

Sistema de Gestión Cinemar

Contexto

Cinemar es una empresa que se dedica a proyectar películas esencialmente dedicadas al público adolescente.

El cine cuenta con una cantidad de salas con diferentes capacidades (siendo esta capacidad la cantidad de butacas), también dispone de salas 2D como 3D variando el precio de las entradas.

Cuando un cliente se presenta en ventanilla muestra su tarjeta de descuento, si la tiene, se le efectúa un descuento en el valor de la entrada, sino pueden solicitar una si acudieron al menos 6 veces en 3 meses, en caso contrario el precio de la entrada no tendrá descuento alguno.

Actualmente la tabla de descuentos para los que tienen la tarjeta de descuentos es la siguiente:

- Lunes y Miércoles: 20%
- Martes y Jueves: 15%
- Viernes, Sábados y Domingos: 10%

Siendo modificable según los directivos.

Problemática

Los directivos de **Cinemar** comentaron a nuestro equipo que no cuentan con un control de los clientes, para realizar reservas de butacas y otorgarles descuentos para aquellos que son más recurrentes de forma automática.

Todo se efectúa mediante ventanilla y a mano, lo que provoca que en algunas salas a veces se terminan vendiendo más entradas que la capacidad de la sala, y perdiendo ventas en funciones por no contar con reservas por página web en horarios específicos.

Solución

Nos llega desde la administración del cine a nuestro equipo de desarrolladores que tenemos que implementar una solución que nos permita lo siguiente.

Para el cliente:

- Registrarse.
- Iniciar sesión.
- Crear una reserva.
- Modificar una reserva.
- Observar mis reservas.
- Ver el histórico de mis entradas.

Para la Administración:

- Ver reservas de todos los clientes.
- Ver reservas de un cliente en particular.
- Crear una sala con la película.
- Modificar una sala.
- Eliminar una sala.
- Modificar descuentos.

Troncales

- Ver salas

Consideraciones

- No se vencerán las películas, sino que será por la creación de una sala.
- La reserva implica el pago de la entrada.
- Las reservas solo se pueden modificar siempre y cuando se hagan antes de la función.
- Las funciones son todos los días.

Entregables

Checkpoint 1 (Fecha: 13/06 al 16/06)

- 1) Elaborar un diagrama de clases proponiendo la solución.
- 2) Elaborar otro diagrama de clases mostrando el método de registro de clientes e inicio de sesión.

Checkpoint 2 (Fecha: 20/06 al 24/06)

- 1) Elaborar el DER (diagrama de entidad relación) de la solución.
- 2) Presentar los Script de generación de esquemas de la base de datos.

Checkpoint 3 (Fecha: 04/7 - 08/07)

- 1) Avances en codificación y dudas técnicas.

Entrega Final (Fecha: 10/07)

- 1) Presentación grupal del proyecto.
- 2) Explicación del código.
- 3) Decisiones de diseño.

Endpoints

Registro de Cliente

POST /register

Request

```
{
  name: "Aldo",
  lastname: "Silvestre",
  age: 32,
  email: "aldosilvestre@email.com"
  password: "12345678",
  repeat_password: "12345678"
}
```

Response (HTTPSTATUS 204)

```
{ }
```

Iniciar Sesión

POST /login

Request

```
{
  username: "test@usuario.com",
  password: "12345678"
}
```

Response (HTTPSTATUS 200)

```
{
  access_token:
    "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ",
  is_admin: false,
  username: "test@usuario.com"
}
```

Realizar una reserva

POST /reserves/new

Request

```
{
  nro_sala: 15,
  date: "2022-05-16"
  time: "21:30",
}
```

Response (HTTPSTATUS 204)

```
{ }
```

Modificar reserva

PATCH /reserves/modify/{nro}

PathParam (nro): Se enviará el número de reserva a modificar.

Request

```
{
  new_time: "21:00",
}
```

```
new_date: "2022-05-23"
}
```

Response (HTTPSTATUS 200)

```
{ }
```

Obtener las entradas del cliente

```
GET /tickets/get
```

QueryParams: Opcionales de filtro, fecha, película, hora, usuario (en caso de admin), si no se envía nada se traerá todo lo que los permisos permitan.

Response (HTTPSTATUS 200)

```
[{
  movie: "ironman",
  time: "22:00",
  date: "20/05/2022"
}, {
  movie: "thor",
  time: "20:30",
  date: "21/06/2022"
}]
```

Creación Sala

```
GET /rooms/new
```

Request

```
{
  movie: "dragon ball",
  time: "22:30",
}
```

Response (HTTPSTATUS 204)

```
{ }
```

Modificación Sala

```
PUT /room/modify
```

Request

```
{
  nro_room: "123",
  new_movie: "hercules",
  new_time: "21:00"
}
```

Response (HTTPSTATUS 204)

```
{ }
```

Baja de Sala

```
DELETE /room/delete/{nro}
```

PathParam (nro): Se enviará el número de sala que se eliminará.

Request

```
{ }
```

Response (HTTPSTATUS 204)

```
{ }
```

Modificar descuentos

Request

```
{
  day: "Lunes",
  new_discount: 0.2
}
```

Response (HTTPSTATUS 204)

```
{ }
```

Ver salas

```
GET /rooms/get
```

QueryParams: Se podrá agregar como filtro película, día y hora.

Request

```
{ }
```

Response (HTTPSTATUS 200)

```
[{  
  nro: 52,  
  date: "2022-06-18",  
  movie: "dragon ball, Broly",  
  time: "22:30",  
},{  
  nro: 25,  
  date: "2022-06-21",  
  movie: "101 dalmatas",  
  time: "17:00"  
}]
```

Manejo de Errores

Todos los resultados de las llamadas se recibirán por códigos HTTP mayores al 400, con su respectivo mensaje de error.