

Requerimientos

Posibles Usuarios:

Usuario Visualizador (aquel usuario que aún no está registrado en la plataforma para quiere visualizar y buscar información)

Usuario Suscriptor: Usuario Registrado en la plataforma, el cual puede acceder a diferentes ofertas y a visualizar las películas.

Usuario Cinema: Usuario registrado en la plataforma relacionado a los cines, el cual puede publicar ofertas, dar y quitar puntos entre otras acciones.

Usuario Administrador: Usuario que gestiona la plataforma (CRUD de diferentes entidades)

1. Pantalla Principal

Página de inicio con información de la plataforma

(puede ser código quemado junto con información de relleno)

Barra superior o sección para diferentes botones, links o los que consideren.

Acciones Posibles: (botones)

- Ingresar a la plataforma
- Registrarse en la plataforma (Aún no estás registrado?)
- Visualizar Películas disponibles
- Visualizar Ofertas disponibles

Permisos de la pantalla:

- Cualquier Usuario

*** Cuando el usuario esté logueado, que se pueda visualizar en algún lugar, la opción de:

- Ver perfil
- Cerrar sesión
-

** Tener en cuenta la funcionalidad de Cerrar Sesión

2. Pantalla de Ingreso a la Plataforma

Pantalla que da posibilidad al usuario de ingresar a la plataforma con su cuenta.

****Debería mantener la opción de la pantalla de inicio para poder navegar a las diferentes pantallas como registrarse-visualizar películas- visualizar ofertas.**

Formulario de ingreso:

Usuario (máximo 100 caracteres)

Contraseña: (máximo 20 caracteres)

Si el usuario ingresa correctamente sus credenciales es dirigido a la pantalla inicial del usuario

**** Cuando el usuario ya está logeado, la página debería redirigir a la pantalla inicial del usuario.**

Endpoint para el formulario:

tipo de petición: Post

Url: {url}/users/login

Campos:

username_user_cinema

password_user_cinema

Permisos de la pantalla:

- Cualquier Usuario

Response 200 (usuario se encuentra con estado 1 - activo)

Status: **OK**

data:

id_user_cinema

name_user_cinema

surname_user_cinema

email_user_cinema

type_user_cinema

Response 401 (No se encuentra el usuario o se registró con unas credenciales inválidas - usuario o contraseña)

Status: **UNAUTHORIZED**

data:

error_response: "Credenciales inválidas" si no existe el usuario o la contraseña no coincide.

Response 403 (usuario se encuentra con estado 0 - dado de baja)

Status: **FORBIDDEN**

data:

error_response: "El usuario ha sido dado de baja"

posibles type_user_cinema:

- 1) Administrador
- 2) Suscriptor
- 3) Cinema

3. Pantalla de Registro

Pantalla que da posibilidad al usuario de registrarse en la plataforma.

****Debería mantener la opción de la pantalla de inicio para poder navegar a las diferentes pantallas como registrarse-visualizar películas- visualizar ofertas.**

Formulario de Registro:

Usuario (máximo 100 caracteres)

Contraseña: (máximo 20 caracteres)

Nombres: (máximo 200 caracteres)

Apellidos: (máximo 200 caracteres)

email: (máximo 200 caracteres)

Si el usuario registra los datos correctamente, lo redirige a la página inicial de su cuenta.

**** Cuando el usuario ya está logeado, la página debería redirigir a la pantalla inicial del usuario.**

Permisos de la pantalla:

- Cualquier Usuario

Endpoint para el formulario:

tipo de petición: Post

Url: {url}/users/register

Campos:

username_user_cinema

password_user_cinema

name_user_cinema

surname_user_cinema

email_user_cinema

Response 409 (conflicto con los registros, el nombre del usuario no se encuentra disponible / el usuario con el email ya se encuentra registrado)

Status: **CONFLICT**

data: (excluyente)

- error_response: "El usuario ya existe en la plataforma"
- error_response: "Ya existe un usuario con el correo"

Response 200 (Se registra el usuario con éxito)

Status: **OK**

data:

id_user_cinema
name_user_cinema
surname_user_cinema
email_user_cinema
type_user_cinema

4. Visualizar Películas Disponibles - Filtros

Pantalla que da posibilidad a cualquier usuario(Registrado o no) de visualizar las películas que se encuentran en la plataforma.

**Debería mantener la opción de poder navegar a las diferentes pantallas como Ver Perfil- visualizar ofertas si está logueado

*** Debería mantener la opción de poder navegar a las diferentes pantallas como Ingresar a la plataforma- Registrarse- visualizar ofertas si el usuario no está logueado.

Si no existen películas, debería salir un letrero que indique que no es posible acceder al catálogo de películas o algo

SI ES USUARIO ADMINISTRADOR MOSTRAR ACTOR/DIRECTOR/CATEGORÍA

SI ES ADMINISTRADOR:

un botón que lo lleve a la página para EDITAR PELÍCULA.

Permisos de la pantalla:

- Cualquier Usuario

Endpoint para listar películas:**tipo de petición:** get**Url:** {url}/films/list**Campos:****Response 204 (no existe contenido disponible)**Status: **NO CONTENT**

data:

- error_response: "No existe contenido disponible"

Response 200 (Devuelve la lista de películas con la información)Status: **OK**

data:

```
id_film: (array)
  title_film
  description_film
  release_date_film
  director_films: (array)
    id_director
    name_director
  actor_films:(array)
    id_actor
    name_actor

  category_films:(array)
    id_category
    name_category
```

Endpoint para Filtrar películas por categoría: (Posible)**tipo de petición:** get**Url:** {url}/films/list/{category}**Campos:****Response 204 (no existe contenido disponible para el filtro indicado)**Status: **NO CONTENT**

data:

- error_response: "No existe contenido disponible"

Response 200 (Devuelve la lista de películas con la información con las películas que coinciden con el filtro)Status: **OK**

data:

```
name_category
id_film: (array)
  title_film
  description_film
  release_date_film
  director_films: (array)
    id_director
    name_director
  actor_films:(array)
    id_actor
    name_actor

category_films:(array)
  id_category
  name_category
```

5. Visualizar Ofertas Disponibles

Pantalla que da posibilidad a cualquier usuario(Registrado o no) de visualizar las ofertas que se encuentran en la plataforma.

****Debería mantener la opción de poder navegar a las diferentes pantallas como Visualizar Películas disponibles- Perfil si está logueado**

***** Debería mantener la opción de poder navegar a las diferentes pantallas como Ingresar a la plataforma- Registrarse- visualizar películas Disponibles si el usuario no está logueado.**

Si no existen ofertas, debería salir un letrero que indique que no es posible acceder a las ofertas disponibles o algo

SI ES USUARIO ADMINISTRADOR MOSTRAR ACTOR/DIRECTOR/CATEGORÍA

Permisos de la pantalla:

- Cualquier Usuario

Endpoint para listar Ofertas:

tipo de petición: get

Url: {url}/offers/list

Campos:

Response 204 (no existen ofertas disponibles)

Status: **NO CONTENT**

data:

- error_response: "No existen ofertas disponibles"

Response 200 (Devuelve la lista de las ofertas con la información)

Status: **OK**

data:

```
id_offer: (array)
  description_offer
  deadline_offer
  points
  add_points
  sub_points
  id_cinema:
    web_cinema
    address_cinema
    city_cinema
```

6. Modificar Datos de Usuario/darse de Baja

A esta pantalla se llega cuando el usuario clickea sobre ver perfil/ver cuenta/

Pantalla que da la posibilidad al usuario de visualizar sus datos y poder editarlos.

****Debería mantener la opción de poder navegar a las diferentes pantallas como Ver Películas o visualizar ofertas**

***** Además de poder visualizar su información puede existir un botón para poder darse de baja.**

***** VALIDACIÓN:** si el usuario trata de acceder a la información que no es la de él, debe salir un error.

SI ES USUARIO ADMINISTRADOR MOSTRAR ACTOR/DIRECTOR/CATEGORÍA

Para esta pantalla se van a usar 3 servicios

```
get: información del usuario
put: editar información
delete: darse de baja
```

Permisos de la pantalla:

- Usuario Administrador
- Usuario Suscriptor
- Usuario Cinema

Endpoint para traer la información del usuario:

tipo de petición: get

Url: {url}/users/info/{id}

Campos:

Response 204 (No se encontró la información solicitada)

Status: **NO CONTENT**

data:

- error_response: "Error al acceder a la información solicitada"

Response 200 (Devuelve la información del usuario si es Admin)

Status: **OK**

data:

username_user_cinema(Static)

password_user_cinema

name_user_cinema

surname_user_cinema

email_user_cinema (Static)

type_user_cinema (texto)

Response 200 (Devuelve la información del usuario si es suscriber)

Status: **OK**

data:

username_user_cinema(static)

password_user_cinema

name_user_cinema

surname_user_cinema

email_user_cinema (Static)

type_user_cinema (texto)

current_points (static)

Response 200 (Devuelve la información del usuario si es cinema)

Status: **OK**

data:

username_user_cinema(static)

password_user_cinema

name_user_cinema

surname_user_cinema

email_user_cinema (Static)

type_user_cinema (texto)
web_cinema
address_cinema
city_cinema
id_user_cinema

Endpoint para actualizar la información del usuario:

tipo de petición: put

Url: {url}/users/edit

Campos:

id_user_cinema

****Depende de la entidad**

administrador y suscriptor:

password_user_cinema
name_user_cinema
surname_user_cinema

cinema:

password_user_cinema
name_user_cinema
surname_user_cinema
web_cinema
address_cinema
city_cinema
id_user_cinema

Response 201

Status: **CREATED**

data: "Información Actualizada"

Endpoint para eliminar el usuario (Darse de baja):

tipo de petición: delete

Url: {url}/users/edit

Campos:

id_user_cinema

Response 202 (se da de baja al usuario)

Status: **ACCEPTED**

data: ""

Response 403 (no se puede dar de baja al usuario, problemas internos)

Status: **FORBIDDEN**

data: "error al darse de baja"

**** preguntar si esta seguro de darse de baja**

7. CRUD Actor/Director/Categoría

A esta pantalla se llega cuando el usuario Administrador clickea sobre Actor o Director o Categorías

la pantalla muestra una tabla con la información correspondiente, si es actor director o es una categoría.

****Debería mantener la opción de poder navegar a las diferentes pantallas como Ver Películas o visualizar ofertas o perfil.**

la tabla puede ser genérica ya que cada una de estas entidades tiene id_xxxxx y name_xxxxxx siendo xxxxxx = Actor/Director/categoría
ejemplo de una tabla crud básica con angular:

<https://github.com/marinantonio/angular-mat-table-crud>

Permisos de la pantalla:

- Usuario Administrador

Endpoint para listar xxxxx:

tipo de petición: get

Url: {url}/category/list

Url: {url}/director/list

Url: {url}/actor/list

Campos:

Response 204 (no existen xxxxx disponibles)

Status: **NO CONTENT**

data: "no existe xxxxx en la plataforma"

Response 200 (Devuelve la lista de xxxxxx con la información)

Status: **OK**

```
data: array (
    id_xxxxx
    name_xxxxxx
)
```

Endpoint para crear xxxxx:

tipo de petición: post

Url: {url}/ category/

Url: {url}/ director/

Url: {url}/ actor/

Campos:

name_category/name_director/name_actor

Response 400 (error al momento de crear xxxxxx)

Status: **ERROR**

data: (excluyente)

- error_response: "Error al momento de crear xxxxx"

Response 200 (Se registra xxxxxx con éxito)

Status: **OK**

data: "Registro Creado"

Endpoint para editar xxxxx:

tipo de petición: put

Url: {url}/ category/

Url: {url}/ director/

Url: {url}/ actor/

Campos:

id_category/id_director/id_actor

name_category/name_director/name_actor

Response 400 (error al momento de actualizar xxxxxx)

Status: **ERROR**

data: (excluyente)

- error_response: "Error al momento de actualizar xxxxx"

Response 200 (Se realizó la actualización con éxito)

Status: **OK**

data: "Registro actualizado"

Endpoint para eliminar xxxxx:

tipo de petición: delete

Url: {url}/ category/

Url: {url}/ director/

Url: {url}/ actor/

Campos:

id_category/id_director/id_actor

Response 400 (error al momento de eliminar xxxxxx)

Status: **CONFLICT**

data:

error_response: "Error al momento de eliminar xxxxxxxx"

Response 200 (Se Eliminó el registro con éxito)

Status: **OK**

data: "Registro Eliminado"

8. Adminstrador - Cinemas (crear) Actor/Director/

#TODO

name_user_cinema
surname_user_cinema
email_user_cinema (Static)
type_user_cinema (texto)
web_cinema
address_cinema
city_cinema
id_user_cinema

9. Cinemas CRUD oferta

#TODO

10. Cinemas Asignar oferta a un suscriptor

#TODO

11. Administrador CRUD película

#TODO