



UNIVERSIDAD DE BUENOS AIRES

FACULTAD DE INGENIERÍA

## 75.42 - TALLER DE PROGRAMACIÓN

LENGUAJE: C++

Primer cuatrimestre de 2023

Manual de Proyecto

---

---

Bravo, Nicolas Francisco	106753	nbravo@fi.uba.ar
Chávez Cabanillas, José E.	96467	jchavez@fi.uba.ar
Davico, Mauricio	106171	mdavico@fi.uba.ar

---

---

## Índice

<b>1. Integrantes y División de tareas</b>	<b>2</b>
<b>2. Evolución del Proyecto</b>	<b>2</b>
<b>3. Inconvenientes del proyecto</b>	<b>2</b>
3.1. Cliente . . . . .	2
<b>4. Features</b>	<b>3</b>
4.1. Completados . . . . .	3
4.1.1. Cliente . . . . .	3
4.1.2. Servidor . . . . .	3
4.2. No alcanzados o no Completados . . . . .	4
4.2.1. Cliente . . . . .	4
4.2.2. Servidor . . . . .	4
<b>5. Conclusión</b>	<b>4</b>

## 1. Integrantes y División de tareas

- Bravo, Nicolas: Encargado de la implementación del Protocolo y de todo lo relacionado a la comunicación del cliente y servidor. Implementación estadísticas y su persistencia. Además, estuvo a cargo de los múltiples test del protocolo, colisiones y estadísticas del juego.
- Chávez Cabanillas, José: Encargado de la interfaz del cliente, tanto el Launcher en Qt, como todas las partes del juego (Lobby, GameSdl, EndGame) hecho en SDL.
- Davico, Mauricio: Encargado de implementar la lógica de juego del Servidor, ejecución del game loop, lógica de colisiones, manejo de tiempo, patrón Command, diferentes Factory, todo lo relacionado al GameWorld.

## 2. Evolución del Proyecto

- Semana 1: Desde el lado del cliente se inicio con el Launcher, movimiento local del cliente, primeras pruebas en Sdl. Por el lado del server se realizo la configuración del Cmake, y se plantearon bosquejos e ideas sobre la conexión cliente-servidor y protocolo.
- Semana 2: Creación del mapa, comunicación entre el cliente y el server implementando los hilos de comunicación y el hilo game del server (envío y recepción de datos de ambas partes, testeo con tiburoncín).
- Semana 3: Implementación del Gameloop, renderizado del Hud, lógica de movimiento, animaciones de los operadores.
- Semana 4: Agregado de Infectados sin comportamiento implementado, animaciones, daño de armas.
- Semana 5: Implementación de lógica de movimiento y daño de los infectados a los operadores, ataque de los infectados, primera implementación de la cámara, tamaño del escenario.
- Semana 6: Implementación de los Stats, tests, mejora de renderizados y actualización de los objetos en el cliente, implementación de las correcciones, mejora de la cámara.
- Semana 7: Arreglo de bugs, implementación de habilidades de los operadores, finalización del juego, documentación.

## 3. Inconvenientes del proyecto

### 3.1. Cliente

Al principio del proyecto se presento el inconveniente del correcto cerrado de los hilos de ejecución tanto para el Juego, como para la comunicación entre con el Servidor. Sumado a esto uno de los mayores inconvenientes que se presento para la etapa del juego fue el escalado del Hud, para que se ajustará a cualquier resolución, el renderizado correcto del mapa teniendo en cuenta el movimiento de la cámara.

## 4. Features

### 4.1. Completados

#### 4.1.1. Cliente

- Conexión y manejo de múltiples clientes.
- Creación de múltiples partidas, unión de clientes a dichas partidas.
- Gameloop del juego.
- Recepción de eventos por parte de multiples clientes.
- Procesamiento de eventos en la partida.
- Envío de Snapshots con estado actual de la partida a cada cliente.
- Colisiones.
- Movimiento de jugadores e infectados.
- Respawn aleatorios de Obstaculos e Infectados.
- Ataque por parte de Infectados.
- Diferentes lógicas de disparos dependiendo del arma por parte de los Operadores.
- Habilidades especiales habilidades de los Operadores (Granadas explosivas y de humo, y el bombardeo aéreo).
- Las acciones dependen del tiempo.
- Estadísticas para el modo de juego survival y sistema de rankings con persistencia en un archivo.
- Cheats (terminar juego, super velocidad, matar a todos los infectados, hacerse inmortal).

#### 4.1.2. Servidor

- Conectividad al Servidor.
- Ventanas para creación y unión de partida.
- Mensajes de error en Launcher.
- Música en Launcher.
- Lobby de espera hasta inicio de partida.
- Gameloop del juego.
- Detección y envío de eventos.
- Renderizado de mapas.
- Renderizado y escalado del Hud.

- Animaciones del Operadores e Infectados.
- Música de algunos estados de los Operadores e Infectados.
- Animaciones de habilidades del los Operadores (Granadas explosivas y de humo, y el bombardeo aéreo).
- Renderización del EndGame y Estadísticas.

## 4.2. No alcanzados o no Completados

### 4.2.1. Cliente

- Mejora de señalización en la cámara al alejarte del grupo de jugadores.
- Más efectos de sonido en las acciones de los Operadores e Infectados.
- Escalamiento de los objetos del mapa, Operadores, Infectados y Obstáculos.
- Animaciones de ataques especiales en los Infectados.

### 4.2.2. Servidor

- Ataques especiales de los infectados. Si bien este feature estaba en desarrollo en la rama InfectedAttack, no se mergeo a master debido a que no fue finalizado en el plazo acordado.

## 5. Conclusión

Durante el desarrollo de este proyecto, se adquirió un amplio conocimiento, tanto en el desarrollo backend como en el frontend con SDL. Se enfrentaron a dificultades comunes en el trabajo de un programador, como el trabajo en equipo, la gestión de plazos de entrega, la toma de decisiones en cuanto a prioridades, cumplimiento de requerimientos, y la necesidad de aprender nuevas herramientas de forma autodidacta (YAML, CMAKE, SDL, QT y GoogleTest), entre otras habilidades necesarias.

El aprendizaje sobre las conexiones cliente-servidor y la utilización de hilos resultó especialmente interesante. A pesar de la magnitud del proyecto, no surgieron mayores dificultades para la organización, en todo momento los tres integrantes tuvimos en claro nuestras responsabilidades y tareas asignadas, por lo que se logró llevar a cabo el trabajo y se obtuvo un producto que cumplió con los requerimientos en el tiempo acordado.

Durante el proceso de desarrollo, se mantuvieron sesiones de programación en grupo y reuniones, en las cuales se hacían puestas en común y se tomaban decisiones de implementación para secciones específicas de la aplicación. El uso de GitHub como herramienta de control de versiones fue muy organizado, por lo que no hubo dificultad en evitar conflictos entre los diferentes desarrollos.

En resumen, este proyecto ha proporcionado un crecimiento significativo en términos de habilidades técnicas y trabajo en equipo. Los resultados obtenidos son satisfactorios y las experiencias adquiridas serán de gran valor en futuros proyectos.