

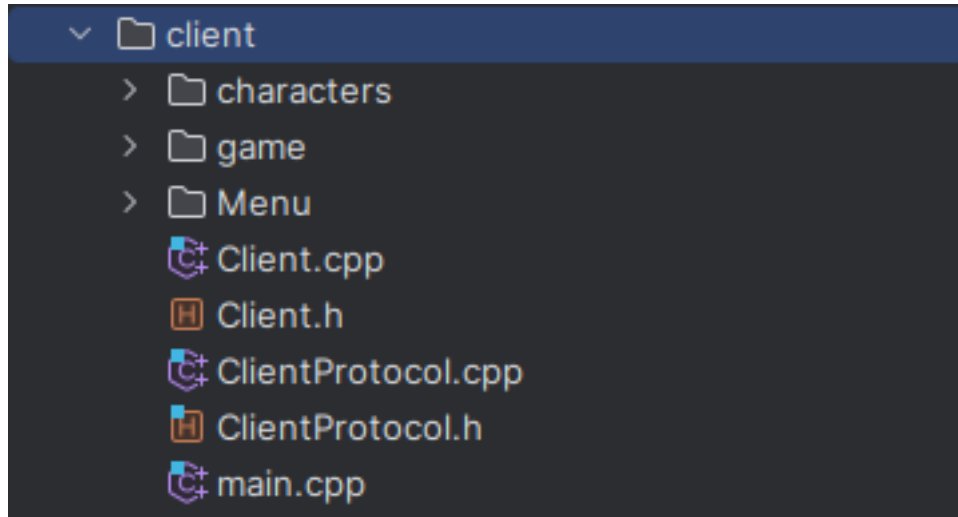
# Left 4 Dead Manual Técnico

Ignacio Garcia Pizales, Gabriel Peralta

June 2023

# 1 Lado del Cliente

## 1.1 Estructura

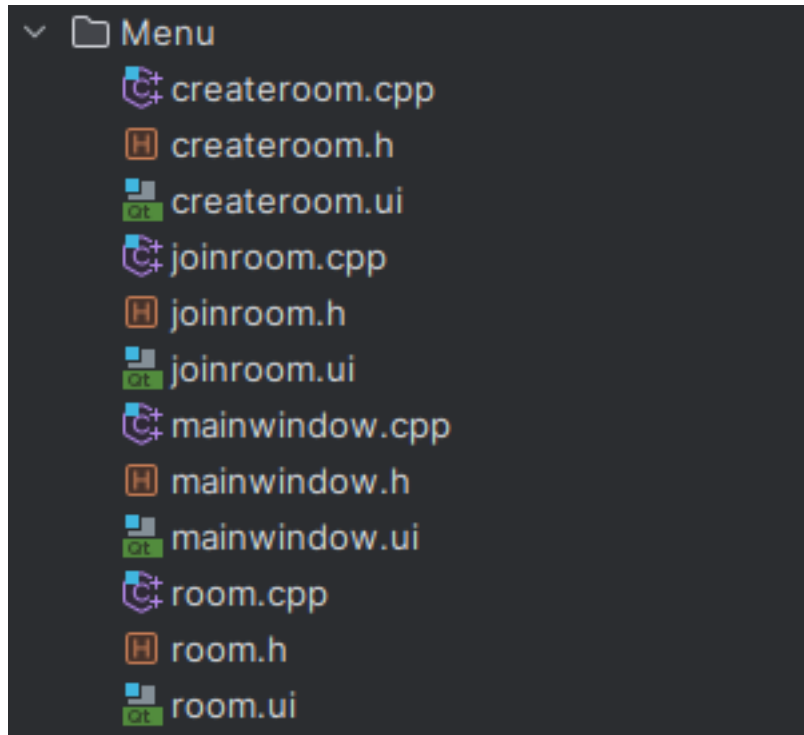


El cliente consta fundamentalmente de 4 partes. Client, Characters, Game y Menu

## 1.2 Client

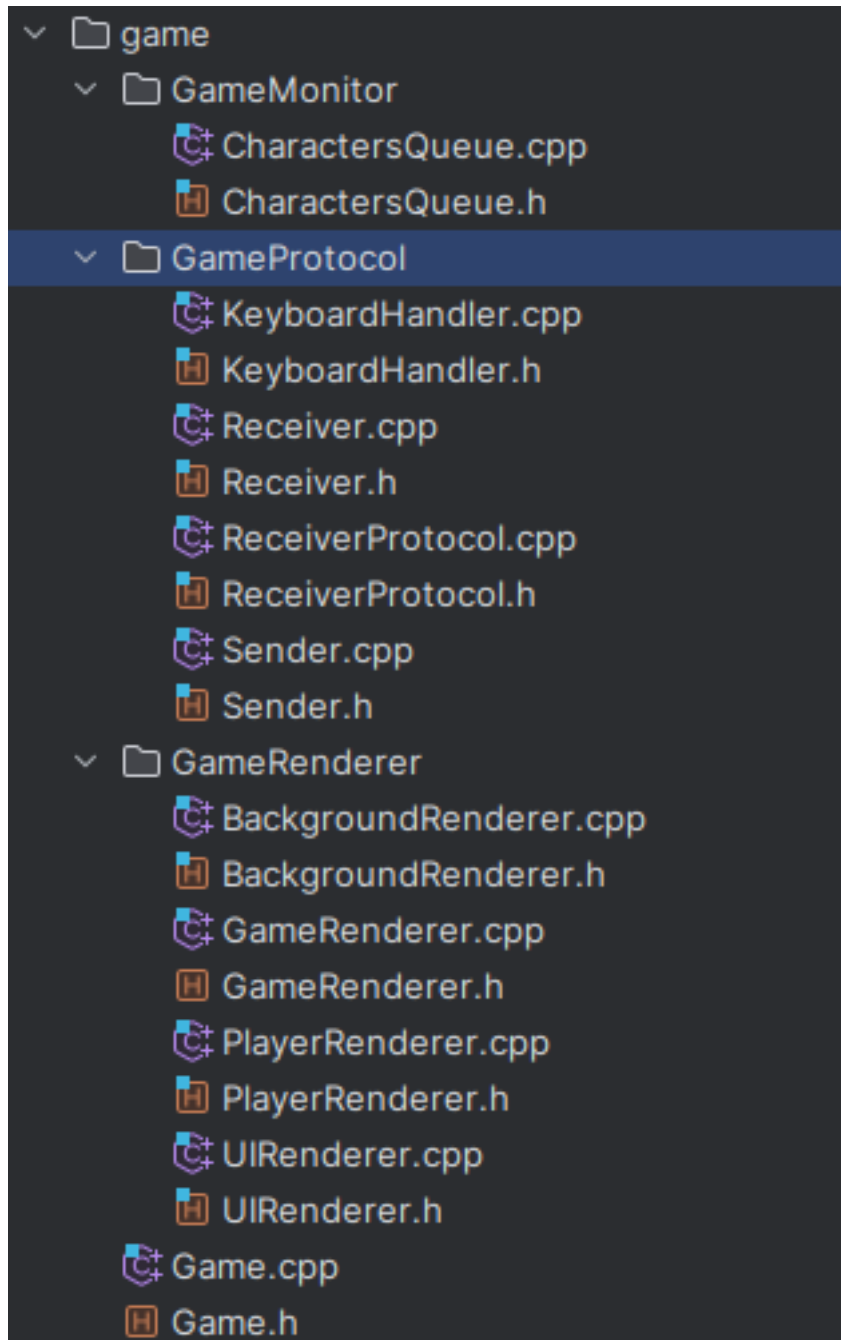
Cuando se ejecuta `./client < hostname > < port >`. Se carga el main, que inmediatamente carga el Client. Este crea un programa de Qt que va a ser el lobby y le agrega el ClientProtocol.

### 1.3 Qt Lobby (Menu)



Aquí está el Main Room, el lobby del game y los dialogs para crear/unirse a las partidas. Cuando se inicia una partida se cierra el programa de QT y se inicia el Game.

## 1.4 Game

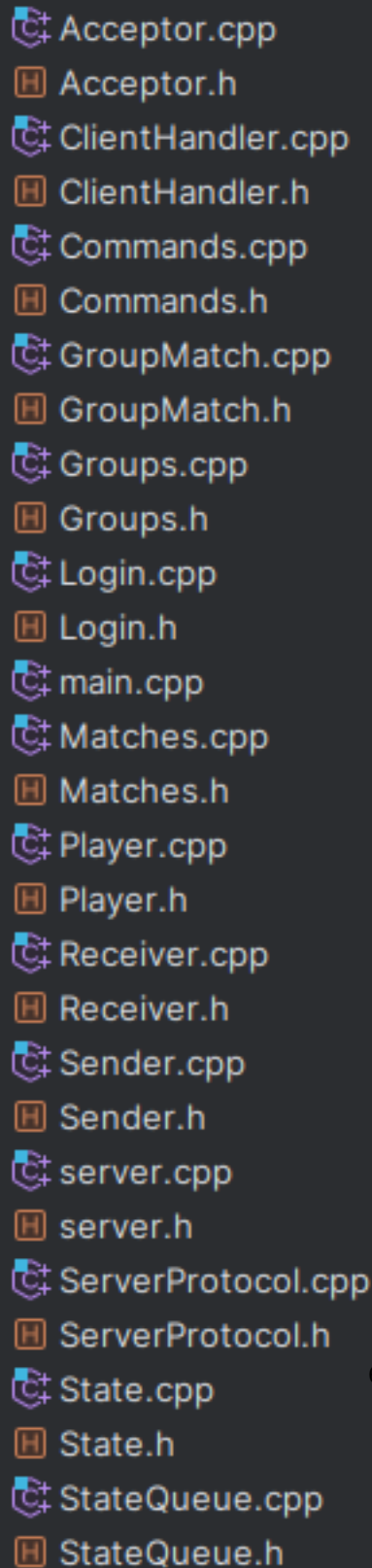


El Game esta compuesto del Game Monitor, Game Protocol, Game Renderer.

- El game Monitor se ocupa de que no haya un race condition entre el game protocol y el game renderer. Sobre todo cuando se cargan los characters por primera vez.
- El Game Protocol tiene los thread Receiver y Sender (que usa el KeyboardHandler, que recibe los inputs del teclado) para comunicarse con el servidor y actualizar la pantalla.
- El Game Renderer se encarga de renderizar el juego en la pantalla. Consta del Background Renderer, PlayerRender y Ui Renderer.
  - El Background Renderer carga el background y le da el efecto de parallax. Y crea el efecto de dejar atrás a los characters cuándo se mueve el player.
  - El Player Renderer renderiza a los players que se encuentran en pantalla, identificando su estado.
  - El UI renderer carga el UI
  - El GameRenderer es un thread que itera sobre cada uno de estos, de manera indefinida (hasta que se acaba el juego)

## 2 Servidor

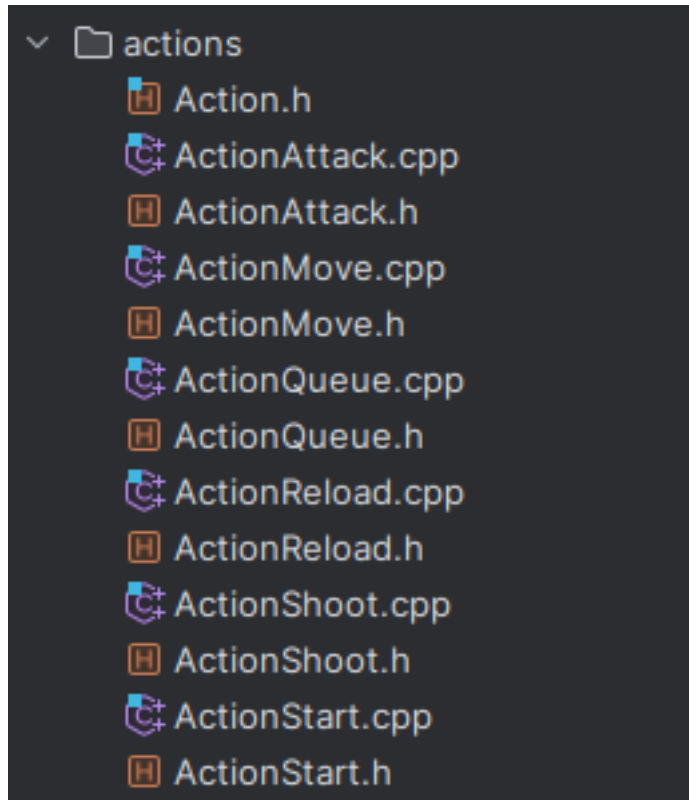
### 2.1 Estructura



- Acceptor.cpp
- Acceptor.h
- ClientHandler.cpp
- ClientHandler.h
- Commands.cpp
- Commands.h
- GroupMatch.cpp
- GroupMatch.h
- Groups.cpp
- Groups.h
- Login.cpp
- Login.h
- main.cpp
- Matches.cpp
- Matches.h
- Player.cpp
- Player.h
- Receiver.cpp
- Receiver.h
- Sender.cpp
- Sender.h
- server.cpp
- server.h
- ServerProtocol.cpp
- ServerProtocol.h
- State.cpp
- State.h
- StateQueue.cpp
- StateQueue.h

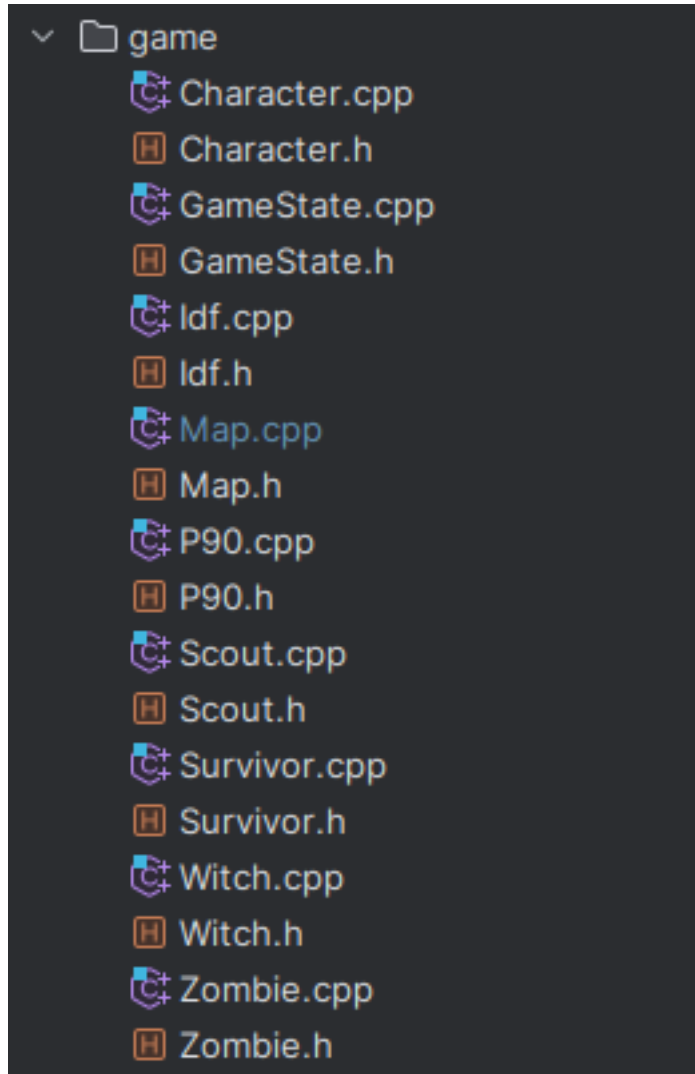
Se crean las partidas y se lanzan los 4 hilos de Sender, Receiver, Group Match y del Acceptor.

## 2.2 Actions



Son las acciones que puede realizar el jugador. Estas se reciben por el Server Protocol.

## 2.3 Game



El Game incluye al GameState el cual tiene un mapa que contiene a todos los personajes (Character.cpp).