



MasivMovies

**Test Senior
Backend Developer**

Estimado Candidato

En primer lugar, agradecemos tu interés en formar parte de Masiv y por postularte a nuestra vacante.

Esta prueba técnica tiene como objetivo evaluar tus habilidades de programación en un escenario similar al que podrías encontrar en tu día a día. A continuación, encontrarás las instrucciones para completarla y enviarla correctamente.

Confidencialidad

Este documento es estrictamente confidencial. No debe ser compartido ni divulgado en ninguna circunstancia. Asimismo, la solución que desarrolles es de carácter privado y no debe publicarse en Internet ni en repositorios públicos.

1. Propuesta arquitectura (40 Puntos):

MasivDevelopments es una empresa que desarrolla software para la gestión de cines. La empresa ha sido contratada para crear un sistema integral de gestión de cines que podrá ser vendido a diferentes empresas que operan teatros de cine pero que actualmente no tienen presencia digital. Se realizará el primer acercamiento del Backend en el cual el sistema debe ser capaz de manejar todas las operaciones relacionadas con películas, teatros, salas, funciones y boletos, y debe ser escalable y robusto para soportar múltiples clientes.

- Gestión de teatros: Se deberá poder realizar toda la gestión de diferentes teatros en cuanto al registro y control de teatros, funciones, cofitería.
- Reportes de transacciones: Se deberá tener un registro de todas las transacciones como compras de boletos por cliente y por su teatro correspondiente, además se debe generar un reporte personalizado, si quiere generar reportes por boletas vendidas por película o por boletas vendidas por teatros.
- Se contará con microservicios transversales a la empresa, estos microservicios deberán estar centralizados preferiblemente en una cuenta de AWS. Donde se podrá encontrar servicios de autenticación de usuarios, manejo de herramientas transversales, entre otros.

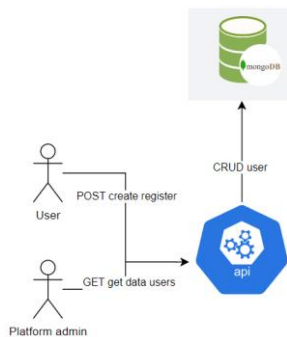
Requerimientos No Funcionales:

- Para la gestión de salas de cine, boletería, transversales y reporteria debe escalar transversalmente.
- Para compras de boletería debe soportar picos de 1000 Transacciones por segundo (TPS), con promedio de 300 transacciones por segundo y mensualmente con una volumetría aproximada de 10'000.000(tener en cuenta que hay días en el mes que no hay tráfico o el tráfico puede ser mucho menor).

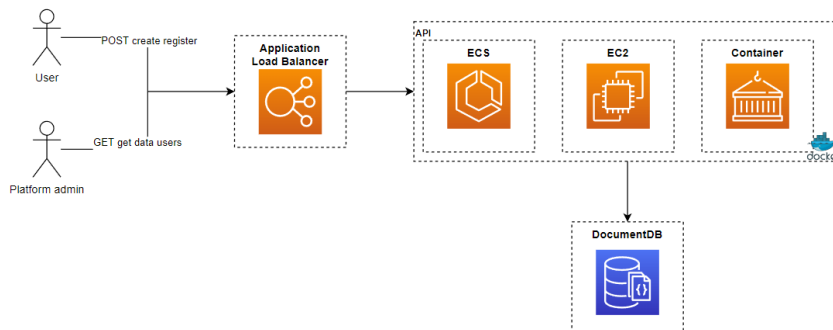
Con la anterior información diseñe una propuesta arquitectura en la que se vea como estaría inicialmente distribuida la arquitectura del software para la gestion de películas y boletería, herramientas transversales, reporterias y FrontEnd. Como puntos extras se puede tener estimación de costos.

Ejemplos:

Sin especificación de nube:



Con especificación de herramientas en la nube de AWS(deseable):



	PRUEBA TÉCNICA	Código: MV-RH-FOR-057
		Fecha: 03/05/2024
		Página: 4 de 7
		Información Interna

Este diagrama podrá ser presentado en un Draw.io o como una imagen.
La documentación será vital para la prueba **(10 Puntos extras)**.

2. Creación prototipo (60 puntos):

Desarrollo del Prototipo de Backend

Considerando el contexto previamente establecido, se requiere diseñar e implementar un prototipo funcional del backend. Este prototipo deberá desarrollarse utilizando tecnologías modernas y robustas, tales como API Rest y workers, entre otras.

Libertad de Diseño:

Para este proyecto, tendrás total libertad para definir la firma de cada endpoint, así como la elección de la base de datos que consideres más adecuada. El objetivo principal es evaluar tus habilidades en el diseño de arquitecturas escalables, eficientes y bien estructuradas, además de aplicar buenas prácticas de Clean Code.

Requisitos Funcionales:

El prototipo deberá cumplir con los siguientes requisitos funcionales:

1. Creación de Películas

Objetivo: Permitir la gestión de la información de las películas disponibles en el sistema.

- **Crear Película:** El sistema debe tener un endpoint que permita la creación de una nueva película.

2. Registro de una Sala

Objetivo: Permitir la gestión de las salas disponibles en cada sede de cine.

- **Registrar Sala:** El sistema debe exponer un endpoint que permita el registro de una nueva sala.

3. Registro de Funciones

Objetivo: Facilitar la programación y gestión de funciones de películas en las salas.

- **Registrar Horarios de Función:** El sistema debe exponer un endpoint que permita registrar horarios específicos para la proyección de una película en una sala.

- **Cancelar Funciones:** El sistema debe permitir la cancelación de funciones programadas. Al cancelar una función, se deben liberar los asientos reservados (Opcional).

4. Compra y Verificación de Boletos

Objetivo: Gestionar la compra de boletos y la disponibilidad de asientos.

- **Compra de Boletos:** El sistema debe exponer un endpoint que permita comprar boletos para una función específica. Para este punto usted será libre de escoger la firma, al igual que en el requerimiento de “verificación de estado”.

5. Reportes Mensuales

Objetivo: Generar reportes para el análisis de las funciones y la comunicación con los usuarios.

- **Reporte de Funciones Exitosas:** Generar un reporte mensual que muestre las funciones más exitosas, ordenadas de mayor a menor, basadas en la cantidad de boletos vendidos.

3. Pruebas Funcionales (15 puntos extras):

1. Para dicho prototipo se requieren realizar pruebas E2E. Elabore una propuesta de escenarios de pruebas los cuales podrá documentar en un documento a parte y/o plasmar en código o apoyado con prácticas de automatización de pruebas.

Adicional si lo considera necesario realizar pruebas de caja negra para garantizar puntos clave del desarrollo.

Puntos Extra:

- Implementación en un Lenguaje Fuertemente Tipado: Se recomienda utilizar un framework como .NET Core 8.0 para el desarrollo.
- Uso de Docker: Utilizar Docker para cada componente.
- Arquitecturas hexagonales, DDD, CQRS.
- Diseños de bases de datos.
- Control de Versiones con GitHub: Utilizar GitHub para el control de versiones y la gestión del código fuente.
- Costos mensuales de la arquitectura solución.
- Reporte de Sillas y Mensajes: Generar un reporte dado un rango de fechas que incluya:

- Cantidad de sillas reservadas al mes por sala.
- Mensajes enviados y recibidos relacionados con la compra de boletos
- Creación de Usuarios y Autenticación: Implementar un sistema de autenticación para usuarios utilizando tipos de autenticación como Bearer y Basic, para la compra de asientos.
- SOLID: Manejo específicamente de single responsibility

NOTA IMPORTANTE: El código de esta prueba no debe estar público. Como base se puede modificar firmas, adicional, el código debe ser compartido al área de selección en un archivo zip, de encontrarse que hay plagio, se castigará tanto al autor original (por ayudar con el plagio) como a quien lo realice. Solicitamos no incurrir en esta falta, de lo contrario se penalizará con el cierre definitivo del proceso.

Cualquier duda adicional, puede ponerse en contacto con la persona a cargo de su proceso de selección.

¡Le deseamos muchos éxitos!

CONTROL DE CAMBIOS

CONTROL DE CAMBIOS				
Versión	Fecha	Detalle	Elaborado Por	Aprobado Por
1		Versión inicial	Talent Acquisition Specialist	Talent Acquisition Leader
2		Actualización de formato	Talent Acquisition Specialist	Talent Acquisition Leader