

# Manual de proyecto

[75.41] Taller de Programación I  
Primer cuatrimestre de 2019

Grupo 3

- Camila Bojman 101055 [camiboj@gmail.com](mailto:camiboj@gmail.com)
- Cecilia Hortas 100687 [ceci.hortas@gmail.com](mailto:ceci.hortas@gmail.com)
- Nicolas Vazquez 100338 [vazquez.nicolas.daniel@gmail.com](mailto:vazquez.nicolas.daniel@gmail.com)

## 1 División de tareas

Para la división de tareas se eligió dividir el trabajo de la forma dispuesta en el enunciado.

1. Uno se encargó de realizar el cliente gráfico, implementado todo lo relativo a la interfaz gráfica del trabajo con el uso de la herramienta **SDL** y el soporte de partidas multijugador, aplicando los conceptos vistos en la cursada de multithreading y sockets.
2. Otro se encargó de realizar el servidor, modelando el juego desde su parte física con el uso de la librería **Box2D** y el testing correspondiente con la librería **CxxTest**.
3. Finalmente, el último integrante se encargó de realizar el editor de escenarios, con el uso de la herramienta **SDL** para graficar los elementos, **ffmpeg** para brindar el soporte para la grabación de partidas y **yaml** para colocar en un archivo la información del escenario.

## 2 Evolución del proyecto

El cronograma propuesto por la cátedra pudo seguirse en las primeras semanas de resolución, no así las últimas semanas. Se consideró que hubo un aumento abrupto de tareas en la transición de la semana 3 a la semana 4, lo cual causó un atraso que fue reproduciéndose en el transcurso de las siguientes semanas. El soporte físico del juego se consideró finalizado 1 semana antes de la fecha de entrega final y el cliente gráfico 1 semana antes, lo cual significó que el soporte de sockets y multithreading se desarrolló en las últimas 2 semanas del trabajo, cuando lo previsto era empezar 2 semanas antes. En cuanto al editor, sucedió algo similar debido a que la finalización del mismo se concluyó 1 semana antes y lo esperado era algunas semanas atrás.

## 3 Inconvenientes encontrados

En cuanto al servidor, un gran inconveniente encontrado fue relativo a la librería **Box 2D**. Si bien la misma está muy bien documentada y se encuentra un amplio material sobre ello en internet, algunos elementos no son amigables para su implementación e interpone algunas dificultades. Así mismo, en lo personal algunos detalles de implementación que propone la librería resultan contrarias a la filosofía de programación orientada a objetos que se espera obtener con el trabajo, por lo que implicó un gran tiempo empleado en un diseño relativamente compatible con ella. Otro factor a tener en cuenta es que la amplia variedad de funcionalidades que se espera obtener del juego puede resultar un trabajo bastante pesado para la cantidad de tiempo estipulado y genera muchos casos borde que pueden ser difíciles de alcanzar y sobretodo debuggear en un trabajo de este nivel.

Editor: agregar dificultades

Cliente: agregar dificultades

## 4 Análisis de puntos pendientes

En cuanto al modelo de juego no se llegó a implementar los portales sobre los bloques diagonales y la muerte de Chell al cerrar una compuerta. Además se podrían pulir más los valores de las fuerzas y velocidades utilizados para darle un enfoque más realista al juego. Por ejemplo hubiese estado bueno modelar la toma de la roca de una manera en la cual se pueda hacer clic en la misma y Chell la transporte además de que sea posible poder saltar con ella. Esto no pudo realizarse en un primer lugar por falta de tiempo y en un segundo lugar por el protocolo planteado de comunicación entre cliente y servidor, ya que hacer clic en una piedra implicaba un factor de error en el cálculo de los centros de masa y se consideró que empeoraba la performance al tener que ser una búsqueda lineal sobre las mismas.

Agregar cuando terminemos el resto

## 5 Herramientas

Las herramientas utilizadas para el desarrollo del trabajo, además de las mencionadas que son obligatorias en la resolución son las que siguen:

- **IDE Clion**

Ofreció un gran apoyo para el desarrollo general del trabajo. Su herramienta de debugger fue especialmente útil así como el compilador **CMake** que ofrece, permitiendo agilizar enormemente el proceso de debuggeo y compilación.

- **Git**

Se utilizó como herramienta de control de versiones, permitiendo el desarrollo individual de cada parte del trabajo de una forma organizada y segura.

- **Latex**

Se utilizó para escribir la documentación y los manuales requeridos por el trabajo.

## 6 Conclusiones

A modo de conclusión se puede afirmar que el objetivo fue cumplido a grandes rasgos y que constituyó una gran experiencia de aprendizaje. Igualmente cabe mencionar que los objetivos planteados son bastantes para la cantidad de semanas estipulada, aunque no podría haber una alternativa frente al tiempo necesario para adquirir los conceptos a emplear.