

Explicación de diagrama de clases

Se tiene una clase abstracta llamada "Ente", esta tiene los atributos ALTO, el cual es constante de tipo int; ANCHO, también constante de tipo int; dirección, de tipo String; imagen, de tipo ImageIcon; juego, de tipo Juego; xTemp, de tipo int el cual señala la posición de las abscisas iniciales temporales de su ubicación; yTemp, del mismo tipo que el anterior y el cual tiene la misma función. Sus métodos son pintar, el cual recibe como parámetro un elemento de tipo Graphics, el cual se encarga de establecer el diseño a cada una de las entidades y el método colision que indica cuándo un ente choca con otro, una de las funciones claves del programa.

De la clase anterior descienden las subclases "Enemigo", "Personaje" y "Recompensa".

Con respecto a la primera, tiene como atributos a 'velocidadX' de tipo int, el cual nos brinda información sobre la velocidad del enemigo en el eje x, además esta velocidad depende del nivel que el usuario escoge antes de comenzar el juego, su otro atributo es 'velocidadY', este tiene la misma funcionalidad que el anterior, con la diferencia de que se orienta al eje Y, por último contiene al atributo 'nombre', ya que puede variar, esto justificado en que nuestro juego cuenta con 4 enemigos, estos son Perrovaca, El Profesor, La Bika y La Trika. Sobre sus métodos, solo contiene uno y es 'moverse', este se encarga de darle una movilidad aleatoria por todo el mapa a los enemigos, los cuales tienen la capacidad de restarle la vida al personaje cuando colisionen con él.

Con respecto a la segunda subclase, este tiene los atributos 'endGame' de tipo booleano, el cual nos señala si el jugador perdió el juego o no; el atributo 'pisarPasto', parecido al anterior, con la particularidad de que si el personaje pisa las áreas verdes pierde automáticamente; 'nombre', de tipo String y es captado mediante un textField cuando el usuario ingresa su nickname antes de iniciar el juego; el atributo 'movimiento' que nos muestra la posición inicial del personaje; para finalizar con esta clase, contemplamos su único método 'moverse' el cual es distinto al del enemigo ya que este tiene como parámetros las teclas A, W, S y D en forma de eventos de teclado que sirven para que se desplace a la izquierda, arriba, abajo o a la derecha respectivamente.

Con respecto a la tercera subclase, esta solo contiene el método 'moverse', debemos de señalar que este es un método abstracto, es por eso que en cada uno de las subclases varía y en esta clase en particular trata de una aparición aleatoria en el mapa el cual aporta al personaje mayor puntaje y vidas.

Otra de las clases principales que tenemos es a la clase "Juego", esta tiene la relación de agregación con las clases "Enemigo", "Recompensa", "Personaje" y "Laberinto". Entre sus atributos tenemos a 'laberinto' de tipo Laberinto, el cual es su base y una de los atributos más importantes; 'personaje' del tipo Personaje, que controla el usuario; bica, de tipo "Enemigo"; trica, de tipo "Enemigo"; 'profe', de tipo "Enemigo"; 'recomp' de tipo "Recompensa" al igual que 'comida'; 'vidas' de tipo int que se inicializa en 5 y representa las oportunidades que tiene el jugador; 'puntaje' de tipo int, el que se incrementa a medida que el jugador va recorriendo el mapa y recoge recompensas; 'PTJMAX' de tipo int constante que establece un puntaje máximo que cuando el jugador lo alcance se declara ganador; 'velocidadEnemigos', de tipo int que señala la rapidez que tendrán los enemigos; 'nivelDif', de tipo int que trata sobre el nivel de dificultad que tendrá el juego y comprende los niveles de fácil, intermedio y difícil; 'perrovaca' que también es de tipo Enemigo y 'mapaSelect' de tipo int para seleccionar un mapa de los tres disponibles. Sus métodos son 'inicio()', justamente para iniciar el juego;

'gameOver()', cuando se acaba el juego; 'paint' que recibe como parámetro un elemento de tipo Graphics que se encarga de darles diseños a los elementos como mapas o componentes y move() que inicializa el movimiento de todos los personajes.

Y como última clase tenemos a "Laberinto", sus atributos son 'ANCHO', de tipo int y constante, que estable el ancho del mapa de laberinto; 'ALTO', de tipo int y constante, que define el alto del mapa; 'NCOLUMNAS' de tipo int y constante que señala la cantidad de columnas que tendrá la matriz en la que se basa el laberinto y 'NFILOS' de tipo y constante que realiza lo mismo que lo anterior, pero en las filas. Su único método es 'pintarLaberinto()' que recibe como parámetros un elemento de tipo Graphic y uno de tipo int y realiza lo que señala su nombre.

Mapas

Tercero

Lo que diferencia a este nivel de los anteriores es que fue diseñado tomando como referencia una foto real de nuestra facultad, como puede apreciarse en la imagen:



Fuente: GoogleMaps

Finalmente, se decidió mantener una estética similar a la de los otros, por lo que optamos por utilizar sprites de diversos videojuegos para dibujar este laberinto.