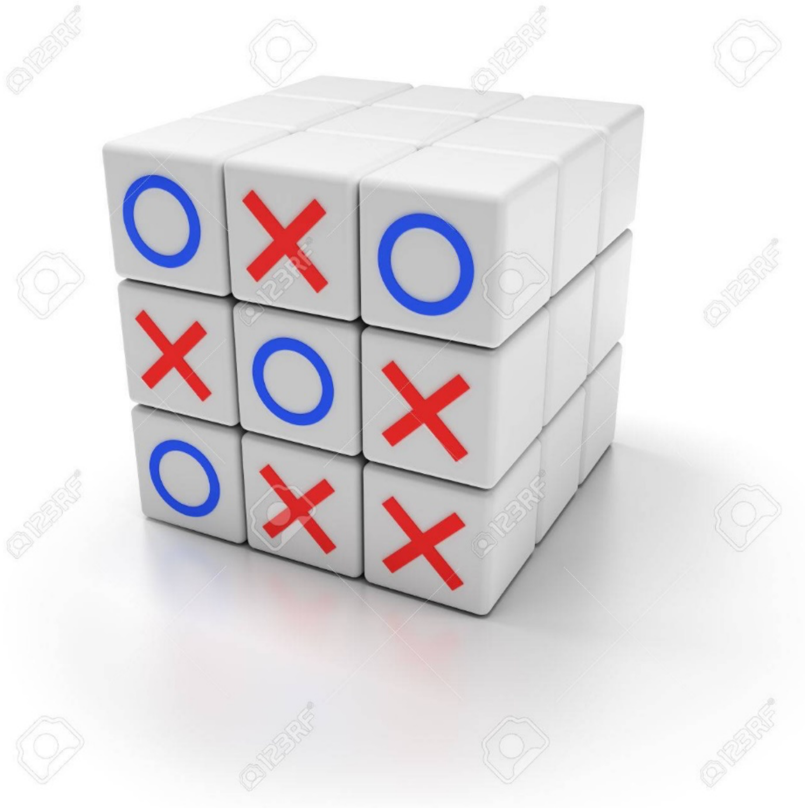
Score : 96/100

3D Tic-Tac-Toe Project Report



Prepared by
Nick Toledo, Omar Flores,
Stephen Lambert, Yash Shah
at the
University of Illinois Chicago

December 2020

Table of Contents

List of Figures

List of Tables

I Project Description

1 Project Overview

3D Tic Tac Toe is a multiplayer game available on a PC and a mobile device, similar to the original tic tac toe game. The game board is designed on a 3x3 cube, each player can rotate the board in any direction, place their characters, and see a live update of the game. The goal of the game is to connect three characters on any side of the board. The game is designed to be fun for every player and create income for the developers.

2 The Purpose of the Project

The purpose of this project is to extend a single player game into a multiplayer game that has multiple dimensions. We plan to create a user-friendly application that gives live updates of the gameboard and detects the top three players that connect their characters.

2a The User Business or Background of the Project Effort

Our main client's business is the gaming industry. The gaming industry allows for a lot of creativity and new development projects. This allows us to turn a two-dimensional game into a three-dimensional game that we are interested in developing but could also be interesting to the gaming community.

Our plan is to produce a new multi-dimensional tic tac toe game that is significantly different from the original, with live updates of the game board. We were inspired to create this game, because we all enjoy playing video games, and thought it would be challenging to create a new game, that is significantly different from the original.

2b Goals of the Project

The goal of this project is to make a new multiplayer and multi-dimensional game that is significantly different from the original, and that is profitable. The game aims to be fun and enjoyable for anyone over the age 3 and must be able to attract new customers and keep current customers. The game must be profitable, covering the cost of development, and future improvements.

2c Measurement

The enjoyment of this game will be measured using different methods, for example, customer reviews, number of downloads, and user activity. It is important to gather this information, because then the developers can determine the success, or failure of the product, and find ways to improve it.

Customer reviews will help improve the game, because it shows the developers how the users feel about the product, and if they are happy with the game. This will help the developers determine whether or not users will keep playing the game, and help developers determine other games that users would enjoy playing. The user activity measurement will determine how many users are playing the game weekly or monthly, and if they are most likely to continue playing the game. This will help developers find ways to keep new and existing users, because not every player will submit reviews.

3 The Scope of the Work

The main purpose of this game would be to entertain people on a regular basis and make an income from this game.

3a The Current Situation

Tic-Tac-Toe is a classic game that can be traced back to ancient Egypt. This simple yet timeless game has managed to survive the test of time thanks to it is easy to setup and play nature. All somebody needs to play the original game is pen, paper, and a friend. However, now that we are entering a more advanced age many people have moved away from traditional forms of entertainment in favor digital entertainment. One of which is mobile games, according to an article by Medium, almost a third of the world's population will play a mobile game by the end of 2019.

As our entertainment interests grow more advanced, so should our entertainment. For this reason, we want to take advantage of the growing mobile game industry and bring back the classic game Tic-Tac-Toe as a more modern update, 3D Tic-Tac-Toe.

Although the original required only pen, paper, and a friend to play, 3D Tic-Tac-Toe will only require a mobile phone or pc running our product which will find other players to play with. Most, if not a majority of the population has a smart phone, so we will be hitting a very large demographic. No longer will people need to scrounge for pen and paper to play Tic-Tac-Toe with their friends.

3b The Context of the Work

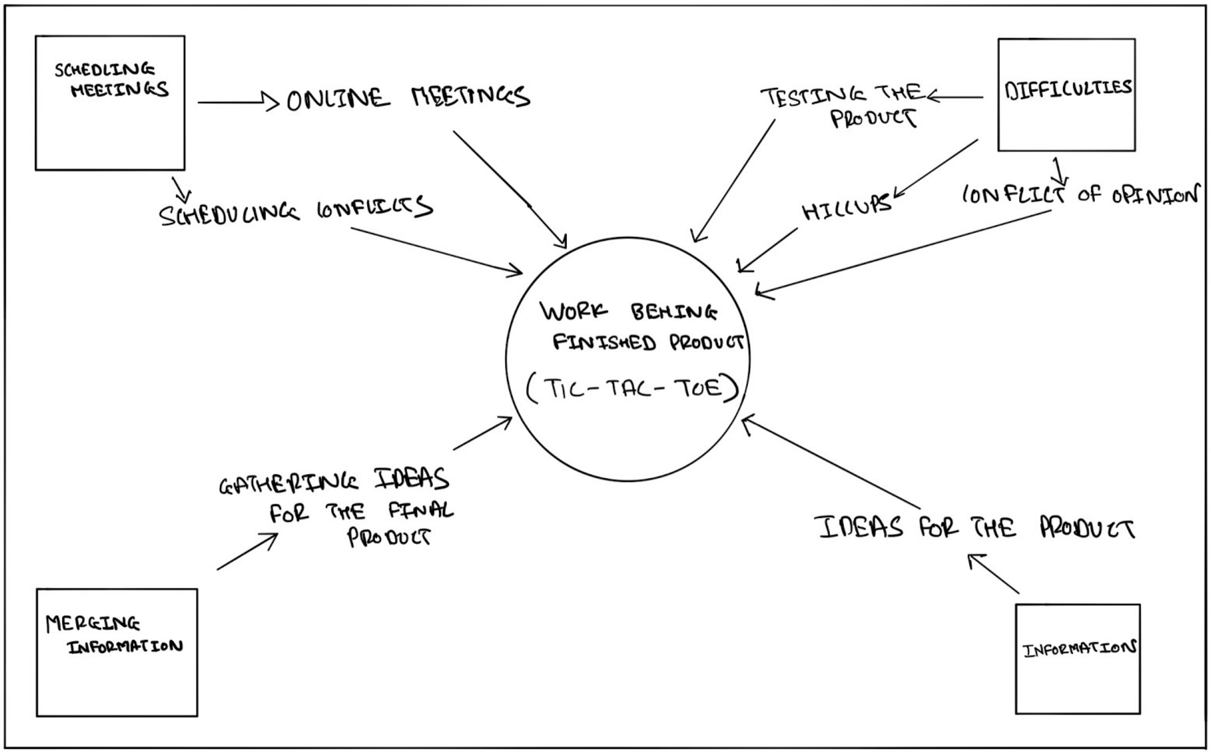


Figure 1 - Work Context Diagram

3c Work Partitioning

Event Name	Input /Output	Summary
Ideas	Getting Ideas of peers (input)	Gathering all the ideas
Working on product	Applying all the ideas for the product (output)	Testing all ideas for the development of the final product
Problems in the development phase	dealing with opinion conflicts and issues (input)	Solving all the issues before the release of the final product

Table 1 - Work Partitioning

3d Competing Products

When we think of the direct competitors of the current product, there are no other products like this. But the original Tic tac Toe will be the direct competitor to this product. The only shortcoming of this product will be to understand the entire game due to its complicated working methodology.

4 The Scope of the Product

User's will be able to find and install our product either through the App/Play store for both Android and IOS as well as on Steam for PCs. The product will allow for account creation as well as linking pre-existing accounts from Steam, Play Games, or Game Center. Upon logging in users will be able to choose from one of the pre-existing game pieces to use in game.

Afterwards user's will be able to start the game by choosing which game mode they would like to play, which will put the user in a queue to find an opponent. As user's continue playing, they will level up and gain experience points which can be used to unlock in game rewards.

4a Scenario Diagram(s)

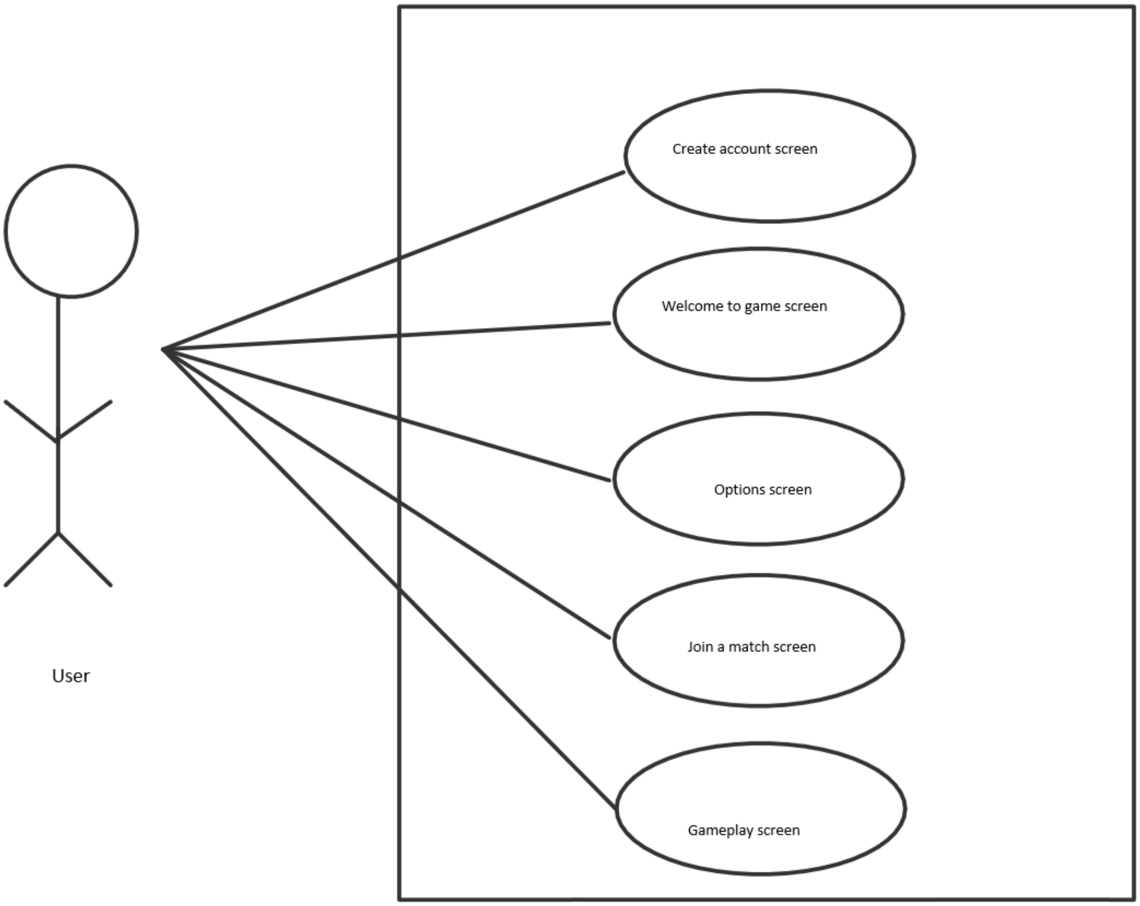


Figure 2 - User Scenario Diagram

4b Product Scenario List

PC Game

1. Login account screen – Screen for logging in.
2. Welcome screen – Welcome screen which is also the main menu.
3. Options screen – Screen for changing in-game options.
4. Join a match screen – Screen for joining an online match.
5. Gameplay screen. – Screen for showing gameplay.

Mobile Game

6. Create/Login account screen - Screen for logging in on mobile
7. Welcome screen – Mobile welcome screen which is also the main menu.
8. Options screen - Screen for changing in-game options on mobile devices.
9. Join a match screen - Screen for joining an online match in mobile device.
10. Gameplay screen - Screen for showing mobile gameplay.

4c Individual Product Scenarios

Users will start the application on the respective device. Once the application starts, the user will be prompted to create an account if the user doesn't have one already or sign in using a pre-existing account. When the user signs in, their online profile will display on the screen.

Users will have the ability to mess with the game settings such as audio, key configurations, and other options that lead to a better user experience. Users will have the opportunity to join an online match or play with a friend. Both options will lead to the game populating.

5 Stakeholders

5a The Client

We are the client. The goal is to make a product that will sell. Stakeholders/investors are clients too.

5b The Customer

The customer will be anyone who is into strategic games and those seeking a tougher challenge than the traditional Tic-Tac-Toe game.

5c Hands-On Users of the Product

Our main hands-on users for the product are going to be gamers. Specifically, gamers that like strategic games/games in general. However, it doesn't necessarily mean that it's just limited to gamers as the product will be accessible by anyone who can operate a computer. There will be no discrimination within the users. Users can be any gender with the age range being open.

5d Maintenance Users and Service Technicians

Server administrators will be needed to make sure the servers are working properly so that there are no issues when it comes to users playing. Game developers will be needed to improve the user experience such as the visuals, game animations, etc. Software developers will be needed to help keep the game up to date and performing with no issues/bugs.

5e Other Stakeholders

Tester

Testers will have a small stake in the whole development of the game. Testing is important for companies who are on the verge of releasing a new product. As a creator of a product, proper testing must be done so there's no bugs or issues come release date. The knowledge needed by the testers will be the logical aspect of the game.

Testers must know the rules of the game and the proper outputs of different moves. Aside from logic, testers must also inspect the visual aspect of the game. Do all components on the screen give a proper response, the transition from screens, etc. Any conflict between stakeholders who have an interest in the same knowledge will be handled by a meeting with other stakeholders and the clients.

Marketing Experts

Marketing experts will have a small stake in the whole development of the game. Marketing is important for companies who are on the verge of releasing a new product. Exposure will lead to success for products. The knowledge needed by marketing experts is our target audience. Based on our target audience, the marketing experts will offer suggestions for improvements, plan and implement social media or other marketing campaigns to help lure that audience.

Aside from our target audience, marketing experts will be able to gather information on certain groups of people that aren't being attracted by the product and offer suggestions on how to attract those groups. Any conflict between stakeholders who have an interest in the same knowledge will be handled by a meeting with other stakeholders and the clients.

Business Analysts

Business analysts will have a small stake. Business analysts will help facilitate solutions for stakeholders. They are responsible for bridging the gap between IT and the business using data analytics to assess processes, determine requirements and deliver data-driven recommendations and reports to executives and stakeholders.

Growth is what every company looks for and business analysts will be there to advance the company's future. Any conflict between stakeholders who have an interest in the same knowledge will be handled by a meeting with other stakeholders and the clients.

5f User Participation

A couple of months before the product releases, there will be a beta. During this time, 100 users will be selected to participate in the beta. Selected users will be allowed to interact with the product and provide user feedback. Through this, we hope to address any issues or suggestions to enhance the user experience.

5g Priorities Assigned to Users

Key Users - Gamers/Younger Audience

Gamers will be our key users. Preferably the younger gamers. Typically, this audience is already exposed to a variety of different games and won't mind adding another game to the collection. After marketing the product, it should be easier to attract this group of users.

Secondary Users - Strategic enthusiasts/Older Adults

Given that the product is based on the traditional Tic-Tac-Toe game, we expect that anybody who loves strategic games will come to appreciate a challenge. Doesn't necessarily have to be a hard core gamer. Older adults tend to have more time and are known to play more strategic games in their free time.

Unimportant Users - Children

Given that the product is a more complex game than the traditional Tic-Tac-Toe game, we don't expect infant children to grasp a full understanding of the product. Therefore, infants won't be much of a focus compared to our key users and secondary users.

.

6 Mandated Constraints

6a Solution Constraints

Description: Product shall run on average mobile specs.

Rationale: The client will not pay for a new mobile device to play 3D-Tic Tac Toe.

Fit Criterion: Mobile version of the game shall be playable on most if not all modern phones.

Description: Product shall run on both Android and IOS.

Rationale: The client will not switch phone OS just to play 3D Tic-Tac-Toe

Fit Criterion: Game will be available to download from both the Apple App Store, and the Android Play Store.

6b Implementation Environment of the Current System

Our product will be available for both Android and IOS devices through their respective app stores as well as for PCs through Steam. The 3D graphics of the game would be designed using the cross-platform game engine Unity with C# as the scripting language. Using a Unit will allow us to design our product for both mobile and PC operating systems using most if not all of the same code base. This will help reduce the cost of development by avoiding having to develop our product for IOS, Android, and PCs natively.

6c Partner or Collaborative Applications

The mobile version of our application will allow for users to log and save their data using both Google and Apple accounts. Likewise, PC users will also be able to log in using corresponding online retailers such as Steam.

6d Off-the-Shelf Software

The PC version of the application will be taking advantage of the already existing online network for online distributors such as steam. Doing so users will be able to log in with their online distributor credentials and access their already existing friends list as well as gain achievements towards their account while using our application.

Likewise, the mobile version of the game will take advantage of Play Games on Android, and Game Center on IOS to provide user's access to their friends lists and achievements while using the application

6e Anticipated Workplace Environment

The mobile version could be used in both quiet and loud environments; it must provide both audio and visual cues.

The application will be aimed towards all age demographics, it must be easy enough to be casually played by an older audience as well as children

6f Schedule Constraints

Game must be released prior to the next major Android and IOS updates to avoid the application from becoming obsolete during production

6g Budget Constraints

For the development of our product, we expect our budget to stay within our budget of $400,000. Most of our budget will go towards the initial development of our product, as well as future product support. However, we will also need to budget for app testing to ensure that we are not releasing a buggy and faulty product into the market which could give our company a bad reputation. We will also need to pay for marketing to ensure we gain a big enough audience for our product.

7 Naming Conventions and Definitions

7a Definitions of Key Terms

PC: Personal computer, referring to both laptop and desktop computers.

IOS: iPhone Operating System used for mobile Apple devices.

Android: An open source mobile operating system, used by most mobile manufacturers.

Steam: A cloud boused game library which allows consumers to buy and download pc games using their steam account.

Android Play Store: An Android app equivalent to the Apple app store for Iphones.

7b UML and Other Notation Used in This Document

This report adheres to the Version 2.0 OMG UML standard which is detailed by Fowler in [1].

7c Data Dictionary for Any Included Models

User account info will consist of

- Username
- Email
- Password
- Possible linked third party account (Steam/Apple/Google)

Games will have the following data

- Record of icons for each player.
- State of the board.
- Score for each player.

8 Relevant Facts and Assumptions

8a Facts

-Possible rating for our product would be E for everyone.

-Android OS is open source, and in more mobile devices.

-IOS and Play Store have requirements to be added to respective stores.

8b Assumptions

-User's should be familiar with using product platforms such as phones, tablets, and Pcs.

-Assuming mobile players will either be using IOS or Android.

-Players are more likely to play on mobile devices as opposed to PCs.

II Requirements

9 Product Use Cases

9a Use Case Diagrams

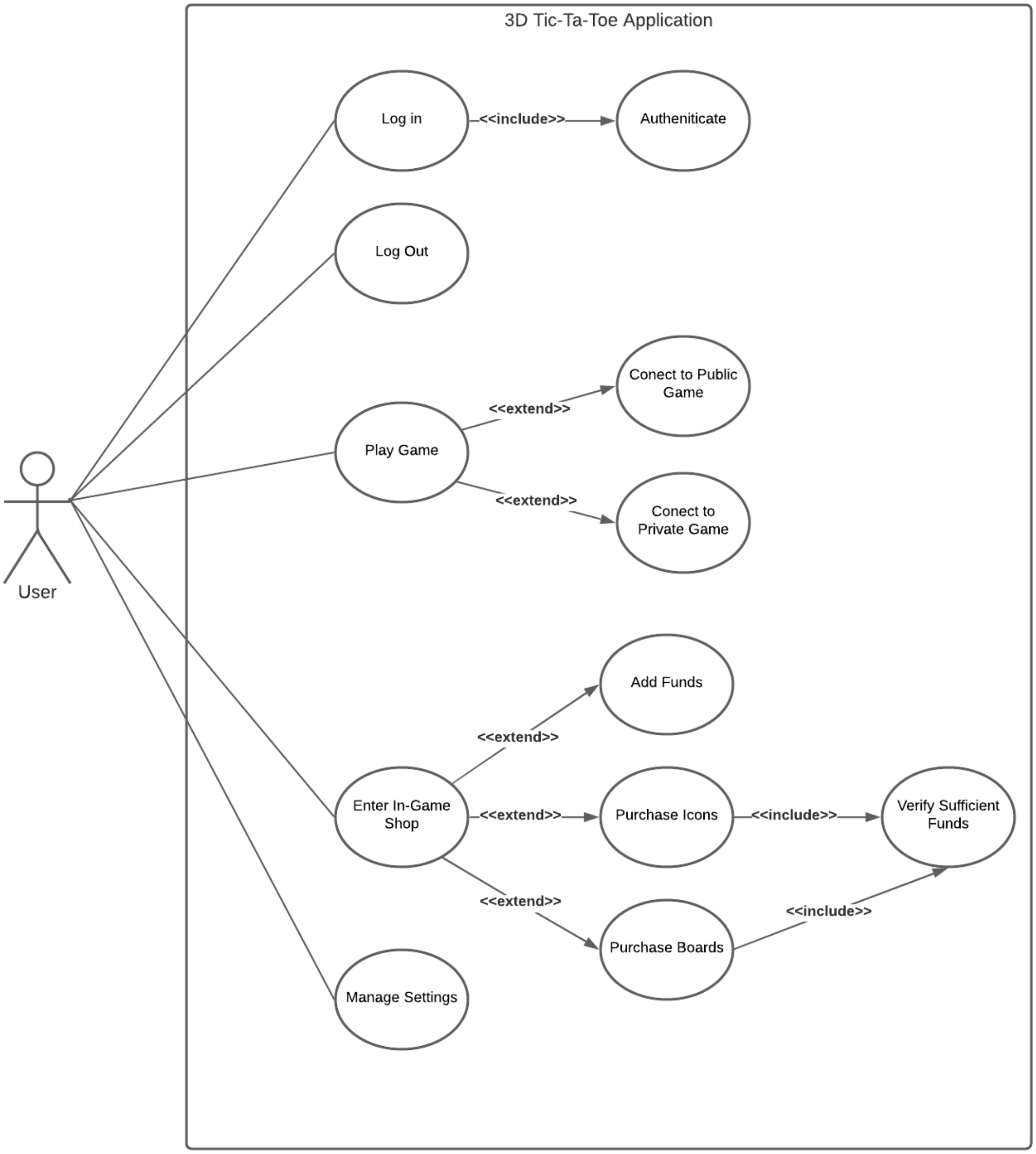


Figure 3 - Application Use Case Diagram

9b Product Use Case List

1) Log In – Player logging in.
2) Authenticate Login – Authenticating log-in credentials.

3) Log Out – Player logging out.
4) Play Game – Player playing a match.
5) Connect to Public Game – Players starting a public game search.
6) Connect to Private Game – Player connecting to a private game lobby.
7) Enter In-Game Shop – Players entering shop menu.
8) Add Funds – Player adding funds to account for in-game shop.
9) Purchase Icons – Player purchasing icons from in-game shop.
10) Purchase Boards – Player purchasing board from in-game shop.
11) Verify Sufficient Funds – Verifying sufficient funds for purchases.
12) Manage Settings – Player managing in-game settings.

9c Individual Product Use Cases

Use case ID: 1 Name: Log In

pre-conditions: None

post-conditions: Login Credentials are authenticated

Initiated by: User

Triggering Event: Starting application

Additional Actors: None

Sequence of Events:

1. User enters login username/email and password.
    2. System stores credentials
3. User presses login button.
    4. System authenticates login credentials.

Alternatives: User can also login by connecting 3rd party accounts (Facebook, google, etc.)

Exceptions: Will display an error when no internet connection is available.

Use case ID: 2 Name: Authenticate

pre-conditions: User has attempted to log in.

post-conditions: User is logged into account.

Initiated by: User

Triggering Event: User attempts to log in.

Additional Actors: None

Sequence of Events:

1. User attempts to log in
    2. System authenticates log in credentials.
3. User is logged into account.
    4. System shows user ais online.

Alternatives: In the case of incorrect credentials an error message is shown, and user is not logged into account.

Exceptions: Will display an error when no internet connection is available.

Use case ID: 3 Name: Log Out

pre-conditions: User Must be logged in.

post-conditions: User is no longer logged in.

Initiated by: User

Triggering Event: User pressed log out button.

Additional Actors: None

Sequence of Events:

1. User presses log out button.

    2. System logs out user from account.

Alternatives: None

Exceptions: None

Use case ID: 4 Name: Play Game

pre-conditions: User must be logged in.

post-conditions: User will join a game.

Initiated by: User

Triggering Event: User presses Play Game button.

Additional Actors: None

Sequence of Events:

1. Primary user presses play game button.

    2. System will show two additional options, “Public Match” and “Private Match”

Alternatives: None

Exceptions: None

Use case ID: 5 Name: Connect to Public game

pre-conditions: User has chosen to play a game post-conditions: User will join a public game with random players.

Initiated by: User

Triggering Event: User presses Public Match option.

Additional Actors: Other Users to play with.

Sequence of Events:

1. Primary User presses Public Match option
    2. System puts User into a queue.
3. Other users also press Public Match option
    4. System starts a public match when enough users are in the queue.
5. Users take turns each round.
    6. System reflects game state for each user.

Alternatives: None

Exceptions: None

Use case ID: 6 Name: Connect to Private Game

pre-conditions: User has chosen to play a game

post-conditions: User will join a private game.

Initiated by: User.

Triggering Event: User presses Private Match option.

Additional Actors: Other Users to play with.

Sequence of Events:

1. Primary User presses Private Match option
2. System puts user into private game lobby.
3. User sends invites to friends.
4. System directs invites to other users
5. User chooses to start private game
6. System starts a game with players present in the lobby.
7. Users take turns each round.
8. System reflects game state for each user.

Alternatives: None

Exceptions: None

Use case ID: 7 Name: Enter In-Game Shop

pre-conditions: User must be logged in.

post-conditions: User is in in-game shop menu.

Initiated by: User

Triggering Event: User presses In-Game Shop button on menu.

Additional Actors: None

Sequence of Events:

1. User press In-Game shop button.
2. System shows in game shop items.
3. User choose shop category.
4. System shows categories items.

Alternatives: None

Exceptions: None

Use case ID: 8 Name: Add funds

pre-conditions: User is logged in, and in shop menu.

post-conditions: User will have additional funds.

Initiated by: User

Triggering Event: User presses Add Funds option from shop menu.

Additional Actors: None

Sequence of Events:

1. User presses add funds option.
    2. System requests payment option.
3. User enters payment option and info.
    4. System executes transaction and adds funds User’s account.

Alternatives: System offers different payment options such as Credit Card, PayPal, Apple Pay, etc.

Exceptions: If payment information is incorrect, will display an error and fail to add funds.

Use case ID: 9 Name: Purchase Icons

pre-conditions: User has sufficient funds.

post-conditions: User unlocks new icons to use in-game.

Initiated by: User

Triggering Event: User purchases icon from shop using purchase option.

Additional Actors: None

Sequence of Events:

1. User chooses Purchase option for an icon.
    2. System asks for purchase confirmation.
3. User confirms purchase.
    4. System deducts funds and unlocks icon for User.

Alternatives: None

Exceptions: If user has insufficient funds, he will be prompted to add funds.

Use case ID: 10 Name: Purchase Boards

pre-conditions: User has sufficient funds.

post-conditions: User unlocks new game boards to be used in-game.

Initiated by: User

Triggering Event: User attempts to purchase a board from shop using purchase option.

Additional Actors: None

Sequence of Events:

1. User chooses Purchase option for a game board.
    2. System asks for purchase confirmation.

3. User confirms purchase.
    4. System deducts funds and unlocks new game board for User.

Alternatives: None

Exceptions: If user has insufficient funds, he will be prompted to add funds.

Use case ID: 11 Name: Verify Sufficient Funds

pre-conditions: User is logged in, with available funds,

post-conditions: Allows user to complete a purchase.

Initiated by: User

Triggering Event: User attempts to make a purchase.

Additional Actors: None

Sequence of Events:

1. User attempts to make a purchase
    2. System checks for sufficient funds, allowing user to complete purchase

Alternatives: If user has insufficient funds, he will be prompted to add funds.

Exceptions: None

Use case ID: 12 Name: Manage Settings

pre-conditions: User is logged in

post-conditions: User settings will be changed.

Initiated by: User

Triggering Event: Users presses options button from menu.

Additional Actors: None

Sequence of Events:

1. User choose Options button from main menu.
    2. System displays option menu.
3. User changes options.
    4. System reflects new settings.
5. User exits options menu.
    6. System saves newly changed options.

Alternatives: None

Exceptions: None

10 Functional Requirements

Requirement #1-Functional requirement- acceptance system

Description: The system should be able to accept all the inputs from the user.

Rationale: acceptance of inputs is necessary for the gameplay.

Fit Criterion: The system shall accept the inputs from the user.

Acceptance Tests: accept inputs for gameplay.

11 Data Requirements

Requirement #2 -Data- user input

Description: System will have different kinds of input by different users which is supposed to be saved by the system.

Rationale: different data should be handled by the system to support the gameplay.

Fit Criterion: data handler of the system should be at its best when in use.

Acceptance Tests: Large amounts of data have to be stored in the system.

12 Performance Requirements

12a Speed and Latency Requirements

Requirement #3 - speed-lag

Description: The system shall be able to minimize the lag between the user input and the system output.

Rationale: Reduced game lag is really important in improving the user experience and gameplay responses.

Fit Criterion: input and output responses should be lag free

Acceptance Tests: refresh rate and gameplay should be lag free.

12b Precision or Accuracy Requirements

Requirement #4 - Accuracy- detailed user interface

Description: The user interface should have real world graphics with detailed GUI.

Rationale: The real game GUI should replicate some high end graphics from popular games.

Fit Criterion: The game GUI should be pinpoint precise without any glitches.

Acceptance Tests: try using the interface on high graphics to real test the accuracy of GUI.

12c Capacity Requirements

Requirement #5 -Capacity-high refresh rate and storage

Description: The gameplay should run on high intensity graphics and should store the data of gameplay.

Rationale: The capacity should be to the point to store all the data of the user's progress.

Fit Criterion: The game should handle the data and refresh rate at constant intervals.

Acceptance Tests: At the end of each gameplay, the data should be updated with the new one.

13 Dependability Requirements

13a Reliability Requirements

Requirement #6 - reliability-long term usability

Description: The system must be able to handle the long term usability of the game.

Rationale: Long term usability is mandatory for the longevity of the game.

Fit Criterion: Long term usability is compulsory to maintain the value of the game to the public.

Acceptance Tests: Regularly test the game on high graphics to make sure the usability is up to the point.

13b Availability Requirements

Requirement #7 - availability-24hrs running

Description: The system should be available for use for the most part of the time the game is online .

Rationale: All time available for the system is necessary for the users for enjoying the gameplay.

Fit Criterion: System should be online for 24hrs in a day.

Acceptance Tests: availability for 24hrs.

13c Robustness or Fault-Tolerance Requirements

Requirement #8 - Fault-Tolerance-No-Internet

Description: The system shall be able to support offline gameplay.

Rationale: Offline gameplay is necessary for users to still have something to do when there's not an established online connection.

Fit Criterion: The product shall work for users without an internet connection.

Acceptance Tests: Play an offline game without an internet connection.

Requirement #9 - Fault-Tolerance-Saved-Data

Description: The system shall store user's offline game data on their local machine.

Rationale: Storing user's game data is necessary so users don't lose their progress while playing offline.

Fit Criterion: User's game data shall be saved to the local machine automatically every time a user hits save or quits the game.

Acceptance Tests: Play an offline game, hit save and quit, then return back to the offline game continuing from the saved point.

Requirement #10 - Fault-Tolerance-Update-Failure

Description: The system shall notify users if an update fails to install.

Rationale: Notifying users about an update failing is necessary so users shall be aware that their version is not up to date.

Fit Criterion: Within twenty seconds a pop up message shall display when update fails to install.

Acceptance Tests: Disconnect from an internet connection, try to update, and look for the pop up message to appear as updates require an internet connection.

Requirement #11 - Fault-Tolerance-Report-Errors

Description: The system shall give users the ability to report an error in the product.

Rationale: Allowing users to report an error in the product is necessary as it'll allow developers to tackle those errors and allow the product to be a great experience for users.

Fit Criterion: When an error occurs, players will have the option to report and submit the error that's ongoing to the developers from a pop-up.

Acceptance Tests: Force a game crash and check for pop-up to report an error.

13d Safety-Critical Requirements

Requirement #12 - Game-Crash-Safety

Description: The system shall not cause harm to the user's local machine when game crashes occur.

Rationale: Causing no harm to the user's local machine when the game crashes is necessary since we do not want to mess up their local machine which will result in unhappy customers.

Fit Criterion: One hundred percent of game crashes shall not affect any data or information on a user's local machine.

Acceptance Tests: Force the game to crash multiple times while verifying that no data or information of the user's local machine was affected.

Requirement #13 - Customer-Safety

Description: The system shall encrypt and protect all personal/sensitive information of all users.

Rationale: Encrypting and protecting all personal/sensitive information of all users is necessary as we want to ensure that their information is safe and not vulnerable to potential hackers.

Fit Criterion: All user information in the system's database shall be inaccessible to all but the pertaining user.

Acceptance Tests: All potential attempts to access information stored in the database will fail.

14 Maintainability and Supportability Requirements

14a Maintenance Requirements

Requirement #14 - Maintenance-Update

Description: The system shall force users to download a software update if available.

Rationale: Forcing the users to download a software update if available is necessary as it shall ensure an updated version of the game not containing bugs that were located in the previous version.

Fit Criterion: The user shall have to download the software update to play our game.

Acceptance Tests: When an update is available, users shall not be able to play the game. Once the update is installed and the system detects it, the user will have access to play the game.

Requirement #15 - Game-Maintenance-Users

Description: The system shall address reports made by the users.

Rationale: Addressing reports made by the users is necessary so that users shall enjoy a great experience.

Fit Criterion: Developers shall be notified by the reports and will work immediately to address the issues.

Acceptance Tests: Report an issue. The developers shall be notified and shall be given a time table to resolve the issue. Once the issue seems to be resolved, a software update shall be pushed out to all the users.

14b Supportability Requirements

Requirement #16 - Tutorial

Description: The system shall offer a game tutorial to users.

Rationale: Offering a game tutorial to users is necessary especially for new users who are trying to gain an understanding of how the game works.

Fit Criterion: Users shall be able to get a tutorial first time logging in as well at any time after.

Acceptance Tests: Upon completion of account creation and logging in for the first time, users shall be loaded into tutorial mode. Tutorial mode shall also be accessible at any point in the game.

14c Adaptability Requirements

Requirement #17 - Platforms

Description: The system shall be able to operate on iOS, Android, Linux, Mac, and Windows.

Rationale: Having the system available on multiple platforms is necessary as it allows for more exposure amongst customers.

Fit Criterion: Users who operate on iOS, Android, Mac, Linux, and Windows shall have access to the game at any time.

Acceptance Tests: The server shall check connectivity on all the supported platforms on a 24-hour basis.

14d Scalability or Extensibility Requirements

Requirement #18 - Scalability

Description: The system shall be capable of processing 50,000 customers initially. This number is expected to grow to 200,000 within three years and the system shall be ready to support the extra growth.

Rationale: Being able to handle more than the initial expectation is necessary as it shows the growth of the product.

Fit Criterion: Upon passing the initial expectation of 50,000 users, the servers shall be ready to handle another 20,000 users without crashing.

Acceptance Tests: Upon passing the initial expectation, the servers shall be tested to ensure proper functionality within the game.

Requirement #19 - Microtransactions

Description: The system shall be capable of processing 10,000 microtransactions per hour within the first year of launch.

Rationale: Being able to handle a good amount of microtransactions is necessary because that’s how as a company we profit.

Fit Criterion: Upon passing the initial expectation of 10,000 microtransactions, the system shall be ready to increase the amount of microtransactions being done per hour in 10,000 increments.

Acceptance Tests: Self-made transactions shall be made to test the amount of microtransactions being done.

14e Longevity Requirements

Requirement #20 - Longevity

Description: The system shall be up at least 4 years after initial launch.

Rationale: The system lasting at least 4 years is necessary as it will allow a good period of time for users to enjoy the product and allow developers to plan a successor to this product.

Fit Criterion: System will continue to receive updates for the first 4 years to keep all gameplay functionality consistent.

Acceptance Tests: Users will be able to access and play the product 24/7 for the first 4 years without fail.

15 Security Requirements

15a Access Requirements

Requirement #21 - Game-Records

Description: The system's records shall only be visible and altered by the game management.

Rationale: Having game management be the only group to see and alter users' records prevents other users to manipulate their own records.

Fit Criterion: One hundred percent of users shall be prevented from changing their own game records in the system.

Acceptance Tests: Attempts to alter personal game records will allow in users being reported.

15b Integrity Requirements

Requirement #22 - Self-Abuse

Description: The system shall protect itself from accidental corruption.

Rationale: Having protection from accidental corruption is necessary as it won't impact data that the system has collected.

Fit Criterion: Databases will be kept intact from all accidental corruptions.

Acceptance Tests: Purposeful accidental corruption will result in no damage to the system.

Requirement #23 - User-Security

Description: The system shall not reveal data of other users.

Rationale: The system not disclosing data of other users is necessary as it shall let users trust the system and its security of their personal information.

Fit Criterion: All user information shall be kept confidential at all times of the lifetime of the system.

Acceptance Tests: Potential hackers shall receive gibberish information.

15c Privacy Requirements

Requirement #24 - Privacy

Description: The system shall make its users aware of its information practices before collecting data from them.

Rationale: Notifying users of the system's information practices will educate the users on what the system will do.

Fit Criterion: Users will see a popup window containing a message of the system's information practices.

Acceptance Tests: Upon educating users of the system's information practices, users will have to click "Accept" in order to continue using the product.

Requirement #25 - Privacy-Update

Description: The system shall notify customers of changes to its information policy.

Rationale: Notifying users of the system's change of information policy is necessary as it will keep the users up to date on what the system will do.

Fit Criterion: Users will see a popup window containing a message of the updated system's information practices.

Acceptance Tests: Upon educating users of the updated system's information practices, users will have to click "Accept" in order to continue using the product.

15d Audit Requirements

Requirement #26 - Transaction-Audit

Description: The system and team shall be ready for an audit.

Rationale: Being prepared for an audit is necessary as it will update the team on the profitability of the business.

Fit Criterion: Team shall be ready to change and adjust things where need be according to the audit result.

Acceptance Tests: Audit score of 90% will be passing.

15e Immunity Requirements

Requirement #27 - Immunity

Description: The system will not be vulnerable to viruses.

Rationale: The system not being vulnerable to viruses is necessary as it will allow users to feel safe when downloading and using our product.

Fit Criterion: Users shall be able to download and use our product without endangering their local machine with viruses, trojans, etc.

Acceptance Tests: System shall be scanned and checked before release and daily to ensure the cleanliness of the system.

16 Usability and Humanity Requirements

16a Ease of Use Requirements

Requirement #28 - 12-Year-Old-Friendly

Description: The system shall be easy for 12 year old children to use.

Rationale: Having the system be easy for a 12 year old child to use is necessary as it will expand our community to wide age ranges and display that someone at a younger age can understand the product.

Fit Criterion: Eighty percent of a test panel of 12 year old children shall understand how to use the product within an hour.

Acceptance Tests: More than eighty percent of 12 year old children from the test panel understanding how to use the product within an hour results in acceptance.

Requirement #29 - Customized-To-User

Description: The system shall make users want to use it.

Rationale: Having customers coming back to use the system is necessary as it shall increase popularity within the system and drive success.

Fit Criterion: Seventy percent of users shall log in at least twice a day.

Acceptance Tests: More than seventy percent of users logging in twice a day results in acceptance.

Requirement #30 - Customer-Satisfaction

Description: The system shall introduce great customer satisfaction.

Rationale: Having the system providing great customer satisfaction is necessary as it shall drive users to continue using the system and spreading the word to friends/family about the system.

Fit Criterion: An anonymous survey shall show that ninety percent of users are happy with the system.

Acceptance Tests: More than ninety percent of users reporting a positive experience about the system shall result in acceptance.

16b Personalization and Internationalization Requirements

Requirement #31 - Language-Preference

Description: The system shall allow the user to select a chosen language.

Rationale: Having the system allowing a user to choose a language is necessary as it shall diversify our fanbase and allows for more users around the world to be a part of the community.

Fit Criterion: At launch, our system shall be able to support up to eight different languages.

Acceptance Tests: Upon launching the system, users shall be able to choose a language. From the list, there shall be at least eight different languages that the user can choose from.

Requirement #32 - Personal-Configurations

Description: The system shall allow the user to customize the system to how they want.

Rationale: Having the system allowing a user to customize the system to how they would like is necessary as it will increase the user's chance of returning to use the system. Users shall like that they'll be able to tailor the system to their liking.

Fit Criterion: Users shall be able to customize the system to their liking within twenty minutes of using the system.

Acceptance Tests: Anonymous report shall be sent to users asking how long it took them to uncover that the system was customizable. Thirty percent of users reporting more than twenty minutes results in failure.

16c Learning Requirements

Requirement #33 - Tutorial

Description: The system shall provide a tutorial for users.

Rationale: Having a tutorial is necessary as it shall provide users with intuition on how to use the product.

Fit Criterion: Ninety percent of users shall complete the tutorial within 20 minutes.

Acceptance Tests: At least ninety percent of users shall complete the tutorial within 20 minutes.

Requirement #34 - Ease-Of-Use

Description: The game shall be easy for the average gamer to use.

Rationale: Having the product easy for the average gamer to use is necessary as it shall not discourage users from attempting the product.

Fit Criterion: Gamers will rate the system on a scale of 0-5. Eighty percent of gamers shall rate the system a score of 4 or greater on ease of use.

Acceptance Tests: At least eighty percent of gamers shall rate the system a score of 4 or above.

16d Understandability and Politeness Requirements

Requirement #35 - No-Explicit-Language

Description: The system shall not condone explicit language.

Rationale: Having the system prohibit use of explicit language is necessary as it'll promote a positive community.

Fit Criterion: One hundred percent of explicit language will be filtered out from the system.

Acceptance Tests: Explicit language by players will be filtered out through the lifetime of the system.

16e Accessibility Requirements

Requirement #36 - Captions

Description: Subtitles of dialog in the game should be properly displayed on the game screen.

Rationale: This will allow people that have hearing impairments to still be able to play the game.

Fit Criterion: People that have hearing impairments.

Acceptance Tests: Subtitles should appear on the screen whenever a character is talking, or there is sound.

16f User Documentation Requirements

Requirement #37 - Manual

Description: A manual should come with the game, that tells the users the rules of the game, and provides information that would be important to the users.

Rationale: Users will be able to read the manual for information about the game.

Fit Criterion: The users can read the manual to learn the rules of the game.

Acceptance Tests: The manual should properly display the rules of the game.

16g Training Requirements

Requirement #38 - Trainings

Description: There will be no training for the game.

Rationale: Players can learn the game by playing.

Fit Criterion: 3D Tic Tac Toe is an easy game to learn.

Acceptance Tests: Test will be accepted when users are able to learn the game intuitively without additional training.

17 Look and Feel Requirements

17a Appearance Requirements

Requirement #39 - Key Users

Description: The game appearance should be attractive to anyone over the age of 3, interested in this game.

Rationale: The appearance of the game should be easy to use and understand when users play the game.

Fit Criterion: This requirement will pass if the game is visually appealing.

Acceptance Tests: Test will be accepter when users submit good feedback on UI.

17b Style Requirements

Requirement #40 - Theme

Description: The theme of the game should be inline with similar tic tac toe games.

Rationale: The game should be easy to use and understand, while being visually appealing.

Fit Criterion: This requirement will pass if the game theme is visually appealing.

Acceptance Tests: Test will be accepted when users are able to change game themes.

18 Operational and Environmental Requirements

18a Expected Physical Environment

Requirement #41 - Lights

Description: The product should have different light modes.

Rationale: Users can play this game anywhere, therefore, the product should be able to conform to different levels of lights.

Fit Criterion: Users can play the game in different levels of lights.

Acceptance Tests: Test the game with different light settings.

18b Requirements for Interfacing with Adjacent Systems

Requirement #42 - Online

Description: The game will be able to play online, therefore, the internet is needed so players can play together.

Rationale: The game will be sharing data online.

Fit Criterion: The game should be multiplayer.

Acceptance Tests: This test will pass if players can play together, with separate devices.

18c Productization Requirements

Requirement #43 - Deployment

Description: Users should be able to have access to the Apple app store, Google play store, or the internet in order to download and play.

Rationale: Users should have access to these platforms in order to play.

Fit Criterion: The game should be deployed on these platforms.

Acceptance Tests: This test will pass when the games are deployed on the platforms.

18d Release Requirements

Requirement #44 - Errors

Description: Each release should not have errors, or cause errors.

Rationale: New releases should have new features that help improve the game and experience for the users.

Fit Criterion: Each new release should not cause past features to have errors.

Acceptance Tests: This test will pass when there is a new release and no errors with that release.

19 Cultural and Political Requirements

19a Cultural Requirements

Requirement #45 - Cultures

Description: The product should not be offensive to anyone's culture.

Rationale: Our game is not intended to be offensive to anyone’s culture.

Fit Criterion: Anything that appears in the game, should not be offensive to anyone’s culture.

Acceptance Tests: Check that there is no offensive language.

19b Political Requirements

Requirement #46 - Politics

Description: The product should meet the requirements of the Apple app store, Google play store, or the internet.

Rationale: In order for our game to be deployed it should meet the platforms requirements.

Fit Criterion: The game should follow the requirements of the platforms.

Acceptance Tests: This test will pass as long as the requirements for the platforms are met.

20 Legal Requirements

20a Compliance Requirements

Requirement #47 - Data-Privacy

Description: The game should comply with data privacy requirements for all countries the game is offered in.

Rationale: Breaking data privacy will open the company up to lawsuits.

Fit Criterion: The product should follow data privacy for each country that the game is offered in.

Acceptance Tests: This test will pass as long as data privacy requirements are met for all countries the game is offered in.

20b Standards Requirements

Requirement #48 - Ethics

Description: The product should comply with game development ethics.

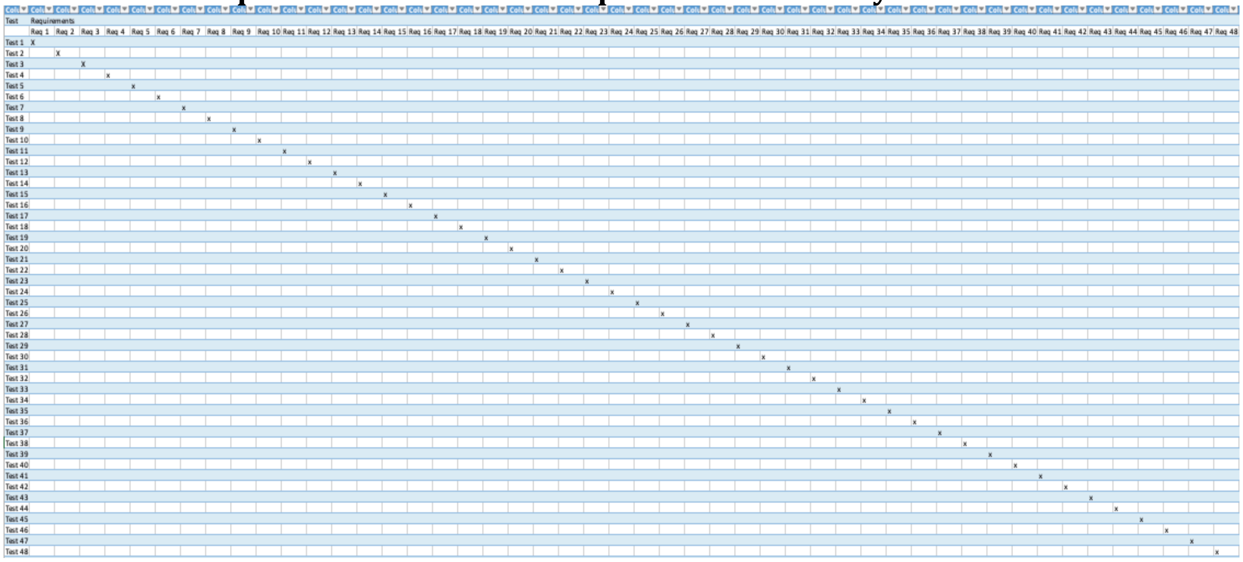
Rationale: Our product aims to be ethical to the game development standards.

Fit Criterion: This requirement will be met when our game is approved to be deployed, based on the game development ethics.

Acceptance Tests: Test will be accepted when geographical ethics are taken into consideration in all our markets to ensure approval of all our markets.

21 Requirements Acceptance Tests

21a Requirements – Test Correspondence Summary



Test	Requirements																																															
	Req 1	Req 2	Req 3	Req 4	Req 5	Req 6	Req 7	Req 8	Req 9	Req 10	Req 11	Req 12	Req 13	Req 14	Req 15	Req 16	Req 17	Req 18	Req 19	Req 20	Req 21	Req 22	Req 23	Req 24	Req 25	Req 26	Req 27	Req 28	Req 29	Req 30	Req 31	Req 32	Req 33	Req 34	Req 35	Req 36	Req 37	Req 38	Req 39	Req 40	Req 41	Req 42	Req 43	Req 44	Req 45	Req 46	Req 47	Req 48
Test 1	X																																															
Test 2		X																																														
Test 3			X																																													
Test 4				x																																												
Test 5					x																																											
Test 6						x																																										
Test 7							x																																									
Test 8								x																																								
Test 9									x																																							
Test 10										x																																						
Test 11											x																																					
Test 12												x																																				
Test 13													x																																			
Test 14														x																																		
Test 15															x																																	
Test 16																x																																
Test 17																	x																															
Test 18																		x																														
Test 19																			x																													
Test 20																				x																												
Test 21																					x																											
Test 22																						x																										
Test 23																							x																									
Test 24																								x																								
Test 25																									x																							
Test 26																										x																						
Test 27																											x																					
Test 28																												x																				
Test 29																													x																			
Test 30																														x																		
Test 31																															x																	
Test 32																																x																
Test 33																																	x															
Test 34																																		x														
Test 35																																			x													
Test 36																																				x												
Test 37																																					x											
Test 38																																						x										
Test 39																																							x									
Test 40																																								x								
Test 41																																									x							
Test 42																																										x						
Test 43																																											x					
Test 44																																												x				
Test 45																																													x			
Test 46																																														x		
Test 47																																															x	
Test 48																																																x

Table 2 - Acceptance Tests Correspondence

21b Acceptance Test Descriptions

Requirement #1-Functional requirement- acceptance system

Description: The system should be able to accept all the inputs from the user.

Requirement #2 -Data- user input

Description: System will have different kinds of input by different users which is supposed to be saved by the system.

Requirement #3 - speed-lag

Description: The system shall be able to minimize the lag between the user input and the system output.

Requirement #4 - Accuracy- detailed user interface

Description: The user interface should have real world graphics with detailed GUI.

Requirement #5 -Capacity-high refresh rate and storage

Description: The gameplay should run on high intensity graphics and should store the data of gameplay.

Requirement #6 - reliability-long term usability

Description: The system must be able to handle the long term usability of the game.

Requirement #7 - availability-24hrs running

Description: The system should be available for use for the most part of the time the game is online.

Requirement #8 - Fault-Tolerance-No-Internet

Description: The system shall be able to support offline gameplay.

Requirement #9 - Fault-Tolerance-Saved-Data

Description: The system shall store user’s offline game data on their local machine.

Requirement #10 - Fault-Tolerance-Update-Failure

Description: The system shall notify users if an update fails to install.

Requirement #11 - Fault-Tolerance-Report-Errors

Description: The system shall give users the ability to report an error in the product.

Requirement #12 - Game-Crash-Safety

Description: The system shall not cause harm to the user's local machine when game crashes occur.

Requirement #13 - Customer-Safety

Description: The system shall encrypt and protect all personal/sensitive information of all users.

Requirement #14 - Maintenance-Update

Description: The system shall force users to download a software update if available.

Requirement #15 - Game-Maintenance-Users

Description: The system shall address reports made by the users.

Requirement #16 - Tutorial

Description: The system shall offer a game tutorial to users.

Requirement #17 - Platforms

Description: The system shall be able to operate on iOS, Android, Linux, Mac, and Windows.

Requirement #18 - Scalability

Description: The system shall be capable of processing 50,000 customers initially. This number is expected to grow to 200,000 within three years and the system shall be ready to support the extra growth.

Requirement #19 - Microtransactions

Description: The system shall be capable of processing 10,000 microtransactions per hour within the first year of launch.

Requirement #20 - Longevity

Description: The system shall be up at least 4 years after initial launch.

Requirement #21 - Game-Records

Description: The system's records shall only be visible and altered by the game management.

Requirement #22 - Self-Abuse

Description: The system shall protect itself from accidental corruption.

Requirement #23 - User-Security

Description: The system shall not reveal data of other users.

Requirement #24 - Privacy

Description: The system shall make its users aware of its information practices before collecting data from them.

Requirement #25 - Privacy-Update

Description: The system shall notify customers of changes to its information policy.

Requirement #26 - Transaction-Audit

Description: The system and team shall be ready for an audit.

Requirement #27 - Immunity

Description: The system will not be vulnerable to viruses.

Requirement #28 - 12-Year-Old-Friendly

Description: The system shall be easy for 12 year old children to use.

Requirement #29 - Customized-To-User

Description: The system shall make users want to use it.

Requirement #30 - Customer-Satisfaction

Description: The system shall introduce great customer satisfaction.

Requirement #31 - Language-Preference

Description: The system shall allow the user to select a chosen language.

Requirement #32 - Personal-Configurations

Description: The system shall allow the user to customize the system to how they want.

Requirement #33 - Tutorial

Description: The system shall provide a tutorial for users.

Requirement #34 - Ease-Of-Use

Description: The game shall be easy for the average gamer to use.

Requirement #35 - No-Explicit-Language

Description: The system shall not condone explicit language.

Requirement #36 - Captions

Description: Subtitles of dialog in the game should be properly displayed on the game screen.

Requirement #37 - Manual

Description: A manual should come with the game, that tells the users the rules of the game, and provides information that would be important to the users.

Requirement #38 - Trainings

Description: There will be no training for the game.

Requirement #39 - Key Users

Description: The game appearance should be attractive to anyone over the age of 3, interested in this game.

Requirement #40 - Theme

Description: The theme of the game should be inline with similar tic tac toe games.

Requirement #41 - Lights

Description: The product should have different light modes.

Requirement #42 - Online

Description: The game will be able to play online, therefore, the internet is needed so players can play together.

Requirement #43 - Deployment

Description: Users should be able to have access to the Apple app store, Google play store, or the internet in order to download and play.

Requirement #44 - Errors

Description: Each release should not have errors, or cause errors.

Requirement #45 - Cultures

Description: The product should not be offensive to anyone's culture.

Requirement #46 - Politics

Description: The product should meet the requirements of the Apple app store, Google play store, or the internet.

Requirement #47 - Data-Privacy

Description: The game should comply with data privacy requirements for all countries the game is offered in.

Requirement #48 - Ethics

Description: The product should comply with game development ethics..

III Design

22 Design Goals

One of the most important design goals for 3D Tic-Tac-Toe is to have an easy to pick-up and learn user interface. As mentioned earlier, we plant to hit a large range of demographics, and some of the bigger demographics we want to target are the older and younger demographics. In order to target these audiences, we must be able to have an interface that is easy enough for a young child to use, as well as older audiences that may not have a much experience using technology. Although both of these demographics are large in the mobile gaming industry, we are especially interested in capturing the older demographic which statistics show are more likely to spend time playing mobile games and making microtransaction purchases.

Another important design goal is to make the overall system free of latency and lag. As with most online games, it is important to make the online experience as smooth as possible. Most gamers have very little tolerance for laggy/buggy games, and for this reason it is especially important to ensure that the game releases in a state that has minimal lag/latency.

23 Current System Design

There is no pre-existing system design.

24 Proposed System Design

24a Initial System Analysis and Class Identification

Player: We will need this class to hold information about the user, as well as other player's from online matches.

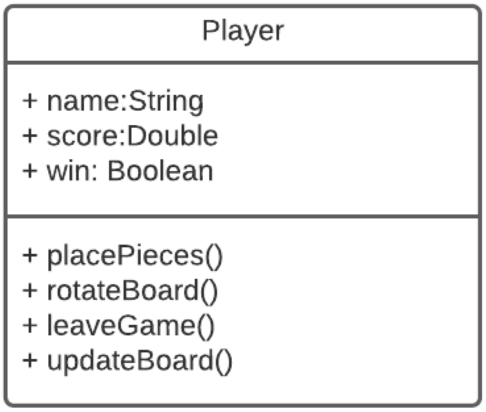


Figure 4 - Player Class

Board: This class will be used to keep track of changes to the game board, as well as displaying itself on the UI screen.

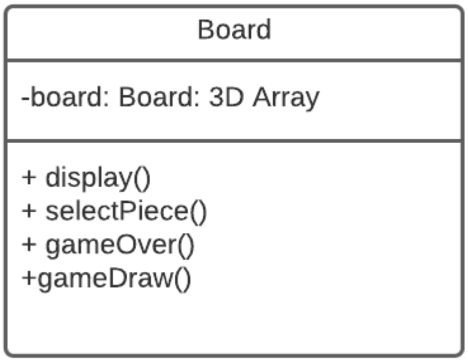


Figure 5 - Board Class

Match: The purpose of this class will be to keep track of all the players currently in a match, their scores, as well as the shared game board.

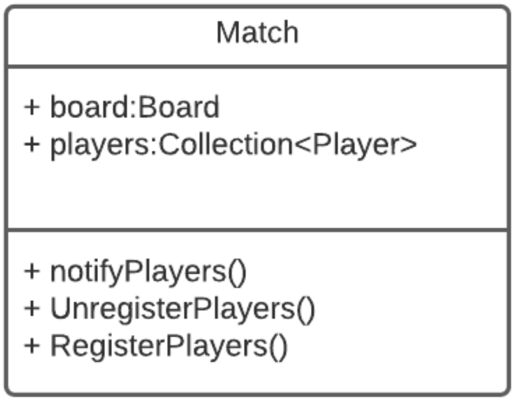


Figure 6 - Match Class

Game: This class will keep track of all the elements of the overall game.

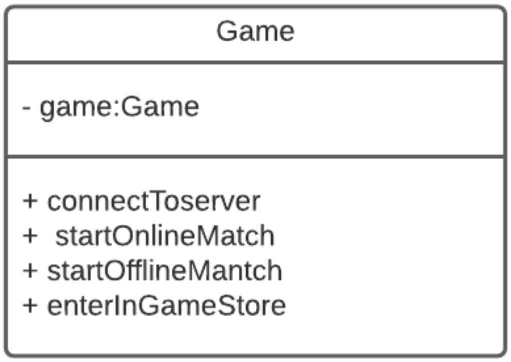


Figure 7 - Game Class

GameShop: A class to be used for purchasing microtransaction from the in-game shop.

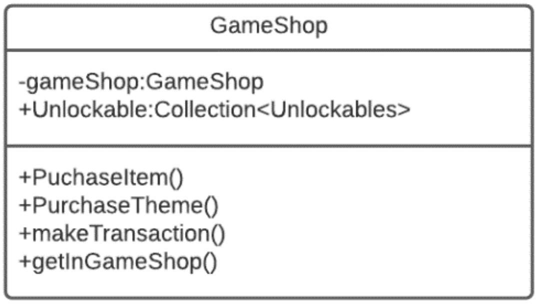


Figure 8 - GameShop Class

24b Dynamic Modelling of Use-Cases

The Sequence diagram below models 4 of our main use cases. Logging in, playing online, making a move, and purchasing content.

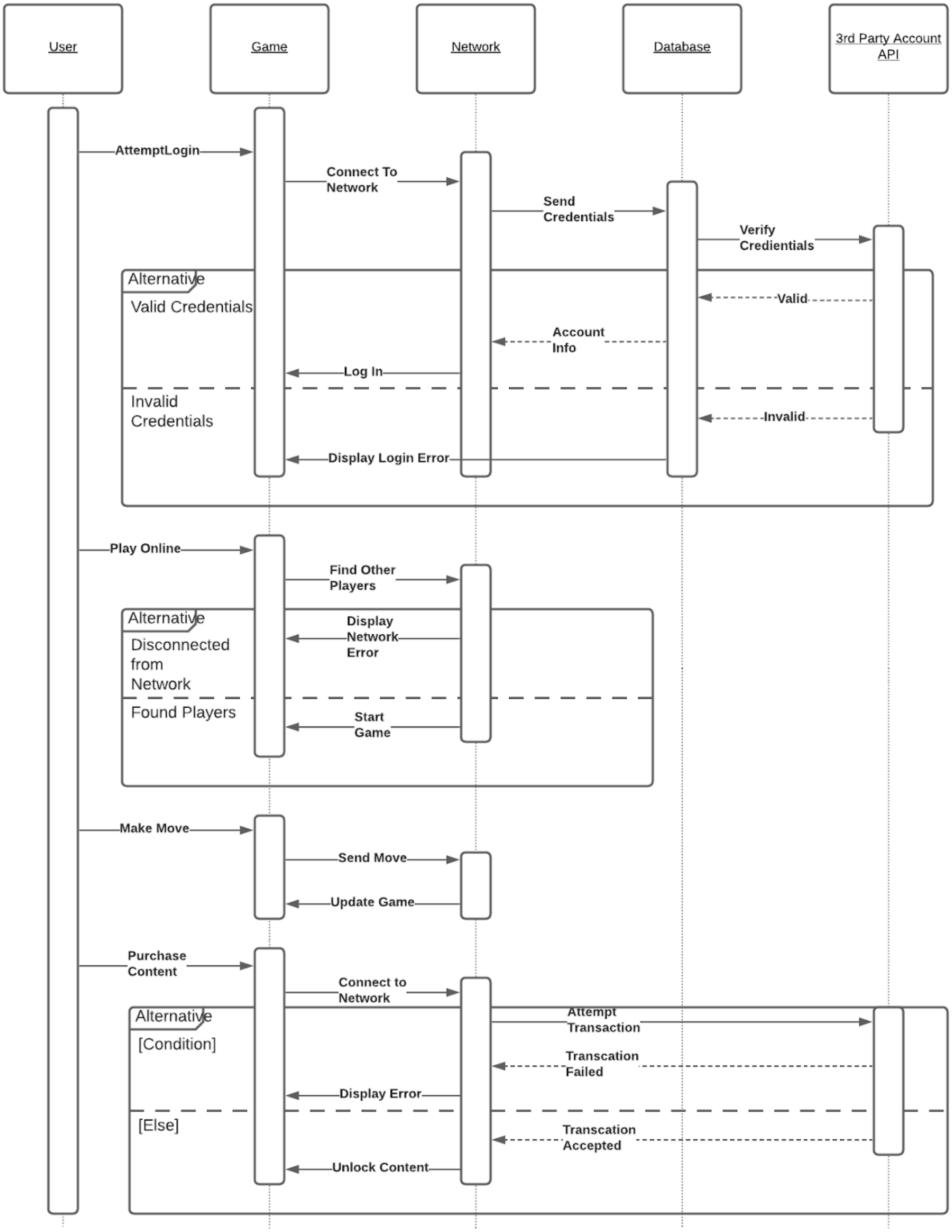


Figure 9 - Sequence Diagram

24c Proposed System Architecture

For 3D Tic-Tac-Toe we will be using the Peer-to-Peer system architecture for the focus of the system. The reason we chose to use Peer-to-Peer is due to the multiplayer aspect of our system. When connecting players for an online match we do not need a specific player to be in charge. For that reason, we believe that Peer-to-Peer is an excellent choice. However other aspects of the system would be using the Client-Server architecture, such as connecting to the network to find other players, logging in, or making online microtransaction purchases.

24d Initial Subsystem Decomposition

Interface Subsystem: The purpose of this subsystem is to handle interface options of the system such as logging in, starting a game, or purchasing a microtransaction.

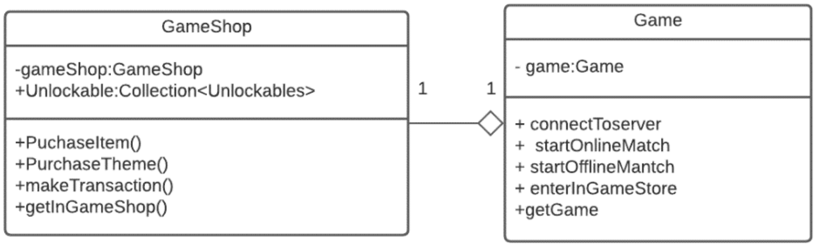


Figure 10 - Interface Subsystem

Player Subsystem: The purpose of this subsystem is to keep track of player information, as well as their actions.

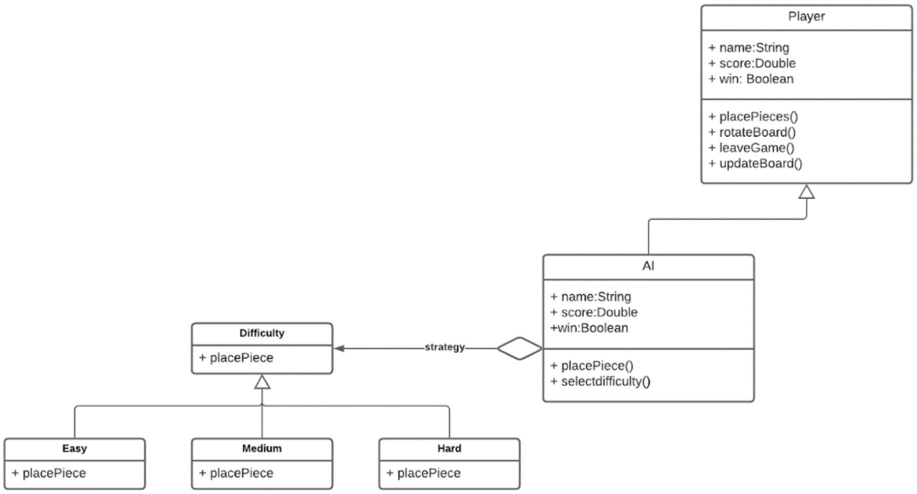


Figure 11 - Player Subsystem

Match Subsystem: The purpose of this subsystem is to keep track of the players in a match as well as the changes to the matches board from other players.

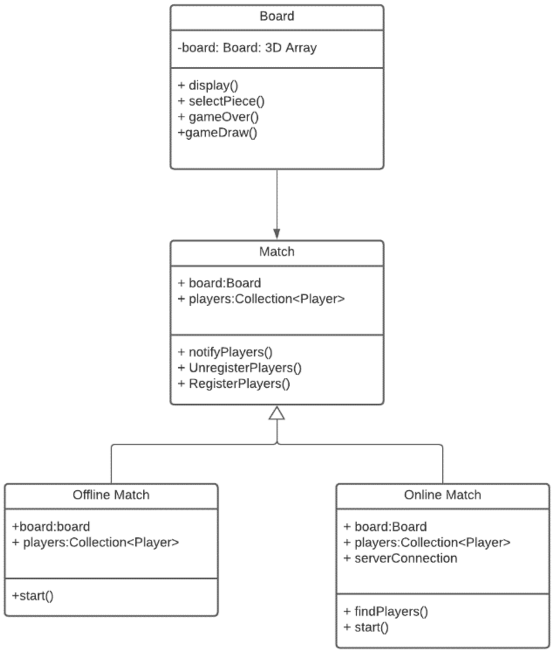


Figure 12 - Match Subsystem

25 Additional Design Considerations

25a Hardware / Software Mapping

3D Tic-Tac-Toe will be hosted on various platforms, for example, online, steam, google play, and apple game center, so hardware and software mapping would not be applicable for the software. The developers of the game will use third party API's to allow users to access the game on their devices. Additional servers will be added to the application as more users begin to play. We also plan to add holiday themes and events to the application and include chat functionality. To ensure that our game is not offensive to anyone we want to include functionality that blocks explicit language

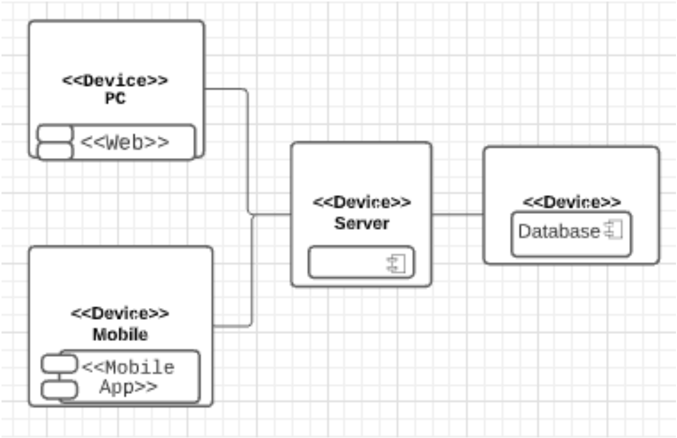


Figure 13 - HW/SW Mapping

25b Persistent Data Management

A database will be used to store information about the game and the current users. The database will store information about the players current move as well as usernames and passwords for all of the users accounts, messages sent through chat, purchases bought from the store, customization settings for the users characters, and difficulty settings for the game.

25c Access Control and Security

As mentioned above the users will be able to access the game on their devices using third party API's implemented in the application by developers. Accessing the accounts in the game must be secure, because users will be using personal information and make purchases in the store that will be stored in the database. Developers should use third API's for purchases, so that the application is more secure and to avoid storing personal information in the database.

25d Global Software Control

Users will have to login through third party API's to access the game, then users will be able make purchases and start the game. The application will respond to user input, therefore, the software control for this game will be selection/decision based.

25e Boundary Conditions

To handle boundary conditions for the application users information will be stored in a database and the system will get that data when the application is started, and save the information when the application is shut down. When the application is started, the last saved data from the database will be loaded if there is any data lost, and when the application is shut down the database will save the information for later. The applications must automatically save information every couple minutes to ensure that data is not lost if there is an accidental shutdown.

25f User Interface

The user interface of 3D-Tic-Tac-Toe should match similar themes of the game, be appealing to the players, and be user friendly. When the application is started, the game will load the login screen and users will be able to login using third party API's. Then the users will be directed to the main menu screen, this will allow users to play the game, make purchases, logout, and change the settings.

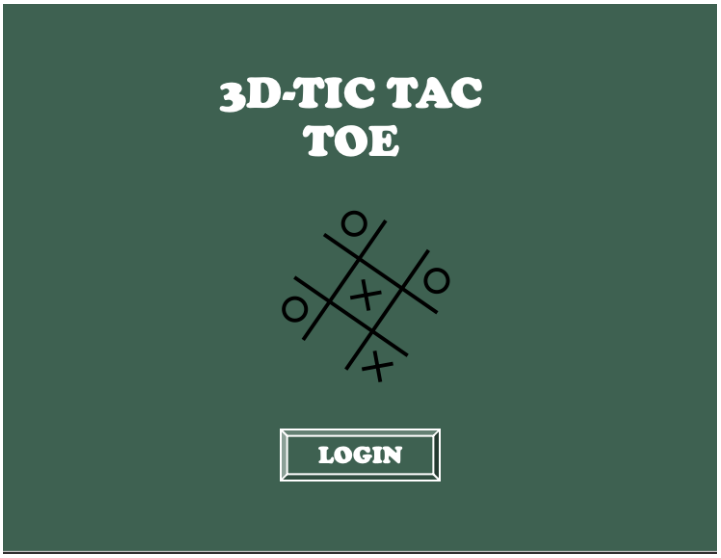


Figure 14 - UI Login

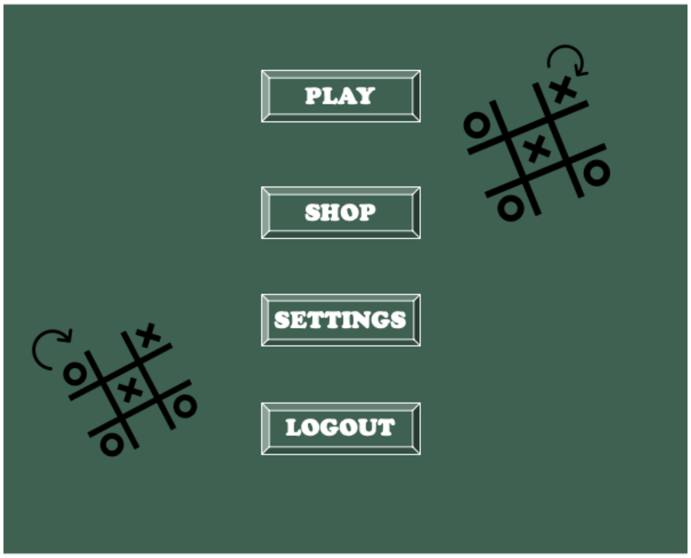


Figure 15 - UI Main Menu

25g Application of Design Patterns

Singleton: The Singleton design pattern was used for both the GameShop and Game classes. We chose to use the singleton pattern for these classes due to only needing at most one instance of each at any time. In order to reduce any complications of having multiple instances we chose to use the Singleton design pattern.

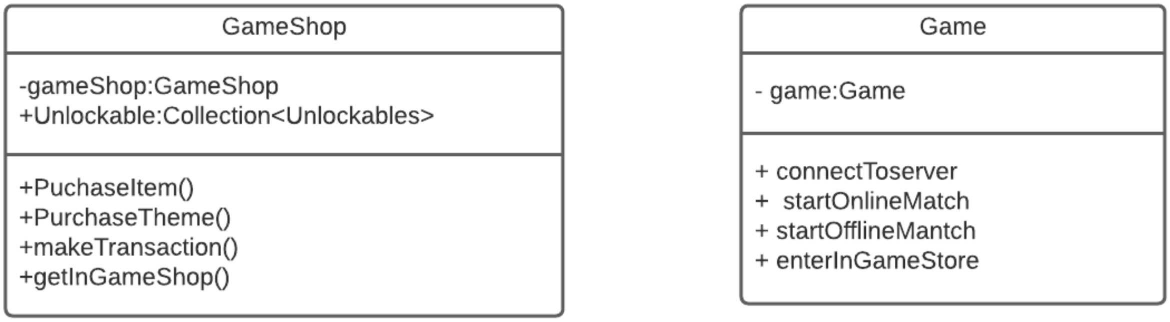


Figure 16 – Singleton

Observer: We chose to use the observer pattern between the Player objects and the Match object. In this relationship each player in a match is observing the shared match they are in. From this relationship whenever any player makes a change to the match/board, the match will notify all it's observers/players to update their respective UIs/games.

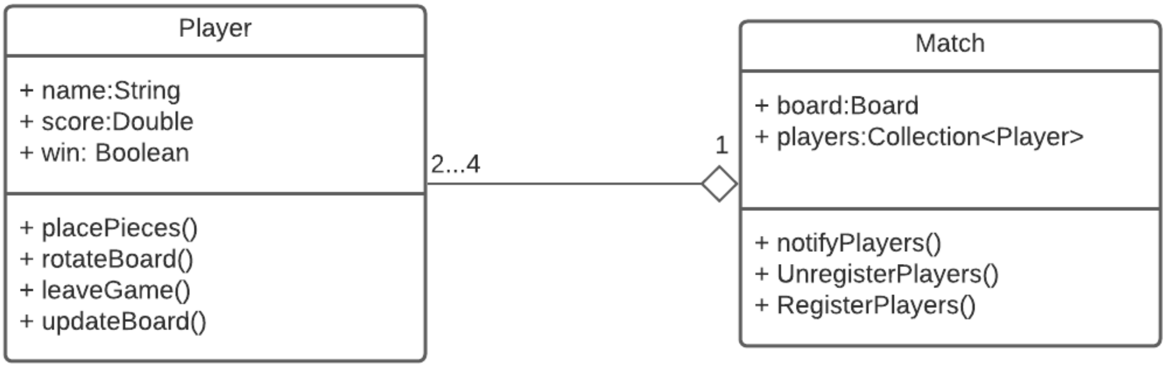


Figure 17 - Observer

Strategy: In our system we chose to use the Strategy design pattern to configure the difficulty of AI players during offline game. The Strategy design pattern is perfect for this scenario as it makes it easier to change the difficulty on the fly. The three included difficulties include Easy, Medium, and Hard.

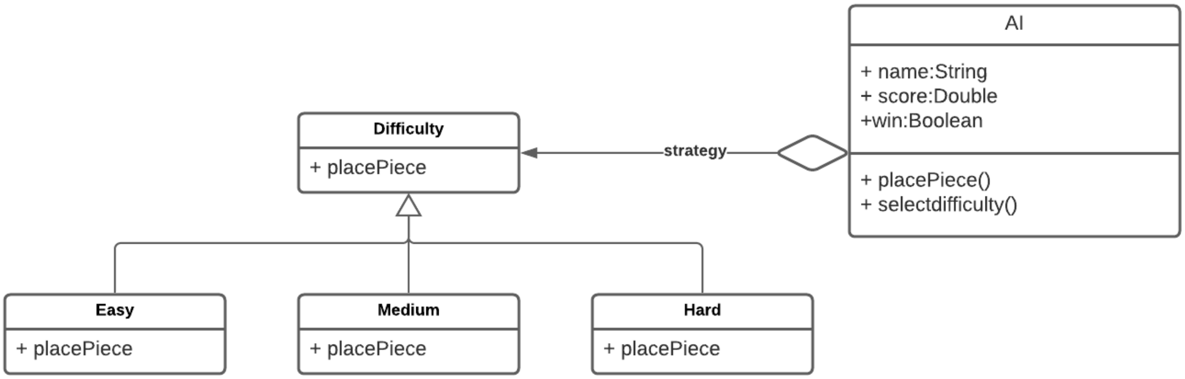


Figure 18 – Strategy

26 Final System Design

Below is a UML showing the overall system design including all the classes we identified and design patterns implemented.

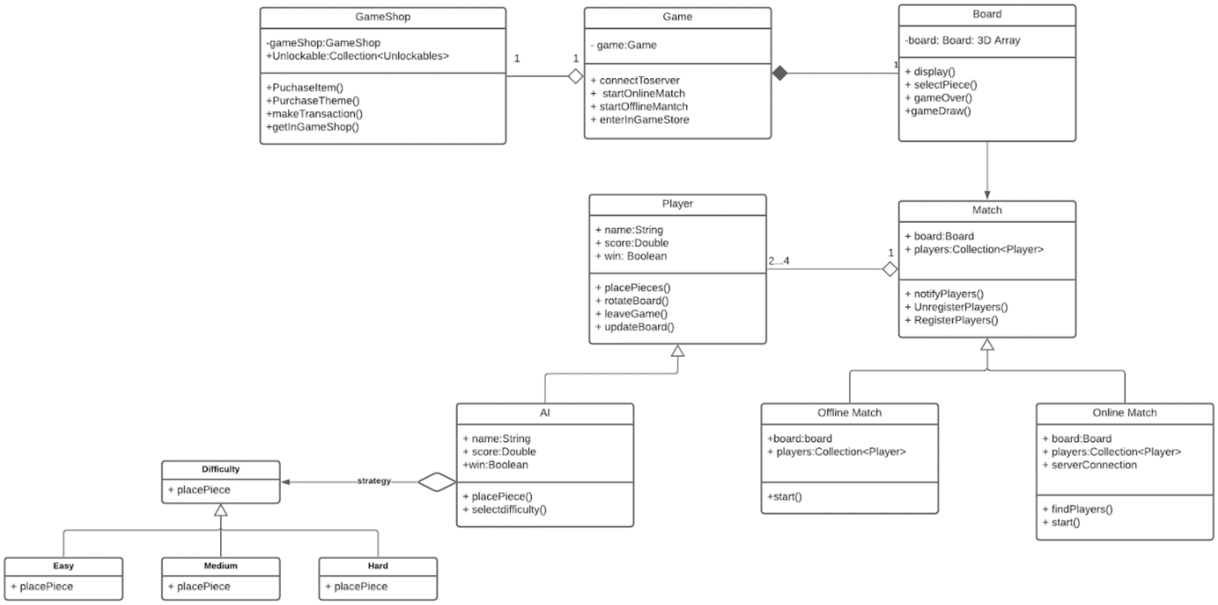


Figure 19 - System Design UML

27 Object Design

27a Packages

Interface Package The purpose of this package is to handle interface options of the system such as logging in, starting a game, or purchasing a microtransaction.

- In game shop
- Purchase items
- Purchase themes
- Current game
- Connect to server
- Start online
- Start offline

Match Package - The purpose of this package is to keep track of player information, as well as their actions.

- 3D array board
- Game over
- Game draw
- Display game
- Select a piece
- Current players for match
- Register players
- Unregister players
- Notify players
- Start game
- Online match
- Offline match

Player Package - The purpose of this subsystem is to keep track of the players in a match as well as the changes to the matches board from other players.

- Player name
- Players score
- Update the board
- Place a piece
- Leave the game
- Player AI
- Set Difficulty

27b Subsystem I

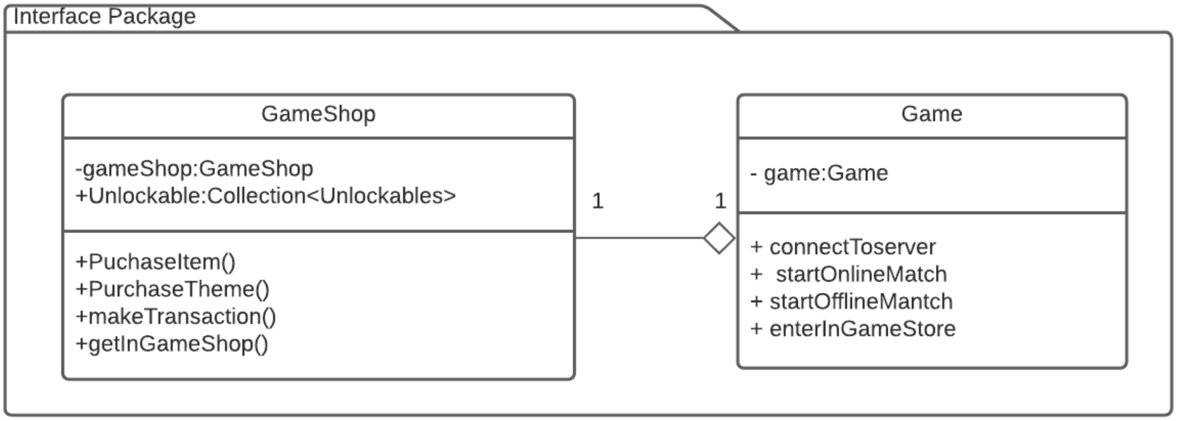


Figure 20 - Interface Package

27c Subsystem II

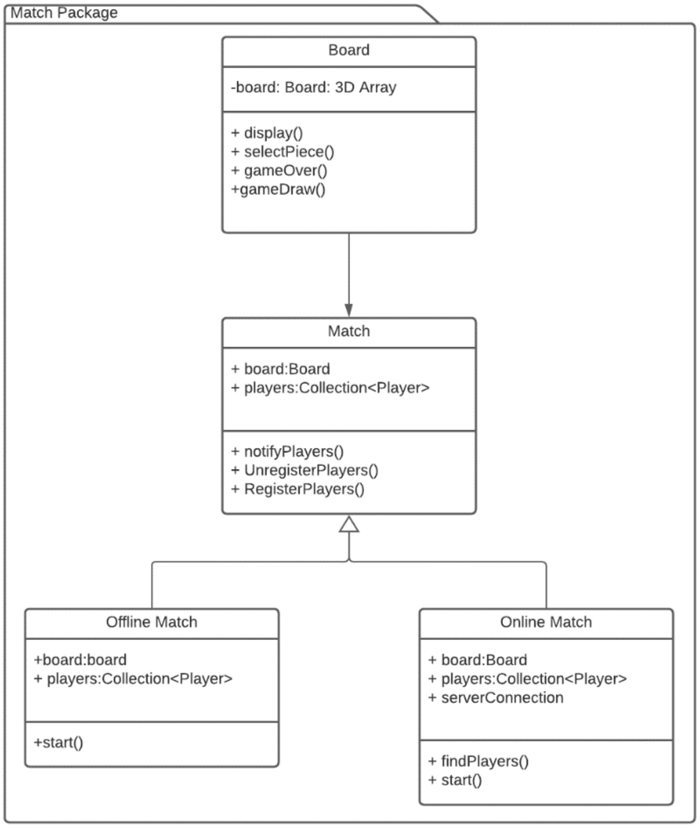


Figure 21 - Match Package

27d Subsystem III

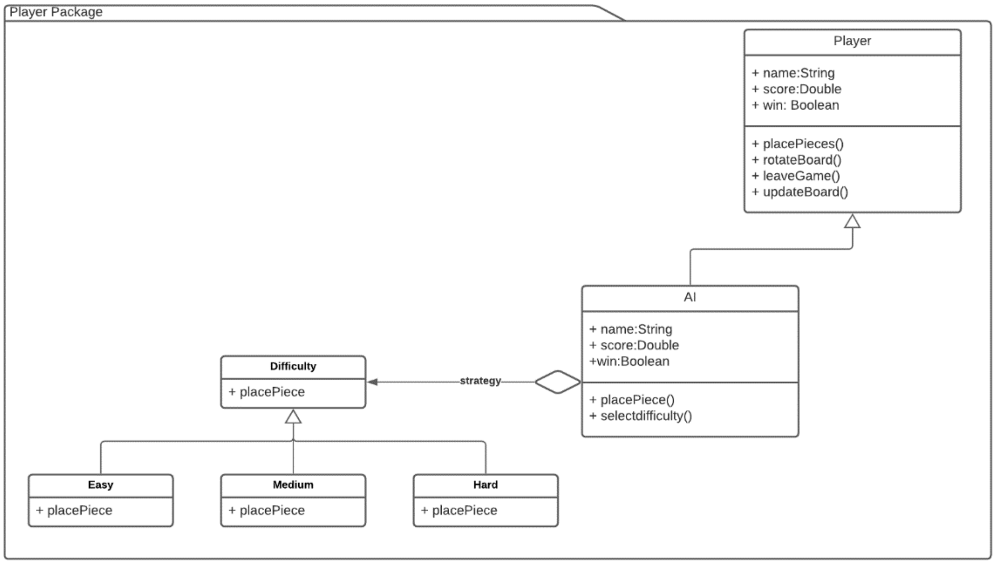


Figure 22 - Player Package

IV Project Issues

28 Open Issues

Open Issue #1: Linux Availability

Content: Currently have the product available on Windows, Android, iOS, and macOS.

Motivation: Is it worth making the game available on Linux?

Example: Linux has less than 2% of the desktop market share.

Open Issue #2: Hardware

Content: Different users run different hardware components on their machine.

Motivation: Users might not have the minimum hardware components necessary to run the product.

Examples: People will lower budget CPUs and GPUs will more than likely not be able to run the product at an optimized playing level.

Open Issue #3: More than the target market

Content: All games focus on certain target audiences.

Motivation: The goal is to capture more than that with the help of marketing analysts.

Example: Possibility of the target audience to not like the product. So it’s best to capture more to sustain the existence of the product.

Open Issue #4: New devices

Content: New, popular devices could become available to the market where a new market will become available to approach.

Motivation: Coming up with a way to bring our product to new devices if available.

Example: Let’s say a new handheld console comes out, for instance when the PSP came out in 2005. There’s a chance our product won’t be available, but the goal is to be prepared to make it readily available.

29 Off-the-Shelf Solutions

To get the product started, we can use Unity as our game engine. However, that’s just one example as we can perhaps reuse other game engines that are legal to do so such as CryEngine or Unreal. For storing data, we can use SQL. As mentioned earlier we will also be utilizing third party login API’s as an alternative to creating our own login system.

29a Ready-Made Products

To get the product started, we can use Unity as our game engine. However, that’s just one example as we can perhaps reuse other game engines that are legal to do so such as CryEngine or Unreal. For storing data, we can use SQL. As mentioned earlier we will also be utilizing third party login API’s as an alternative to creating our own login system.

29b Reusable Components

All programming languages and libraries within those languages that are used for the development of the project will be reused. No physical product though that can be used.

29c Products That Can Be Copied

We can look into buying another company's specifications on their customer service system to cut costs. We would add our own modifications on top of that to distinguish it.

30 New Problems

30a Effects on the Current Environment

Content: the product will be able to run on the currently available specs.

Motivation: Is it possible to update the specs to run this product more smoothly?

Example: The individual having this product will update the specs if required for this product to run smoothly.

30b Effects on the Installed Systems

Content: The effects that the product will have on the installed system.

Motivation: Is it possible to find a way to prevent the effects on the installed system.

30c Potential User Problems

Content: The problems users will face while using the product in the system.

Motivation: Will the change in the current product solve the user problems.

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

Content: Lack of reliable power will be the main cause of issue that can occur while implementation

Motivation: Can getting a reliable source of power solve the issue of the needed environment.

30e Follow-Up Problems

Content: lack of reliable environment for the product will affect the user

Motivation: Can getting a reliable environment for the product solve the issue for the user.

31 Migration to the New Product

31a Requirements for Migration to the New Product

N/A

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

N/A.

32 Risks

- Misunderstanding or miscommunication between developers, clients, and users.
- Inaccurate assumptions about our demographics and markets.
- Loss of documents due to disorganization or changing in systems.
- Bias work introduced by the developers.
- Low productivity due to environmental factors.
- Cancellation of events from unforeseen circumstances such as COVID-19.

33 Costs

We expect our project to staying with a budget $300,00. The budget for this project will cover development costs, testing, as well as marketing. Marketing is one of the more important costs since without marketing we could release our product and have no userbase making it a failure.

34 Waiting Room

Content: the ability of multiple players i.e. more than 4 players.

Motivation: what will be the method to add more than 4 players who can play this game together. This would require major changes to the gameplay as well as the board.

35 Ideas for Solutions

Content: Unity can be used in future consideration as it is multi-platform and powerful.

Motivation: Will a software like unity be helpful in the future development of the project.

36 Project Retrospective

Content: The best thing that worked well was the coordination of the entire team. The thing that went a bit wrong was the numbering issues in this document.

Motivation: Will sequential ordering resolve the issue we face during completing the report.

V Glossary

PC: A Personal computer, which could refer to both laptop and desktop computers.

IOS: iPhone Operating System used for mobile Apple devices.

Android: An open source mobile operating system, used by most mobile manufacturers.

Steam: A cloud boused game library which allows consumers to buy and download pc games using their steam account.

Android Play Store: An Android app equivalent to the Apple app store for iPhones.

Play Games: Android/Google's online game service, for connecting with friends and keeping track of achievements on Android.

Game Center: Apples online mobile game service, for connecting with friends and keeping track of achievements on IOS.

VI References / Bibliography

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

VII Index