**VideoGame**

**Contexto**

Se debe crear un prototipo de videojuego basado en Space invader o Plantas vs Zombies. El juego aumenta su dificultad a medida que pasa el tiempo, razón por la que habrá enemigos infinitos y el jugador gana entre más tiempo esté dentro del juego. Debe mostrar el tiempo transcurrido y el puntaje del jugador, así como una pantalla de instrucciones y resumen del juego.

**Entidades (clases)**

Player

Enemy

Screen

**Requerimientos funcionales**

1.Mostrar pantalla de inicio

2.Permitir iniciar el juego

3.Mostrar avatar de jugador

4.Permitir al jugador mover su avatar con las teclas ASDW

5.Permitir al jugador lanzar osos de peluche

6.Mostrar el tiempo transcurrido en el juego

7.Mostrar el puntaje del jugador

8.Mover a los enemigos

9.Eliminar enemigos

10.Mostrar pantalla de resumen

11.Mostrar pantalla de instrucciones

12.Mostrar pantalla de información del juego

13.Permitir regresar a la pantalla de inicio

14.Aumentar el nivel de dificultad de acuerdo con el tiempo

15.Aumentar score según tiempo trascurrido

**RF1:**

|  |  |
| --- | --- |
| **RF1** | |
| **Descripción** | El **programa** debe mostrar pantalla de inicio |
| **Entradas** | -- |
| **Salidas** | -- |
| **Precondición** | Haber cargado imagen previamente |
| **Postcondición** | -- |

**RF2:**

|  |  |
| --- | --- |
| **RF2** | |
| **Descripción** | El **programa** debe permitir iniciar el juego |
| **Entradas** | posX y posY del mouse, posX y posY de botón play |
| **Salidas** | -- |
| **Precondición** | El programa debe haber cargado la pantalla de inicio |
| **Postcondición** | Pantalla de juego se carga, el tiempo de juego inicia |

**RF3:**

|  |  |
| --- | --- |
| **RF3** | |
| **Descripción** | El **programa** debe mostrar avatar del jugador |
| **Entradas** | -- |
| **Salidas** | -- |
| **Precondición** | El programa debe haber cargado la imagen avatar previamente |
| **Postcondición** | -- |

**RF4:**

|  |  |
| --- | --- |
| **RF4** | |
| **Descripción** | **El programa** debe permitir al usuario mover su avatar |
| **Entradas** | Swicth teclas ASDW |
| **Salidas** | -- |
| **Precondición** | El programa debe haber cargado la pantalla de juego |
| **Postcondición** | Avatar se mueve según la tecla presionada |

**RF5:**

|  |  |
| --- | --- |
| **RF5** | |
| **Descripción** | **El programa** debe permitir al usuario lanzar osos de peluche |
| **Entradas** | ClickPressed |
| **Salidas** | -- |
| **Precondición** | El programa debe haber cargado la pantalla de juego y usuario haber hecho click izquierdo |
| **Postcondición** | -- |

**RF6:**

|  |  |
| --- | --- |
| **RF6** | |
| **Descripción** | **El programa** debe mostrar tiempo transcurrido |
| **Entradas** | Tiempo int, int |
| **Salidas** | Tiempo en números enteros (minutos: segundos) |
| **Precondición** | El programa debe haber cargado pantalla de juego |
| **Postcondición** | -- |

**RF7:**

|  |  |
| --- | --- |
| **RF7** | |
| **Descripción** | **El programa** debe mostrar el puntaje del jugador |
| **Entradas** | Puntaje int |
| **Salidas** | Puntaje en número entero |
| **Precondición** | El programa de haber cargado pantalla de juego |
| **Postcondición** | -- |

**RF8:**

|  |  |
| --- | --- |
| **RF8** | |
| **Descripción** | **El programa** debe mover enemigos |
| **Entradas** | posX y posY de enemigos |
| **Salidas** | -- |
| **Precondición** | El programa de haber cargado pantalla de juego |
| **Postcondición** | Enemigos se mueven hacia el avatar del jugador |

**RF9:**

|  |  |
| --- | --- |
| **RF9** | |
| **Descripción** | **El programa** debe eliminar enemigos |
| **Entradas** | posX y PosY de peluche, posX y posY de enemigo |
| **Salidas** | -- |
| **Precondición** | El programa debe haber cargado pantalla de juego, el jugador debe haber disparado algún peluche, posX y posY de peluche debe haber coincido con posY y posY de enemigo |
| **Postcondición** | Enemigo es eliminado de la pantalla |

**RF10:**

|  |  |
| --- | --- |
| **RF10** | |
| **Descripción** | **El programa** debe mostrar pantalla resumen |
| **Entradas** | -- |
| **Salidas** | Entero de puntaje, entero de tiempo, entero de enemigos derrotados |
| **Precondición** | El programa de haber cargado pantalla de juego, el jugador debe haber perdido o terminado el juego |
| **Postcondición** | -- |

**RF11:**

|  |  |
| --- | --- |
| **RF11** | |
| **Descripción** | **El programa** debe mostrar pantalla de instrucciones |
| **Entradas** | posY y posY de botón “instrucciones”, posX y posY del mouse |
| **Salidas** | -- |
| **Precondición** | El programa debe haber cargado pantalla de inicio, usuario debe haber clickeado en el botón “instrucciones” |
| **Postcondición** | -- |

**RF12:**

|  |  |
| --- | --- |
| **RF12** | |
| **Descripción** | **El programa** debe mostrar pantalla de información del juego |
| **Entradas** | -- |
| **Salidas** | -- |
| **Precondición** | El programa debe haber cargado imagen de pantalla de información del juego previamente |
| **Postcondición** | -- |

**RF13:**

|  |  |
| --- | --- |
| **RF13** | |
| **Descripción** | **El programa** debe permitir regresar a pantalla de inicio |
| **Entradas** | posX y posY del botón “regresar”, posX y posY del mouse |
| **Salidas** | -- |
| **Precondición** | El programa de haber cargado una pantalla diferente a la de inicio, usuario debe haber clickeado en botón “regresar” |
| **Postcondición** | -- |

**RF14:**

|  |  |
| --- | --- |
| **RF14** | |
| **Descripción** | **El programa** debe aumentar el nivel de dificultad de acuerdo al tiempo trascurrido |
| **Entradas** | -- |
| **Salidas** | -- |
| **Precondición** | Cómo precondición el tiempo debe haber empezado a correr |
| **Postcondición** | -- |

**RF15:**

|  |  |
| --- | --- |
| **RF15** | |
| **Descripción** | **El programa** debe aumentar el score según el tiempo transcurrido |
| **Entradas** | Int score |
| **Salidas** | Entero de puntaje |
| **Precondición** | Cómo precondición el tiempo debe haber iniciado a correr |
| **Postcondición** | -- |

**Requerimientos no funcionales**

1. El código debe funcionar en el lenguaje de programación Java o JavaScript
2. Debe usar POO
3. Debe contar con una coherencia y alta calidad gráfica
4. Debe ser fácil de usar por el usuario, incluir componentes que indiquen al usuario cuando un elemento es “clickeable” por ejemplo.

**Link Behance:** <https://www.behance.net/gallery/114908377/Sweet-Dreams-video-game-prototype>

**Diagrama UML:**

