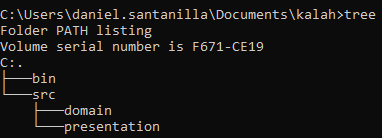
# LABORATORIO #5: INTERFAZ

## DESARROLLO

### **Directorios**

El objetivo de este punto es construir un primer esquema para el juego **Kalah**.

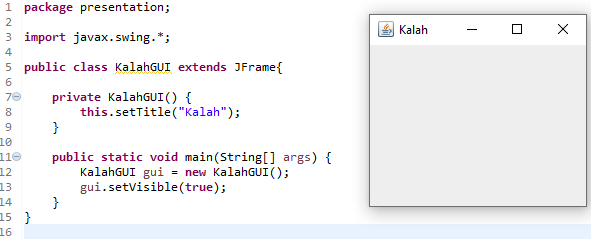
1. Preparen un directorio llamado **Kalah** con los directorios src y bin y los subdirectorios presentación y dominio.



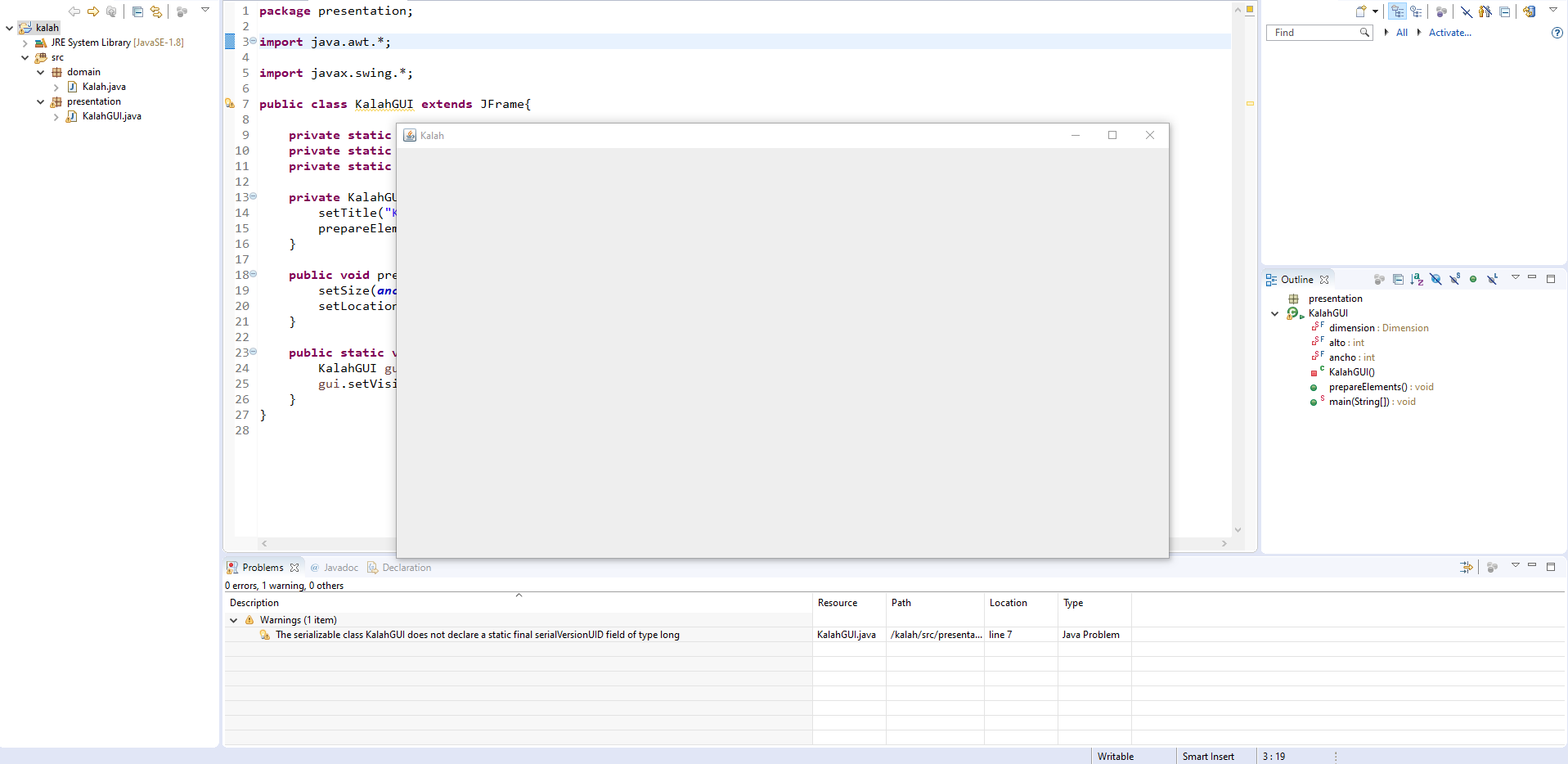
### **Ciclo 0: Ventana vacía – Salir**

El objetivo es implementar la ventana principal de **Kalah** con un final adecuado a la aplicación desde el icono de cerrar (pidiendo confirmación a usuario). Utilizar el esquema de prepareElements-prepareActions

1. Construyan el primer esquema de la ventana de **Kalah** únicamente con el título “**Kalah**”. Para esto implemente la clase **Kalah**GUI como un JFrame con su creador (que sólo coloca el título) y el método main que crea un objeto **Kalah**GUI y lo hace visible. Ejecútenlo. Capturen la pantalla. (Si la ventana principal no es la inicial en su diseño, después deberán mover el main al componente visual correspondiente)



1. Modifiquen el tamaño de la ventana para que ocupe un cuarto de la pantalla y ubíquenla en el centro (prepareElements). Capturen esa pantalla



1. Traten de cerrar la ventana. ¿Termina la ejecución? ¿Qué deben hacer para terminar la ejecución? ¿Por qué?

Si se cierra la ventana la ejecución no va a terminar, para que esta termine se debe añadir un método que se encargue de esto.

1. Estudien en JFrame el método setDefaultCloseOperation. ¿Para qué sirve? ¿Cómo lo usarían en este caso?

Este método sirve para establecer la operación que ocurre cuando el usuario inicia un cierre en la ventana. En este caso se usaría para terminar la ejecución cuando el usuario cierre la ventana.

1. Preparen el “oyente” correspondiente al icono cerrar que le pida al usuario que confirme su selección. Para esto Implementen parcialmente el método prepareActions y el método asociado a la acción (exit). Ejecuten el programa y cierren el programa. Capturen las pantallas.

Texto

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente

### **Ciclo 1: Ventana con menú – Salir**

El objetivo es implementar un menú clásico para la aplicación con un final adecuado desde la opción del menú para salir. El menú debe ofrecer mínimo las siguientes opciones: Nuevo, Abrir – Salvar y Salir. Incluyan los separadores de opciones. No olviden los criterios de calidad del código.

1. Defina como atributos los componentes visuales necesarios del menú.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

1. Construyan la forma del menú propuesto (prepareElements - prepareElementsMenu). Ejecuten. Capturen la pantalla.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

1. Preparen el “oyente” correspondiente al icono cerrar con confirmación (prepareActions - prepareActionsMenu). Ejecuten el programa y salgan del programa. Capturen las pantallas.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

Interfaz de usuario gráfica

Descripción generada automáticamente

### **Ciclo 2: Salvar y abrir**

El objetivo es preparar la interfaz para las funciones de persistencia

1. Detallen el componente JFileChooser especialmente los métodos: JFileChooser, showOpenDialog, showSaveDialog, getSelectedFile.

* JFileChooser proporciona un mecanismo simple para que el usuario elija un archivo.
* showOpenDialog Aparece un cuadro de diálogo de selección de archivos "Abrir archivo".
* showSaveDialog Aparece un cuadro de diálogo de selección de archivos "Guardar archivo".
* getSelectedFile Devuelve el archivo seleccionado.

1. Implementen parcialmente los elementos necesarios para salvar y abrir. Al seleccionar los archivos indique que las funcionalidades están en construcción detallando la acción y el nombre del archivo seleccionado.

Texto

Descripción generada automáticamente

1. Ejecuten las dos opciones y capturen las pantallas más significativas.

Interfaz de usuario gráfica, Texto, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

### **Ciclo 3: Forma de la ventana principal**

El objetivo es codificar el diseño de la ventana principal (todos los elementos de primer nivel)

1. Definan como atributos privados todos los componentes visuales necesarios.
2. Continúe con la implementación del método prepareElements(). Para la zona del Board defina un método prepareElementsBoard() y un método refresh() que actualiza la vista del tablero considerando, por ahora, el tablero inicial por omisión (definido en el trabajo en clase) Este método lo vamos a implementar realmente en otros ciclos. Ejecuten y capturen esta pantalla.

Tabla

Descripción generada automáticamente

### **Ciclo 4: Cambiar color**

El objetivo es implementar este caso de uso.

1. Expliquen los elementos (vista – controlador) necesarios para implementar este caso de uso.
2. Detallen el comportamiento de JColorChooser especialmente el método estático showDialog

* JColorChooser: Proporciona un panel de controles diseñado para permitir que un usuario manipule y seleccione un color.
* showDialog: Muestra un cuadro de diálogo con selector de colores.

1. Implementen los componentes necesarios para cambiar los colores del juego.
2. Ejecuten el caso de uso y capture las pantallas más significativas.

Tabla

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente

Imagen que contiene Calendario

Descripción generada automáticamente

### **Ciclo 5: Modelo Kalah**

El objetivo es implementar la capa de dominio para **Kalah**.

1. Construyan los métodos básicos del juego (No olvide MDD y TDD)
2. Ejecuten las pruebas y capturen el resultado.

### **Ciclo 6: Jugar**

El objetivo es implementar el caso de uso jugar.

1. Adicionen a la capa de presentación el atributo correspondiente al modelo.
2. Perfeccionen el método refresh() considerando la información del modelo de dominio.
3. Expliquen los elementos necesarios para implementar este caso de uso.
4. Implementen los componentes necesarios para jugar.
5. Ejecuten el caso de uso y capture las pantallas más significativas.

### **Ciclo 7: Reiniciar**

El objetivo es implementar este caso de uso.

1. Expliquen los elementos a usar para implementar este caso de uso. El objetivo es implementar este caso de uso. 1. Expliquen los elementos a usar para implementar este caso de uso 2. Implementen los elementos necesarios para cambiar la configuración del juego. 3. Ejecuten el caso de uso y capture las pantallas más significativas.
2. Implementen los elementos necesarios para reiniciar
3. Ejecuten el caso de uso y capture las pantallas más significativas.

### **Ciclo 8: Cambiar la configuración del juego**

El objetivo es implementar este caso de uso.

1. Expliquen los elementos a usar para implementar este caso de uso
2. Implementen los elementos necesarios para cambiar la configuración del juego.
3. Ejecuten el caso de uso y capture las pantallas más significativas.

## Retrospectiva

### ¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes? (Horas/Nombre)

(/Angie Natalia Mojica)  
( /Daniel Antonio Santanilla)

### ¿Cuál es el estado actual del laboratorio? ¿por qué?

### Considerando las prácticas XP del laboratorio. ¿cuál fue la más útil? ¿por qué?

### ¿Cuál consideran fue el mayor logro? ¿Por qué?

### ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

### ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?