



TP INTEGRADOR

PROGRAMACION CONCURRENTES

QUEENS

RODRIGUEZ, EMANUEL
LEDESMA, MIGUEL
GALO, SANTIAGO
CASTELLANA, JULIAN
CATANIA, EZEQUIEL

02/07/2025

Índice

Repositorio de GitHub.....	2
Descripción.....	2
Introducción.....	2
Objetivos del Proyecto.....	2
Estrategia de hilos.....	3
Lenguaje y Frameworks Utilizados.....	3
Restricciones y Requisitos Técnicos.....	4
Manual de Usuario.....	4
1.Ejecución.....	4
2. Menú Principal.....	4
1. Start.....	5
2. Credits.....	5
3. Exit.....	5
3. En juego.....	6
1. Presentación.....	6
2. Movimientos.....	7
3. Victoria.....	9
Videos.....	9
Conclusiones.....	9

Repositorio de GitHub

<https://github.com/MiguelLedesma02/PCGame.git>

Descripción

Introducción

Este proyecto consiste en el desarrollo de un juego de lógica interactivo, inspirado en el problema de las reinas, con una variante basada en colores y restricciones adicionales. El juego fue desarrollado utilizando Godot Engine 4.4.1, empleando su lenguaje nativo GDScript y su sistema de nodos y escenas.

El objetivo del juego es que el jugador coloque coronas sobre un tablero de 8x8 celdas, cumpliendo las siguientes condiciones:

- Debe haber una sola corona por fila.
- Debe haber una sola corona por columna.
- Debe haber una sola corona por cada región de color.
- No puede haber coronas adyacentes entre sí (ni en diagonal, ni ortogonalmente).

Cada vez que el usuario interactúa con el tablero (coloca o quita una corona), el sistema valida automáticamente las condiciones anteriores y muestra un mensaje de victoria si se cumplen todas.

Objetivos del Proyecto

- Implementar un juego interactivo tipo puzzle utilizando Godot Engine.
- Aplicar programación concurrente con hilos para validar condiciones en paralelo.
- Practicar el diseño modular mediante escenas reutilizables (Cell.tscn).
- Leer y procesar archivos externos (Colors.csv) para personalizar regiones del tablero.
- Asegurar la corrección lógica del juego mediante validaciones automáticas tras cada clic.

- Fomentar buenas prácticas de desarrollo como uso de Mutex, separación de responsabilidades y eficiencia en el manejo de eventos.

Estrategia de hilos

Para validar el tablero de forma eficiente, se implementó una estrategia con múltiples hilos que permite verificar simultáneamente distintas reglas del juego. Esto mejora notablemente el rendimiento, especialmente cuando el tablero está más completo, y permite ofrecer una respuesta inmediata al usuario tras cada cambio.

La validación del tablero debe cumplir cuatro condiciones: que haya una sola corona por fila, una por columna, una por color, y que no existan coronas adyacentes en diagonal. Para controlar cada una de estas reglas en paralelo, se crean 8 hilos por tipo de validación (uno por fila, uno por columna, uno por color y uno por fila para las diagonales), dando un total de 32 hilos activos cada vez que se modifica el tablero.

Cada hilo ejecuta una función que analiza una parte del tablero. Estas funciones agregan a un diccionario si esa sección es válida y, en caso de detectar conflictos, incluye las posiciones de las coronas que no cumplen la regla. Con estos resultados, el sistema puede marcar en rojo las coronas conflictivas, lo que le da al jugador una visualización clara de los errores.

Para evitar problemas de condición de carrera entre los hilos, se implementó una solución con un mutex para sincronizar cuando un hilo agrega al diccionario global si esa sección es válida o inválida, y luego se utiliza un mutex para cada caso: filas, columnas, adyacentes y colores, donde cada uno nos asegura la mutua exclusión en caso de escribir sobre los contadores de coronas correctas.

Una vez que todos los hilos terminan su ejecución, se recopilan las posiciones inválidas y se marcan las coronas correspondientes. Si no se detecta ningún conflicto, y la cantidad de coronas coincide con la cantidad de colores en el tablero, se considera que el jugador resolvió correctamente el tablero y se muestra una ventana de felicitación, finalizando la partida.

Lenguaje y Frameworks Utilizados

- Motor: Godot Engine 4.4.1
- Lenguaje: GDScript (lenguaje nativo de Godot, similar a Python)
- Framework visual: Sistema de nodos de Godot
- UI: Escenas (.tscn), controles (Control, GridContainer, Button, etc.)

Restricciones y Requisitos Técnicos

- Sistema operativo compatible: Windows 10 o superior, Linux o macOS
- Versión de Godot recomendada: Godot 4.4.1 (no requiere .NET)
- Resolución mínima recomendada: 1240x720
- No requiere navegador ni acceso a internet.
- No tiene requisitos de hardware especiales (funciona correctamente en equipos de gama media o baja)

Manual de Usuario

1.Ejecución

Primero, ejecutamos el archivo Queens.exe

2. Menú Principal



1. Start

Si queremos jugar, presionaremos en el botón Start y nos dirigiremos a la pantalla de niveles



Luego, podremos seleccionar un nivel, y el juego comenzará. O podemos volver atrás al menú principal, presionando el botón con el dibujo de la flecha.

2. Credits

Con este botón, podremos ver un agradecimiento al usuario por probar el juego, y un breve contexto de la elaboración del proyecto. Podremos pulsar en el botón con la flecha para volver al menú principal.



3. Exit

Si simplemente deseamos salir del juego, presionamos Exit y se cerrará la ventana.

3. En juego

1. Presentación

Una vez seleccionado el nivel, estaremos listos para comenzar a jugar. Se presenta un tablero 8x8, dividido en zonas de colores, y nuestro objetivo será poner una corona por cada color, sin que estén en la misma columna, fila, o sean adyacentes.

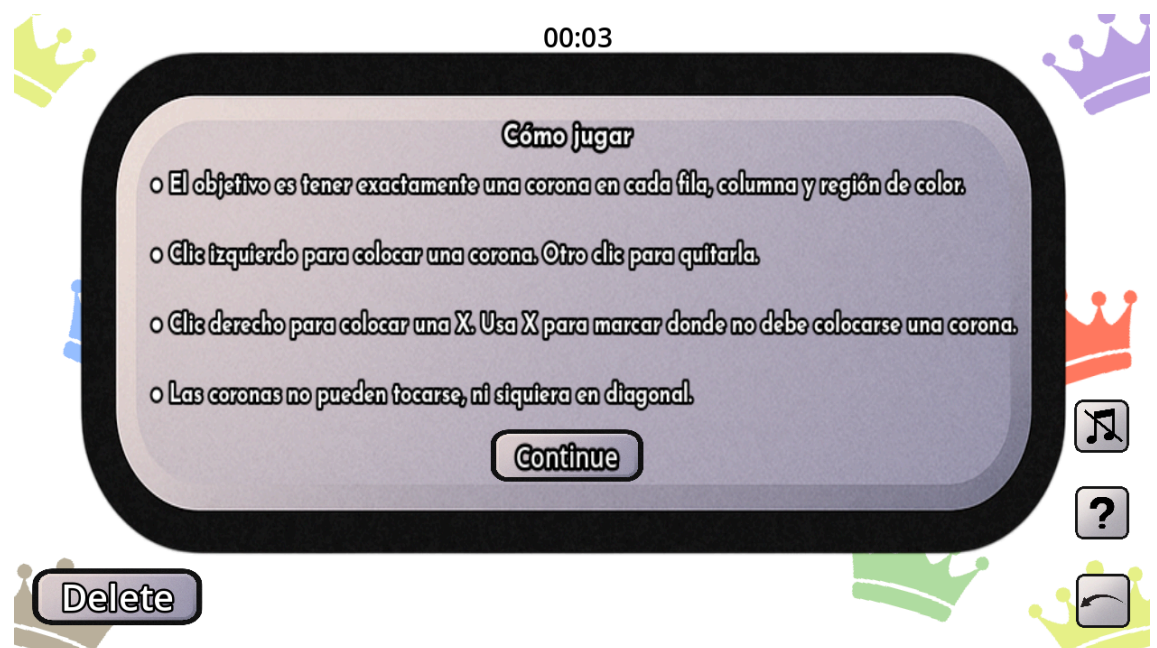
Oprimiendo el botón Delete podemos borrar todas las coronas o cruces que hayamos colocado en el tablero.

Con clic izquierdo colocaremos una corona en la celda seleccionada, y con clic derecho podremos colocar una cruz, las cuales sirven de modo de guía para el usuario, pudiendo marcar las posiciones en donde no se puedan colocar las coronas.

Las cruces sirven a modo de ayuda visual al usuario, no afectan en la determinación de movimientos válidos o inválidos.

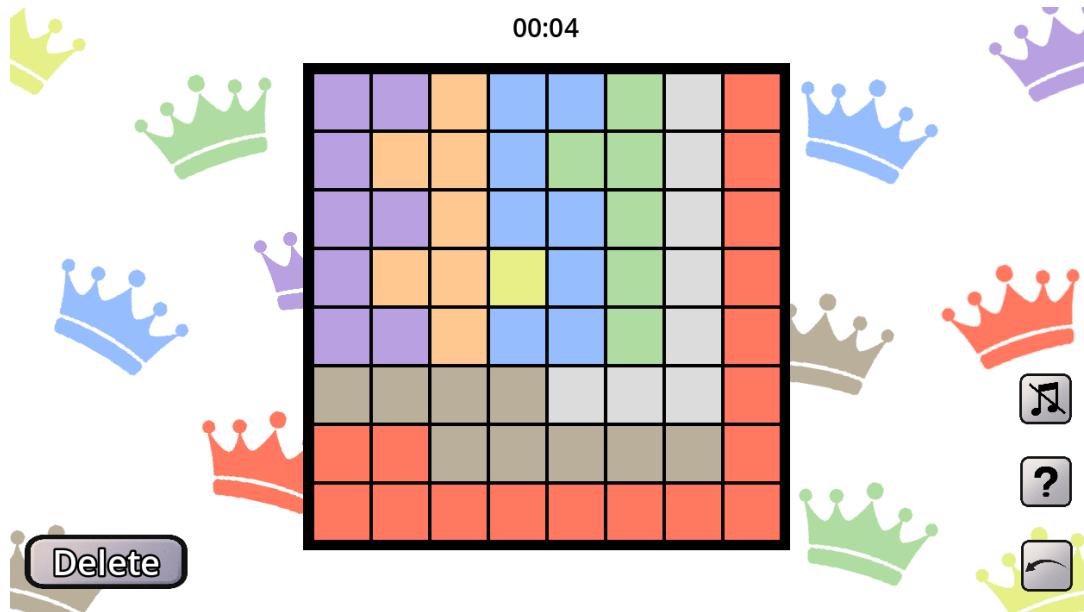
También tendremos un cronómetro en la parte superior de la ventana, para saber el tiempo que estamos tardando en resolver el nivel.

Para información sobre cómo jugar, podremos oprimir el botón “?” donde nos saldrá un instructivo.



También contaremos con una música de fondo, que podremos silenciar o reactivar oprimiendo el botón con el símbolo musical.

NOTA: en cualquier momento que deseemos podremos volver al menú de niveles, pulsando en el botón con la flecha.



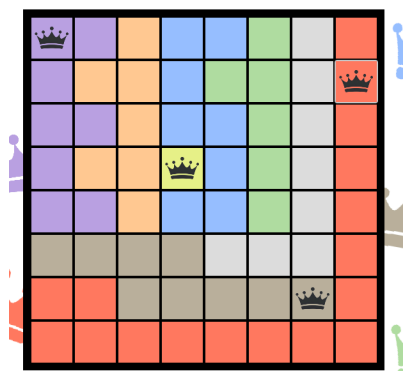
2. Movimientos

El juego nos pintará las coronas de color rojo cuando hacemos algun movimiento invalido, ya sea que hay dos o mas en la misma columna, fila, color o en adyacencia.

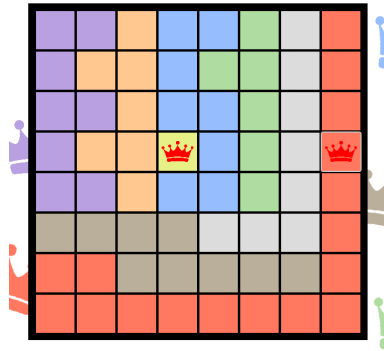
Se podrán eliminar todas pulsando en Delete (recordar que borra TODAS las coronas puestas), o simplemente pulsando con clic izquierdo en la celda que deseemos sacar la corona.

Cuando deje de haber movimientos inválidos, las coronas volveran a estar de color negro.

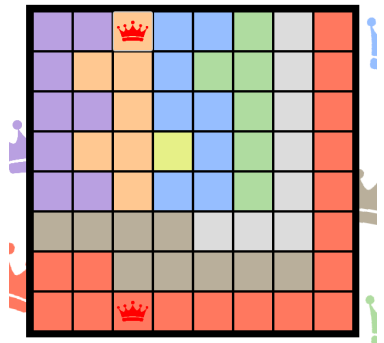
Normal



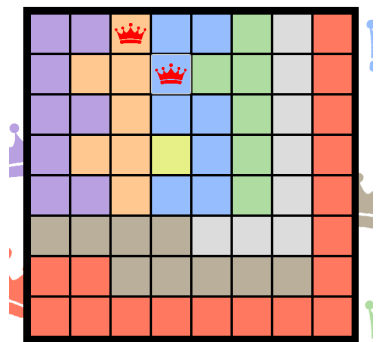
Misma Fila



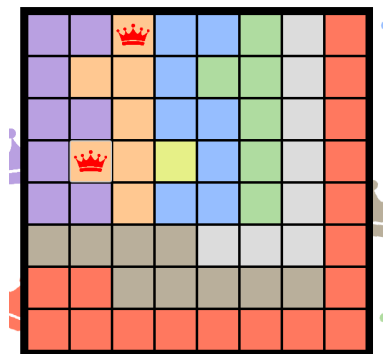
Misma Columna



Adyacencia



Mismo Color



3. Victoria

Una vez coloquemos todas las coronas en su posición correcta, sin que haya ninguna invalida, nos aparecerá un cartel, felicitándonos por haber completado el nivel, y a su vez, podremos ver en la parte superior de la ventana el tiempo que tardamos.



Además, el juego tiene una planilla donde guarda los ganadores, con la fecha y hora de la victoria, y el número de intentos que realizó, a modo de tener un ranking.

Planilla:

<https://docs.google.com/spreadsheets/d/18HPa29B9eeYCz5zGhjiVy6WRO5TDEgvBA6cUMf6K5Y/edit?gid=0#gid=0>

Videos

A continuación, adjuntamos unos vídeos donde muestra el avance que fue teniendo el juego, donde con el tiempo fuimos puliendo el código y agregando más funcionalidades para mejorar la experiencia del usuario.

<https://drive.google.com/file/d/1mqubWrEYx2ovoIEhF51oomqhNbZmnpRY/view?usp=sharing>

<https://drive.google.com/file/d/1Gzxfnk5cvPVXxjT0kdVVt9tYDpxVTEN2/view?usp=sharing>

Conclusiones

Al realizar este trabajo práctico integrador, nos encontramos con varios desafíos. Uno de los principales fue pensar cómo implementar correctamente los hilos, ya que

teníamos pensado en dónde usarlos, pero había varias alternativas de implementación.

También fue un desafío aprender a usar Godot, ya que era la primera vez que todos lo usábamos. Tuvimos que entender cómo funciona su sistema de nodos, cómo se organiza el proyecto y cómo escribir scripts con GDScript. Al principio nos costó, pero a medida que avanzamos fuimos entendiendo mejor y logramos aplicar lo aprendido para la creación del juego.

En general, el proyecto nos ayudó a aprender cosas nuevas y a aplicar conocimientos adquiridos en la materia en un entorno distinto al que estamos acostumbrados, como el que es la programación de videojuegos.