



ANÁLISIS DINÁMICO– BAD DOPO CREAM

POOB



ANDRES FELIPE PINEDA GAITAN
HEIDY ALEJANDRA ORJUELA RAMIREZ

1. Resultado inicial del análisis dinámico

El análisis dinámico se realizó observando el comportamiento del sistema durante su ejecución, evaluando el flujo de eventos, interacción entre objetos, respuesta a entradas del usuario y actualización del estado del juego.

En el estado inicial del sistema se identificaron los siguientes comportamientos:

- El ControladorJuego gestionaba simultáneamente:
 - ✓ El tiempo del juego
 - ✓ La IA
 - ✓ Las colisiones
 - ✓ La actualización del HUD
- La comunicación entre teclado, controlador y dominio funcionaba, pero con dependencias directas.
- La actualización del estado del juego y la renderización estaban muy ligadas temporalmente.
- La lógica de colisiones y reglas del juego estaba centralizada, dificultando su modificación o reutilización.
- No existía un flujo claramente definido para pausar, reiniciar o finalizar el juego desde distintos estados.

A pesar de que el juego era funcional, el análisis dinámico inicial mostró que la complejidad del flujo de ejecución podía generar errores difíciles de rastrear y dificultar la incorporación de nuevas funcionalidades como guardado de partidas o nuevos modos de juego.

2. Decisiones tomadas tras el análisis dinámico

Con base en el análisis dinámico inicial, se tomaron las siguientes decisiones:

1. Clarificar el rol del controlador del juego, limitándolo a coordinar eventos y delegar la lógica específica al dominio (Juego, Mapa, Jugador).
2. Separar claramente el ciclo del juego, distinguiendo:
 - ✓ Actualización de estado (movimientos, IA, colisiones)
 - ✓ Renderización (dibujado del mapa y entidades)
 - ✓ Interacción con el usuario (teclado y menús)
3. Delegar comportamientos específicos a las entidades, por ejemplo:
 - ✓ Las frutas manejan su propia recolección.

- ✓ Los enemigos reaccionan según su tipo.
 - ✓ Las baldosas calientes aplican efectos al helado.
4. Preparar el flujo de ejecución para nuevos estados, como:
- ✓ Pausa
 - ✓ Fin de nivel
 - ✓ Victoria o derrota
 - ✓ Guardado y carga de partidas
5. Reducir dependencias directas en tiempo de ejecución, permitiendo que la presentación consulte el estado del dominio sin modificarlo directamente.

3. Resultado final del análisis dinámico

Después de aplicar las decisiones tomadas, el análisis dinámico final evidenció:

- Un flujo de ejecución más ordenado y predecible.
- Menor complejidad en el controlador.
- Mejor sincronización entre lógica del juego y renderizado.
- Mayor facilidad para incorporar nuevas funcionalidades dinámicas (nuevos enemigos, nuevas IA, distintos niveles).
- Comportamientos encapsulados en las clases correspondientes, mejorando la cohesión.

En conclusión, el sistema pasó de un funcionamiento correcto pero rígido a un comportamiento dinámico más flexible, extensible y alineado con buenas prácticas.