

Distribución de datos y *Map Reduce*

PEC2

Ejercicio 1 (25%)

OpenAI ha encomendado a nuestro equipo la tarea de desarrollar una base de datos para almacenar las interacciones de los usuarios con sus diversos modelos de inteligencia artificial, como GPT-4, ChatGPT, entre otros. La razón detrás de esta solicitud es la escasez de conjuntos de datos de conversaciones, los cuales son fundamentales para la adaptación y mejora de los modelos de lenguaje.

Adicionalmente, OpenAI nos ha solicitado la creación de una API REST que permita a los usuarios acceder a esta base de datos y extraer las conversaciones almacenadas. Esto brindará a los usuarios la capacidad de generar sus propios conjuntos de datos de conversaciones utilizando esta información.

Es importante destacar que OpenAI tiene modelos de inteligencia artificial desplegados en diferentes regiones (por ejemplo, España). Además, las consultas están dirigidas normalmente a recuperar conversaciones de una región en concreto. De esta manera, el usuario se asegura en cierta medida que la mayoría de las conversaciones están en el mismo idioma.

Se espera que el número de escrituras supere al de lecturas, dado que: en primer lugar, los modelos de OpenAI han ganado una amplia notoriedad desde la introducción de ChatGPT, y prácticamente son utilizados por todo el mundo; y en segundo lugar, los usuarios que acceden a la base de datos son principalmente ingenieros especializados en inteligencia artificial, un público bastante reducido. Asimismo, se reconoce que el uso principal de las lecturas es la creación de conjuntos de datos con miles de conversaciones, por lo que la pérdida ocasional de alguna conversación no resulta crítica.

Se pide lo siguiente:

1. Razonar qué modelo de datos sería el más adecuado.
2. Explicar qué estrategia de fragmentación (partición) y de replicación de los datos sería la óptima.
3. Indicar el modelo transaccional más adecuado para los requisitos de la aplicación descrita y justificar el porqué.
4. Supongamos que se ha optado por el uso de quóruns como sistema de replicación con $N=5$, ¿qué valores asignarías a W y a R ?

Exponed la solución de forma argumentada en dos páginas como máximo.

Los criterios de evaluación para evaluar este ejercicio son los siguientes:

Pregunta	No logrado (D/C-)	Mínimamente logrado (C+)	Logrado (B)	Logrado de forma sobresaliente (A)
Pregunta 1	La respuesta es incorrecta o no está justificada.	La respuesta es correcta y está justificada, pero sólo analiza el modelo de datos ganador.	La respuesta es correcta, analiza todos los modelos de datos y está justificada. La justificación en algún caso es incompleta (obvia algunos aspectos relevantes) o discutible.	La respuesta es correcta e indica, por cada tipo de base de datos, su adecuación y su justificación. Los argumentos planteados son completos y adecuados.
Pregunta 2	La respuesta es incorrecta o no está justificada.	La respuesta es correcta y está bien justificada en la mayoría de las características del sistema (disponibilidad, escalabilidad, distribución de carga y localidad de datos).	La respuesta es correcta y está bien justificada en casi todas las características del sistema (disponibilidad, escalabilidad, distribución de carga y localidad de datos).	La respuesta es correcta y está bien justificada para todas las características del sistema (disponibilidad, escalabilidad, distribución de carga y localidad de datos).
Pregunta 3	La respuesta es incorrecta o no está justificada.	La respuesta es correcta y está mínimamente justificada.	La respuesta es correcta, considera los distintos modelos disponibles y está bien justificada.	La respuesta es correcta, considera los distintos modelos disponibles, está bien justificada y justifica de acuerdo al enunciado la decisión tomada (consistencia vs disponibilidad).
Pregunta 4	La respuesta es incorrecta o no está justificada.	La respuesta es incorrecta en algún aspecto, pero la justificación es correcta, demuestra conocimiento sobre	La respuesta es correcta y está mínimamente justificada.	La respuesta es correcta y la ampliamente desarrollada y justificada.

		el tema y está referenciada.		
--	--	------------------------------	--	--

SOLUCIÓN

Modelo de datos

Dado que el sistema gestionará una cantidad enorme de datos y soportará una gran cantidad de escrituras concurrentes, usar un modelo de datos relacional no es una buena idea ya que la necesidad de mantener una consistencia estricta generaría cuellos de botella en las escrituras y podría rechazar algunas escrituras por fallos en la red. Según los requisitos, parece adecuado realizar una fragmentación horizontal de los datos. Por otra parte, el número de relaciones entre los datos no es importante, por lo que descartaríamos una base de datos orientada a grafos. Además, debido a la amplia adopción de los modelos de IA de interés, es posible que las necesidades de recursos de la base de datos crezcan a corto plazo, por lo que un sistema que proporcione escalabilidad horizontal sería muy adecuado. Es por todo ello que la mejor elección es utilizar un modelo orientado hacia agregados: clave-valor, orientada a documentos u orientada a columnas: ofrecen alta escalabilidad y facilitan la fragmentación horizontal de los datos. La elección de una u otra, vendría condicionada por el tipo de consultas que se realicen en el sistema: en caso de consultas por conversación (a partir de su identificador) una base de datos clave-valor podría ser una buena opción, en caso de buscar consultas con algunos elementos (parámetros en común) una base de datos de agregación de grandes columnas sería una buena opción y, en caso de que las consultas sean más abiertas o sobre el contenido de los agregados, entonces una base de datos documental sería una buena opción.

Fragmentación

Con el objetivo de agilizar las escrituras (tanto el usuario como el modelo de inteligencia artificial están en una misma región) y las lecturas (consultas dirigidas a recuperar conversaciones por región), se propone distribuir los datos en la red almacenándolos geográficamente en las regiones consideradas por OpenAI mediante una fragmentación horizontal. Respecto a la replicación, sería conveniente guardar algunas réplicas (quizá no demasiadas para no penalizar en exceso a las escrituras) para promover la disponibilidad de la base de datos.

Consistencia de base de datos

Para garantizar los requisitos de alta disponibilidad y velocidad de acceso al sistema se utilizará el modelo BASE, que potencia la disponibilidad de los datos (AP) frente a la consistencia (AC) de los mismos. Hay que tener en cuenta que es posible que se

produzca una pérdida de consistencia de la información que impida mostrar algún valor o lo muestre posteriormente. Sin embargo, tal como se indica en el enunciado, perder alguna conversación no es crítico para el caso de uso. Por tanto es asumible una pérdida temporal de la consistencia. El uso de un modelo ACID en este caso sería poco eficiente, ya que garantizar el nivel de consistencia en un gran número de escrituras concurrentes podría repercutir negativamente en el rendimiento de la base de datos.

Valores de W y R

El sistema ha de ser consistente en el tiempo, es decir, los valores de W y R deben cumplir $W + R < N$. Por otro lado, el enunciado establece que las escrituras son mucho más frecuentes que las lecturas. De modo que tenemos que $W < R$ para no sobrecargar el sistema a la hora de escribir. Así pues, un posible ejemplo de solución válida podría ser $N=5$, $W=1$ y $R=3$. Así tendríamos lecturas inconsistentes, pero con una gran probabilidad de obtener valores consistentes (la lectura retorna después de leer un mínimo de 3 réplicas de las 5), manteniendo un sistema altamente disponible (el sistema permite la caída de 4 nodos al realizar escrituras y la caída de 2 nodos al realizar lecturas). Además, el hecho de que las escrituras sean más frecuentes, haría que para las operaciones más comunes la disponibilidad y el rendimiento sean mayores.

Ejercicio 2 (25%)

A partir de la lectura de los apuntes del curso, exponed, para cada una de las afirmaciones, si creéis que es cierta o falsa, indicando una breve argumentación que justifique vuestra respuesta.

Las afirmaciones en las que no se indique explícitamente si es **CIERTA o **FALSA**, o bien carezcan de argumentación, se considerarán no válidas. Se valorará la concisión** (una página y media para las 5 afirmaciones como máximo) y **el uso de referencias** (sección/página de los apuntes del curso) para justificar las respuestas. Las respuestas con referencias **sin explicaciones propias** (tipo CIERTA, “como dice... / como se encuentra en...”) tendrán poco valor.

Afirmación 1

Dos transacciones concurrentes T1 y T2, que sólo realizan operaciones de lectura, nunca provocarán interferencias en la Base de datos, aun accediendo a los mismos datos.

Afirmación 2

Las bases de datos orientadas a documentos solamente admiten replicación, resultando imposible las técnicas de distribución como sharding.

Afirmación 3

Un modelo de replicación peer to peer tendrá los mismos problemas de rendimiento y disponibilidad asociados a la replicación master-slave.

Afirmación 4

Una ventaja del modelo master-slave asíncrono frente al modelo master-slave síncrono es que en el modelo asíncrono el nodo que contiene la réplica primaria deja de ser un cuello de botella a efectos de rendimiento y disponibilidad.

Afirmación 5

El desarrollo de bases de datos distribuidas no enfrenta alguna dificultad extra en comparación con el de base de datos centralizadas.

Los criterios de evaluación para evaluar este ejercicio son los siguientes:

No logrado (D/C-)	Mínimamente logrado (C+)	Logrado (B)	Logrado de forma sobresaliente (A)
La respuesta es incorrecta o no está justificada/referenciada.	La respuesta es incorrecta, pero la justificación es correcta, demuestra conocimiento sobre el tema y está referenciada.	La respuesta es correcta y está convenientemente justificada/referenciada.	La respuesta es correcta, está correctamente justificada/referenciada y es concisa (ocupa menos de una página y media).

SOLUCIÓN

Afirmación 1

La afirmación es **CIERTA**.

Las operaciones de 'consulta' no pretenden cambiar el estado del database, pues nunca interfieren entre ellas. Es decir, sea cual sea el orden de ejecución de las operaciones de T1 y T2, siempre obtendremos el mismo resultado.

[p. 12 GestionTransacciones.pdf] "b) Como mínimo una de las dos transacciones, sobre esta porción común de la BD a la que acceden, efectúa operaciones de actualización."

Afirmación 2

La afirmación es **FALSA**.

No existe ninguna razón teórica por la que las bases de datos documentales no puedan soportar el sharding: de hecho, como veremos más adelante en MongoDB, se puede implementar de diferentes maneras. Si una base de datos no permitiera fragmentación, pero si replicación, estaríamos hablando de una base de datos centralizada con mirrors (copias de la misma en distintos nodos) y, como hemos visto durante el curso, las bases de datos de agregación son bases de datos que, en la mayoría de los casos, fomentan la distribución.

[p.14 B3_T5_3_BDD_Disenyo.pdf] “el caso de bases de datos orientadas a documentos, el sharding se puede efectuar, bien aplicando técnicas de hash o bien a partir del valor que toman ciertos atributos”.

Afirmación 3

La afirmación es **FALSA**.

En un modelo P2P las operaciones de escritura pueden ser realizadas sobre cualquier réplica, al igual que las lecturas.

[p.8 B3_T7_BDD_BASE.pdf] “las políticas de replicación P2P resuelven los problemas de rendimiento y disponibilidad asociados a la replicación master-slave.”

Afirmación 4

La afirmación es **FALSA**.

En el modelo master-slave, con independencia de si es síncrono o asíncrono, hay un problema con el nodo que contiene la réplica primaria, dado que, si este falla el sistema solo podrá ofrecer operaciones de lectura pero no de escritura y que si este se sobrecarga penalizará las siguientes operaciones de escritura. En el modelo master-slave asíncrono el nodo secundario es el que deja de ser un cuello de botella.

[p.7 B3_T5_2_BDD_Arquitecturas.pdf] “Al igual que en la política master-slave síncrona, el nodo que contiene la réplica primaria sigue siendo un cuello de botella a efectos de rendimiento y disponibilidad”

Afirmación 5

La afirmación es **FALSA**.

La implementación de una base de datos distribuida se complica mucho a la hora de ‘orquestrar’ el funcionamiento coherente de todas las partes.

[p.9 B3_T5_1_BDD_Introduction.pdf] “garantizar las funcionalidades que clásicamente se asocian a un sistema gestor de bases de datos es más complicado, debido a la distribución de los datos, de las tareas y del control de la ejecución de dichas tareas”

Ejercicio 3 (30%)

Considerar los datos de clientes de una pequeña compañía telefónica. Por un lado, se dispone de información personal de cada cliente, identificado de forma única por su DNI.

DNI	Nombre	Apellido1	Apellido2	Fecha Nac.	Residente en
12345678A	Carlos	García	Pérez	01/01/1980	Madrid
23456789B	Laura	Martínez	Sánchez	02/02/1981	Barcelona
34567890C	David	López	González	03/03/1982	Valencia
45678901D	Marta	Fernández	Moreno	04/04/1983	Sevilla
56789012E	Ana	Torres	Jiménez	05/05/1984	Valencia
67890123F	Jose	Gómez	Ruiz	06/06/1985	Málaga
78901234G	Beatriz	Díaz	Hernández	07/07/1986	Murcia
89012345H	Juan	Álvarez	Molina	08/08/1987	Valencia
90123456I	Carmen	Ramos	Navarro	09/09/1988	Valencia
01234567J	Francisco	Romero	Cantero	10/10/1989	Bilbao

Por otro lado, se cuenta con el registro de todas las facturas de cobro mensual de 2023 a los clientes.

DNI	Mes Facturación	Cantidad Facturada	Descuento %	Pagada
12345678A	enero	100 €	10	Sí
23456789B	abril	200 €	15	No
34567890C	abril	150 €	5	Sí
45678901D	abril	250 €	0	No
56789012E	mayo	300 €	20	Sí
67890123F	abril	110 €	0	No
78901234G	julio	120 €	10	Sí
89012345H	agosto	130 €	15	No
90123456I	abril	140 €	5	Sí
01234567J	octubre	160 €	0	No

Se desea identificar, el cliente residente en Valencia que ha gastado más en abril del 2023. La salida deberá indicar el nombre y apellidos del cliente y el gasto del mes de abril del 2023.

Se aconseja usar dos pasadas map-reduce. Para ello se pide explicar en cada pasada lo que ocurre en cada fase (map, shuffle, reduce). Concretamente para cada fase:

- Mostrar todos los datos de entrada (input)
- Explicar qué acciones se producen
- Mostrar el resultado producido (output) por esas acciones en todos los datos de entrada.

Es imprescindible explicar claramente la lógica de las acciones realizadas, enseñando en detalle la entrada y la salida de cada fase. Se valorarán la **lógica, la eficiencia del algoritmo y las explicaciones**, y no se tendrá en cuenta el (pseudo)código.

Los criterios de evaluación para evaluar este ejercicio son los siguientes:

No logrado (D/C-)	Mínimamente logrado (C+)	Logrado (B)	Logrado de forma sobresaliente (A)
La respuesta es incorrecta o no está justificada.	La respuesta es parcialmente incorrecta, pero la justificación es correcta y demuestra conocimiento sobre el tema.	La respuesta es correcta y los diferentes pasos realizados se describen convenientemente y se ejemplifican mediante la evolución de los datos. En algún punto aislado, la respuesta puede ser incompleta (la ausencia de justificación o de cálculo de datos en algún punto) o incorrecta (contener algún error menor).	La respuesta es correcta, presenta y describe los pasos realizados y los ejemplifica correctamente mediante todos los cálculos de los datos desde el principio hasta el resultado final.

SOLUCIÓN:

Primera Pasada de map-reduce, M1

M1 Fase MAP

INPUT: Todos los datos

ACCIONES:

- Para cada tupla de la primera tabla: se ignora (i.e. no se emite nada) si los clientes no son residentes en Valencia. Se emite una tupla de la forma (34567890C, (David, López, González)), para las que cumplen con la condición de residencia.
- Para cada tupla de la segunda tabla: se ignora (i.e. no se emite nada) si el mes de facturación no es abril. Se emite una tupla de la forma (DNI, cantidad), como por ejemplo (90123456I, 140), para las que cumplen con la condición de fecha de facturación.

Observad que, para cada tupla input, solo se propaga información que puede ser relevante para el output final del algoritmo. Por ejemplo, no es necesario propagar registros relativos a clientes no valencianos, o propagar información ('columnas') sobre descuento, estado de pago de la factura. Esto sirve para simplificar la lógica y reducir la cantidad de datos que viajan por el clúster.

De otra manera: en esta fase, se minimizan los datos a procesar, y se etiqueta (o 'taguea') cada fila. El etiquetado por DNI permite implementar lo que en SQL es un FULL OUTER JOIN.

OUTPUT:

```
[
(34567890C,(David, López, González)),
(56789012E,(Ana, Torres, Jiménez)),
(89012345H,(Juan, Álvarez, Molina)),
(90123456I,(Carmen, Ramos, Navarro)),
(23456789B, 200),
(34567890C, 150),
(45678901D, 250),
(67890123F, 110),
(90123456I, 140)
]
```

M1 Fase SHUFFLE

INPUT = OUTPUT MAP

ACCIONES:

- Se agrupan por DNI los pares

OUTPUT:

```
[
(34567890C: ((David, López, González), 150)),
(23456789B, 200),
(45678901D, 250)
(90123456I: ((Carmen, Ramos, Navarro), 140)),
(67890123F, 110),
56789012E, (Ana, Torres, Jiménez)),
(89012345H, (Juan, Álvarez, Molina))
]
```

M1 Fase REDUCE

INPUT = OUTPUT MAP

ACCIONES:

- Ninguna acción en sentido 'clásico' de reduce, pero sí descartar registros irrelevantes, es decir, esos que no tienen datos en ambas tablas (INNER JOIN). Eso nos permitirá minimizar el tráfico de red y la computación requerida en la segunda fase.

OUTPUT:

```
[
(34567890C: ((David, López, González), 150)),
(90123456I: ((Carmen, Ramos, Navarro), 140))
]
```

Segunda Pasada de map-reduce, M2

M2 Fase MAP

INPUT = OUTPUT **M1**

ACCIONES:

- Se 'taguea' cada tupla de entrada, por ejemplo emitiendo la tupla (TAG, tupla)

OUTPUT:

```
[
(TAG, (34567890C: ((David, López, González), 150))),
(TAG, (90123456I: ((Carmen, Ramos, Navarro), 140)))
]
```

M2 Fase SHUFFLE

INPUT = OUTPUT MAP

ACCIONES:

- Se agrupan los pares por clave

OUTPUT:

```
[(TAG, [(34567890C: ((David, López, González), 150))), (90123456I: ((Carmen, Ramos, Navarro), 140)))]
```

M2 Fase REDUCE

INPUT = OUTPUT MAP SHUFFLE

ACCIONES:

- Recorrer el registro, seleccionar el cliente que ha pagado más, y emitir la solución requerida (nombre, apellidos y gasto).

OUTPUT (RESULTADO FINAL):

((David, López, González), 150)

Ejercicio 4 (20%)

Considerar los siguientes sistemas:

- Un sistema de sensores para analizar la calidad del aire de una ciudad (consistencia final en el tiempo, Escrituras más frecuentes que lecturas)
- Un periódico online que es visitado por lectores de diferentes lugares de un país (final en el tiempo, Lecturas más frecuentes que escrituras)
- Un sistema de control aéreo (fuerte, Escrituras y lecturas de la misma importancia)
- Área de clientes de un banco (consistencia fuerte, lecturas más frecuentes que escrituras)

Se pide argumentar qué configuraciones de los valores W, R y N encajan mejor con cada caso con respecto al tipo de consistencia (fuerte/final en el tiempo) y al tipo de lecturas y escrituras que son necesarias realizar (Lecturas más frecuentes que escrituras, Escrituras más frecuentes que lecturas o Escrituras y lecturas de la misma importancia).

C1: N=9, W=3, R=4

C2: N=7, W=4, R=5

C3: N=7, W=7, R=7

C4: N=7, R=3, W=9

C5: N=9, R=11, W=4

C6: N=9, W=6, R=4

C7: N=7, W=4, R=1

Para resolverlo:

1. **Para cada configuración**, identificar y razonar tipo de consistencia, y tipo de lecturas y escrituras.
2. A continuación, analizar cuál de las configuraciones dadas encaja con cada uno de los sistemas propuestos.

Los criterios de evaluación para evaluar este ejercicio son los siguientes:

No logrado (D/C-)	Mínimamente logrado (C+)	Logrado (B)	Logrado de forma sobresaliente (A)
La respuesta es incorrecta o no está justificada.	La respuesta es incorrecta en algún aspecto, pero la justificación es correcta, demuestra conocimiento sobre el tema y está referenciada.	La respuesta es correcta y está convenientemente justificada, pero sólo justifica la configuración ganadora o bien el análisis de alguna de las configuraciones es incorrecta.	La respuesta es correcta, está correctamente justificada y aborda todas las configuraciones posibles, indicando para cada una de ellas, su adecuación al problema planteado y el porqué.

SOLUCIÓN

1) Antes de nada, analizaremos las distintas configuraciones y veremos cuáles ofrecen consistencia fuerte y cuáles consistencia final en el tiempo.

Recordemos las condiciones bajo las cuales los quóruns se comportan de una forma u otra:

- Consistencia Fuerte: $W > N/2$ y $W+R > N$
- Consistencia Final en el Tiempo: $W + R < N$

Por ello, las diferentes configuraciones ofrecen los siguientes tipos de consistencia:

- C1: $N=9$, $W=3$, $R=4$: Consistencia Final en el Tiempo, ya que $W+R=3+4=7 < N=9$
- C2: $N=7$, $W=4$, $R=5$: Consistencia Fuerte, ya que $W=4 > N/2=7/2=3.5$ y $W+R=4+5=9 > N=7$
- C3: $N=7$, $W=7$, $R=7$: Lecturas y escrituras síncronas ya que $N=W=R$. Por consiguiente, consistencia estricta, ya que obliga a escribir en todos los nodos y no tolera la caída de ningún nodo.
- C4 y C5 no son válidas puesto que W o R son mayores que N . En otras palabras, no es posible que se escriban/lean en más réplicas de las que existen.
- C6: $N=9$, $W=6$, $R=4$: Consistencia Fuerte, ya que $W=6 > N/2=9/2=4.5$ y $W+R=10 > N=9$
- C7: $N=7$, $W=4$, $R=1$: Consistencia Final en el Tiempo, ya que $W+R=4+1=5 < N=7$

2) Con esta información, pasamos a resolver los distintos problemas planteados.

- A. Un sistema de sensores para analizar la calidad del aire de una ciudad (consistencia final en el tiempo, escrituras más frecuentes que lecturas)

En un sistema de estas características, las escrituras de datos que llegan del sensor serán mucho más frecuentes que las lecturas que se hagan. Además, la consistencia del sistema no es prioritaria dado que incluso se asume que los sensores en algunas ocasiones devuelven valores equivocados o dejan de funcionar. Es por ello que el tipo de consistencia que mejor encaja sería la final en el tiempo. Si se consideran las configuraciones dadas, se tiene que son candidatas:

C1: $N=9$, $W=3$, $R=4$

C7: $N=7$, $W=4$, $R=1$

Y de ellas, la más indicada en un contexto con más frecuencia de escrituras que lecturas es la configuración C1 dado que requieren tan solo 3 nodos para escribir, por lo que las escrituras (la operación más frecuente) serán más rápidas.

- B. Un periódico online que es visitado por lectores de diferentes lugares de un país (final en el tiempo, lecturas más frecuentes que escrituras)

En un sistema de estas características, las lecturas de datos por parte de los lectores serán mucho más frecuentes que las escrituras o actualizaciones que realice el periódico sobre la página. Además, la consistencia del sistema no es prioritaria dado que es asumible y ocurre a veces que las réplicas no tienen la información actualizada. Es por ello que el tipo de consistencia que mejor encaja sería la final en el tiempo. Si se consideran las configuraciones dadas, se tiene que son candidatas:

C1: $N=9$, $W=3$, $R=4$

C7: $N=7$, $W=4$, $R=1$

Y de ellas, la más indicada en un contexto con más frecuencia de lecturas que escrituras sería la configuración C7 dado que es la que menos nodos necesita leer para resolver las operaciones más frecuentes.

- C. Un sistema de control aéreo (consistencia fuerte, escrituras y lecturas de la misma importancia)

En un sistema de estas características, las lecturas y escrituras tendrán la misma importancia dado que el controlador consulta y escribe datos con la misma frecuencia. Con respecto a la consistencia del sistema es un factor prioritario y crítico dado que las consecuencias de cualquier inconsistencia podrían ser

catastróficas. Es por ello que el tipo de consistencia que mejor encaja sería una fuerte. Si se consideran las configuraciones dadas, se tiene que son fuertes:

C2: N=7, W=4, R=5

C3: N=7, W=7, R=7

C6: N=9, W=6, R=4

En primer lugar nótese que la configuración C3 es muy restrictiva ya que, si se cae un solo nodo, el sistema completo caería (no se permitirían ni lecturas ni escrituras). C6 y C2 tendrían un rendimiento parecido. C6 sería más rápida para lecturas y C2 para escrituras.

D. Área de clientes de un banco (consistencia fuerte, lecturas más frecuentes que escrituras)

En un sistema de estas características, las lecturas de datos serán mucho más frecuentes que las escrituras. Normalmente los clientes realizan consultas de sus cuentas y el tipo de escritura que pueden realizar son más esporádicas tales como transferencias o contratación de algún producto y, de forma similar, le ocurre al banco sobre el sistema. Con respecto a la consistencia del sistema, es un factor prioritario y crítico, dada la información que gestiona. Es por ello que el tipo de consistencia que mejor encaja sería la fuerte. Si se consideran las configuraciones dadas, se tiene que son fuertes:

C2: N=7, W=4, R=5

C3: N=7, W=7, R=7

C6: N=9, W=6, R=4

Descartamos C3 por ser muy restrictiva. De las que quedan, la más indicada en un contexto con más frecuencia de lecturas que escrituras será la configuración C6, dado que es la que más favorece las lecturas (menos nodos para leer en cada transacción de lectura).

Criterios de valoración

Los apartados 1 y 2 tienen un peso del 25% cada uno, y los apartados 3 y 4 tienen un peso del 30% y el 20% respectivamente. Se valorará, para cada apartado, la validez de la solución y la claridad de la argumentación. Cualquier solución no justificada se considerará incompleta.

Formato y fecha de entrega

Tenéis que enviar la PEC al buzón de Entrega y registro de EC disponible en el aula (apartado Evaluación). El formato del archivo que contiene vuestra solución puede ser .pdf, .odt, .doc y .docx. Para otras opciones, por favor, contactar previamente con vuestro profesor colaborador. El nombre del fichero debe contener el código de la asignatura, vuestro apellido y vuestro nombre, así como el número de actividad (PEC2). Por ejemplo nombreakellido1_nosql_pec2.docx. La fecha límite para entregar la PEC2 es el **5 de mayo**.

Propiedad intelectual

Al presentar una práctica o PEC que haga uso de recursos ajenos, se tiene que presentar junto con ella un documento en que se detallen todos ellos, especificando el nombre de cada recurso, su autor, el lugar donde se obtuvo y su estatus legal: si la obra está protegida por el copyright o se acoge a alguna otra licencia de uso (Creative Commons, licencia GNU, GPL etc.). El estudiante tendrá que asegurarse que la licencia que sea no impide específicamente su uso en el marco de la práctica o PEC. En caso de no encontrar la información correspondiente tendrá que asumir que la obra está protegida por el copyright.

Será necesario, además, adjuntar los ficheros originales cuando las obras utilizadas sean digitales, y su código fuente, si así corresponde.