***Ikon Wars Project Report***

**

***Prepared by***

***Nemil Shah, Wishy Parikh, Ruchit Patel, David Qiao***

***for use in* CS 440**

**at the**

**University of Illinois Chicago**

**Table of Contents**

[*REMOVE OR REPLACE ALL TEXT IN RED ITALICS BEFORE SUBMITTING REPORT* 2](#_30j0zll)

[*How to Use This Document* 2](#_3z7bk57)

[List of Figures 9](#_1fob9te)

[List of Tables 10](#_3znysh7)

[I](#_3znysh7) Project Description 11

[1](#_2et92p0) Project Overview 11

[2](#_tyjcwt) The Purpose of the Project 11

[2a](#_3dy6vkm) The User Business or Background of the Project Effort 11

[2b](#_1t3h5sf) Goals of the Project 11

[2c](#_4d34og8) Measurement 12

[3](#_17dp8vu) The Scope of the Work 13

[3a](#_3rdcrjn) The Current Situation 13

[3b](#_26in1rg) The Context of the Work 13

[3c](#_lnxbz9) Work Partitioning 16

[3d](#_35nkun2) Competing Products 18

[4](#_1ksv4uv) The Scope of the Product 18

[4a](#_44sinio) Scenario Diagram(s) 19

[4b](#_2jxsxqh) Product Scenario List 19

[4c](#_z337ya) Individual Product Scenarios 19

[5](#_3j2qqm3) Stakeholders 20

[5a](#_1y810tw) The Client 20

[5b](#_4i7ojhp) The Customer 20

[5c](#_2xcytpi) Hands-On Users of the Product 21

[5d](#_1ci93xb) Maintenance Users and Service Technicians 22

[5e](#_3whwml4) Other Stakeholders 22

[5f](#_2bn6wsx) User Participation 23

[5g](#_qsh70q) Priorities Assigned to Users 24

[6](#_3as4poj) Mandated Constraints 25

[6a](#_1pxezwc) Solution Constraints 25

[6b](#_49x2ik5) Implementation Environment of the Current System 26

[6c](#_2p2csry) Partner or Collaborative Applications 27

[6d](#_3o7alnk) Off-the-Shelf Software 28

[6e](#_23ckvvd) Anticipated Workplace Environment 29

[6f](#_ihv636) Schedule Constraints 30

[6g](#_32hioqz) Budget Constraints 31

[7](#_1hmsyys) Naming Conventions and Definitions 31

[7a](#_41mghml) Definitions of Key Terms 31

[7b](#_2grqrue) UML and Other Notation Used in This Document 32

[7c](#_vx1227) Data Dictionary for Any Included Models 33

[8](#_3fwokq0) Relevant Facts and Assumptions 34

[8a](#_1v1yuxt) Facts 34

[8b](#_4f1mdlm) Assumptions 34

[II](#_2u6wntf) Requirements 36

[9](#_19c6y18) Product Use Cases 36

[9a](#_3tbugp1) Use Case Diagrams 36

[9b](#_37m2jsg) Product Use Case List 38

[9c](#_1mrcu09) Individual Product Use Cases 38

[10](#_46r0co2) Functional Requirements 40

[11](#_2lwamvv) Data Requirements 41

[12](#_111kx3o) Performance Requirements 42

[12a](#_3l18frh) Speed and Latency Requirements 42

[12b](#_206ipza) Precision or Accuracy Requirements 43

[12c](#_4k668n3) Capacity Requirements 44

[13](#_2zbgiuw) Dependability Requirements 45

[13a](#_1egqt2p) Reliability Requirements 45

[13b](#_3ygebqi) Availability Requirements 46

[13c](#_2dlolyb) Robustness or Fault-Tolerance Requirements 47

[13d](#_sqyw64) Safety-Critical Requirements 48

[14](#_3cqmetx) Maintainability and Supportability Requirements 49

[14a](#_1rvwp1q) Maintenance Requirements 49

[14b](#_4bvk7pj) Supportability Requirements 50

[14c](#_2r0uhxc) Adaptability Requirements 51

[14d](#_1664s55) Scalability or Extensibility Requirements 52

[14e](#_3q5sasy) Longevity Requirements 52

[15](#_25b2l0r) Security Requirements 53

[15a](#_kgcv8k) Access Requirements 53

[15b](#_34g0dwd) Integrity Requirements 54

[15c](#_1jlao46) Privacy Requirements 55

[15d](#_43ky6rz) Audit Requirements 56

[15e](#_2iq8gzs) Immunity Requirements 57

[16](#_xvir7l) Usability and Humanity Requirements 57

[16a](#_3hv69ve) Ease of Use Requirements 57

[16b](#_1x0gk37) Personalization and Internationalization Requirements 59

[16c](#_4h042r0) Learning Requirements 60

[16d](#_2w5ecyt) Understandability and Politeness Requirements 61

[16e](#_1baon6m) Accessibility Requirements 62

[16f](#_3vac5uf) User Documentation Requirements 63

[16g](#_2afmg28) Training Requirements 64

[17](#_pkwqa1) Look and Feel Requirements 65

[17a](#_39kk8xu) Appearance Requirements 65

[17b](#_1opuj5n) Style Requirements 66

[18](#_48pi1tg) Operational and Environmental Requirements 67

[18a](#_2nusc19) Expected Physical Environment 67

[18b](#_1302m92) Requirements for Interfacing with Adjacent Systems 68

[18c](#_3mzq4wv) Productization Requirements 69

[18d](#_2250f4o) Release Requirements 70

[19](#_haapch) Cultural and Political Requirements 71

[19a](#_319y80a) Cultural Requirements 71

[19b](#_1gf8i83) Political Requirements 72

[20](#_40ew0vw) Legal Requirements 73

[20a](#_2fk6b3p) Compliance Requirements 73

[20b](#_upglbi) Standards Requirements 74

[21](#_1tuee74) Requirements Acceptance Tests 75

[21a](#_4du1wux) Requirements – Test Correspondence Summary 75

[21b](#_184mhaj) Acceptance Test Descriptions 76

[III](#_3s49zyc) Design 76

[22](#_279ka65) Design Goals 76

[23](#_meukdy) Current System Design 76

[24](#_36ei31r) Proposed System Design 76

[24a](#_1ljsd9k) Initial System Analysis and Class Identification 77

[24b](#_45jfvxd) Dynamic Modelling of Use-Cases 77

[24c](#_zu0gcz) Proposed System Architecture 77

[24d](#_3jtnz0s) Initial Subsystem Decomposition 77

[25](#_1yyy98l) Additional Design Considerations 77

[25a](#_4iylrwe) Hardware / Software Mapping 77

[25b](#_2y3w247) Persistent Data Management 78

[25c](#_1d96cc0) Access Control and Security 78

[25d](#_3x8tuzt) Global Software Control 78

[25e](#_2ce457m) Boundary Conditions 78

[25f](#_rjefff) User Interface 78

[25g](#_3bj1y38) Application of Design Patterns 79

[26](#_1qoc8b1) Final System Design 79

[27](#_4anzqyu) Object Design 79

[27a](#_2pta16n) Packages 79

[27b](#_14ykbeg) Subsystem I 79

[27c](#_3oy7u29) Subsystem II 79

[27d](#_243i4a2) etc. 79

[IV](#_j8sehv) Project Issues 80

[28](#_338fx5o) Open Issues 80

[29](#_1idq7dh) Off-the-Shelf Solutions 80

[29a](#_42ddq1a) Ready-Made Products 81

[29b](#_2hio093) Reusable Components 81

[29c](#_wnyagw) Products That Can Be Copied 81

[30](#_3gnlt4p) New Problems 82

[30a](#_1vsw3ci) Effects on the Current Environment 82

[30b](#_4fsjm0b) Effects on the Installed Systems 83

[30c](#_2uxtw84) Potential User Problems 83

[30d](#_1a346fx) Limitations in the Anticipated Implementation Environment That May Inhibit the New Product 84

[30e](#_3u2rp3q) Follow-Up Problems 84

[31](#_2981zbj) Migration to the New Product 85

[31a](#_odc9jc) Requirements for Migration to the New Product 85

[31b](#_38czs75) Data That Has to Be Modified or Translated for the New System 86

[32](#_1nia2ey) Risks 86

[33](#_47hxl2r) Costs 87

[34](#_2mn7vak) Waiting Room 88

[35](#_11si5id) Ideas for Solutions 89

[36](#_3ls5o66) Project Retrospective 90

[V](#_4kx3h1s) Glossary 90

[VI](#_302dr9l) References / Bibliography 91

[VII](#_1f7o1he) Index 91

### 

### Project Description

## Project Overview

Game named ‘Ikon Wars’ is to be developed two team competes with different avatars aka as characters. The main purpose of this game development is to combine the different qualities of players. For example, the best player is who has education skills like critical and analytical thinking, problem solving skills, interpersonal skills like teamwork, communication, has sarcastic values with how the character is designed etc. The game will enable the player to enhance various skills that he might use in various situations of real life.

The target audience for this game will be for teenagers with age range of 15- 30 years. However, there is no age restriction used. The game is played on online server. Design for this game should contain artistic and creativity for designing characters and making features look more interesting.

Further, we may need to add some problems, puzzles, hints which involves educational values to the game. The title of the game is ‘Ilkon Wars’, but doesn’t have need to have aggressive violence to defeat the player. A player can’t challenge other players with his guns, arms etc, but also with other strategies and powers that can defeat other players without involving violence.

Overall, the project is designed with various features like character build, different levels of the game, real world maps etc. In this game, character designing all will be built before the start of the game. Once the game starts, player gets certain frame time to solve puzzles, riddles and other skills to get the artifacts in their possession. After that, players with similar skill level will be matched in the teams. After that, two teams will be competing.

## The Purpose of the Project

Purpose of this project is to promote the individual growth in all dimensions which includes educational, artistic, and other interpersonal skills.

### The User Business or Background of the Project Effort

There are many online games that have been criticized for the use of violence and brainwashing the minds of teenagers. Because of that many schools in some countries ban that game. Our game should not follow in that category. Whenever any kid or teenager plays the game, no parents or teacher should feel regret about that. The target audience should be very concrete and specific. When all development values are included, audience is wide with less restriction.

Many games are banned with excessive violence which damages the mental health of client to some extend. Our deliver product should enable the player to gain some skills which he/she never had before. Once we reached that, the business will be enabled from all the corners.

Can a student build his aptitude by solving only math problems? Participating in academics is not the only way to increase your talent. Many are struggling in leadership and communication skills. Without solving above problems and visions, the project won’t be started.

Games normally tends to become over additive. Addiction in studying for hours is also harmless. So we are considering certain ways to restrict the user to play the game for only certain hours. Hence, we want the universal acceptance of the game.

### Goals of the Project

Project is carried with vision to sharpen the skills of client in all aspects. The main goal of this game is to combine entertainment values with educational values. Getting artifacts will required the user to solves clues, puzzles and logical reasonings. Further, creating avator will build the creativity side by side. Moreover, during the war, players will require strong teamwork, communication skills and leadership quality. Hence, product is meant for overall personality development of the client.

Special Consideration: This product should also encourage the educational department and social organizations to keep an eye on this.

### Measurement

Several statistics will be made for measuring the success rate for this product. The key measure at the initial stages is to look for how many customers are buying this product in certain time frame. We want to keep track of the graph and see how it is increasing. Next, we want to see that whether the product is accepted universally or not. For example, our age range is from 15-30, but we want to see whether the older people will be able to play or not. Not only this, we also want to check and see whether the game is attracted by other class of people like teachers, elder people, and other critics. We need research team to carry out this process. Other innovative measure we will do is to take short aptitude test for the clients at different stages of the game played. If we somehow able to show that clients are becoming smarter and more intelligent, many doors will be able to market this product in better way.

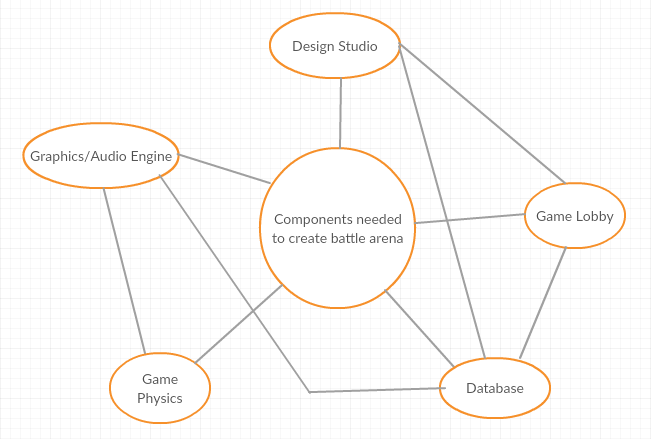
## The Scope of the Work

There are many different maps that users can play on. Each game provides a different experience as there are multiple maps and many different kinds of players can join and play together.

### The Current Situation

This can be published on Steam so; it can be used on Mac and windows. For enthusiasts, this can be costly as it requires playing and winning the games to increase the matchmaking rank to reach on top. These locations can be very large or small. For large maps time limit can be added that will force the avatars to fight with each other. Not only is this, organizing groups of people who can play a virtual game on a real life map near to impossible. In addition, the game will allow users to host an Ikon War server anytime of the day from the comfort of their own homes. Players can save replays of their matches so they can watch it later and learn what they did right or wrong in the match.

### The Context of the Work



### Work Partitioning

|  |  |  |
| --- | --- | --- |
| Event Name | Input and Output | Summary |
| Create Account | User will enter their account info into the database | Allows a player to create and account with his unique ID that will be used in game play |
| Select Map | User will select the map that he wants to play in. | Allows players to download a map from the map database. |
| Find Match | User selects the region that he is in. | Allows players to play in their own region and find players similar to their rank. |
| Start the game | Input: game info from the database.  Output: render information and create game using the game and media engines | When a game is started, information about to the players and the specific scenario being played are sent to the game engines to be rendered and allows the users to play the game. The engines will aid in creating the life-like feel and immersive experience we are aiming for |
| Select Avatar | User will select an Avatar from the database for one game | Allows player to select and apply cosmetics linked to their account. |
| View Profile | Select profile from database | Allows player to check their stats/current progress and watch their old recorded games |
| Watch a game | Server | Allows player to spectator a game that their friends are playing. |

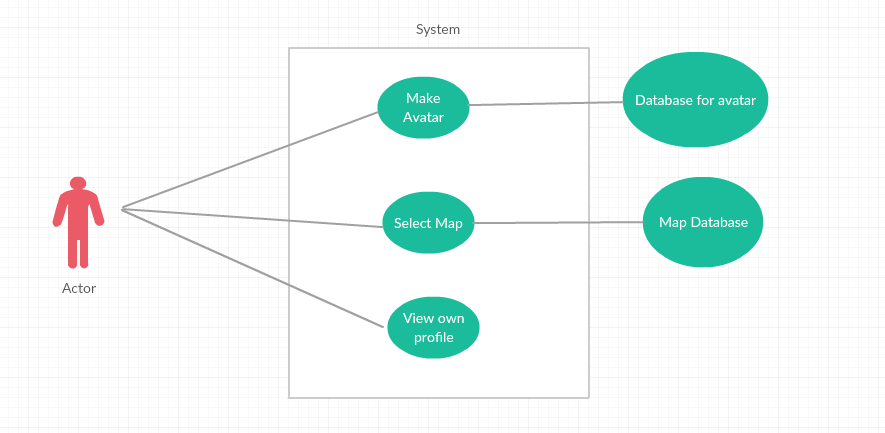
### Competing Products

After doing much research, we found there are other products in MOBA(multiplayer online battle arena) but, we are introducing change in maps and different avatars that are not in the market yet. So it will help players to interact with the real world maps that they live in. Users can create their own virtual world inside their real world maps.

## The Scope of the Product

The main focus of this game is to develop a virtual world inside the real world maps.

### Scenario Diagram(s)



### Product Scenario List

* + - 1. Single Player
      2. Party Game
      3. Bots

### Individual Product Scenarios

Single Player: - In single player game user is matched with other 4 players in the team. Other players will have similar match making rank as the user. Server will find teammates as well as 5 opponents and place them on the map to play.

Party Game: - In party matches user can play with their friends it can be 2, 3, 4 or 5 players. Server will find other players accordingly.

Bots: - In this user will play with the predesigned algorithm which will have 3 levels crusader which will be an easy level. Legend which will be medium level and Immortal will be the hard level.

## Stakeholders

### The Client

The client is the company that are really interest to spread ideas of the map-based game. and motivate how is this interesting while playing. They could start finding target with startup or big company like PS4 or XBox because they have plenty of experience about making games and marketing.

### The Customer

The game customer are the more than average people that love to fight. This game are multiplayer because of that one player connect with other player and make a team while playing game. In addition, Two teams are make challenges and get rank based on who won or lose.

### Hands-On Users of the Product

● User category: Teen age or older.

● User role: Role is play the game

● Subject matter experience: No specific knowledge required to know how to play game, just follow instruction if user is new.

● Technological experience: User need to know how to use computer, but there will also be instruction to guide the user.

● Other user characteristics: The User should enjoy the game with different maps, and different Avatar. Also, some maps are little changing to play. After all, user feels enjoyable and challenging game.

### Maintenance Users and Service Technicians

The people who will work with product maintenance they should know if they find any bug then how to troubleshoot. Also, they know about how to update game.

### Other Stakeholders

Company name: Mixer

Mixer is video game live streaming platform. This company serve large amount of users who enjoy watching other people’s content live. This platform provide good example for streams, who are really interested to playing game. Also, how enjoyable this game and how challenging. In addition, viewers are provide their reviews regarding game.

### User Participation

The user must have computer and some knowledge about how to use computer. There is no additional requirement need accept follow instruction in game.

### Priorities Assigned to Users

● Key users: Simple computer knowledge of use.

● Secondary users: Player will recommend to other players and playing in team is very enjoyable.

● Unimportant users: N/A

## Mandated Constraints

### Solution Constraints

*Description:* The product shall allow to play Ikon game in Single player, Party Games and Bots.

*Rationale:* The client can select avatar and map, this are available with game. There are not other requirements to pay for any extra maps.

*Fit criterion:* Starting menu show display about Single player, Party Game or Bots. User can select anyone. Also, select which map user want to play game.

*Description:* This product work with Window operating system and Mac operating system.

*Rationale:* The client does not have to change their operating system.

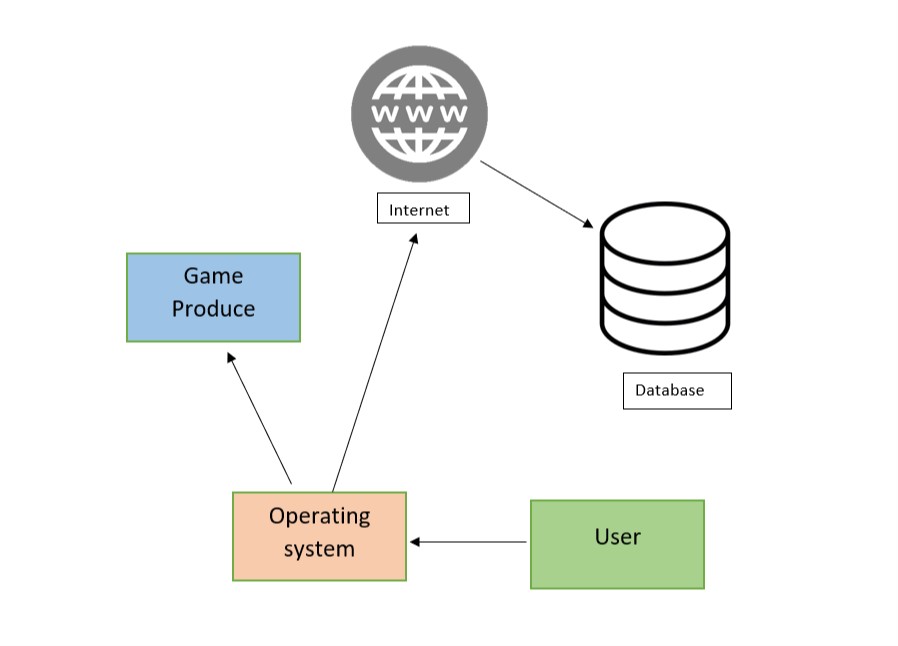
*Fit criterion:* This product connects between Window to Mac or Mac to Window OS. There is not require any particular OS.

*Description:* The product shall allow users to quit in middle of the game, even quit when playing in Party Game, and later resume game. Progress will be saved.

*Rationale:* If Party Game player may not have enough time to play, still they can keep on side their progress and resume later. Between those times, user can play single player game or start Party Game player with different user.

*Fit criterion:* User does not lose any progress with any player.

### implementation Environment of the Current System



### Partner or Collaborative Applications

*Game Engine:* We need connect game engine with Google map to render 3D objects to display map and avatar. We get from another company.

*Game Server:* Product need server. User can save progress and game per player connect with each other and make teams for game, for all this product need server. We can get from another company.

### Off-the-Shelf Software

Content:

Does not need any off-the-shelf software is required for this product.

Motivation:

The product shall handle all requirements such as single player, party game, bots and save progress without depending on any off-the-shelf software.

### Anticipated Workplace Environment

The workplace environment for this product is anywhere. User can access computer/operating system with product install. Internet connection does not require but user want more enjoy with party game than Internet connection is recommended.

### Schedule Constraints

The product requirement will met on its schedule release, as well as any post -release update the product may have.

### Budget Constraints

The budget would depend on project developer team for this product. In Addition rent a database and maintaining after post launch.

## Naming Conventions and Definitions

### Definitions of Key Terms

Player: Human controlled entity, interacts with other players or bots within the map.

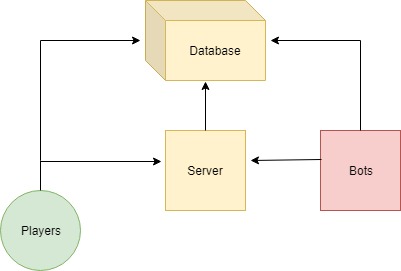
Map: The environment where the match takes place.

Match: A period when entities are allowed to interact with each other. Max players varies by match.

Spectator: Human controlled, allows users to watch matches.

Bot: AI controlled entity, interacts with other players or bots within the map.

### 7b UML and Other Notation Used in This Document



### 7c Data Dictionary for Any Included Models

Database: Holds information of all entities, bots, players, and the match.

Bot Difficulty: Ranges from Crusader to Immortal. (Easy -> Hard)

## Relevant Facts and Assumptions

### 8a Facts

Players and bots alike will aim to bring their team to victory each match.

### 8b Assumptions

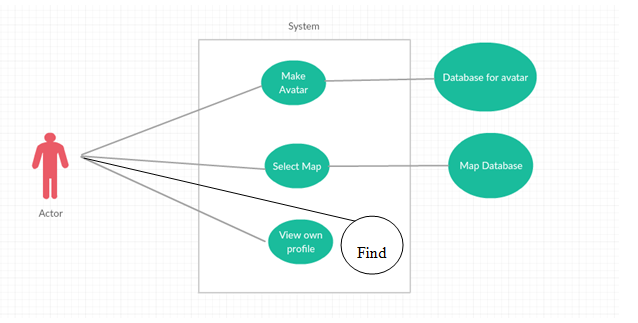
Players will connect to online servers during online ranked matches.

Players will host their own local server if they choose to play solo with bots.

# Requirements

## Product Use Cases

### Use Case Diagrams



### Product Use Case List

* Make Avatar
* Select Map
* View Profile
* Find
  + Normal Match
    - Single Player
    - Party Match
  + Ranked Match
    - Single Player
    - Party Match
  + Bots

### Individual Product Use Cases

|  |
| --- |
| Use Case Name: Make Avatar |
| Actors: User |
| Sequence of Events   1. User can select the Heroes tab from the top of menu 2. User can select what map he wants to play on. |

|  |
| --- |
| Use Case Name: Select Map |
| Actors: User |
| Sequence of Events:   1. User can go in the main Menu to download a map from the database 2. User can select what map he wants to play on. |

|  |
| --- |
| Use Case Name: View User Profile |
| Actors: User |
| Sequence of Events:   1. User can go on the top tabs to view Profile. 2. User can view his stats. 3. User can view his rank. 4. User can view most played heroes and their stats on selected hero. |

|  |
| --- |
| Use Case Name: Find Match |
| Actors: User |
| Sequence of Events   1. User can select Normal Match, Ranked Match, Bots. 2. User has 2 options in normal and ranked matches. 3. In Normal games User is matched with players in region regardless of rank. 4. In ranked games User is matched with the similar ranked players. 5. In Bot games User just plays with the Pre defined lineup. |

## Functional Requirements

**Requirement#:** 1

**Requirement Type** : functional

**Event/Use case#:** 3

**Description:** The system shall allow the user to create an account

**Rationale:** This allows the user to gain access the product’s services and online

capabilities while letting the user personalize their profile

**Fit Criterion:** The option will be available on product start-up

**Requirement#:2**

**Requirement Type** : functional

**Event/Use case#:** 3

**Description:** The system shall allow the user to login with an existing account

**Rationale:** The user needs to access to their saved progress or work in-game

**Fit Criterion:** The option will be available on product start-up

**Requirement#:3**

**Requirement Type** : functional

**Event/Use case#:** 1

**Description:** The System should allow to select Heroes from the database

**Rationale:** The user needs to access database to select available heroes.

**Fit Criterion:** The option will be available on product start-up

**Requirement#:4**

**Requirement Type** : functional

**Event/Use case#:** 2

**Description:** The system shall allow user to select map from the database

**Rationale:** The user needs to access the database to download available maps

**Fit Criterion:** The option will be available on product start-up

**Requirement#:5**

**Requirement Type** : functional

**Event/Use case#:** 4

**Description:** The system shall allow user to find the match in their region

**Rationale:** The user needs to access to the server to find the match

**Fit Criterion:** The option will be available on product start-up

**Requirement#:6**

**Requirement Type** : functional

**Event/Use case#:** 4

**Description:** The system shall allow the user to find match according to rank

**Rationale:** The user needs to get the match according to his rank

**Fit Criterion:** The option will be available on product start-up

## Data Requirements

Two Database are required to fetch information to play the game.

First database is needed to keep the different types of heros. This database is not that space occupied so it will also store user information that will keep track of the stats. Its easy with the heroes because each stat is merged with the heroes.

The first database can also be created to make the AI. Most picked hero, Hero meta can me determined from the same database.

Second database will he for the Maps. This is the one with most of the space because each map is customized according to the virtual game. As the live map has many different places that cannot be utilized for the game. But, the database can store all the maps accordingly.

## Performance Requirements

### Speed and Latency Requirements:

* Game should open quickly within 15 seconds and load all the data that required for the game within next 10 seconds.
* Database for different varieties of avatar must be load in next 5 seconds to allow the user to create his own avator with no lag.
* Database for selecting maps to play the game should be available in 5 seconds.
* After creating avator and selecting maps, player should quick load the game through other submenus within 15 seconds. If the player has to connect online for multiplayer battle, then server should respond in atmost 10 ms ping.
* During the game, there should no lag and game’s status should be updated as efficiently as it can.
* Every match should run with speed, and data and records after the match must be recorded in database.
* User interface for the game should respond back in 1.5 seconds.

### Precision or Accuracy Requirements

* Recorded data after the match and player’s profile must be accurate.
* Database for maps and avators should be precisely available even after updates.
* Database systems to record the played matches, total games bought by the players, total players join in the server should have precise information which must be available to the client.
* Server should be available 24/7 to host the game during online battle.
* Game should be played by the user as precisely it was designed in game engine. For example, actions, movements, storage, etc.

### Capacity Requirements

* System should handle all the database for maps and avators and accurate available for the users.
* Game shall cater for 300 simultaneous users within the period from 9:00 A.M. to 11:00 A.M. Maximum loading at other periods will be 150 simultaneous users.
* Server should hold upto 1 million players atleast through different geographical locations.
* Database should have atleast 10 million records entry preserve for the players. If it is in demand, then it should be double over the course of time.
* User must be given 500mb data to store his avator, maps, recording matches, and various statistics.

.

## Dependability Requirements

### Reliability Requirements:

* Game should easily retrieve data from avator and map database without getting server or unknown load problem.
* Game should not crash during the course of the game. Even during the game, where they are multiple players playing should not cause a bug or error to any of the player during the price poll tournaments.
* During the tournaments, if game crashes, then game must be immediately stopped throwing exceptions to all players saying, ‘Game has crashed and resume as soon it is fixed’. However, it should save the current status of all the players accurately and then resume. A careful consideration must be taken since many players have invested money for the tournaments. Even in the course of big problem, tournament must be cancelled, and each player must be refunded money invested for the tournament.
* Player’s data and their own profile must not be misused or corrupted in any sense.

### Availability Requirements

* For offline players, game is available during 24/7 day and throughout the year.
* For online players, it is available during 24/7 day and throughout the year; however, during the heavy load on server or any kind of maintenance requirement, developer must informed users in advance noticing, ‘Game is under maintenance and will be available after 2 days’.
* Server should response user or give any feedback during the problem caused in money priced tournaments.

### Robustness or Fault-Tolerance Requirements

* User must be given a quick option to message to the developer during the crash of the game.
* During the important tournaments, developer must resolve important issues causing crash of the game. Developer team must be available and prepared to resolve those unexpected issues.
* During the weak connections, player must be informed before entering the tournament prompting, ‘Please make sure you have good connection’.
* When the game is under maintenance during big issues, it should be available locally. Developer should informed the exact dates and times when the game is available online again.

### Safety-Critical Requirements

* Statistics and data of all the players must not be corrupted or sell by any third party.
* Players must be restricted for only certain hours to play the game. For example, player may be restricted to play for eights hour a day max. For small children, parents are given authority to impose more restriction. This avoids the addiction of the player.
* Further, weapons in the game must not be brutal enough. If the weapons are violent and brutal in certain course of game, user must be given restriction according to his/her age. For example, if the user is under 15 years, he will not play those games which have little to no violent weapons. This is to make sure the mental health of small children.
* Game must be developed such that it doesn’t cause over heat to the phone, and if it is then user must be prompted immediately.

## Maintainability and Supportability Requirements

### Maintenance Requirements:

* Game must be updated with new avators and maps during certain course of time.
* Game must be updated with new added features every two months to keep the software trending in market.
* Every six months, game must be put under maintenance to eradicate errors, bugs and flaws and improve the performance and sideby keeping more optimized.
* Developer must ensure that there is no duplicate identity of the user, and user’s identity must not be sell in any course.

### Supportability Requirements

* Game must have pages of manual to show how the game works in detail.
* Game must have other practice sessions and tutorials. For online servers, before the player entered in a big price match tournament, every player may be given chance to test their product with connection and audio.
* 24/7 online staff must be available to support the user for issues and problems.
* For abusive language and spams, support staff should do quick action.

### Adaptability Requirements

* Game is supported in machine with dual core processor with four gb ram, and three gb graphics card.
* User can enhance their grahics of the game according to their own machine capability. For example, machine with high ram and high graph card can have more advance visuals and graphics.
* Game is supported on Windows 10, Mac 12, and Android latest version.

### Scalability or Extensibility Requirements

* During launch, game must support volume of 1 million customers.
* A scalability formula will be design during the finishing of the development process. This formula will ensure how much is game trending and how much volume it should support during the next year. Our statistics team will be managing this.
* Database for maps, avatars and profiles of user should be extensible according to the growth of the customers. Everything will be dynamically managed by the scalability formula.
* Server and system requirements should also support the over heat and load during this rush.

### Longevity Requirements

* Game supports player with multiple age groups, with convenience available for all systems and machines. Game has many online district, state and national based tournament available during certain course of times. Awards and price ceremony will be hold personally by developers to accolade the winners.
* Game should span minimum of seven years as per five million budget spend on it.

## Security Requirements

### Access Requirements

**User and Product owner**

**Description:** User only access their own account.

**Rationale:** User select their own Avatar, Map and Scores.

**Fit Criterion:** Developer staff or owner of product can add Maps, Avatar, and other data

**Acceptance Tests:** User can access personal account and owner or Developer select can change additional features.

### Integrity Requirements

**Accident corruption**

**Description:** By accident or someone tries to hack system product able to protect itself.

**Rationale:** User data store as encryption, so its protected very strong.

**Fit Criterion:** The product shall protect itself from internal abuse.

**Acceptance Tests:** Protect information itself.

### Privacy Requirements

The product shall get permission from user before take information. In addition, product owner make promise to user he will not abuse information.

If information will update or change, first notify to user.

The product shall not share any information to any third party. If in future require to share information, first take permission from user to share details.

### Audit Requirements

**Product retain**

**Description:** Product keep many information.

**Rationale:** Product have server information, network and user personal information.

**Fit Criterion:** Product keep information so, Audit is required.

**Acceptance Tests:** Audit by product owner or other organization by law.

### Immunity Requirements

**Security Update**

**Description:** Product connect to internet.

**Rationale:** On internet transferring data, that warns viruses may come.

**Fit Criterion:** Check online antivirus when downloads data.

**Acceptance Tests:** Keep product updated with security update.

## Usability and Humanity Requirements

### Ease of Use Requirements

**First time user:**

**Description:** First time user follows the instruction for use this product.

**Rationale:** For first user know, what is in game.

**Fit Criterion:** User actual play game and enjoy game.

**Acceptance Tests:** User should not miss guided when use first time product.

### Personalization and Internationalization Requirements

**Language Preference:**

**Description:** This product is available globally.

**Rationale:** In USA default language should be English.

**Fit Criterion:** South korea speaking Korean same as many countries so player connect online in same country.

**Acceptance Tests:** Default language should be based on region.

### Learning Requirements

**Learning:**

**Description:** Learn product for first time.

**Rationale:** First time Make avatar and select maps.

**Fit Criterion:** If user follow the tutorial, then easy to understand.

**Acceptance Tests:** Tutorial is not mandatory but highly recommended.

### Understandability and Politeness Requirements

**Understandability:**

**Description:** User can not see details how product is actual work.

**Rationale:** In the product user easily familiar with used symbol.

**Fit Criterion:** This product available globally, so far symbols are available with description when mouse cursor goes over symbols.

**Acceptance Tests:** User should not be thinking bit for what is symbols is doing.

### Accessibility Requirements

**Affects of color and volume:**

**Description:** Change color and give warning before makes to high volume

**Rationale:** User are colorblind and warning for high level volume.

**Fit Criterion:** If users are colorblind than need to color change and if user make high volume than user get warning of high level of volume.

**Acceptance Tests:** Please follow government guideline for this features.

### User Documentation Requirements

**Manual:**

**Description:** Product come with manual.

**Rationale:** How to start with game and for major direction in game.

**Fit Criterion:** Manual is mandatory with product.

**Acceptance Tests:** Manual also available as e-manual on website.

### Training Requirements

**Training/ Tutorial**

**Description:** In the game there is tutorial.

**Rationale:** First time user doesn’t feel uncomfortable.

**Fit Criterion:** In the game tutorial must available that make product familiar.

**Acceptance Tests:** Tutorial should cover every single part of game.

## Look and Feel Requirements

### Appearance Requirements

**Looks:**

**Description:** Color combination and graphics should attractive.

**Rationale:** If color combination not attractive than user feel old fashion.

**Fit Criterion:** Color combination and graphics feel more enjoyable game.

**Acceptance Tests:** First time see product by customer feel this product will be awesome.

### Style Requirements

**Theme:**

**Description:** Theme play significant for while playing game.

**Rationale:** Theme reflect originality of game.

**Fit Criterion:** Theme change while select Avatar. Also, add music that also change as theme.

**Acceptance Tests:** User not feel every avatar is only look difference, also have different powers as well.

**18 Operational and Environmental Requirements**

**18a Expected Physical Environment**

* Game is expected to be played via desktop computer, or laptop. Internet connection is required for online play.
* Used by all age groups
* Indoor leisurely use

**18b Requirements for Interfacing with Adjacent Systems**

* Internet connection is required for online play.
* Other programs can run in the background while Ikon War is running.
* System requirements are not steep, very basic computer can run this product.

**18c Productization Requirements**

* Product will be distributed via .zip file
* Product will be small enough to fit on a single CD
* Product installation will be very user-friendly

**18d Release Requirements**

* If any bugs / glitches are found, patches will be released as soon as possible

**19 Cultural and Political Requirements**

**19a Cultural Requirements**

* Product will not contain any offensive material in respect to religion or ethnicity

**19b Political Requirements**

* Product will contain no restrictions. to be used on all operating systems.

**20 Legal Requirements**

**20a Compliance Requirements**

* Personal information will not be shared, copied, or distributed, as in compliance with Data Protection Act
* Credit to authors of pictures used will be given

**20b Standards Requirements**

* Product will comply with standard videogame distribution laws

**21 System Design**

**21a Design Goals**

There are many design goals for this product.

Server should be available 24 hours a day and 7 days of week. Because, User play game any time any region.

Regular update comes for Avatar. This avatar should able to change color. Also, Avatar available in Males and females. Change theme and background music when change of Avatar selection.

This product should be available on Window, Mac, and Linux.

The system should be recoverable, if user crash from some point of the game or power gone. This product starts from either start over or resume they left.

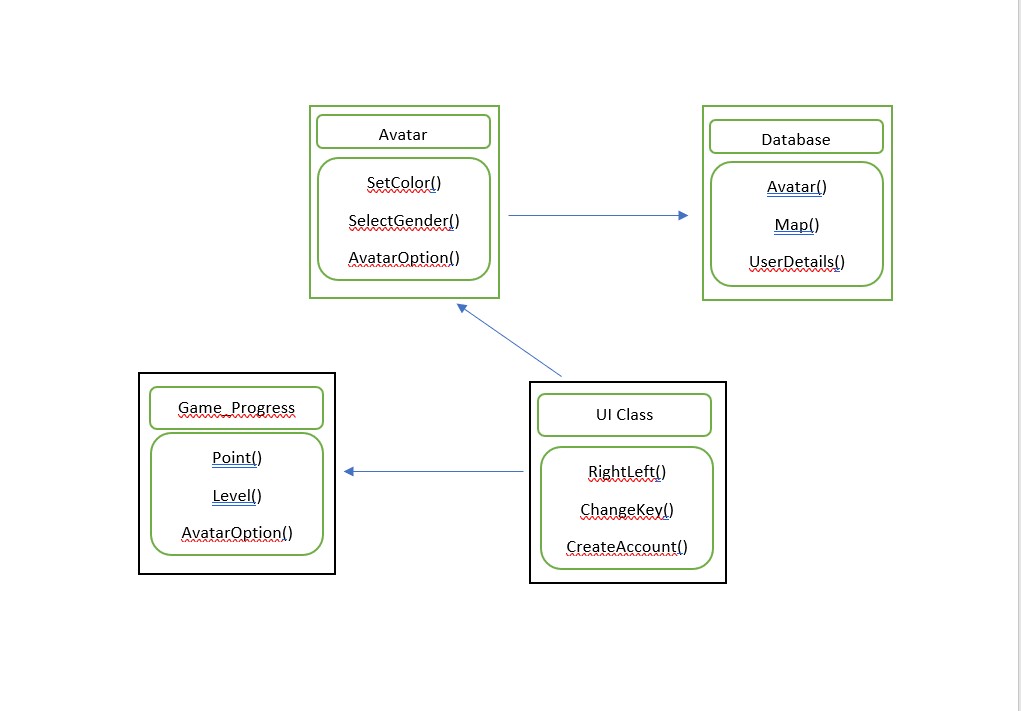
There should be feedback box pop up after user keep play game. From, receiving feedback product should improve or update in future.

## Current System Design

There is no pre-existing system needs.

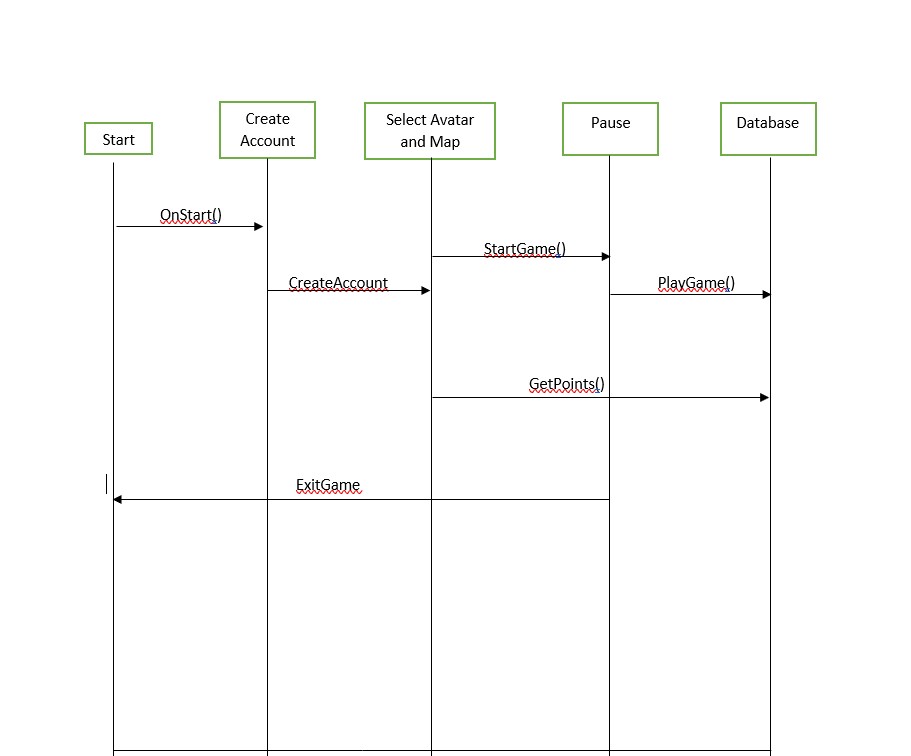
## Proposed System Design

### Initial System Analysis and Class Identification



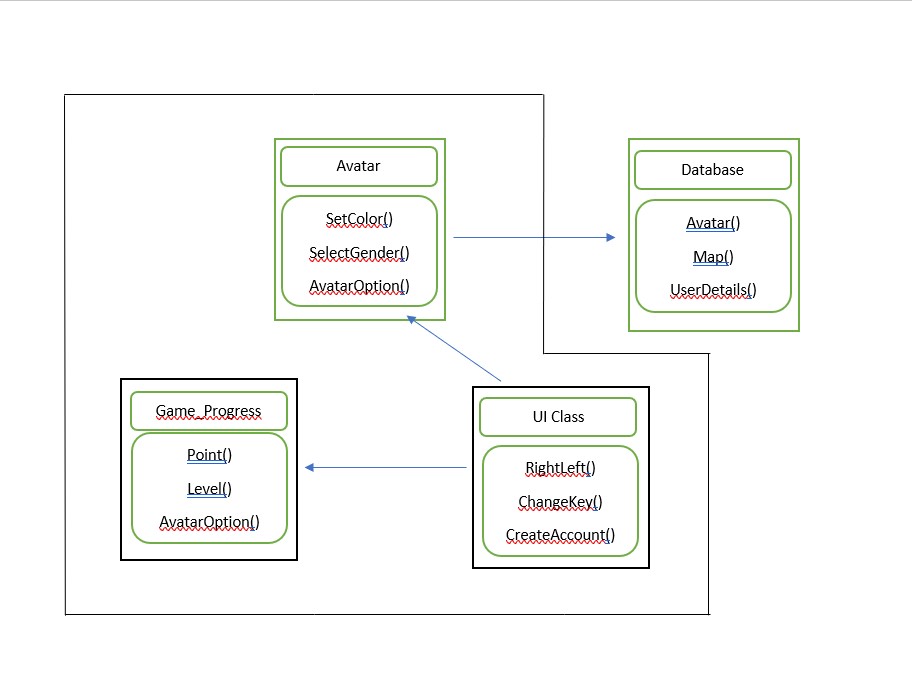
Simple class diagram shows how user interact and how database work in product.

### Dynamic Modelling of Use-Cases



Basic cycle for Game.

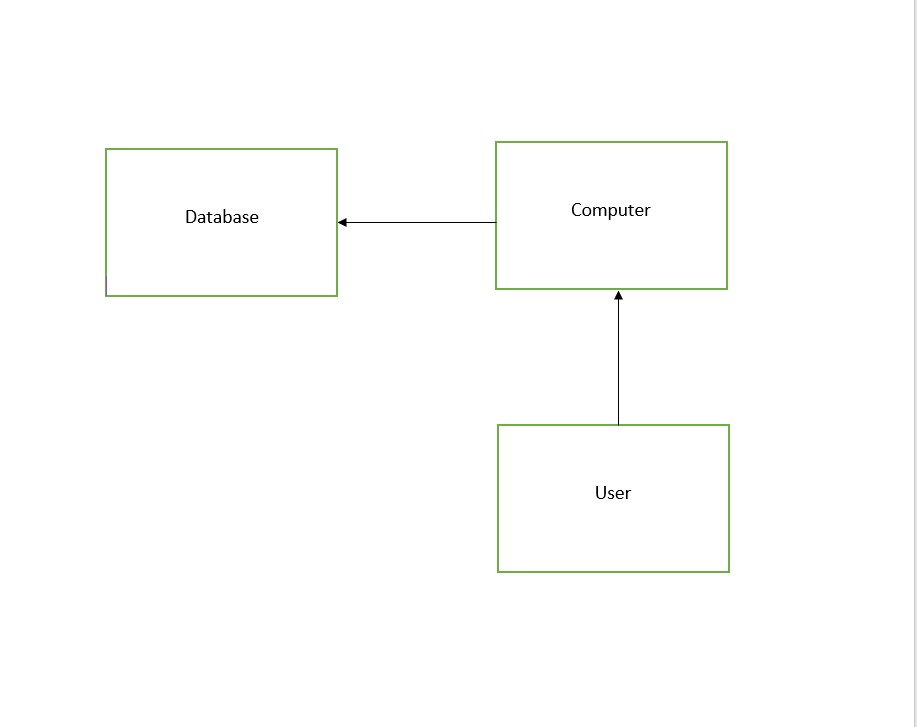
### Proposed System Architecture



This demonstrates that the user doesn’t have direction communication to the database.

## Additional Design Considerations

### Hardware / Software Mapping



User connect Database via computer. Same way he access database but user need computer for access.

### Persistent Data Management

The data of Avatar and Maps stores into NoSQL database. This storage is not relational. If new update available for Avatar and Maps, still old record remains same. Avatar data contains color, background music, weapons and many more items

### Access Control and Security

User only can see details of database and use that database. User can not anything change or update those data are on server.

### Global Software Control

The game is designed to have multiple player. Those players interact with each other and all those access data from database. Basically, handle large amount of data. For that amount of data use need dedicated server.

### Boundary Conditions

If the server is going to maintenance, then inform the user ahead of time about the time of start and end for the downtime of the server, all information’s should be handled correctly so on informations are lost.

### User Interface

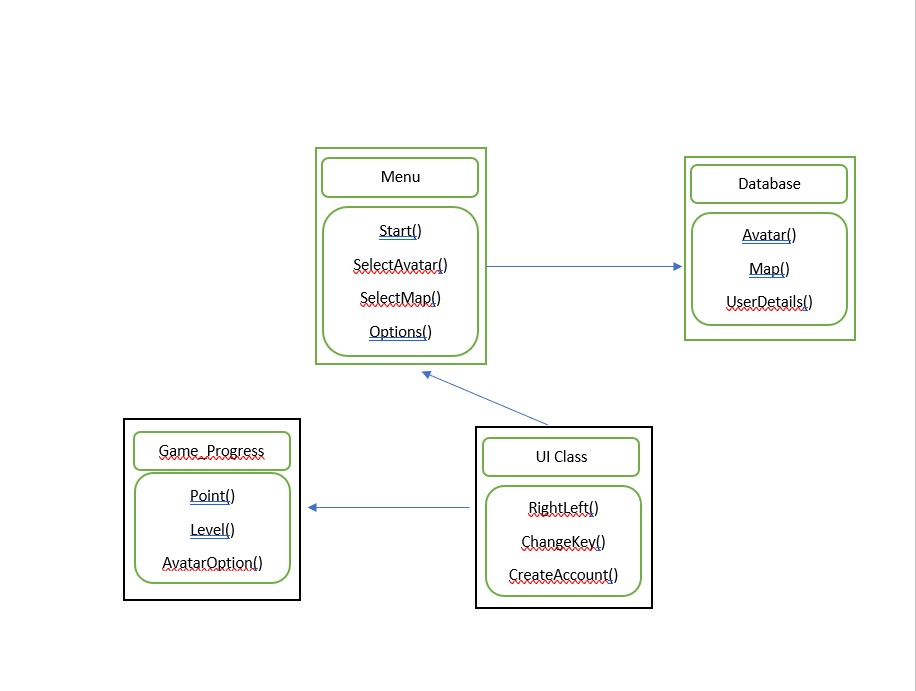
This game play by keyboard as well as mouse. There should be option for user he can change setting for User Interface. Example, if user is righty user then according to change setting on mouse click and keyboard shortcuts. Same as for leafy user. Change setting according to user.

In addition, there are many options are available to change, suppose, key settings, and shortcut keys.

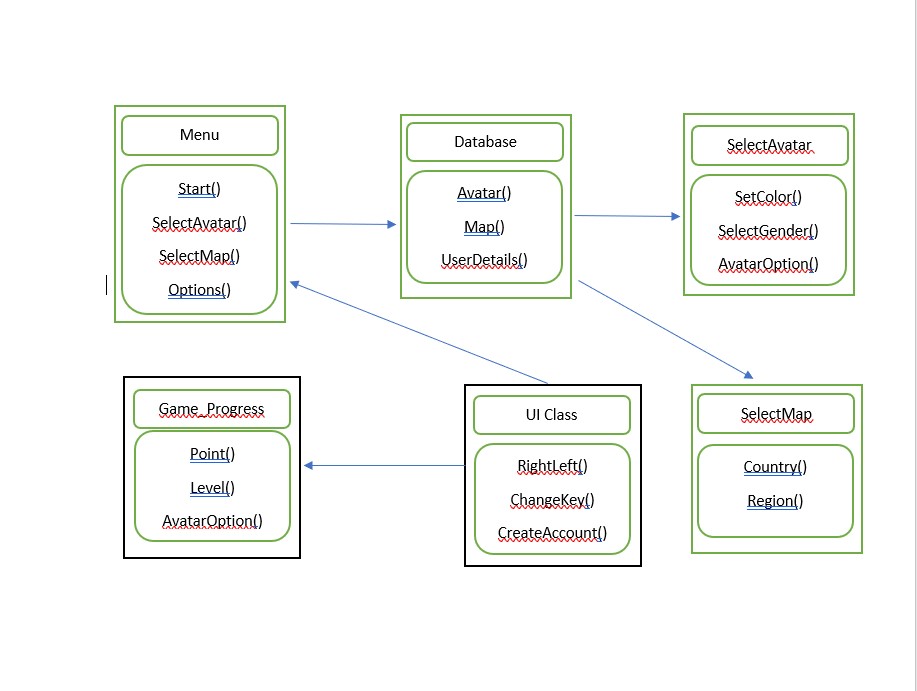
Also, there are option available for music and background sound. Same as brightness option need and regarding setting shows recommendation about settings.

### Application of Design Patterns

Application patterns there



## Final System Design



## Object Design

### Packages

Each package should contain documentation on which classes and sub classes belongs to it. Classes also contain information about which other classes are connected to this class and

# Project Issues

## Open Issues

This Game played by any computer. But, we planning to add features for the graphic card in computer. If user’s system does have graphic card then, they get better resolution while playing game.

## Off-the-Shelf Solutions

### Ready-Made Products

Avatar and Maps remain stores in database. This Avatar and Map can find readymade. Also, creating new Avatar looked old and what is coming in updated that also give helps while creating new shape of Avatar.

### Reusable Components

Any libraries found from open source and themes for game that are reusable components.

                  Update game should not change Database design.

### Products That Can Be Copied

There are many wars game with advanced features cames in last few years. But adding Avatar and Maps features that will attract people. They first look in our game is what are the new.

Also, we will keep update for new design, look and new features, that keep together our users.

## New Problems

### Effects on the Current Environment

Because our game is online based, any latency experienced server side could create problems for players. This includes inconsistent animations, actions/consequences occurring incorrectly in-game, player disconnects, as well as general client-side lag. Having clean, efficient code, as well as a dependable server host, is the key to keeping online gameplay smooth and consistent.

Upgrading in-game models could alter how those models react when animations are given. Clipping issues and maybe odd bugs (limbs detaching, flailing, etc.) could occur. Such issues could ruin player visibility and create inconsistencies in game. For example, players may not be able to tell what the enemy’s next move is, since the animations could be broken. This creates an unfair advantage for the enemy, since their moves are undecipherable by the players.

### Effects on the Installed Systems

The new system should have no major problems with the hardware / software we are implementing. Certain graphical issues may arise, such as drop in frame rate, and maybe increasing loading times due to more data having to be processed. However, these are minor issues, and can be fixed if they occur.

### Potential User Problems

As stated before, updates to models, the map, and UI, could cause some drop in frames, and increase in loading times. Features are new, and may not be as efficiently implemented initially.

In respect to character model updates, animations might be bugged due to a new model used (depending if it is created using the same method, or a new one). Maybe inconsistencies with animations and sound files could occur, but this is highly unlikely.

If UI is updated extensively, users could experience problems with pressing certain buttons and screen animations could be off. On-screen UI animations (menu flipping, menu panning, and other flashy animations) are not a priority, as functionality is regarded as more important. If those features aren’t working properly, it can be deemed as a lesser problem. Priority is placed on a fully navigable menu.

### Limitations in the Anticipated Implementation Environment That May Inhibit the New Product

If our servers are full and need to be updated, future server size and power estimates can be tricky waters to traverse. How big are we trying to go? How powerful? Will it be enough? Are our players happy?

Upgrades to in-game features might call for hardware upgrades. Not only for developers, but users might feel these affects as well. Upgrades to hardware can create an additional issue of lack of space (if they are that extensive).

### Follow-Up Problems

Keep a close connection with players. They are our play-testers and our market. Their opinions and concerns matter, and we, as developers, must work out a solution with them. We cannot give them everything that they as for, but find a suitable compromise. Our aim is to keep the game consistent and fair, making it fun for everyone playing.

## Migration to the New Product

### Requirements for Migration to the New Product

Knowledge of new engine software is a must. If we are using programs such as Unity, full knowledge is needed to make efficient use of our time and efforts.

Additional staff could aid in processing work and efficiently releasing patches to the public. If our game becomes so extensive, our 4 man team will not be enough to keep the game up to date for players. Managing a team will greatly increase our work output.

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

Models and maps could be upgraded using Unity or other engine. This would mean that only the model and map concepts would carry over. In the new engine, models might have to be built from scratch, depending on differencing arising in changing engine.

Character movement, and general UI might possibly remain the same. Hopefully integrating the new models and maps won’t destroy what previously worked.

With a new map, comes new problems. Player traversal of the map might not be as consistent as possible (i.e. sometimes random stops, misplaced walls, etc.). As well as added latency and possible frame drops due to new software being implemented.

## Risks

Many risks and concerns have been brought up in previous sections of this report. They include, but are not limited to:

* After a significant patch is released, latency and frame drops could occur for players
* With server upgrades, players could disconnect from games, experience lag, server might shut down
* Upgrades to character models might cause inconsistencies with character animations
* Map updates could create inconsistencies when traversing it
* If product is so heavily updated, software / hardware upgrades are probably needed. This will increase cost for producing product. This might also call for players to upgrade their systems at home (might be a significant downside for some).

New character releases might affect the balance of the overall game.

## Costs

If updates to our hardware, software, development team, and server are a necessity, our costs will greatly increase. Purchasing enough copies of software, as well as pieces of hardware can be a one-time increase in our budget. However, increasing our development team size and upgrading server provider can be a longer team budget increase. It will increase our monthly cost of maintaining our game.

Switching to a new engine or coding environment will require an investment of time to learn such programs. This is much harder to estimate, as depending on the program, it could take weeks, months, or even years to fully get in touch with it. Maybe the hiring of new employees with such skills might mitigate this time cost.

## Waiting Room

What assets will be reused? Have we established a good baseline for assets being created (reusability, versatility, effectiveness, etc)?

How is the current meta game? Are players having a fun, interactive experience? Or are the bogged down by over powered assets available to them?

If hardware / software is to be updated, specifically which are we switching to? Is this in-line with our budget?

## Ideas for Solutions

If updating character models, maybe it would be wise to not switch software, as it would take too long to implement and create the model from scratch.Also learning the new software could take away valuable time.

Hiring new employees could mitigate the problem of learning new software, however, it would entrust our new associate with our product. We would have to be selective in our hiring.

Feedback, feedback, feedback. We need to take feedback from the community, players, and organizations. Everyone’s opinion matters, but we must make changes according to our own jurisdiction.

## Project Retrospective

Time management could be tweaked. Lots of items released were near the deadline. If this weren’t the case, the product might be a little more refined and polished.

However, work division and implementation went with a success. Overtime, our project grew and grew. New ideas and concepts were wonderfully given by team members, and the project has been influenced by every single one of them.

For future iterations, scheduling issues will have to be dealt with, so our deliverables will be on time, and as good as they can be.

# Glossary

Model: in-game representation of a player character in the game.

Animations: in-game character model movements. They will depict if the model is moving, attacking, casting a spell, etc.

Map: in-game battle arena, where player interaction takes place.

Client: the players of our game will be client-side. They will have to log in, to obtain access to our game server.

Server: the game will be hosted via server. Players client-side will connect to the server

Attack:battle will take place in Ikon Wars. An attack is an action that will deal damage to the opposing players.

Skills: certain characters will have special skills to utilize in battle. (i.e. fireball, ice lance, Chronosphere, etc.)

# References / Bibliography

Bernd Bruegge and Allen H. Dutoit, "Object-Oriented Software Engineeering - Using UML, Patterns, and Java", Third Edition, Prentice Hall, ISBN-10: 0136061257, ISBN-13: 9780136061250

John T. Bell’s CS440 Classnotes

Alessandro Artale, Diego Calvanese, Angélica Ibáñez-García, “Full Satisfiability of UML Class Diagrams”

# Index

Design 61, 63

Requirements 35, 51, 58

Test 64, 65