**Software Requirements Specification**

**“Strange Multiplayer Game”**

**version 1.0**

**Fabio Bove** | 216219@studenti.unimore.it

**1.0. Introduction**

## **1.1. Purpose**

## The purpose of this document is to present a detailed description of the system, purpose and features, interfaces of the “**Strange Multiplayer Game**”.

What the system will do, the constraints under which it must operate and how the system will react to external stimuli.

**1.2. Scope of Project**

This software will be a **Strange Multiplayer Game**, where people around the world can connect and challenge their opponents in a nonsense fight.

Each ***player*** connects to the main ***server*** and is asked to provide an unique ***username*** to join the game. Players are identified by their ***ids***, stored in a queue where two players close to each other are considered ***opponents*** for a given game.

Whenever a player leaves the game is considered closed and its opponent who’s still connected gets reinserted into the queue so he can start a new game, maintaining its original username.

For a game: A long string it’s given to two players, and they need to find the exact number of duplicates for a given character. The first player who finds out the correct number wins, and can continue to play, the previous **match** is closed, and a new one is started with the first opponent available in the queue.

Scores of the games are recorded on a “.csv” file, locally on the server. In this way we can keep a sort of ranking for players.

## **1.3. Glossary**

| **Term** | **Definition** |
| --- | --- |
| Player | Person connected to the server. |
| csv file | Collection of records that stores information of users' score in the game - It allows to rank players. |
| Username | Unique String that identifies a user or player |
| Id | Int number that identifies a player and its position on the queue |
| opponents | Two players that challenge each other in a given game |
| match | Consist of one round, each player needs to identify as fast as he can the occurrence of a given char in a string. |

## **1.4. References**

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.