Software Design Descriptions

Scavenger Hunt Game Using BLE Beacon

Ümmügülsüm Kaşıkçı, Aslıhan Asena Şahin, and Nuri Akseli

(c1311028, c1311054, c1611653)@student.cankaya.edu.tr

Department of Computer Engineering,

Çankaya University, Turkey

December 14, 2018

Table of Contents

1 OVERVIEW

1.1 Scope

Scavenger Hunt Game Using BLE Beacon is an Android application project. It is a game which is based on the locations and missions. In this game, there are game creators and players. To begin playing, game creators should create a game in a specific location and add missions for players. Players can earn points by completing the given missions and find a place for themselves in the scoreboard of the game. This application is in Android platform that means the users need a mobile device with Android operating system version 7.0 or above.

1.2 Purpose

The purpose of this Software Design Descriptions (SDD) document is describing the details of the project called "Scavenger Hunt Game Using BLE Beacon". We prepared this document according to "IEEE Standard for Information Technology - Systems Design - Software Design Descriptions - IEEE Std 1016 - 2009". In this document, we indicated how our application's software should be developed. We represented the details of design of the application by using graphical notations such as class diagrams, use case diagrams, deployment diagrams, and other supporting design information.

1.3 Definitions

Term	**Definition**
IEEE	Institute of Electrical and Electronics Engineers
Software Design Description (SDD)	The complete description of the design of the system.
UML Diagram	A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system.
Stakeholder	A person, group or organization that has interest or concern in an organization.

1.4 References

IEEE. IEEE Std. 1016-2009 IEEE Standard for Information Technology - Systems Design - Software Design Descriptions. IEEE Computer Society, 2009.

2 CONCEPTUAL MODEL FOR SOFTWARE DESIGN DESCRIPTIONS

In this part, conceptual model for the SDD is introduced. This conceptual model mainly explains the context and stakeholders in which SDD is prepared.

2.1 Software Design in Context

In Scavenger Hunt Game Application, Incremental Software Development Methodology is used as a development method. The idea of this methodology is to divide the project into modules. After that, project is developed module by module. Hence, potential defects are spotted early, and changes to project scope are less costly and easier to implement. At the end of the project, the application has fewer bugs, and it works correctly.

2.2 Software Design Descriptions within the Life Cycle

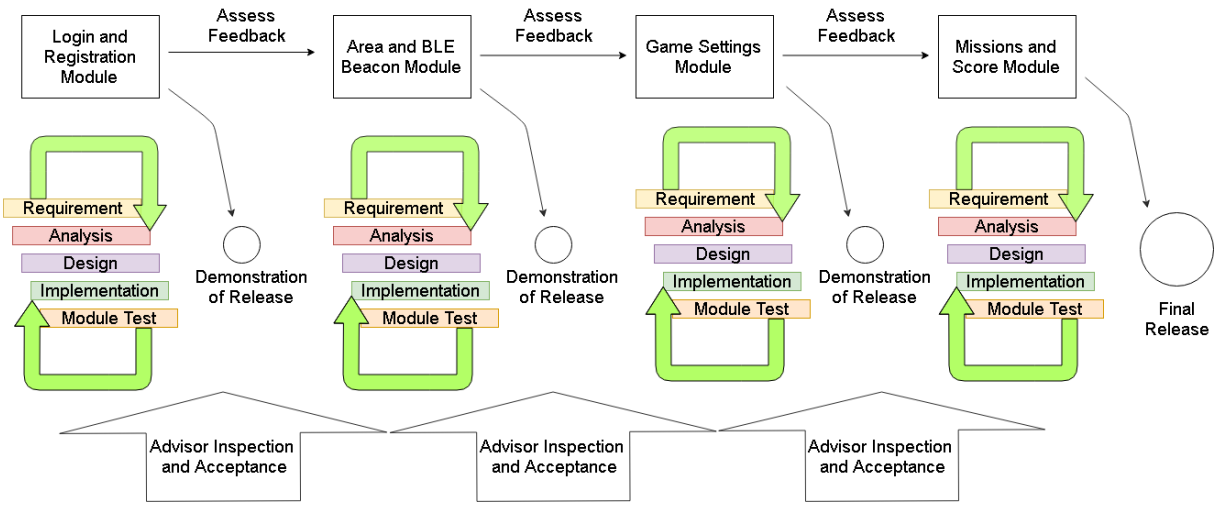

Figure 1 The development life cycle of the project

2.2.1 Influences on SDD Preparation

The critical software life cycle product that drives this software design is the software requirement specifications (SRS) of the project. All the details and requirements are taken from the SRS document to prepare this document.

2.2.2 Influences on Software Life Cycle Products

This SDD document influences the content of SRS of this project. It also has influences on the whole implementation phase of the Scavenger Hunt Game Application. Also, the test plans and documentation of the system are also be influenced by the SDD.

2.2.3 Design Verification and Design Role in Validation

Test cases are prepared after the SDD document phase. With these test cases, the software is tested, and all modules are evaluated. After the results, the success rate of this software is observed and documented.

3 DESIGN DESCRIPTION INFORMATION CONTENT

3.1 Introduction

In this part, SDD of this project that gives information about design and implementation are presented. In this part, also, the topics explained includes SDD identification, design views, design elements, design overlays, design rationale, and design languages.

3.2 SDD Identification

This SDD report is prepared concerning the IEEE 1016 - 2009 standards, and this is the first version of SDD for this project. UML notation is selected for diagrams and Draw.io website is used for drawing these diagrams.

This SDD contains development life cycle, class diagram, use case diagrams, deployment diagram, class diagram, ER diagram and flowchart diagram of this project.

3.3 Design Stakeholders and Their Concerns

Design stakeholders are the developer team and their advisor in the Scavenger Hunt Game project. Our developer team members are computer engineer candidates, and they know and understand software development. The concerns of this project's stakeholders are shown in below:

- The interface should be easy to use.
- The application must open in maximum three seconds.
- The application should work with every current Android device.
- The application should not need high-end hardware requirements.
- The application should be safe and secure.

3.4 Design Views

To represent the diagrams, UML is used in this project. This SDD document contains the design views which are governed by design viewpoitns that are explained in part 4.

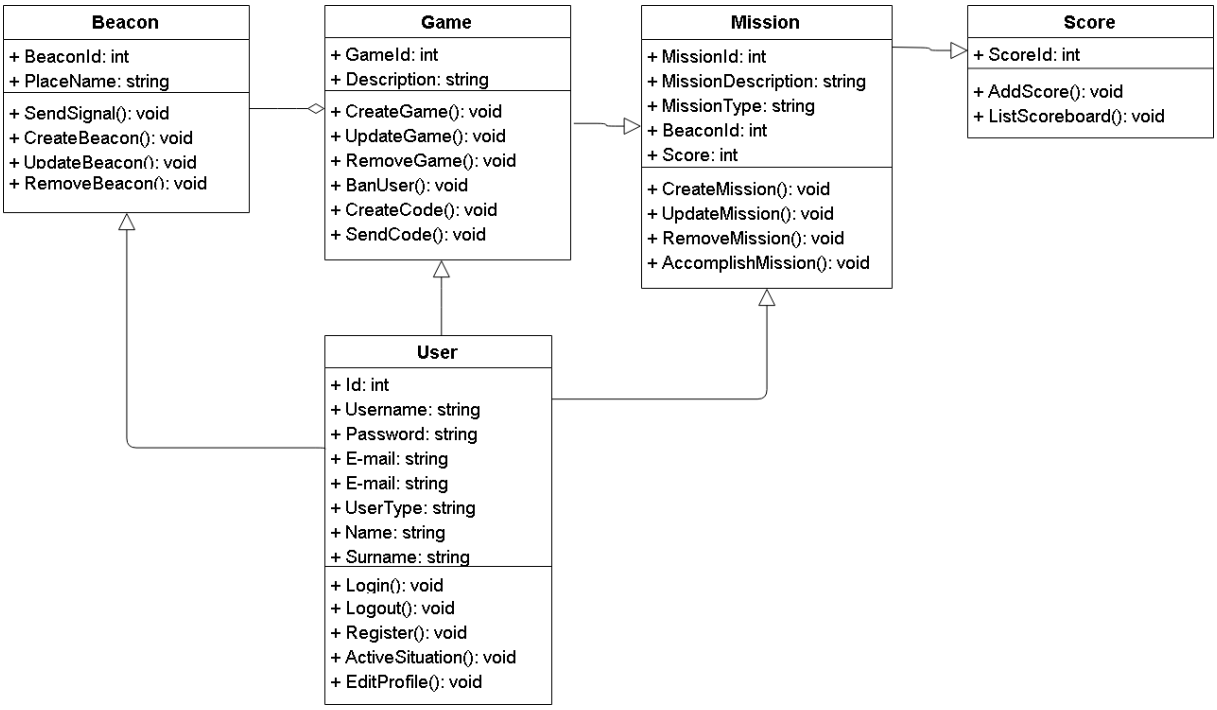

Figure 2 The class diagram of the project

As an example, Figure 2 shows the UML class diagram of Scavenger Hunt Game project. It shows the connection between all classes.

3.5 Design Viewpoints

This part is used to give a short explanation on the main design viewpoints which are used in part 4. It is defined in the IEEE 1016 - 2009 standards.

3.5.1 Context Viewpoint

The context viewpoint is about the relationships and interactions between the system and its environment.

3.5.2 Composition Viewpoint

The composition viewpoint shows the main work components of the project.

3.5.3 Logical Viewpoint

The logical viewpoint identifies all of the classes and the relations between those classes.

3.5.4 Information Viewpoint

The information viewpoint describes the storing, managing and distributing of information.

3.5.5 Interface Viewpoint

The interface viewpoint gives the necessary information about how the design project looks like and be used by anyone who is interested.

3.5.6 State Dynamics Viewpoint

The state dynamics viewpoint shows the behaviour of the system when there are some specific events.

3.6 Design Elements

This part is about main design elements like entities, design relationship, constraints.

3.6.1 Design Entities

3.6.1.1 Database Management System

In the Scavenger Hunt Game, Google's Firebase system is used for database management. It has many advantages for mobile devices and easy to integrate into the project. It is used to store users' information, beacons' information and games' information.

3.6.1.2 User System

Users can connect to Scavenger Hunt Game with their mobile devices. Users must use mobile devices which has Android Operating System on it, and its Android version must be 7.0 or above.

3.6.1.3 Location System

When system determines users' location, it uses BLE Beacon devices. So, Bluetooth modules of users' mobile devices must work properly.

3.6.1.4 Programming Language

The application is an Android application. Therefore, we use the Android Software Development Kit (SDK) in the software development process.

3.6.2 Design Relationships

This project's main parts are Google's Firebase as database, Android programming, BLE Beacon and Android mobile devices.

3.6.3 Design Constraints

- Google's Firebase System should be used for database management system.
- Software must be programmed with according to Android programming.
- The rules of this game should be similar to board game of Scavenger Hunt Game.
- BLE Beacon devices should be used for determining location.

3.7 Design Overlays

Scavenger Hunt Game's main factors are simplicity and optimized design. So, it does not require a powerful hardware, and everyone can play this game easily.

3.8 Design Rationale

In this project, design choices are made according to simplicity and performance concerns, but stakeholders may have request further requirements. Therefore, the system must consist of modular parts and developers of the system have to use comments in their code. In this way, other developers can understand the existing code and the system.

3.9 Design Languages

In this project, UML is chosen as a part of design viewpoint. It will be used for explaining design viewpoints.

4 DESIGN VIEWPOINTS

4.1 Introduction

In this part, Scavenger Hunt Game's design viewpoints are explained in detail. In this part, UML diagrams are used to enhance intelligibility. In this part, we explain the main design viewpoints in detail.

4.2 Context Viewpoint

4.2.1 Design Concerns

There are two main concerns in our system. There are user and game. User is divided into two as player and game creator in this project. Users are the people who play and manage this game respectively.

4.2.2 Design Elements

Design Entities: Design entities are user and its functions in the application. The use case diagram of user functions is shown in Figure 2.

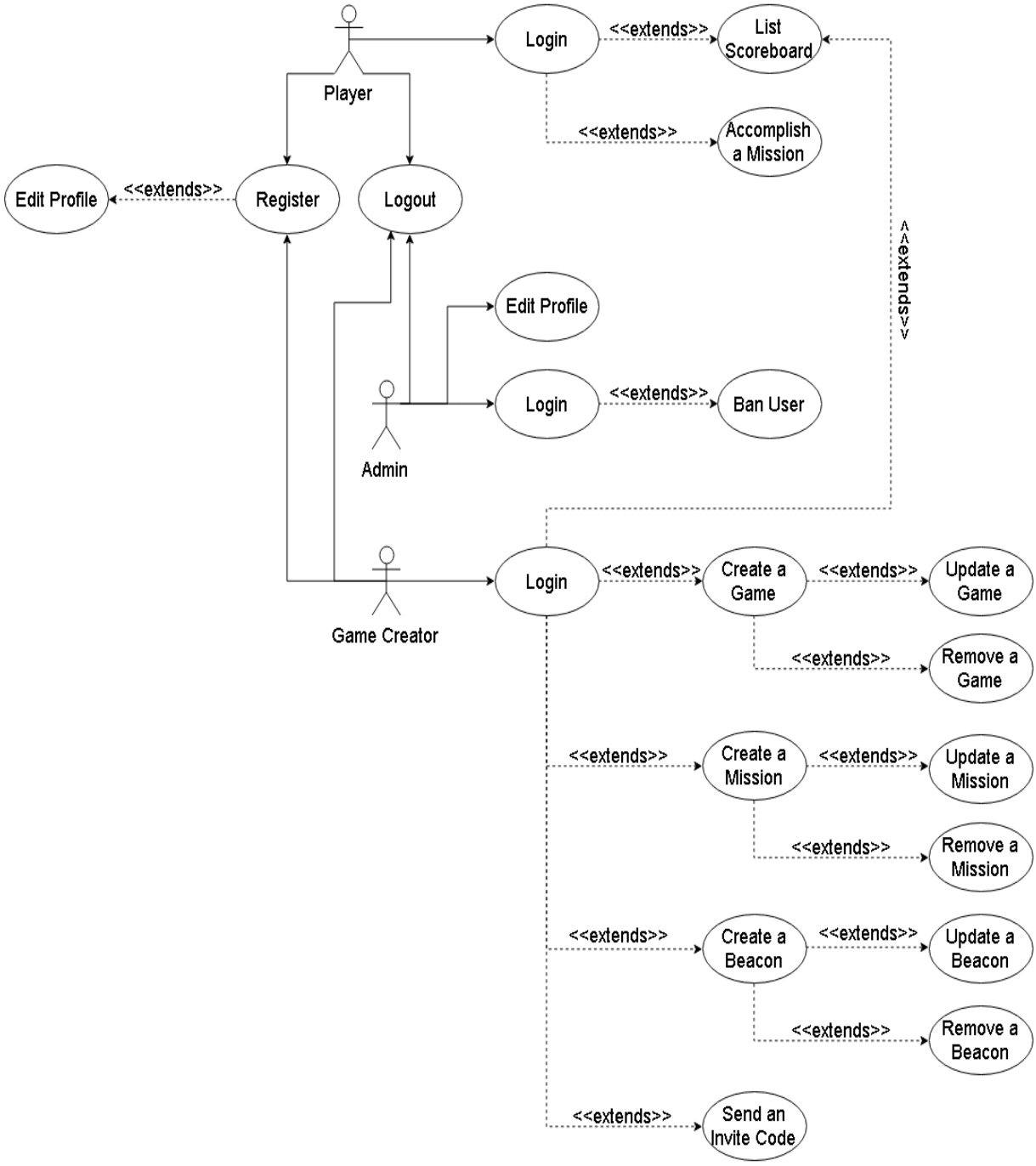

Figure 3 The use case diagram of all functions

4.2.2.1 Login

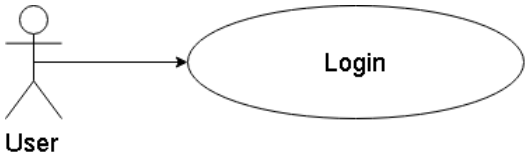

Figure 4 The use case diagram of login

Use Case Number	1
Use Case	Login
Summary	User can log in to the system with his/her username and password.
Actor	Player, Game Creator, Admin
Trigger	Login Button
Primary Scenario	To log in to the system, firstly, User must be registered to the system. After that, he/she can login with writing his/her username and password and clicking to login button.
Exceptional Scenario	• Not registered. • Incorrect information to log in.
Pre-Conditions	User must register to system.
Post-Conditions	User can enter the system.
Assumptions	User must be connected to the Internet.

4.2.2.2 Register

Figure 5 The use case diagram of register

Use Case Number	2
Use Case	Register
Summary	User can register the system to log in and use the application.
Actor	Player, Game Creator
Trigger	Register Button

Primary Scenario	After User get the application, User can register in registration page with required information.
Exceptional Scenario	Username is being used error.
Pre-Conditions	• User must have the application. • User must have an e-mail address.
Post-Conditions	After registration, user can enter the system.
Assumptions	User must be connected to the Internet.

4.2.2.3 Logout

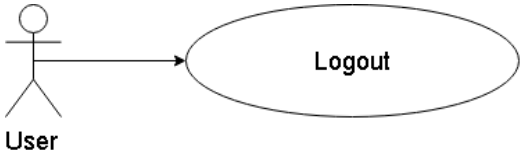

Figure 6 The use case diagram of logout

Use case number	3
Use case	Logout
Summary	User can logout from the system.
Actor	Player, Game Creator, Admin
Trigger	Logout Button
Primary Scenario	After user login the system, user can logout from the system anytime.
Exceptional Scenario	None
Pre-Conditions	User must log in to the system.
Post-Conditions	User can see login page.
Assumptions	User has already log in to the system.

4.2.2.4 Accomplish a Mission

Figure 7 The use case diagram of accomplish a mission

Use case number	4
Use case	Accomplish a Mission
Summary	Player can accomplish missions to earn points.
Actor	Player
Trigger	Accomplish a Mission Button
Primary Scenario	After player accomplishes a mission, player can earn points with connecting to a Beacon device.
Exceptional Scenario	• Mission is not over yet. • Beacon connection is failed.
Pre-Conditions	Player must open Bluetooth module of his mobile device.
Post-Conditions	After completing a mission, player earns points and improves his/her score.
Assumptions	• Player must be connected to the Internet. • Player's mobile device's Bluetooth module must stand on.

4.2.2.5 List Scoreboard

Figure 8 The use case diagram of list scoreboard

Use case number	5
Use case	List Scoreboard
Summary	User can list a scoreboard.
Actor	Player, Game Creator
Trigger	List Scoreboard Button
Primary Scenario	After user begins to play or create a game, user can list the scoreboard which belongs to the game.
Exceptional Scenario	None.
Pre-Conditions	User must join a game or create a game.
Post-Conditions	User can list the scoreboard of the game.

Assumptions	User must be connected to the Internet.

4.2.2.6 Create a Game

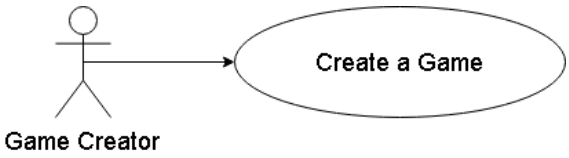

Figure 9 The use case diagram of create a game

Use case number	6
Use case	Create a Game
Summary	Game Creator can create a game.
Actor	Game Creator
Trigger	Create a Game Button
Primary Scenario	None.
Exceptional Scenario	None.
Pre-Conditions	None.
Post-Conditions	Game Creator can create a game for players and players can join a game.
Assumptions	Game Creator must be connected to the Internet.

4.2.2.7 Update a Game

Figure 10 The use case diagram of update a game

Use case number	7
Use case	Update a Game
Summary	Game Creator can update a game.
Actor	Game Creator
Trigger	Update a Game Button

Primary Scenario	Game Creator need to have at least one game.
Exceptional Scenario	If Game Creator does not have a game, he/she cannot change anything.
Pre-Conditions	Game Creator needs to have at least one game.
Post-Conditions	Game Creator can update a game.
Assumptions	Game Creator must be connected to the Internet.

4.2.2.8 Remove a Game

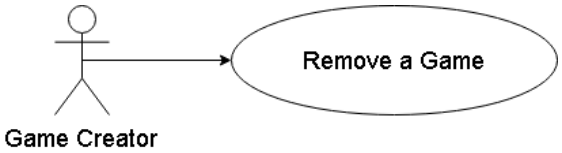

Figure 11 The use case diagram of remove a game

Use case number	8
Use case	Remove a Game
Summary	Game Creator can remove a game.
Actor	Game Creator
Trigger	Remove a Game Button
Primary Scenario	Game creator needs to have at least one game.
Exceptional Scenario	If Game creator does not have a game, he/she cannot remove anything.
Pre-Conditions	Game creator needs to have at least one game.
Post-Conditions	Game Creator can remove a game.
Assumptions	Game Creator must be connected to the Internet.

4.2.2.9 Create a Beacon

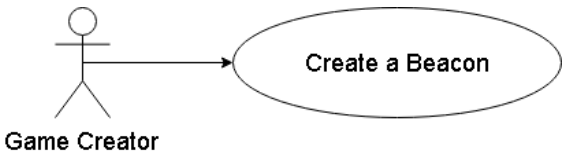

Figure 12 The use case diagram of create a beacon

Use case number	9
Use case	Create a Beacon
Summary	Game Creator can create beacons to expand his area.
Actor	Game Creator
Trigger	Create a Beacon Button
Primary Scenario	None.
Exceptional Scenario	None.
Pre-Conditions	None.
Post-Conditions	Game Creator can expand his/her area, and he/she can add missions on it.
Assumptions	Game Creator must be connected to the Internet.

4.2.2.10 Update a Beacon

Figure 13 The use case diagram of update a beacon

Use case number	10
Use case	Update a Beacon
Summary	Game Creator can update a beacon.
Actor	Game Creator
Trigger	Update a Beacon Button
Primary Scenario	Game Creator has to have at least one beacon in the system.
Exceptional Scenario	Game Creator has no beacon.
Pre-Conditions	Game Creator needs to have at least one beacon in the system. After that, he/she can update its condition.
Post-Conditions	Game Creator can update a beacon.
Assumptions	Game Creator must be connected to the Internet.

4.2.2.11 Remove a Beacon

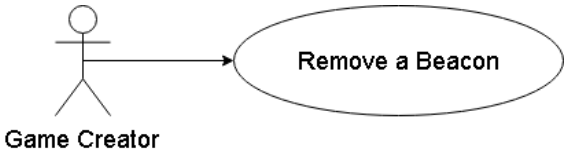

Figure 14 The use case diagram of remove a beacon

Use case number	11
Use case	Remove a Beacon
Summary	Game Creator can remove a beacon.
Actor	Game Creator
Trigger	Remove a Beacon Button
Primary Scenario	Game Creator has to have at least one beacon in the system.
Exceptional Scenario	Game Creator has no beacon.
Pre-Conditions	Game Creator needs to have at least one beacon in the system. After that, he/she can remove it.
Post-Conditions	Game Creator can remove a beacon.
Assumptions	Game Creator must be connected to the Internet.

4.2.2.12 Create a Mission

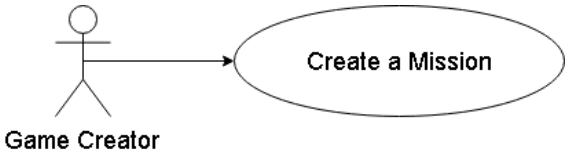

Figure 15 The use case diagram of create a mission

Use case number	12
Use case	Create a Mission
Summary	Game Creator can create a mission for players to do.
Actor	Game Creator
Trigger	Create a Mission Button
Primary Scenario	If Game Creator does not have a Beacon, he/she need to add a Beacon. After that, he/she can create a mission.

Exceptional Scenario	None.
Pre-Conditions	If Game Creator does not have a Beacon, he/she need to add a Beacon first.
Post-Conditions	Player can see the missions.
Assumptions	• Game Creator must be connected to the Internet. • Game Creator must have at least one Beacon in the system.

4.2.2.13 Update a Mission

Figure 16 The use case diagram of update a mission

Use case number	13
Use case	Update a Mission
Summary	Game Creator can update mission.
Actor	Game Creator
Trigger	Update a Mission Button
Primary Scenario	If Game Creator created a mission before, he/she can update that mission.
Exceptional Scenario	• The mission's date is over. • Game creator has no mission.
Pre-Conditions	Game Creator needs to create a mission first.
Post-Conditions	Player can see the updated mission.
Assumptions	• Game Creator must be connected to the Internet. • Game Creator must have at least one mission.

4.2.2.14 Remove a Mission

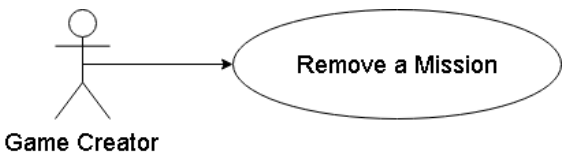

Figure 17 The use case diagram of remove a mission

Use case number	14
Use case	Remove a Mission
Summary	Game Creator can remove a mission.
Actor	Game Creator
Trigger	Remove a Mission Button
Primary Scenario	Game Creator needs to have at least one mission.
Exceptional Scenario	If Game Creator does not have a mission, he/she cannot remove anything.
Pre-Conditions	Game Creator needs to have at least one mission.
Post-Conditions	Game Creator can remove a mission.
Assumptions	Game Creator must be connected to the Internet.

4.2.2.15 Edit Profile

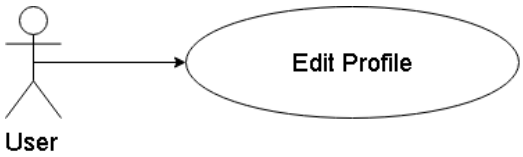

Figure 18 The use case diagram of edit profile

Use case number	15
Use case	Edit Profile
Summary	User can update his/her personal information.
Actor	Player, Game Creator
Trigger	Edit Profile Button
Primary Scenario	User needs to be registered to the system first. After that, he/she can change his/her information.
Exceptional Scenario	User is not registered.
Pre-Conditions	User needs to be registered to the system first. After that, he/she can change his/her information.
Post-Conditions	User can update his/her information.
Assumptions	User must be connected to the Internet.

4.2.2.16 Send an Invite Code

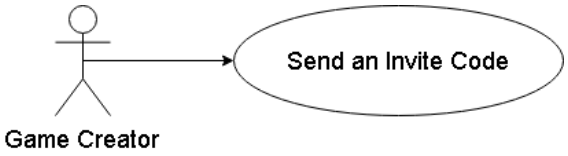

Figure 19 The use case diagram of send an invite code

Use case number	16
Use case	Send an Invite Code
Summary	Game Creator can invite players to his/her game.
Actor	Game Creator
Trigger	Send an Invite Code Button
Primary Scenario	After a game is created, Game Creator can invite players.
Exceptional Scenario	Game is not created.
Pre-Conditions	Game Creator need to create his/her game first.
Post-Conditions	Game Creator can invite players.
Assumptions	Game Creator must be connected to the Internet.

4.2.2.17 Ban User

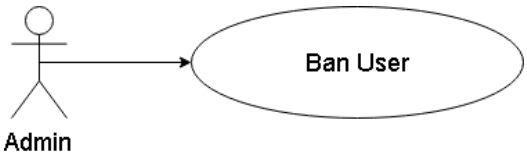

Figure 20 The use case diagram of ban user

Use case number	17
Use case	Ban User
Summary	Admin can ban users.
Actor	Admin
Trigger	Ban User Button
Primary Scenario	None.
Exceptional Scenario	User is not registered.
Pre-Conditions	None.

Post-Conditions	Admin can ban users.
Assumptions	Admin must be connected to the Internet.

4.3 Composition Viewpoint

4.3.1 Design Concerns

With the help of composition viewpoint software process will be understood. In this part, main work components and their inside components will be explained. There are four main work components in this software. Namely: Database, web server, client, and BLE Beacon.

4.3.2 Design Elements

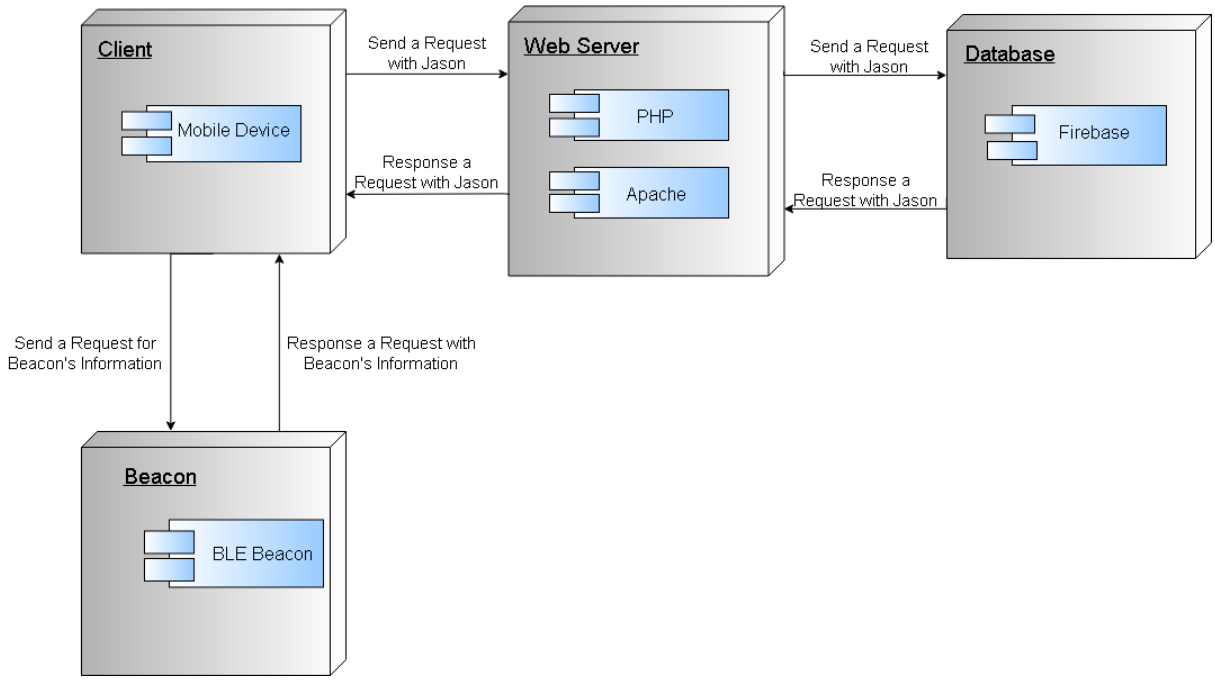

Figure 21 The deployment diagram of the project

Design Entities: There are four main design components in our project which are database, web server, client, and BLE Beacon. With web server, the client and the database has a connection.

Design Attributes: Design attributes are discussed in the following two parts.

4.3.2.1 Function Attribute

Database, BLE Beacon, web server, and client are the main components of our project. Web server is responsible for providing an interaction between the client and the database. The database stores information and BLE Beacon is responsible for detecting location.

4.3.2.2 Subordinates Attribute

All of the components mentioned above are composed together to build this project.

4.4 Logical Viewpoint

4.4.1 Design Concerns

The logical viewpoint identifies all classes and relations between classes. The goal of this viewpoint is to define and simplify the system design.

4.4.2 Design Elements

4.4.2.1 Class Relations

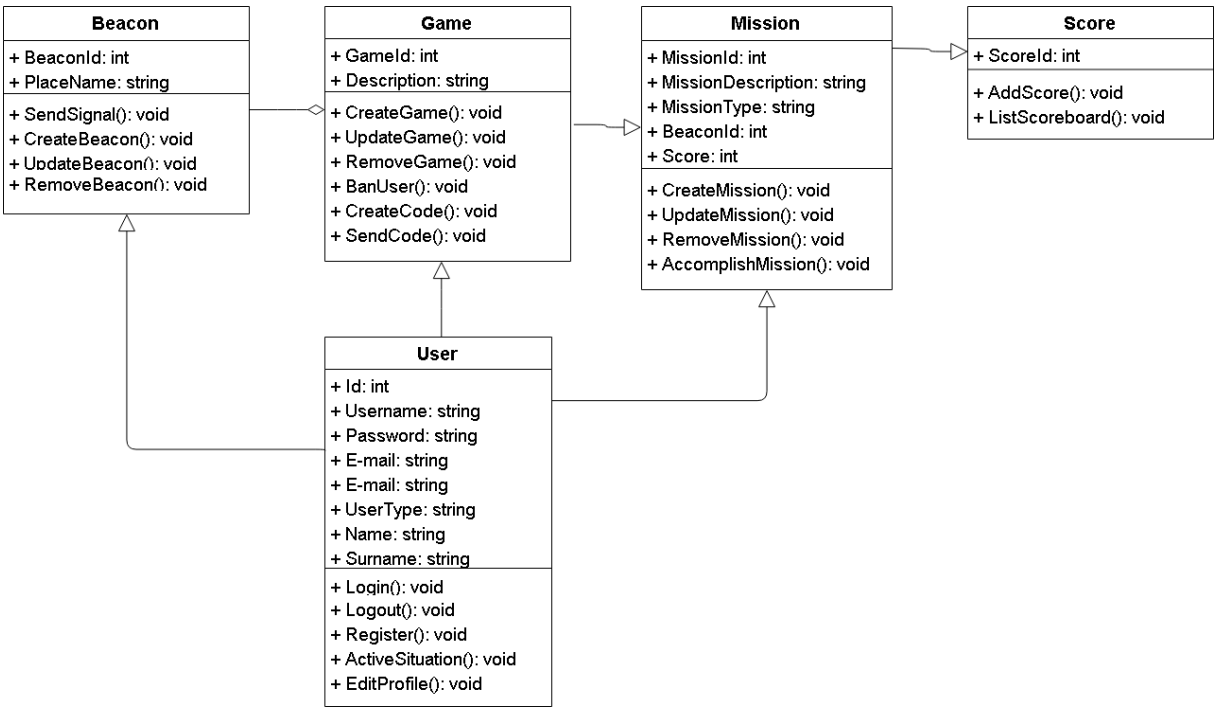

Figure 22 The class diagram of the project

4.4.2.2 User Class

Method/Field	Definition
Int Id	Unique id of user.

String Username	The username of user.
String Password	The password of user.
String E-mail	User’s email address.
String UserType	The authority of user.
String Name	User’s name.
String Surname	User’s surname.
Void Login()	The way of enter to the system.
Void Logout()	The way of logout from the system.
Void Register()	The way of register to the system.
Void ActiveSituation()	User’s active or deactive situation.
Void EditProfile()	The way of edit user’s profile information.

4.4.2.3 Game Class

Method/Field	Definition
Int GameId	The unique id of game.
String Description	The description of game.
Void CreateGame()	The way of create a game in the system.
Void UpdateGame()	The way of change a game in the system.
Void RemoveGame()	The way of remove a game from the system.
Void BanUser()	The way of ban a player in a game.
Void CreateCode()	The system creates an invite code for invitation.
Void SendCode()	The way of send an invitation code to players.

4.4.2.4 Mission Class

Method/Field	Definition
Int MissionId	The unique id of mission.
String MissionDescription	The description of the mission.
String MissionType	The type of mission.
Int BeaconId	The beacon id that work for the mission.
Int Score	The score that belongs to the mission.
Void CreateMission()	The way of create a mission.

Void UpdateMission()	The way of change a mission.
Void RemoveMission()	The way of remove a mission.
Void AccomplishMission()	The way of end a mission.

4.4.2.5 Score Class

Method/Field	Definition
Int ScoreId	The unique id of score.
Void AddScore()	The way of add score to user's total score.
Void ListScoreboard()	The way of see the scoreboard of the game.

4.4.2.6 Beacon Class

Method/Field	Definition
Int BeaconId	The unique id of beacon.
String PlaceName	The location of beacon.
Void SendSignal()	The signal that beacon sends.
Void CreateBeacon()	The way of add a beacon.
Void UpdateBeacon()	The way of change a beacon's information.
Void RemoveBeacon()	The way of remove a beacon.

4.5 Information Viewpoint

The information viewpoint describes the relationships between the classes. The class diagram explained differently with ER diagram. We can easily understand the relationships between classes with this diagram.

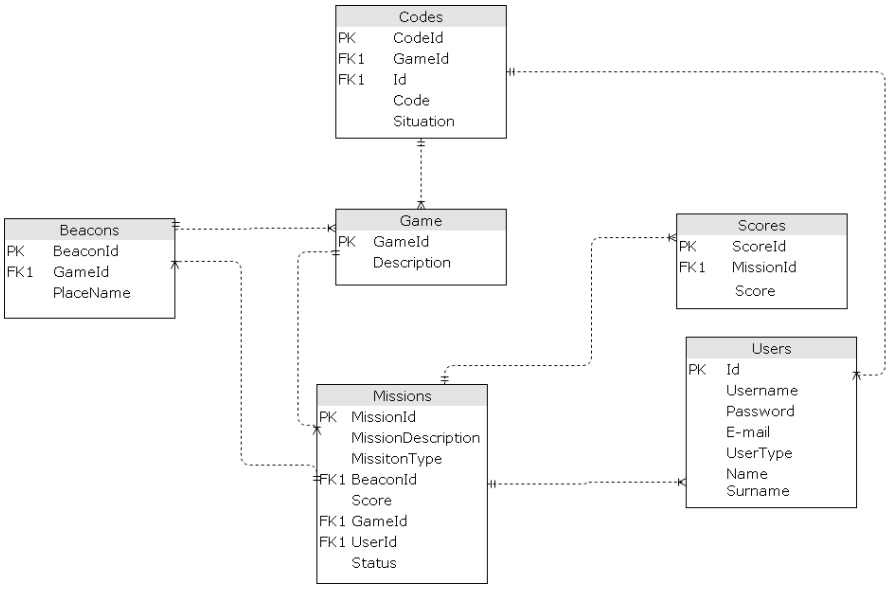

Figure 23 The ER diagram of the project

4.6 Interface Viewpoint

4.6.1 Design Concerns

The interface viewpoint provides all the information for anyone who is interested in our application. In the parts below, we created drafts of our application.

4.6.2 Design Elements

4.6.2.1 Welcome Page

After downloading and opening the Scavenger Hunt Game application, Welcome page is displayed. The user is directed to the login page with clicking to Start button.

Figure 24 Welcome page

4.6.2.2 Login Page

After clicking the Start button, Login page is displayed. After user fills the username and password fields, he/she should click the Login button. If the user does not have an account, he/she can register by using Register button.

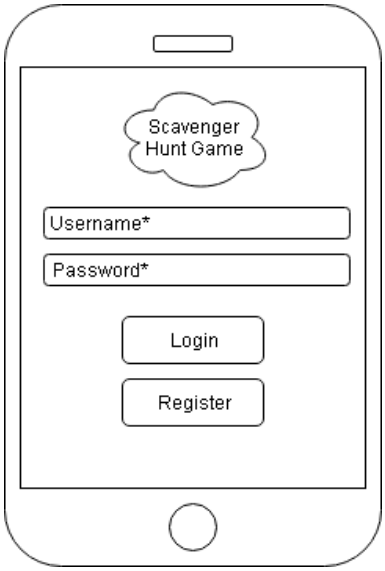

Figure 25 Login page

4.6.2.3 Register Page

Users who want to register to Scavenger Hunt Game can register to application after filling the required fields. After clicking the Register button, the user should be registered to the system.

Figure 26 Register page

4.6.2.4 Player's Home Page

This page is the home page of player. Player can edit his/her personal information by clicking to Edit Profile button. He/she can logout from the system with the Logout button. With clicking to Join a Game button, the list of games will display and player can choose any game he/she wants to join.

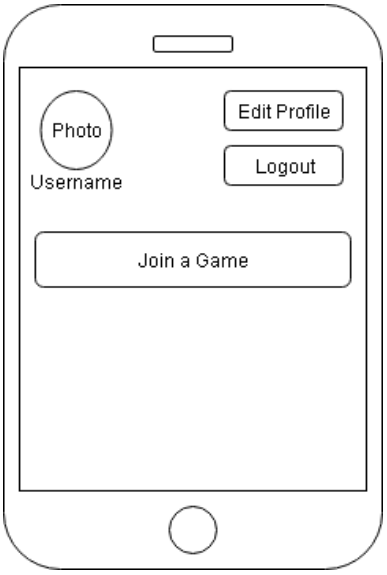

Figure 27 Player's home page

4.6.2.5 List of Games Page

When user clicked the Join a Game button, this page is opened. This page includes all the games in the system. And, there is a Home button to return the home page.

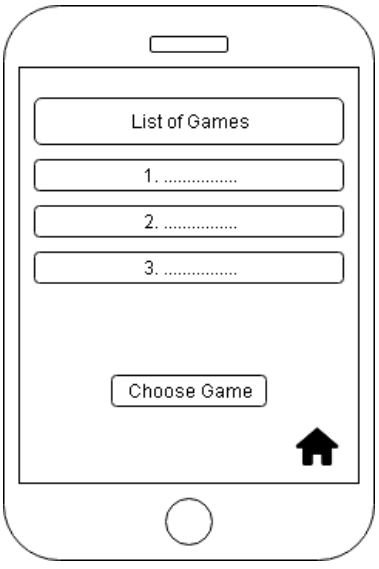

Figure 28 List of games page

4.6.2.6 Chosen Game Page

This page includes the operations of a player can do in the game. There is a Do a Mission button which redirects player to a mission and a List Scoreboard button which shows all of the player's scores in it. And, there is a Home button to return the home page.

Figure 29 Chosen game page

4.6.2.7 Mission Page

After clicking the Do a Mission button, a mission is displayed in the mission page. If player does a mission, he/she should connect to a Beacon device to complete the mission by clicking Accomplish a Mission Button. And, there is a Home button to return the home page.

Figure 30 Mission page

4.6.2.8 Game Creator's Home Page

This page is the home page of game creator. Game creator can edit his/her personal information by clicking to Edit Profile button. He/she can logout from the system with the Logout button. Game creator can create a game by clicking to Create a Game button, remove a game by clicking to Remove a Game button and change a game by clicking to Update a Game button.

Figure 31 Game creator's home page

4.6.2.9 Active Game Page

In this page, game creator can add a mission by clicking to Create a Mission button, change a mission by clicking to Update a Mission button, remove a mission by clicking to Remove a Mission button. He/she also can add a Beacon by clicking to Create a Beacon button to expand his/her area, change a Beacon's information by clicking to Update a Beacon button and remove a Beacon by clicking to Remove a Beacon button. There is also a List Scoreboard button to see scores of game's players. And, there is a Home button to return the home page.

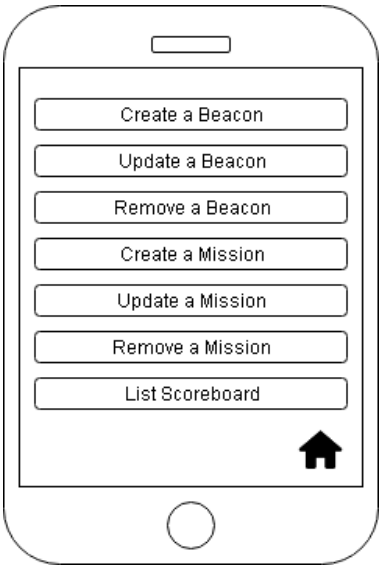

Figure 32 Active game page

4.6.2.10 Scoreboard Page

This page shows the scoreboard of the existing game. This page exists for both player and game creator but the contents can be different from each others. And, there is a Home button to return the home page.

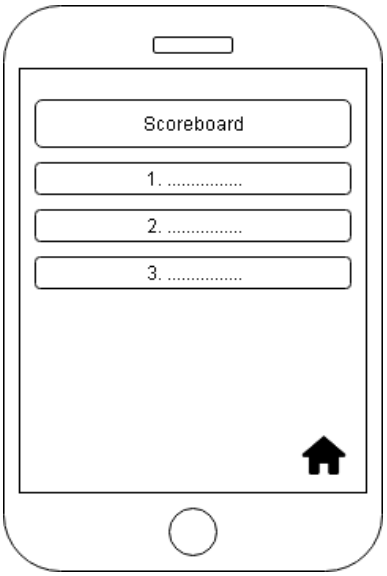

Figure 33 Scoreboard page

4.6.2.11 Edit Profile Page

In this page, users can change their personal information. If user updates any of the fields, in order to save the changes, he/she should click to Update button. If he/she does not want to

change anything, he/she should use the Cancel button. Both Cancel and Update buttons will redirect user to his/her home page.

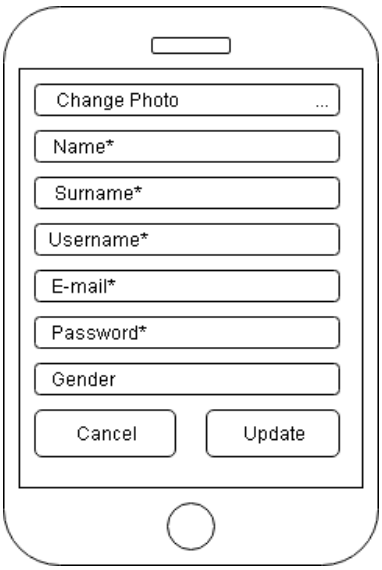

Figure 34 Edit profile page

4.6.2.12 Admin's Home Page

This page is the home page of admin. Admin can edit his/her personal information by clicking to Edit Profile button. He/she can logout from the system with the Logout button. With clicking to Ban User button, he/she can ban a player or game creator.

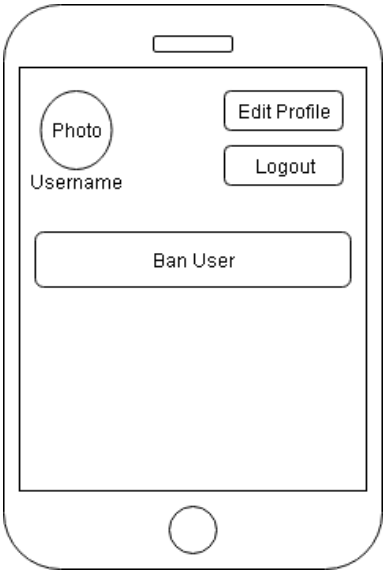

Figure 35 Admin's home page

4.7 State Dynamics Viewpoint

4.7.1 Design Concerns

The state dynamics viewpoint shows the behaviour of the system when there are some specific events. This viewpoint also related to the logical view. When a user enters the application, a user is redirected to the login page. A user must register to Scavenger Hunt Game to log in to the system. If the login is successful, a user is redirected to user's home page according to their authority. There are three home pages. There are one for players, one for game creators, and one for admins. From player's home page, a player can join a game and do a mission which game creator created for players. Also, when a player comes to the end of the mission, system check the mission's Beacon, and after the confirmation, the system gives the player a score. From game creator's home page, a user can create a game, update a game or remove a game. Also, a game creator can create a mission for his player to do or update and remove that mission. From admin's home page, admin can ban users from the games. To log out, there is a logout button for all of the users, and if users trigger that button, they can logout from the system.

4.7.2 Design Elements

In the state diagram, design elements begin with a start state, and it is divided into two different states; one is for a game creator, and another one is for a player. State diagram continuous with login states, display menu states, create a mission state, update a mission state, remove a mission state, create a beacon state, remove a beacon state, update a beacon state, accomplish a mission state, list scoreboard state, logout state, and end with end state.

Design entities can be observed with using the state transition diagram in Figure 36 below.

Figure 36 The flowchart diagram of the project