Software Requirements Specification Document

CENG 490 VANA Project

Barış Çavuş - 1819754

Erenay Dayanık - 1819192

Memduh Çağrı Demir - 1819218

Mesut Balcı – 1819093

Date: 30.11.2014

Table of Contents

1 Introduction

This document is a software requirement specification for the Light My Way Game Project which is an android application. After giving information about the definition of the project at the beginning part of the document, we will give complete description for overview and list the requirements which meet the needs of the users.

1.1 Problem definition

Smartphone markets such as Play Store or App Store are accessible from the all over the world, this project affects all people around the world. Although we do not have accurate statistical data, according to researches 56% of the people in the world use smartphones and they spend 80% of their time on mobile platforms with apps or games(Smith,2013).Thus, our user domain is quite large including both adults and kids. The range of the age of users may differ widely.

There are a lot of companies all over the world making mobile games. “In 2013, app developers brought in $26 billion in revenue from app stores.In 2017, that number is expected to hit $77 billion.”(Russell,2013).9 big game companies including Gameloft and Rovio made over $100 million from sales in 2013. As a result, there are variety of games in market but there is not much co-operative multiplayer games to entertain people. Furthermore it doesn't mean that an existing game on a category will fulfill the market gap since game choices depend on people's admiration.

1.2 Purpose of document

This document aims to give a brief description about the Light My Way Game Project. With the help of this document the needs of the company and the solution that will be provided to that needs will be clearly presented. In other words purpose is to outline the functional requirements of Light My Way Game. This document is intended for:

- Instructors
- Devolopers
- Testers

1.3 Scope of project

Light My Way is a game application on android phones.The aim of the project is to develop a game which entertain people by getting two people together and make them pass levels with cooperation of each other. Levels will be 2-D dark environments decorated as inside of an Egyptian pyramid. There will be some blocks with mounted light sources on them and mirrors next to some of these blocks. Players are expected to escape these levels by interacting with the objects near them and lightening each others path. They will be able to walk, move blocks in levels and rotate mirrors in the room. In order to achieve collaboration between players, they will also be able to communicate.

1.4 Definitions,acronym ,abbreviations

Term	Definition
User / Player	A person who plays the game.
Character	The human figure controlled by the player.
Unity3D	a rendering engine fully integrated with a complete set of intuitive tools and rapid workflows to create interactive 3D and 2D content.
Multiplayer	More than one players playing in the same game with separate devices.
Co-op / Co-operative	A multiplayer game type that allows players to play as a team against to computer.
Android	Android is a mobile operating system (OS) based on the Linux kernel and currently developed by Google.
Ingame Screen	After invoking play in game menu , this is screen is where users play with in a drawn interface.

Emote	To express emotion, especially in an excessive or theatrical manner
Server	Server is a running instance of an application (software) capable of accepting requests from the client and giving responses accordingly
Google Play Store	Web store for android based applications
Application Programming Interface(API)	A set of routines, protocols, and tools for building software applications
Software Development Kit(SDK)	A programming package that enables a programmer to develop applications for a specific platform

1.5 References

IEEE STD 830-1998, IEEE Recommended Practice for Software Requirements Specifications

Software Project Survival Guide. (2001). *Quill and Quire*, (4)

Smith, A. (2013, June 5). Smartphone Ownership 2013. Retrieved October 16, 2014.

Russell, K. (2013, December 11). These 9 Mobile Game Companies Got Over $100 Million Sales In 2013. Retrieved October 16, 2014.

1.6 Overview

This software requirement document is prepared according to IEEE Std. 830-1998 and it is divided into sections and practices. User classes are used, while organizing functional requirements, which is recommended in template of SRS section in standard document.

This document starts with explanation of the project. It continues with the overview of the project and describes the general factors that can affect the system requirements. On the requirements specification chapter, system requirements will be explained in detail with the technical terms. It is explained for the developers mainly. On this process, use case diagrams,mock interfaces and er diagrams will be used.

2 Overall Description

This sections main focus is to show the main factors which affects the product and its requirements in detail. This section also contains information about features and why it is selected. In some sections there are some attributes which enable the product to be able to evolve in the future.

2.1 Product Perspective

Light My Way is a standalone product and it is designed to run in android based smart phones. The sole requirement for the user is downloading the game from google play market on available devices.

2.1.1 System Interfaces

Light My Way is a stand-alone application, therefore System Interfaces will not be needed through this project.

2.1.2 User Interfaces

Interfaces described below are will be visible when the user started up the game from their phones.

2.1.2.1 Play (in game menu)

Used in game menu, this icon will lead player to the ingame interface where two user play.

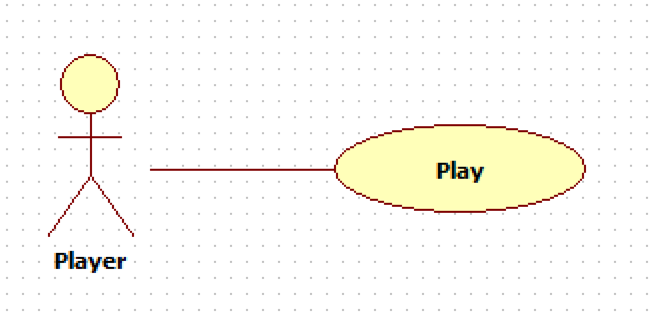


Figure 1 – Play Use Case

Description of Play Use Case:

- User touches play icon
- Waiting for matchmaking
- server matches two player
- game screen come
- game is started

2.1.2.2 MovePlayer

Used in ingame screen, this movement triggered by touch movements.

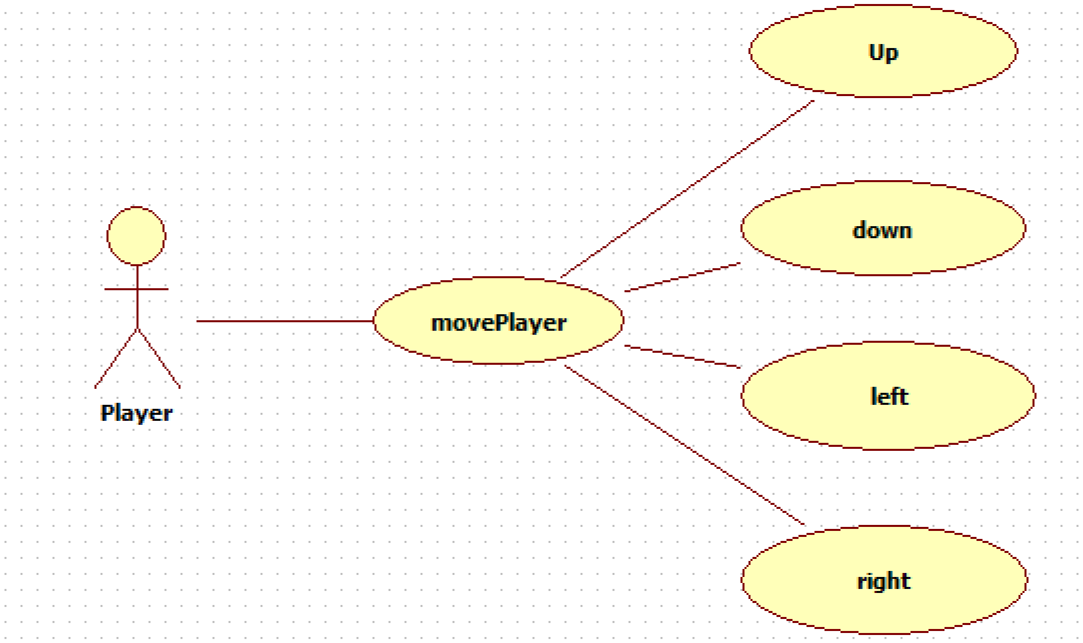


Figure 2 – movePlayer Use Case

Description of movePlayer Use Case:

- User touches screen place where he/she wants to move
- Character moves using Up, Down, Left or Right movements
- Character stops in closest available destination to the point touched
- World state changes propagated in two user playing together

2.1.2.3 InteractObject

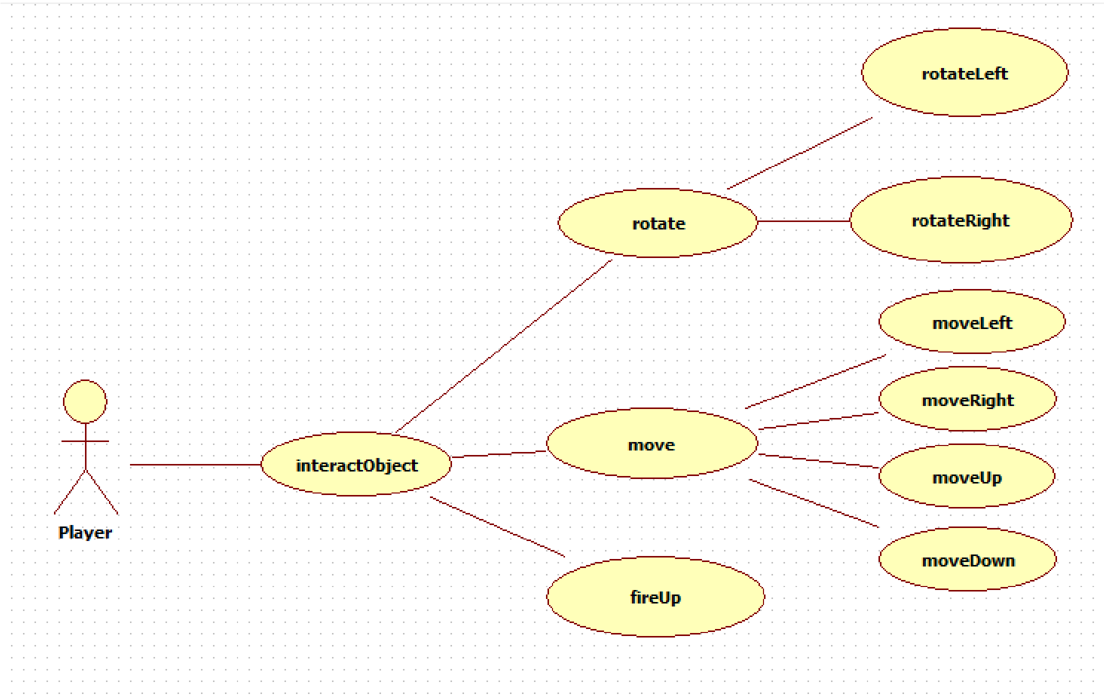


Figure 3 – interactObject Use Case

Description of InteractObject Use Case:

- The user clicks the object ingame screen.
- The user decides action type.
- The user choose left or right rotation if action is rotation
- The user choose one of the left, right, down or up movement if action is move.
- Fire is lightened if the selected action is fireUp

2.1.2.4 howToPlay

Used in Game menu, which is used to get help about game dynamics and purpose.

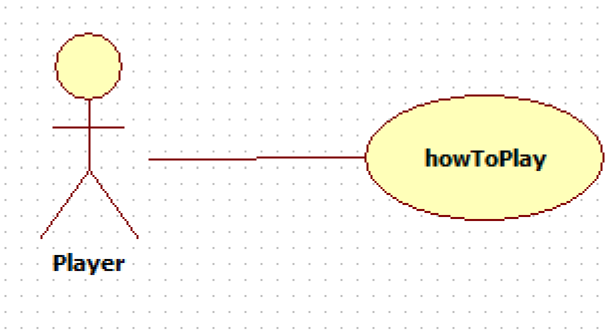


Figure 4 – howToPlay Use Case

Description of howToPlay Use Case:

- The user clicks the How to Play icon
- System explains game rules and how to play.

2.1.2.5 Exit

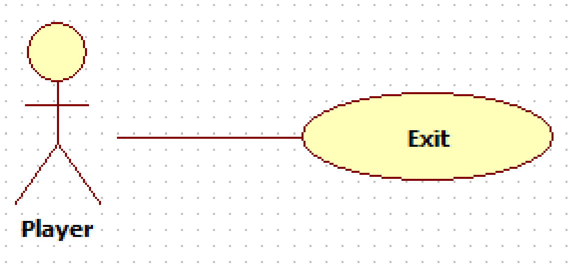


Figure 5 – Exit Use Case

Used in Game menu, which is used to close application.

Description of Exit Use Case:

- The user clicks the Exit icon.
- The application is closed by the system.

2.1.2.6 Options

Used in Game menu which is used to adjust volume and username.

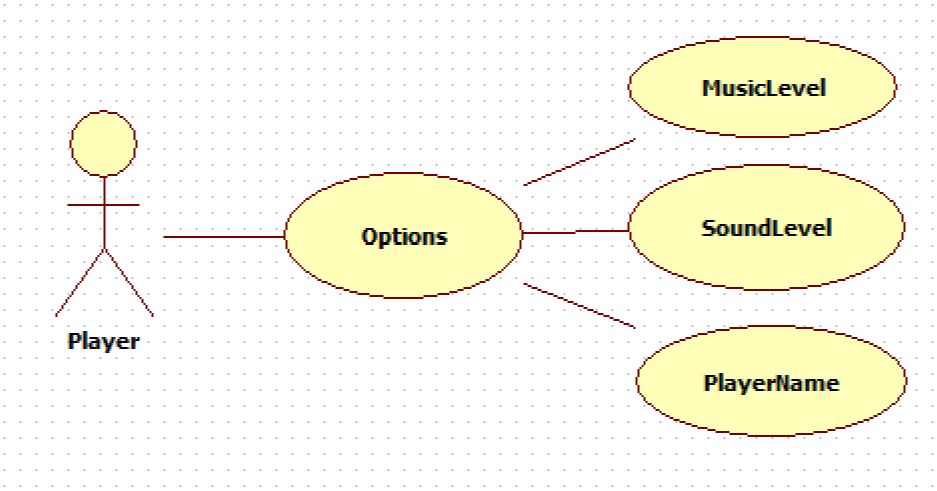


Figure 6 – Options Use Case

Description of Options Use Case:

- The user clicks the Options icon.
- User determines the level of music and sound.
- User defines his/her username.

2.1.2.7 emote

Used in Game menu which is used to communicate with other player during the game.

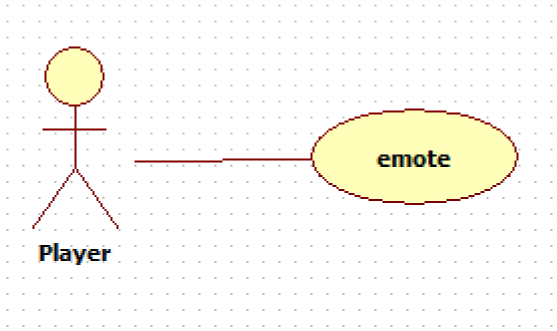


Figure 7 – emote Use Case

Description of emote Use Case:

- The user clicks the emote icon.
- The selected signal is sent to other player.

2.1.2.8 Pause

Used in Game menu which is used to pause the game.

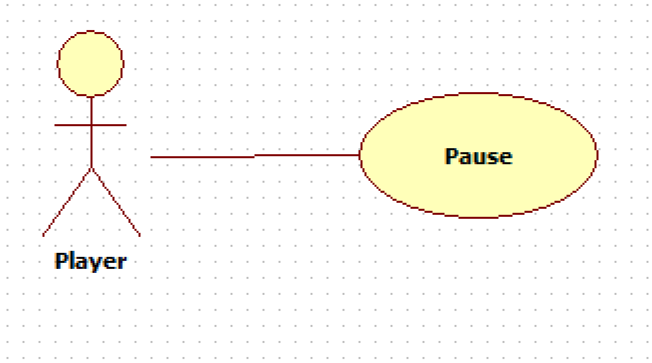


Figure 8 – Pause Use Case

Description of Pause Use Case:

- The user clicks the pause icon.
- The game is paused until user clicks it again.

2.1.3 Hardware Interfaces

Light My Way game will need the standard Android provided controls and device hardware buttons for a reasonable game play.

[R.2.1] Light My Way should respond to the input of the phone’s hardware home button.

[R.2.2] Light My Way should respond to the input of the phone’s hardware back button.

Android devices should exist to play the game. Internet connection is also required to connect two player. Therefore android devices must have wi-fi property or cellular data.

2.1.4 Software Interfaces

Since this game is developed as a mobile game, a mobile operating system is the most important required software for this game to work. Asset managements including file operations such as storing and retrieving assets are done by using operating system calls.

Unity3D is used for creating user interface and rendering in-game animations.

[R.2.3] Light my Way will run on the Android version 4 and above.

[R.2.4] Light my Way will use the Unity3D game engine for handling physics computations throughout the game.

[R.2.5] Light my Way will use the Unity3D game engine for handling object collisions and interactions.

[R.2.6] Light my Way will use the Unity3D game engine for handling animations of character and objects.

[R.2.7] Light my Way will translate into the sleep after receiving corresponding signal from Android OS.

[R.2.8] Light my Way will translate out of the sleep after receiving corresponding signal from Android OS.

2.1.5 Communication Interfaces

The game will be capable of exchanging player and object's information with each other using network protocols.

[R.2.9] Light My Way will use the available wifi to communicate with the game server.

[R.2.10] Light My Way will use the available cellular internet connection to communicate with the game server.

2.2 Product Functions

There are four main functions that need to be satisfied by the game:

2.2.1 Basic Movement of Player

Both players playing in separate devices will be able to move their characters in four primary directions.

2.2.2 Basic Interaction of Character with Objects

Unlike concrete objects, there will be interactable objects in the game. Characters will be able to push and rotate these objects in the game.

2.2.3 Light Reflections and Movement Decisions

Since characters are not allowed to move out of lighted paths, movements of characters will be checked at every move. If movement is against the rules, then movement will be interrupted and player is notified by an animation. Lighted paths is not static and can be extended by using light reflections.

2.2.4 Exchanging Information

As a multiplayer game, synchronisation of game map will be done by exchange of information including coordinates and angles of objects and characters coordinates.

2.3 User Characteristics

There will be no required specific characteristic to play Light My Way game. The only expected skill is to know basics of smartphone usage. Any other necessary usage of the game will be provided in help menu.

2.4 Assumptions and Dependencies

We will use Unity3D for graphics , after the testing phase, we will decide the minimum requirements and oldest android version to be supported then release on market. The game is dependent on the availability of Internet Connection. Any connection loss from a client will be resulted in loss in game and other user will be warned.

3 Specific Requirements

3.1 External Interface

Since these requirements are stated in the Section 2 of this document in detail, these requirements will not be covered again.

In this section the details of requirements which are already mentioned in features will be provided.

If the explanation in section 2 is already sufficient it will not be mentioned again.

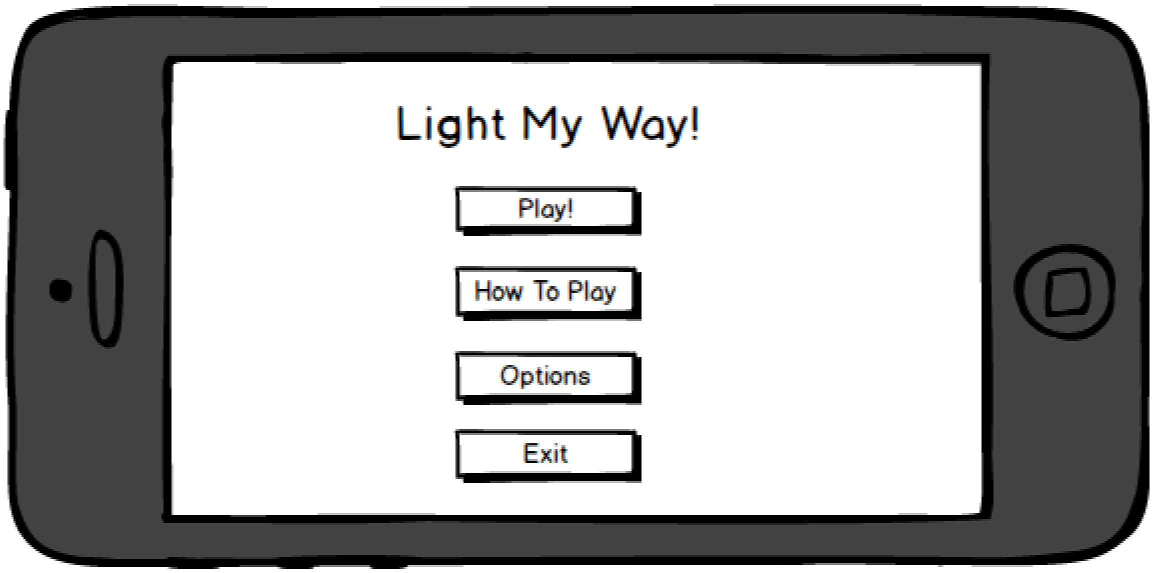


Figure 9 – Light My Way Main Screen

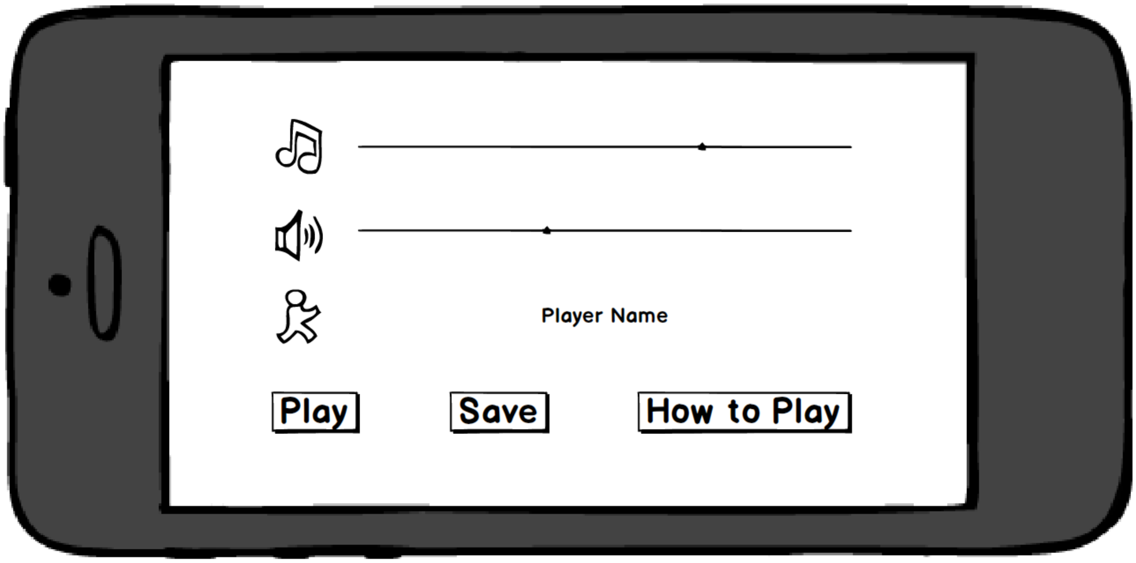
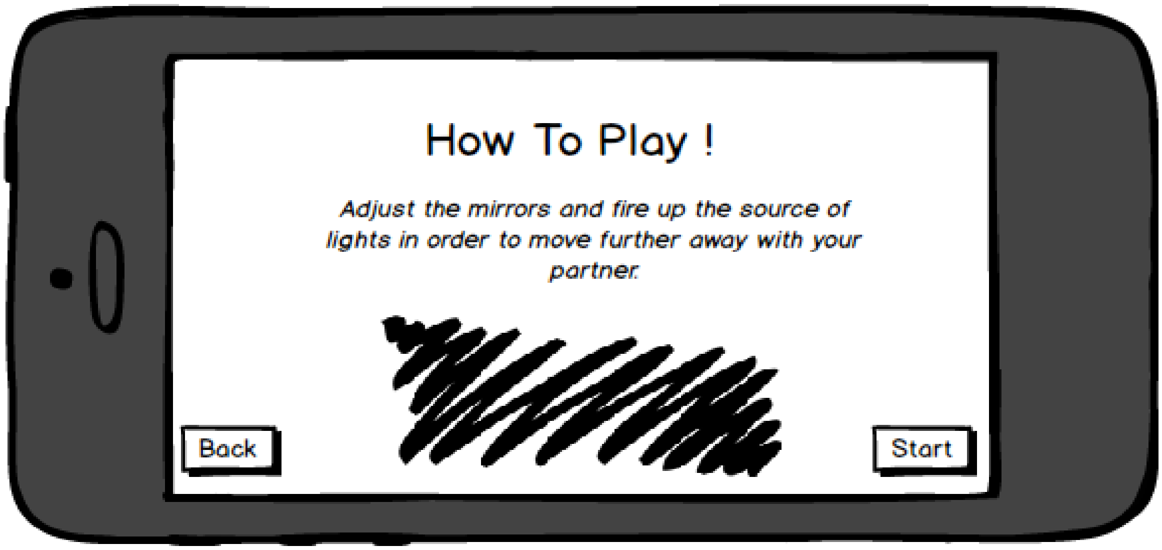


Figure 10 – Light My Way Options Screen



How To Play !

Adjust the mirrors and fire up the source of lights in order to move further away with your partner.

Figure 11 – Light My Way How To Play Screen

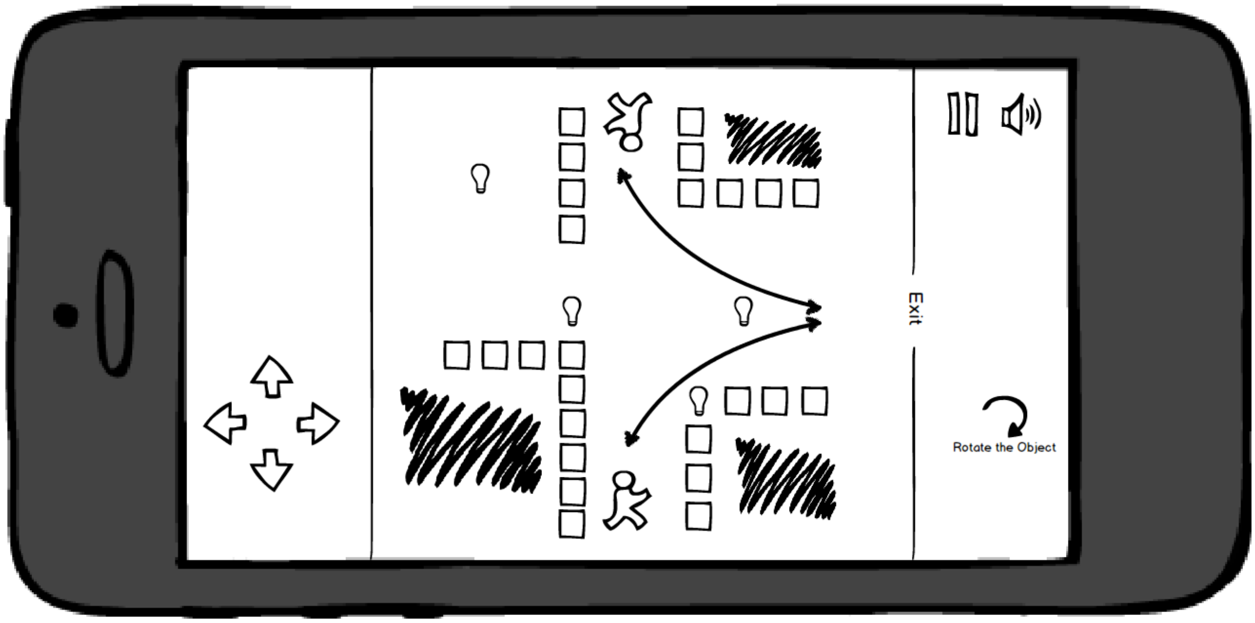


Figure 12 – Light My Way Game Play Screen

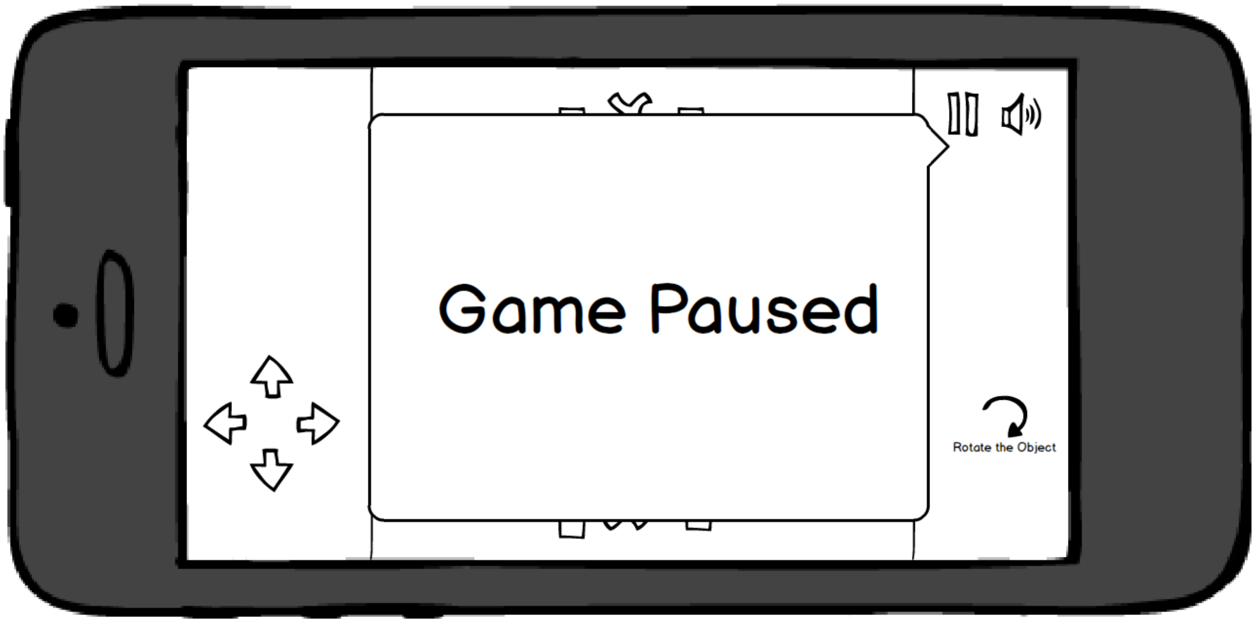


Figure 13 – Light My Way Game Paused Screen

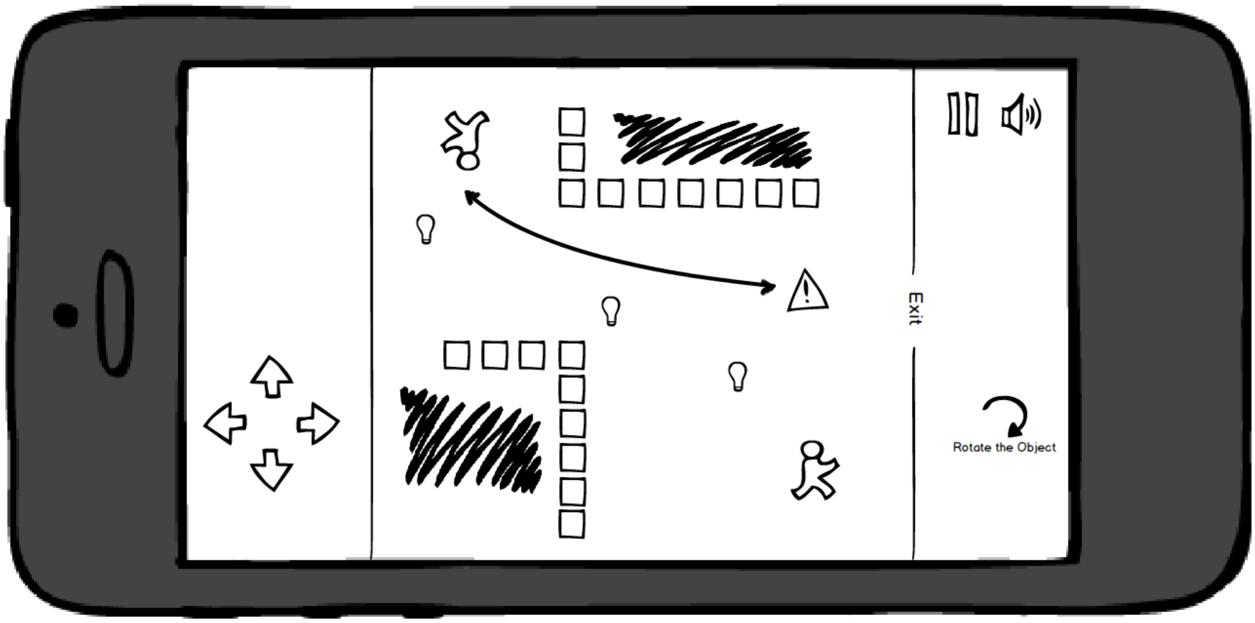


Figure 14 – Light My Way emote usage

3.2 Functional Requirements

3.2.1 Play

Play is the main function which will match two players. This function will lead users to ingame screen.

Use Case Name	Play
Xref	Section 2.1.2 Play
Precondition	User opens the game from their Android device.
Basic Path	1. User touches the play button. 2. User waits for matchmaking. 3. User matches with another user according to their ping.

Post condition	Users start playing together on ingame screen.
Exception Paths	Users may leave matchmaking anytime. In such cases users drop from matchmaking queue.
Other	If user leaves the game after matching with another user. Game will not start.

Play function Requirements :

[R.3.1] When the user selects the play icon the menu screen shall transition to

the play game screen.

[R.3.2] Server shall match the player with another player.

[R.3.3] When the matched user found , ingame screen shall prompt the matched user’s name.

[R.3.4] When a matched user found prompt disappears, game shall start in ingame screen.

[R.3.5] Ingame screen shall be in two dimensional environment Which all objects shall rest in ground.

[R.3.6] Ingame screen shall allow characters to move in Up, Down, Right and Left directions.

[R.3.7] Ingame screen shall allow to drag object chosen.

[R.3.8] Ingame screen shall allow to rotate the mirror chosen.

[R.3.9] Only enlightened places shall be visible in ingame screen.

3.2.2 MovePlayer

This section describes characters movement functions’ data flows and descriptions.

3.2.2.1 Up

Use Case Name	movePlayerUp
Xref	Section 2.1.2 MovePlayer
Precondition	User opens the game from their Android device. User joins the game using Play function.
Basic Path	1.user presses the top of their screen or top button.
Post condition	His/her ingame character moves up.
Exception Paths	If user presses more than one time or holds his/her fingers on top of their screen, character keeps moving up.
Other	

3.2.2.2 Down

Use Case Name	movePlayerDown
Xref	Section 2.1.2 MovePlayer
Precondition	User opens the game from their Android device.

	User joins the game using Play function.
Basic Path	1.user presses the down button.
Post condition	His/her ingame character moves down.
Exception Paths	If user presses more than one time or holds his/her fingers on bottom on their screen, character keeps moving down.
Other	

3.2.2.3 Left

Use Case Name	movePlayerLeft
Xref	Section 2.1.2 MovePlayer
Precondition	User opens the game from their Android device. User joins the game using Play function.
Basic Path	1.user presses the left button.
Post condition	His/her ingame character moves left.
Exception Paths	If user presses more than one time or holds his/her fingers on left of their screen, character keeps moving left.
Other	

3.2.2.4 Right

Use Case Name	movePlayerRight
Xref	Section 2.1.2 MovePlayer
Precondition	User opens the game from their Android device. User joins the game using Play function.
Basic Path	1.user presses the right button of their screen.
Post condition	His/her ingame character moves right.
Exception Paths	If user presses more than one time or holds his/her fingers on right of their screen, character keeps moving right.
Other	

Movement functions Requirements:

[R.3.10] When users touches up button character shall move up.

[R.3.11] When users touches down button character shall move down.

[R.3.12]When users touches left button character shall move left.

[R.3.13]When users touches right button character shall move right.

[R.3.14]If there is no light in the destination, movement shall be interrupted.

[R.3.15]If movement path is blocked or unavailable , movement shall be interrupted.

3.2.3 InteractObject

This section is the description and data flow of functions used to interact with ingame objects such as torch , mirrors or blocks.

3.2.3.1 Rotate

Use Case Name	InteractObjectRotate
Xref	Section 2.1.2 InteractObject
Precondition	User opens the game from their Android device. User joins the game using Play function.
Basic Path	1. User selects the object that he/she wants to interact with. 2. User selects the rotate functions from interactions menu. 3. Users selects rotateLeft or rotateRight. 4.rotateLeft function rotates the object 90^o counter clockwise. 5.rotateRight function rotates the object 90^o clockwise.
Post condition	Chosen object changes its rotation.
Exception Paths	-
Other	-

3.2.3.2 InteractObjectMove

Use Case Name	InteractObjectMove
Xref	Section 2.1.2 InteractObject
Precondition	User opens the game from their Android device. User joins the game using Play function.
Basic Path	1. User selects the object that he/she wants to interact with. 2. User selects the move functions from interactions menu. 3. Users selects moveUp,moveDown,moveLeft,moveRight. 4.moveUp function moves object to up. 5.moveDown function moves object to down. 6.moveLeft function moves object to left. 7.moveRight function moves object to right.
Post condition	Chosen object moves according to selected function.
Exception Paths	-
Other	-

3.2.3.3 FireUp

Use Case Name	fireUp
Xref	Section 2.1.2 InteractObject
Precondition	User opens the game from their Android device. User joins the game using Play function. User selects torch ingame item.
Basic Path	1.User can fire torch up with touching the torch.
Post condition	Torch is a light source now.
Exception Paths	
Other	

Object interations functional requirements:

[R.3.16] Objects shall be chosen by touching one time on them in Ingame Screen.

[R.3.17] When object is selected it shall be brightened and will be distinct from other objects.

[R.3.18] Each object has its own allowed operations.

[R.3.19] When touched on a chosen mirror, mirror shall rotate once 90 degree on chosen direction.

[R.3.20] Mirror shall change the direction of its light reflection when rotated.

[R.3.21] User can only fire the torch if character is near the torch.

[R.3.22] Torch can be fired up by touching one it.

[R.3.23] Object can be dropped by leaving it to its initial place

[R.3.24] Object can be dropped to destination by touching to destination point after selecting the object.

3.2.4 howtoPlay

Use Case Name	howtoPlay
Xref	Section 2.1.2 howtoPlay
Precondition	User opens the game from their Android device.
Basic Path	1.User touches the howtoPlay button. 2.User enters the training session.
Post condition	User learns the basic control of the game.
Exception Paths	-
Other	-

[R.3.25] When user touches how to play button on menu screen, how to play screen shall be displayed.

[R.3.26] By sliding the windows users shall be able to see screenshots of tutorial screens.

3.2.5 Exit

Use Case Name	Exit
Xref	Section 2.1.2 Exit
Precondition	User opens the game from their Android device.
Basic Path	1.User touches the exit button. or 2.User touches the return button of their device, when he/she is on the main menu.
Post condition	Game closes.
Exception Paths	User can close game with forcing game to exit. He/she can close it with killing the application process.
Other	

[R.3.27] When user touches the exit icon, game shall be closed.

[R.3.28] Phones back buttons shall also perform the closing operation.

3.2.6 Options

Use Case Name	Options
Xref	Section 2.1.2 Options
Precondition	User opens the game from their Android device.
Basic Path	1.User touches the options button. 2.User adjust ingame music level. 3.User adjust ingame sound level. 4.User decide his/her username.
Post condition	Music or sound level is adjusted according to user choice. Username can be seen different to another users.
Exception Paths	
Other	

[R.3.29] When option button is chosen from menu screen, Options Screen will be displayed.

[R.3.30] Game music shall be adjustable by touching horizontal slider in Options Screen.

[R.3.31] Game sounds shall be adjustable by touching horizontal slider in Options Screen.

[R.3.32] User can change their nicknames from textbox in Options Screen.

3.2.7 Emote

Use Case Name	Emote
Xref	Section 2.1.2 Emote
Precondition	User opens the game from their Android device. User joins the game using Play function.
Basic Path	1.User touches the emote button. 2.User can send signals or smileys to other users.
Post condition	Users can communicate with this smileys.
Exception Paths	
Other	

[R.3.33] Users can choose emote button in ingame screen by touching on them.

[R.3.34] When emote button selected , emotes will be displayed.

[R.3.35] When user chooses one of the displayed emotes, chosen emote shall be displayed to other user.

3.2.8 Pause

Use Case Name	Pause
Xref	Section 2.1.2 Pause
Precondition	User opens the game from their Android device. User joins the game using Play function.
Basic Path	1.User touches the pause button. 2.Game will be paused.
Postcondition	Game will stay paused until the user unpauses it.
Exception Paths	
Other	

[R.3.36] When user touches the pause button, game will be paused in Ingame Screen.

[R.3.37] Pause screen shall be displayed immediately when game is paused.

[R.3.38] Game shall be continued when any player touches the resume button in Pause Screen

3.3 Performance requirements

The game should support 100 simulations players playing with two-people pairs. Response time for any request should not be more than 5 seconds.

Delay between pairs should not exceed 4 seconds.

3.4 Database Requirements

High level logical structure of database is illustrated as an ER Diagram in the figure below.

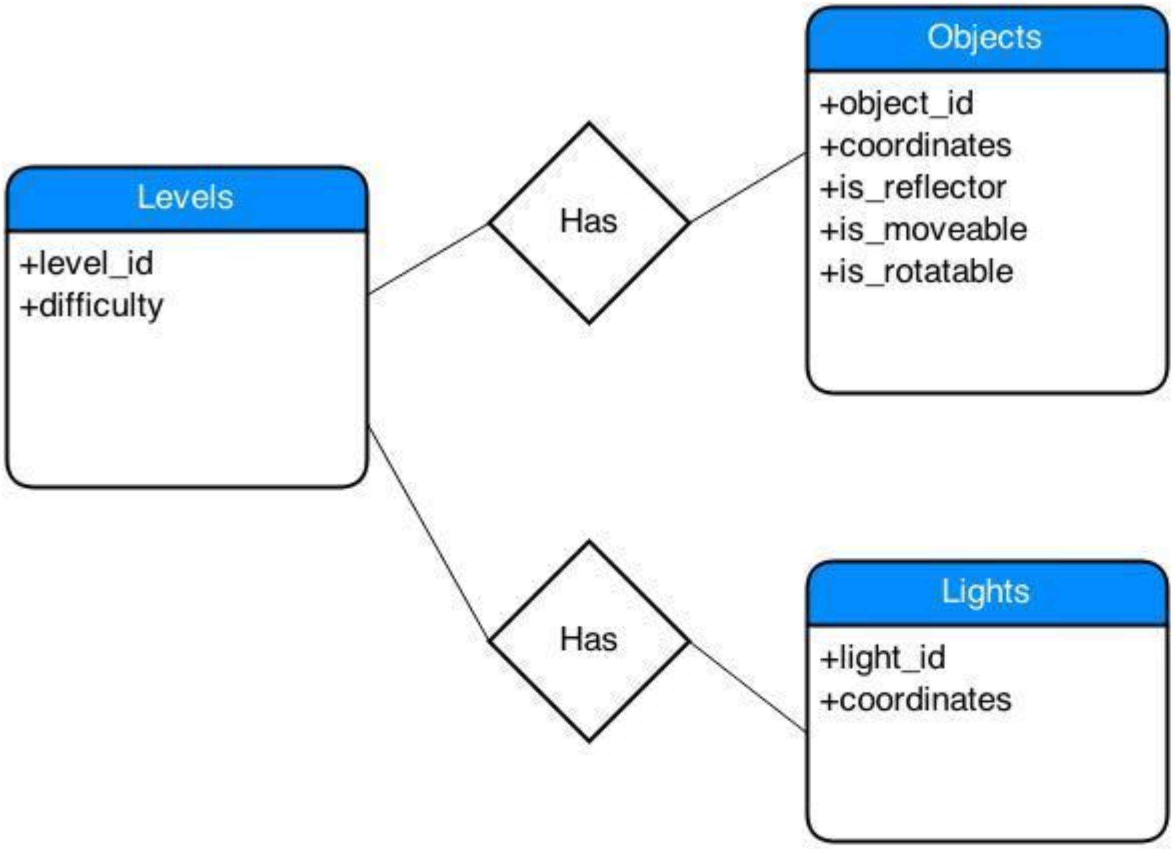


Figure 15 – Light My Way ER Diagram

Levels relation contains levels added to the game with calculated difficulties of that level. It has two relations with the Objects and Lights.

Objects relation contains all the information about an object in the designed level. Objects can be reflective, moveable and rotatable as well as being static and un-interactable. All objects must have coordinate values in order to place them in the corresponding level.

Lights relation contains placed light entities in the corresponding level. As in objects, coordinates of lights must be defined.

3.5 Design Constraints

The only standard we complied with at this stage is the recommended practice for software requirements specifications (IEEE std 830-1998).

3.6 Software System Attributes

3.6.1 Reliability

[R.3.39] The Mean-time-to-repair shall be at most 1 hour

[R.3.40] The Mean-time-to-failure shall be at least 24 hours.

[R.3.41] In case of any system failure, System shall display an error message in the main screen to inform user about possible problem.

3.6.2 Availability

[R.3.42] System should be available 24 hours per day, 7 days per week.

[R.3.43] In case of an unexpected failure in the system such as server connection problem, system inform users and does not allow them to continue.

3.6.3 Security

[R.3.44] Since Light My Way does not create profile for users, it does not need to access user credentials. Therefore, security is not a concern in this application.

3.6.4 Maintainability

[R.3.45] Documentation should be supplied for all modules of the system

[R.3.46] Requirement and change management should be used in development phase.

3.6.5 Portability

[R.3.47] Light My Way should run on Android OS. It is not a cross product application.

4 Planning

4.1 Team Structure

Ali Fatih Gündüz - Advisor

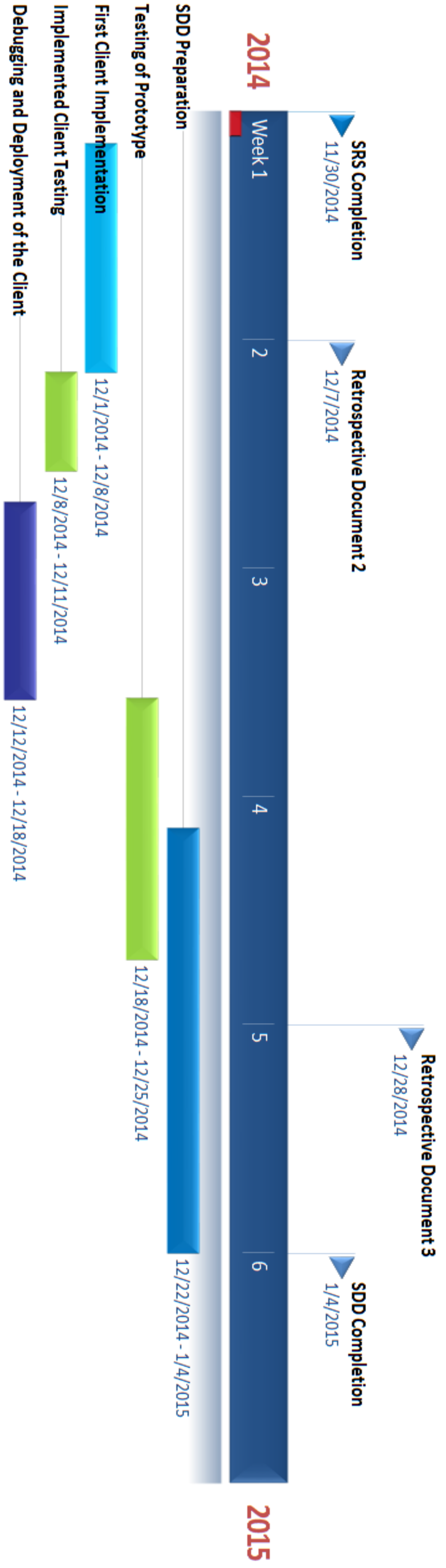
Barış Çavuş - Researcher, Developer

Erenay Dayanık - Researcher, Developer, Scrum Master

Memduh Çağrı Demir - Researcher, Developer

Mesut Balcı - Researcher, Developer

4.2 Estimated Schedule



4.3 Process Model

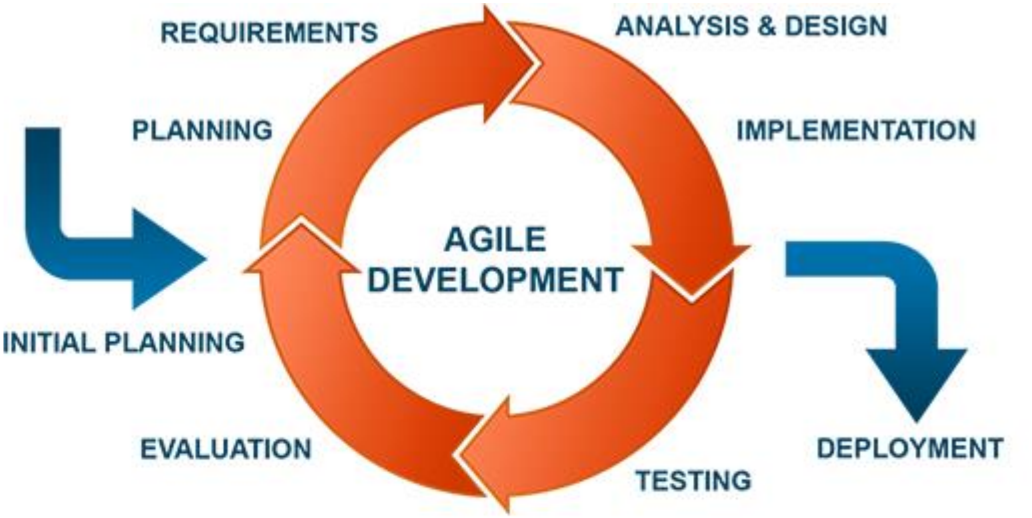


Figure 16 – Agile Method Development Process Model

5 Conclusion

In this document, the project we develop which implements a co-operative multiplayer game is explained in detail by using use-case diagrams, er diagram and figures. Firstly, the Light My Way project that we try to develop is explained briefly. Then the gameplay is explained with use cases. All the possible actions in the game is explained in detail. Secondly, non-functional requirements are explained in separate section including performance requirements and database requirements. Database schema and its associations are shown with an er diagram. Finally, we included our team structure, our planned schedule and process model for this project.