

Objetos/Roles:

1. Identifique los componentes candidatos de la solución utilizando las diferentes perspectivas estudiadas en clase:

- El dominio del problema y los elementos con los cuales trabaja el sistema.

- Pac-man

- Galletas

-Fantasmas

-Mapas

-Tablero

-Frutas

-Puntaje

-Leaderboard

- Las responsabilidades del sistema (para buscar elementos que cumplan con esas responsabilidades o con partes de esas responsabilidades).

-Esquivar

-Comer

-Almacenar puntaje

-Cambiar estado de invencibilidad

-Restar vida

-Pausar juego

-Reiniciar juego

-Crear nueva partida

-Actualizar LeaderBoard

-Registrar nombre

-Crear nueva personalidad para fantasmas

-Game Over

-Win

- Elementos conectados o afectados por nuestro sistema (digitales o reales).

-Detector de colisiones

-Contador de galletas consumidas

-Actualizador de LeaderBoard

-Controlador del Jugador

- Información que fluya por el sistema.

-Puntaje

-Invencibilidad

-Mejores Puntajes

- Actividades de toma de decisiones, control y coordinación.
- Estructuras y grupos de objetos
- Representación de objetos del mundo real que deban aparecer dentro del sistema.

2. Agrupe elementos de los que identificó.

El tablero podría estar compuesto por una agregación de casillas, quienes se encargan de determinar si están vacías, tienen una fruta o tienen una galleta. Así mismo, cuando Pac-Man pasa sobre ellas, las casillas deberían ser capaces de cambiar a un estado “vacío”, mediante el cual el tablero pueda determinar que todas las galletas han sido comidas por Pacman, y por tanto el jugador ha ganado.

Pacman es el encargado de definir el estado de los fantasmas, de manera que cuando éste se coma una fruta, debe avisarles a los fantasmas que deben cambiar de comportamiento. Además, éste debe informarles a los fantasmas sobre su posición en todo momento, para que ellos puedan ejecutar su comportamiento programado en concordancia.

En cuanto a la interacción Pacman – Fantasmas, lo más pertinente es que sean estos segundos los que determinen cuando se han chocado entre sí, y así determinar qué acción llevar. De esta forma se distribuye la responsabilidad de saber si ha ocurrido un choque entre los 4 fantasmas en lugar de sólo sobre Pacman. En caso de que colisionen, el fantasma respectivo está encargado de informarle a Pacman que ha perdido una vida, y que todos deben volver a su posición inicial para reanudar el juego.

Es necesario crear una clase llamada inicio que implemente a casilla y que represente el punto inicial en el que los fantasmas inician, y donde son traídos de vuelta al ser comidos por Pacman, o cuando éste pierde una vida. Cuando un fantasma informe que Pacman ha perdido, Inicio estaría encargado de informarle a Pacman y a cada fantasma cuál es su respectiva posición inicial en el tablero, para que éstos vuelvan y se pueda continuar con el juego.

3. Asígneles estereotipos a los elementos.

En el diseño propuesto encontramos tres grandes colaboraciones que permiten que el juego se ejecute de forma adecuada: Pacman se limita a recibir las instrucciones del usuario y a informarle a los fantasmas de su posición y de que se ha comido una fruta; las casillas están encargadas de determinar si Pacman ha pasado sobre ellas, y así informarle al tablero que ya no hay nada en ellas para que éste pueda saber cuándo el jugador ha ganado; por último, los fantasmas conocen la posición de Pacman permanentemente, de forma que pueden perseguirlo o huir de él dependiendo de su estado, y están encargados de determinar

Mediante estas colaboraciones, se pueden realizar las acciones fundamentales que componen el juego de Pacman: Comer galletas y frutas, definir el estado de los fantasmas y las interacciones que tienen en ambos cuando colisionan con Pacman, y saber cuándo el jugador ha ganado.

4.

Pese a que en un principio se evaluó la posibilidad de hacer que Pacman contuviera toda la información del juego, siendo este el eje central de toda la aplicación, rápidamente se descartó esta idea, y se optó por un diseño mejor distribuido entre los diferentes elementos que componen la aplicación, como se mencionó en los puntos anteriores. Esto mejora significativamente la fluidez, los tiempos de respuesta y reduce la dependencia de todo el programa hacia Pacman (de forma que, si este elemento deja de funcionar, el resto no se ve afectado en sus funciones individuales).

