

**Escuela Superior Politécnica del Litoral**  
**Programación Orientada a Objetos**  
**Proyecto del segundo parcial**  
**Msc. Verónica Duarte**  
**II PAO 2024**

Se le solicita desarrollar una aplicación móvil para un cine que permita comprar entradas para funciones de películas.



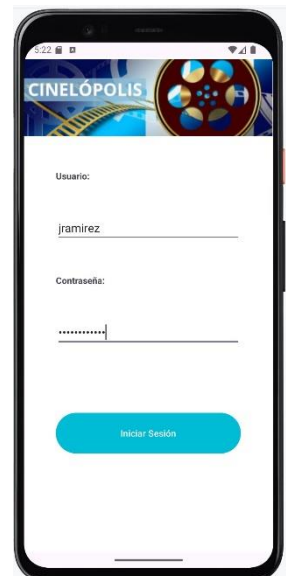
1. Al inicio, se muestran tres opciones: Iniciar sesión y Próximos Estrenos con las funcionalidades que se muestran a continuación.

### **Iniciar Sesión**

2. Esta opción está dirigida a los usuarios que desean comprar entradas.

Al dar clic en ese botón se mostrará una nueva actividad que solicita los datos al usuario para iniciar sesión. Si las credenciales (usuario y password) son inválidas, debe crear manualmente una excepción propia verificada de tipo **CredecialesInvalidasException**, el mensaje de la excepción debe indicar que el usuario o la contraseña son incorrectos. Muestre el mensaje al usuario usando mensajes temporales con `Toast.makeText()`.

Los datos de los usuarios están en el archivo **usuarios.txt**.



## Películas



3. Al iniciar sesión, se muestra la actividad que permite consultar las películas disponibles. En esta actividad se cargarán los posters de las películas en botones usando ImageView.

Los datos de las películas estarán en el archivo **películas.txt** con el formato:  
***idPelícula, título, duración, nombreArchivImagen***

Los archivos de posters de películas debe buscarlos en internet. Así también puede modificar los títulos de las películas e incluir sus propios datos.

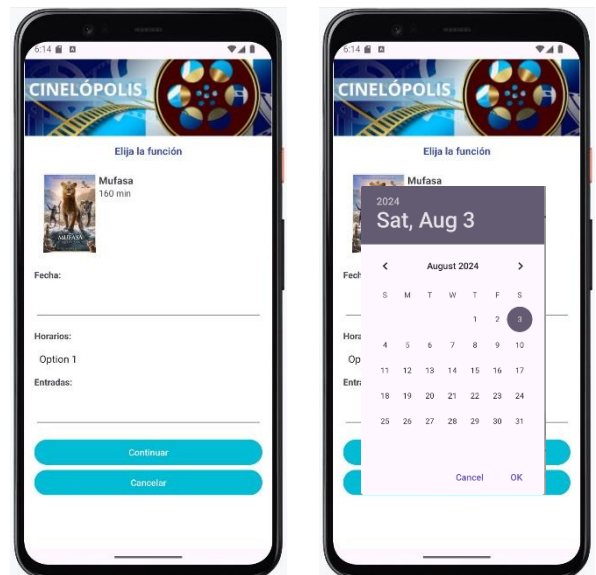
Debe usar un scrollView para mostrar los posters de todas las películas. Al dar clic en algún poster (botón), se mostrará la siguiente Activity.

## Elegir función

4. En esta Activity se muestra la información de la película seleccionada en el Activity anterior.

Además, debe elegir la fecha de la función a partir del calendario.

Al elegir una fecha, en el Spinner se cargarán **solo** las funciones disponibles en esa fecha.



Las funciones se encuentran en el archivo **funciones.txt**. De cada función, se conoce:

***idFuncion, idPelícula, fecha, hora, sala.***

Se mostrarán solo la hora y la sala donde se realizará la función (toString).

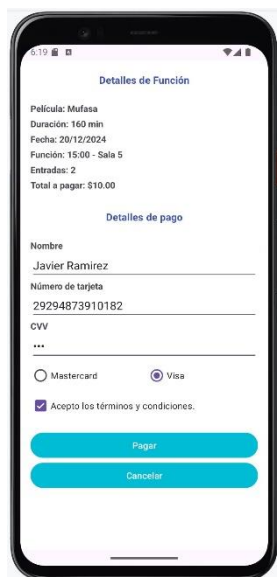
También debe ingresar el número de entradas que desea comprar.

Si elige una fecha que no tiene funciones asociadas entonces se creará una excepción **no verificada** llamada **sinFuncionException**. El mensaje de la excepción debe indicar que no hay funciones en la fecha seleccionada. Muestre el mensaje al usuario usando mensajes temporales con Toast.makeText().

Al dar clic en Continuar debe verificar primero que haya una fecha elegida, un horario seleccionado y un valor entero ingresado en el campo Entradas. Si no es así, se crea una excepción **verificada** llamada **datosIncompletosException**. El mensaje de la excepción debe indicar que no están todos los datos completos. Muestre el mensaje al usuario usando mensajes temporales con Toast.makeText().

En cambio, si todo va bien, se muestra la nueva Activity para realizar el pago. Al dar clic en Cancelar, debe regresar a la actividad que muestra las películas disponibles (punto 3).

## Pago



5. En esta activity, primero se muestran los detalles de la función elegida. Y luego se muestran los campos necesarios del método de pago, que debe registrar el usuario.

Una entrada cuesta **siempre** 5.00 dólares.

Al dar clic en el botón Pagar, primero se debe validar que el usuario haya ingresado todos los campos de pago. Si no es así, se crea una excepción **verificada** llamada **datosIncompletosException**. El mensaje de la excepción debe indicar que no están todos los datos completos. Muestre el mensaje al usuario usando mensajes temporales con Toast.makeText().

En cambio, si todo está bien se procede a crear el objeto Pago. Además, se escriben los datos del pago en el archivo **pagos.txt** con el formato:

***idPago, idFuncion, nombre, numero de tarjeta, tipo, numeroEntradas, totalPagar***

Además, el objeto pago debe serializarse en un archivo cuyo nombre tendrá el formato: **función+idFuncion.bin**

Por ejemplo: **funcion34522**

### Próximos Estrenos

6. En esta opción se muestran los datos de los próximos estrenos de películas. Los datos están en el archivo **estrenos.txt**. Debe leer el archivo y cargarlos en esta activity dentro de un TableLayout.

De los estrenos se conoce:

***idPelicula, titulo, fecha***

Al inicio se muestran tal y como se obtienen del archivo. Pero si da clic en el botón de Ordenar por nombre, se reconstruye el TableLayout para mostrar los elementos ordenados por el título de la película, en orden ascendente.

Ubique el TableLayout en un scrollView para que se muestren todos los datos en el espacio de la activity que le corresponde.



Al dar clic en Salir debe regresar a la activity inicial (Punto 1).

Archivos con datos de entrada:

- usuarios.txt (usar archivo existente)

- peliculas.txt (usar archivo existente)
- funciones.txt (usar archivo existente)

Archivos con datos de salida

- pagos.txt
- archivos binarios de funciones.

## CONSIDERACIONES

- Debe seguir la distribución de los elementos en las pantallas, como se indica en cada gráfico de este documento, pero puede agregar su propio diseño. No agregar estilo a sus formularios equivale a puntos menos en la evaluación final.
- Debe trabajar con objetos de las clases correspondientes a **Funcion, Pelicula, Usuario, Pago**, excepciones propias, etc.
- Debe procurar que el usuario entienda lo que está pasando en cada momento, de **feedback** al usuario con mensajes.
- Recuerde hacer uso de las herramientas mencionadas.
- Recuerde que debe implementar en su código el manejo de excepciones necesarias en cada caso. Esto quiere decir que su programa no debe caerse y si ocurre alguna excepción debe mostrarse al usuario algún mensaje personalizado más no la excepción en sí. Por ejemplo: “Problemas técnicos. Estamos resolviendo”.
- **Si el proyecto no está completo, no se revisará.**
- La nota final del proyecto será igual a:  $\text{nota} \times \text{sustentación} \times \text{participación}$ .

**PAR#\_PROY2P\_Apellido1\_Apellido2\_Apellido3**

- Debe presentar su documentación completa, en caso contrario, su proyecto no será revisado.

## Entregables

- Proyecto java.
- Diagrama de clases.
- Documentación: Reporte.
- Archivo apk de la aplicación.

Nota: El proyecto se presentará en un dispositivo móvil.