CSD2161 Computer Networks Assignment 4 Design Document

|  |  |  |
| --- | --- | --- |
| Name | Digipen Student ID | SIT Student ID |
| Poh Jing Seng | Jingseng.poh | 2301363 |
| Gwee Boon Xuen Sean | g.boonxuensean | 2301326 |

Table of Contents

[1. Game Design 3](#_Toc194699483)

[1.1 Introduction 3](#_Toc194699484)

[1.2 How to play 3](#_Toc194699485)

[1.2.1 Controls 3](#_Toc194699486)

[1.2.2 Game flow 3](#_Toc194699487)

[2. Software architecture 4](#_Toc194699488)

[2.1 Communication commands 4](#_Toc194699489)

[2.1.1 CONN\_REQUEST 4](#_Toc194699490)

[2.1.2 CONN\_ACCEPTED 4](#_Toc194699491)

[2.1.3 CONN\_REJECTED 4](#_Toc194699492)

[2.1.4 ACK\_CONN\_REQUEST 5](#_Toc194699493)

[2.1.5 REQ\_START\_GAME 5](#_Toc194699494)

[2.1.6 START\_GAME 5](#_Toc194699495)

[2.1.7 ACK\_START\_GAME 5](#_Toc194699496)

[2.1.8 SELF\_SPACESHIP 6](#_Toc194699497)

[2.1.9 NEW\_BULLET 6](#_Toc194699498)

[2.1.10 ACK\_NEW\_BULLET 6](#_Toc194699499)

[2.1.11 ALL\_ENTITIES 7](#_Toc194699500)

[2.1.12 END\_GAME 8](#_Toc194699501)

[2.1.13 ACK\_END\_GAME 8](#_Toc194699502)

[2.1.14 KEEP\_ALIVE 8](#_Toc194699503)

[2.2 Communication flow 9](#_Toc194699504)

[3 Implementation block diagram 10](#_Toc194699505)

[3.1 Server block diagram 10](#_Toc194699506)

[3.2 Client block diagram 11](#_Toc194699507)

[4. Individual contribution 11](#_Toc194699508)

# Game Design

## 1.1 Introduction

This is a multiplayer capable player vs player (PVP) asteroids game where players compete to achieve the best possible score by destroying asteroids. Players have 3 lives and will be unable to continue playing when they lose all their lives crashing into asteroids. The game ends when the game time of 60 seconds is up, or all players are dead. This game is also available in single player if only one player connects to the server.

## How to play

### 1.2.1 Controls

W – Accelerate forwards

S – Accelerate backwards

A – Rotate left

D – Rotate right

Space - Shoot

### 1.2.2 Game flow

Upon starting the game and entering the player’s name, the player will be brought to the intermediate page where it is waiting for other players to connect. Any player, at any time, can use ENTER on the keyboard to start the game for all players. Upon the destruction of an asteroid using a bullet the player fired, the player’s score will be incremented and if the asteroid is big enough, it will split into 2 smaller asteroids. Just like the larger asteroids, small asteroids kill players on collisions, grants the player 1 point on being shot down, and gets destroyed when collided with. When the game ends, the game winner is declared along with the top 5 all time best scores on the leaderboard – complete with player name, score, and date and time achieved.

# Software architecture

## 2.1 Communication commands

### 2.1.1 CONN\_REQUEST

A computer screen shot of a green dot

AI-generated content may be incorrect.

Connection is initiated by client sending this payload to the server.

### 2.1.2 CONN\_ACCEPTED

A screen shot of a computer program

AI-generated content may be incorrect.

Connection is accepted by server if the server can accept more players. Server provides the player’s spaceship spawn position, and spawn rotation.

### 2.1.3 CONN\_REJECTED

A group of colorful circles

AI-generated content may be incorrect.

Connection is rejected by server if the server cannot accept any more players

### 2.1.4 ACK\_CONN\_REQUEST

A screen shot of a computer

AI-generated content may be incorrect.

Client must send acknowledgement to server upon receiving if connection is accepted or rejected.

### 2.1.5 REQ\_START\_GAME

A group of colorful circles

AI-generated content may be incorrect.

Clients can request the server to start the game.

### 2.1.6 START\_GAME

A screen shot of a computer program

AI-generated content may be incorrect.

Server tells all clients to start game and sends the names and ID of all connected players.

### 2.1.7 ACK\_START\_GAME

A computer screen shot of a computer code

AI-generated content may be incorrect.

To facilitate reliable communication, clients must acknowledge START\_GAME command.

### 2.1.8 SELF\_SPACESHIP

A screen shot of a computer program

AI-generated content may be incorrect.

Client sends its own spaceship’s movement vectors and current rotation to server.

### 2.1.9 NEW\_BULLET

A screenshot of a computer program

AI-generated content may be incorrect.

Client sends new bullet fired to server for synchronization.

### 2.1.10 ACK\_NEW\_BULLET

A screenshot of a computer

AI-generated content may be incorrect.

Server needs to acknowledge new bullet for synchronization.

### 2.1.11 ALL\_ENTITIES

A screen shot of a computer program

AI-generated content may be incorrect.

Server updates all entities’ positions and sends data essential to rendering to client.

### 2.1.12 END\_GAME

A computer screen shot of text

AI-generated content may be incorrect.

Server sends END\_GAME to client to inform client of game status and share current leaderboard.

### 2.1.13 ACK\_END\_GAME

A green dot on a black background

AI-generated content may be incorrect.

Client needs to acknowledge receipt of END\_GAME command.

### 2.1.14 KEEP\_ALIVE

A green dot on a black background

AI-generated content may be incorrect.

Client must send this to server at least once every 15 seconds, else, the server will consider the client disconnected.

## 2.2 Communication flow

A diagram of a server

AI-generated content may be incorrect.

# 3 Implementation block diagram

## 3.1 Server block diagram

A diagram of a company

AI-generated content may be incorrect.

## Client block diagram

A diagram of a software company

AI-generated content may be incorrect.

# 4. Individual contribution

Poh Jing Seng (jingseng.poh) 50%

Gwee Boon Xuen Sean (g.boonxuensean) 50%