

# PROGRAMACIÓN ORIENTADA A OBJETOS

## INTERFAZ

### 2024-1

#### Laboratorio 5/6

#### OBJETIVOS

1. Desarrollar una mini aplicación gráfica considerando el patrón MVC.
2. Implementar el esquema de manejo de eventos con clases anónimas
3. Experimentar el comportamiento de las ventanas JFrame, JDialog y JOptionPane
4. Seleccionar los lienzos más apropiados para un diseño: JPanel, JScrollPane, JTabbedPane
5. Revisar las posibilidades de los estilos:FlowLayout, BorderLayout y GridLayout
6. Apropiar algunos componentes básicos: JLabel, JTextField, JButton, JMenuBar,
7. Apropiar algunos componentes especiales: JFileChooser y JColorChooser
8. Vivenciar las prácticas XP : [Acceptance tests](#) *are run often and the score is published*  
*When a bug is found tests are create*

#### 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 espacio 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 **Square**.

**El trabajo se debe hacer desde CONSOLA**

El propuesto por ustedes SquareGUI	El acordado en laboratorio Square
<b>Vista - Controlador</b>	<b>Modelo</b>

**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 **Square**.

1. Preparen un directorio llamado **Square** con los directorios src y bin y los subdirectorios para presentación, dominio y pruebas de unidad.

## Ciclo 0: Ventana vacía - Salir

[En \*.java y lab05.doc]

El objetivo es implementar la ventana principal de **Square** con un final adecuado desde el icono de cerrar. Utilizar el esquema de `prepareElements-prepareActions`.

1. Construyan el primer esquema de la ventana de **Square** únicamente con el título "**Square**". Para esto cree la clase `SquareGUI` como un `JFrame` con su creador (que sólo coloca el título) y el método `main` que crea un objeto `SquareGUI` 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)
2. Modifiquen el tamaño de la ventana para que ocupe un cuarto de la pantalla y ubíquela en el centro. Para eso inicien la codificación del método `prepareElements`. Capturen esa pantalla.
3. Traten de cerrar la ventana. ¿Termina la ejecución? ¿Qué deben hacer en consola para terminar la ejecución?
4. Estudien en `JFrame` el método `setDefaultCloseOperation`. ¿Para qué sirve? ¿Cómo lo usarían si queremos confirmar el cierre de la aplicación? ¿Cómo lo usarían si queremos simplemente cerrar la aplicación?
5. Preparen el "oyente" correspondiente al icono cerrar que le pida al usuario que confirme su selección. Para eso inicien la codificación del método `prepareActions` y el método asociado a la acción (`exit`). Ejecuten el programa y cierren el programa. Capturen las pantallas.

## 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 y Salir . Incluyan los separadores de opciones.

1. Expliquen los componentes visuales necesarios para este menú. ¿Cuáles serían atributos y cuáles podrían ser variables del método `prepareElements`? Justifique.
2. Construya la forma del menú propuesto (`prepareElements` - `prepareElementsMenu`) . Ejecuten. Capturen la pantalla.
3. Preparen el "oyente" correspondiente al icono cerrar con confirmación (`prepareActions` - `prepareActionsMenu`). Ejecuten el programa y salgan del programa. Capturen las pantallas.

## 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`.
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 opciones y capturen las pantallas más significativas.

### 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. Presenten el bosquejo del diseño de interfaz con todos los componentes necesarios.
2. Continúe con la implementación definiendo los atributos necesarios y extendiendo el método `prepareElements()`.

Para la zona del tablero defina un método `prepareElementsBoard()` y un método `refresh()` que actualiza la vista del tablero considerando, por ahora, el tablero inicial por omisión. Este método lo vamos a implementar realmente en otros ciclos.

3. 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 (vista - controlador) necesarios para implementar este caso de uso.
2. Detalle el comportamiento de `JColorChooser` especialmente el método estático `showDialog`
3. Implementen los componentes necesarios para cambiar el color de las fichas.
4. Ejecuten el caso de uso y capture las pantallas más significativas.

### Ciclo 5: Modelo Square

[En \*.java y lab05.doc]

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

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 `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. ¿Cuántos oyentes necesitan? ¿Por qué?
5. 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

[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 cambiar el tamaño del juego
3. Ejecuten el caso de uso y capture las pantallas más significativas.

## **RETROSPECTIVA**

1. ¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes?  
(Horas/Hombre)
2. ¿Cuál es el estado actual del laboratorio? ¿Por qué?
3. Considerando la práctica XP del laboratorio ¿por qué consideran que es importante?
4. ¿Cuál consideran fue su mayor logro? ¿Por qué? ¿Cuál consideran que fue su mayor problema? ¿Qué hicieron para resolverlo?
5. ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?