

Manual de Proyecto

MicroMachines v 1.00

Integrantes

- Brian Zambelli Tello
- Matias Levi
- Florencia Rodríguez

Enunciado

Se adjunta al final de este manual el enunciado brindado por el curso.

División de Tareas

A continuación se muestra cada una de las tareas que se realizaron durante el desarrollo del trabajo junto con aquel miembro que estuvo a cargo.

Se desea aclarar que pese a dividir las tareas se recurrió a la ayuda de los demás compañeros al encontrarse con problemas a resolver o necesitar opiniones para el diseño.

Cliente

- Animación y visualización del juego (Brian)
- Grabación de partida (Flor)
- Script y conexión de Lua (Flor)
- Sonidos (Flor)
- Manejo de estados de juego y datos provenientes del server (Mati)

Server

- Mods (Flor)
- Soporte multijugador y multipartida (Mati y Flor)
- Representación de las entidades físicas y procesamiento de comandos con Box2D (Mati)

Elementos en común

- Lectura de archivos YAML de configuración (Flor)
- Estructura de envío de eventos (Mati)

- Protocolo y sockets(Flor)
- Script para la instalación del juego (Flor)
- Threads (Mati)
- Manuales (Mati, Brian y Flor)

Evolución de Proyecto

Semana 1

OBJETIVOS PROPUESTOS POR LA CÁTEDRA

- Draft de modelo
- Prueba de concepto de Lua
- Prueba de concepto bibliotecas dinámicas
- Mostrar imagenes con SDL, mostrar animaciones, movimiento.

TAREAS REALIZADAS

- Draft de modelo con algunas user stories
- Prueba de concepto de Lua
- Mostrar imagenes en SDL, mostrar animaciones también.

Semana 2

OBJETIVOS PROPUESTOS POR LA CÁTEDRA

- Pista, sus exteriores y autos moviéndose libremente
- Renderizado del escenario incluyendo la cámara
- Modelo de plugins en el servidor

TAREAS REALIZADAS

- Más detalle en user stories faltantes
- Conexión Server/Cliente con protocolo y conversión de información a YAML
- Modelo Box2D
- Implementación de la cámara

Semana 3

OBJETIVOS PROPUESTOS POR LA CÁTEDRA

- Autos interactuando con la pista y otros elementos dinámicos
- Animación de los elementos dinámicos

- Modelo de scripts en el cliente

TAREAS REALIZADAS

- Introducción de elementos visuales a la cámara.
- Resolución de casos particulares, tales como imágenes partidas por culpa de la misma
- Modelo de servidor con soporte multipartidas

Semana 4

OBJETIVOS PROPUESTOS POR LA CÁTEDRA

- Servidor multipartidas con partidas multijugador, con condiciones de victoria y derrota
- Comunicación con el servidor
- Pantalla de conexión
- Scripts en lua para manejar los vehículos

TAREAS REALIZADAS

- Primeros menús interactivos.
- Implementación final del modelo Servidor multijugador
- Pantalla de conexión
- Implementación de la selección de autos
- Implementación y cargado del mapa en el cliente

Semana 5

OBJETIVOS PROPUESTOS POR LA CÁTEDRA

- Incorporación de plugins
- Incorporación de scripts
- Captura de video de una partida: inicio y frenado a voluntad

TAREAS REALIZADAS

- Manual de Usuario (primera versión)
- Creación e incorporación de Plugins
- Optimización de cargas de imágenes
- Efectos variados en busca de feedback para el usuario
- Assets finales para la interface

Semana 6 (Primera Entrega)

OBJETIVOS PROPUESTOS POR LA CÁTEDRA

- Testing, correcciones y tuning del Servidor. Documentación
- Testing, correcciones y tuning del Cliente. Documentación

TAREAS REALIZADAS

- Optimización de consumo de CPU
- Correcciones de bugs encontrados al renderizar el juego
- Dibujado de objetos correctamente
- Selección de mapa

Semana 7

OBJETIVOS PROPUESTOS POR LA CÁTEDRA

- Testing, correcciones y tuning del Servidor. Documentación
- Testing, correcciones y tuning del Cliente. Documentación

TAREAS REALIZADAS

- Captura de video de una partida: inicio y frenado a voluntad.
- Adición de funcionalidad a objetos (Boost, Caja de Vida, etc) incluyendo tanto modificaciones al modelo como animaciones necesarias.
- Soporte multi partida para el Cliente
- Scripts en lua para manejar los vehículos e incorporación de estos en el cliente.
- Sistema de lobbies final para el servidor
- Leaderboard al finalizar el juego

Semana 8 (Semana final)

OBJETIVOS PROPUESTOS POR LA CÁTEDRA

- Testing de todos los elementos
- Correcciones sobre la primer entrega
- Armado de entregable

TAREAS REALIZADAS

- Manual de Usuario, Proyecto y Técnico
- Últimos efectos finales a los objetos
- Correcciones primera entrega
- Corrección de bugs
- Armado de entregable
- Testing

Inconvenientes encontrados

Durante el desarrollo del trabajo nos encontramos con varios problemas. De parte del cliente, se decidió utilizar la herramienta SDL2. El principal problema que tiene esto, es la documentación. No es que la documentación sea mala, sino que en ciertos aspectos o ciertas funciones es, literalmente, inexistente. Por ello, si se tiene alguna consulta un poco más específica, como por ejemplo para que utilizan ciertos parámetros algunas funciones, se debe caer a buscar en la red algún caso de uso similar por parte de algún otro usuario. Un ejemplo muy simple es el hecho de renderizar imágenes con una escala específica. Al buscar, encontramos que era un problema que muchos tenían con la librería y que solucionan simplemente reduciendo la escala del asset. Únicamente, si existe un método alternativo de renderizado que soluciona en parte este problema, pero, la mala documentación, hizo que se perdiera mucho tiempo buscando formas alternativas.

Por otra parte, encontramos problemas al utilizar ffmpeg como librería para realizar la grabación del juego. Pese a que se realiza un video del juego de ser deseado, este es una versión más corta de la escena real, ya que por el alto consumo del CPU debimos omitir la escritura de muchos.

Análisis de puntos pendientes

Nuevamente, empezamos de parte del cliente. Hay algunos aspectos que claramente necesitan algún que otro pulido. Por ejemplo, la pista. Hay algunas imágenes que no quedan del todo bien unas al lado de las otras. Esto es así, debido a que se tomó la decisión de que todas las tiles del escenario ocuparan la misma cantidad de pixeles para optimizar algunas funciones. Sin embargo, esto recae en tener que editar las imágenes para que funcionara. Esto no quedo siempre con el resultado esperado.

Finalmente, otra cosa que nos hubiera gustado agregar, seria alguna animación más para el auto. Por ejemplo, una al momento de impactar dos vehículos. U otra al momento de tocar tierra.

Herramientas

- Para crear los informes se utilizó Google Docs
- Para el control de versiones utilizamos Git
- El proyecto de alojó en Github
- El editor de texto utilizado fue CLion
- Para comunicarnos y organizar el flujo de trabajo utilizamos Slack y Trello.

Conclusiones

Consideramos que se alcanzaron la mayoría de los objetivos propuestos de forma interna por el grupo y todos los objetivos propuestos por la consigna del trabajo. Como se especificó en una sección anterior, dado el limitado tiempo que se tuvo para realizar el trabajo, algunas mejoras o cosas extras que se quisieron implementar no pudieron serlo.

Aun así, consideramos que el factor determinante que nos llevó a conseguir dichos objetivos fue la buena organización que tuvo el grupo debida en gran parte a una comunicación continua entre los miembros. Como ya se dijo, si bien se separó el trabajo, utilizamos la herramienta Slack. Allí, al tener canales dedicados a cada uno de los aspectos del trabajo, era muy simple no solo saber en que estaba trabajando cada uno, sino que también era simple saber que cosas quedaban por hacer.

En resumen, gracias a la buena coordinación, se pudo encarar un proyecto mucho más ambicioso que cualquiera de los trabajos prácticos anteriores correctamente.