

## **TALLER 3**

### **Diseño y Programación orientada a objetos**

Ernesto Pérez – 202112530

Germán Moreno – 202116701

Sofia Velasquez – 202113334

**NOTA: Los diagramas UML y de secuencia están al final del documento.**

#### **Iteración #1:**

- El sistema debe ser capaz de:
  1. Matar a pac-man (responsabilidad de los fantasmas)
  2. Comer galleta (responsabilidad de pac-man)
  3. Cazar fantasmas (responsabilidad temporal de pac-man)
  4. Conseguir puntos (responsabilidad de pac-man)
  5. Ir a celda (responsabilidad temporal de fantasmas)
  6. Cargar mapa (responsabilidad user)
  7. Insertar usuario (responsabilidad del usuario)
  8. Guardar top10 (responsabilidad del sistema)
  9. Insertar personalidad (responsabilidad del sistema)
- Información que fluye:
  1. Pac-man no aplica para elementos conectados
  2. Posición de pac-man y fantasmas
  3. Ubicación de puntos y galletas
  4. Estado de pac-man (vidas, invencible)
  5. Estados de fantasmas (vivos, cazados)
  6. Puntuación y top 10
- Actividad y toma de decisiones
  1. De los movimientos de pac-man se encarga el usuario mediante controladorPacman
  2. De afectar de pac-man (invencible, menos vidas) se encarga el sistema mediante la clase estadoPacman
  3. De los fantasmas se encarga el sistema incluyendo personalidad mediante la clase Fantasma.
  4. De la carga de mapa se encarga la clase Tablero
  5. De mostrar la interface se encarga la clase Interface

#### **Requerimientos Funcionales:**

- Jugar
- Cargar tablero
- Consultar mejores 10 puntajes

#### **Roles:**

1. Aplicación

#### **Objetos:**

1. En movimiento (fantasmas y pacman)
2. Estático (galletas y frutas)
3. Tablero

**Responsabilidades:**

- Iniciar Juego
- Cargar tablero
- Mostrar mejores 10 puntajes
- Asignar personalidad a los fantasmas

**Iteración #2:****Requerimientos Funcionales:**

- Jugar
- Cargar tablero
- Consultar mejores 10 puntajes
- Mover el pac-man
- Asignar personalidad a los fantasmas

**Roles:**

1. Pac-man
2. Juego
3. Fantasma
4. Puntuaciones

**Objetos:**

1. Pacman <<>>
2. Fantasmas <<>>
3. Frutas <<>>
4. Galletas <<>>
5. Tablero <<>>

**Responsabilidades:**

- Mover pac-man **(1)**
- Comer galleta **(1)**
- Comer fruta (hacerse invencible) **(1)**
- Estado pac-man (invencible o no) **(1)**
- Estado Fantasma **(3)**
- Mostrar puntuación **(4)**
- Repartir por el tablero los elementos (galletas, frutas, fantasmas) **(2)**

**Iteración #3:****Requerimientos Funcionales:**

- Jugar
- Cargar tablero
- Consultar mejores 10 puntajes
- Mover el pac-man
- Asignar personalidad a los fantasmas

**Roles:**

1. Pac-man
2. Nivel
3. Fantasma
4. Puntuaciones
5. Menú

**Objetos:**

1. Pacman << Information holder >>
2. ControladorPacman <<Controller>>
3. Fantasmas << Information holder >>
4. Frutas << Information holder >>
5. Galletas << Information holder >>
6. ControladorElementos <<Controller>>
7. Tableros <<Information holder>>

**Responsabilidades:**

- Mover pac-man **(1)**
- Comer galleta **(1)**
- Comer fruta (hacerse invencible) **(1)**
- Estado pac-man (invencible o no) **(1)**
- Cazar fantasma (solo si es invencible) **(1)**
- Estado Fantasma (cazado o no) **(3)**
- Mostrar puntuación actual **(4)**
- Repartir por el tablero los elementos (galletas, frutas, fantasmas) **(2)**
- Mostrar mejores puntuaciones **(5)**
- Iniciar nivel **(5)**
- Cargar tablero **(5)**
- Comportamiento fantasma **(3)**
- Cazar pac-man **(3)**
- Huir de pac-man (cuando es invencible) **(3)**

**Iteración #4:****Requerimientos Funcionales:**

- Jugar
- Cargar tablero
- Consultar mejores 10 puntajes
- Mover el pac-man
- Asignar personalidad a los fantasmas

**Roles:**

1. Pac-man
2. Nivel
3. Fantasma
4. Puntuaciones
5. Juego
6. Casilla

**Objetos:**

1. Pacman << Information holder >>
2. ControladorPacman <<Controller>>
3. Fantasmas << Information holder >>
4. ControladorFantasma <<Controller>>
5. Galletas << Information holder >>
6. Frutas << Information holder >>
7. ControladorElementos <<Controller>>
8. Tablero <<Information holder>>
9. Puntuación <<Information holder>>

#### Objetos y roles:

1. Los objetos Pacman y Controlador Pacman hacen parte del Rol de Pacman
2. Los objetos Fantasmas y ControladorFantasma hacen parte del rol Fantasma
3. Los objetos Galletas, Frutas y ControladorElementos hacen parte del rol Casilla
4. El objeto Tableros hace parte del rol juego.
5. El objeto Puntuación hace parte del rol puntuaciones

#### Responsabilidades:

Responsabilidad	Rol
Mover pac-man	Pac-man
Comer galleta	
Comer fruta (hacerse invencible)	
Estado pac-man (invencible o no)	
Cazar fantasma (solo si es invencible)	
Ir a base (si fue cazado)	
Quitar vida	
Mostrar puntuación actual del nivel	Puntuaciones
Sumar puntaje del nivel	
Eliminar elemento (galletas, frutas, fantasma)	Casilla
Quitar vida	Nivel
Terminar nivel	
Repartir por el tablero los elementos (galletas, frutas, fantasmas)	
Actualizar puntaje	
Iniciar nivel	
Mover fantasma dependiendo de la personalidad	Fantasma
Estado Fantasma (cazado o no)	
Cazar Pacman	
Huir de Pacman	
Ir a base (si fue cazado)	
Cambiar mapa	Juego
Cargar tablero	
Mostrar mejores puntuaciones	

Pedir nickname	
----------------	--

- Las responsabilidades de 'Cazar Pacman' y 'Huir de Pacman' dependen del estado de pacman y solo puede haber 1 activa.

### Colaboraciones:

1. El comportamiento de los fantasmas depende del estado de pacman.
  2. Quitar vida necesita las ubicaciones de los fantasmas y de pacman para saber si chocan o no y así saber si debe o no quitar vidas.
  3. Comer fruta/galleta colabora con casilla para que estos elementos desaparezcan del mapa y también con Nivel para actualizar la puntuación.
  4. Ir a base colabora con casilla para saber cuál es la posición de inicio del fantasma / pacman
  5. Para Terminar juego hay una colaboración entre el Nivel y Pacman, pues se necesita saber cuántas vidas tiene pacman
- En el dominio del problema nos encontramos con un juego simple que contiene varios elementos como lo son: pac-man, los fantasmas, las galletas, los puntos y las casillas.

### Reflexión

#### Ventajas y desventajas (tradeoffs)

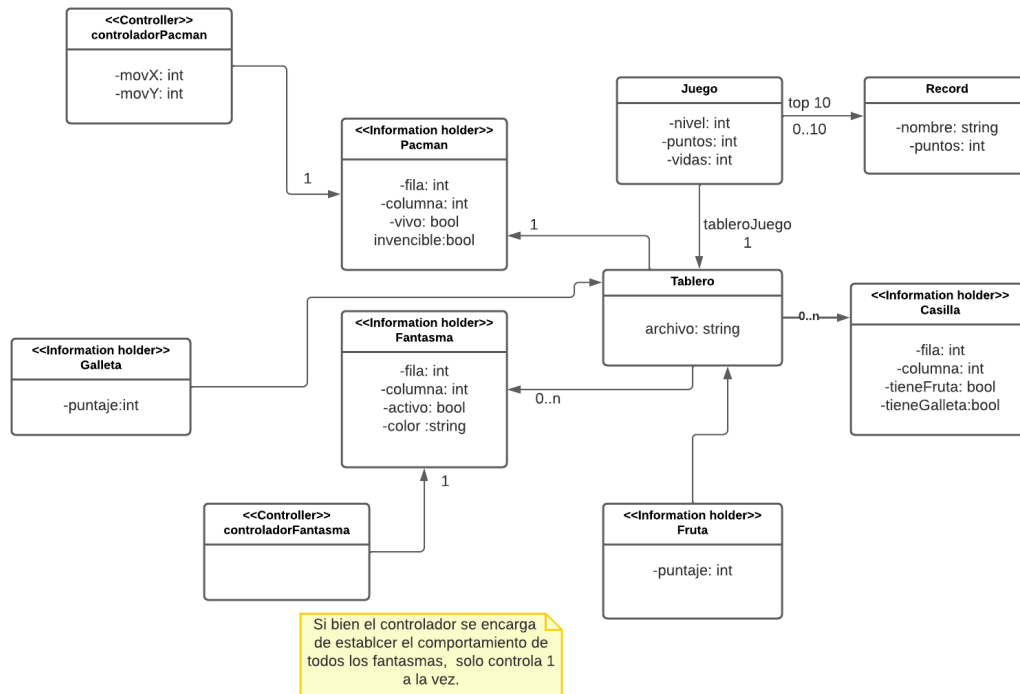
El diseño iterativo es muy bueno en el sentido en que es muy sistemático y permite elaborar un proyecto grande con un alto nivel de detalle.

Una desventaja es el tiempo que consume esta iteración tan minuciosa de todos los elementos pues nos dimos cuenta de que podíamos hacer la primera iteración con un alto nivel de detalle y en las siguientes iteraciones más que desglosar nuestros elementos en elementos más pequeños nos hubiéramos concentrado en 'pulir' responsabilidades, roles, colaboraciones y demás, haciendo del proceso mucho más rápido sin mayores complicaciones.

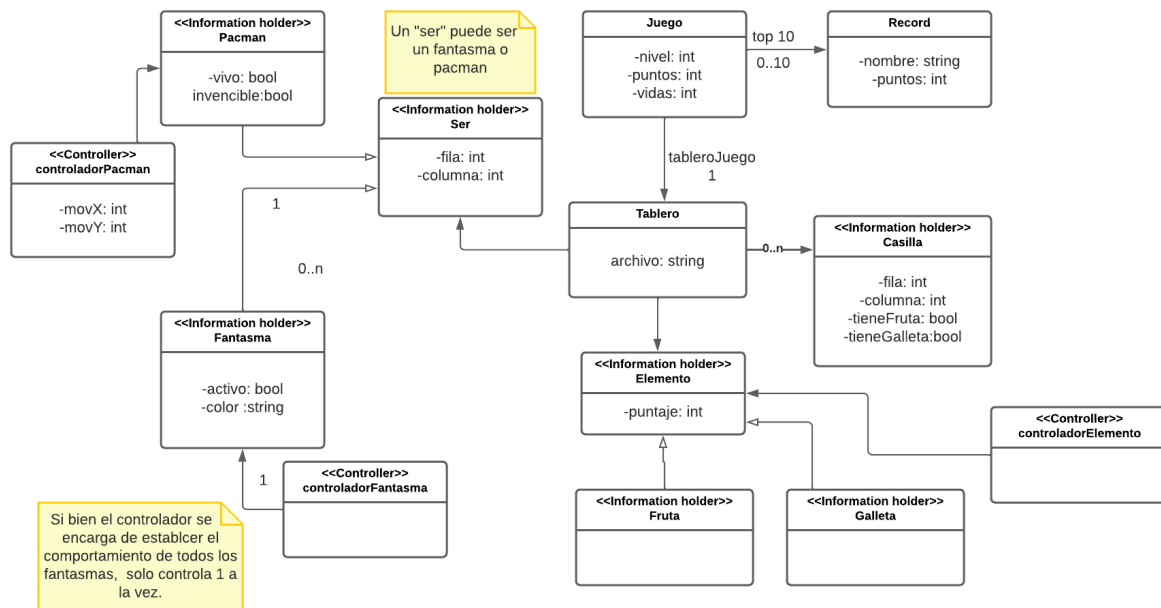
#### Elementos problemáticos.

La forma en que se manejan elementos más del juego, el nivel y no tanto con 'la jugabilidad' del juego. Como el puntaje, la carga de mapa, el pedir Nick, es difícil saber si deberían ir bajo un mismo rol responsable o distribuir. Se discutió también la manera en que se controlan los fantasmas ya que, al tener distintas personalidades, sus movimientos no se pueden generalizar, se concluyó que es tarea del controlador Fantasma determinar su comportamiento.

## UML iteración 1



## UML final



## Diagrama de secuencia

