Integrantes:

Agustin Rodriguez Dei Rossi

Andres Fedeli

Andres Nicolas Lattuca

Giuliana Ferraro

Tomás Perez

**UTN - FRM - Tecnicatura Superior en Programación**

**METODOLOGÍA DE LA INVESTIGACIÓN**

**Proyecto de desarrollo de software**

**Introducción**

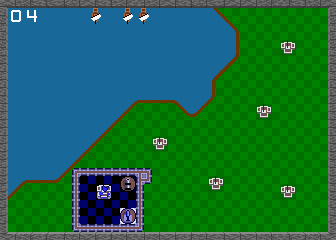
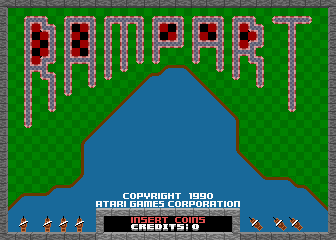
Para el proyecto de Metodología de la Investigación, se optó por el desarrollo de un software de videojuego del género *tower defense*. Se seguirá el marco de desarrollo ágil SCRUM. Se trabajará de modo colaborativo presencial y remotamente, con ayuda del servicio de versionado GitHub.

Se eligió desarrollar en este subgénero debido a la versatilidad del mismo, pues da la posibilidad de crear un juego sencillo pero que mantenga el dinamismo de los videojuegos de estrategia. Además, los antecedentes del género indican que es apropiado como proyecto para equipos reducidos y desarrolladores independientes. Incorporando los elementos básicos que definen al tower defense, se espera desarrollar un software simple, sin gran detalle gráfico, pero que encuadre en el género.

**Marco teórico**

El *tower defense* es un subgénero de los juegos de estrategia. El objetivo de los juegos tower defense consiste en evitar que los enemigos lleguen a la base/torre que se desea proteger. Si esto sucediera, el jugador pierde. Para lograr defender la base/torre se colocan unidades de defensa a lo largo del recorrido de los enemigos. Estas puede disparar proyectiles, lasers, etc. A lo largo del juego las unidades enemigas comienza a volverse más fuertes por lo que es necesario mejorar el armamento de las unidades de defensa. Mientras más enemigos se destruyen, más recursos se obtienen. Estos recursos son utilizados para mejorar el armamento, obtener mejores unidades, etc.

**Antecedentes**Los primeros juegos del género *tower defense* datan de la década del 1980. Se considera que los precursores fueron los juegos de arcade Space Invaders (1978) y Missile Command (1980). El clásico de estrategia Rampart, del año 1990, es uno de los primeros juegos de arcade que se ajusta al estilo de los tower defense modernos. ***Rampart***, que fue diseñado por John Salwitz y lanzado por Atari Games, combina los géneros *shooter* y *puzzle*, al dar al jugador la oportunidad de construir un castillo y muros de defensa con piezas al estilo del Tetris.



(https://www.arcade-museum.com/game\_detail.php?game\_id=9263)

No obstante el éxito de este lanzamiento, no surgieron muchos otros juegos del género hasta la adopción masiva del mouse. Desde entonces la popularidad de los tower defense incrementó considerablemente, sobre todo impulsada por desarrolladores independientes.

Entre los años 2007 y 2008, con la implementación de Adobe Flash, los juegos tower defense se convirtieron en un fenómeno de los juegos RTS (de estrategia en tiempo real) para PC. Dos ejemplos dignos de mención de esta época son Flash Element Tower Defense y Desktop Tower Defense. ***Flash Element TD*** comenzó como un proyecto personal de David Scott, quien se inspiró WarCraft III Element TD y decidió practicar sus habilidades de programación creando su propiotower defensesimplificado para Flash*.* Asimismo, ***Desktop Tower Defense*** fue desarrollado por Paul Preece, un conocido de David Scott, quien quiso crear un tower defense para navegador que permitiera erigir torres en forma de laberinto para bloquear el avance del enemigo. Ambos proyectos son emblemáticos del desarrollo independiente de videojuegos de baja complejidad gráfica.

El éxito de estos títulos produjo que el género siguiera evolucionando, tanto en PC como en consolas, y adaptándose a las nuevas plataformas y tecnologías móviles y táctiles. Actualmente los tower defense son muy variados en su complejidad y en sus características gráficas, temáticas y formales (modalidad de juego *single-player*/*multi-player*, en tiempo real/por turnos, cámara, punto de vista), etc.

**Backlog**

Uno de los requisitos de proyecto es que debe desarrollarse para Linux. Por lo tanto se ha elegido un entorno multiplataforma para el lenguaje C++. Se desarrollará en el IDE AppGameKit, que facilita el manejo de recursos gráficos.

Inicio del proyecto: 05/04/2017

Fin del proyecto: 07/06/2017

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **TAREA** | **REQUISITO** | **PRIORIDAD** | **ESTADO** | **ESTIMACIÓN** |
| **1** | Conceptualización | Experiencia de juego coherente | Media | Completa | 6 |
| **2** | Diseño del sistema: menú e interfaz de usuario | Interacción del usuario con el software | Media | En progreso | 12 |
| **3** | Diseño del sistema: activos gráficos | Interacción del usuario con el software | Media | En progreso | 12 |
| **4** | Codificación: respuesta al puntero | Posibilidad de jugar | Muy alta | No iniciada | 2 |
| **5** | Codificación: configuración del escenario | Posibilidad de jugar | Muy alta | No iniciada | 7 |
| **6** | Codificación: dinámica y movimiento | Posibilidad de jugar | Muy alta | No iniciada | 8 |
| **7** | Codificación: personaje y defensa | Posibilidad de jugar | Muy alta | No iniciada | 16 |
| **8** | Codificación: enemigos | Posibilidad de jugar | Muy alta | No iniciada | 10 |
| **9** | Codificación: reglas y aspectos lógicos | Posibilidad de jugar | Muy alta | No iniciada | 20 |
| **10** | Codificación: puntaje | Avance del personaje en el juego | Alta | No iniciada | 8 |
| **11** | Codificación: tiempo de juego | Avance del personaje en el juego | Alta | No iniciada | 8 |
| **12** | Pruebas | Jugabilidad adecuada | Alta | No iniciada | 12 |
| **13** | Puesta en marcha | Jugabilidad adecuada | Alta | No iniciada | 2 |
| **14** | Documentación | Acceso a material de referencia | Media | En progreso | 5 |

**Actas de reuniones**

5 de abril de 2017

- Constitución del grupo y planteo de posibles temas. Definición de roles del modelo SCRUM:

* Scrum Master: Agustin Rodriguez Dei Rossi
* Product Owner: Giuliana Ferraro
* Development Team: Andres Fedeli; Andres Nicolas Lattuca, Tomas Perez

14 de abril de 2017

- Preparación del product backlog: objetivos y prioridades.

- Definición del género del juego (defensa de torre) y del entorno de desarrollo. IDE Appgamekit compatible con Windows y Linux, con su pseudocódigo para C++.

19 de abril de 2017

- Investigación de antecedentes.

- Avance en la lectura de la documentación del IDE y del lenguaje.

- Búsqueda de posibles recursos gráficos para la interfaz del juego.

26 de abril de 2017

- Planificación del sprint:objetivo del sprint y las tareas necesarias para conseguirlo.

03 de mayo de 2017

- Creación de repositorios en GitHub.

BIBLIOGRAFÍA

<http://julian.togelius.com/Avery2011Computational.pdf>

<https://en.wikipedia.org/wiki/Desktop_Tower_Defense>

<https://en.wikipedia.org/wiki/Flash_Element_TD>

<https://www.raywenderlich.com/30636/top-developer-interviews-ironhide-game-studio>

<https://mygaming.co.za/news/features/6341-tower-defense-a-brief-history.html>

<http://www.scrummanager.net/files/sm_proyecto.pdf>

<https://revistas.ucm.es/index.php/HICS/article/viewFile/45178/42539>