

Sistemas Distribuidos - Primera Evaluación - I Término 2021

Profesora: Vanessa I. Cedeño-Mieles

Fecha de Inicio: Julio 5, 2021 14:00 PM

Fecha de Entrega: Julio 5, 2021 16:00 PM (vía SidWeb); + 1 hora de entrega tardía

Instrucciones: (1) Incluya su nombre y número de matrícula en el documento de entrega, (2) Lea cada pregunta detenidamente, (3) Conteste claramente cada pregunta.

NOTA: NO se permite colaboración, el trabajo es individual.

Pregunta	1	2	3	4	5	6	7	Total:
Puntos	20	10	10	10	10	10	10	80
Puntaje								

Diseño de Aplicación web y móvil para venta de tickets de cine (ej: App de Supercines)

- Se desea una solución en la nube (con AWS, Azure o GCP).

IMPORTANTE: Más que diseño perfecto, evaluaremos que Ud. demuestre poder aplicar los conocimientos adquiridos en las unidades 1-4 y laboratorios del semestre (a la fecha), y que tenga claro las limitaciones de las tecnologías utilizadas en su diseño, así como que pueda identificar qué partes de su diseño deban ser mejoradas antes de realizar una implementación real.

- Requerimientos funcionales: Similares a cualquier App de venta de tickets en línea con App Web y Móvil; como ejemplo, pueden ver cómo funciona la aplicación web de Supercines.
 - Almacenamiento de datos: Creación de cuentas, compras, horarios, ingreso de datos de teatros, películas, etc.
 - Consulta de datos: Usuarios, películas, horarios, compras, etc.
 - Compra de tickets para una o más películas que se proyectan en uno o más teatros.
 - Cancelar compra
 - Actualizar datos
 - Reportes
- Requerimientos no funcionales
 - Baja latencia** en los requerimientos de los usuarios finales (los que compran los tickets).
 - Auto-escalable**, para soportar demanda aumentada en días de estreno (viernes), pre-estreno (jueves) y sobre todo, en días de pre estreno y estreno de películas con demanda muy alta (como la última película de los Avengers).
 - Confiabilidad:** Se debe poder soportar fallos de hasta un nodo (container / VM / etc.) de cada componente.
 - Minimizar costos** de infraestructura / nube en la medida de lo posible.
 - Otros (considerar añadir: Fácil mantenibilidad, ...)
- Preguntas:
 - [20 pts] Proponga un diseño y diágrámelo a mano (suba foto) o de preferencia, usando una herramienta como <https://www.draw.io/>
 - [10 pts] Liste cada uno de los componentes mostrados en su arquitectura y explique por qué lo incluye, así como indique qué producto específico usaría y por qué (ej., qué base de datos). Justifique su respuesta.
 - [10 pts] Para cada par de componentes que se comuniquen en su diseño ¿Cómo se comunican los componentes entre sí? (ej.: HTTP, REST, API gRPC, RabbitMQ, Kafka, Sockets TCP, etc.)
 - [10 pts] Explique cómo su diseño cumple con ser auto-escalable.
 - [10 pts] Explique cómo su diseño cumple con ser confiable (tolerar fallas de hasta un nodo de cada componente).
 - [10 pts] ¿Qué problemas podría tener su diseño? O ¿Qué debilidades tiene su diseño?
 - [10 pts] ¿Hay partes de su diseño que Ud. piensa que no están bien por su falta de experiencia en el tema? (explique).