**INCEPTION DOCUMENT**

**Multiplayer Tic-Tac-Toe Game**

**By**

**Aditya Taneja ( 104604869 )**

**Vatsal Jani ( 104672987 )**

**Prabhleen Singh ( 104683866 )**

# **Table of Contents**

1. **Vision and business case**
2. **Use Case Diagram**
3. **Supplementary Specifications**
4. **Glossary**
5. **Risk List and Management Plan**
6. **Prototypes and proof of concepts**
7. **Iteration Plan**
8. **Phase Plan and software development plan**
9. **Development Case**
10. **References**

# **Vision and business case**

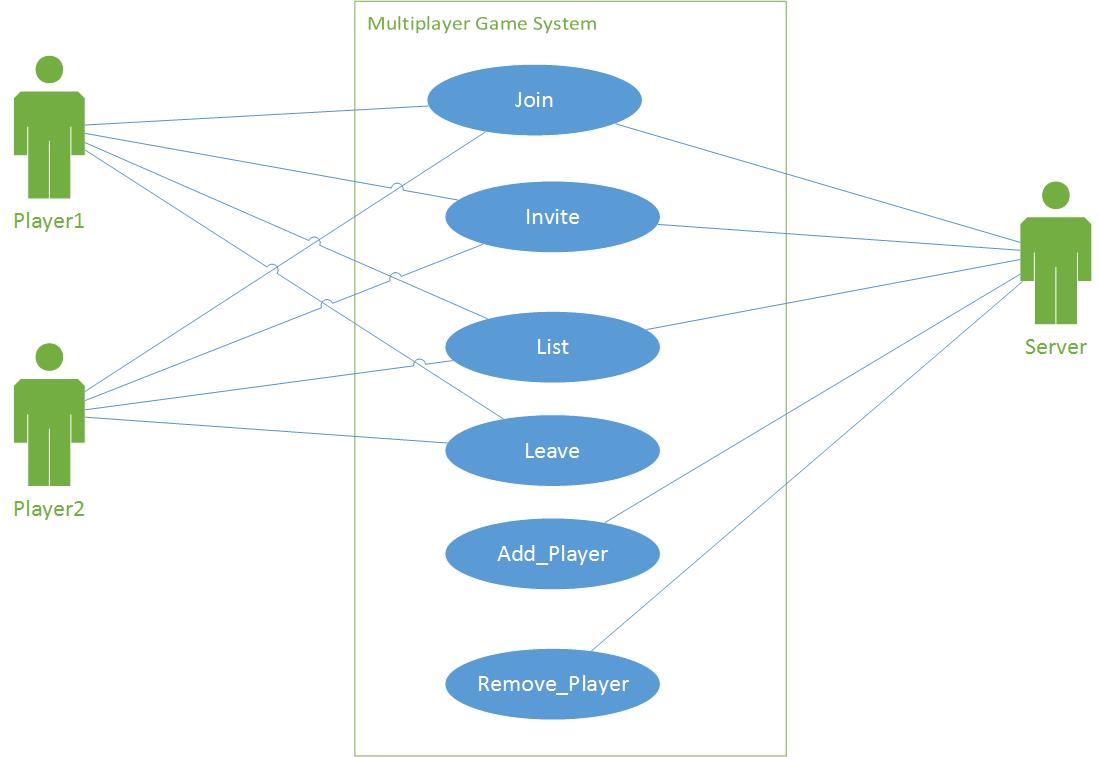
This project focuses upon development of a multiplayer distributed gaming application based on a client-server architecture. The server side shall be accepting multiple client processes. The client side will be participating through a separate machine and implemented in C language. It will be usable via an Android device. We aim at developing the server side in C language and an ad-hoc based connection between client and server using Sockets.

The multiplayer functionality of the Tic-tac-toe application would enable multiple clients/users to connect to the server in real-time and

Probable **constraints** in achieving the above stated goals are as follows –

1. Learning Android Development to develop the client side.
2. Timely acquiring the skills to implement Socket to enable the conglomeration of the client and the server side.
3. Implementing the client and server sides together to form a client-server architecture.
4. Removal of various probable bugs and errors to assure software quality within the initial timebox.

# **Use Case Diagram**



# **Supplementary Specifications**

Certain non-functional requirements of the system include -

**Reliability –** The application is supposed to have a reliable client-server architecture implemented by Socket.

**Quality –** The software application will go through intense debugging in order to ensure premium software quality.

**Robustness –** The gaming application will accommodate robust gameplay. As the system is easy to handle and navigates in the most expected way with no delays. In that case the system program reacts accordingly and transverses quickly between its states.

**Modifiability –** The open-source gaming application will be modifiable providing a broad scope of improvements.

**Availability** -If the internet service gets disrupted while sending information to the server, the information can be sent again for verification.

**Platform Independent** – The Tic Tac Toe multiplayer game will be built to be able to run on Android devices apart from Windows, Mac OS, Linux Operating Systems.

### Extensible and Maintainable - The gaming application will be extensible in order to enable ease of customization in the future upgrades. Additional functionalities such as game tournaments can be added making the players compete in a hierarchical manner.

# **Glossary**

A list and dictionary is extremely important in order to guarantee that everyone involved in the project will have a clear understanding of some of the key words.

**join {name}** – Server checks whether player by this name already exists in the hash map, if not then the player is added.

**invite {playername}** – The server searches for a player by this name inside the hash map. If it succeeds, it sends the IP of this player to the requesting player.

**list** – The server iterates over the hash map, creates a list of all players and sends it to the client.

**leave** – Server closes the connection with this client and removes the player from the hash map.

**Add Player** – The server can add a player in case a player tries to re-connect or if a player needs to be added to the game.

**Remove Player** – The server can remove a player in case a player is not responsive or disconnects from the game for a long time.

**GitHub –** Remote repository based on Git from source code and version control.

**DevOps** - It aims at establishing a culture and environment where building, testing, and releasing software can happen rapidly, frequently, and more reliably.

# **Risk List and Management Plan**

The risk list bellow contain the identified risks based on early investigation by the team. Managing the risks tend to maximize our changes to have a successful project so that a plan on how to manage them can also be built.

Risks are categorized as below.

## **Development environment and Management**

* Team may not be familiar or ready to manage the project with the project management tools available.
* Project will be developed in visual studio and android studio.
  + Team should gain the necessary knowledge in time.
  + Gathering enough help resources and technical support to meet the developer’s needs can be challenging.
* Potential risks regarding the source and version control process.
* Developers may not have minimum necessary experience to manage their work in the shared Git repository. Versioning and Branching may be new concepts to them.

**Technical**

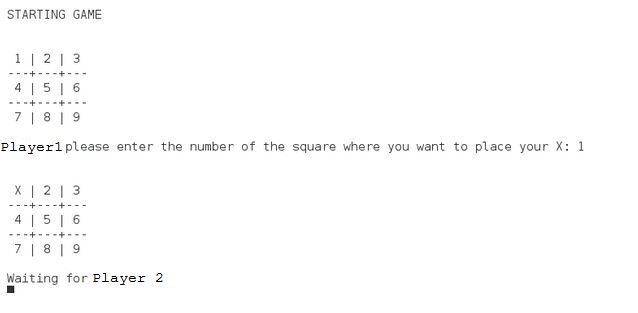
* Team has a basic knowledge of Android Programming.
* Team may not have the minimum experience to use DevOps in order to have a continuous and automatic system build and deployment.

**Other Potential Risks**

* Along the very first team meeting, the risks involved were discussed in the discussion. Most of them are directly related to the fact that the schedule is very tight and most of technologies and development environment are completely new to the team of developers. In order to minimize these risks all member of the team focused on learning and getting the initial development set up.
* Environment configuration for initial development and system deployment was configured. Once done it will be replicated to all of the members in order to make use of all, the developers should have some personal experience in early stages of the project.

# **Prototypes and proof of concepts**

The basic layout of the game application is as below -

****

Initial Design has been made in order to have a preview about the general interface of the application.

# **Iteration Plan**

## **Phase 1 – Due February 24th**

* Initial analysis along with the use-case diagram.
* Definition of technologies and development platform
* Elaboration of prototype and proof of concept
* Inception Plan
* Basic development environment configured and deployed

## **Phase 2 – Due March 20th**

* Front end development based on the proof of concept.
* Backend development
  + Basic communication with front end and back end i.e. the client(s) and server.
* Elaboration of prototype and proof of concept.
* Final report start /Project presentation start

## **Phase 3 – Due March 29th**

* Project presentation finalization.
* Full prototype working with basic functions.
* Final report in its initial version done.

# **Phase Plan and software development plan**

The team is composed of the following developers –

* **Aditya Taneja ( 104604869 )**
* **Vatsal Jani ( 104672987 )**
* **Prabhleen Singh ( 104683866 )**

Based on the technologies defined to develop the Project, below is the list of tools that will be used in the development of the project -

* DevOps for tracking and CI
* GitHub for SVC
* Visual Studio 2013
* Android Studio

All development will be equality shared with everyone being part of team and contributing to the whole development process.

# **Development Case**

Weekly project status will report progress of the system development. Team members will share Minutes of meetings through postings in the Group 21 Discussion Board.

The phases and milestones will be as follows:

**Inception phase**

* + System conception with clear definition of its boundaries.
  + Definition of development tools, technologies and deployment environment.

**Elaboration phase**

* + Sufficient working functionalities for a prototype usage level
  + Project presentation

**Revision phase**

* + Fully working prototype with proper suggestions of the public services
  + Project Report publication

# **References**

1. "Github - Adityataneja94/Project1". *Github.com*. N.p., 2017. Web. 25 Feb. 2017.
2. "Download Android Studio And SDK Tools | Android Studio". *Developer.android.com*. N.p., 2017. Web. 25 Feb. 2017.
3. "Visual Studio | Developer Tools And Services | Microsoft IDE". *Visual Studio*. N.p., 2017. Web. 25 Feb. 2017.