Interactive Street Based RISK Game

CS 440 – Group 4
Luqmaan Baiyat
Joshua Herman
Hamza Shahid
Patrick Gundry

Table of Contents *

I. Project Description

1. Project Overview

The project is an interactive city based RISK game that takes place on a virtual battlefield that is constructed from real life city data provided by the Google Maps API. The battlefield is comprised of intersections in a district of the city. Each intersection is controlled by a player who stations military units there. Each match in the game requires multiple players to play against each other and/or CPUs. Players can select which virtual battlefield they would like to play on or choose randomly. Players in the game are meant to attack other intersections to take over control of intersections from other players. The primary goal of each player is to control all the intersections in a match, which results in them being the winner. If a player loses control of all their intersections, they are eliminated from the game.

2. The Purpose of the Project

The project is being undertaken in order to create a game that would be more dynamic than traditional board games where the locations on the board remain the same for every playthrough. There is a finite amount of strategy and ways to play a game of RISK on the normal static board players are used to. It is played with the same territories through every game and the same points based system for capturing those territories. In our dynamic game the players will be faced with new and creative challenges with every play through, and consistently tested in their skills with a map they have never seen before. In this way, the players will enjoy playing the game for a longer amount of time as every game will be different from the last.

2a The User Business or Background of the Project Effort

The client for this product is a video game publisher who has in the past released games for personal computers and mobile devices. They are looking to publish an entertaining and dynamic board game. They do not have much experience in strategy, war, and territory games, but they are great in competitive multiplayer games. They are looking to make a static board game into a dynamic competitive multiplayer game, where each game is unique from the next. Given these specifications by the publisher an interactive and dynamic street based RISK game would be exactly what they are looking to publish.

2b Goals of the Project

The goal of the project from the point of the client is to get more players interested in this genre of game to explore the possibilities of publishing similar games in the future. They are looking to bring a revival and new look to many of the board games that were created decades ago. Many of these board games have been forgotten by the people who once enjoyed playing them, and the want to bring these people and others back into the game. They also want to put out a game that is always changing and that has different mechanics to these games created in the past. Also these board games were often just played in your house with family members and friends, but as people get older it is difficult to get together, so this allows them to play against each other online. This also allows the player to play against anyone around the world, so rather than it being a game for just the home, it is a game that can be played anywhere against anyone. Through our product we will have all of these features stated in order to create a great product that this publisher is looking for.

2c Measurement

The goal can be considered met when concurrent user activity and persistent play of the game reaches a certain threshold within a few months after the game's release. In terms of analytics, our goal is to have 1,000 people sign up for beta testing where we will get consistent feedback how to improve our game upon launch. On the first week we are looking to have at least 100,000 downloads which is standard for our client, and 35,000 people that play more than one day in that week. After that in the month we want at least 200,000 downloads, 100,00 people that played more than 5 days in that month, and 30,000 daily users. By 3 months we want 350,000 downloads, 175,000 people that played more than 5 times in that 3rd month, and 70,000 daily users. We also want to keep our rating in stores above 4.5/5 stars at all times. In order to keep these number high we want to receive consistent community feedback and make changes to the game based on this feedback. We will also keep track of statistics on whether or not people played RISK before so we have a better idea of our core users.

3. The Scope of the Work

The software of a city-based RISK game that is for mobile and desktop devices

3a The Current Situation

The client currently is not conducting the work now as they do not have a city-based RISK game or any thing like RISK with which consists of war, strategy, and territory. They have published many other games but nothing of this type, and nothing with other board games, so they are trying to break into this space in their industry.

3b The Context of the Work

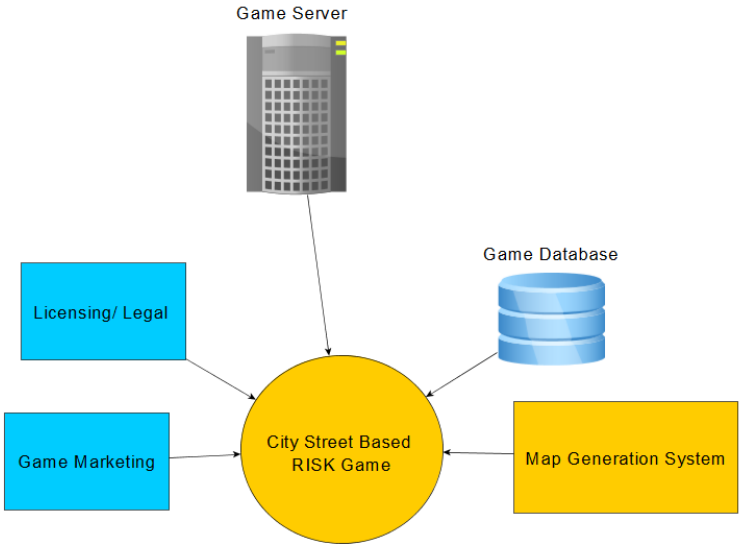


Figure 1.

1. The external entities are the users who will install the game on their device and play it.
2. Items included in the work:
    a. Creating a game map generation system which takes in a city name and gives a game map build around that city.
    b. Maintaining a game server to support a large amount of games happening concurrently.
    c. Maintaining a database of player information
3. Items not included in the work:
    a. Making sure a reliable connection for the game server is established.
    b. Marketing for the game
    c. Licensing issues

3c Work Partitioning

Events	Summary
Pre-full Release Testing (in/out)	Before the official launch of the game to the public, users are going to have the option to apply for beta testers which will give them early access to the product. Through this early access, bugs and any other unusual activity can be reported inorder to make the final product as fluid as possible.
In-game Lobby Map Selection (out)	In order for the game to begin, players are going to have to select which map they would like to play in from the vast amount of cities districts to choose from. Through the use of Google Maps API, a map is going to be downloaded for players to play in.
Daily/Weekly Challenges (out)	Updating game on a daily basis inorder to add daily/ weekly challenges which will reward players with cosmetic upgrades as well in in-game bonuses and power-ups.
Cross Platform Support (out)	In order to allow players in the same game to play on any device, the game is going to allow cross platform play. This cross platform play will even allow players to begin the game and one device and and finish that same game on another device.
Leaderboards (out)	Allow players to see the leaderboards consists of just friends, as well as a worldwide leaderboard consisting of all the players in the world. There would also be leaderboards consisting of just cities which would give players an idea of who is a better player in a particular city. There will also be leaderboards which will distinguish the players with longest winning streak as well as other measurable metrics for the game.

3d Competing Products

In terms of the dynamic nature of this game there are not any games in our category that create and download maps via the Google Maps API and have a different game based on this in every game. There are also not any games that extend RISK or another board game to be played on a dynamic map versus the static board everyone

is used to. There are a good amount of war, strategy, and territory games out there, but they service a different market and do not overlap with our product enough to compete. The games that tend to be similar to us in terms of this are often solo player versus the computer. There is also RISK which can compete if people want to play the original game, and where people can play against people around the world online, but this game is as static as the original and does not have its own website or app so they are not dedicated solely to this game. Our product is still beneficial despite some competing products, because it is a new twist on board games that has never been seen before. It also will be on its own app for desktop and mobile, and will have our full dedication and effort put into it. Nothing exists out in the market that takes the Google Maps API and makes a different game of RISK based on players choice of where they want to play. No other game like this also has a competitive multiplayer and weekly challenges.

4. The Scope of the Product

The purpose of this product is to create a version of the board game RISK. In this adaption of the game, there would be multiple different maps for players to play on. Each map would be comprised of intersections of a district in a city that players choose. Each intersection is going to have atleast two streets touching each other, but in theory could have no maximum limit of streets intersecting

After the successful completion of the product, players would be able to choose from the following three options and the 4th provides what the player can do in game:

1. Play single player
    a. This single player mode is designed specifically for those who prefer playing against the computer. In this mode, the player can choose how many instances of the computer they would like to face.
    b. The player can choose the difficulty of each instance of the computer before a game starts. The difficulties will be measured by how well the computer executes their overall strategy.
    c. The player will receive exclusive single player rewards for each win single player win.
2. Play multiplayer
    a. This multiplayer mode is designed for players to play with their friends as we well as random people from around the world. This is more of a competitive game mode since players of all kinds of levels will be playing here. This vast community of players will get the best competitive

experience possible which will further improve their skills, and would eventually lead them to play tournaments.

   b. For the competitive game mode, there will be a matchmaking ranking system so that players of a similar skill level will play against each other and not someone of a different skill level. Winning more games will increase the player's matchmaking ranking, but losing more games will decrease the player's matchmaking ranking.
   c. In a casual game, there will be a lot emphasis on the skill of the player and will focus more on the player having fun instead of being in a competitive state.

3. Play Tournaments
   a. The multiplayer mode is designed for players who really want to challenge themselves by choosing from a vast array of game modes, all of which will set them apart from others.
   b. The way that the matchmaking would work for this is that the game will keep track of how many wins a player has, and when that player goes to the tournament mode, he/she will be placed in lobbies who have about the same wins as the other.
4. In Game
   a. Reinforcing - At the start of every turn the player is able to reinforce their intersections by placing units at any intersection they currently hold. They first will be able to turn in their intersections cards if they have a correct combination of cards. The game then gives default units per turn plus units from possible intersection cards plus the units earned by chains. The player is then able to put their units at any intersection and end the reinforcing.
   b. Attacking - After reinforcing the player will be able to attack any intersection that is adjacent to them. They can select an intersection and then the adjacent intersection they want to attack if they have more than 1 unit at that intersection. The attacker and defender will then roll dice and the system will compare the rolls to see if the user or defender will lose units. If they defender goes down to 0 units at the territory the attacker takes possession of it can can move any number of units to the intersection that was attacked except 1. The user can then end the attack whenever they want.
   c. Fortifying - After attacking the player will be able to move units from any intersection that is adjacent to another or that is connected to each other in a chain. They are able to move all but 1 unit from the old intersection to

the new one and they are only allowed to fortify once. After fortifying the user can end their turn.

4a Scenario Diagram(s)

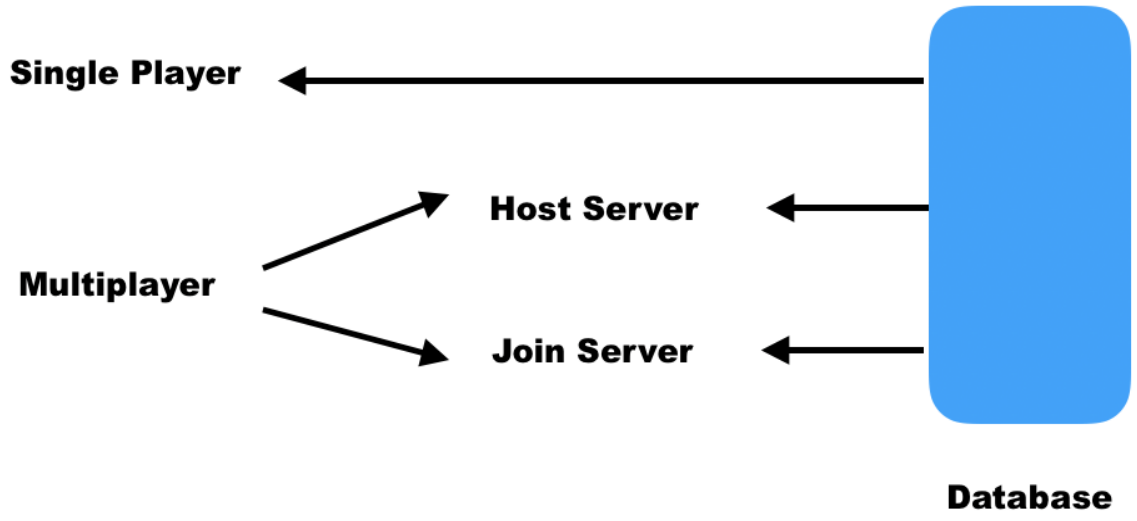


Figure 2.

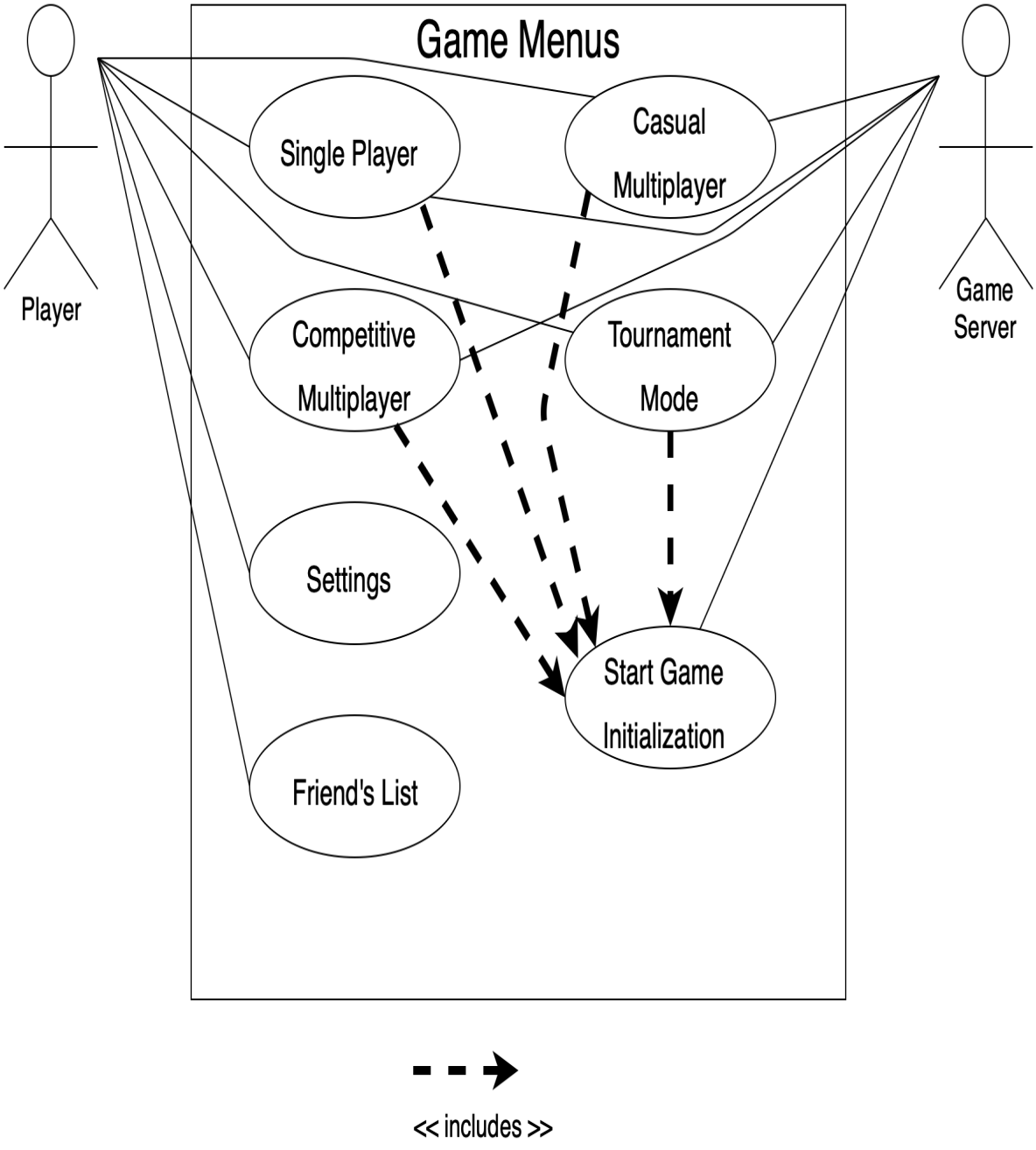

Game Menus
Single Player
Casual
Multiplayer
Competitive
Multiplayer
Tournament
Mode
Settings
Start Game
Initialization
Friend's List
Player
Game
Server
<< includes >>

Figure 3.

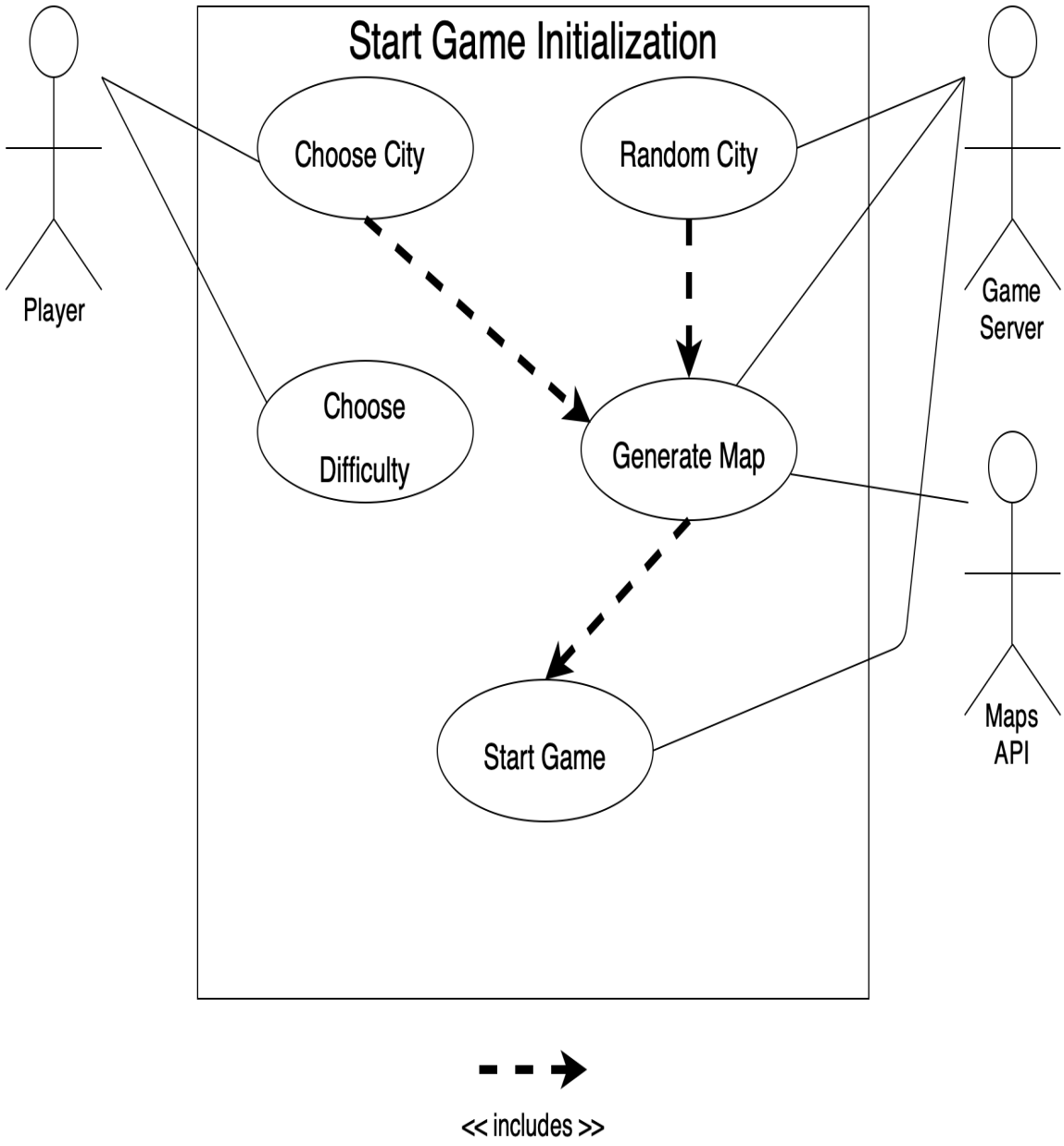

Start Game Initialization
Choose City
Random City
Choose
Difficulty
Generate Map
Start Game
Player
Game
Server
Maps
API
<< includes >>

Figure 4.

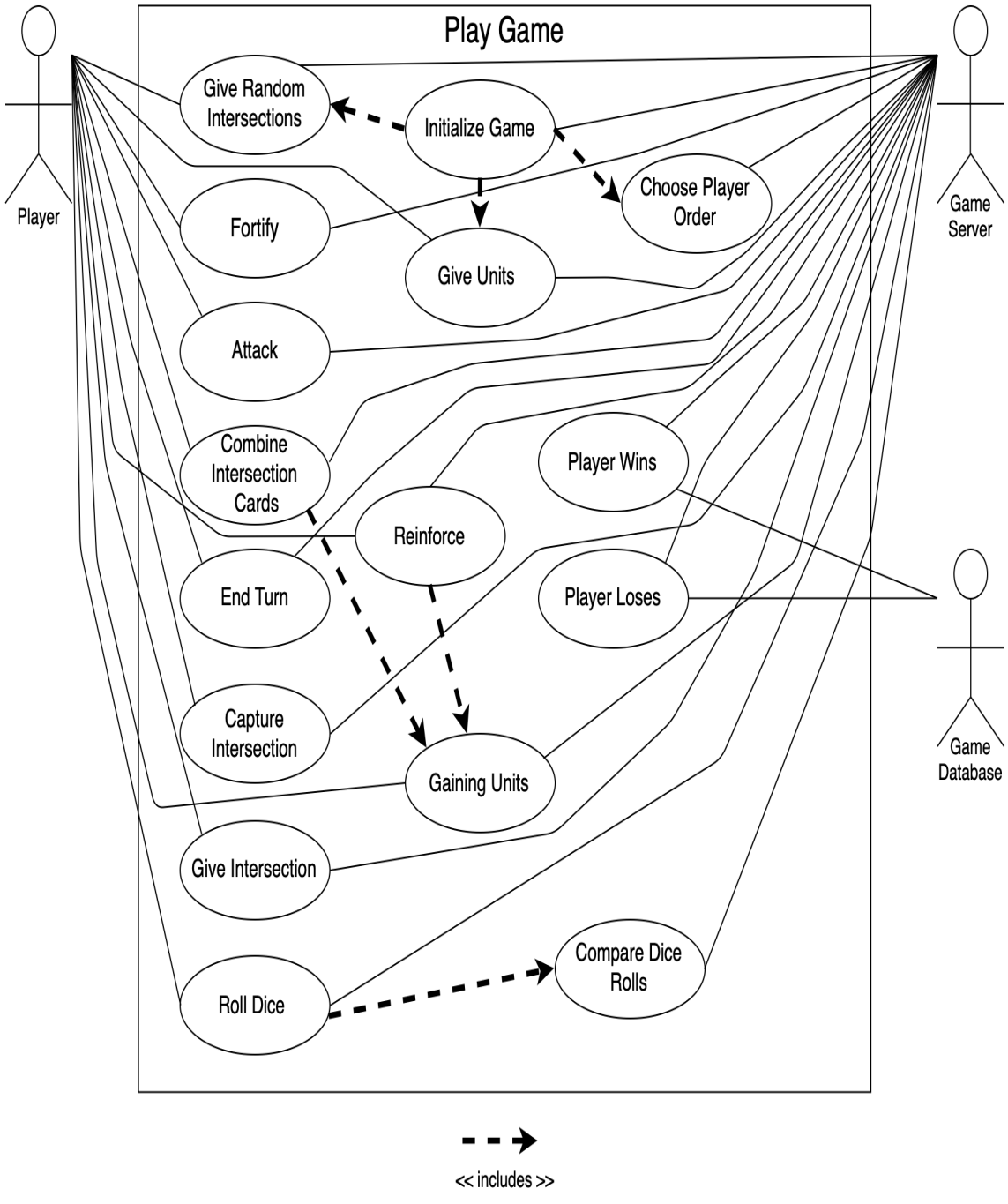

Play Game
Player
Game Server
Game Database
Give Random Intersections
Initialize Game
Choose Player Order
Fortify
Give Units
Attack
Combine Intersection Cards
Player Wins
Reinforce
End Turn
Player Loses
Capture Intersection
Gaining Units
Give Intersection
Compare Dice Rolls
Roll Dice
<< includes >>

Figure 5.

4b Product Scenario List

1. Single Player
2. Multiplayer
3. Tournaments
4. Settings
5. Game Menus
6. Start Game Initialization
7. Play Game

4c Individual Product Scenarios

Single Player	If the player chooses single player, he/she will be playing against the computer. Before starting the game, the player will be asked to select the difficulty level which will consist of Easy, Medium, Hard, Legendary. All these difficulty levels will give the player a great challenge as they will use Artificial Intelligence in such a way that will make the player really put a lot of effort into defeating the player.
Multiplayer	If the player chooses to multiplayer, he/she will be taken to lobby where he/she would be asked if they would like to join a friends lobby or join a random lobby. Since each lobby consists of a host/leader, when the player decides on what lobby to join, the host/leader is going select the map, game duration, team size, and much more. During the play of the games, players can talk to their teammates through message or voice chat for better communication among each other.
Tournaments	If the player chooses to play in tournaments, he/she will be taken to another screen where he/she will be asked to select the kind of tournament which would consist of either single player or multiplayer. For the single player tournaments, players can select from various options consisting of number of maximum players. For multiplayer tournaments the format of selection is going to be the same but would number of players in a team would be limited to 10 maximum. Once a while for both kinds of options, there would be special tournaments which would be intended towards more competitive players.

Settings	If the player chooses settings they will be taken to another screen where they will be able to turn change the volume, language, various account settings like password and email, graphics settings, accessibility among others. For volume they can adjust sliders or mute all or certain sounds. For language they can choose any language we have available to them. For various account settings they can change password and email. For graphics they can adjust to high low or medium graphics. For accessibility the player can choose color blindness settings and text to speech settings.
Games Menus	When the player opens the game they will be shown a menu where they can choose to play a game in single player mode, multiplayer mode casual and competitive, and tournament mode. They can also select friend's list to message or invite friends to a game, and settings to change around the settings in the game. Selecting any of the game mades stated, will trigger a game initialization.
Start Game Initialization	Once a player selects a game mode the game will be put into a game initialization phase. Here the user can select to choose a city in game modes that are casual multiplayer or single player. In single player they will also choose the difficulty of the game mode. In those casual and single player modes the player can also opt to play on a random map. In competitive multiplayer and tournament mode a random map will be selected. After the map is selected the map will be generated by the Maps API. After the map is generated start game will be triggered.

Play Game	Once start game is initialized the users are now able to play the game. At the start of the game the players' order is chosen, they are given units, and they are given random intersections by the game server. At the start of a turn the player is in the reinforcement by their turn. Here they can first turn in any correct combination of intersection cards to receive their units. They will be given a certain number of units based on the card combination turned in. Then they are given units to reinforce their intersections with a default number plus units earned from chains plus units from intersection cards if they turned them in this turn. They can place their units at intersections and once finished end reinforcing. After reinforcing the player moves into the attack part of their turn where they can attack any adjacent intersections to their current as long as they have more than 1 unit at the intersection attacking with. When the attack is triggered the attacker and defender are given dice to roll up to 3 for the attacker and up to 2 for the defender based on the number of units at the intersection. Once the dice is rolled they are compared to each other to see whose is higher or if it is a tie. If they defender goes down to 0 units at the territory the attacker takes possession of it can can move any number of units to the intersection that was attacked except 1. Units are lost from the attacker, the defender, or both depending on the rolls. They can end the attack at any time which will transition them into the fortifying part of the turn. In fortifying the player can choose to move any number of units to another adjacent intersection or any intersection that exists in a chain. They can only do this once in a turn and then they can choose to end their turn.

5. Stakeholders

The stakeholders of this project include individuals who enjoy playing board games and/or competitive games, employees, as well as the publisher of computer and mobile games.

5a The Client

Ultimately, this project is being developed for a publisher of video games that has published numerous games for computer and mobile platforms, but does not have much experience in board games, or games that are war, strategy, and territory games.

There are currently not a large amount of similar games on the market that are as dynamic as this game is intended to be. Because of the high potential this game carries, it is very important that as many crucial features are implemented for the sake of the customers, as well as the stakeholders.

5b The Customer

The customer of this product will be individuals who enjoy to play board games and/ or competitive games, and download our game one or many platforms. This is generally geared to people who enjoyed playing RISK growing up, or others who like war, strategy, and territory games, but it can also reach a larger audience of any people who like board games and/or want to play competitively. Even wider it can reach people who just enjoy to play dynamic games.

5c Hands-On Users of the Product

This product is intended for all kinds of people. In fact, this game would be enjoyed by people of any age group people of all age groups have played some sort of a board game in their life. As long as the player is old enough to understand how the game works and the player can figure out how the user-friendly app works then they will be fine. Nowadays, there are so many in-game competitive events going on which is why this is also be a perfect way for those people to compete against other competitive players. Players who might not be suited to play this game are:

Potential user - Any age

Disability - Blind or Reading Illiteracy

Age group: Any

Reason: Reading dialogues and cards are a part of the game that is necessary to understand how it functions and the current state of the game. The game is available in many languages so translation is not an issue. A young child might not be suited to play this game without assistance if they cannot read yet. Any age might not be suited to play the game without assistance if they are unable to read, but we will provide voice so the dialogues can be read out loud if a person is blind or cannot read.

Potential user - Any age

Attitude toward technology - Negative

Age group: Any

Reason: Users who have a negative attitude towards technology and are not very good at working with phones or apps on a desktop might struggle to figure out how to play the game, and download the app. We make our user interface as simple as possible, but we cannot guarantee that all players will understand how to navigate especially if they are not good with technology.

5d Maintenance Users and Service Technicians

As a company we will maintain and update the servers as needed. We guarantee a reliable uptime, reliable functionality, consistent updates and saving of users data. All of this will happen internally.

5e Other Stakeholders

Lawyers that we might need if the licensing goes bad for the RISK game, or if we need them to change our game enough to not infringe upon the copyright laws associated with RISK. Beta testers of the game who are helping us test bugs and giving feedback to improve out game. Investors that we might need extra funding for the product that we have not procured yet. Also, our current investors who expect a return on their investment. The stores we launch our game on are stakeholders, as they will also be making a percentage of the profits from our product.

5f User Participation

Users will initially make contributions to the product by playing early releases of the game to test features and give their general feedback. In addition, they can report any technical issues they encounter. After a full release of the game, users will be able to continue giving suggestions and report bugs to help with future updates. The users are crucial to our development because we cannot possibly find every bug as a company, and there are many more of them than us to test the product.

5g Priorities Assigned to Users

Key Users: Players who are coming to play because they wanted to play a dynamic interactive city based RISK game. Also those who are coming for a competitive game. Ideally these are frequent players. These players are key especially for feedback because they are the ones coming from a RISK background and have specific ideas of the game that they want to be fulfilled. Not living up to what these users want can lose a huge chunk of our playerbase and the people who were

originally interested in it for what it is. It is also important to take feedback from our competitive players, because we need to make sure we are living up to the level of competition we originally strived for.

Secondary Users: Players who are coming to play because they wanted a dynamic game and are coming from a war, strategy, and territory based background. These users have not played RISK before, but they are interested in learning the game and playing it. They came to the game for the dynamic nature of it and the genre of it. They have valuable ideas that we can consider but they are not our number 1 priority because what they want might differ from our goal of making the game RISK based, and they are missing that background knowledge.

Unimportant Users: Infrequent and sporadic users. While there feedback might be valuable in things that will get them to play more frequently, their feedback might only suit a small subset of the people who play the game and catering the game to infrequent players will cause a shrinking of our key users. The goal is to be geared to adapt to our core players rather than the fringes.

6. Mandated Constraints

6a Solution Constraints

Description: The product shall operate on all mobile smartphone devices and operating systems that are less than 5 years old.

Rationale: We want to be able to reach a wide audience, but cannot guarantee to support outdated systems

Fit criterion: The system will work on all mobile smartphone devices and operating systems less than 5 years old

Description: The product shall save the data of all users in a database

Rationale: We want to have user accounts, leaderboards, friends lists, among others

Fit criterion: Any player that creates an account will have all of its data saved

Description: The product will use a SQL backend

Rationale: We want a reliable and useful database with a structured schema.

Fit criterion: The backend will be implemented using the rules of the structured query language, and will have a simple schema that fits the data.

Description: The user should be able to play with friends or random players

Rationale: We want to give a wide range of ways a user can play the game that reaches a large audience

Fit criterion: The app will allow users to add and store friends, challenge them in groups, or play against random players.

Description: The user should be able to play against AI computers at different difficulties that uses AI techniques in Python

Rationale: We want different challenges for users based on skills and intelligent bots that can play at certain levels

Fit criterion: The app will implement AI for the user to play against.

Description: The user should be able to play competitively and in tournaments based on the ELO system

Rationale: We want a competitive aspect for the player base so they can challenge themselves against good players.

Fit criterion: The app will have competitive play based on the ELO rating system and tournaments to play in

Description: The app must support cross platform from mobile to desktop

Rationale: We do not want our playerbase to be restricted by the device they are using

Fit criterion: Those using desktop must be able to play against mobile and vice versa

6b Implementation Environment of the Current System

This product is going to be operated on all mobile smartphone devices, and operating systems as long as the software for those devices and operating systems is no more than five years old. It must work in all of the relevant stores in these devices so the app can be downloaded by users. It also will have accessibility components for those who are blind and need text read to them. These platforms also need to be able to play cross platform.

6c Partner or Collaborative Applications

The ability to read and write to a SQL database. The ability to have working AI agents written in Python. The product must work with dedicated servers with a high uptime in order to provide a consistent service to our customers.

6d Off-the-Shelf Software

The off-the-shelf library that our product depends on is the Google Maps API. The game runs off of the Google Maps API to generate maps for gameplay and will not work without this library.

6e Anticipated Workplace Environment

The product can be used anywhere and does not have a anticipated workplace environment. Our app while have background music, but this can be turned off through the device of the owner or through our settings in the game. There will also be other settings to deal with any issues the user may have, such as picture quality, sound management, default settings etc.

6f Schedule Constraints

From launch date we are intending for the project to be released 15 months from the start into the beta version. Here we will have another 7 months of a feedback loop and fixing the bugs from these until we plan to release the app entirely around the Christmas holiday in 2021. The project must be released before this Christmas as that is when our client plans to. It can be released no later than 12/22/2021. Our beta testing release is a loose deadline as we might not be ready for it within that time frame, and depending on how well the production goes we might not need beta testing to be so long. If we do not release it before that Christmas we might not get the revenue and publicity that we were expecting to receive on the product.

6g Budget Constraints

With the speed the product needs to get out there and the large team and high number of domain experts it requires to create the budget for this project is set for $5,000,000. If this is exceeded we can reach out to other investors to give an investment for a percentage of the product ownership.

7. Naming Conventions and Definitions

7a Definitions of Key Terms

Dynamic in our case means our game of RISK will be played on a map that is either random or the user can choose that will be provided by the Google Maps API. It is dynamic in the sense that the user can play on any part of the map provided it has streets.

Competitive play in our case is an ELO based rating system that will match players based on their ELO rating and this will go up and down based on their performance in the game.

ELO/MMR refers to the players skill rating that will be used in matchmaking

Intersections in our case can mean any place where two or more roads meet.

Cross platform in our case means that users on the desktop can play against users on mobile and they are no one is limited to who they can play by the device they have.

RISK refers to the board game RISK released in 1957

Database refers specifically to the SQL database we will store all of the users data in

Testing refers to both beta testing that will be going pre-release on along with consistent testing and feedback we will get from the players throughout post-release

Technology specifically refers to the mobile devices and desktop devices that the app will be available app

App refers to both the mobile and desktop versions unless specified otherwise.

AI refers to Artificially Intelligent bots that we will used to play Human vs CPU games

CPU refers to the AI that will play against the human

Players refers to both human and AI players unless specified otherwise.

Map refers to the Google Maps API or more specifically the subsection of the Google Maps API that the players are currently playing on

Strategy, war, and territory game refers to a specific genre of games that pertain to those categories

Phone will always refer to a smartphone that can download apps

7b UML and Other Notation Used in This Document

The conventions used in this document will follow the UML standard "UML Distilled" 3rd edition by Martin Fowler. If exceptions are used we will define them as we go and as they happen.

7c Data Dictionary for Any Included Models

1. The database of user accounts which will hold players' information such as settings, matchmaking ranking, friend's list, and recent match history.
2. Users will have data analytics taken on them such as play time, days visited in a month, experience playing RISK, and other factors that will allow us to understand why and who is our core playerbase.
3. The game server, which will host games between multiple players as well as single player games.
4. The game-map creation system, which will take a city name as input, and create a map consisting of intersections as the the playing field.

8. Relevant Facts and Assumptions

8a Facts

1. Given that the game is cross platform, it is going to be available on all mobile and desktop devices.
2. Game is going to be in 3D to give the player a realistic impression of the game.
3. Game competes with RISK online on Pogo and the board game RISK

4. The Google Maps API consists of streets and intersections used in our game
5. Internet access will be needed to play an online game
6. The game is going to have daily and weekly challenges.
7. Players will be able to play in either offline or online mode
8. Players will be able to play competitively and in tournaments
9. Players will be able to play with friends or random opponents
10. Players win the game by capturing all territories
11. Players lose the game by having all their territories captured
12. Players win/loss ratio will be kept track.

13. Players analytics will be kept track of
14. Players friend's list, MMR, settings, and match history will be kept track of.
15. Players will be placed on a highscores list
16. There are no games that follow the same model as this project on the market currently.
17. The game will be available on the Steam store, Apple App Store, and Google Play Market.

8b Assumptions

1. Players are going to give regular feedback.
2. We already have the legal rights to make this RISK based game
3. We will be able to get enough dedicated players in order to beta test
4. Players will report bugs as they see them
5. Players will fill out a small questionnaire at registration where they will honestly answer their experience with the game RISK
6. The Google Maps API will not restrict the players from playing on any part of the map as long as there are enough streets and intersections
7. The game is vast enough so that individual players will rarely if not ever repeat the same map twice if chosen randomly
8. Stores on all the devices we have listed will accept our application along with our client as long as we meet he expectations.
9. Our dedicated server will have a high uptime

II. Requirements

9 Product Use Cases

9a Use Case Diagrams

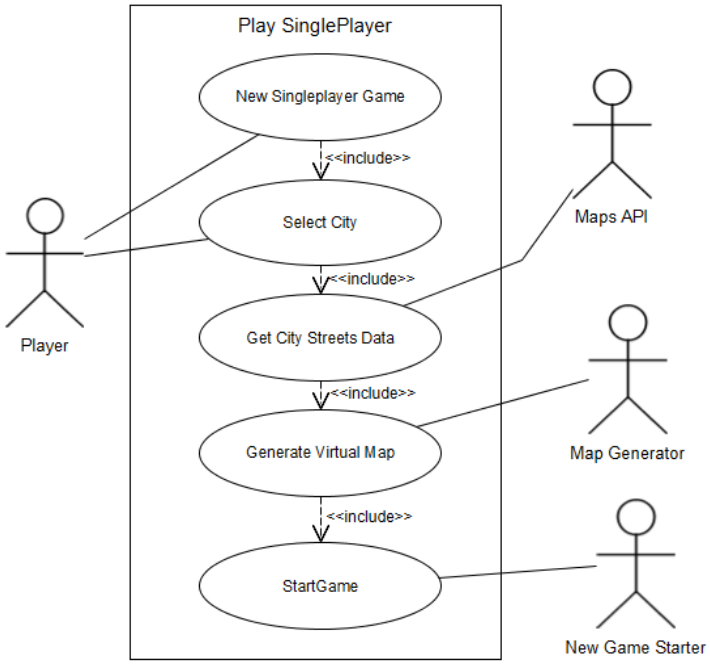


Figure 6 - Use Case Diagram for starting a new single player game

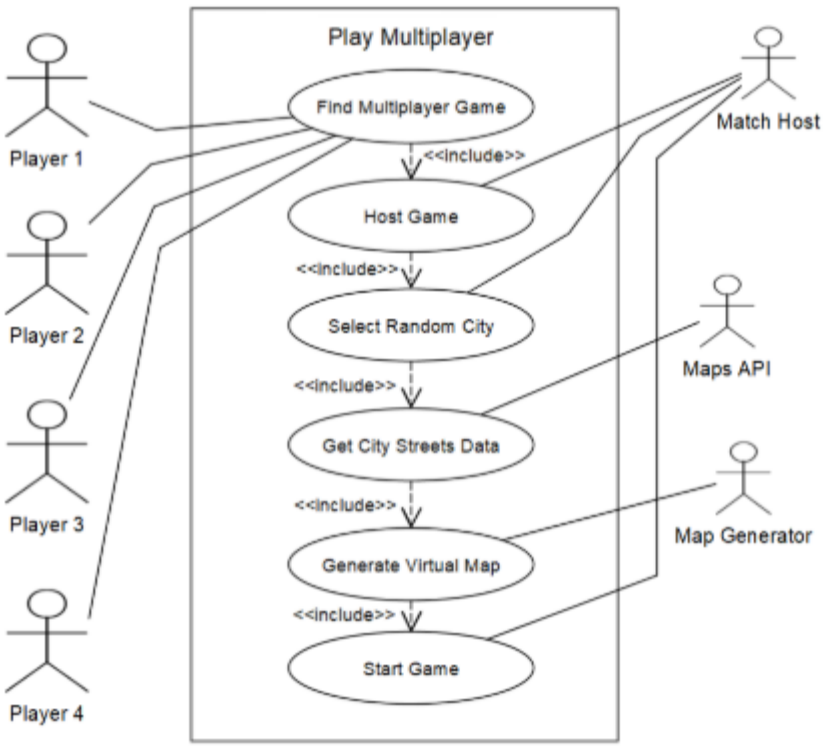


Figure 6 - Use Case Diagram for finding a multiplayer game

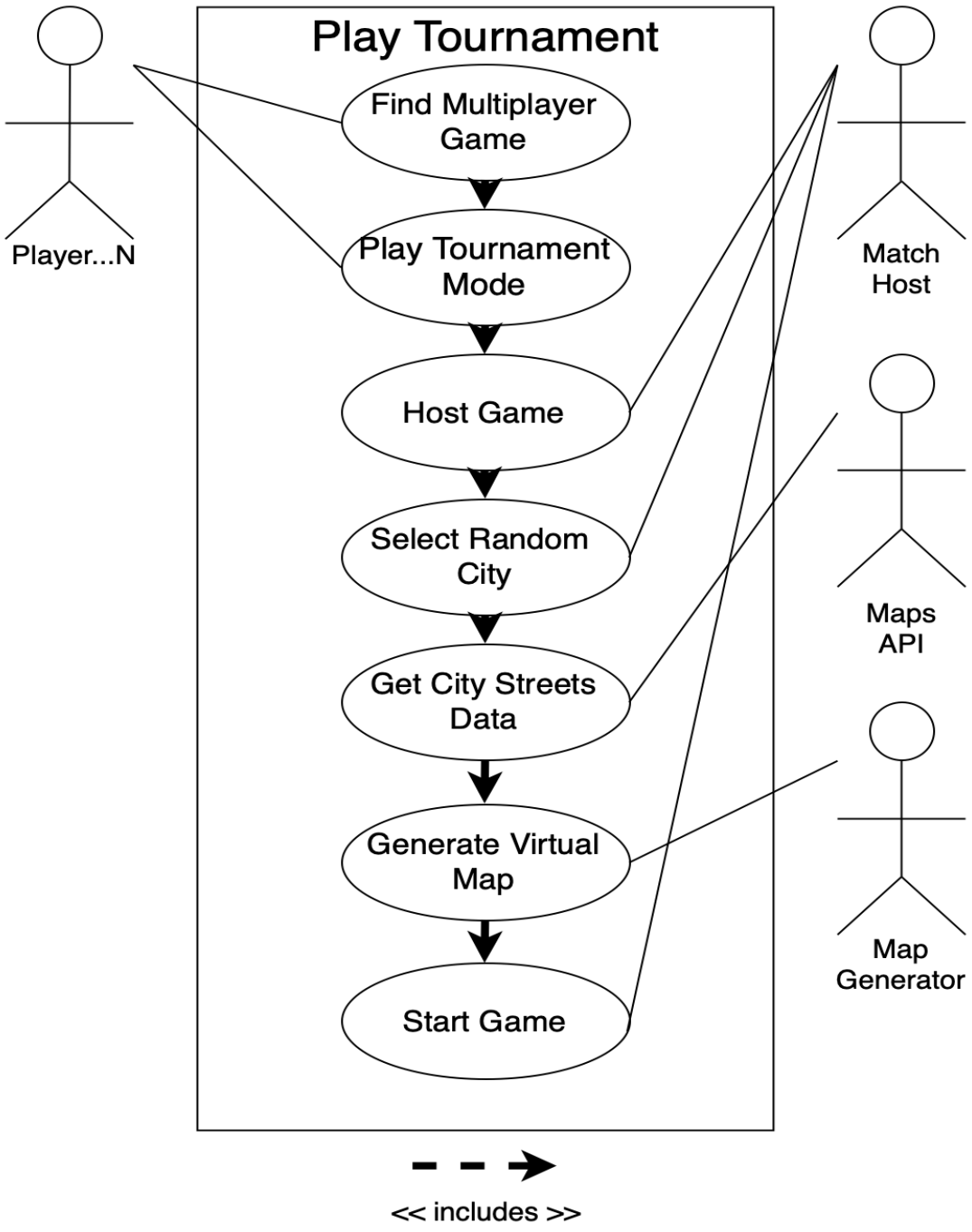


Figure 8 - Use Case Diagram for finding a tournament game

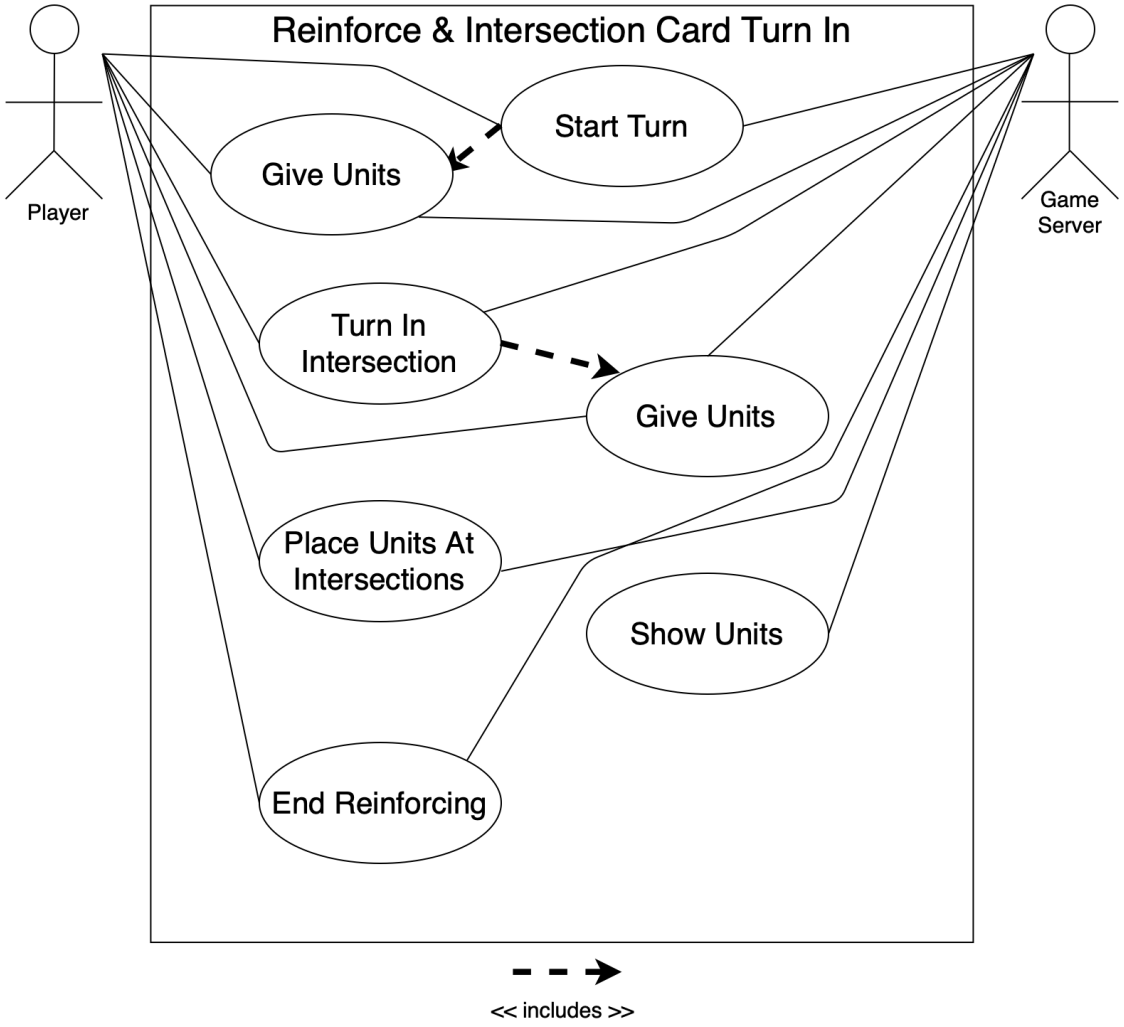


Figure 9 - Use Case Diagram for reinforce and intersection card turn in

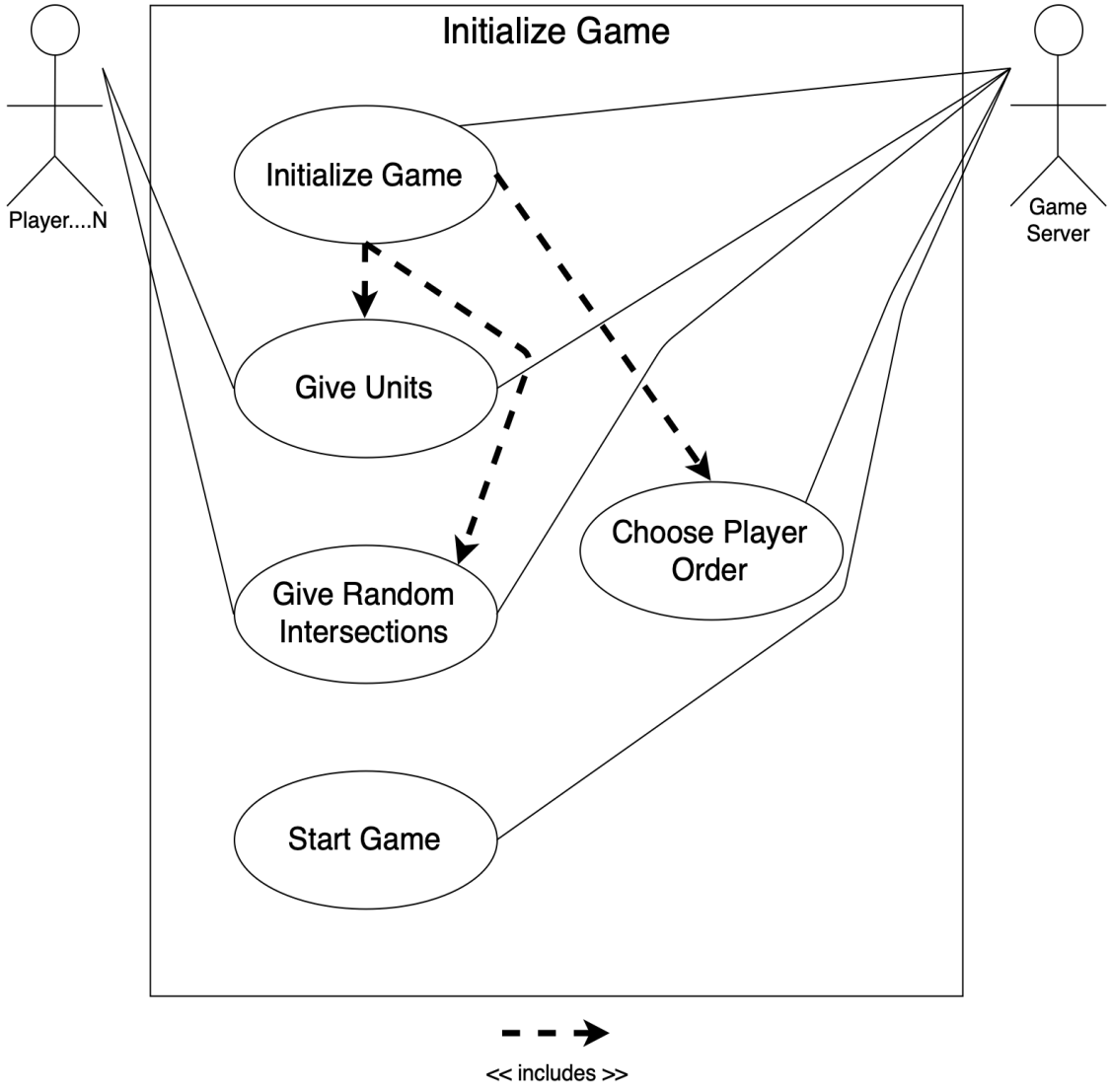


Figure 10 - Use Case Diagram for initialize game

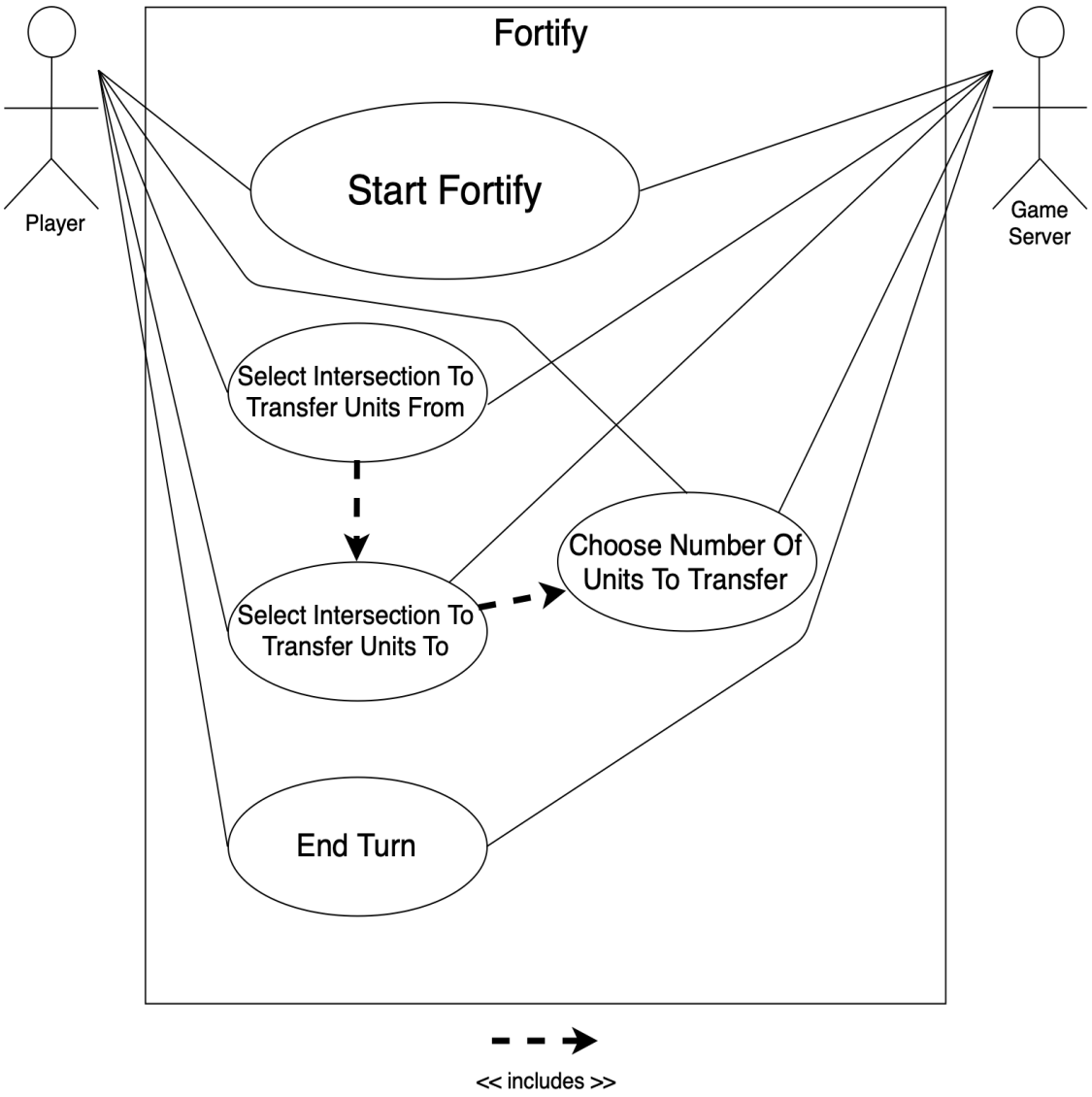


Figure 11 - Use Case Diagram for fortify

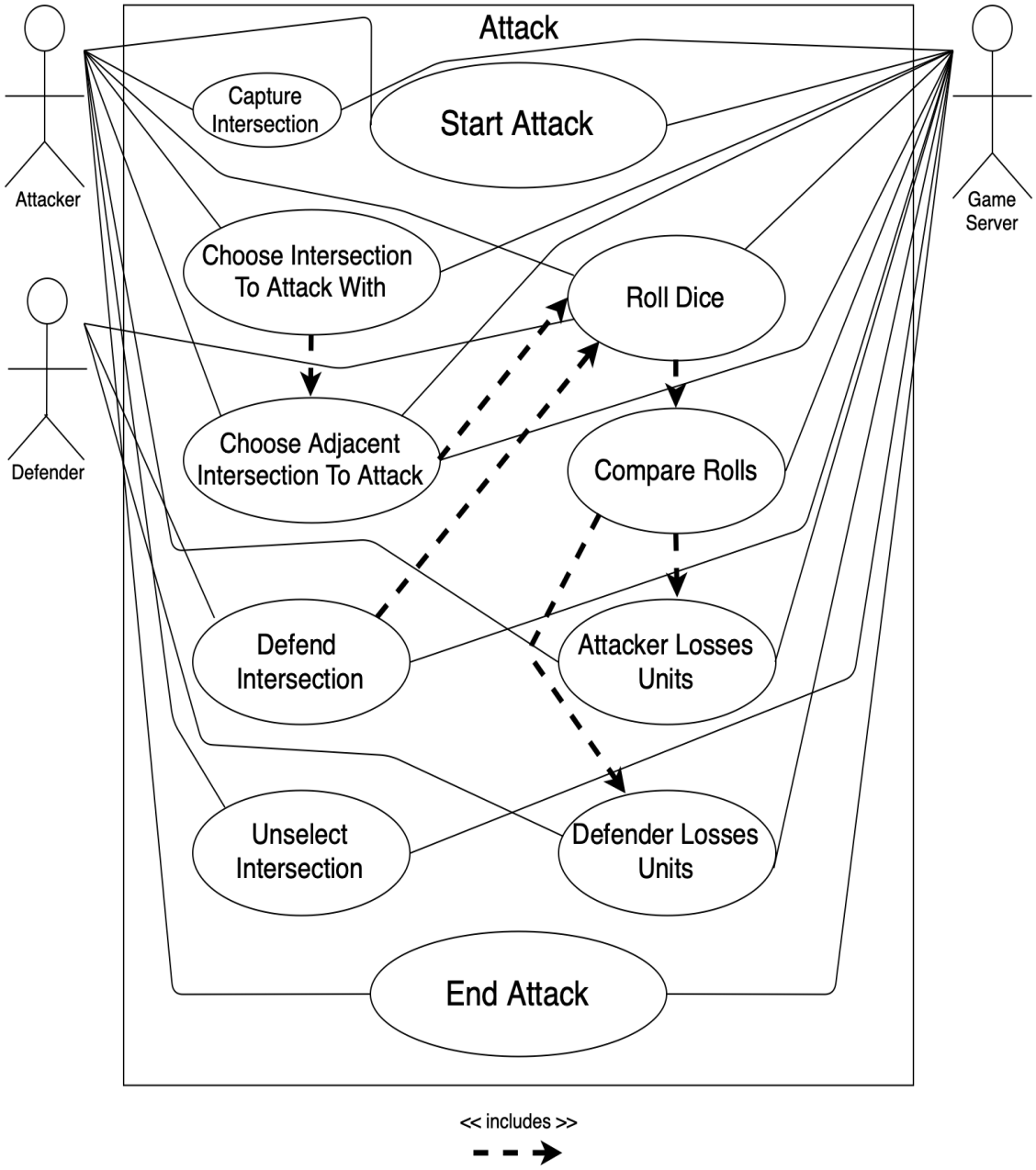


Figure 12 - Use Case Diagram for attack

Figure 2 - Sample Use Case Diagram from Robertson and Robertson

9b Product Use Case List

1. Play single player
2. Play multiplayer

3. Play tournaments
4. Reinforce & turn-in intersection cards
5. Initialize Game
6. Fortify
7. Attack

9c Individual Product Use Cases

Use case ID:
1 - Play single player
2 - Play multiplayer
3 - Play tournaments
4 - Reinforce & turn-in intersection cards
5 - Initialize Game
6 - Fortify
7 - Attack

Name: pre-conditions:
1 - Single Player
- User in main menu

2 - Multiplayer
- User in main menu

3 - Tournament
- User in main menu

4 - Reinforce & turn-in intersection cards
- User in game at start of turn

5 - Initialize Game
- Start of game

6 - Fortify
- User in game at end of attacking

7 - Attack
- User in game at end of reinforcing

post-conditions:
1 - Single Player
- New single player game.
- Select choose city or random city
- Get city and street data.
- Generate virtual map.
- Start game.

Sequence of Events: #1

1. User selects single player option.
2. New Single player game.
3. Select choose city or random city.
4. System and maps API gets city and street data.
5. Generate virtual map. Start game.

Sequence of Events: #2

1. User selects find multiplayer game
2. User selects multiplayer option
3. User selects to host or join a random lobby
    a. System selects random city
    b. System gets city and street data from API
    c. Virtual map generated
    d. Game started

Sequence of Events: #3

1. User selects tournament mode
2. User selects tournament type
    a. Game hosted by server
    b. Random city selected
    c. City and street data received from API
    d. Virtual map generated
    e. Gams is started

Sequence of Events: #4

1. System starts players turn
2. Player asked if they want to turn in intersection cards
    a. Player allowed to turn in if they have a correct combination
    b. Player given units after possible

Alternatives: N/A

Exceptions: N/A

10 Functional Requirements

Requirement # 1 - Singleplayer

Description: The system shall allow users to play the game solo via the single player mode.

Rationale: This will allow users to play the game at their own pace without having to deal with other users.

Fit Criterion: The system will allows users to select a city and intersection before starting the game.

Acceptance Tests: Test 1

Requirement # 2 - Multiplayer

Description: The system shall allow users to play the game with other users via the multiplayer mode.

Rationale: This will allow users to play the game at a much faster pace, and will give them a challenge.

Fit Criterion: The system will allow users to either host a lobby or join a random lobby which will be based off of a player's ELO score.

Acceptance Tests: Test 2

Requirement # 3 - Tournament Mode

Description: The system shall allow users to play the game via the tournament mode.

Rationale: This will allow users to challenge themselves and put their skills to the test.

Fit Criterion: The system will allow users to host a lobby tournament or join a random tournament hosted by players or the system.

Acceptance Tests: Test 3

Requirement # 4 - Player Win/Loss Statistics

Description: The system shall keep track of a player's win/loss ratio.

Rationale: This will allow users to go back, and look at their personal stats.

Fit Criterion: The system will allow users to go to the leaderboards page and see their personal stats.

Acceptance Tests: Test 4

Requirement # 5 - Friends List

Description: The system shall allow players in the game to have a list of other player's usernames.

Rationale: Players will be able to play more games with new people they met online, or with people they already know.

Fit Criterion: The system must allow a list that allows players to view their friends in the game, as well as add new ones or delete existing ones. The system must also allow them to invite their friends to games.

Acceptance Tests: Test 5

Requirement # 6 - Reinforce Intersections

Description: The system shall allow players to reinforce their intersections at the beginning of their turn and then advance when finished.

Rationale: Players will be able to reinforce their intersections at the beginning of rounds to prepare for an attack.

Fit Criterion: The system must allow the user to reinforce their intersections at the beginning of their turn by telling them how many units they have that they can add, allowing them to click on their intersections to add units one at a time per click, and press stop reinforcing to stop reinforcing units and move on to attack

Acceptance Tests: Test 6

Requirement # 7 - Attack Adjacent Intersection and Receiving Dice

Description: The system shall allow players to attack intersections adjacent to them and end the attack and advance when finished.

Rationale: Players will need to attack other intersections in order to capture their opponents' intersection.

Fit Criterion: The system must allow the user to attack adjacent intersections to the intersection they are attacking with. When attacking the attacker and defender will receive dice based on the number of units to roll. The attacker must receive 3 dice if units are at least 3, 2 if units are 2, and 1 if units are 1. The defender must receive 2 dice if units are at least 2 and 1 if units are 1. The system must allow the user to end the attack at any time.

Acceptance Tests: Test 7

Requirement # 8 - Dice rolls

Description: The system shall allow players to roll dice when they are attacking or defending a territory.

Rationale: Players will need to roll dice when they are attacking or defending a territory in order to determine how many units the attacker or defender will lose.

Fit Criterion: The system must allow the to roll dice when they are attacking or defending a territory.

Acceptance Tests: Test 8

Requirement # 9 - Compare dice

Description: The system shall compare the dice rolls of the players.

Rationale: The system needs to calculate how many units to remove from the attacker's and or defender's intersections and it does so by looking at the dice rolls.

Fit Criterion: The system must compare the dice rolls of the players by checking the top dice roll of the attacker to the top of the defender. The system must compare again if they both have at least 2 dice where it will compare the 2nd

highest from both. Each comparison will be given as attacker die higher, defender die higher, or tie.

Acceptance Tests: Test 9

Requirement # 10 - Dice roll Tie

Description: The system shall remove 1 unit from the attacker's territory for each die they tie with the defender.

Rationale: When players are rolling dice when attacking or defending if the attacker's die ties with the defenders then the attacker must lose units.

Fit Criterion: The system must remove 1 unit from the attacker's intersection for each die they tie with the defender.

Acceptance Tests: Test 10

Requirement # 11 - Attacker's Die Higher

Description: The system shall remove 1 unit from the defender's intersection for each attacker's die that is higher than the defender's die.

Rationale: When players are rolling dice when attacking or defending if the attacker's die is higher than the defenders then the defender must lose units.

Fit Criterion: The system must remove 1 unit from the defenders intersection for each die of the attackers that is higher.

Acceptance Tests: Test 11

Requirement # 12 - Defender's Die Higher

Description: The system shall remove 1 unit from the defender's intersection for each defender's die that is higher than the attacker's die.

Rationale: When players are rolling dice when attacking or defending if the defender's die is higher than the attackers then the attacker must lose units.

Fit Criterion: The system must remove 1 unit from the attacker's intersection for each die of the defenders that is higher.

Acceptance Tests: Test 12

Requirement # 13 - Initializing Game

Description: The system shall allocate players to random intersections on the map equally. The order of play will be determined via a random number generator. Then they can reinforce their randomly assigned intersections with 5 units per player to start.

Rationale: To start the game players need to be assigned to random initial intersections to make it fair, along with an order they will take turns in. To start off players will receive 5 units to reinforce intersections to prepare for attacking or defending.

Fit Criterion: The system must allocate the players to random intersections on the map evenly, so each player gets the same amount of intersections. The system also must determine an order of player randomly. Also it must give each player 5 units and allow them to reinforce their intersections.

Acceptance Tests: Test 13

Requirement # 14 - Fortify Intersections

Description: The system shall allow players to fortify intersections after the attack is over. They shall be able to move units to one other intersection as long as they are adjacent. This will work in a chain so any adjacent intersections in a chain can move units. After the player is done fortifying they system will allow them to end turn.

Rationale: In order to prepare for the next turn a player can move around their units to any connected intersections in order to prepare for attacking, defending, or other strategies.

Fit Criterion: The system must allow players to fortify their intersections. They must be able to move their units to one adjacent intersection, or one other intersection existing in a chain of adjacent intersections. The system must allow them to end their turn after they are done fortifying.

Acceptance Tests: Test 14

Requirement # 15 - Capturing Territories

Description: The system shall allow players to move their units to the new intersection after they defeat it in battle. The system shall allow them to move any number of units besides 1 to the new intersection.

Rationale: In order to hold a new intersection a player must have units there in order to hold it. There must be at a minimum of 1 unit per intersection. To add units to the new intersection the player must transfer them from the current attacking unit.

Fit Criterion: The system must allow players to transfer move their units to a new intersection after they defeat it in battle. They must be allowed to only move units from the attacking intersection to the intersection that they just defeated. They must be able to only move any number of units except 1, and also need to move at least 1 to capture the intersection.

Acceptance Tests: Test 15

Requirement # 16 - Winning the game

Description: The system shall declare the winner of the game when they capture every intersection.

Rationale: The game is won when a player captures all of the intersections

Fit Criterion: The system must declare the winner of the game to the player that captures all of the intersections. This can be checked also when there is 1 player left and all other players are eliminated

Acceptance Tests: Test 16

Requirement # 17 - Eliminated from the game

Description: The system shall eliminate a player from the game when they hold no more intersections

Rationale: The player loses the game when they do not hold an intersection anymore.

Fit Criterion: The system must eliminate a player from the game when they hold 0 intersections. They must not be able to take anymore turns, but they are still allowed to spectate. They must be placed in a position from first to last from when they were eliminated from the game.

Acceptance Tests: Test 17

Requirement # 18 - Add Units at the Beginning of Turn and Keeping Track of Units

Description: The system shall give units to the player at the beginning of their turn. The system shall keep track of the number of units a player will receive at the beginning of a turn. Number of units are based on the number of intersections they hold in a chain. Every three adjacent intersections gives 1 more unit per turn plus the 3 given by default plus the amount gained from intersection cards last turn-in.

Rationale: Players receive units at the beginning of the turn to place in the reinforce part of the turn. The number of units is given by default plus bonus units for those who hold a chain to intersections plus amount gained from intersection cards last turn-in. . The larger chain the more units they will get.

Fit Criterion: The system must give 3 units to the player at the beginning of the turn plus 1 more for each three adjacent intersections amount gained from intersection cards last turn-in. The system must keep track of the number of units each player will receive at the beginning of a round.

Acceptance Tests: Test 18

Requirement # 19 - Give Intersection Cards

Description: The system shall gives intersection cards to a player each time they capture at least one intersection. Intersection cards consist of wildcards, infantry, jeeps, and tanks. They will be associated with landmarks in the city.

Rationale: Intersection cards are another way players can gain more units, and need to be distributed when appropriate.

Fit Criterion: The system must give a random intersection card to a player at the end of a turn where they have captured at least 1 intersection.

Acceptance Tests: Test 19

Requirement # 20 - Gain Units from Intersection Cards and Turn-In

Description: The system shall allow a user to turn-in intersection cards at the beginning of a turn before they place any units. The system shall allow the user to turn-in intersection cards with 3 of the same cards, or 2 of the same with 1

wildcard, or 1 with 2 wildcards. If the user turns in 3 wildcard they receive 4 units, if they turn in a combination with tanks they receive 8, if they turn in a combination with jeeps they receive 6, and if they turn in infantry they receive 4.

Rationale: Intersection cards are another way players can gain more units. The artillery are more rare so they receive more units when a player can make a combination of them. It is another way to strategize in the game to win.

Fit Criterion: The system must allow a user to turn-in intersection cards at the beginning of the turn before they place units. The system must allow them to turn-in any 3 of the same cards, or 2 of the same with 1 wildcard, or 1 with 2 wildcards. When a user turns the cards the system must give the player 8 units if they turn in a combination with tank 8, if they turn in a combination with jeep they receive 6, and if they turn in infantry or wildcards they receive 4.

Acceptance Tests: Test 20

11 Data Requirements

Requirement # 21 - Account Database

Description: The system shall allow users access to their account information.

Rationale: This will allow users to edit their name, email, payment information, and friends' list.

Fit Criterion: The system must allow users to store and access their account information.

Acceptance Tests: Test 21

Requirement # 22 - Game Server

Description: The system shall consist of a game server that has the capability for hosting games.

Rationale: To allow multiple different games to be hosted concurrently.

Fit Criterion: The system will have one game server running a large amount of games at the same time.

Acceptance Tests: Test 22

Requirement # 23 - Match History Database

Description: The system should consist of a database that will be used to hold match history information.

Rationale: To allow players to view their own match history, or the match history of other players.

Fit Criterion: The system will have an interface that will list items, where each item displays the date of the match, how long the match took, and the winner of the match.

Acceptance Tests: Test 23

12 Performance Requirements

12a Speed and Latency Requirements

Requirement # 24 - Launch Time

Description: The application shall load completely within 10 seconds upon opening.

Rationale: To get users into the game in a reasonable amount of time.

Fit Criterion: The game will be at the main menu screen within less than 10 seconds of launching.

Acceptance Tests: Test 24

Requirement # 25 - Connection Time

Description: The application shall connect to the user to the online server within 3 minutes upon choosing an online feature if they are connected to the internet.

Rationale: To make connection to opponents in a reasonable amount of time

Fit criterion: The game will connect the user to the online server within less than 3 minutes, if all users do not connect to the game within this time the lobby will be abandoned and the game cancelled.

Acceptance Tests: Test 25

12b Precision or Accuracy Requirements

Not Applicable

12c Capacity Requirements

Requirement # 26 - Game Server Capability

Description: The game server shall be able to host as many matches that would include all of the playerbase playing in a match simultaneously.

Rationale: To ensure every who has the game open will be able to play in an online match.

Fit Criterion: Each game server will be able to host at least 1 million games.

Acceptance Tests: Test 26

Requirement # 27 - Player Server Capability

Description: The game shall be able to have as many players online that would include all of the player base online simultaneously

Rationale: To ensure every who has the game open will be able to play and/or go through the menus

Fit Criterion: Each player server will be able to host at least 3 million players.

Acceptance Tests: Test 27

13 Dependability Requirements

13a Reliability Requirements

Requirement # 28 - Game Server Reliability and Maintenance

Description: The player shall have access to the game server except during scheduled maintenance.

Rationale: To allow players to have the maximum amount of uptime to play the game, but allow us to also update the game and maintain it as needed.

Fit Criterion: The game has a continuous connection to the game server and will have that connection closed when server maintenance begins. The player will no

longer be able to start any kind of game and will have to reboot and update in order to play again.

Acceptance Tests: Test 28

13b Availability Requirements

Requirement # 29 - Continuous Connection to Server

Description: The system should be continuously connected to the game server except for the 30 minute server maintenance period.

Rationale: To give players the most amount of time during to play the game.

Fit Criterion: Connection tests done throughout the week checking for fidelity.

Acceptance Tests: Test 29

13c Robustness or Fault-Tolerance Requirements

Requirement # 30 - Offline Mode

Description: Players shall be able to play offline games when the game server is down or if the player doesn't have an internet connection.

Rationale: To ensure players don't need a connection to the game server to play the game.

Fit Criterion: While there is no active connection to the game server, or while there is no internet connection from the player's end, the player will be able to create and play a game offline. Because there is no connection to the Google Maps API, a predefined set of maps will be available.

Acceptance Tests: Test 30

13d Safety-Critical Requirements

Requirement # 31 - Seizure Warning

Description: The game shall warn players of epileptic seizure risks that may arise during play of the game.

Rationale: To inform players who have epilepsy that playing the game may trigger a seizure.

Fit Criterion: This notification will be visible from the loading screen and will be a warning and disclaimer to all users.

Acceptance Tests: Test 31

14 Maintainability and Supportability Requirements

14a Maintenance Requirements

Requirement # 32 - Server Maintenance

Description: The system servers shall undergo maintenance once every week.

Rationale: To clean up bugs on the server as well as push any pending updates to the game.

Fit Criterion: Once a week, the servers will stop hosting new games for a period no greater than 30 minutes.

Acceptance Tests: Test 32

14b Supportability Requirements

Requirement # 33 - Player Help

Description: The system shall have the infrastructure for the users to be able to get support 24/7 via help desk, live chat help, and community forums.

Rationale: We want the user to be able to get help with any questions that they have, or any support they need at all times

Fit Criterion: The user will be able to go to the help page in the application to get access to live chat, the help desk number, and community forums. They then will be able to use any of these to ask a question and get help within 10 minutes.

Acceptance Tests: Test 33

Requirement # 34 - In Game Help

Description: The system shall allow the users to enable or disable in game tips that would make using playing the game easier and more enjoyable for them.

Rationale: Sometimes it will be more convenient for users to get help from in game rather than seeking help outside of it.

Fit Criterion: An interface that allows users to enable or disable hints, and having the hints presented at the appropriate times if they're enabled

Acceptance Tests: Test 34

14c Adaptability Requirements

Requirement # 35 - Platform Requirements

Description: The game shall be able to run on Android and Apple devices, as well as Windows, Mac, and Linux operating systems.

Rationale: To allow anyone to play the game, regardless of their platform.

Fit Criterion: Users will play separate matches on Android devices, Apple devices, Windows computers, Mac computers, and Linux computers.

Acceptance Tests: Test 35

14d Scalability or Extensibility Requirements

Requirement # 36 - Growing Database and Server Capability

Description: The databases and servers for the game shall be upgraded as needed when more players purchase the game.

Rationale: To ensure that every player's account is storable and that there enough server capability to host as many games as needed.

Fit Criterion: The databases and servers will be able to service at least more players than are currently in the database.

Acceptance Tests: Test 36

14e Longevity Requirements

Requirement # 37 - Game Lifetime

Description: The game servers shall run for a period of at least 5 years from the initial release.

Rationale: To give players the opportunity to have a longer period of time to play the game.

Fit Criterion: The game will allow players to play online matches and the game will have consistent updates until at least 5 years from release.

Acceptance Tests: Test 37

15 Security Requirements

15a Access Requirements

Requirement # 38 - Player Access to Data

Description: The players shall have access to their account info and can edit their name, email, payment information, and friends' list

Rationale: The user should have access to their own data and be able to change it as needed

Fit Criterion: The system will allow the user to be able to edit their account info, including their name, email, payment information, and friends' list

Acceptance Tests: Test 38

Requirement # 39 - Help Staff Access to Data

Description: Help staff will have access to players' account information for when players contact them for help.

Rationale: The users will be able to receive professional help for any questions or concerns they may have.

Fit Criterion: The system will allow users to contact help staff when they have any questions

Acceptance Tests: Test 39

15b Integrity Requirements

Requirement # 40 - Information Backups

Description: The account information database shall create backups of itself and store it to another database frequently.

Rationale: To retain account information for users in case of unforeseen accidents or faults.

Fit Criterion: The system will restore the backup and validate the information to the backup to ensure the proper information is restored.

Acceptance Tests: Test 40

Requirement # 41 - Database Security

Description: The account information database shall have sophisticated encryption.

Rationale: To ensure user account information is secured and not accessible for malicious intent. This is important to protect the purchase information of users.

Fit Criterion: A developer in a controlled environment will attempt to break into the information database. The data will be validated to be properly encrypted.

Acceptance Tests: Test 41

15c Privacy Requirements

Requirement # 42 - User Privacy

Description: The system shall ensure that a user's personal information such as email address and payment information is only accessible by the appropriate and qualified staff.

Rationale: To ensure that user information is not mishandled.

Fit Criterion: Restrict permissions on accessing specific user data.

Acceptance Tests: Test 42

15d Audit Requirements

Requirement # 43 - Purchase History

Description: The system shall allow users to view their purchase history for the game.

Rationale: To allow users to keep track of what they purchased in game

Fit Criterion: An interface will list the user's purchases as individual items with monetary values and date of purchase.

Acceptance Tests: Test 43

Requirement # 44 - Returns

Description: The system shall allow users to make returns

Rationale: To allow users to make returns on purchases they made have made by accident, or things they do not want anymore.

Fit Criterion: The system will allow the user to return any purchase with explanation.

Acceptance Tests: Test 44

15e Immunity Requirements

Requirement # 45 - Security 1

Description: The system shall be able to resist malicious attacks.

Rationale: To ensure system data is not able to be corrupted or maliciously retrieved.

Fit Criterion: The system will conduct routine checks to make sure appropriate data has not been wrongfully modified or accessed.

Acceptance Tests: Test 45

Requirement # 46 - Security 2

Description: The system shall be able to resist viruses, spyware, malware, and other attacks

Rationale: To ensure system data is not able to be corrupted or maliciously retrieved. Also protect our system from attackers.

Fit Criterion: The system will run constant antivirus software and scans with as much uptime as the servers are up. Scans will be made as soon as the system boots back up after going down.

Acceptance Tests: Test 46

16 Usability and Humanity Requirements

16a Ease of Use Requirements

Requirement # 47 - Simple User Interface

Description: The system shall have an interface that makes navigation through the product simple for users of all ages.

Rationale: To ensure users do not have any problems navigating through the various game modes.

Fit Criterion: The system will allow users to simply navigate through the interface with simple user menus and help overlays for things that might need more explanation. The user will take random surveys asking how satisfied they are with the ease of using the app ranging from very unsatisfied to very satisfied. The user will be able to give feedback on any response that is somewhat satisfied or lower.

Acceptance Tests: Test 47

16b Personalization and Internationalization Requirements

Requirement # 48 - Language Support

Description: Different languages and currencies shall be added as time goes on.

Rationale: We want this to be a global game where nobody is limited by the language they can speak. We also want people to be able to make purchases in currencies they are familiar with.

Fit Criterion: The game will be available in every language that is available Google Translate, and will have microtransactions for every currency available in the countries of the languages we have. The prompts we give in all languages will be verified by native speakers of the language.

Acceptance Tests: Test 48

16c Learning Requirements

Requirement # 49 - Game Learnability

Description: The system shall allow the player to learn the basic mechanics of the game after completing the game tutorial.

Rationale: To allow players to understand the basics of the game.

Fit Criterion: The system will have an option for the player to go through different levels of the tutorial, starting from the basics of the game to more advanced tactics.

Acceptance Tests: Test 49

16d Understandability and Politeness Requirements

Requirement # 50 - User help

Description: Each part of the user interface shall have a question mark near items that might need more explanation.

Rationale: To allow players to pick up the game quickly and make it more understandable overall

Fit Criterion: The system will provide the user with a digital manual and question marks near items that might need more explanation.

Acceptance Tests: Test 50

16e Accessibility Requirements

Requirement # 51- Colorblind Mode

Description: The system shall include a color blind mode.

Rationale: To allow color blind users to have access to the game and make it easier for them to distinguish the different colors used in the game.

Fit Criterion: The system will allow the functionality of a colorblind mode when chosen.

Acceptance Tests: Test 51

Requirement # 52- Text to Speech

Description: The system shall have Text to speech.

Rationale: To allow players to play without hands and/or read the text without having to see it to aid the user.

Fit Criterion: The system will allow the user to enable or disable Text to Speech mode and the system will say the text on the screen out loud. The system will say the item when pressed once and the will provide the user to press it again to select or choose another option.

Acceptance Tests: Test 52

16f User Documentation Requirements

Requirement # 53 - Legal Agreement

Description: The system shall give the users a legal notification when they launch the game for the first time. They shall accept it in order to play.

Rationale: To inform players that they cannot hold the developers liable for any harm they encounter while playing the game and to protect our companies intellectual property.

Fit Criterion: The system will provide the user a terms of service agreement that the player will have to sign in order to play the game.

Acceptance Tests: Test 53

16g Training Requirements

Requirement # 54 - Tutorial

Description: The game shall have a tutorial explaining how the game is played.

Rationale: To teach new players how to play the game.

Fit Criterion: The system will have an option for players to start the game tutorial in the game interface.

Acceptance Tests: Test 54

Requirement # 55 - Help Option

Description: The game shall contain a series of help menus.

Rationale: To allow players to understand what certain actions do in the game.

Fit Criterion: The system will have visible help options visible throughout the game for the user to select.

Acceptance Tests: Test 55

17 Look and Feel Requirements

17a Appearance Requirements

Requirement # 56 - Logo Rights

Description: The product shall have full rights to use the official RISK logo and product from Hasbro Inc.

Rationale: We intent to expand the target audience for the product since the use of the official logo will instantly grab attention.

Fit Criterion: The company will receive the full rights to use the RISK logo and product.

Acceptance Tests: Test 56

Requirement # 57 - Themes

Description: The system shall have a wide variety of themes depending on what time of the year users are active. The system shall include seasonal themes as well as holiday themes. Seasonal themes shall include summer, fall, winter, and spring whereas holiday themes shall include Independence Day, Christmas, Valentines Day, and Saint Patrick's Day.

Rationale: In order to ensure the number of users active is consistent, various themes will allow player(s) to stay engaged in the game.

Fit Criterion: The system will have a wide variety of themes based on the time of the year and/or current holiday event that is occurring. This will line up with the current update that is released to push out these themes.

Acceptance Tests: Test 57

17b Style Requirements

Requirement # 58 - Overall Appearance

Description: The system shall have a modern look.

Rationale: In order to maintain a steady user base by making the interface more appealing and aesthetic.

Fit Criterion: The user will take random surveys asking how satisfied they are with the appearance of the app ranging from very unsatisfied to very satisfied. The user will also be able to give feedback for any rating lower than somewhat satisfied that can help improve the application.

Acceptance Tests: Test 58

18 Operational and Environmental Requirements

18a Expected Physical Environment

Requirement # 59 - Internet Accessibility

Description: The physical environment shall allow access to the internet to allow the user to play online.

Rationale: To allow the user to play matches online.

Fit Criterion: The system will allow users with online access to play online games.

Acceptance Tests: Test 59

18b Requirements for Interfacing with Adjacent Systems

Requirement # 60 - External APIs

Description: The system shall interface with the Google Maps API everytime a new match is created in order to download the relevant city street information.

Rationale: To use the city street information to create the virtual map battlefield.

Fit Criterion: The system will have successful retrieval of map information and apply that to the game map.

Acceptance Tests: Test 60

18c Productization Requirements

Requirement # 61 - Distribution Services

Description: The game shall be available on Steam for computers, as well as the Apple App Store and Google Play Store for Apple devices and Android devices respectively.

Rationale: To allow players on different platforms to have access to playing the game.

Fit Criterion: The system will provide users with access to installing the game from the Apple App Store, Google Play Store, and Steam.

Acceptance Tests: Test 61

18d Release Requirements

Requirement # 62 - Release Timeline

Description: The system shall be available as a full release 22 months after the project officially begins in development.

Rationale: To ensure there is a reasonable amount of time for the development

Fit Criterion: The game will be released within 22 months after the project fully begins the development.

Acceptance Tests: Test 62

Requirement # 63 - Beta Release Timeline

Description: The system shall be available for beta testing 15 months after the project fully begins development

Rationale: To ensure there is a reasonable amount of time for the fixing bugs and taking in beta user input

Fit Criterion: The game will be released in beta within 15 months after the project fully begins development.

Acceptance Tests: Test 63

19 Cultural and Political Requirements

19a Cultural Requirements

Not Applicable

19b Political Requirements

Not Applicable

20 Legal Requirements

20a Compliance Requirements

Requirement # 64 - Game Rights

Description: The game shall contain the rights for RISK from Hasbro Inc.

Rationale: To avoid legal issues from Hasbro Inc.

Fit Criterion: The company will receive a legal agreement with Hasbro Inc. in which they allow the usage of the RISK trademark and game functionality.

Acceptance Tests: Test 64

20b Standards Requirements

Requirement # 65 - Game Rating

Description: The game shall be rated by the Entertainment Software Rating Board(ESRB)

Rationale: To conform with legal regulations of digital games and to ensure the consumer knows the ages allowed to play the game.

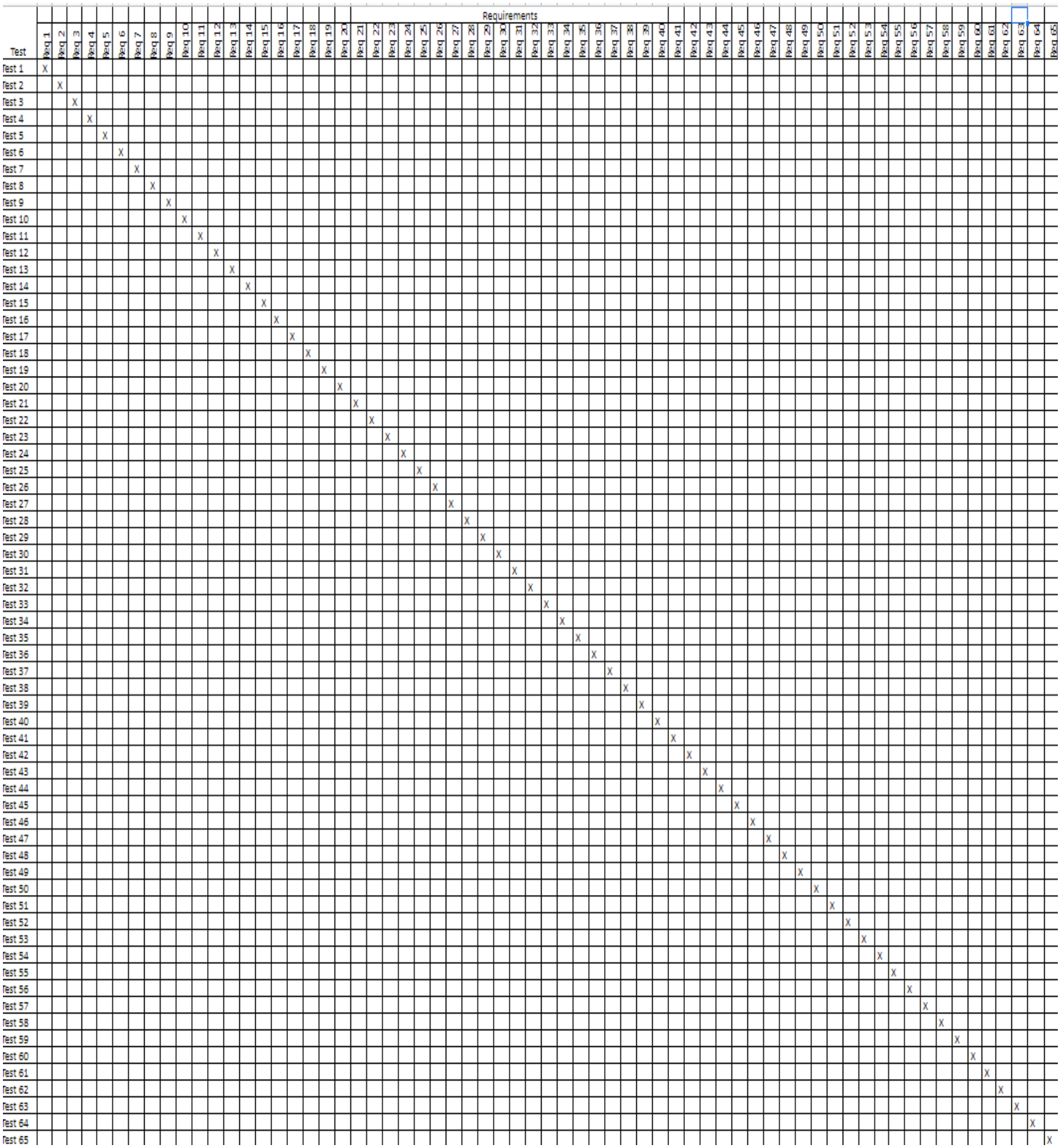
Fit Criterion: The product shall receive a game rating by the ESRB.

Acceptance Tests: Test 65

21 Requirements Acceptance Tests

21a Requirements – Test Correspondence Summary

Table 1 - Requirements - Acceptance Tests Correspondence



21b Acceptance Test Descriptions

ID # 1 - Singleplayer Test **Description**: This test will ensure that the user can select single player mode and be able to select a map and difficulty to play on.
ID # 2 - Multiplayer Test **Description:** This test will ensure a player can host a lobby or join a random one that is based off their ELO score. It will also ensure that the ELO scores are within 200 of each other.
ID # 3 - Tournament Mode Test **Description:** The test will ensure the player can join tournament lobbies or create one on their own.
ID # 4 - Player Win/Loss Statistics Test **Description:** This test will ensure players can go to the leaderboards page and see their stats, it will also ensure that they are seeing the correct stats.
ID # 5 - Friends List Test **Description:** This test will ensure players can view their friends list in game and that they are able to add and delete existing players in the game as friends as well. This will also check that they will be able to invite their friends. This test will also try to delete friends and add friends that do not exist and make sure it fails.
ID # 6 - Reinforce Intersections Test **Description:** This test will ensure players will be able to reinforce their intersections at the beginning of a turn. The player must be able to see the number of units they can add and on each click of an intersection a unit must be added until 0 units are left. At the end the player must be able to end their turn to pass. This test will also try to reinforce intersections that the user does not own and make sure it fails.
ID # 7 - Attack Adjacent Intersection and Receiving Dice Test **Description:** The test will ensure a player will be able to attack and adjacent intersection and receive their dice when choosing this option. Also it ensures the defender also get their dice. Checking the dice works as follows: the attacker must get 3 if 3 units are on the attacking square, 2 if 2, and 1 otherwise; for the defender they will get 2 if it is at least 2 or 1 otherwise. The test will also end attacks at random points to ensure this works as well. It also will attempt to attack non adjacent intersections and see what happens.
ID # 8 - Dice rolls test **Description:** This test will ensure that the attacker and the defender can roll dice it will also check the dice rolls are valid numbers.

ID # 9 - Compare dice test
Description: This test will ensure that the dice rolls are being compared correctly. It will check that the top dice rolls are being compared for both. To do this it must check the top dice roll of the attack and defender and compare it then if they both have at least 2 units it will check the 2nd highest of both. To make the test more difficult it can have iterations where all of the dice is the same and there is no 1st and 2nd highest. Also it must ensure the 3rd highest test is never being checked.

ID # 10 - Dice roll Tie Test
Description: This test will ensure that the dice roll is in fact a tie and if so it will remove the unit from the attacker's intersection for each tie with the defender. It must fail to remove anything on other dice results, and ensure the proper intersections are having units from them.

ID # 11 - Attacker's Die Higher Test
Description: This test will ensure that the attacker's die is in fact higher and then it will proceed to remove 1 unit from the defender's intersection if so for each attacker's die that is higher. It must fail to remove anything on other dice results. and ensure the proper intersections are having units removed from them.

ID # 12 - Defender's Die Higher Test
Description: This test will ensure that the defender's die is in fact higher and then it will proceed to remove 1 unit from the attacker's intersection if so for each defender's die that is higher. It must fail to remove anything on other dice results, and ensure the proper intersections are having units removed from them.
Rationale: When players are rolling dice when attacking or defending if the defender's die is higher than the attackers then the attacker must lose units.
Fit Criterion: The system must remove 1 unit from the attacker's intersection for each die of the defenders that is higher.
Acceptance Tests: Test 10

ID # 13 - Initializing Game Test
Description: This test will ensure that the players are given random intersections on the map evenly that is that each user will get the same number of intersections. It also must ensure that there is a random order for the users. It also must ensure that each player is receiving 5 units each.

ID # 14 - Fortify Intersections Test
Description: This test will ensure players can fortify their intersections after attacking. It must ensure that a user can only move to an adjacent one, or ones in an existing chain. This test must also try to move to non adjacent intersections and ensure that it fails when doing so. Also it must try to move from a intersection with 1 unit and also fail. After all this it must allow them to end the turn and test must check that the turn is ended.

ID # 15 - Capturing Territories Test
Description: This test will ensure players can capture the adjacent intersection that they have just defeated. It will also check that they are able to transfer their units to it and ensure they are not sending too few or too many. They must send at least one and the test must fail if they send 0. They also must send at most all of the units at the capturing, expect 1. It must try to check these passing the boundary cases and fail is any of these pass.

ID # 16 - Winning the game test
Description: The test must ensure that when a person wins the game they have actually captured all of the intersections and all other players have been eliminated. It also must ensure they are shown a winning screen.

ID # 17 - Eliminated from the game test
Description: This test must ensure that when a player no longer holds an intersection that they are eliminated from the game. It must continue on and ensure that their turn is skipped over after elimination. It also must check that they are placed in the right win loss order, so 2 players that lost in the same turn the first one is in a lower place.

ID # 18 - Add Units at the Beginning of Turn and Keeping Track of Units Test
Description: The test must ensure that the user is given 3 units at the beginning of the turn plus 1 for each 3 adjacent intersections gained plus and amount gained from any intersection cards that were turned in in the last turn-in. This will be kept track by the system so this test can have many sanity checks making sure the number of units are coming from the right places from the right things.

ID # 19 - Give Intersection Cards Test
Description: This test will ensure the players are given random intersection cards a t the end of the turn when they have captured at least 1 intersection. It must make sure not to give a card out when no intersection is captured. The cards must only be a jeep, an artillery, or a tank.

ID # 20 - Gain Units from Intersection Cards and Turn-In Test
Description: This test will ensure the ensure the user can turn in intersection cards at the beginning of a turn before they place any units. The test will ensure user can turn-in intersection cards with 3 of the same cards, or 2 of the same with 1 wildcard, or 1 with 2 wildcards. If the user turns in 3 wildcard they receive 4 units, if they turn in a combination with tanks they receive 8, if they turn in a combination with jeeps they receive 6, and if they turn in infantry they receive 4. So it must ensure they receive the proper number of units and receive nothing if the right combinations are not used.

ID # 21 - Account Database Test **Description:** This test will ensure the users can store and access their account information at all times by displaying their data and allowing them to change it. This test will fail if the user cannot change their information or access it when the servers are up.
ID # 22 - Game Server Test **Description:** This test will ensure that each game server can have a large amount of games at the same time. To test this there will be many simulated bot games that will max out the servers to ensure they can hold that many concurrently.
ID # 23 - Match History Database Test **Description:** This test will ensure that the interface will show the user the history of their matches consisting of dates, length, and match winners. In order to pass the user must be able to see the history for all matches.
ID # 24 - Launch Time Test **Description:** This test will ensure that the game will be at the main menu screen within 10 seconds of opening the application. If not the test will fail.
ID # 25 - Connection Time Test **Description:** This test will ensure that the user can connect to the online server in no more than 3 minutes. It will also ensure if any player connects in this time that the game will fail to create and it will be abandoned.
ID # 26 - Game Server Capability Test **Description:** This test will ensure each game server can host 1 million games. In order to test this each server must have 1 million games going concurrently and it will be able to run all of these. It will also ensure each server has at most 1 million games at once.
ID # 27 - Player Server Capability Test **Description:** This test will ensure the server can hold at least 3 million players at once. To test this the server must be loaded to its boundary. Also the server should not hold more than 3 million or they must be put to another server, it should test the server is not overloaded.
ID # 28 - Game Server Reliability and Maintenance Test **Description:** This test will ensure that the game has a continuous connection at all times and ensure the connection gets properly closed when maintenance occurs so no players are playing during maintenance. Once maintenance is turned on the system must prevent users from starting a new game until the update is finished.

ID # 29 - Continuous Connection to Server Test
Description: This test will ensure the connection to the server is constant. It will also check the connection fails when the server is being updated for 30 minutes a week.

ID # 30 - Offline Mode Test
Description: This test will ensure that the player can still play single player game when they are not connected to the internet or when the game server is down. It must be able to be played regardless of the game server status, and they will only be able to play on preloaded maps.

ID # 31 - Seizure Warning Test
Description: This test will ensure the players will see a seizure warning from the loading screen that gives them the full details.

ID # 32 - Server Maintenance Test
Description: This test will ensure our servers are only down for the 30 minutes specified time for the tests.

ID # 33 - Player Help Test
Description: The test will ensure that the player will always have access to live chat, the help desk number, and community forums. This test will fail if they don't get help within 10 minutes.

ID # 34 - In Game Help Test
Description: This test will allow the players to enable or disable hints. It will fail if the hints are shown when they shouldn't be or the inverse.

ID # 35 - Platform Requirements Test
Description: This test will ensure players can play matches and get the application on Android, IOS, Windows, Mac, and Linux. It will also allow them to play with any of the other devices.

ID # 36 - Growing Database and Server Capability Test
Description: This test will ensure that we do not have a larger player base existing in our database than our servers and databases can hold at full capacity given our current capacities. If it exceeds it we will be warned and told to upgrade or increase the number of servers and databases.

ID # 37 - Game Lifetime
Description: This test will ensure that we are doing consistent updates for at least 5 years from release and that players can still make online matches. These tests will happen incrementally and if a player is unable to make a match because we are not maintaining our servers, or if we do not push out at least 1 update every 2 week this test will fail.

ID # 38 - Player Access to Data Test
Description: This test will ensure the player should have access to their user information and be able to edit their account info, name, email, payment information, and friends' list including adding to these are removing these any any time. If any of this information is unable to be changed or removed at any time the test will fail.

ID # 39 - Help Staff Access to Data Test
Description: This test will ensure the help staff should have access to the players at all times so when they have a question the user can get help. This will also ensure that when a player has a question they are able to get help from help staff. It will fail if the user is unable to get help.

ID # 40 - Information Backups Test
Description: This test will ensure the system can restore a backup whenever necessary and it will test to validate the information that is being backed up when a restore occurs. If any data is inconsistent or a backup cannot be made or restored to this will fail.

ID # 41 - Database Security Test
Description: This test will ensure that we have an advanced enough encryption that a team of white-hat hackers cannot break into our system. We need to ensure there are no ways to break in and that our data is properly encrypted if they do. This will fail if this team can break in or important information is not being encrypted.

ID # 42 - User Privacy Test
Description: This test will ensure the players are not able to access specific data of them that should be only accessible by staff and developers. This test will also ensure players cannot see this restricted information of each other. It will fail if there are any security flaws here.

ID # 43 - Purchase History Test
Description: This test will ensure that that the interface will show the user the items they have purchased, the date, and the price. If any of this information is inconsistent or incorrect this test will fail. This test can also make sure items being returned are being shown as well.

ID # 44 - Returns Test
Description: This test will ensure the user can return any of their purchases within 2 days of making it without explanation, or 7 days with an explanation that is inspected by help staff. They must not be able to return an item after 3 days without first being approved so this should fail until approval.

ID # 45 - Security Test 1

Description: This test will ensure that our system is robust to malicious attacks. This will also do routine checks to make sure that data is not being accessed by a hacker, or modified. So if there is any improper access, system failing to attacks, or changed data this will fail.

ID # 46 - Security Test 2

Description: The system test will ensure that our antivirus software is constantly running and scanning as long as the servers are up. If our software is not running at any time this test will fail. This will also ensure instant scans as soon as the system is rebooted.

ID # 47 - Simple User Interface Test

Description: This test will ensure the users can easily navigate through the interface. In order to check we will give users random tests to check out satisfied they are will the usability of the interface. If the surveys are averaging lower than satisfied then the test will fail and we need to make a better interface. Also we require the help staff to check through at least 90% of responses within 7 days of receiving feedback.

ID # 48 - Language Support Test

Description: This test will ensure that every language in Google translate can be played in our game, and that microtransactions can be made for every currency of the countries that we have a language for. This test will fail until we have support for all of these languages and currency. The prompts will also be checked by native speakers so they also must pass these tests.

ID # 49 - Game Learnability Test

Description: This test will ensure new players can go through all of the levels of the tutorial. This will also ensure that they can skip through the tutorial in different parts, or entirely. Also it will ensure that players can go back to the tutorial after completing it. All of these must fulfill to pass.

ID # 50 - User help Test

Description: This test will ensure that the user can view the digital manual and that they can press the question marks near any items that might need more explanation. If there are any items that we deem need further explanation, or if there are question marks that can’t be pressed this test will fail.

ID # 51- Colorblind Mode Test

Description: This test will ensure the user is able to enable and disable colorblind mind when it is selected. It will also ensure the proper color blind interfaces are being shown when chosen and not when otherwise.

ID # 52- Text to Speech Test

Description: This test will ensure the user is able to enable or disable text to speech mode at any time and that the proper speech occurs when doing so. In order to pass this test this mode must work in all languages we support and read everything on the screen to the user. This will also ensure when the user presses something the speech is said out loud and it will prompt the user to press again or choose another option. All of these must pass.

ID # 53 - Legal Agreement Test

Description: This test will ensure the player must fill out the terms of service agreement in order to continue playing the game. If they do not fill it out or disagree then the game cannot be played, if they can still play the test must fail. This also must ensure the players can actually play after agreeing to the TOS.

ID # 54 - Tutorial Test

Description: This test will ensure that the players are able to choose the game tutorial menu and actually play through it at any time. If they are unable to access the tutorial or any parts of it this will fail.

ID # 55 - Help Option Test

Description: This test will ensure that the game has help options throughout for the player to select. If certain help options are unavailable or not seen to any user at any time this test will fail. This menus also must give help and if they do not give proper help the test will fail as well.

ID # 56 - Logo Rights Test

Description: This test will ensure that we have the full right to use the RISK logo. In order to check this the test will make we have the license and that it is valid at the time of using it.

ID # 57 - Themes Test

Description: This test will ensure that we will have themes during all of the major holidays during the year in America. This will ensure we will have a theme for all of these holidays and that all of the users can see this theme after updating. If users cannot see the theme or we fail to make a theme 10 days before the holiday the test will fail.

ID # 58 - Overall Appearance Test

Description: This test will ensure our app have a modern look and that the users are satisfied with our app. There will be random surveys pushed out and if there is a rating below satisfied then the test will fail. Users also will be able to give feedback for any rating below somewhat satisfied so it will fail if they are unable to. Also help stuff must check 90% of the feedback within 7 days to pass as well.

ID # 59 - Internet Accessibility Test
Description: This test will ensure any player connected to a proper internet connection can play our online games. If a player on proper internet cannot play our online mode than this test will fail. This must also ensure those not connected to the internet cannot play online, or access the online modes.

ID # 60 - External APIs Test
Description: This test will ensure we are able to access to maps from the external API at all times. If at any time we fail to access a map this test will fail and the users will play on a preloaded map instead. This test will also ensure we are able to apply information to the map. It also must ensure we cannot grab invalid maps or places with too few streets.

ID # 61 - Distribution Services Test
Description: This test will ensure any user will access to the App Store, Google Play Store, and Steam are able to access and install our game. If any player is denied due to our fault this test will fail.

ID # 62 - Release Timeline Test
Description: This test will ensure our game is released within 22 months of our start. If we go over this at any time our test fails and so has our goal.

ID # 63 - Beta Release Timeline Test
Description: This test will ensure that the beta is released within 15 months after the project has begun development. If we pass this mark then the test will fail and so has our goal.

ID # 63 - Beta Release Timeline Test
Description: This test will ensure that the beta is released within 15 months after the project has begun development. If we pass this mark then the test will fail and so has our goal.

ID # 64 - Game Rights Test

Description: This test will ensure that we have reached a legal agreement with Hasbro before we can push out the game and use their likeness publically. This will ensure we have the trademark and can use their functionality and will check that we have valid licenses.

ID # 65 - Game Rating Test

Description: This test will ensure we get a rating by the ESRB and ensure our game fits the rating of E for everyone. If we cannot achieve this rating the test will fail and the game needs to be more friendly to all ages. Also, we must ensure that once we get a rating our rating is valid.

III.Design

1. Design Goals

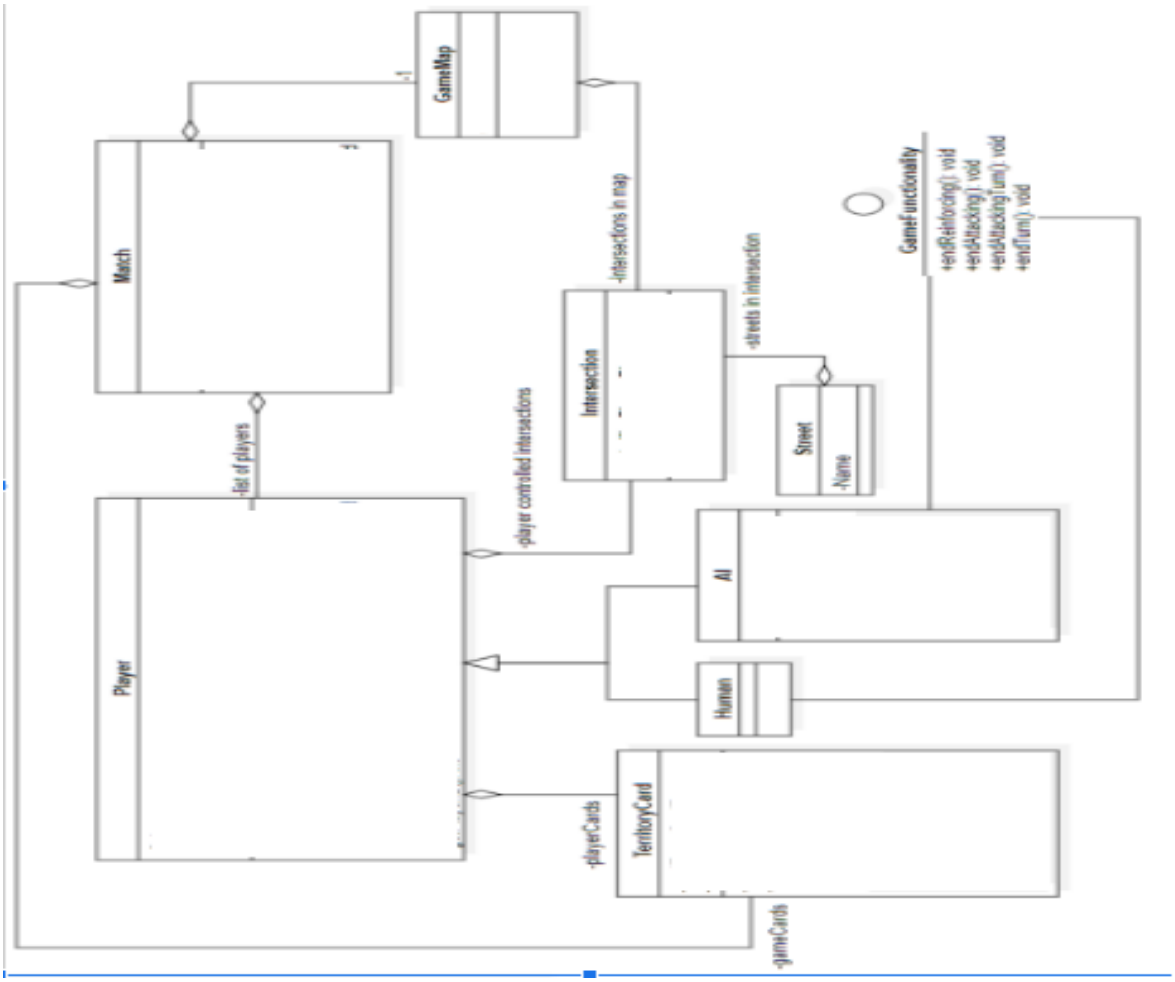
1. Proper functionality of the game
2. Proper implementation of *Casual*, *Competitive*, *Tournament*, and *Offline* game modes.
3. Proper user account creation setup
4. Robust graphical user interface
5. Consistently stable connection between players and the game servers.
6. Adequate storage capacity for each unique user account's information.

2. Current System Design

There is no pre-existing system

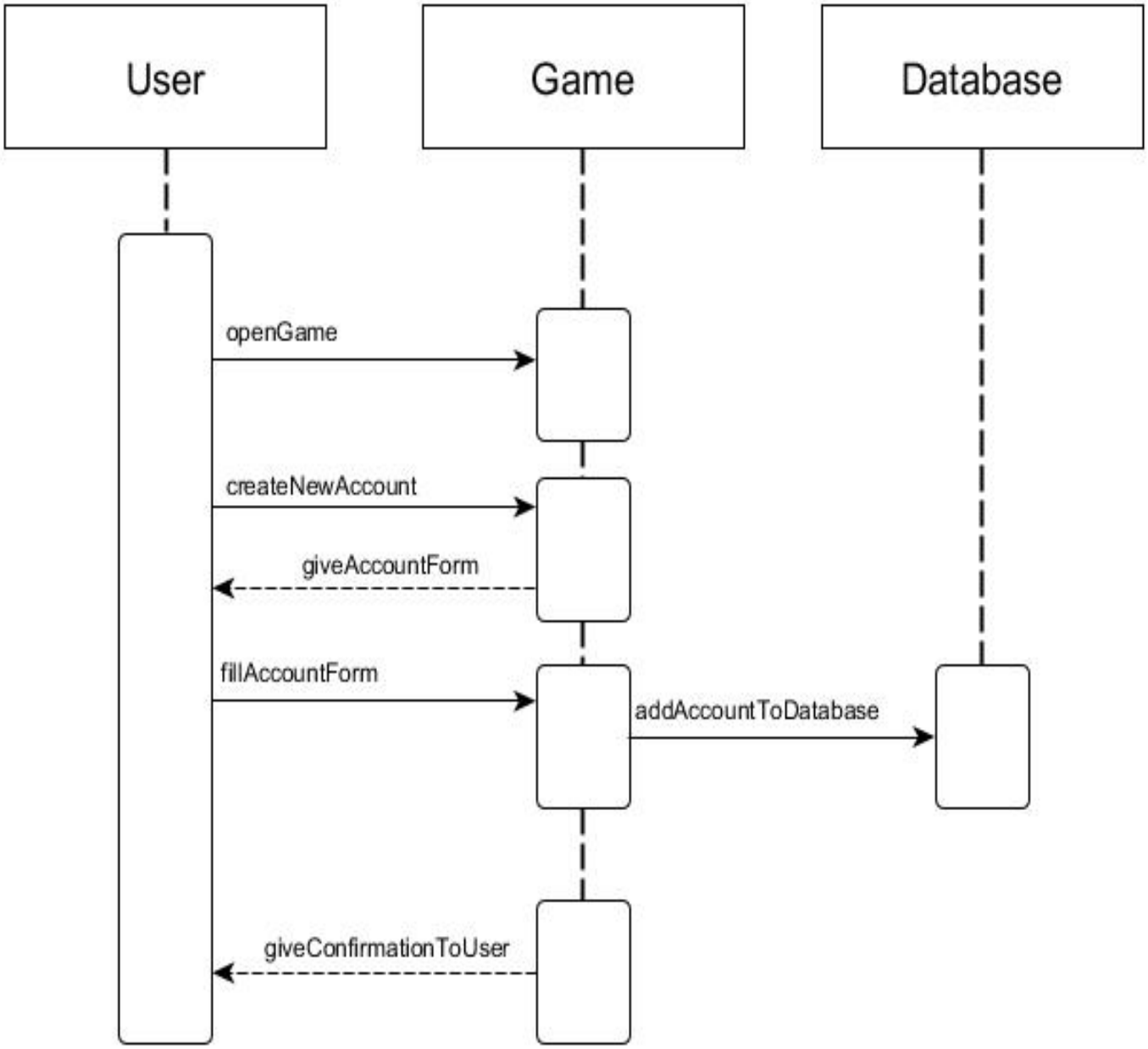
3. Proposed System Design

24a Initial System Analysis and Class Identification

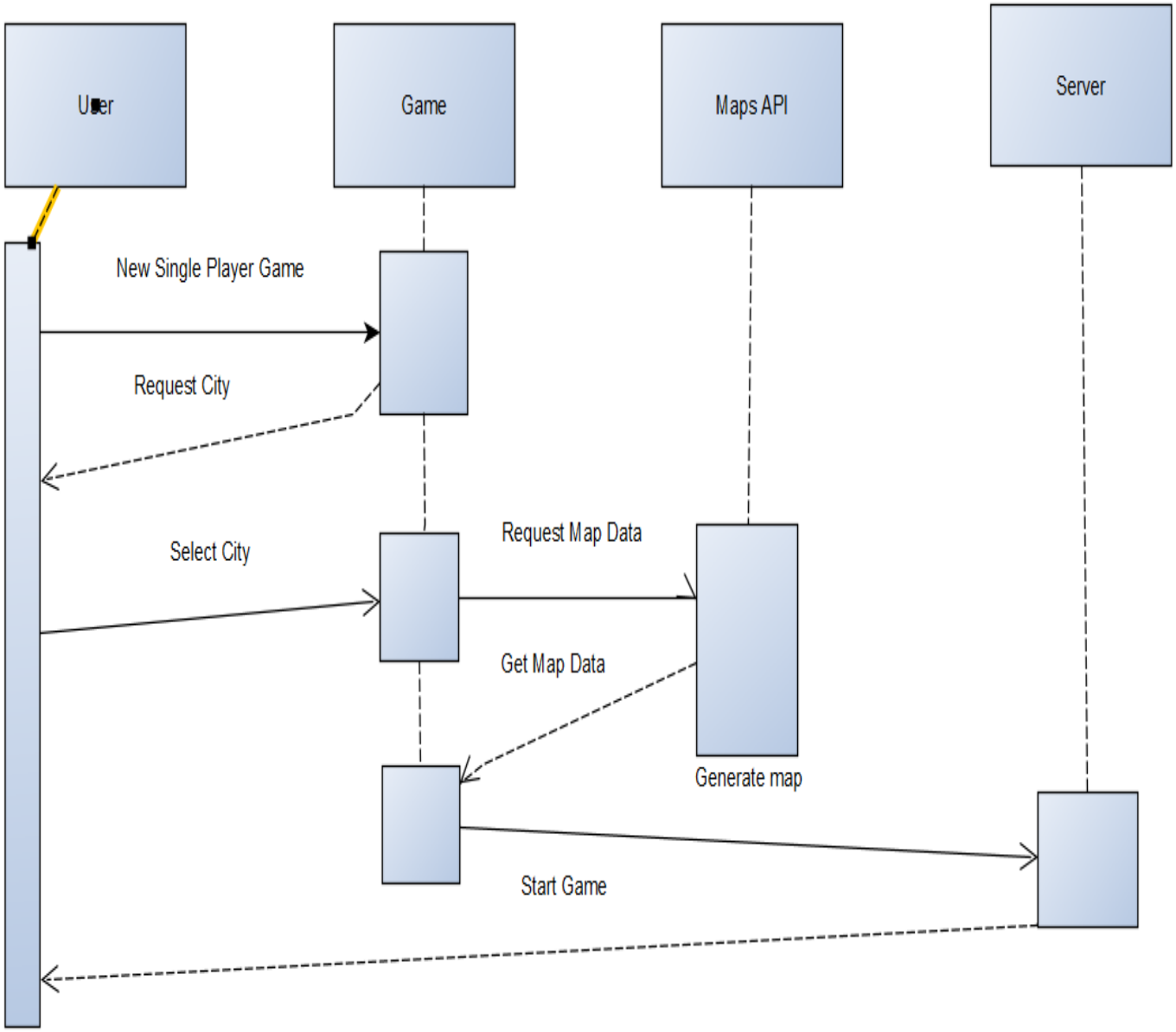


24b Dynamic Modeling of Use-Cases

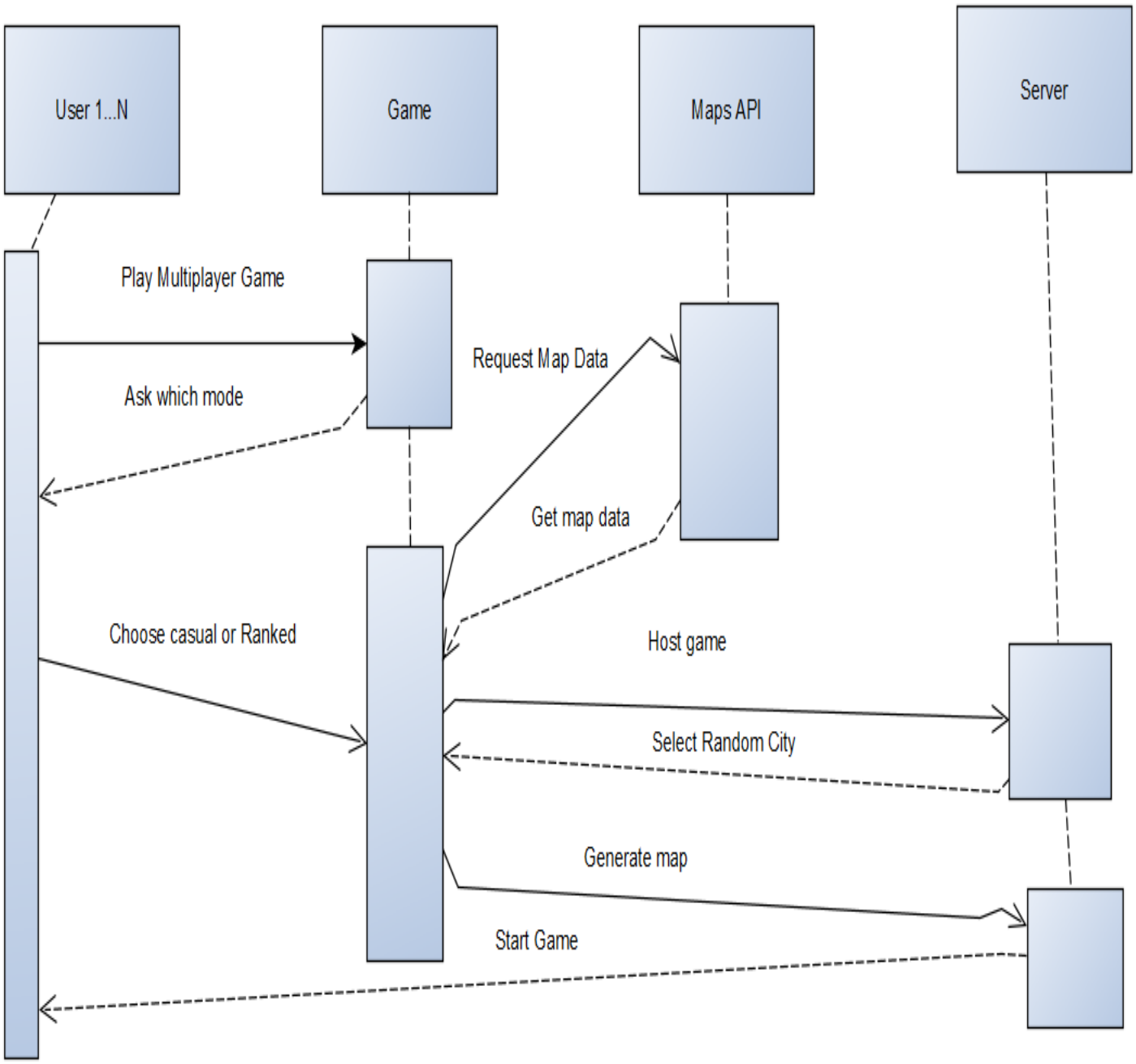
Account Creation



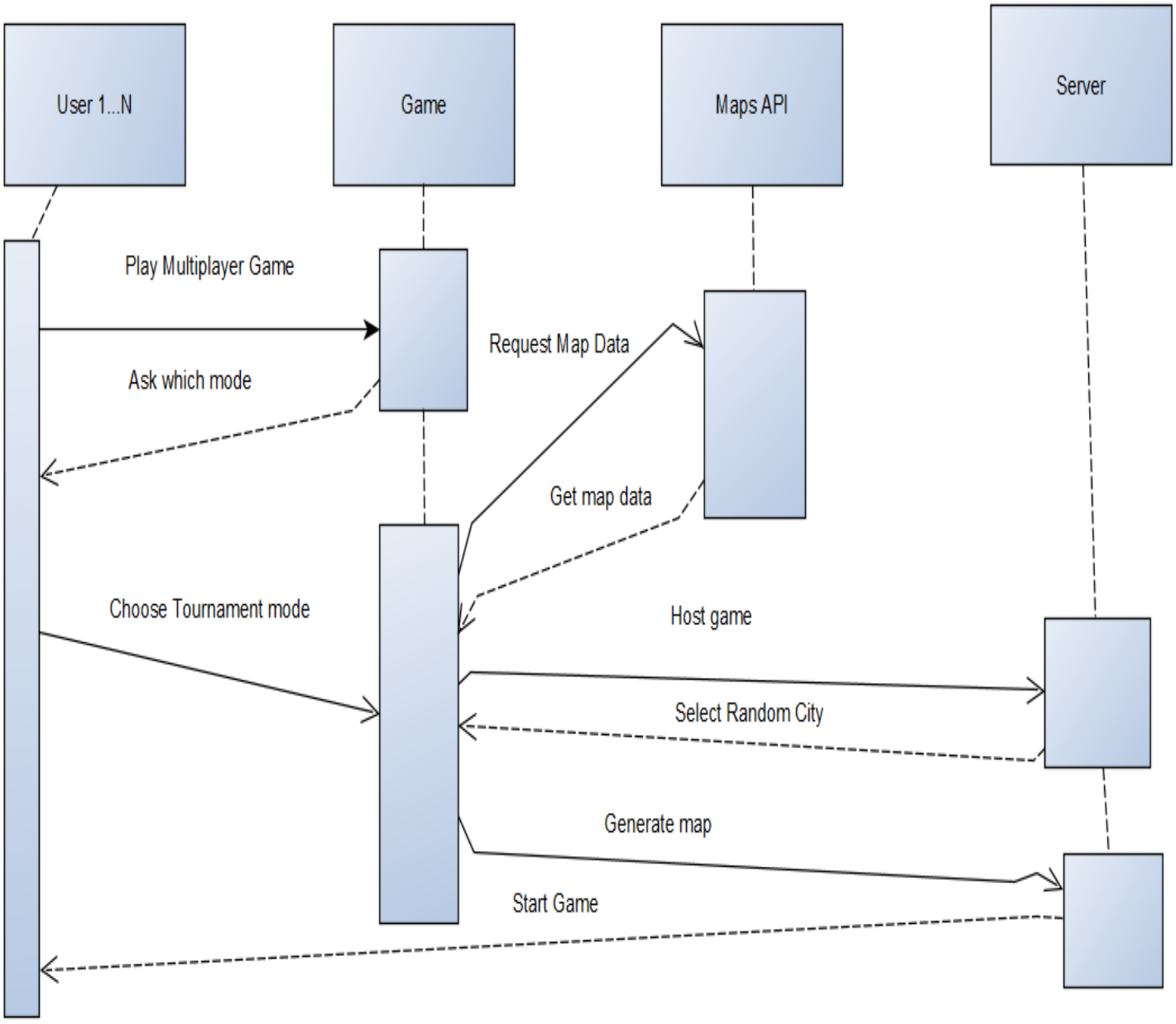
Single Player



MultiPlayer



Tournament mode

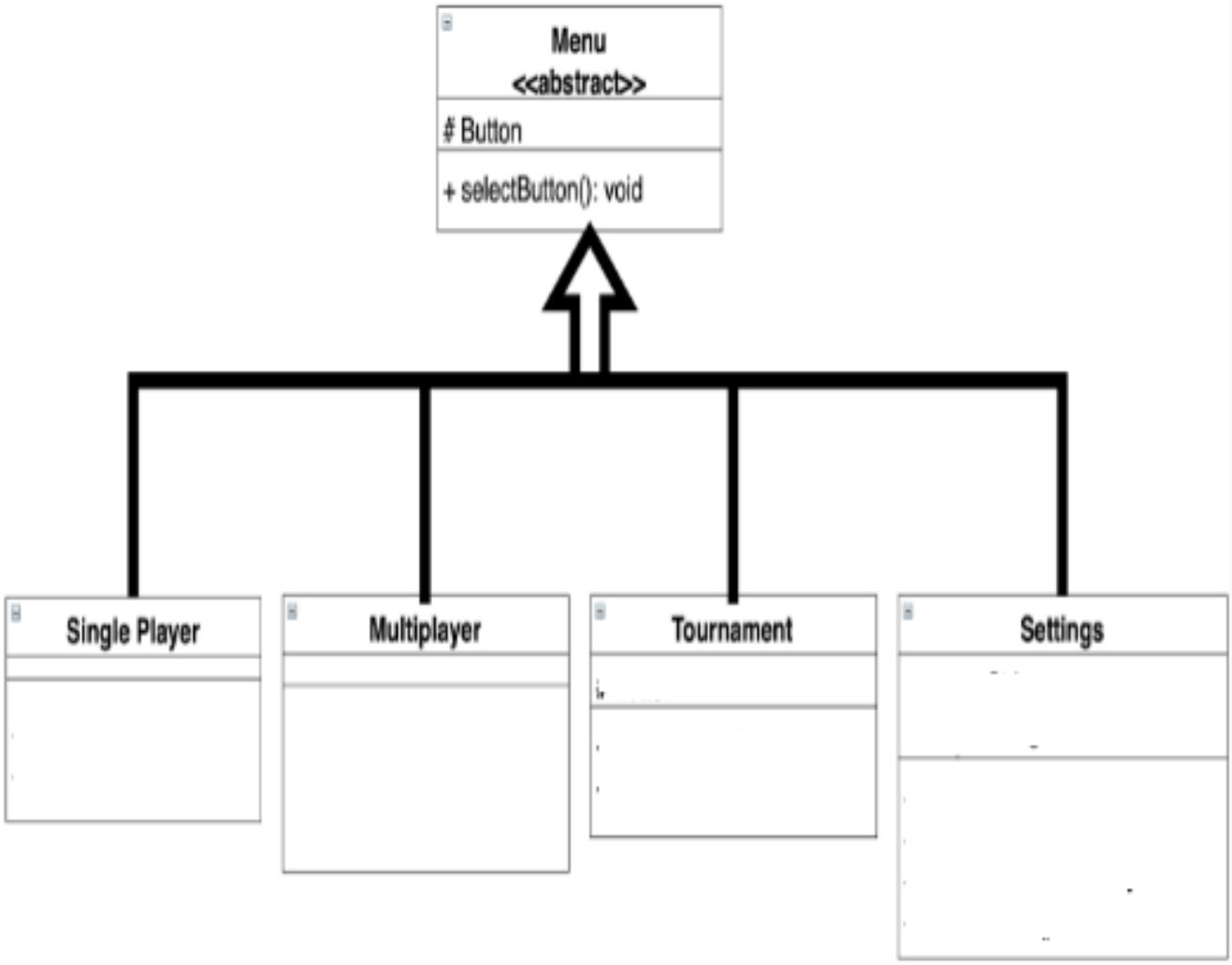


24c Proposed System Architecture

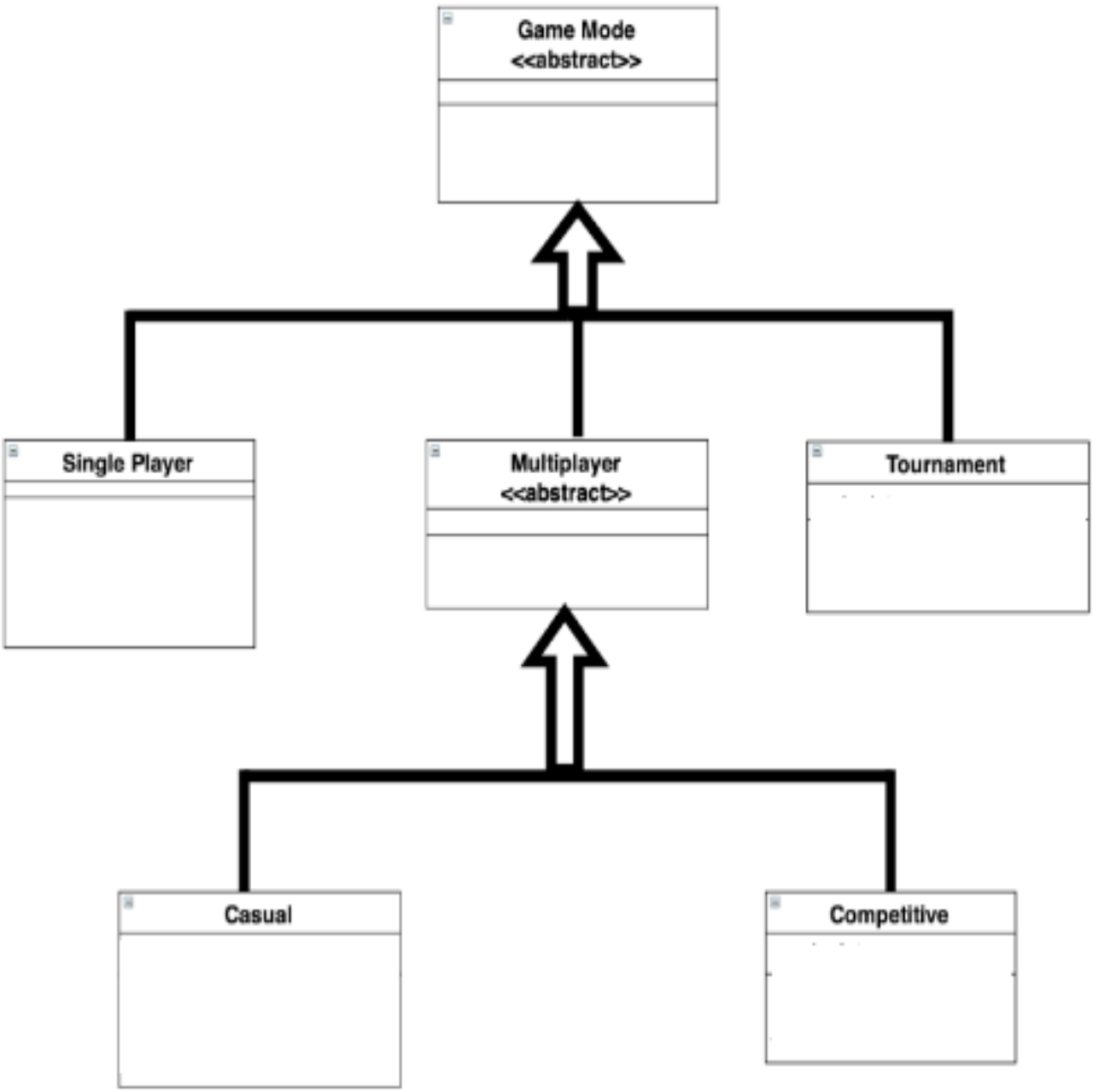
The proposed system will follow the Client Server architecture due to the many instances of players as clients, and the server in the relationship being the game servers.

24d Initial Subsystem Decomposition

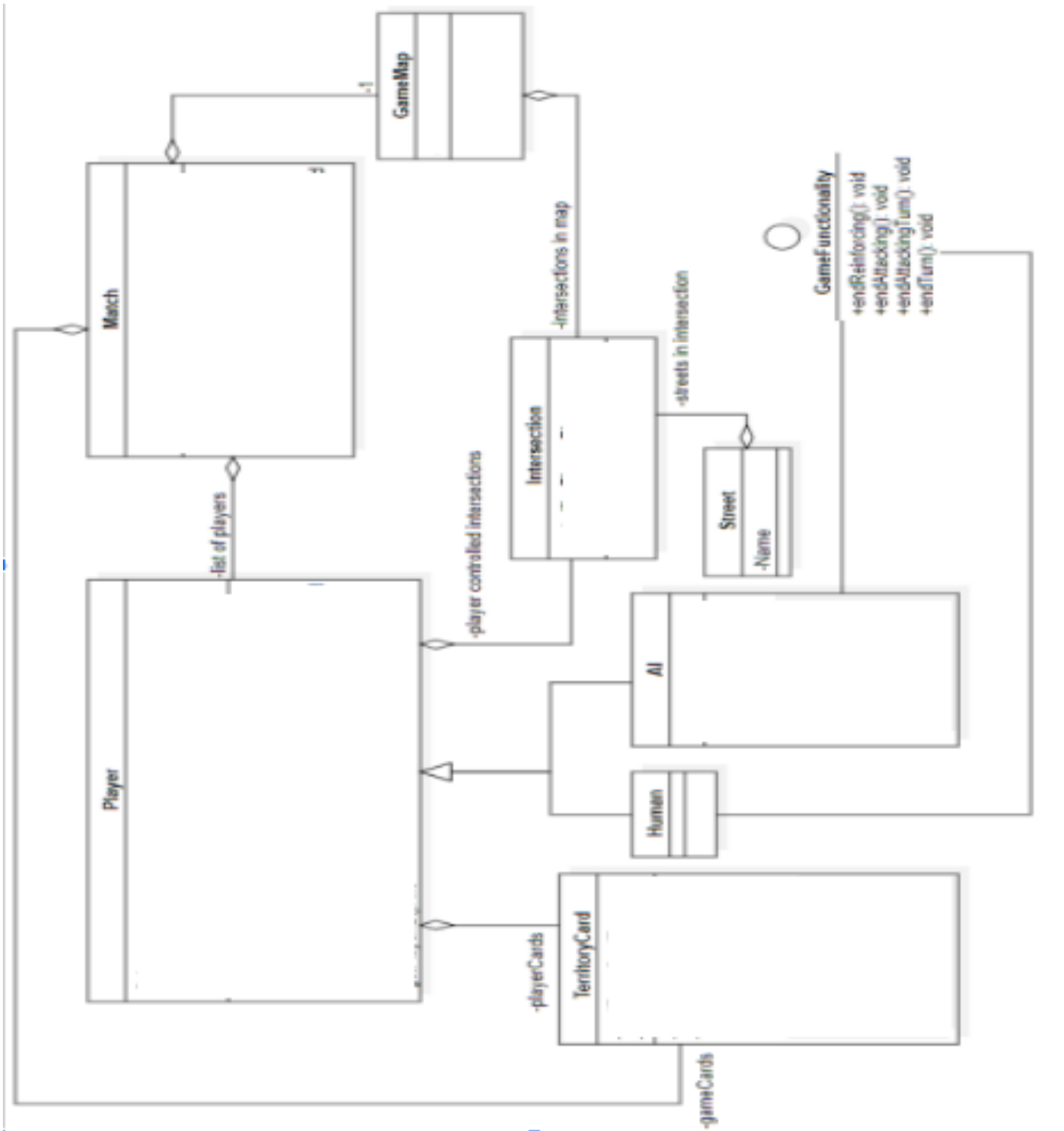
Main Menu Subsystem



Game Mode Subsystem



Game Functionality Subsystem

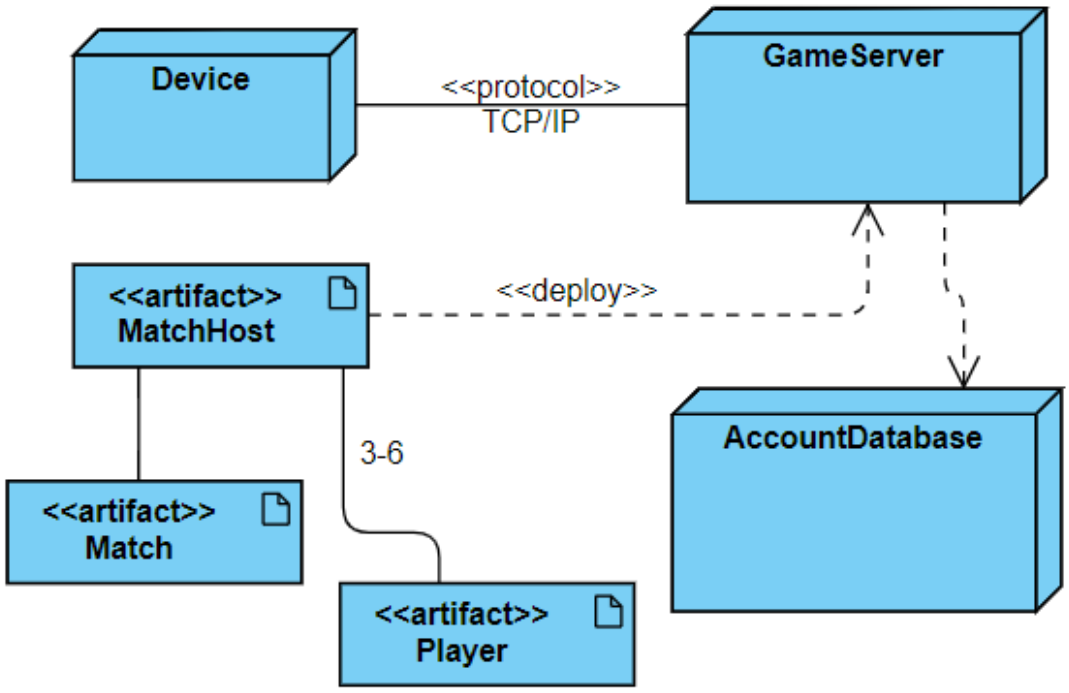


Subsystem Descriptions

Main Menu Subsystem - This subsystem encloses all of the main menu options a player can choose. It has single player, multiplayer, tournament, and settings, and they things the player can do to select these. All of these work together with tight cohesion.
Game Mode Subsystem - This subsystem encloses the game modes a player can possibly play from. This has single player, multiplayer with casual and competitive, and tournament mode. All of these work together with tight cohesion
Game Functionality Subsystem - This subsystem encloses all of the game functionality within it. It has classes for Players, Matches, Intersections, and TerritoryCards. All of these have to do with how the game runs and function so they are placed in a cohesive subsystem.

4. Additional Design Considerations

25a Hardware / Software Mapping



25b Persistent Data Management

Classes:

- UserAccount: object that will hold a player's account information such as account name, creation date, win/loss records as well as other needed information.

Inorder to ensure all the necessary data is never lost, it will be stored in a SQL database. This database will contain various kinds of tables which will store information such as account information, bug reports, rankings, each player's single-player records, multiplayer record and tournament records. In the case of a system shut down due to a game update or unforeseen circumstance, the system would successfully be able to restore all the data through the databases.

Here is a list of information which will be included in the tables mentioned above.

1. Account Information
    a. This table will contain a user's personal information such as their name, address, and credit card number.
2. Bug Reports
    a. This table will contain bugs reported by the players. The bug's name, date it was reported, and whether or not it was fixed would be included.
3. Singleplayer
    a. This table will include a player's win/loss, kills/deaths, fastest game, longest game, most areas in control, and win streak information.
4. Multiplayer/Tournament
    a. These tables will include each player's win/loss, kills/deaths, fastest game, longest game, most areas in control, win streak information, and number of tournaments won. In addition, for each player, it will also contain records of which other player did he/she kill the most, or got killed most by.

25c Access Control and Security

Inorder to minimize the risk of unauthorized access to the physical and logical systems of the game, the SQL database should be accessed with read-write operations. The player is the only who should have the ability to have access to their account information due to privacy.

25d Global Software Control

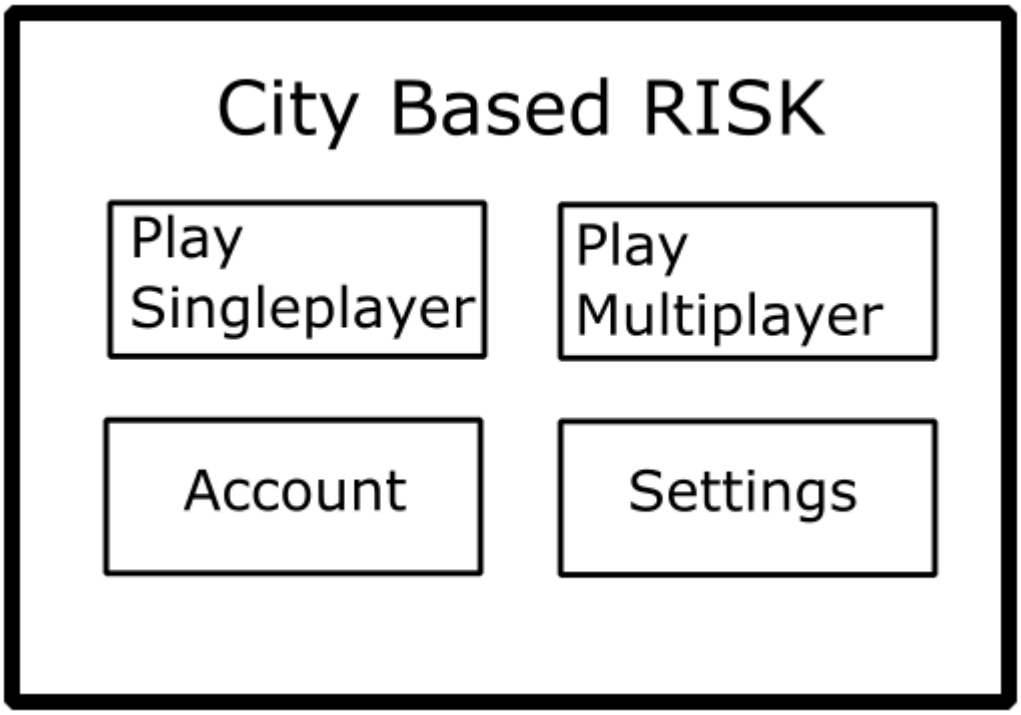
Not applicable.

25e Boundary Conditions

During any server maintenance, the players must be notified at-least 1 week before downtown begins. The players must also be notified once the server is up online again.

As to obtaining the users information during account creation we plan to offer text fields and buttons to enter that information. These objects will also be offered when logging in if the user already has an account and the entered data will be validated against data from the database to verify if the information is correct and if the user is to be signed in or not.

25f User Interface



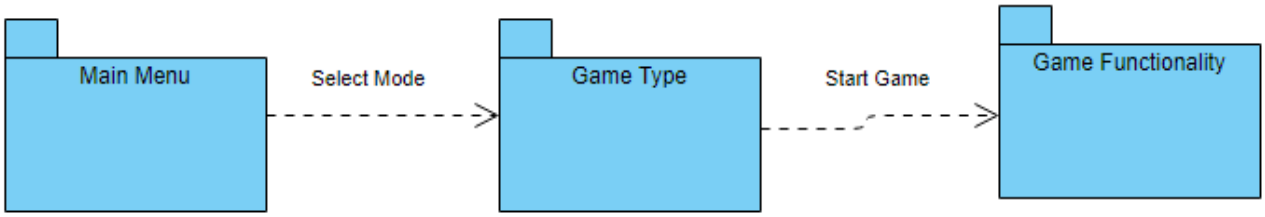
25g Application of Design Patterns

The Match class should follow the *Builder* design pattern.

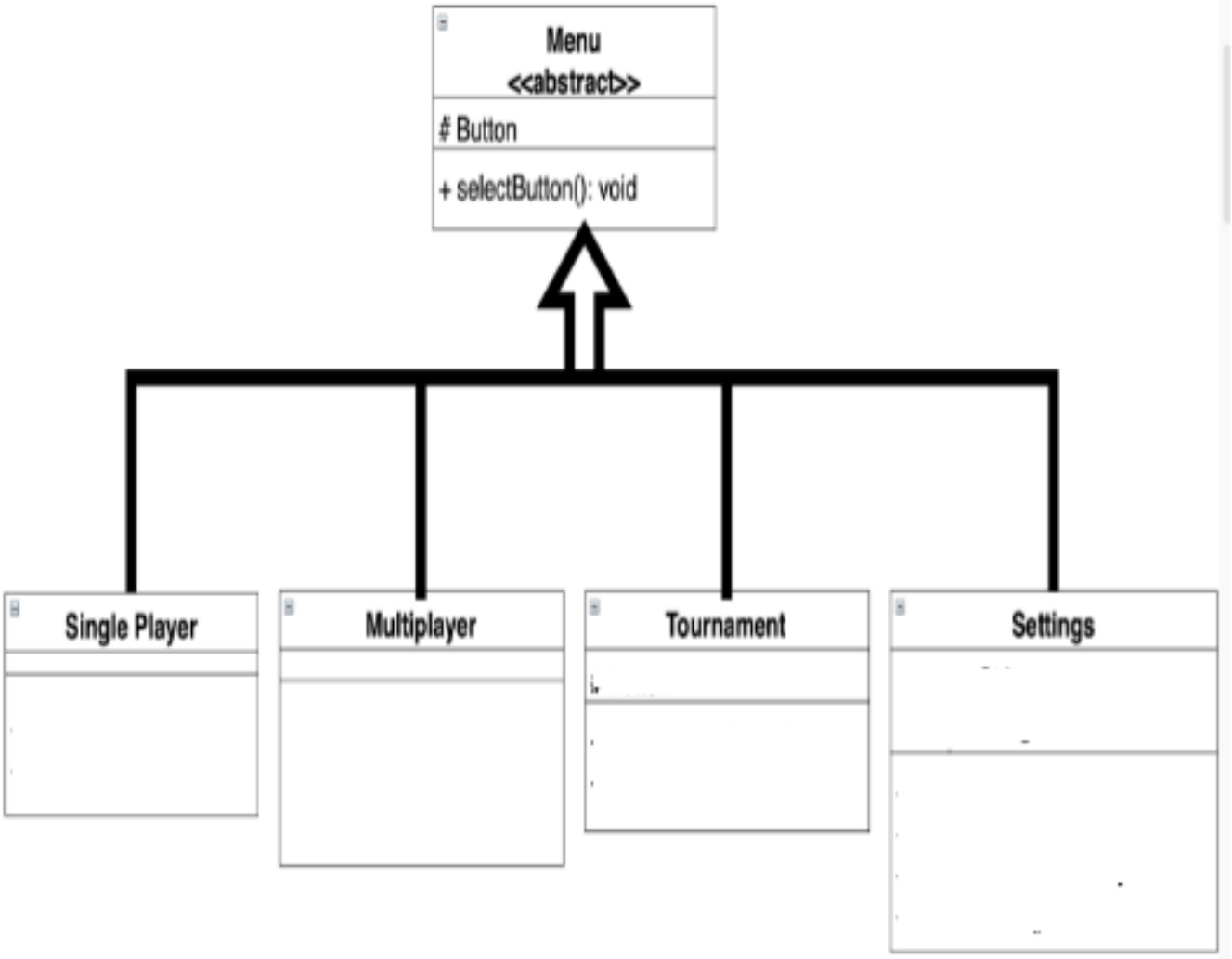
GameMap in each instance of a match should follow the Singleton design pattern, which is to say for each Match, there should only be one GameMap.

5. Final System Design

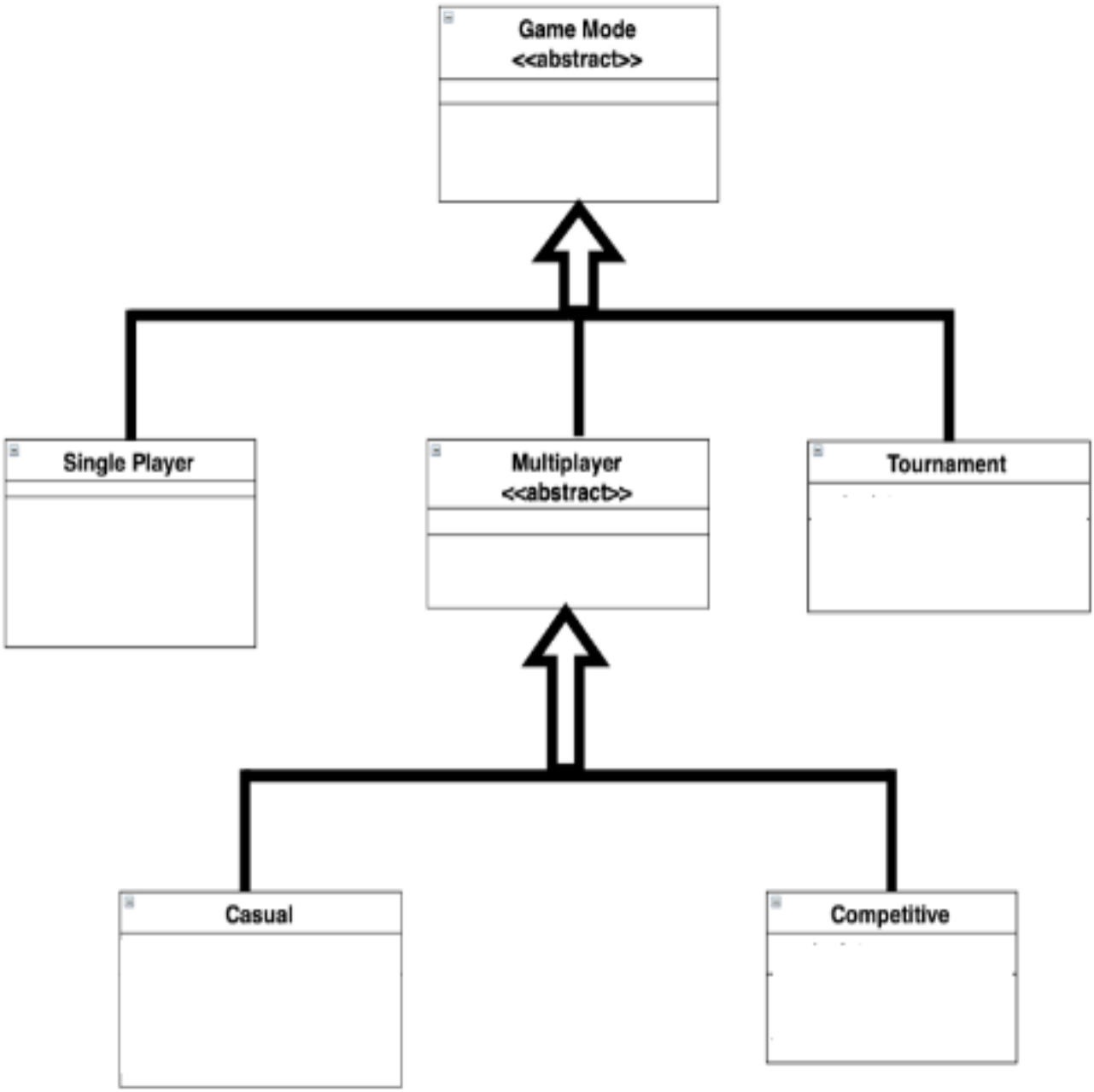
Overall Package Design



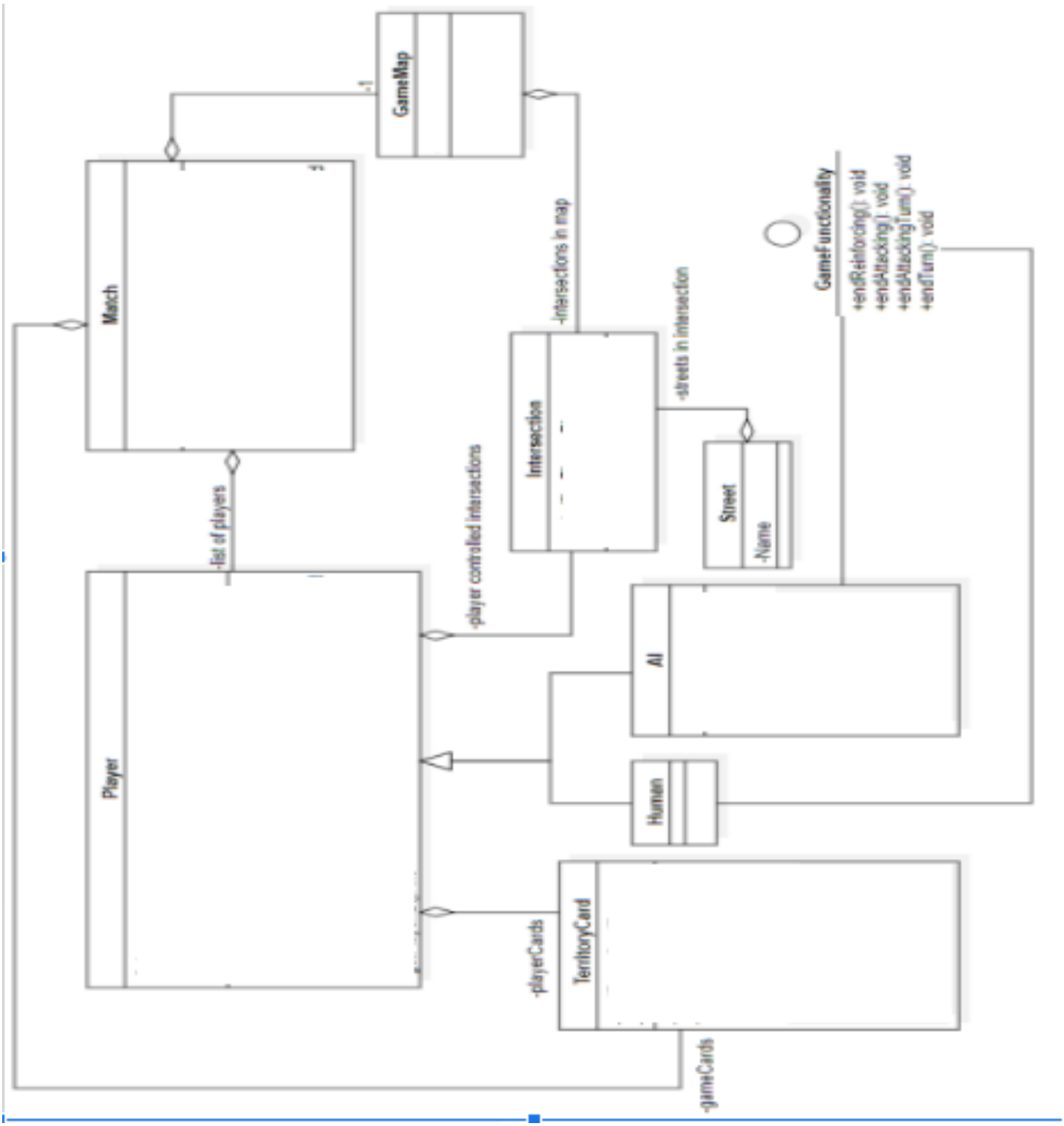
Main Menu Subsystem



Game Mode Subsystem

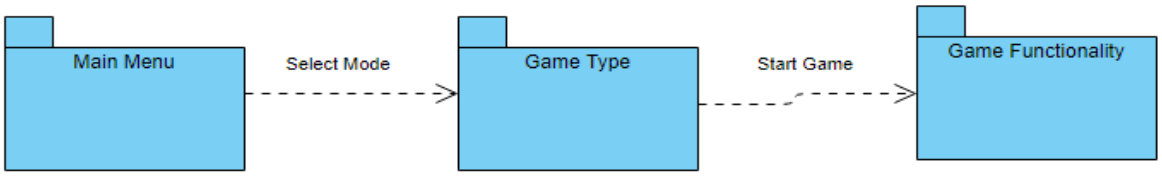


Game Functionality Subsystem

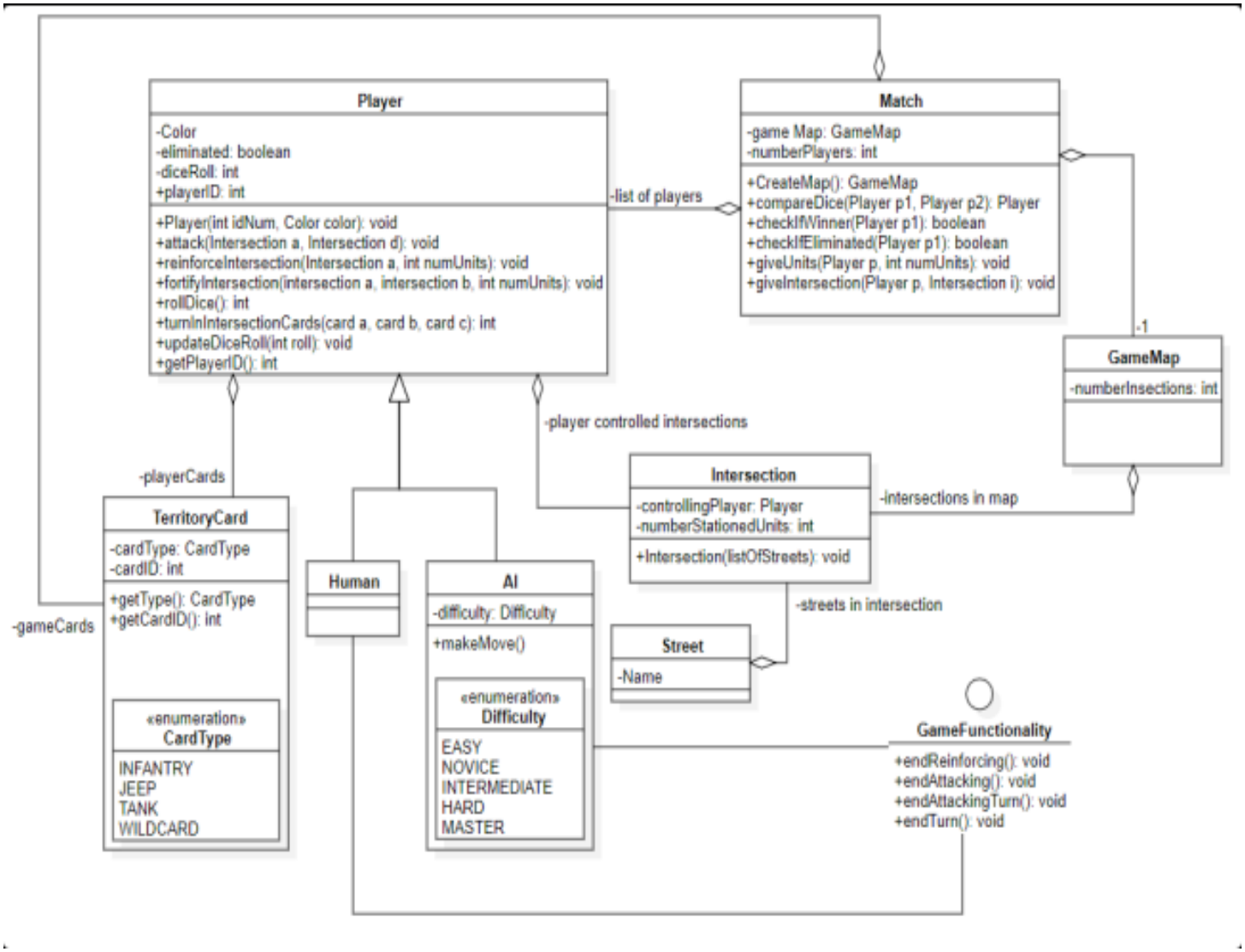


6. Object Design

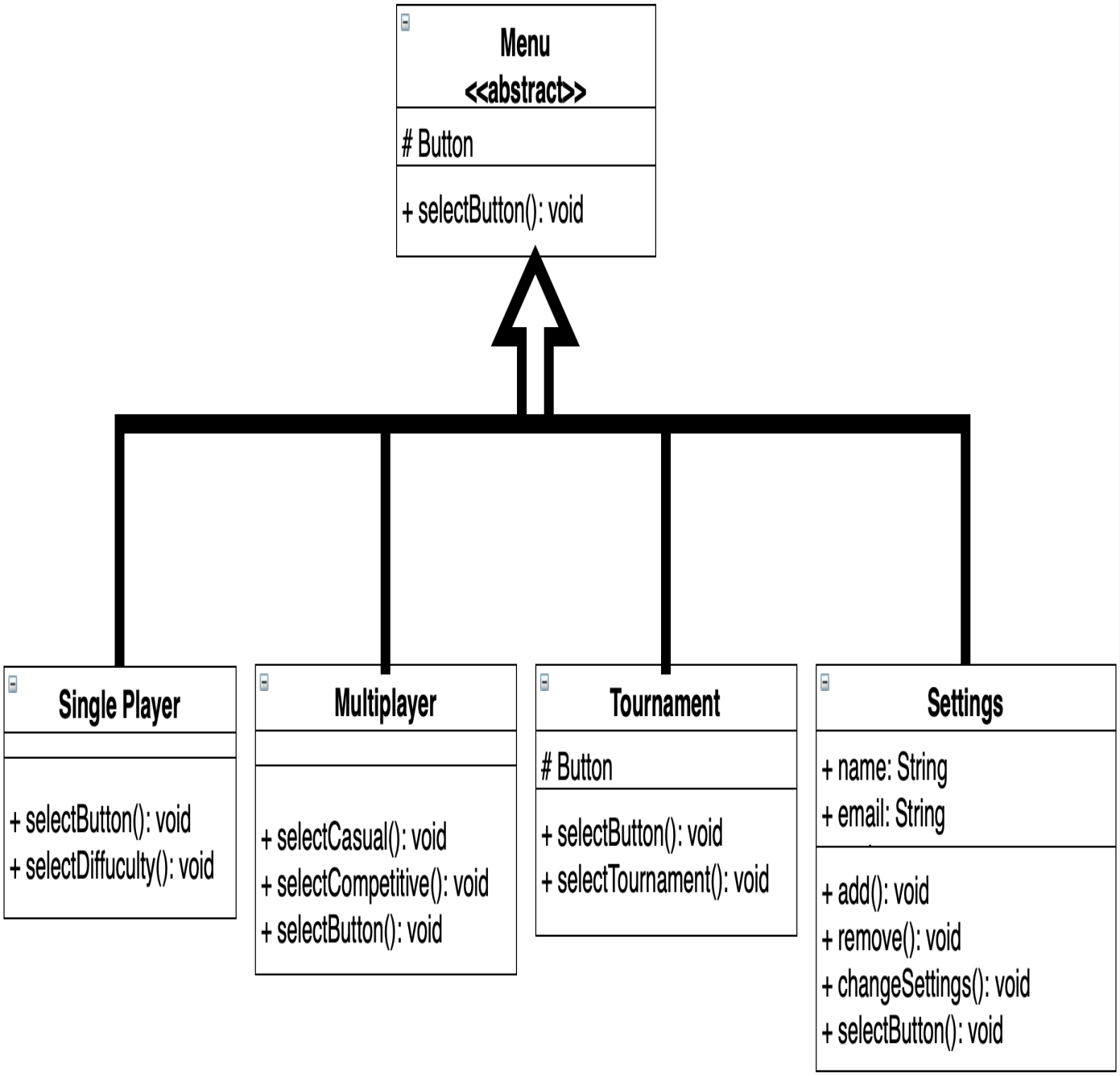
27a Packages



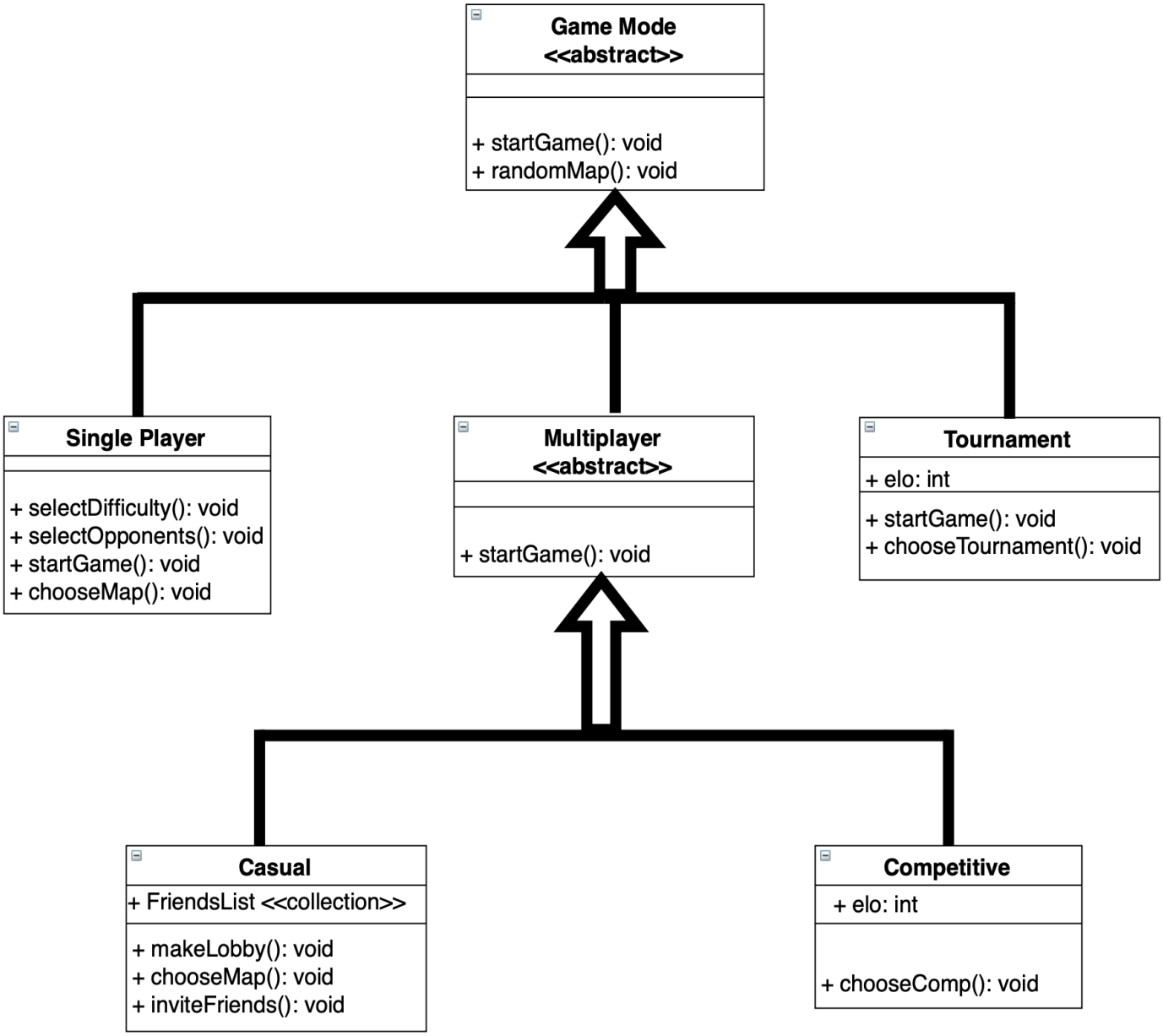
27b Subsystem I



27c Subsystem II



27d Subsystem III



IV. Project Issues

1. Open Issues

At this point, no issues have been raised concerning the system. Subject to change.

1. Application running on older devices
    a. Given the immersive scale of the game, older devices might come across issues where they might take a while loading the game or they might experience frame-rate issues.

2. Off-the-Shelf Solutions

The Google Maps API will be necessary for the Game Map object creation when the match starts.

29a Ready-Made Products

The Google Maps API is a readily available service that will be used with the map creation system to generate matches for the game. In addition, as mentioned earlier, a SQL database will be used inorder to manage the database.

29b Reusable Components

1. Google Maps API.

29c Products That Can Be Copied

There are no products that can be reused.

3. New Problems

30a Effects on the Current Environment

There are no notable effects on the current environment on behalf of the system.

30b Effects on the Installed Systems

For older devices and /or less powerful devices, the system may slow down performance of the device, and at the same time, the user's client may have slower performance.

30c Potential User Problems

The players of the game could encounter frustration as a result of multiple situations, including server crashes, not being able to create an account, or there not being enough other players to create matches.

30d Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

There are no anticipated physical limitations for the implementation of the physical product.

30e Follow-Up Problems

1. Server crashes
2. Not enough storage on databases.

4. Migration to the New Product

31a Requirements for Migration to the New Product

No migration procedure will be required as there is no existing system.

31b Data That Has to Be Modified or Translated for the New System

No data modification procedure will be required as there is no existing system.

5. Risks

1. Project going over budget.
2. Under-qualified developers.
3. Improper system infrastructure.
4. Similar products entering the market before this product's release.

6. Costs

The project will cost roughly $5,000,000 accounting for 12 months of development time and maintaining the system for at least 5 more years after that.

Front-End Developers:

The front-end developers will be responsible for designing the graphical user interface which the players will be interacting with. They are going to ensure whatever the user interacts with as user friendly as possible.

Back-End Developers:

The back-end developers will be responsible for dealing with all the complex systems that are going to be happening in the background. They are going to make sure that all the necessary features are implemented in the game, and are working appropriately according to the user interface.

QA Testers:

The QA testing team will be responsible for testing and debugging the game to ensure that the game quality is good enough for players to play and so that all bugs are fixed before full release.

DevOps Engineers:

The devOps engineers will be responsible for for creating an online infrastructure for the game, ensure the stability, and security of the online servers. They are going to ensure the cloud is protected against hackers and viruses.

7. Waiting Room

The following features will be considered for future releases:

1. Custom Maps
    a. The system should allow users to create their own maps, and be able to play in them.
2. Seamless Play.
    a. The system should allow users to play the game on one device, and be able to continue the rest of the game on a different device.
3. Cross-Platform
    a. The system should allow users to play against other users playing on any other device.
4. Sharing
    a. The system should allow users to share their in-game characters to their friends for a duration of their choosing.
5. Crews
    a. The system should allow users join another person's crew which will then give those players extra bonuses if their crew wins a game.
6. Replay
    a. The system should allow users to automatically record all games that they play during a week so that they can go back and look at it.
7. Mirroring

a. The system should allows users to mirror their game on a different device than what they are originally playing.

8. Ideas for Solutions

There is not any emphasis on a certain programming language for this project. The developers can decide amongst themselves on which programming language would be a better fit for their project. One thing the developers need to make sure of is to design code in such a way that there is high cohesion and low coupling which will lead to ease in code maintenance as well as ease in the integration of new features of the game. The developer team can also decide on an IDE of their choosing. As far as creating the game on mobile devices is concerned, it can either be doing using Android Studio for android devices, and Xcode for iOS devices or React Native which would work for both android and iOS devices. In addition, the developing team is allowed to choose any library or external framework for this project.

9. Project Retrospective

Throughout the course of this project, we made sure to have meetings where each subject of the meeting was discussed only as long as it was needed, be it short or long. Additionally, we held group calls online where we collaborated on finishing this report. Combining in person meetings and online meetings helped us combine strategy and implementation for the development project.

Our biggest challenge during the course of this project was testing and the diagrams phase. We, as a group had to ensure all the necessary testing was mentioned in this document in order to ensure the game is as fluid as possible once the game fully releases. We want to make sure the users do not run into any issues once the game releases in full scale which is why we created a beta program which will help us further help us better the game.

We also came across issues where during our initial submission of the report, we did not fully implement the scenario diagrams, which is why during the second phase of the report, it took us a while to figure those out since we not only had to include those, but also finish Part 2 of the report.

For consideration, we were assigned as a group of 4 students for this development project, but only 3 people contributed any actual work. We are tremendously disappointed with Patrick, as he has literally no work to show for the project. And as

a group, we have made it clear to each other if anyone needed help, they are always welcome to ask anyone else in the group, if the person they were asking was able to provide it. With that being said, Patrick has stated during meetings he had issues with completing his assigned parts, but never asked for anyone's help.

V. Glossary

Refer to 1.7.a for all important defined terms

VI.References / Bibliography

[1] M. Fowler, UML Distilled, Third Edition, Boston: Pearson Education, 2004

VII. Index