Applications. <<https://dataintensive.net/>>
- Ericsson, K. A. y Pool, R. (2016). Peak — práctica deliberada con realimentación. <<https://www.hachettebookgroup.com/titles/anders-ericsson/peak/9780544456259/>>
- Documentación oficial vigente del servicio implementado; registra URL y fecha de consulta.

Descargar: Parte 23 en PDF · Manual integral

Anterior	Índice	Siguiente
← 287 · Paquete de evidencia, costos y riesgos residuales	Parte 23 · Programa	Fin del programa

Evaluación — 288 Defensa final, retrospectiva y plan de 12 meses

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 final-defense 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.