****

**Grupo no. 3**

Sustentantes:

**Deybby Rosario 2024-0504**

**Sarah Peña 2024-0506**

**Carlos Roa 2024-0510**

**Héctor Reyes 2024-0526**

Materia:

**Juegos Inteligentes**

Docente:

**Carlos Bienvenido Ogando Montas**

Sección:

**2025-C-3-1908-3242-TIA-603**

Fecha:

**08-10-2025**

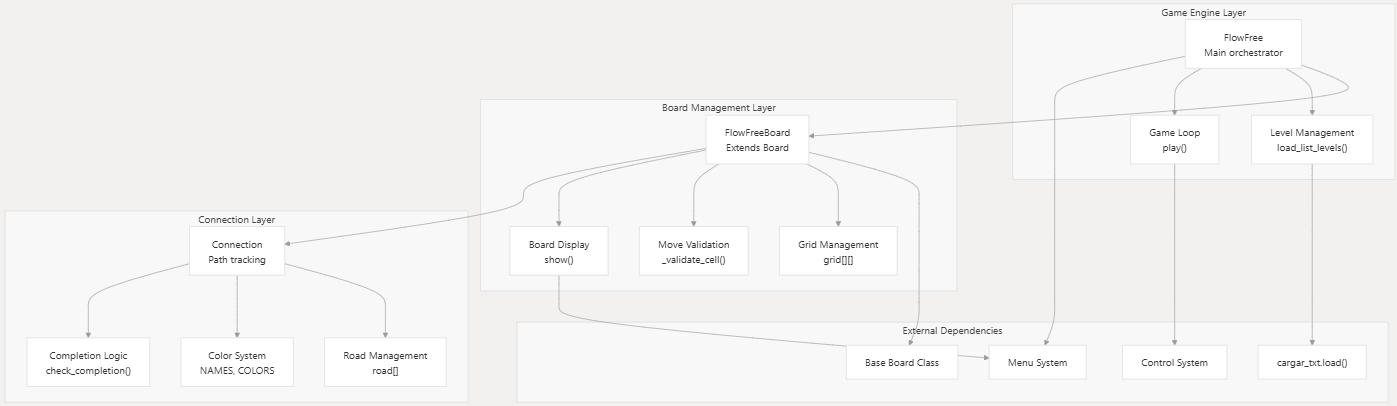
Tema:

**FlowFree - Búsqueda desinformada e informada**

**Game Engine**

El motor sigue una **arquitectura en capas**, con separación clara entre:

* **Orquestación del juego** (gestión de menús, niveles y flujo de juego).
* **Gestión del tablero** (estado del tablero, validación de movimientos, visualización).
* **Control de conexiones** (creación, extensión y ruptura de caminos de color).



|  |  |  |  |
| --- | --- | --- | --- |
| Componente | Responsabilidad principal | Métodos clave | Líneas |
| FlowFree | Orquestación del juego, gestión de niveles, bucle de juego | app(), play(), select\_level() | 212-428 |
| FlowFreeBoard | Gestión del estado del tablero, visualización, validación de movimientos | show(), \_validate\_cell(), percentage\_filled() | 89-211 |
| Connection | Seguimiento y validación de caminos individuales | add\_to\_road(), check\_completion(), break\_road() | 10-85 |

**Gestión del Estado del Juego**

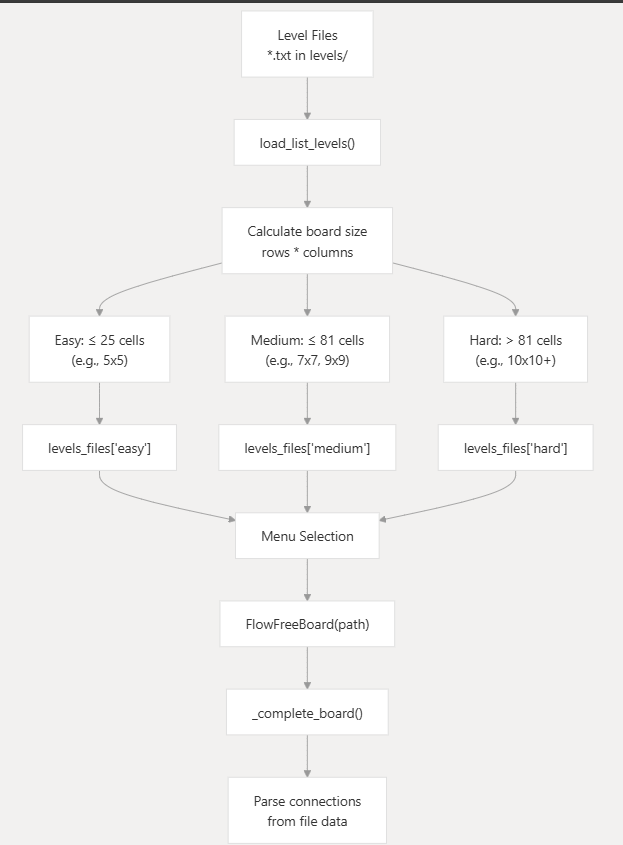
El motor mantiene varias variables clave para coordinar el flujo del juego:

|  |  |
| --- | --- |
| Variable | Función |
| self.board | Instancia activa de FlowFreeBoard. |
| self.last\_color\_position | Rastrea la conexión que se está dibujando actualmente. |
| self.last\_move | Permite validar movimientos y realizar deshacer. |
| self.levels\_files y self.levels\_files\_names | Gestionan los niveles disponibles y sus nombres para la UI. |

**Secuencia del Flujo de Juego**

El flujo típico de ejecución del juego va desde el inicio hasta la finalización de un nivel:

1. **Carga del menú principal** (app()).
2. **Selección de nivel** y carga de datos.
3. **Inicialización del tablero** (FlowFreeBoard).
4. **Bucle de juego** gestionado por play().
5. **Validación de movimientos y actualización del estado** según acciones del jugador.
6. **Finalización de nivel** y retorno al menú principal.



**Carga de Niveles e Inicialización del Tablero**

El motor incluye un sistema de gestión de niveles:

1. **Descubrimiento de archivos**: Escanea el directorio levels/ buscando archivos .txt.
2. **Extracción de metadatos**: Obtiene dimensiones del tablero y número de colores desde el nombre del archivo.
3. **Clasificación por dificultad**: Según la cantidad total de celdas del tablero.
4. **Generación de menú**: Crea nombres de niveles formateados para la interfaz.
5. **Inicialización del tablero**: Carga el nivel seleccionado en una instancia de FlowFreeBoard.

**Validación de Movimientos y Lógica de Juego**

El motor implementa un sistema de validación que maneja distintos escenarios:

|  |  |  |  |
| --- | --- | --- | --- |
| Escenario | Condición | Acción | Referencia de código |
| Celda inválida | !\_validate\_cell(x, y) | Ignorar movimiento | 345-346 |
| Mismo lugar | move == last\_move | Deseleccionar / deshacer | 349-358 |
| Nuevo inicio | !last\_color\_position | Inicia un nuevo camino | 369-383 |
| Conexión completada | Conecta con círculo del mismo color | Completa el camino | 387-399 |
| Intersección de camino | Movimiento hacia celda existente | Rompe el camino en ese punto | 405-409 |
| Extensión de camino | Celda adyacente válida | Extiende el camino actual | 418-421 |