

# Documentación: Lights Out

Daniel Reales y Carlos Garcia<sup>1</sup>

<sup>1</sup>(201822265)& (202013993)

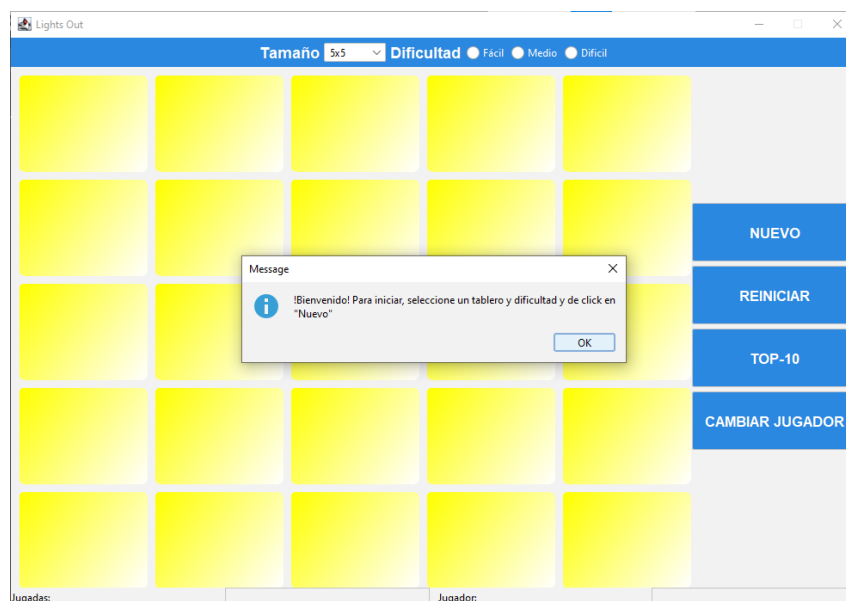
28 de octubre de 2021

A continuación se presenta la documentación de la interfaz gráfica diseñada e implementada para el juego Lights Out.

## Interfaz

### Inicialización

La interfaz gráfica construida comienza por presentar instrucciones al usuario sobre el uso del juego y la forma de inicializar el tablero. A continuación se muestra la implementación de esta parte del sistema:



*Figura 1. Presentación Inicial*

### Tablero

Una vez se inicializó el tablero, el jugador se enfrenta al juego y puede empezar a interactuar con el tablero haciendo click sobre cada una de las casillas. El sistema llevará una cuenta del número de jugadas en la parte inferior izquierda y hará una presentación del nombre del usuario en la parte inferior derecha.

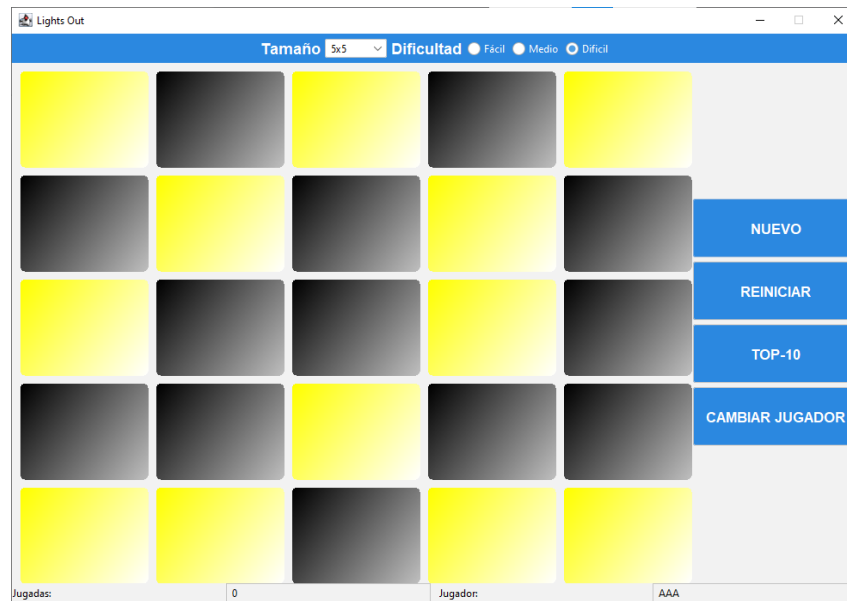


Figura 2. Tablero Inicializado

## Jugador

Cuando el usuario desea cambiar el nombre del jugador, este da click en el botón *Cambiar Jugador* y posteriormente se le muestra una interfaz como la presentada a continuación para ingresar y asignar el nuevo nombre:

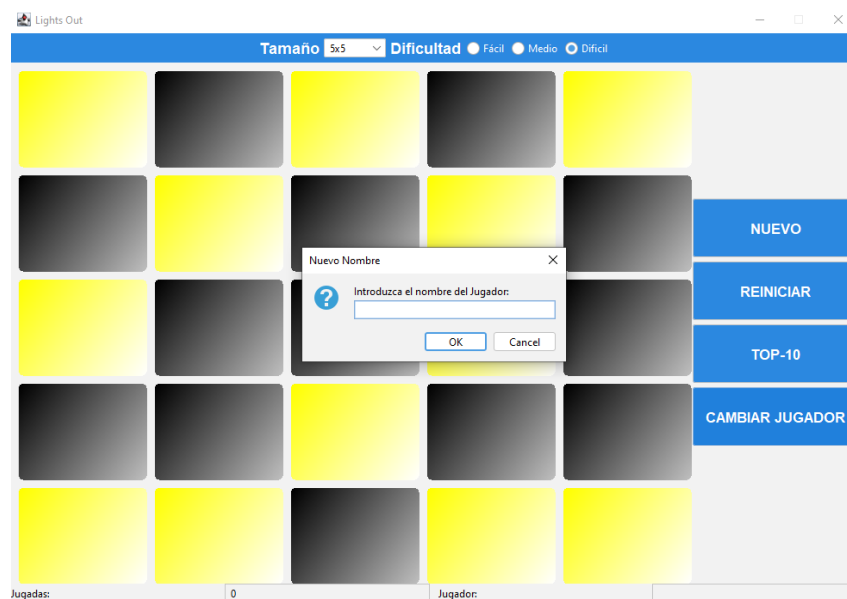


Figura 3. Cambio de Jugador

## Top 10

Finalmente, cuando el usuario da click sobre el botón *TOP-10*, se despliega una ventana de diálogo tal como la que se presenta a continuación:

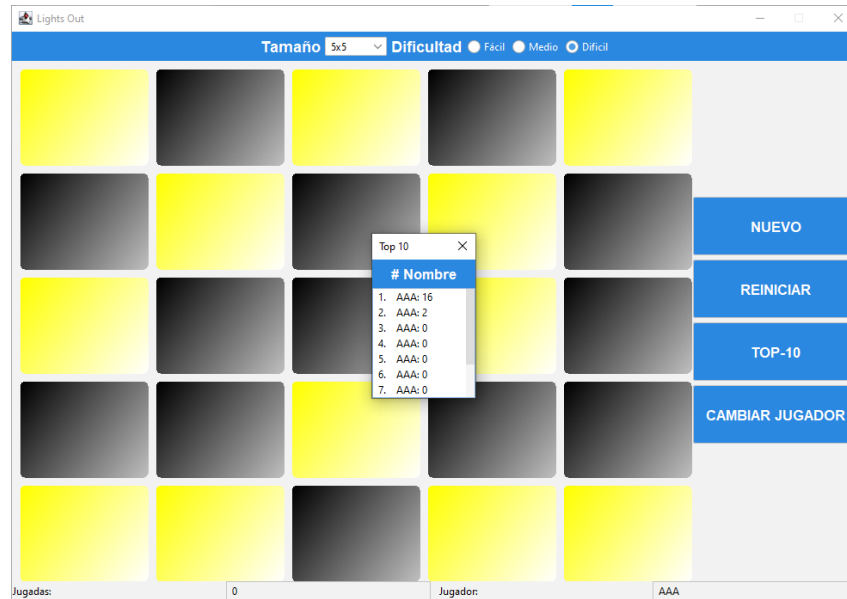


Figura 4. Top 10

## UML Alto Nivel

Para implementar la interfaz antes presentada se diseñó un diagrama UML que detalla las relaciones de alto nivel entre las clases que crean la estructura gráfica de la aplicación.<sup>1</sup>

El primer aspecto a resaltar es que se decidió realizar la creación de 2 paquetes adicionales al ya incluido: 1) *interfaz* y 2) *app*. El primero de estos contiene todas las clases que extienden elementos gráficos. Entre las incluidas encontramos 6:

1. *CoordLightsOut*
2. *InterfazJuego*
3. *PanelConfiguracion*
4. *PanelOpciones*
5. *PanelPartida*
6. *PanelTablero*

La primera de estas clases corresponde a lo que se denominó el coordinador del juego. Esta clase se encarga de conectar la interfaz con el modelo. Más aún, esta decisión de diseño se apoya en la observación de que en la medida de que se reduzca el número de clases del paquete de interfaz que conocen las clases del modelo, el mantenimiento del código será mejor debido a que se necesitarán menos cambios a futuro. La segunda clase corresponde a la clase central de la interfaz. Esta tiene como superclase a *JFrame* y corresponde a la ventana principal del juego. Por consiguiente está relacionada de forma bidireccional con todos los demás paneles. Finalmente, las clases *PanelConfiguracion*, *PanelOpciones*, *PanelPartida*, *PanelTablero* corresponden a cada uno de los paneles que se encuentran al interior del *JFrame* en un orden determinado por el *BorderLayout* utilizado.

Finalmente, existe un segundo paquete denominado *app* que contiene la clase aplicación que sirve como *entry point* del juego. Esto es, aquí se encuentra el método *main* que es llamado para iniciarlo.

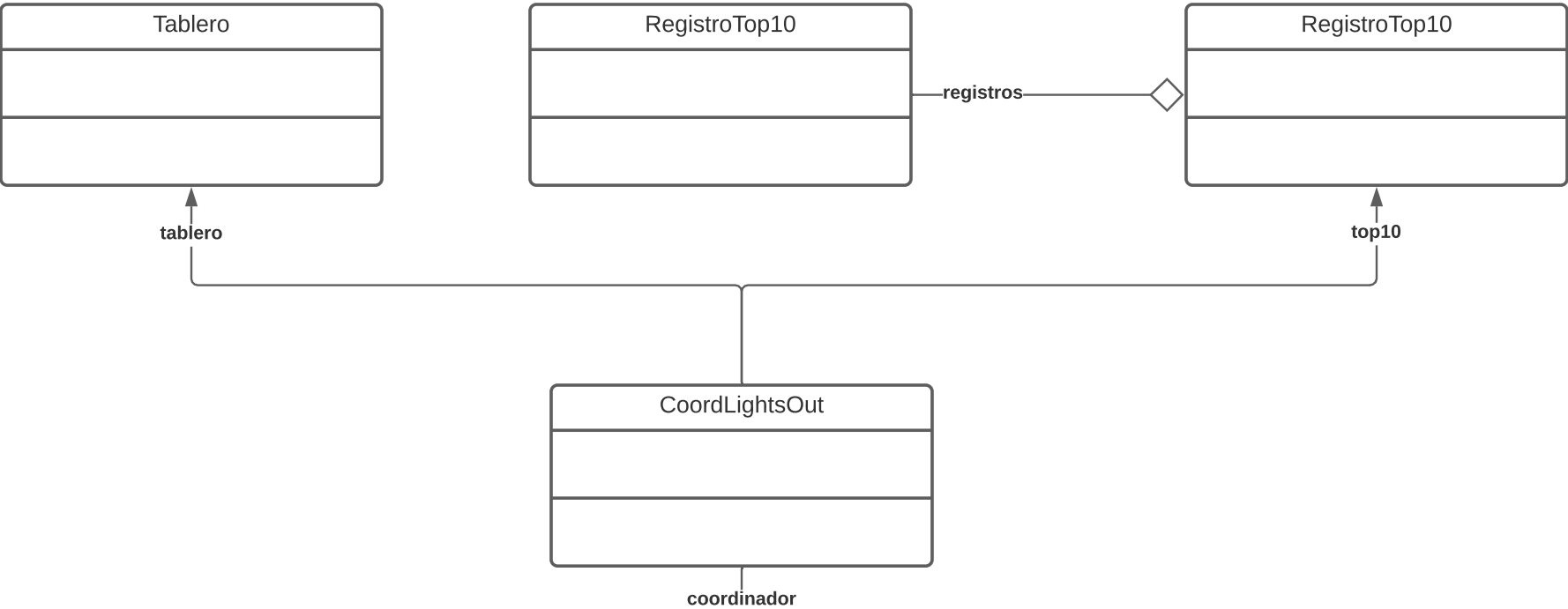
<sup>1</sup>El Diagrama UML de Alto Nivel se encuentra en el Anexo 1

## UML Detallado

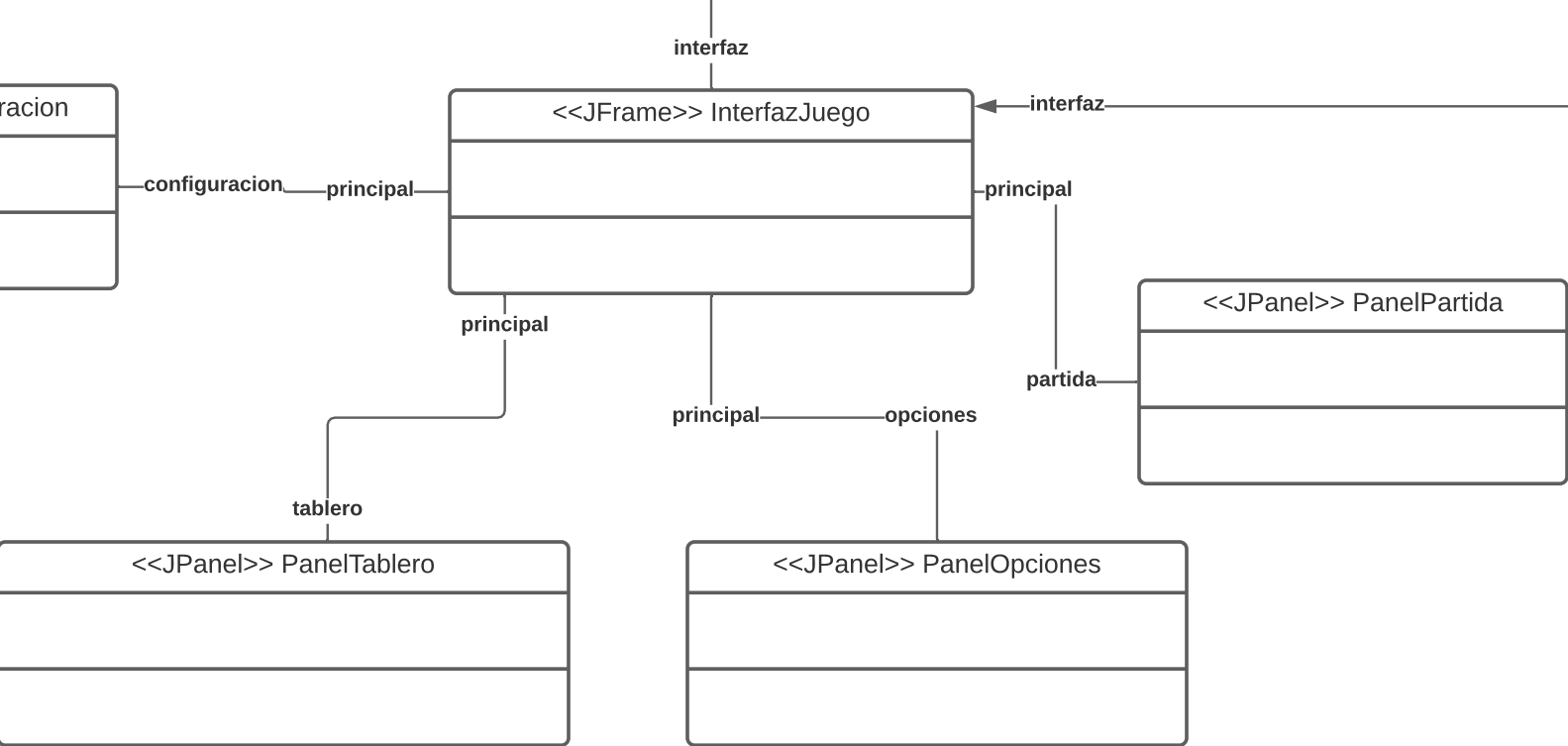
El UML detallado permite elucidar la estructura de manejo de responsabilidades entre las clases del paquete de la interfaz implementada. El rasgo principal de la estructura de responsabilidades es que cada panel se encarga de manejar los eventos de sus componentes (ej: clicks sobre botones o evento de cierre de ventana). En el manejo que cada panel da a los eventos realizados sobre sus componentes, se realiza la conexión con la ventana principal (clase *InterfazJuego*) en donde se atienden los distintos requerimientos funcionales o bien realizando operaciones dentro de los métodos definidos en esta o apoyandose en la clase *CoordLightsOut* que conecta con el modelo.

## Anexos

Modelo



Interfaz



App

