**PROGRAMACIÓN ORIENTADA A OBJETOS INTERFAZ**

**2019-01**

**Laboratorio 5/6**

**OBJETIVOS**

1. Desarrollar una mini aplicación gráfica

2. Experimentar el comportamiento de las ventanas JFrame, JDialog y JOptionPane

3.Seleccionar los lienzos más apropiados para un diseño:JPanel,JScrollPane,JTabbedPane

4. Revisar las posibilidades de los estilos:FlowLayout, BorderLayout y GridLayout

5. Ensayar algunos componentes básicos: JLabel, JTextField, JButton, JMenuBar,

6. Ensayar algunos componentes especiales: JFileChooser y JColorChooser

7. Implementar el esquema de manejo de eventos con clases anónimas

**ENTREGA**

1. Incluyan en un archivo .zip los archivos correspondientes al laboratorio. El nombre debe ser los dos apellidos de los miembros del equipo ordenados alfabéticamente.
2. En el foro de entrega deben indicar el estado de avance de su laboratorio y los problemas pendientes por resolver.
3. Deben publicar el avance al final de la sesión y la versión definitiva en la fecha indicada, en los espacios preparados para tal fin.

**CONTEXTO**

El objetivo es implementar el juego Kalah en versión usuario-computador.

El trabajo se debe hacer desde **CONSOLA**

|  |  |
| --- | --- |
| El propuesto por ustedes | El acordado en laboratorio |
| KalahGUI | Kalah |
|  |  |
| **Vista - Controlador** | **Modelo** |

**Para la capa de presentación NO deben hacer pruebas de unidad ni diagramas de secuencia**

**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 aplicación.

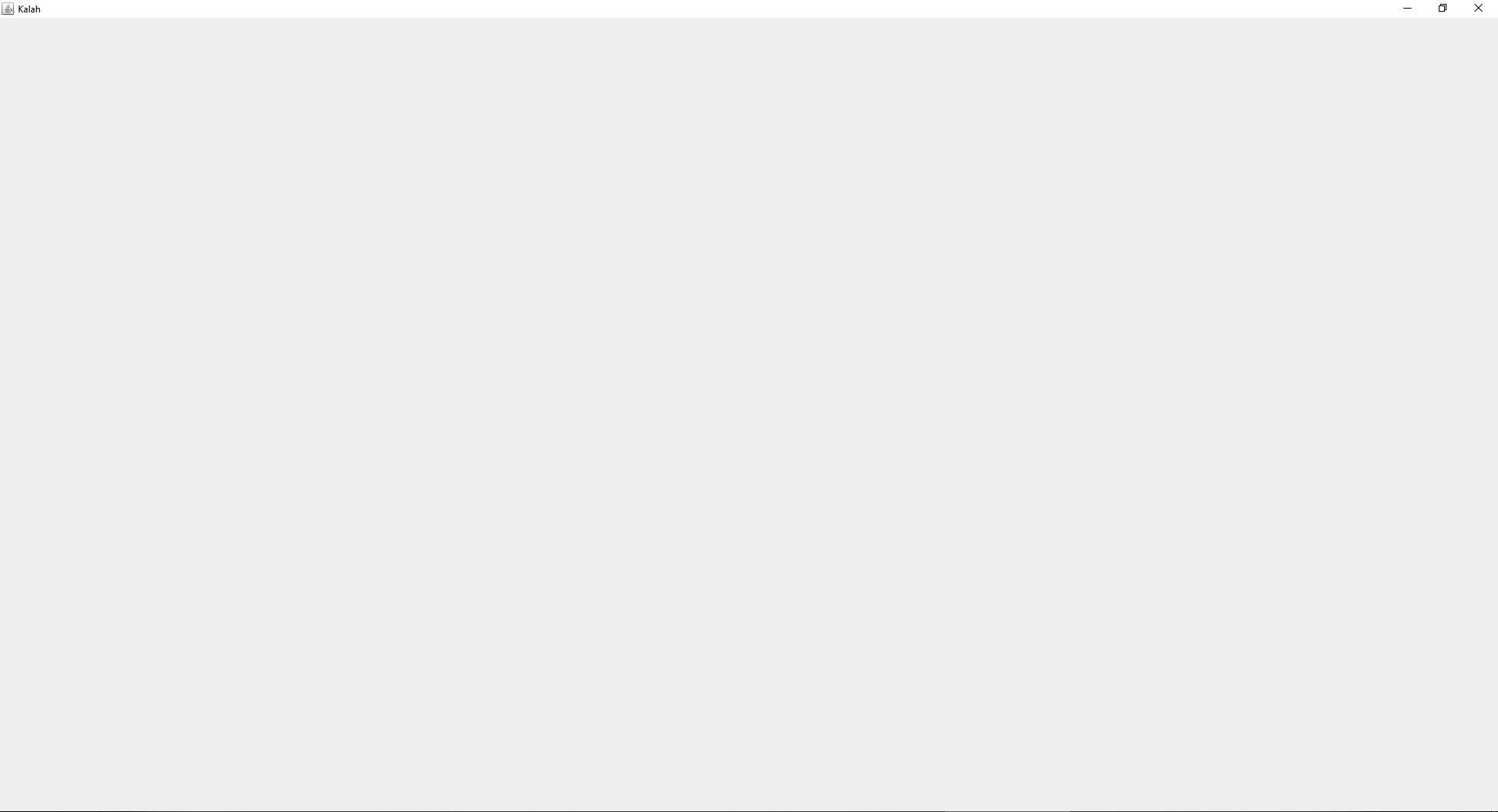
**Ciclo 0: Ventana vacía – Salir**

**[En \*.java y lab05.doc]**

El objetivo es implementar la ventana principal de **Kalah** con un final adecuado a la aplicación desde el icono de cerrar. Utilizar el esquema de prepareElementos- prepareAcciones.

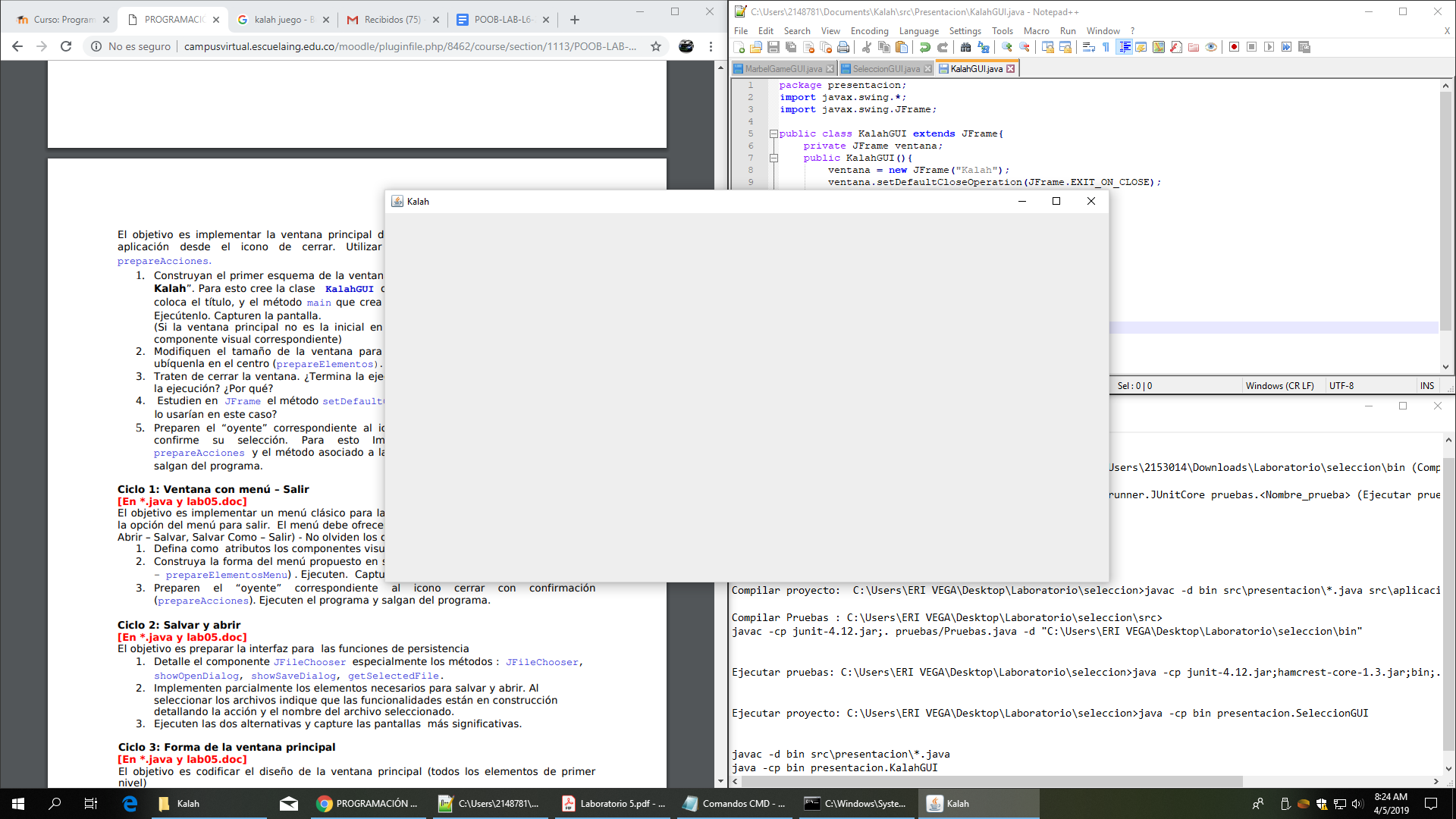
1. Construyan el primer esquema de la ventana de **Kalah** únicamente con el título “**Kalah**”. Para esto cree la clase **KalahGUI** como unJFramecon su creador, que sólocoloca el título, y el método main que crea un objeto **KalahGUI** y lo hace visible.

Ejecútalo. Capturen la pantalla.



(Si la ventana principal no es la inicial en su diseño, después moverá el main al componente visual correspondiente)

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



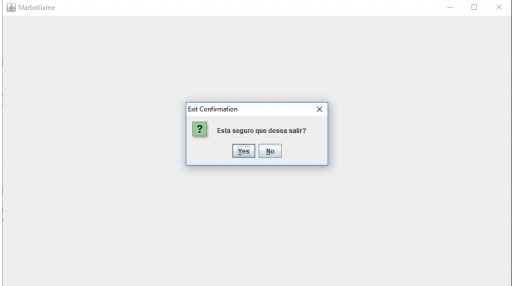
3. Traten de cerrar la ventana. ¿Termina la ejecución? ¿Qué deben hacer para terminar la ejecución? ¿Por qué?

* El programa no termina su ejecución ya que nosotros al momento de crearla tenemos que programar su salida, en este caso sería cuando hace click en la X.

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

* Este método sirve para programar la salida del JFrame al momento de hacer click en la X.
* setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

5. Preparen el “oyente” correspondiente al icono cerrar que le pida al usuario que confirme su selección. Para esto implementan parcialmente el método prepareAcciones y el método asociado a la acción (salga). Ejecuten el programa y salgan del programa.



**Ciclo 1: Ventana con menú – Salir**

**[En \*.java y lab05.doc]**

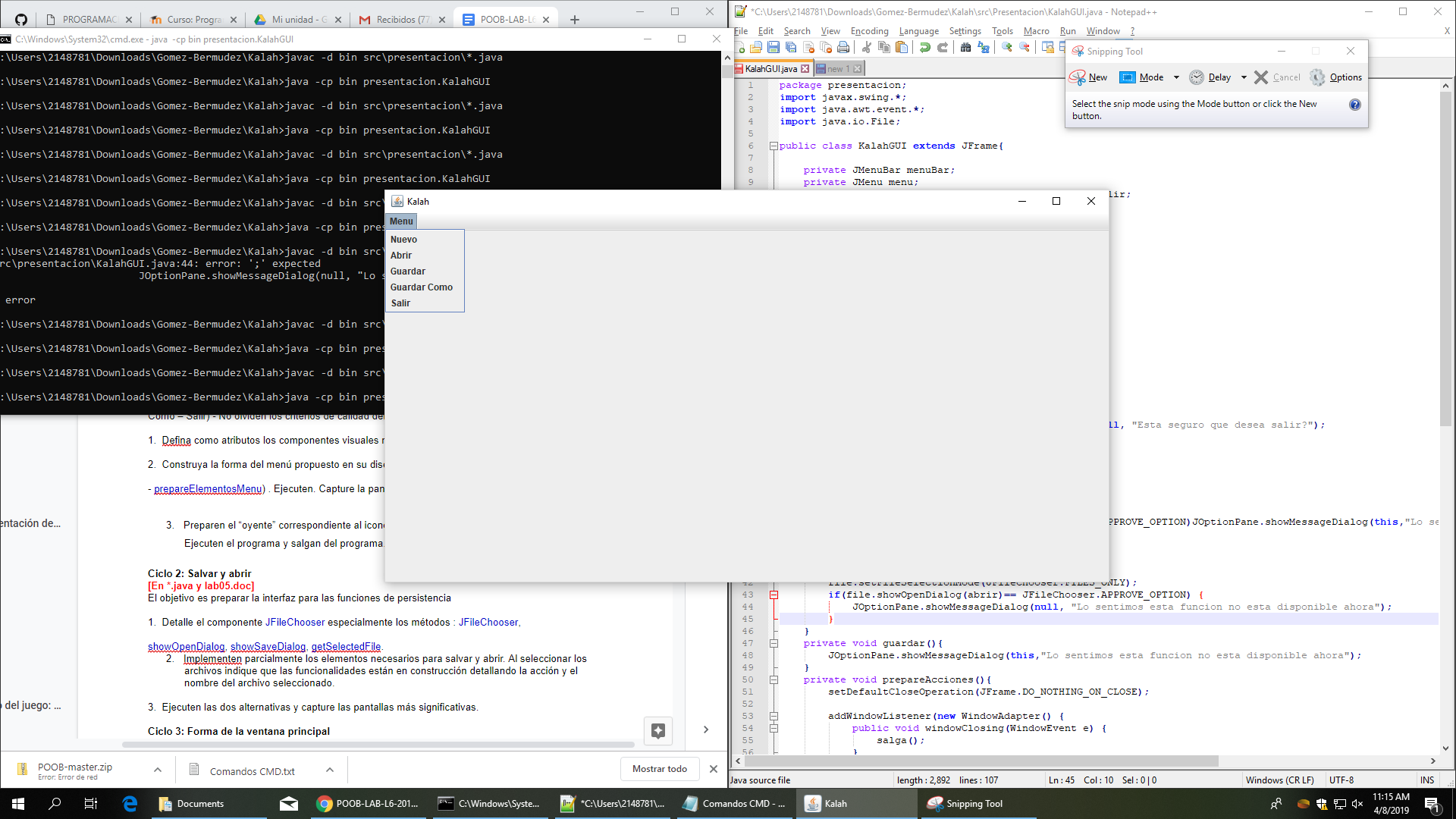
El objetivo es implementar un menú clásico para la aplicación con un final adecuado desde la opción del menú para salir. El menú debe ofrecer mínimo las siguientes opciones (Nuevo, Abrir – Salvar, Salvar Como – Salir) - No olviden los criterios de calidad del código.

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

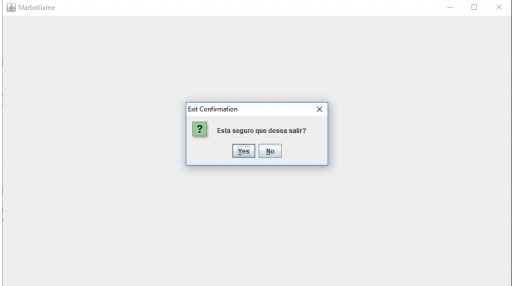
Se definieron un JMenu, JMenuItem y JMenuBar para los atributos

2. Construya la forma del menú propuesto en su diseño de interfaz (prepareElementos

- prepareElementosMenu) . Ejecuten. Capture la pantalla.



3. Preparen el “oyente” correspondiente al icono cerrar con confirmación (prepareAcciones). Ejecuten el programa y salgan del programa.



**Ciclo 2: Salvar y abrir**

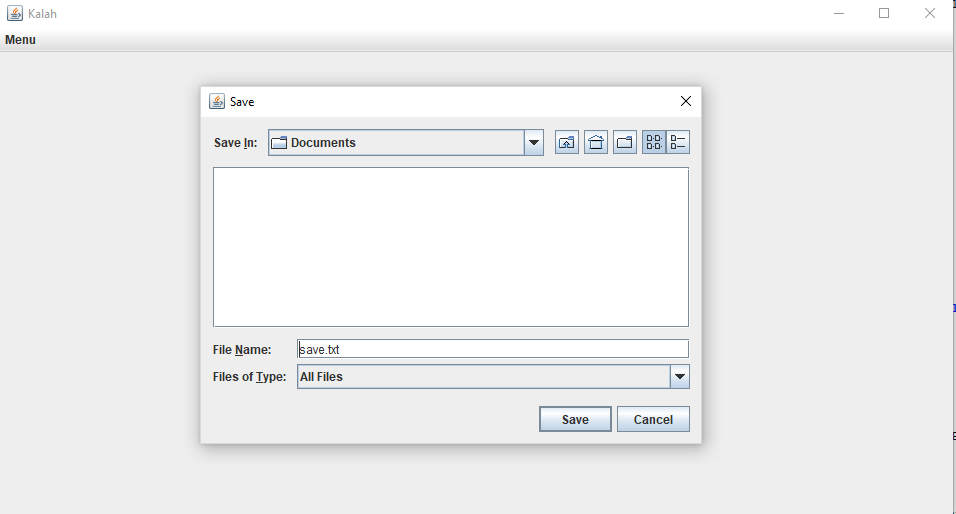
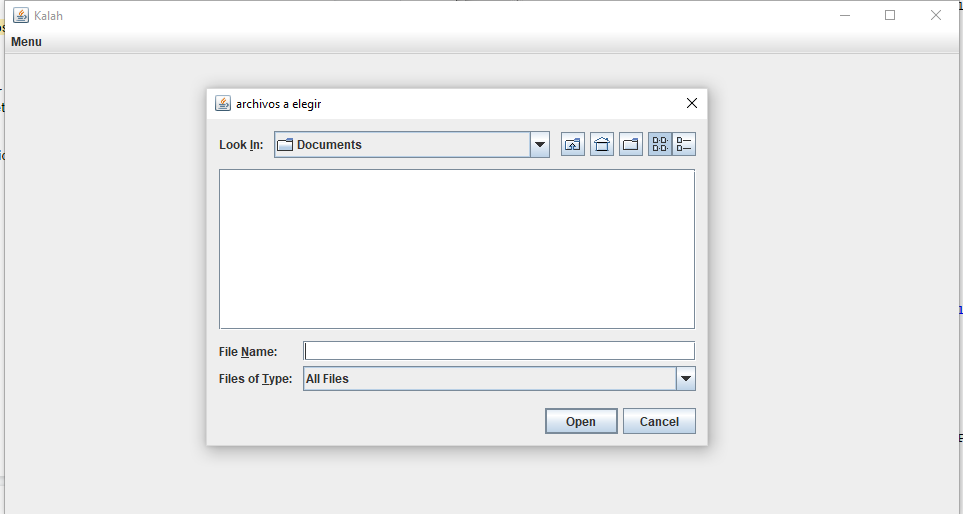
**[En \*.java y lab05.doc]**

El objetivo es preparar la interfaz para las funciones de persistencia

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

* JFileChooser, permite al usuario elegir un archivo en especifico.
* showOpenDialog, muestra un cuadro de diálogo del selector de archivos “Abrir”.
* showSaveDialog, muestra un cuadro de diálogo del selector de archivos “Guardar”.
* getSelectedFile, retorna el archivo seleccionado.

2.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.

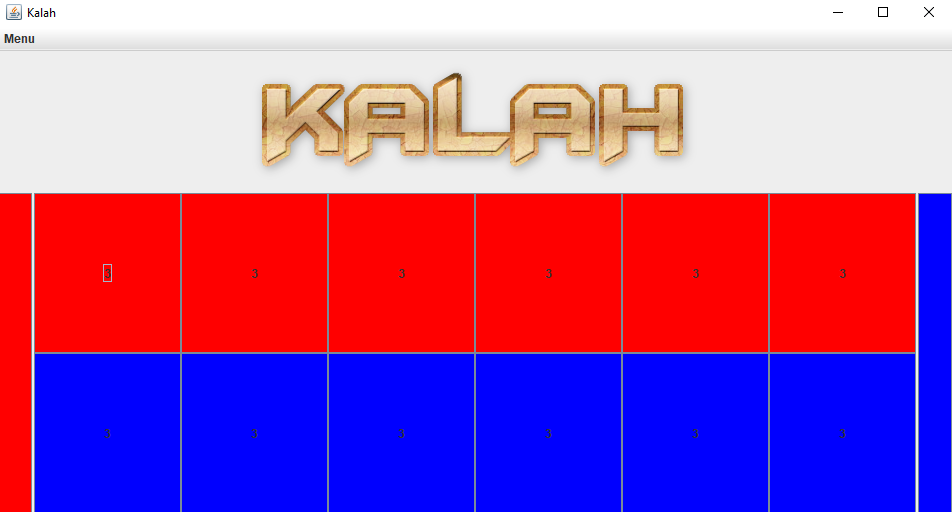
3. Ejecuten las dos alternativas y capture las pantallas más signific

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

**[En \*.java y lab05.doc]**

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

1. Definen como atributos privados todos los componentes visuales necesarios.

2. Continúe con la implementación del método prepareElementos(). Para la zona del tablero defina un método prepareElementosTablero y un método refresque() que actualiza la vista del tablero considerando, por ahora, un tablero inicial por omisión (**azul/usuario – rojo/computador, 6 casas x 3 semillas**) Este método lo vamos a implementar realmente en otros ciclos. Ejecuten y capturen esta pantalla

**Ciclo 4: Cambiar color**

**[En \*.java y lab05.doc]**

El objetivo es implementar este caso de uso.

1. Expliquen los elementos necesario para implementar este caso de uso.

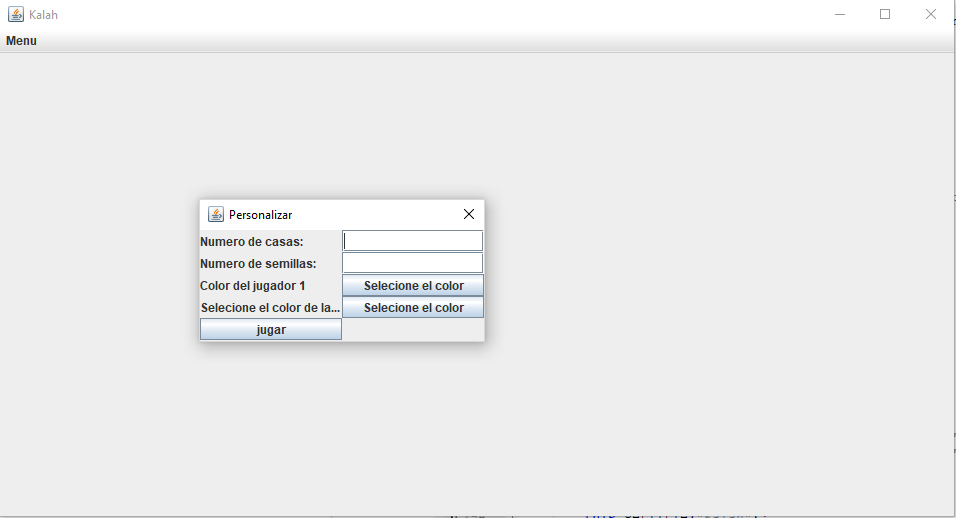
Los elementos que implementamos para este caso de uso fueron los relacionados con JColorChooser, luego de esto se genero la interacción con el JDialog

2. Detalle el comportamiento de JColorChooser especialmente el método estático showDialog

* JColorChooser proporciona un panel de controles diseñados para permitir al usuario manipular y seleccionar un color.
* showDialog, muestra un cuadro de diálogo y bloques de selector de color modal hasta que el cuadro de diálogo esté oculto. Si el usuario presiona el botón “Aceptar”, este método oculta/elimina el diálogo y devuelve el color seleccionado.Si el usuario presiona el botón “Cancelar” cierra el cuadro de diálogo sin presionar “Ok”, este método oculta o elimina el diálogo y devuelve nulo.

3. Implementen los componentes necesarios para cambiar el color de las casas y almacenes.

4. Ejecuten el caso de uso y capture las pantallas más significativas.



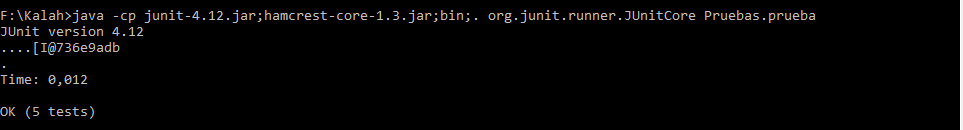
**Ciclo 5: Modelo Kalah**

**[En \*.java y lab05.doc]**

El objetivo es implementar la capa de aplicación para Kalah .

1. Construya los métodos básicos del juego (**No olvide MDD y TDD)**

2. Ejecuten las pruebas y capturen el resultado.



**Ciclo 6: Jugar**

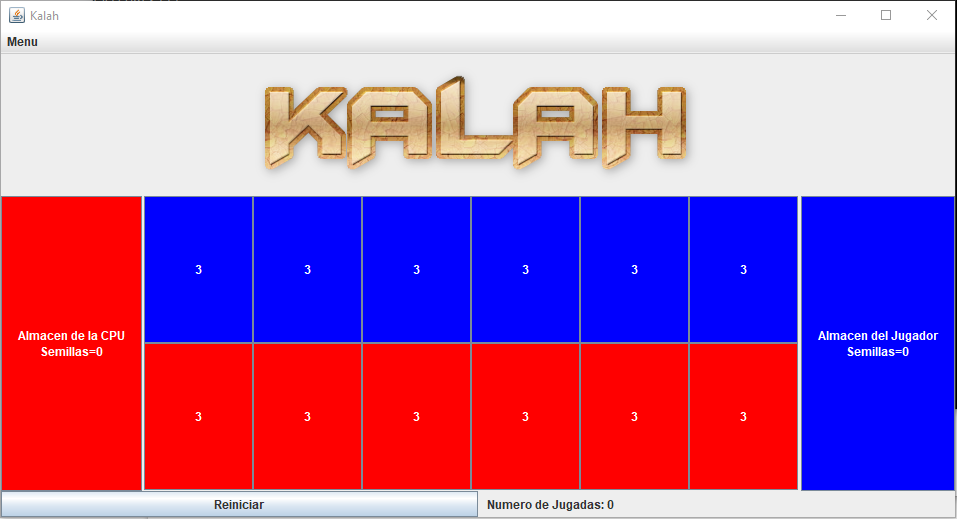
**[En \*.java y lab05.doc]**

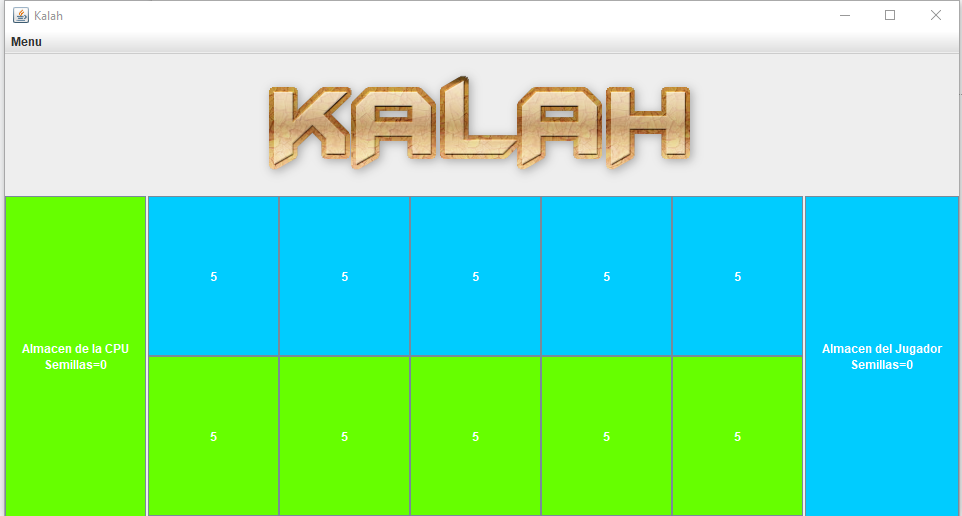
El objetivo es implementar el caso de uso jugar.

1. Adicione a la capa de presentación el atributo correspondiente al modelo.
2. Perfeccionen el método refresque() considerando la información del modelo de dominio.
3. Expliquen los elementos necesarios para implementar este caso de uso.

Se requiere colocar un Kalah en la capa de presentación y colocar un oyente en cada botón que se ve reflejado en el juego el cual ejecuta la forma respectiva de cada movimiento en cada casilla.

1. Implementen los componentes necesarios para jugar .
2. Ejecuten el caso de uso y capture las pantallas más significativas.





**Ciclo 7: Reiniciar**

**[En \*.java y lab05.doc]**

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 reiniciar

3. Ejecuten el caso de uso y capture las pantallas más significativas.



**Ciclo 8: Cambiar el tamaño del juego: casas y semillas [En \*.java y lab05.doc]**

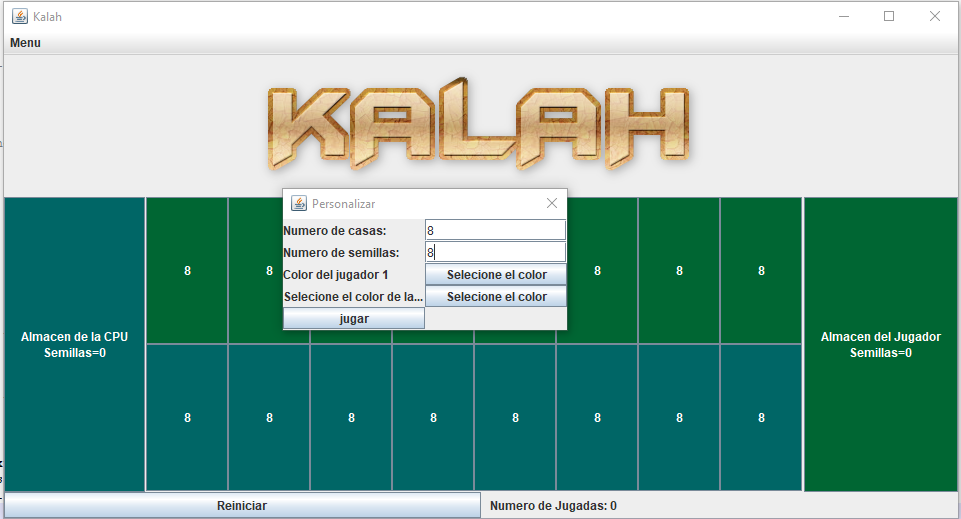
El objetivo es implementar este caso de uso.

1. Expliquen los elementos a usar para implementar este caso de uso

Se utilizó un atributo que tenga el número total de semillas, además de esto se genero uno que tiene el número de casas que se desean crear. Esto desde un JDialog que se genera también en el cambio de color.

2. Implementen los elementos necesarios para cambiar el tamaño 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/ Hombre)

35 horas

* ¿Cuál es el estado actual del laboratorio? ¿Por qué?

Completo pero con algunos bugs en la jugabilidad, debido al tiempo de entrega

* Considerando la práctica XP del laboratorio ¿por qué consideran que es importante? ¿Cuál considera fue su mayor logro? ¿Por qué? ¿Cuál consideran que fue su mayor problema? ¿Qué hicieron para resolverlo?

Programacion a par

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

Distribuimos bien el trabajo y supimos como unificar ideas del uno con el otro al momento de realizar el programa.