

Manual del Proyecto.

Integrantes

Dylan Ledesma (102876)
Ricardo Riluizaga (87528)
Federico Codino(103533)

Enunciado

El trabajo práctico este cuatrimestre consistió en realizar la implementación del videojuego DUNE 2000. Este es un juego de estrategia en tiempo real(RTS) programado por Intelligent Games y lanzado por Westwood Studios en 1998 para Microsoft Windows.

El objetivo del videojuego consiste en destruir a todos los adversarios y quedarse con el control del planeta Arrakis, conocido como Dune.

Planificación del Proyecto

El objetivo inicial fue seguir el plan brindado por la cátedra

	Alumno 1 Servidor - Modelo	Alumno 2 Modelo - Cliente	Alumno 3 Cliente - Editor
Semana 1 (17/05/2022)	Implementación del algoritmo A-Star sobre un mapa con distintos terrenos y con unidades con diferentes movilidades.	Mostrar una imagen. Mostrar una animación. Mostrar ambas en un lugar fijo o desplazándose por la pantalla (movimiento).	Emitir un sonido. Carga y guardado de información a disco. Draft de la interfaz (<i>wireframes</i>).
Semana 2 (24/05/2022)	Finalización del algoritmo de A-Star. Lógica del modelo sobre el ataque y destrucción de unidades y edificios.	Mostrar todo el mapa, incluyendo varios tipos de terrenos distintos, edificios y unidades.	Mostrar una interfaz para el jugador, excepto el mapa: crear nuevas unidades y edificios; mostrar cuales están disponibles y cuales están siendo construidas.
Semana 3 (31/05/2022)	Finalización de la lógica del modelo.	Selección de unidades y vehículos. Acciones de mover y atacar.	Finalización de la interfaz.
Semana 4 (07/06/2022)	Creación de una partida multijugador de N jugadores. Elección de mapa y con qué casa jugar.	Mostrar las unidades con animaciones. Mostrar las animaciones de los disparos y explosiones. Sonidos y música.	Editor, creación de escenarios. Carga de estos en el servidor.
Semana 5 (14/06/2022)	- Testing y corrección de bugs - Documentación	- Testing y corrección de bugs - Documentación	Finalización del editor.
Entrega el día 21/06/2022			

Semana 6 (21/06/2022)	- Testing y corrección de bugs - Documentación	- Testing y corrección de bugs - Documentación	- Testing y corrección de bugs - Documentación
Semana 7 (28/06/2022)	- Correcciones sobre Preentrega - Testing y corrección de bugs - Documentación	- Correcciones sobre Preentrega - Testing y corrección de bugs - Documentación	- Correcciones sobre Preentrega - Testing y corrección de bugs - Documentación
Entrega el día 5/07/2022			

Las tareas fueron divididas de la siguiente manera

Dylan Ledesma: Servidor - Common - Installer - Lobby - Configuración del juego - Protocolo

Ricardo Riluizaga: Cliente

Federico Codino: Editor-Lobby

A pesar de los esfuerzos hechos no pudimos seguir el plan debido a diferentes percances. Por ejemplo en la primera semana de desarrollo el grupo de Federico Codino fue disuelto y se tuvo que unir al grupo conformado por Dylan Ledesma y Ricardo Riluizaga. Si bien esto sucedió al comienzo del desarrollo, se perdieron 4-5 días de trabajo que podrían haber sido muy útiles.

Más allá de los percances sufridos, también el desarrollo se fue retrasando debido a la dificultad del TP, por lo que para semana 4 llevábamos 1 semana de atraso y para la preentrega llevábamos 2 semanas de atraso.

Herramientas

Lenguaje de programación: c++

Librerías de GUI: QT y SDL

Control de versiones: Github

IDE: Visual studio code, CLion, QtCreator

Chequeos de memoria: Valgrind

Dificultades

Las mayores dificultades fueron el manejo de las bibliotecas SDL y QT., ya que con estas bibliotecas interactuamos con el usuario.

La biblioteca SDL fue usada para el en juego en sí y la biblioteca QT fue usada para el editor y el lobby ya que facilitaba la interacción con el usuario

Otro punto dificultoso fue la implementación del protocolo, más que nada a la hora de debuggear y verificar que se manden los datos correctos.

Errores conocidos:

El editor funciona mal cuando se lo corre desde el build del proyecto, pero si se lo ejecuta desde la carpeta editor funciona bien.

Se implementó la lógica de crear y mover unidades sin embargo debido a un bug de último momento no solucionado no funciona su implementación gráfica.

Puntos Pendientes:

Que se puedan atacar entre unidades
Recolectar especia

Si volvieran a hacer el proyecto, que cambiarían (a nivel código o a nivel organizacional).

Lo que cambiaríamos sería realizar el protocolo al principio en lugar de hacer el servidor y cliente completos en paralelo y luego integrarlos con el protocolo.