**Taller 3 - Diseño**

**1ra Iteración:**

Elementos relacionados con el dominio:

* Pac-man (controller).
* Fantasma (controller).
* Casilla (information holder).
* Tablero (structurer).
* Récord (Information holder)
* Movedor

Agrupadores:

* Juego (structurer).

Elementos relacionados con el sistema:

* Cargador de mapa (service provider).??
* Agregador de fantasmas (

Responsabilidades:

* Pedir el nombre del archivo del tablero.
* Cargar la información del tablero
* Alguien debe reconocer los movimientos de todos los elementos.
* Actualizar el tablero.
* Alguien debe poder agregar un nuevo fantasma al tablero.
* Alguien debe preguntar si seguir con el mismo tablero o cambiarlo.
* Alguien debe contar cuántas vidas quedan.
* Mostrar top 10
* Comparar top 10 para evaluar si agregar uno nuevo
* Añadir y quitar alguien al top 10
* Mover a los actores del juego (pacman, fantasmas)
* Contar los puntos del jugador
* Alamacenar top 10
* Descontar una vida en caso de que el fantasma alcance a Pac-man. Y sacarlo del juego en caso de que ya haya perdido 3 vidas.
* Activar el poder de volverse invencible si se come una fruta.
* Si pacman toca un fantasma teniendo poderes, se debe enviar al fantasma a la casilla donde inicio el juego.

**2da Iteración:**

Elementos relacionados con el dominio:

* Pac-man (information holder).
* Fantasma (information holder).
* Casilla (information holder).
* Tablero (structurer).
* Puntuaciones (Information holder)
* Actualizador (controller)

Agrupadores:

* Juego (structurer).

Elementos relacionados con el sistema:

* Cargador de mapa (service provider).
* Agregador de fantasmas (service provider).

Responsabilidades:

* Pedir el nombre del archivo del tablero. *Cargador de mapa.*
* Cargar la información del tablero. *Cargador de mapa.*
* Alguien debe reconocer los movimientos de todos los elementos. *Tablero.*
* Actualizar el tablero. *Actualizador.*
* Alguien debe poder agregar un nuevo fantasma al tablero. *Agregador de fantasmas.*
* Alguien debe contar cuántas vidas quedan. *Juego.*
* Almacenar historial de puntuaciones. *Puntuaciones.*
* Mostrar top 10. *Puntuaciones.*
* Comparar top 10 para evaluar si agregar uno nuevo. *Puntuaciones.*
* Añadir y quitar alguien al top 10 *Puntuaciones*
* Contar los puntos del jugador. *Juego.*
* Descontar una vida en caso de que el fantasma alcance a Pac-man. Y sacarlo del juego en caso de que ya haya perdido 3 vidas. *Juego.*
* Activar el poder de volverse invencible si se come una fruta. *Juego.*
* Si pacman toca un fantasma teniendo poderes, se debe enviar al fantasma a la casilla donde inicio el juego. *Juego.*
* Mostrar graficamente el estado del juego. *Visualizador.*

**Colabaraciones**

* Pacman y tablero: Activar el poder de Pacman para comer fantasmas.
* Pacman, tablero, fantasma, actualizador: Si pacman toca un fantasma teniendo el poder, el actulizador debe enviar una nueva coordenada del fantasma y el tablero debe modificarse.
* Puntuaciones y juego. Puntuaciones almacena los distintos puntajes obtenidos por distintos jugadores, y se los pasa al juego para que los imprima.
* Puntuaciones y juego. Juego va contando la puntuacion actual del jugador, y en los casos donde corresponda le pasa la puntuacion a Puntuaciones para incluirlo en el top 10.
* Pacman, tablero, fantasma, juego, actaulizador: Si pacman toca un fantasma sin poder, la posicion de pacman y los fantasmas regresa al inicio y se le descuenta una vida al jugador.
* Juego y visualizador. El juego almacena y estructura toda la información del estado del juego y visualizador se encarga de volver esto a algo gráfico.

**3ra Iteración:**

Elementos relacionados con el dominio:

* Pac-man (information holder).
* Fantasma (information holder).
* Casilla (information holder).
* Tablero (structurer).
* Puntuaciones (Information holder)
* Actualizador (controller)

Agrupadores:

* Juego (structurer).

Elementos relacionados con el sistema:

* Cargador de mapa (service provider).
* Gestor de mapas (service provider)
* Agregador de fantasmas (service provider).
* Visualizador (interfacer)

Responsabilidades:

* Pedir el nombre del archivo del tablero. *Cargador de mapa.*
* Cargar la información del tablero. *Cargador de mapa.*
* Alguien debe reconocer los movimientos de todos los elementos. *Tablero.*
* Actualizar el tablero. *Actualizador.*
* Alguien debe poder agregar un nuevo fantasma al tablero. *Agregador de fantasmas.*
* Alguien debe preguntar si seguir con el mismo tablero o cambiarlo. *Gestor de mapas.*
* Alguien debe contar cuántas vidas quedan. *Juego.*
* Almacenar historial de puntuaciones. *Puntuaciones.*
* Mostrar top 10. *Puntuaciones.*
* Comparar top 10 para evaluar si agregar uno nuevo. *Puntuaciones.*
* Añadir y quitar alguien al top 10 *Puntuaciones*
* Contar los puntos del jugador. *Juego.*
* Descontar una vida en caso de que el fantasma alcance a Pac-man. Y sacarlo del juego en caso de que ya haya perdido 3 vidas. *Juego.*
* Activar el poder de volverse invencible si se come una fruta. *Juego.*
* Si pacman toca un fantasma teniendo poderes, se debe enviar al fantasma a la casilla donde inicio el juego. *Juego.*
* Mostrar graficamente el estado del juego. *Visualizador.*

**Colabaraciones**

* Pacman y tablero: Activar el poder de Pacman para comer fantasmas.
* Pacman, tablero, fantasma, actualizador: Si pacman toca un fantasma teniendo el poder, el actulizador debe enviar una nueva coordenada del fantasma y el tablero debe modificarse.
* Puntuaciones y juego. Puntuaciones almacena los distintos puntajes obtenidos por distintos jugadores, y se los pasa al juego para que los imprima.
* Puntuaciones y juego. Juego va contando la puntuacion actual del jugador, y en los casos donde corresponda le pasa la puntuacion a Puntuaciones para incluirlo en el top 10.
* Pacman, tablero, fantasma, juego, actaulizador: Si pacman toca un fantasma sin poder, la posicion de pacman y los fantasmas regresa al inicio y se le descuenta una vida al jugador.
* Juego y visualizador. El juego almacena y estructura toda la información del estado del juego y visualizador se encarga de volver esto a algo gráfico.

**Iteración Final:**

Elementos relacionados con el dominio:

* Pac-man (controller).
* Fantasma (controller).
* Casilla (information holder).
* Tablero (structurer).
* Puntuaciones (Information holder)
* Actualizador (controller)

Agrupadores:

* Juego (structurer).

Elementos relacionados con el sistema:

* Cargador de mapa (service provider).
* Gestor de mapas (service provider)
* Agregador de fantasmas (service provider).
* Visualizador (interfacer)

Cosas que fluyen dentro del sistema:

* Numero de vidas (information holder)

Responsabilidades:

* Pedir el nombre del archivo del tablero. *Cargador de mapa.*
* Cargar la información del tablero. *Cargador de mapa.*
* Alguien debe reconocer los movimientos de todos los elementos. *Tablero.*
* Actualizar el tablero. *Actualizador.*
* Alguien debe poder agregar un nuevo fantasma al tablero. *Agregador de fantasmas.*
* Alguien debe preguntar si seguir con el mismo tablero o cambiarlo. *Gestor de mapas.*
* Alguien debe contar cuántas vidas quedan. *Juego.*
* Almacenar historial de puntuaciones. *Puntuaciones.*
* Mostrar top 10. *Puntuaciones.*
* Comparar top 10 para evaluar si agregar uno nuevo. *Puntuaciones.*
* Añadir alguien al top 10 *Puntuaciones*
* Quitar alguien del top 10 *Puntuaciones*
* Contar los puntos del jugador. *Juego.*
* Descontar una vida en caso de que el fantasma alcance a Pac-man. *Juego.*
* Sacar a PacMan del juego en caso de que ya haya perdido 3 vidas. *Juego.*
* Activar el poder de volverse invencible si se come una fruta. *Juego.*
* Si pacman toca un fantasma teniendo poderes, se debe enviar al fantasma a la casilla donde inicio el juego. *Juego.*
* Mostrar graficamente el estado del juego. *Visualizador.*
* *Si pacman toca una bonificacion se le suma puntos extra. Juego*

**Colabaraciones**

* Pacman y tablero: Activar el poder de Pacman para comer fantasmas.
* Pacman, tablero, fantasma, actualizador: Si pacman toca un fantasma teniendo el poder, el actulizador debe enviar una nueva coordenada del fantasma y el tablero debe modificarse.
* Puntuaciones y juego. Puntuaciones almacena los distintos puntajes obtenidos por distintos jugadores, y se los pasa al juego para que los imprima.
* Puntuaciones y juego. Juego va contando la puntuacion actual del jugador, y en los casos donde corresponda le pasa la puntuacion a Puntuaciones para incluirlo en el top 10.
* Pacman, tablero, fantasma, juego, actaulizador, numero de vidas: Si pacman toca un fantasma sin poder, la posicion de pacman y los fantasmas regresa al inicio y se le descuenta una vida al jugador.
* Juego y visualizador. El juego almacena y estructura toda la información del estado del juego y visualizador se encarga de volver esto a algo gráfico.
* Pacman, tablero, juego: Al Pacman tocar una bonificacion en el tablero, se le suma una puntuacion extra.
* Juego, Pacman e Información de Vidas: Basado en la informacion de las vidas y estado del juego se sabe si Pacman esta vivo o no.
* Juego, Visualizador, Cargador de Mapas y Gestor de Mapas: Al acabar un mapa, se le pregunta al jugador si desea cambiar de mapa. Si se desea hacer un cambio de mapa se carga el nuevo mapa y se gestiona su manejo.

**Reflexion:**

Una reflexión sobre el diseño obtenido:

* En este caso, una desventaja sería que cada vez que haya un movimiento el tablero se debe actualizar, lo que podría ser costoso e ineficiente.
* Un trade-off que escogimos, fue tener un estilo de control centralizado, ya que por una parte este nos permite tener todo controlado desde la clase juego y tablero, pero en un futuro si crece o cambia la aplicación se puede dificultar realizar estas acciones.
* Otro trade-off es que decidimos tener una clase central de Juego en donde tiene subfunciones como contar la puntuación. Esto es una ventaja ya que no hay tanta subdivisión y especialización que pueda hacer el sistema muy grande y enredado, pero puede llegar a ser una desventaja si juego llega a crecer mucho y se depende demasiado sobre este rol.

Reflexión sobre el proceso de diseño:

* Un elemento que presento dificultad fue decidir si debiese existir el information holder de Información de Vidas, ya que nos tocó evaluar que tan pertinente y favorable era dividir esta función de la clase principal del Juego. Al final decidimos a favor de separar esta función ya que permite un manejo mucho más estable y adaptable a las circunstancias.
* Un elemento que nos causó dudas al principio fue decidir el estereotipo de los roles como pacman y fantasma, ya que dudamos entre escoger un information holder o un controller. Al final decidimos ponerlos como controller ya que estos además de sostener información también realizan acciones que son fundamentales para el sistema