# Distributed-System-Assignment1

Distributed Player Status System (DPSS) using Java RMI

Submitted By : Vaibhav Malhotra - 40079373

## Development tools

All code is written in IntelliJ IDE, Java JDK version 8.

## Build and Run

**IntelliJ:**

- Open the project in folder DPSS
- Run: AmericaGameServer, EuropeGameServer, AsiaGameServer.
- Run: PlayerClient (to launch a player window)
- Run:  AdminClient (to launch a Admin window)
- To run multiple clients change the configuration to “Allow parallel run”.

**Command Line:**

- Move to *DPSS* directoy: **cd DPSS**.
- Create a new folder named *dist* in the current folder : **mkdir dist**
- Compile the code (outputting into dist folder): **javac -d dist src/**/*.java**
- Move to *dist* folder: **cd dist**
- Run all 3 servers using following commands (will have to open different terminals):
  - java -cp . GameServers.AmericaServer.AmericaGameServer
  - java -cp . GameServers.EuropeServer.EuropeGameServer
  - java -cp . GameServers.AsiaServer.AsiaGameServer

- Run the clients (will have to open in different terminals):
  - java -cp . Client.PlayerClient
  - java -cp . Client.AdminClient

Vaibhav Malhotra – 40079373

# Concepts implemented

## 1. RMI

Remote Method Invocation has been used to invoke client requests on the servers based on user's IP address.

## 2. UDP

For below communication between server UDP is used:

- getPlayerStatus() – When admin requests this method on a server, that server sends UDP request to other two servers to get the player info.
- createPlayer() – When a user tries to create a new player on a server, that server sends a UDP request to other servers to check if Username already exists.

## 3. Multi-threading

- All servers run on their individual thread
- All UDP requests are sent on a new thread
- All client requests are sent on a new thread

## 4. Locks and HashTable

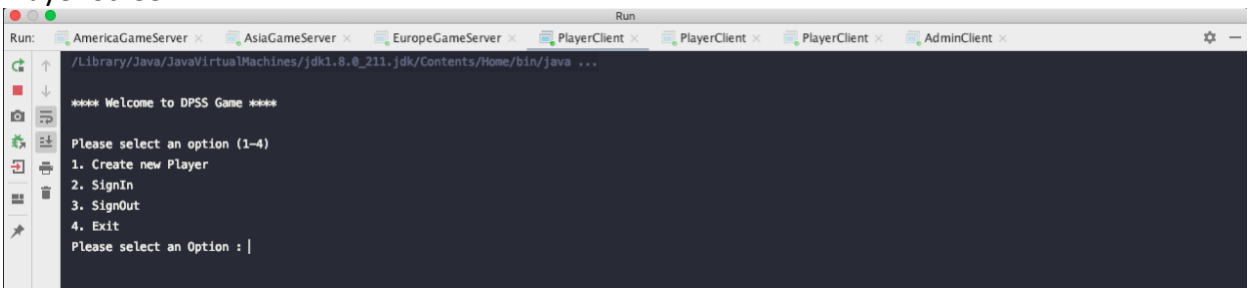
- Player data on server are stored in a Hashtables. Hashtables are thread-safe and promote concurrency.
- Lock (ReentrantLock) is used for proper synchronization to allow multiple users to perform operations for the same or different accounts at the same time.

Vaibhav Malhotra – 40079373

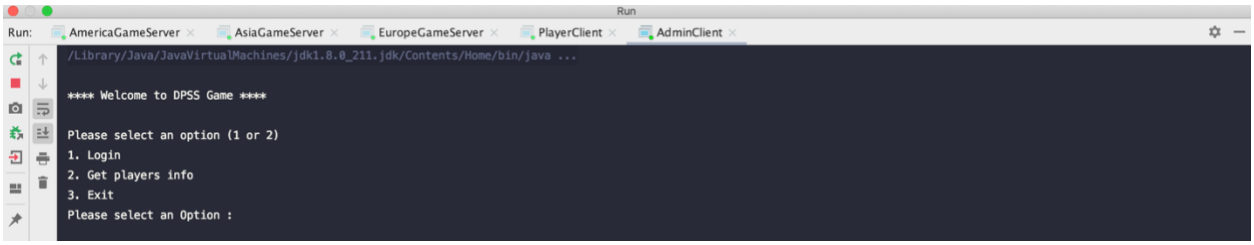
# Testing

**Intro screens:**

Player Screen:

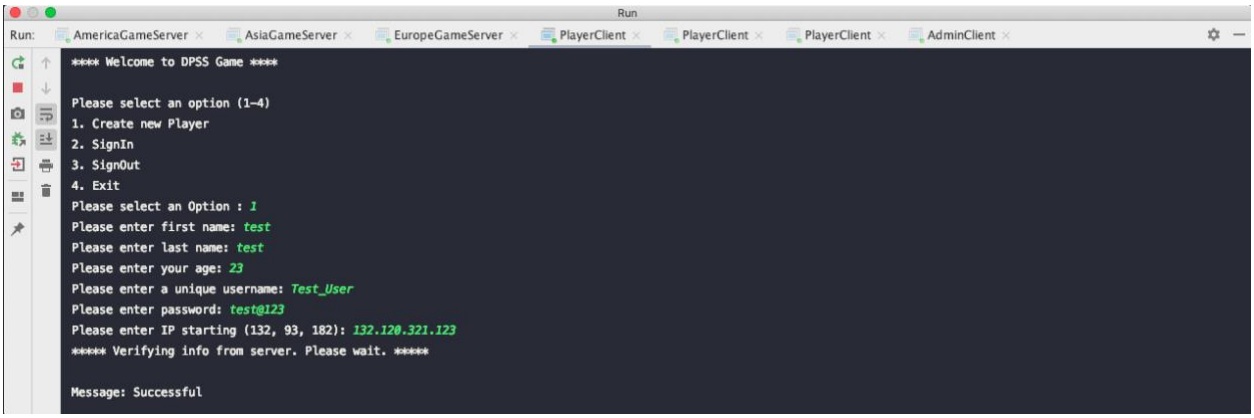



Admin Screen:

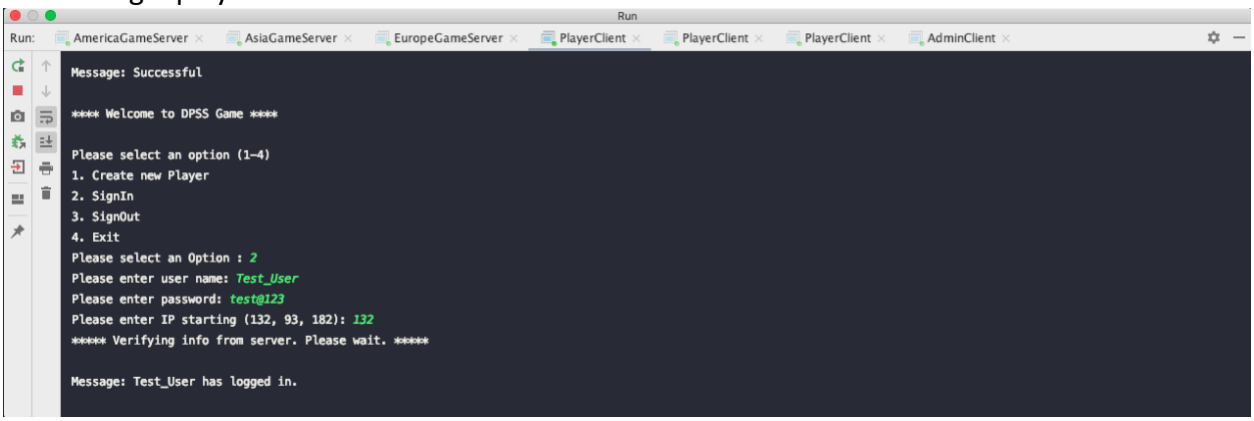



Player testing:

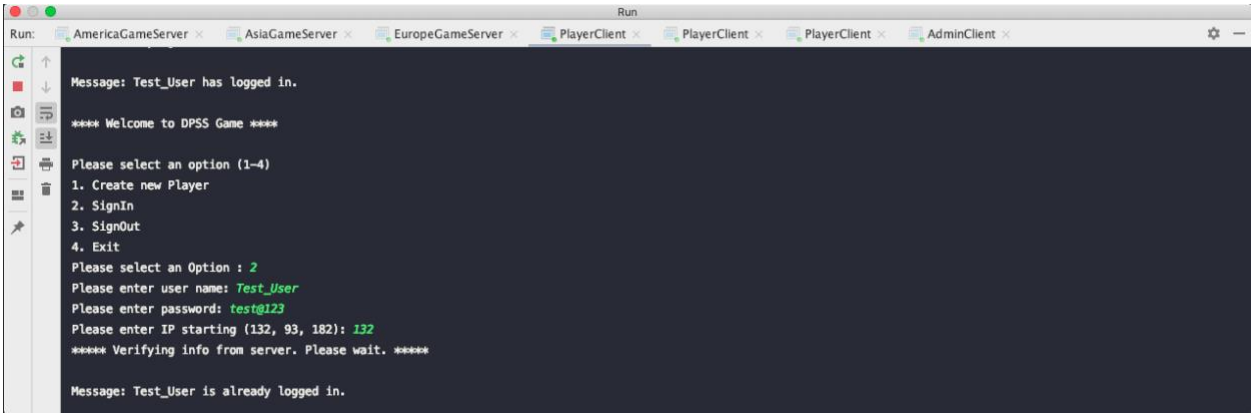
1. Create player



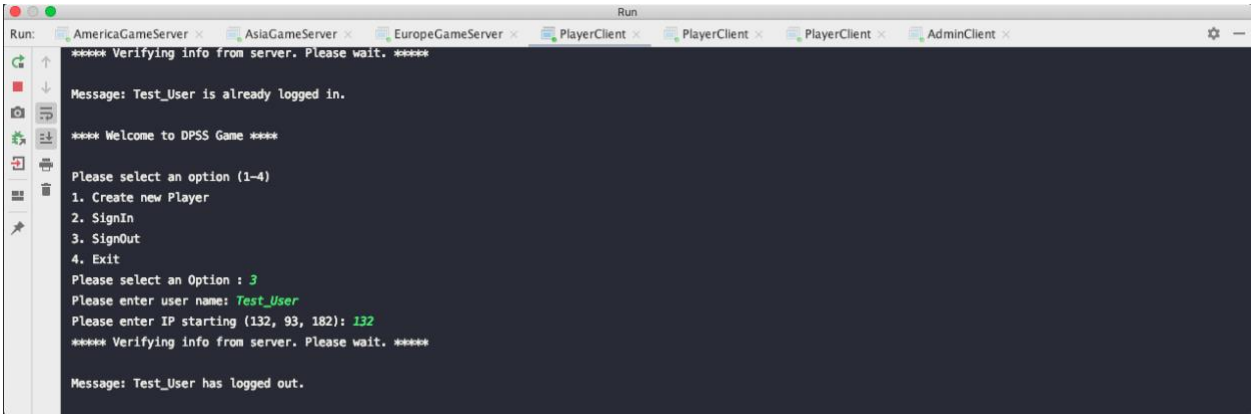
2. Login player




3. Login player who is already logged in.

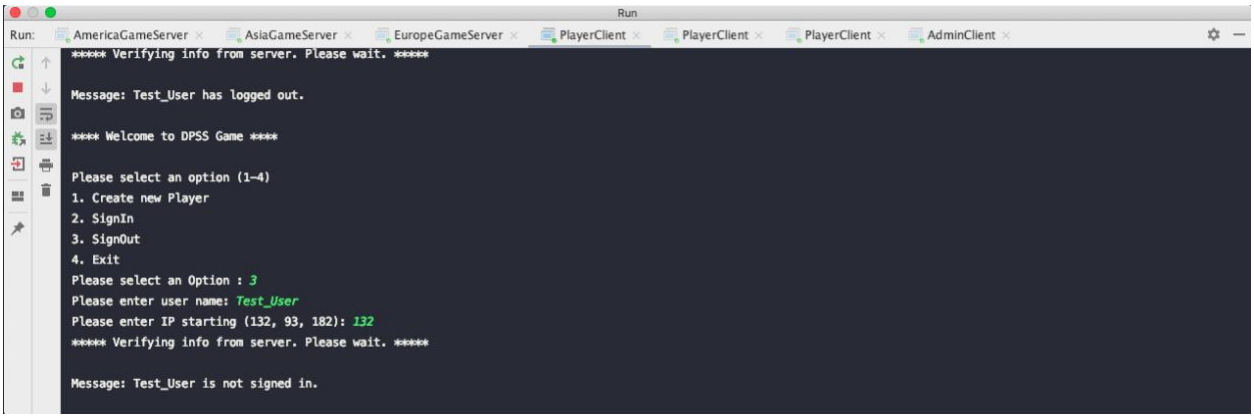



4. Logout player

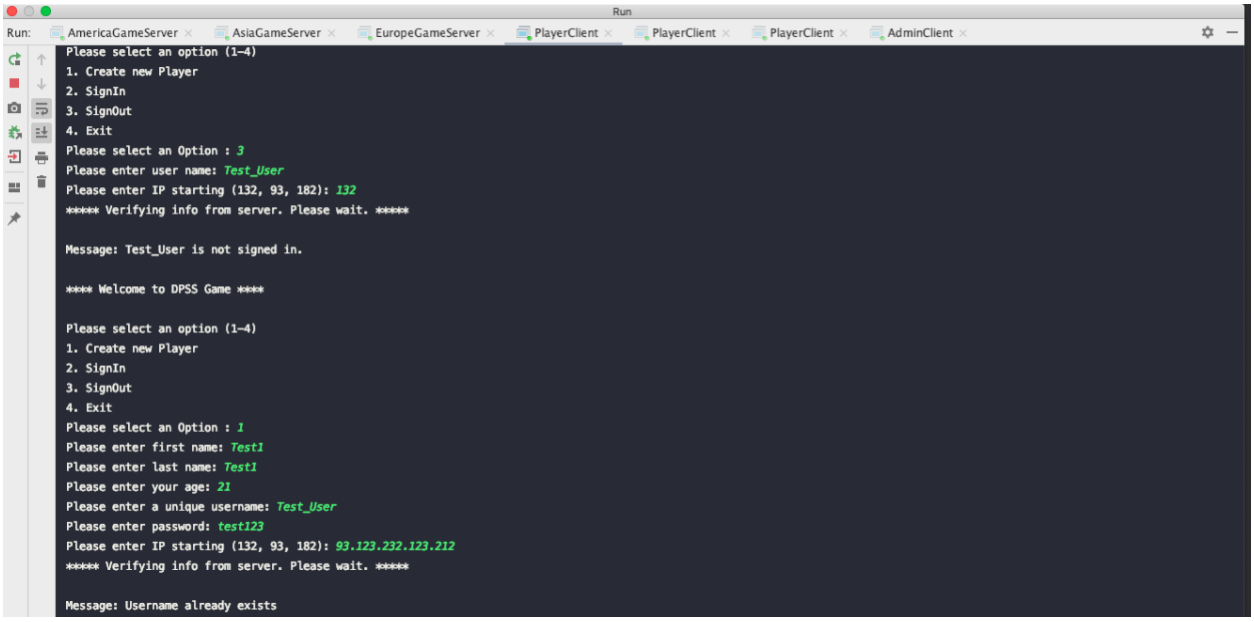


Vaibhav Malhotra – 40079373

5.Logout player who is not logged in



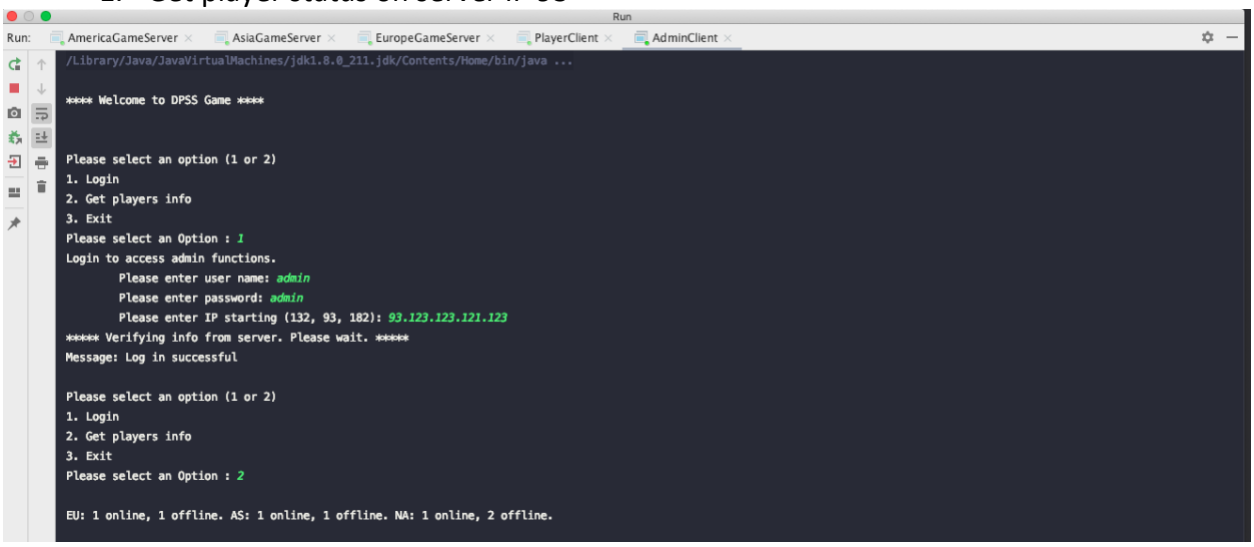

6. Try to create a user (Already on another server)



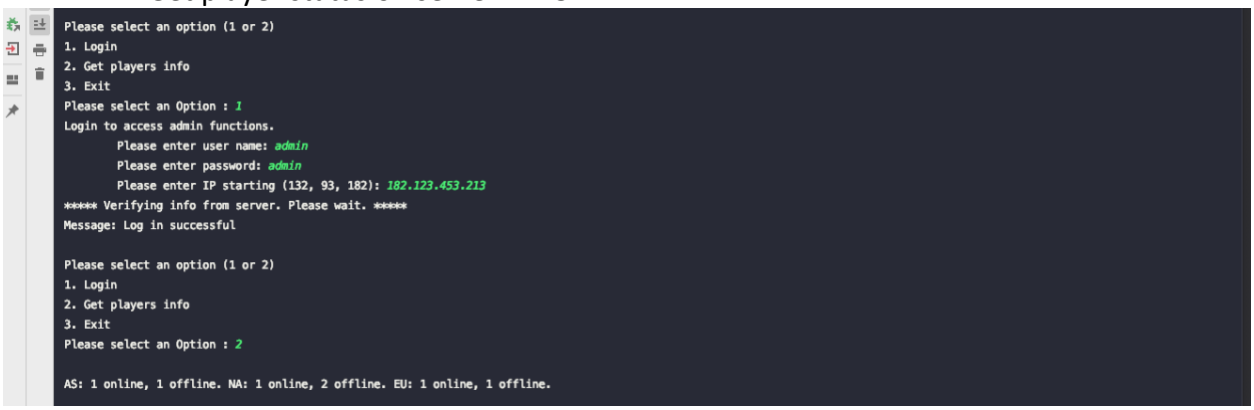
Vaibhav Malhotra – 40079373

Admin:

1. Get player status on server IP 93




2. Get player status on server IP 182




3. Invalid IP or credentials