

PROYECTO INTERMODULAR

“TAQUILLA DE CINE ONLINE”

INTEGRANTES DE GRUPO

Andrés Pérez Guardiola

Siham Fodil Ferouli

Ismael Lucas Aparicio

Juan Mira Alcántara - TEAM LEADER

ENTORNOS DE DESARROLLO

- IDEs: VS Code y IntelliJ
- Frameworks: Spring Boot y Next.js
- Testing: JUnit y Playwright
- Base de datos: MySQL
- Sistema Operativo: Windows

DESCRIPCIÓN DEL PROYECTO

La empresa opera varios cines en el entorno. El cliente pide que sea una página web. Se necesita que los clientes puedan ver la cartelera ofertada, los horarios de las distintas sesiones, puedan elegir su butaca y comprar la entrada para ir a ver la película.

La vista principal de la página incluye películas que están ahora mismo en emisión, estrenos futuros, o películas retro que se están remitiendo.

La App tiene que tener Películas, Actores, Directores, Compositores, Guionistas (Artistas en general), Sesiones, Salas, Butacas, Usuarios Registrados, Forma de Pago.

El cliente quiere que sus clientes puedan ver detalles de la película, como su elenco de actores, director, una breve sinopsis, un trailer de la misma. Al usuario le aparecerán las sesiones del día y también podrá seleccionar más días desde el calendario (donde le aparecerán las sesiones de otro día). Una vez seleccionas una

sesión de un día concreto: Te mandaría a seleccionar tu butaca.

Cuando el usuario decida qué sesión quiere ver, podrá seleccionar una o varias butacas que desee comprar. Allí le aparecerá también el número de la sala.

Realizar el pago, validar pago y mostrar factura.

Un cliente se tiene que registrar. Si quieres comprar tienes que estar registrado. Una vez compras la entrada, ésta te aparecerá en tu área de cliente con una factura, para que lo enseñes en la entrada y te dejen pasar.

ANÁLISIS DE REQUISITOS

Al desglosar la descripción del proyecto, hemos identificado 5 apartados: Cartelera, Proceso de Selección de Butaca y Sesiones, Proceso de Compra, Registro, Área Cliente. Y en cada parte hemos identificado una serie de vistas que el cliente puede ver y dentro de cada vista, las funcionalidades que puede hacer el cliente con ellas.

Requisitos Funcionales:

Cartelera:

Vista principal: Mostramos todas las películas disponibles, los próximos estrenos y otras películas especiales.

- Películas en emisión
- Estrenos
- Detalles del cine (Podría llegar a ser una vista independiente)
 - Ubicación
 - Horarios
 - Instalaciones
 - Fotos

Funcionalidad:

- El usuario puede filtrar películas mediante una serie de filtros o búsqueda avanzada.

(Le cambiaría la vista con las películas que hagan match con la búsqueda.

Paginación?)

- El usuario puede seleccionar la película deseada en esta vista. (Pasamos a [vista detalle película](#))

Vista detalle película: Se muestra información de la película como:

- Título
- Sinopsis
- Duración
- Tráiler
- Género
- Elenco de actores
- Director
- Edad(PEGI)

Funcionalidad:

- El usuario puede consultar los días (fechas) que está disponible la película
- El usuario selecciona el día que quiera ver la película. (Pasamos a vista de sesiones)

Vista sesiones: El usuario vería las sesiones disponibles para dicho día.

Funcionalidad: El usuario selecciona la sesión. (Pasamos a vista butacas)

Proceso de selección de entrada:

Vista butacas: El usuario vería una representación del cine, con las butacas disponibles y vacías.

- Butaca disponible (Gris)
 - Butaca reservada (Rojo)
 - Butaca no disponible actualmente (Negro)
- (recarga la página para actualizar el estado de esta butaca, es posible que esté siendo comprada por otro usuario)
- Butaca seleccionada por ti (Verde)

Funcionalidad:

- El usuario selecciona un elemento butaca dentro del cine o varios.
- El usuario confirma la selección, la app comprueba la disponibilidad de la butaca.
 - Si todo OK continuaría la compra. (Pasamos Vista compra)
 - Si no, pedirá que seleccione otras butacas (Recarga la página)
- Las butacas seleccionadas en proceso de compra no estarán disponibles para que otros las puedan comprar durante un tiempo para evitar el overbooking.

Proceso de compra (carrito):

Vista compra: Se verían los detalles de la compra.

- Película
- Fecha
- Sesión

- Número de butacas - Precio por unidad
- Precio total

Funcionalidad:

- Formulario de pago (Si no registrado, pedir registro -> Vista de registro)
- El usuario mandaría el formulario.
- Se redireccionará al sitio de pago correspondiente.
- La app validará la compra. (Pasamos a vista Área Cliente - Facturas - Factura Actual)
- Se generan las entradas en Vista Entradas con la información de las butacas, etc. y los QR. (Llamar API para generar QR, por ejemplo)

Registro de usuario (login / sign in):

Vista de registro: El usuario vería un pequeño formulario de registro.

- Email
- Contraseña
- Registro automático mediante Google (Auth0)

Funcionalidad:

- El usuario manda el formulario. Comprobamos que el usuario está autorizado y le devolvemos a la Vista de compra (Si estaba comprando) o a la Vista principal.

Área de cliente:

Vista Facturas: Se muestra una serie de elementos con las distintas compras efectuadas.

Funcionalidad: El usuario puede seleccionar una factura. (Vista Factura)

Vista Factura

- Detalles: Se mostrarán los detalles de la compra.
- Película
- Fecha
- Sesión
- Número de butacas
- Precio por unidad
- Precio total

Vista Entradas: Entradas de cine compradas (pendientes de uso)

- Película
- Fecha
- Sala

Funcionalidad: El usuario selecciona la película en concreto que reservaste entrada (Vista entrada)

Vista Entrada: Cada una de las butacas reservadas sería un elemento con información:

- Información de la sesión, con el pase Código QR de entrada (Global)
- Lista de butacas con información (ej. Ala Izquierda, Fila 3, Butaca 40)

Requisitos No Funcionales:

- Correcta visibilidad en todos los navegadores
- Responsividad
- Intuitividad
- Cifrado en la compra (registro de información personal y método de pago)
- Modo claro/oscuro
- 2 versiones (WEB y MÓVIL)

EXTRAS POSIBLES:

- Rendimiento óptimo (que aguante la carga de varios usuarios...)
- Función sin conexión (caché muy grande de datos)
- Descuentos (día del espectador)