

# SE 3XA3: Systems Requirements Specification: Revision 0

Team 03, Pongthusiastics  
Adwity Sharma - sharma78  
Arfa Butt - buttaa3  
Jie Luo - luoj3

October 31, 2016

# Contents

<b>1</b>	<b>Project Drivers</b>	<b>4</b>
1.1	The Purpose of the Project . . . . .	4
1.2	The Stakeholders . . . . .	4
1.2.1	The Client . . . . .	4
1.2.2	The Customers . . . . .	5
1.2.3	Other Stakeholders . . . . .	5
1.3	Mandated Constraints . . . . .	5
1.4	Naming Conventions and Terminology . . . . .	6
1.5	Relevant Facts and Assumptions . . . . .	6
<b>2</b>	<b>Functional Requirements</b>	<b>6</b>
2.1	The Scope of the Work and the Product . . . . .	6
2.1.1	The Context of the Work . . . . .	7
2.1.2	Work Partitioning . . . . .	7
2.1.3	Individual Product Use Cases . . . . .	8
2.2	Functional Requirements . . . . .	9
<b>3</b>	<b>Nonfunctional Requirements</b>	<b>16</b>
3.1	Look and Feel Requirements . . . . .	16
3.2	Usability and Humanity Requirements . . . . .	16
3.2.1	Ease of Use Requirements . . . . .	16
3.2.2	Accessibility Requirements . . . . .	17
3.3	Performance Requirements . . . . .	17
3.3.1	Speed requirement . . . . .	17
3.3.2	Safety critical requirement . . . . .	18
3.3.3	Precision requirement . . . . .	18
3.3.4	Reliability and availability requirement . . . . .	19
3.3.5	Capacity requirement . . . . .	20
3.4	Operational and Environmental Requirements . . . . .	20
3.4.1	Expected Physical Environment . . . . .	20
3.5	Expected Technological Environment . . . . .	20
3.5.1	Partner Application . . . . .	21
3.6	Maintainability and Support Requirements . . . . .	21
3.6.1	Maintenance Requirement . . . . .	21
3.6.2	Supportability Requirement . . . . .	22
3.6.3	Portability Requirement . . . . .	22

3.7	Security Requirements . . . . .	22
3.7.1	Confidentiality . . . . .	22
3.7.2	File Integrity Requirement . . . . .	22
3.7.3	Audit Requirement . . . . .	23
3.8	Cultural Requirements . . . . .	23
3.9	Legal Requirements . . . . .	23
<b>4</b>	<b>Project Issues</b>	<b>24</b>
4.1	Open Issues . . . . .	24
4.2	Off-the-Shelf Solutions . . . . .	24
4.3	New Problems . . . . .	24
4.3.1	Problems in the Current Environment . . . . .	24
4.3.2	Existing User . . . . .	24
4.3.3	Limitations in Implementation Environment . . . . .	24
4.4	Tasks . . . . .	25
4.5	Migration to the New Product . . . . .	25
4.6	Risks . . . . .	25
4.7	User Documentation and Training . . . . .	25
4.8	Waiting Room . . . . .	25
4.9	Ideas for Solutions . . . . .	25
<b>5</b>	<b>Appendix</b>	<b>26</b>
5.1	Symbolic Parameters . . . . .	26

## List of Tables

1	Revision History . . . . .	4
---	----------------------------	---

## List of Figures

1	Context Diagram for the Game . . . . .	7
2	Project Flow for the Redevelopment Process . . . . .	8

Table 1: **Revision History**

<b>Date</b>	<b>Version</b>	<b>Notes</b>
October 7, 2016	1.0	Created the SRS document
October 11, 2016	1.1	Completed SRS Rev-0 for deadline submission
October 24, 2016	1.2	Uploaded diagrams
October 31, 2016	1.3	Updated requirements

# 1 Project Drivers

## 1.1 The Purpose of the Project

We have overtaken this project to improve the electronic version of ping pong game we found in an open source projects in git website. The use of computer and any other electronics for entertainment has been growing for several years now. The electronic version of the ping pong game used to be automatically installed with the installation of new windows. But for some years now Microsoft has stopped adding this game to the windows operating system. While, there are variations of this game available for download in the app store, there is still a market for a multi-player ping pong game for entertainment purposes. The game can also be enjoyed individually. It provides the players with a slightly challenging game that does not require any tools other than a computer that has a java running environment. This game can be played among friends to relieve themselves of boredom. This project will also provide the users with a game that can be played by people of any age group.

## 1.2 The Stakeholders

### 1.2.1 The Client

An external entity is regarded the client for this project. This entity is interested in the redevelopment of the project. Though this project is very open ended and can be developed in endless manners, there still are some restrictions that has been applied on the manner of development. This restrictions, in regards to the development has been applied by the 3XA3 team, and thus

can be looked upon as the client figure.

### **1.2.2 The Customers**

Any person with an access to a computer that has a java running environment can be considered as a costumer for this project. As this game does not have restrictions in terms of age, gender or any other socio-economic factor, everyone can be considered as the customer of this project.

### **1.2.3 Other Stakeholders**

We are also the stakeholders for this project, as the development of this project and its result would be a great effect upon us. The general public can also be considered as the stakeholders as through them we are to gain customer base for our game.

## **1.3 Mandated Constraints**

Description: The project is being developed in java language in eclipse and is run in java running environments. Rationale: Java running environments such as eclipse are available online for free for users and thus can be downloaded by any user intending to play the game. Fit Criterion: There shall be a link provided with the release of this game to download a java running environment, for users that do not have it already.

Description: The project is provided online, in an open source environment, for anyone who wants to use it. Internet connection would be required to download the game the first time the user wants to use it. Rationale: The user should be connected to the internet that can assure the downloading of the game. Fit Criterion: As the game only needs to be downloaded once for the user it isn't very hard to accomplish. It only needs to be downloaded once and can be enjoyed for as long as the user intends to use it.

## 1.4 Naming Conventions and Terminology

Terms	Definitions
Users	Players of the game
The Project	The pong game that is being reconstructed.
Product	The game that is being developed.
Java	Java programming language.
Clients	The group for whom who the project is being developed for.
Git	The GitLab website.
Windows	Microsoft windows.
Customer	Anyone who would like to use this game.
3XA3 team	Professor, course coordinators, teaching assistant and any other personnel responsible for running of the 3XA3 course.

## 1.5 Relevant Facts and Assumptions

Facts: There will be approximately 1000 lines of codes for this game by the time this game is completed. The original of the game that is being reconstructed can be found in <https://github.com/mihneadb/Pong>.

Assumptions: It is assumed that the user will have some access to the internet to download the codes that will be used to run this game when the user wishes to play this game for the first time. It is also assumed that the person will either have access to a java running environment or be willing to download a java running environment. Since internet is readily accessible this should not be much of a trifle.

## 2 Functional Requirements

### 2.1 The Scope of the Work and the Product

The developers would rewrite the code into a structured design. By doing this, the work for future maintenance would be easier.

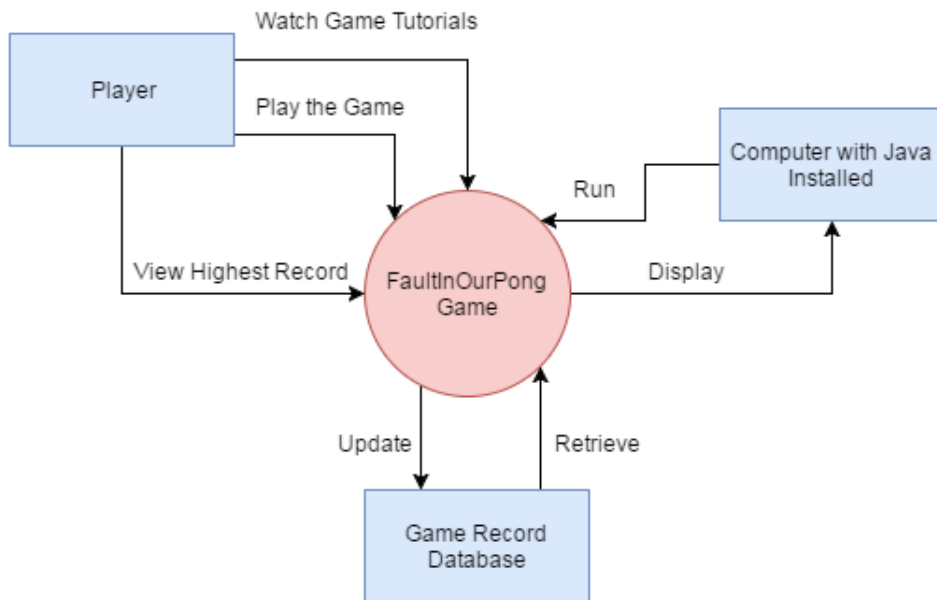


Figure 1: Context Diagram for the Game

### 2.1.1 The Context of the Work

The software application to be developed will follow the same game rules as the original Pong game application. The interface of the new application would be better than the original game. The winning and losing would be better indicated on the screen.

### 2.1.2 Work Partitioning

The Pong project will be following a model similar to the ~~V-Model~~ Water Flow development process. The project will start with developing a problem statement for the new project. It can help team-mates clarify the objective of the work, such as what to improve on the Pong game, and how much we can improve. Team-mates then would like to create a development plan for the Pong game, to make sure each development process is on track and on time. A requirement specification plan is then created to further break down the project into small tasks. This include specifying the requirements for the new software, stating the personnel that is related to this Pong project, and clarifying any concerns and issue related to it. Verification and Vali-

ation Plan would be created in the following, to give team members and other stakeholders a vague idea of what the final product would be. Design Specification Document would be formulated for better clarifications before application implementation. Team-mates would be coding following all the documents mentioned, and Verification and Validation Plan will test the program when it is completed.

The team of three will be working on the redevelopment of the project. Each team member will be assigned different tasks to work on the project. A Gantt Chart will be used to keep track of working activities for each member in the team, that means each member would have to make a mark on the Gantt Chart when they complete a task.

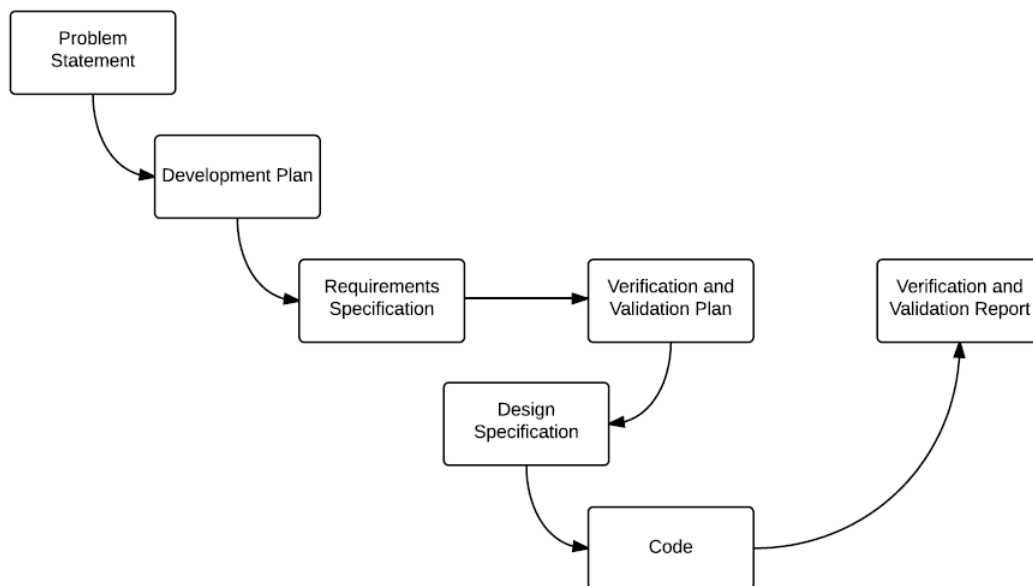


Figure 2: Project Flow for the Redevelopment Process

### 2.1.3 Individual Product Use Cases

Case: User performs mouse click on the Start button

Result: The program would start the game and the game scene would be displayed.



Case: User performs mouse click on the Restart button  
Result: The program would reset the game results and it would display the initial game scene.

Case: User performs mouse click on the Save button  
Result: The program would save the players' record, but the game continues.

Case: User performs mouse click on the Tutorial button  
Result: The program would display a description of the game and teach the user how to play it.

Case: User performs mouse click on the Close button  
Result: The program game would stop, the window of the application would be closed, and the current game result would not be saved.

Case: User taps on UP, DOWN, LEFT, RIGHT on the keyboard.  
Result: The player would be able to move his/her game board.

## 2.2 Functional Requirements

**Requirement number:** FR1

The program should have a button to start a new game.

**Rationale:** Rationale: The user should be able to start a new game in order to play the game.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR2

The program should have a button to load a previously saved game.

**Rationale:** The user must be able to load his/her previously saved game in order to continue the game.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR3

The program should have a button to save the current game.

**Rationale:** The user must be able to save their current game in order to continue playing it at a later time.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR4

The program should be able to display a list of top 20 high scores (name and score).

**Rationale:** The user must be able to open up the high scores list, so that they can see what rank they are on. This will also build healthy competition among the players.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR5

The program should have a button to open up the tutorial page, which will give instructions to the user on how to play the game.

**Rationale:** If the user is playing the game for the first time, they must be able to view the instructions to the game before they can play it.

**Priority:** Medium

**History:** Created October 11, 2016

- **Requirement number:** FR6

The program should have a button to have the option to play a single-player mode, with the computer. The list of high scores will be kept for this mode.

**Rationale:** The user must choose which type of mode of the game they want to play, before they can play the game. This type of mode is for users who want to play alone, or want to beat a certain high score.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR7

The program should have a button for the option to play a single-player mode, with the computer, with obstacles added to the playing field.

**Rationale:** The user must choose which type of mode of the game they want to play, before they can play the game. This type of mode is for users who want to play the single-player mode, but with a little added difficulty.

**Priority:** Medium

**History:** Created October 11, 2016

- ~~• **Requirement number:** FR8~~

~~The user should have the option to play a multiplayer mode, with another user.~~

~~**Rationale:** The user must choose which type of mode of the game they want to play, before they can play the game. This type of game is for users who want to play with a friend, as opposed to playing alone.~~

~~**Priority:** Low~~

~~**History:** Created October 11, 2016~~

- **Requirement number:** FR9

The program should have the option for a user to change the speed of the ball.

**Rationale:** The user must be able to change the speed of the ball to make the game easier or harder for them to play.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR10

At the start of a single-player game, the player should have 3 lives.

**Rationale:** The user must have 3 lives at the start of the game, as that is the standard that we have decided to keep for the game.

**Priority:** High

**History:** Created October 11, 2016

st

**Requirement number:** FR11

At the start of every new game, the scores of the player, or both players in the case of multi-player mode, will be set to 0.

**Rationale:** The user(s) must have a score of 0 at the beginning of each new game, as that is the moment from which the player(s) will start to accumulate points.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR12

The user should be able to move their paddle left and right using the keyboard.

Rationale: The user must be able to move their paddle left and right in order to play the game.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR13

The scores of each player should be increased by 1 whenever the opponent misses a hit.

**Rationale:** The scores must increase by 1, every time the opponent misses a hit, to keep track of the player(s) points.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR14

In single-player mode, whenever the player misses a hit, their life should be decreased by 1.

**Rationale:** The number of lives of the player should decrease by 1, each time the player misses the ball, to keep track of the number of lives the player has left. This is to make sure that the player does not go over 3 lives.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR15

The program should have a button to pause the current game.

**Rationale:** The user must be able to pause the current game so that if they need a short break, they can take one without it affecting the progress of their current game.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR16

The program should have a button to resume the current game (if the game is paused).

**Rationale:** The user must be able to resume the current game so that if the game is paused, it can be resumed from the point where the player left the game.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR17

The user should have the ability to restart the current game.

**Rationale:** The user must be able to restart the current game so that if they are not satisfied with their game, and would like to play the exact same mode again, they may do so by just resetting the game.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR18

In single-player mode, when the player misses the ball 3 times, i.e. the player has run out of all 3 lives, the game will end.

**Rationale:** The standard for this game has been set so that each player gets 3 lives, therefore the game needs to end once the player has used up all 3 of their lives.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR19

In multi-player mode, when one of the players has reached 5 points, the game will end.

**Rationale:** The game needs to end when one of the players in the multi-player mode has reached 5 points as that is the standard that we have set.

**Priority:** Low

**History:** Created October 11, 2016

- **Requirement number:** FR20

Once a game ends, the user should be taken back to the main menu page.

**Rationale:** The user must be taken back to the main menu page so that they can choose which mode of game they want to play, or if they want to access the high scores list.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR21

In single-player mode if the player made it into the high scores list then, at the end of the game before the user is taken back to the main menu, (s)he will be prompted to enter their name so that they could be added into the list.

**Rationale:** The user must be prompted for their name if they break one of the high scores, so that the list of high scores is kept up to date.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** FR22

At the end of the game, a window should pop up to indicate the winning and losing state.

**Rationale:** The user need to be notified the end of the game.

**Priority:** High

**History:** Created October 31, 2016

### 3 Nonfunctional Requirements

#### 3.1 Look and Feel Requirements

- **Requirement number:** NFR1

The game interface should be colorful and vivid.

**Rationale:** The user would be interested in the game if its interface is attractive.

**Priority:** High

**History:** Created October 31, 2016

- **Requirement number:** NFR2

The game frame should be clear for user to play.

**Rationale:** The user should know where the border of the game frame is, in order to move his/her paddle to the proper positions

**Priority:** High

**History:** Created October 31, 2016

#### 3.2 Usability and Humanity Requirements

##### 3.2.1 Ease of Use Requirements

**Requirement number:** NFR3 The game can be played by ninety-nine percents of people who know how to use a computer with



hands.

**Rationale:** This game utilizes only mouse and keyboard.

**Priority:** High

**History:** Modified October 31, 2016

- **Requirement number:** NFR4

The game is easy enough to for a child greater than 10 years old to play. Some practice on the part of the users might be required to play and excel at this game.

**Rationale:** This game should be available for most of the children because the game is focused on the children.

**Priority:** High

**History:** Modified October 31, 2016

### 3.2.2 Accessibility Requirements

**Requirement number:** NFR5

The game can be accessed and executed on majority of computing devices.

**Rationale:** Computers with Java installed should be able to open this game. Meanwhile, as long as there is internet connection, this game should be able for download.

**Priority:** High

**History:** Modified October 31, 2016

## 3.3 Performance Requirements

- **3.3.1 Speed requirement**

**Requirement number:** NFR6

There has to be some space in the user's computer to download the game and a java running environment. The game should be played in a fast computer as it would decrease the entertainment factor if it is lagging.

**Rationale:** As long as the computers with this game available has enough space for caching for Java, the speed of the game should be fast enough for playing.

**Priority:** High

**History:** Modified October 31, 2016

- **3.3.2 Safety critical requirement**

**Requirement number:** NFR7

The program should be able to finish in reasonable amount of time.

**Rationale:** Over-playing the game might cause the user to encounter some strain in the user's eyes. But, this is true for any object that requires an electronic screen to play. There aren't any other health concerns.

**Priority:** Medium

**History:** Modified October 31, 2016

- **3.3.3 Precision requirement**

**Requirement number:** NFR8

The movements of the paddle should be as accurate as actions performed on the keyboard possible.

~~It is important that the game is precise in terms of compliance of the commands sent by the keyboard to the screen. Also, it is equally important to make sure that while the game is going on the position of the ping pong ball is kept into account and any disturbance in hitting of the ball or any other irregularity is imme-~~

~~diately noticed.~~

**Rationale:** Actions performed on the keyboard would directly affect player scores and satisfiability on the game.

**Priority:** Medium

**History:** Modified October 31, 2016

- **3.3.4 Reliability and availability requirement**

**Requirement number:** NFR9

The game shall take less than five seconds to start up on 90 percents of computers.

~~The game shall be made available at all times to anyone who has access to the internet and can be played at any time by the users who have already downloaded a copy of the game in their computers.~~

**Rationale:** The program does not take up a lot of memory spaces or cache on the computer, so the startup should be rather quick.

**Priority:** Medium

**History:** Modified October 31, 2016

- **Requirement number:** NFR10 The program shall take less than two seconds to perform actions after button pressed.

**Rationale:** Similarly, since the program does not take up a lot of memory spaces or cache on the computer, it should not take a long time for responding.

**Priority:** Medium

**History:** Modified October 31, 2016

### 3.3.5 Capacity requirement

**Requirement number:** NFR11

The game shall not have memory explode due to huge player record storage. ~~The game can be played by two players at once or individually. There is no limitations to the number of copies of the game that can be created or distributed.~~

**Rationale:** The game records at most forty player records on a file, and it should not take up much storage on the computer.

**Priority:** Medium

**History:** Modified October 31, 2016

## 3.4 Operational and Environmental Requirements

- **3.4.1 Expected Physical Environment**

**Requirement number:** NFR12

The application should be able to run on computer whenever the computer functions properly.

**Rationale:** The program should be able to run when the computer is running; otherwise the program would be considered failure.

**Priority:** High

**History:** Created October 11, 2016

## 3.5 Expected Technological Environment

- **Requirement number:** NFR13

The application should be able to run on computer with Java version 1.7 or higher installed

**Rationale:** The application is developed under Java version 1.7 environment, so the user should be able to run it under the same environment.

**Priority:** High

**History:** Created October 11, 2016

### 3.5.1 Partner Application

**Requirement number:** NFR14

The application should be able to run on any Java IDE with Java version 1.7 or higher

**Rationale:** The application is developed on a Java IDE with version 1.7 environment, so the user should be able to run it under the same environment.

**Priority:** High

**History:** Created October 11, 2016

## 3.6 Maintainability and Support Requirements

- **3.6.1 Maintenance Requirement**

**Requirement number:** NFR15

The application should be under minimal maintenance.

**Rationale:** Most of bugs occurs during the gaming process should be discovered during or after testing, and they should be corrected and fixed.

**Priority:** High

**History:** Created October 11, 2016

- **3.6.2 Supportability Requirement**

**Requirement number:** NFR16

The application should run on the majority of users' computers.

**Rationale:** Most of the potential players should be able to play the Pong game.

**Priority:** High

**History:** Created October 11, 2016

- **3.6.3 Portability Requirement**

**Requirement number:** NFR17

The application should have their file size less than 50 MB.

**Rationale:** This game should be easy to transfer between users for easy access. Players could use any storage devices to transfer files.

**Priority:** High

**History:** Created October 11, 2016

## **3.7 Security Requirements**

- **3.7.1 Confidentiality**

None applicable to this project.

- **3.7.2 File Integrity Requirement**

**Requirement number:** NFR18

The files for the application should have the extension .JAVA.

**Rationale:** Files need to be recognized as java files before executions.

**Priority:** High

**History:** Created October 11, 2016

- **3.7.3 Audit Requirement**

None applicable to this project.

### 3.8 Cultural Requirements

- **Requirement number:** NFR19

The language that shall be used in this game is English.

**Rationale:** English is the language that all students and faculty members of McMaster University are expected to understand.

**Priority:** High

**History:** Created October 11, 2016

- **Requirement number:** NFR20

The program shall not contain any imagery or language that can be viewed as offensive to users by mocking, insulting, or appropriating their culture, race, or religion.

**Rationale:** User satisfaction will be affected if the material and contents of this game are perceived as offensive to the consumers.

**Priority:** Medium

**History:** Created October 11, 2016

### 3.9 Legal Requirements

- **Requirement number:** NFR21

The program will adhere to the MIT Open License.

**Rationale:** The program will adhere to the MIT License as it is the standard that is set for this project.

**Priority:** Medium

**History:** Created October 11, 2016

## 4 Project Issues

### 4.1 Open Issues

So far there aren't any open issues.

### 4.2 Off-the-Shelf Solutions

The game that this project is attempting to reconstruct would act as an initial model for this project that we have overtaken. There are also other similar games available and the way certain parts are executed can provide a model for us to base our design on. There are also guidance and directions available in java tutorial sites for any programming problems that are likely to occur during the development of this game.

### 4.3 New Problems

#### 4.3.1 Problems in the Current Environment

None applicable in the project.

#### 4.3.2 Existing User

Is there a way to clear user record?

#### 4.3.3 Limitations in Implementation Environment

Is it possible to run on every device that has Java?



## **4.4 Tasks**

Tasks Deadline

Model Development October 14

View Development October 14

Control Development October 17

Project Revision October 31

Project Improvement November 19

## **4.5 Migration to the New Product**

None applicable in the project.

## **4.6 Risks**

There are no risks associated with this project.

## **4.7 User Documentation and Training**

The user documents will be created as per the SE 3XA3 guidelines. There will be an in-game tutorial that will display the instructions on how to play the game. Other than that, no further training should be required..

## **4.8 Waiting Room**

There are many factors in the game that has to be improved upon, as has been discussed throughout various parts in the document. The improvement of functionality of the game is to be looked forward towards. The addition of sound to the game can also be added to this.

## **4.9 Ideas for Solutions**

Proper documentations for the game must be made. The coding for the game should be well commented and easy to follow. Other than this, for the current period there isn't any other ideas for solutions. This section would be updated at a later period.

## **5 Appendix**

### **5.1 Symbolic Parameters**

Section 1.4 provides a list of terms and its definition of any symbols that are used throughout the document.