Chapter 5

Game Design Document and game development process

5.1 Introduction

This chapter provides a description of the Game Design Document and Game development process which with continues from Chapter 4. Our game based learning for outdoor activities' made from our proposed design based on the framework of the curriculum matching model with respect to the knowledge co-creation from among the experts. Then the balance the fun and learning regarding the flow theory from the previous chapter. This chapter will focus on the implementation phase for a high-level system, which will work as proof of the concepts. This section will focus on the conversion of the knowledge to the standard document name Game design document (GDD) which the game designer team can understand. We proposed the architecture, and developed the prototype for game based learning with the collaboration between Cholprathanpateak School, Chiangmai zoo and ICEO Company.

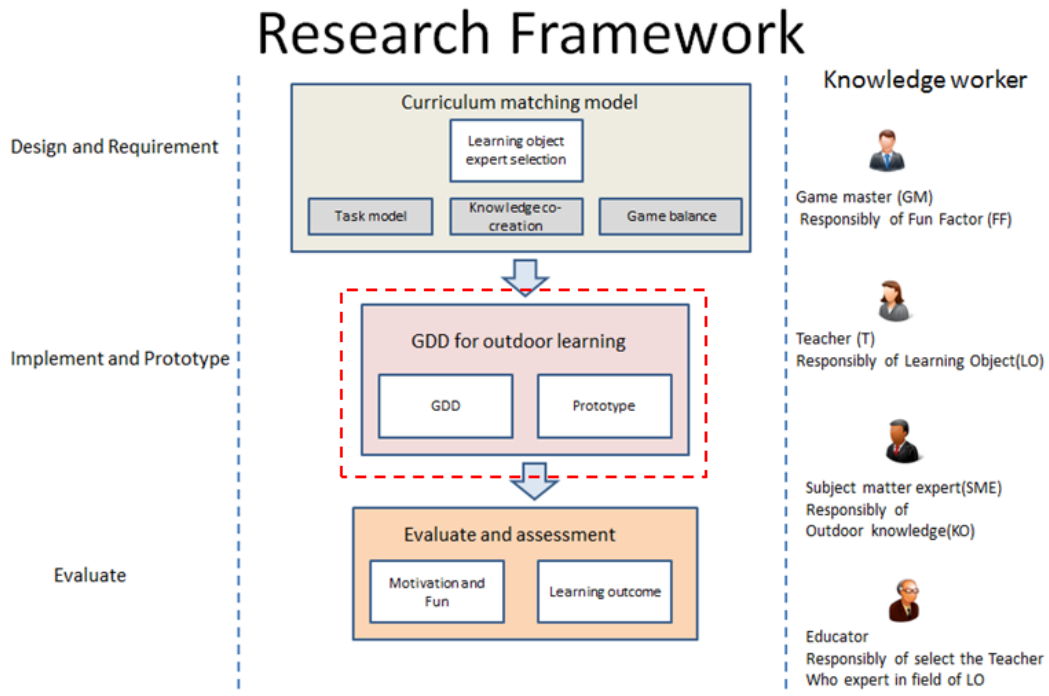

Figure 5.1 overall frameworks for stand of game development and evaluation and assessment

In this chapter focuses on the detail of the Game design document and the game's development process. In this process the Game master is the persons responsible with software's development followed by the design document working with other experts who are familiar with the game's development described in the previous chapter. In our game design document for outdoor learning activities of this consisted with three main parts: KA-document, KC-document and KM-document as seen in the previous chapter too. In this chapter, we will provide in greater detail the software architecture and the details for each specific part in terms of functional specification, technical specification, level designs and educational purposes.

5.2 Our propose guidelines for the Game Concept

Our approach of the guidelines specification outlines consists of three part knowledge analysis (KA-document), knowledge co-creation (KC-document) and knowledge management (KM-document) which add to the features and functionality of the game. The knowledge expert oversees the team work with this guideline and doing

the work responsible for approving the Game Design Document. The guidelines specification is a culmination of the whole idea, discussions and criticisms relating to the main point. It could be seen as the skeleton of the vision as expressed in the game based learning concept. It could make sure that both fun and learning is achieved. The guideline is a platform which links the technical specification and schedule that is derived and the development term can begin the process.

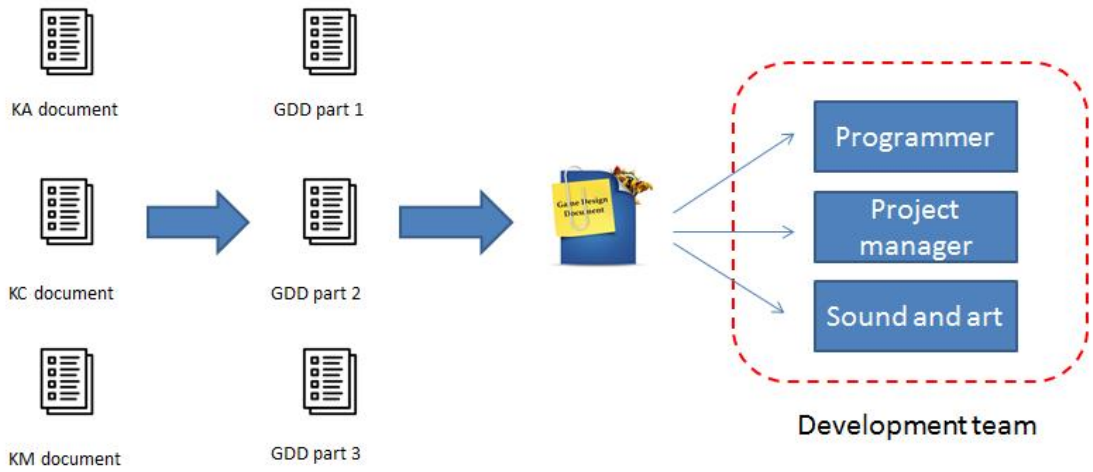

Figure 5.2 our model of Game Design Document

It would very important that the GDD is all written from the Game Master's perspective. In the meantime, the developers showed the experienced or interacted with what is more important and should be the aim of the Game Design Document. Our guideline is very tempting to whole knowledge expert but not especially to the programmer only but to create something that's very effective for the system's orientation. This often leads to a hard-to-understand and distraction documents. Readers are really just looking to this document to visualize what's in the game and not how it works. The length of the GDD can vary from ten pages to a few hundred, depending on the complexity of the game, a page count should not be the main focus, but rather than aim that is effective for both, the fun and the learning point. Here is the summary for the Game Design Document proposed by Chris Taylor in addition to the main three parts of the document.

Table 5.1 The guidelines specification outlines consists of three part knowledge analysis

Section	Title Page	Description
Title game	Title game	
	Game Name or Main advertize, It can put the high concept sentence or subtitle	It can also add a subtitle or high concept sentence or advertize of the game we want them to play.
The Game Overview	The Game Overview	Explain the briefly of the important aspects of the overall of game
	Game Concept	Describes briefly the very short concept of game
	Genre	Select the game genre that suit our game
	Target Audience	Describes the range of target who would focus on
	The Game Flow and Summary of gameplay	How the player play or learn through the gameplay and game mechanic.Explain of the game and the interface
	Style, concept art , feel and look	How the basic idea of the game style, concept , describe feel and look of the game? How is the visual concept art style of the game?
The Gameplay and Mechanics of each elements	The Gameplay and Mechanics of each elements	
	Gameplay	Describes for the general concept idea of gameplay
	Game Progression	Point out the progression of the game

Table 5.1 (continued)

Section	Title Page	Description
	Mission/challenge Structure	What is the mission how to challenge it ?
	Puzzle Structure	If the game described in term of puzzle
	The game objectives or purpose	What are the main and the sub of objectives of the game?
	The Play Flow of gameplay	What is the flow during the game play
	The each detail of of game elements	The model for the virtual world that the game works under the rule of the virtual world. Explain of how to simulation of a world.What are the rules to the game, both implicit and explicit.
	The Physics engine for 2D/3D	What is the the physical universe can interact or How it work?
	Character movement in the gameplay	.How does the player move in the universe work
	The interaction of objects	It can interact with people or how to pick them up and move them
	Action of player	How the means of communication are used to locate player .whatever buttons and switches are used, interacting with objects
	Combat of player	how is this specifically modeled?If there is combat or even conflict
	Resource and Economy of gameplay	What is the resource and economy of the gameplay?How the important of economy system ? How it balance ? How does the gameplay it work?

Table 5.1 (continued)

Section	Title Page	Description
	The screen flow of gameplay	.Description of the purpose of each screen and the graphical description of how each screen is related to every other.
	Game menu and Game Options	how the effective related to the game play and mechanics?What are the options
	The system for replaying and saving	The Game system for replay value is this game worth to play again ? and is this game have saving for each progress of player ?
	Easter Eggs and Cheats	
Story, Setting and Character	Storytelling, setting up and Character	
	Story and Narrative of gameplay	The cut scenes descriptions implement to the actors, history of the game how important of each character, the setting, and the script or storyboard.Explain the backstory of the game, plot elements,cutscenes and game progression.
	The virtual of Game World	
	The General feel and look of world	A graphical description of general look and feel of overall of the game
	Areas	How to design levels and How to use it, how it link to other virtual world? physical characteristics and the general description as well as what it relates to the rest of the world
	The characters of gameplay	For each character should explain how to link to the main and sub story and to link to other characters the backstory, personality, appearance, animations, abilities

Table 5.1 (continued)

Section	Title Page	Description
Game design Levels	Game design Levels	For each level should put the synopsis that required introductory material but it should depending on the game,physical description of the model map and the critical path that the player needs to take, and How to encounters are incidental and importantly.
	The Training Level or Tutorial	For each training level should include a basic of control movement but it should depend on the gameplay for example it might help to improve how to use the physical description of the map, the navigation of the path that the player needs to learn, the required introductory material
The game Interface	UX/UI for the Interface	
	The Visual System	What menus are you displaying? What is the camera model?
	The Control System	For the HUD,What is the game player can control the game? What are the specific input for the gameplay? what is on it?
	sound effects , music and audio	Audio music 2d 3d system what kind of music and effect
	Help System	Description the system to help player in game what is that main system
A.I (Artificial Intelligence)	Artificial Intelligence	

Table 5.1 (continued)

Section	Title Page	Description
	Enemy AI and Opponent	The player can active against to enemy which plays against the game player and it should requires the main decision making
	Friendly Characters and noncombat system	Explain the npc system for help player
	How to support AI player and collision detection, enemy strategy , Pathfinding	The player can interact the collision detection and the pathfinding system
The Technical specific	Technical	
	Focus Hardware	Game platform such as PC, game console or handhole
	How to develop software and hardware, within the Game Engine.what is the main target platform or limitation	Compare and select of the game engine use for game
	Network requirements	Are there any Multiplayer
Game Art	Game concept art , the key assets and What they are being developed in term of intended style.	how they could being implemented for Intended style.

5.3 Game Design Document for Outdoor learning part 1

The first part of the Game Design Document for Outdoor learning focuses on the problems and to identify them in the wider context of an interesting place e.g. museum. It focuses onto the specifics of the learning place e.g. the trail description, time duration, number of points, photographic information to name a few. At the end, the potential

solutions from the educational points of view are examined in order to get a real and explicit understanding in terms of the learning place context itself. In this study, we used this design document for interviewing 1 in order to get the perspective of the Subject Matter Expert for their learning place. The results from the interview were conducted in using the following guideline as seen in table 5.1.

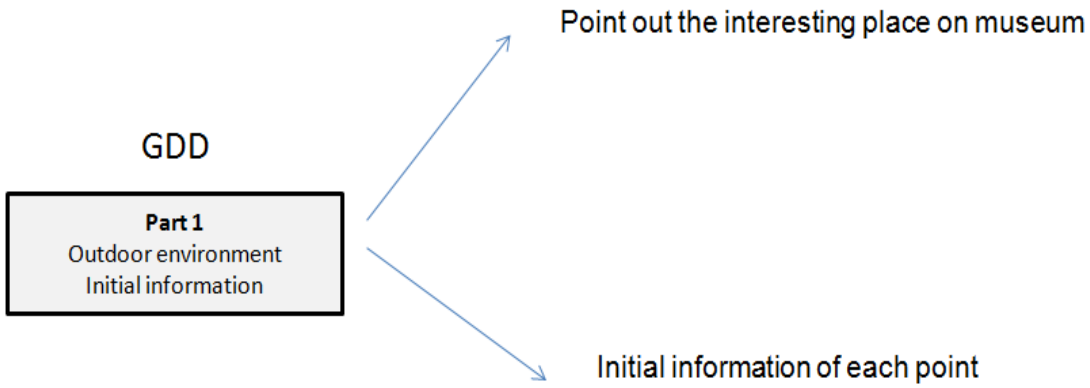

Figure 5.3 The main objective of Game Design Document of chapter 1

From the first chapter, we saw this guideline has much identified factors for learning place environment. However, the weakness of collaboration and lack of knowledge from other participants such as Teacher and Game Master seem to be the major obstacles for the game's development. Thus, the main strategies of the game based learning for Outdoor activities were set for improving the learning objective and linking the knowledge to the environment of the members. The solution proposed by the our team can be considered as planned activities for supporting the strategy with two main objectives for identifying the interesting place in a museum and initial information of each point. The next guideline will present the elements of the core game mechanic.

5.4 Propose Game Design Document for Outdoor learning part 2

The second part of the guideline of the Game Design Document for Outdoor activities concentrates on specific aspects of the applying the education field to the learning place. This consists of seven guidelines: name of point, subject, core indicator, learning objective of the teacher, knowledge, Indigenous knowledge and questions. The

name of point focuses on the participant, unit, or group involved in the core activity of the learning environment. The subject aspect focuses on the subject in the education field process which is relevant to the learning objective relative to the learning place. The Indigenous knowledge aspect refers to the knowledge which is involved around environment. The question of point in the Game based learning is derived from all knowledge guidelines in the previous model. This can be any information system, equipment, or technology which is used within the learning place. The knowledge aspect represents the knowledge element which is required to accomplish the task which student needs to know and do. Lastly, the final question is the question needed to formalize from all knowledge that comes from the Game master and subject matter expert. The whole guideline from the analysis in these aspects is seen in table 3. It describes the basic guidelines of the Game based learning and is considered as an outline for the following reasons. These components are identified by particular codes in order to be referred to accurately by other game design guidelines.

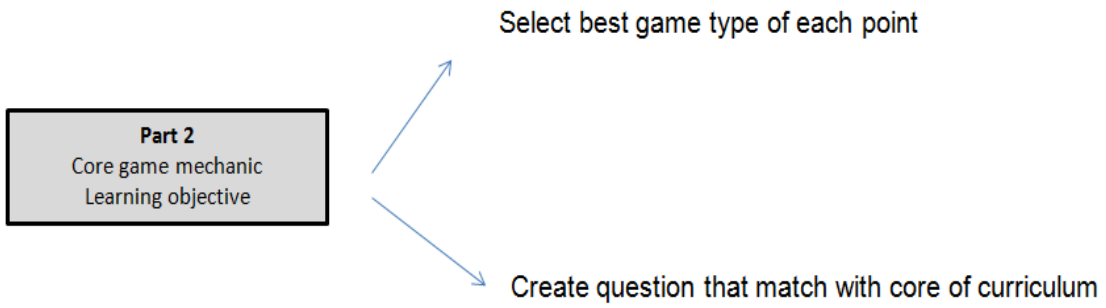

Figure 5.4 the main contribute of GDD part 2

The details of each part or guideline were described in the model map in the previous part. The name of point guideline shows the major point that the students need to learn. The activity starts from obtaining design information until the learning objective is seen and can be divided into three process. Then, the knowledge aspect shows the available linking of knowledge within the learning place. From this aspect, we can see that the Game based learning, in this case study does not rely on high technology or equipment for their development due to this Game based learning is composed of the learning place related to learning object. The knowledge aspect shows a list of required the knowledge for achieving the goal of each process. Indigenous

knowledge aspect describes a specific characteristic of learning place which applies specifics only in that place but have never been discussed as of yet. This problem will be discussed again in the co-creation model step. The defined processes in this design document will be broken down in this chapter.

5.5 Game Design Document for Outdoor learning Part 3

The final processes from the Game Design Document for Outdoor learning is broken down in to greater detail and finalizes the whole guidelines from both part 1 and part 2. The education part of the process is summarize and broken down into the smaller object learning tasks for classifying the knowledge-intensive task from around the environment in place. In order to finalize the Game based learning for conclusion of the whole learning object is linked the curriculum with a consensus from Game Master and Subject Matter Expert and Teacher on the knowledge co-creation of each task to be obtained. There is no easy way for assessing task significance. In doing so we propose the algorithms for adjusting and concluded the time and learning object.

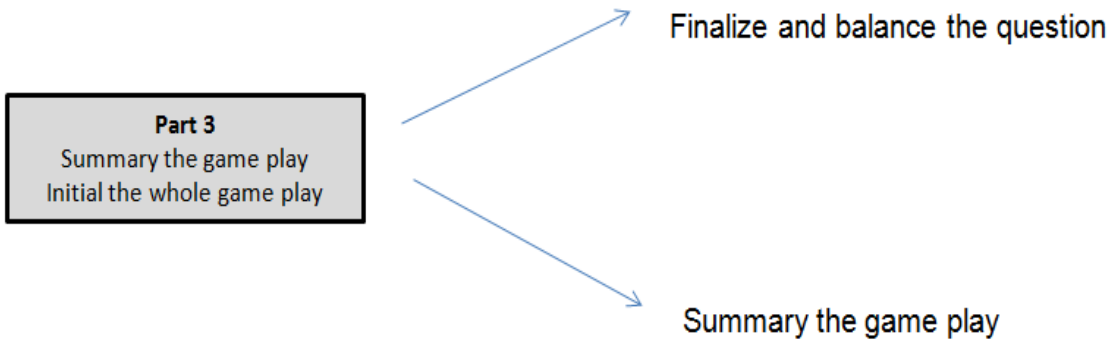

Figure 5.5 the main contribute of GDD part 3

The details of final part of the guidelines were described in the model map shown in Figure 4.14. The name of point guideline showed the major outcome that students needed to learn. The activity starts from obtaining the design information until of learning objective is dearly seen which can be separated into three process. The summary aspect shows the availability of linking knowledge within the learning place. The summary aspect shows a list of required knowledge for achieving the goal of each

process. Lastly, the summary aspect describes specific characteristics of learning place which applies specifics only in that place, but have never been discussed. These problems will be addressed again in the section of the co-creation model. The defined processes in this Game Design Document part 3 will be reviewed later on this chapter.

5.6 Proposed the game development process for outdoor learning

Similar with any creative development project such as the creative arts and architecture, educational game-based development processes also have the same basic processes which consist of many sub stages to make it perfectly. Before one starts the project, the team needs to meet up, plan the timeline and schedule of the project with name design document. In traditional software development process for the software engineering, there is the standard of document to do it. In detail of development process, it would divide to each phase that would involve activities such as the pre-production, production and post-production phase. For example the production phase is when the software actually runs and is ready for use or the application design to implementation.

However, in terms of development process of game based learning for outdoor classroom lifecycle. We would use specific improvement of the data into document for the pre-production phase which includes knowledge such as learning object and outdoor location sessions for game mechanics into the design document, which can use in the production phase of the traditional software engineering process. The production phase of traditional software engineering is actually the outcome and can use to adjust the core mechanics in the post-production phase of the Game based learning process life cycle. All steps of the production the game's release generally runs simultaneously with each other. Running all these tasks simultaneously helps in the co-creation of the asset element and feature programming to game developer.

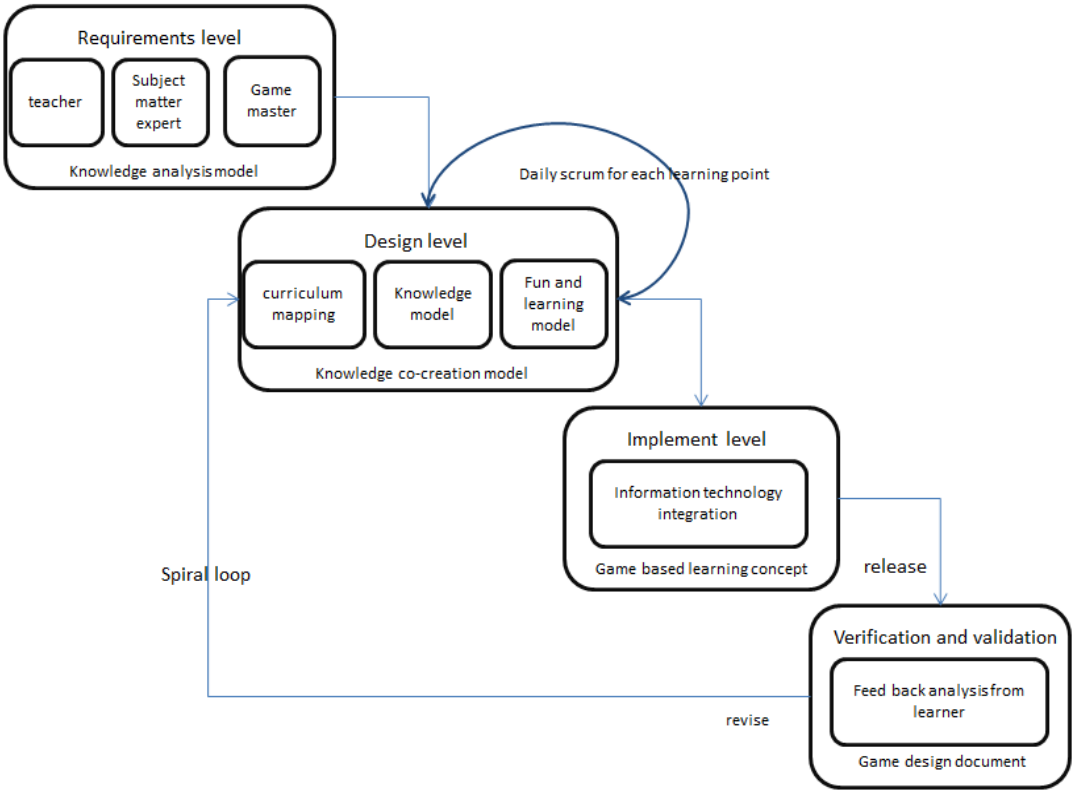

Figure 5.6 Overall of software engineering process for game based learning

Another thing in task of methodology involves communication between different stakeholders and the development team. It would be very important for game programmers, game designers and artists should have a good communication and understanding of each other's work in order to work well collectivity in order to get a good final product. Therefore, Game-based learning for outdoor classroom process life cycle is different from the traditional other software engineering process, and not to many researchers have studied the challenges faced by this domain (Haddad, 2001). A few of researchers have explicitly compared software engineering in terms of game entertainment process, but none of them has studied complete game based learning for outdoor classroom process life cycle and research topics under this domain in great detail. The most prominent observation made in these studies is that to address the challenges faced and the standard of document for the game-based learning process for outdoor classroom life cycle, more rigorous software engineering strategies must be used. For this study, one will provide and invent the new process on these topics and

their standard to develop game based learning for outdoor classroom improvement from the traditional software engineering process.

When the development team has created all the assets and necessary features, there is a phase of polishing and perfecting the project. This phase is called post production. The game studios use these phase to plan the development of a game. Throughout the game development there are specific events that are of great importance. These are called milestones. Most of the time, the milestones are at the end of each stage. The most common milestones are:

- **Alpha**: This is the milestone where the gameplay is functional. There are some assets to be completed and the game is able to run on the target platform.
- **Beta**: In this milestone all the features of the game are present and there is only bug fixing activities. Also, in this milestone, there cannot be any changes mode in terms of the concept of the game.
- **Code Release:** This is the final milestone of the game development, in which the development team can ship the game to the publishers for distribution.

There are many methodologies of developing software. Waterfall methodology is the oldest in terms of software engineering. It is a methodology that allows development to occur in phases that follow one another. After the last stage is being processed and executed, the whole development loops back to the beginning. The waterfall process has been introduced to the video game development too. Figure 5.2 depicts one example of this phases that is executed during the development of a video game or a small section.

Scrum is a very popular and agile methodology in game development. This methodology uses iterations of two or four weeks, known as sprints. An overview of Scrum is shown in Figure 4. In order to implement Scrum, a game development team needs to identify the scrum parts. The first and most important part of Scrum is the sprints. Each sprint has a specific goal, known as a sprint goal, which all the team members are trying to achieve by the end of an iteration. When this is over, the development team presents the outcome of the sprint goal to the stakeholders. Another

important part of Scrum development is the product backlog that is a prioritized list, consisting of all the requirements and features that are needed to for the game's development. The product backlog is not a static list that stays uninterrupted throughout the development. In every sprint the needs and requirements might change and at the same can occur to the product backlog. A subset of the product backlog is called the sprint backlog, which consists of detailed tasks for the development team throughout the sprint.

In agile development, these tasks are known as stories and will be called as such for the rest of the thesis. Apart from the parts of Scrum, there are also certain events that take place during a sprint. Sprint Planning is the first event in an iteration. In this meeting, the sprint goal is determined and set by the whole team. At the same time, the sprint backlog is constructed. In order for all the development team members to keep track of the daily progress, Scrum uses another event. This is called the Daily Scrum. It is a brief meeting of 10 to 15 minutes, whereby all the team members explains what work have done the previous day, what would do that day's goals and if they have encountered any problems. This meeting is an encouraging collaboration and has problem solving activities where all team members can take part. At the end of the sprint, there is the last event of Scrum. This event is returned to as the Sprint Retrospective, where the development team evaluates the progress of the game's development and investigates how the overall progress can be improved. Since Scrum should use the "inspect and adapt" policy, Sprint Retrospective is often seen as the "most important event in agile development".

The software engineering approach difference between the spiral model, waterfall model and agile gaming is that the process of the development software with cycle moves towards an eventual completion from right to lefts top to down for each quadrant in the models. We decided to use the spiral model in for framework. However, in the spiral model, the cycles move clockwise and go back many times over each earlier quadrant, in a repetitive sequence process. The original spiral model approach was illustrated in Figure 5.2.

In the model of Agile Game Development implementation with Scrum is a conceptual framework normally focused on software management projects which include how to organize and coordinate the tasks that every function can be done

without impediments, under which it can use any other agile use case example. In this concept, the model would be more focused on the engineering of the knowledge management project where techniques are best to complete tasks efficiently and can help to reduce the cost of project. The four quadrants indicate recurring and structured steps of software project management activity for game based learning. The project management cycle implement with four tasks (determining objectives, identifying and resolving risk, development and testing, and planning the next iteration) that occur in every cycle of the project. Accordingly, the model fits in this thesis also requires the repeating and revising of the levels of methodology. However, in term of game development process, the field of game uses Game-Scrum which can combine these two methodologies as a basis, foe adapting them with the experience of professionals and focusing an people with little no experience in game development.

Our model approach represents each step of diagram above, from top to down showing the procedures of conducting the software engineering life cycle of game based learning for an outdoor classroom. Setup our concept was developed by using combination of software life cycles of the waterfall model , spiral and daily scrum in order to use spiral loop to improve the knowledge system from the knowledge expert and student's feedback. This feedback can used to help adjust the difficulty system to balance the game mechanics. Thus, it is not necessary to back to the requirement level of document analysis again, due to it only giving the concept of the game mechanics and the fact of data that it already has consensus from the team member with knowledge co-creation experience. On contrary, the design level contributes to the whole detail with minimum requirements from the stakeholder i.e. collaboration model and the knowledge model for the system.

In this part, the details of each function in each phase has been mentioned before. The result from the design level (specifications and requirements from the education propose) will be used to create and design more detail and integration of information technology for education, which would be more explained in more detail in Chapter 6. While the results of data from applying to our proposed methodology in game-based learning for outdoor classroom will be demonstrated in Chapter 6 too.

5.7 The proposed of application architecture

In this section, we provide the details of our proposal of game based learning for outdoor classroom architecture. Apart from game engines, there are further components which are typically included in the architectures for a Game-based learning game type. After which, we carefully look at typical examples of Game-based learning and describe the proposed architecture, focusing on the aspects of the physical and the development view. Each of the described exemplars illustrates typical facets. We divided our platform into two sides a backend and frontend which uses to comprise with mobile, Internet and PC. The computer backend side consists of the server architecture based on e-learning platform standard with uses the phpmyadmin to store the trail description and the gamification, but for the algorithms used map generation, we exchanged the data of GIS from the google map server by requirement the license key to exchange data by POST method with game development environments, which support multiple (target) platforms, to cope with the growth in technology. For the Frontend side we developed using unity game engine so we can alternate between mobiles and PCs and can play on multi devices e.g. tablets and other mobile devices. This section continues with a brief overview on the exemplary architectures for suitable platforms. It provides the detail in terms of the technical of application architecture specific to mobile sensors as architecture components and concludes with the common of components.

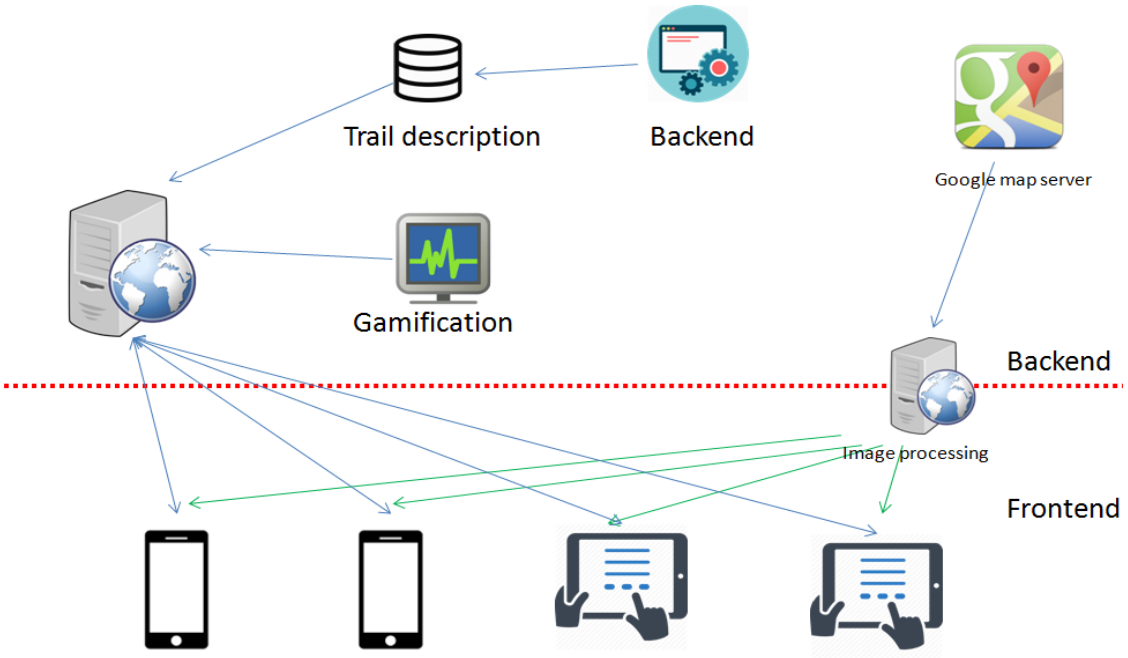

Figure 5.7 Software architecture

5.7.1 Backend of application architecture

The representational state of transfer REST architectural style imposes several design rules or constraints to ensure distinct architectural properties are achieved. These constraints are applied to the components, interactions, and data required by the software solution. The constraints for REST are:

1. Client-Server Model – requires support for on independent evolution of client and server logic. Clients send requests to the server in order to invoke a task. The task is either executed or rejected before a response is sent to the client.

2. Stateless – no state should be kept between communication between the client and server.

3. Cacheable – the server is able to mark messages as cacheable or non-cacheable so clients can reuse any responses.

4. Layered System – no layer can have knowledge of any layer past the directly connected layer.

5. Uniform Interface - all clients and services adhere to a uniform interface. When applying REST constraints to web service application programming interface (API), the HTTP based web services are considered RESTful. By utilizing RESTful

web services, several properties of the software are improved. These properties include: performance, scalability, simplicity, modifiability, visibility, portability, and reliability [Erl et al. 2014]. A RESTful web service consists of a service consumer and service hosted on a web application server. Clients utilize the uniform interface of HTTP to send requests that interact with a resource. The HTTP methods are described in Table 5.2.

Table 5.2 RESTful web service HTTP interface methods.

HTTP Method	Description
GET	Returns the resource location at the uniform resource identifier (URI).
POST	Creates a new resource. A URI is returned that represents the newly created resource.
PUT	Modify an existing resource.
DELETE	Deletes a resource at the provided URI.

In order to satisfy the objectives of this thesis, we need a server application that will enable a large number of mobile clients to send gameplay data for analysis. Additionally, this server application needs to be able to support the ability for players to log in and view their performance data in real-time. These functional requirements are the basis of our server application. Data statistics will be responsible for the following key requirements

1. Enable games to send gameplay data in a lightweight manner.

2. Analyze and generate visualizations of the gameplay data.

3. Provide a web site for players to view generated visualizations of their gameplay in real-time.

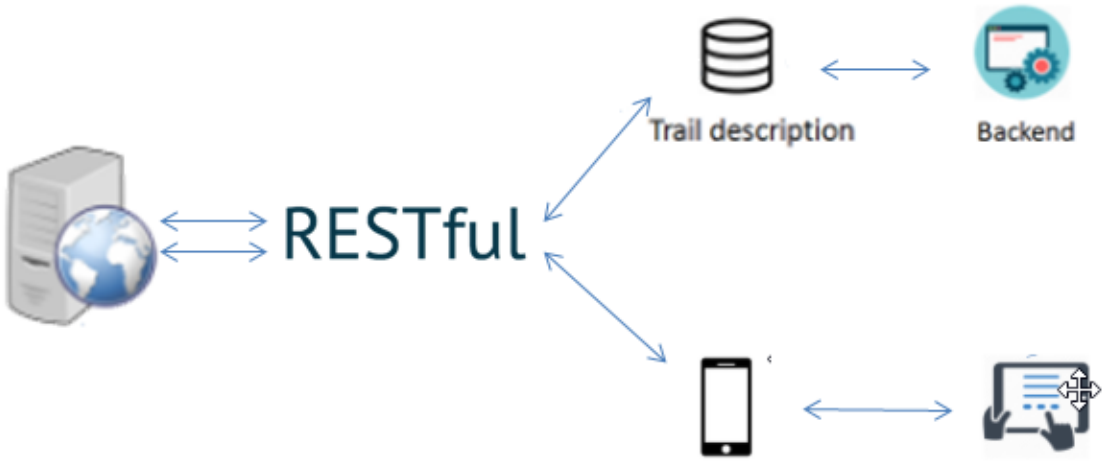

Figure 5.8 RESTful services connection with our proposed software architecture

By identifying these functional requirements of Data statistics, we evaluated various architectural styles against the following architectural requirements:

1. A restful asynchronous communication from a mobile device to a server application over the Internet.

2. A robust infrastructure that can handle many clients by sending large data sets simultaneously.

3. Since we do not know what data client games send, we need a means to store some type of unknown data structure.

Given these constraints, we chose to utilize a mixture of multi-tiered client-server architecture and a collection of RESTful web services. By using RESTful web services for the clients, the lightweight asynchronous communication requirement is satisfied. Light-weight communication allows statistics to be created in Data statistics using simple HTTP commands, reducing bandwidth and hardware resource requirements of the client. Additionally, multi-tiered will allow Data statistics to scale at each tier independently. Scaling each tier will enable a large number of players to send gameplay data to gather Data statistics. Lastly, we incorporated a Mysql database to store gameplay information to satisfy the requirement. Mysql databases enable the storage of schema-less information to be retrieved and stored. The game is responsible for defining the structure of the gameplay data bring transmitted to Data statistics, removing the need for modifying the server when the gameplay data's structure changes.

5.7.1.1 Trail description and backend

This Trail description module stores the high level state of the learning system, and it is responsible for its behavior. Two main aspects are under the control of the logical view:

• World dynamics of learning point: There may be a part of the state of the world that is not dependent on the exercise being executed, but is governed by some rules that always apply within that domain. The tutoring subsystem may want to access to that part of the state but will not change the dynamics of the world. For example, in a system that simulates the working of an engine there are certain rules that specify how the state of the engine evolves as inputs are received.

• Knowledge of learning points: this knowledge system will provide the information for learning in the virtual world. It has to be described from the each learning point subsystem, including video, image and sound of information for each point.

• Exercise dependent state: This is the part of the state in the virtual world whose dynamics are determined by the exercise that is being executed. It has to be manipulated from the each learning point subsystem, including the behavior of other avatars on the map as well as the monitoring of the student actions and decisions on how to interact with them and answer the question.

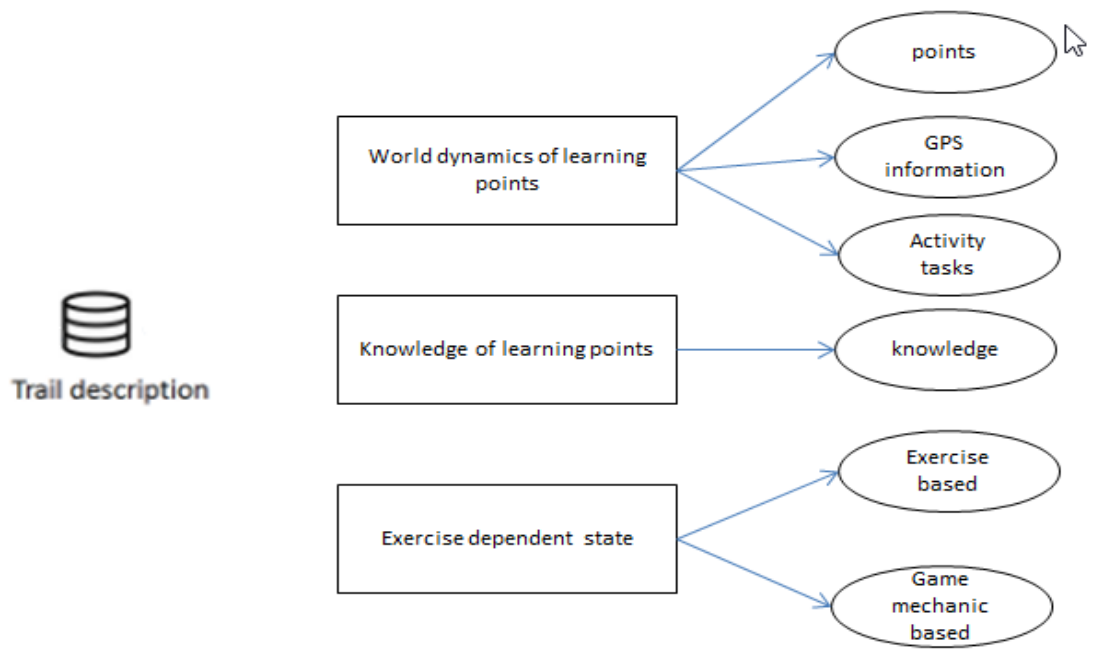

Figure 5.9 Model of Trail description

When the Game based learning loads a new world map, it informs this module that initializes the world dynamics and exercise dependent state accordingly with it. During the exercise resolution, the logical view receives the relevant events from the objects in the environment, and updates its state. It also receives actions from the communication module of the learning point subsystem. Some of them will require performing operations in the virtual world. In that case, the module translates them in primitive operations to be sent to the entities in the module of learning points.

5.7.1.2 Game based diagram

The functions of gameplay include the game flow approach and real-time identification map followed by usecase diagram

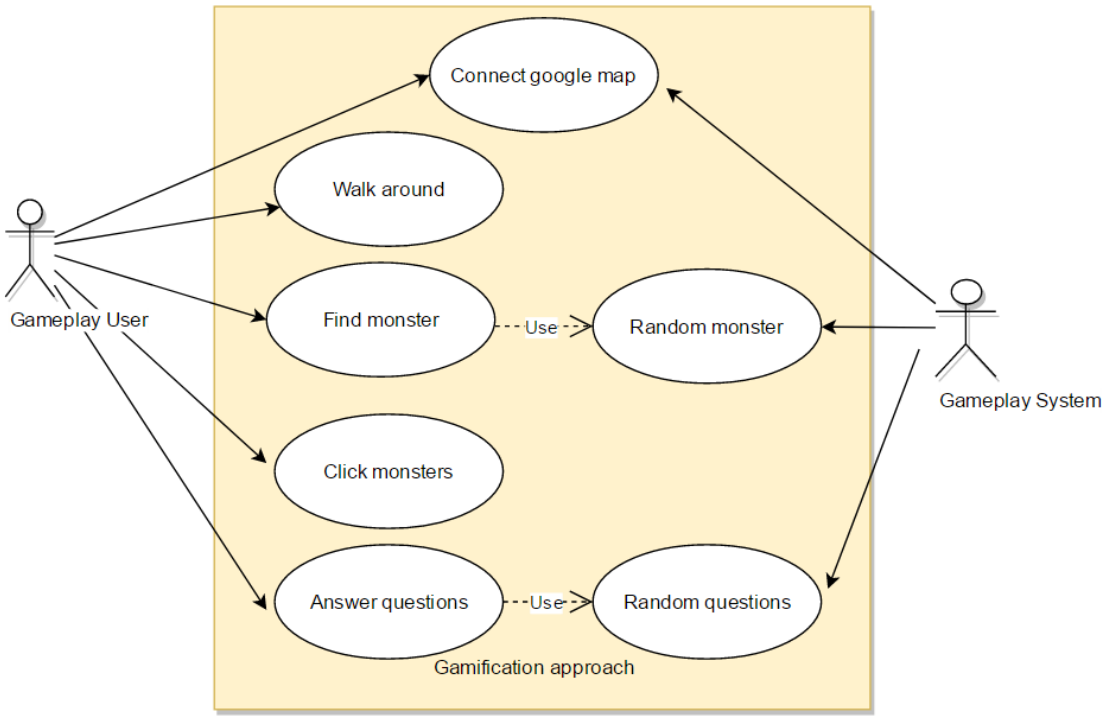

Figure 5.10 Usecase diagram

5.7.1.3 Google map server integrated with Mobile based learning

The Mobile based GPS unit calculates its location using standard latitude and longitude coordinates (lat-longs) and has a refresh rate of about one second. While it would be possible to make the game work in lat-long space, we decided to use a different representation that would make it easier to compute directions. To make the

programming easier, we converted the lat-longs to a street-centric representation. Instead of being a coordinate in an XY space, the player is represented as a position along a network of nodes and edges. Each node is like an intersection of streets, and each edge is a section of a street between two intersections. In other words, instead of saying that the player is at the system, we can say that the player is at Chiangmai zoo and Golden museum of Bangkok. Using a street centric approach makes it easier to measure distances, compute directions, and move virtual enemies for answering the question. The only challenge is that a GPS unit outputs a location as a latlong, and thus a conversion process must be used to transform the player's measured location into a usable representation. The simplest conversion is to snap the lat-long to the nearest street.

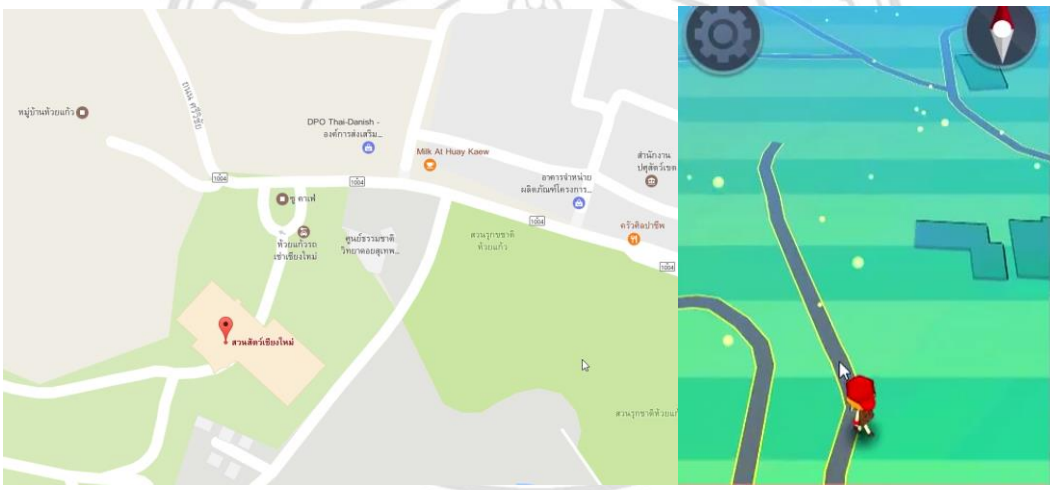

Figure 5.11 A map location can be represented in many different ways to our proposed system.

The system uses the lat-longs of all the street nodes (intersections) to calculate the distance from the player's lat-long to every street in the given network. That distance is the error between the street and the player's position. The street with the lowest calculated distance is likely to be the street that the player is currently on. Unfortunately, just relying on street distance can cause other problems if that distance is less than the average GPS error. If a player is running down a street and passes through an intersection, the program's system could get confused and think that the player turned at the intersection and started running down a side street. Due to GPS uncertainty

that side street may seem closer than the street the player is actually on. At first, we considered looking at old GPS positions to tell if the player was turning. Unfortunately this introduces a delay in the system, and we wanted the response time to be as quick as possible. Figure shows the algorithm assumes the player is following directions and is more responsive to street changes that are directed by the game's guidance system. To fix this, a penalty is given to all streets except the current street the player is on and the street onto which the guidance system is telling them to turn. For example, if the player is on Current Street, and the system has told the player to take a left on Next Street, then any street other than Current Street and Next Street gets a score penalty when the system is calculating the player's new location. As a result, the system is very responsive to a player doing the right thing, but it waits at least 20m before telling the player they turned down the wrong street.

Figure 5.12 A google map api convert to our map for Game based learning.

When the player starts a new mission, they chooses a desired running distance, and the app contacts a map server to get the necessary map data. The server uses the player's parameters to build a custom subset of the parameters that are based on the player's starting position, desired run length, and an optional end position. The map server works by performing a breadth-first search rooted at the player's current position, acquiring all the nodes that are within the player's desired run length. From this collection of nodes, all the corresponding streets are gathered. This process can create virtual dead ends if some of the nodes in a street are too far away. As a result, the virtual dead ends are iteratively removed from the subset map. The system also automatically increases the run length if the player's desired end position is too far away. All the streets and runs are limited at 20 km. Longer runs require more map data in the phone's memory, and we didn't want to risk low memory problems. Nodes are collected, the server outputs the data as a JSON string. This entire map subset calculation takes the server about a second to compute, and the phone needs about 5-10 seconds to download

the data over a 3G or 4G connection. Once the phone downloads the map, it converts the data from the compact "nodes and streets" a representation to a graph structure with nodes and edges. This structure consumes more memory, as each street gets broken down into multiple edges, but it is much more usable. The top level dictionary has keys for each node in the map. The value for any key is a smaller dictionary with keys for each neighboring node. This idea was inspired by Downey work in 2008.

5.7.2 Frontend

Client Side: on the client (game) side RestSharp, a REST and HTTP API client for unity frameworks, was used to access the server's side endpoints. The data was serialized using fastJSON to transform the C# data structures to and from JSON packets that communicated with the server's data endpoints.

Server Side: the server was built using Appserver, a Php Web framework and extension library, helped serve up the information and allowed us to restrict access to the APIs. All of these libraries are open source meaning any one can use at no cost.

Data Instrumentation: We instrumented the game to capture a variety of information ranging from user IDs, chat messages, player positions, and interactions with in-game tools e.g. the Basic Local Alignment Search Tool (described in greater detail in the next section). The Apache library was used to place data into a more structured format and record the data to a file. Php scripts help process the data into a more usable form, and these then were imported into the Unity3D Editor by using a custom extension. This extension rendered the character movements and actions.

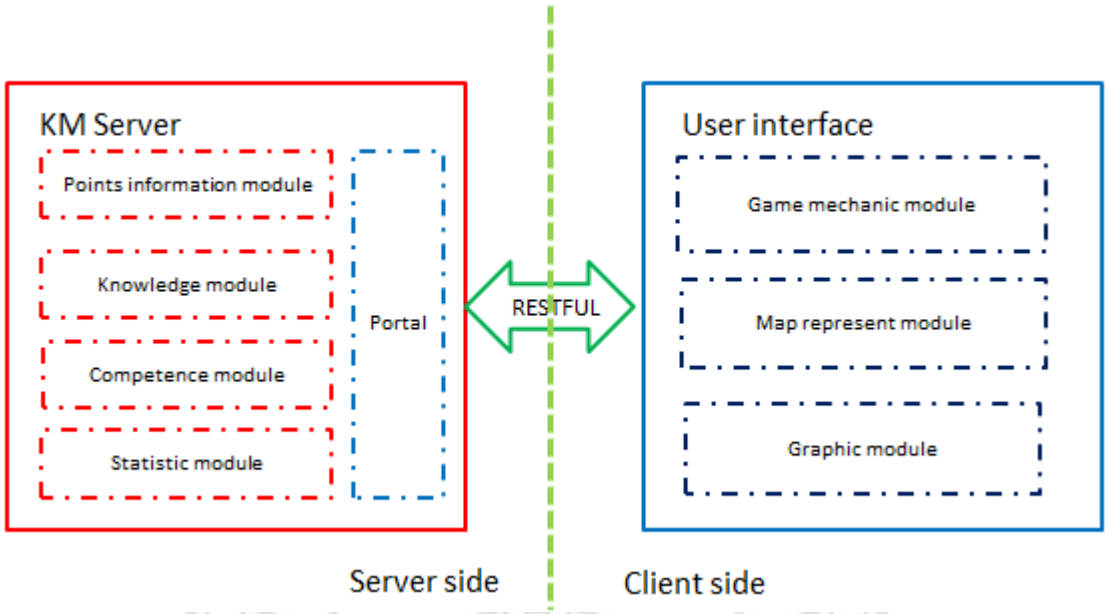

Figure 5.13 The Server side and Client side of Game based learning

5.7.2.1 Game application

When the game application is opened on a mobile device, it first checks for any new available game data. If it finds new data, it will install the game. If no new game data is found, the application will display a list of pre-installed games and let the user choose which of them are wished to play. The game application will then display a map, an icon or progression bar on the left side of the screen and a number that displays the distance to the next location in the upper right corner (see Figure 4.5). The number is decided by GPS, either internaly to the phone or an external GPS unit that is connected to the phone. The game application will also create all the locations, with a margin of error zone of about 30 meters around the GPS location. A margin of errors is necessary for several reasons:

- The locations are created through clicking on a map, this is easy to do and for the most accurate.
- GPS data is subjective to a number of error sources. This depends on the number of satellites available and clarity of the signal, determined by weather.
- The presence of several buildings can also load to an error source. The radius of the margin of error, however, has been the topic of discussion in several studies.

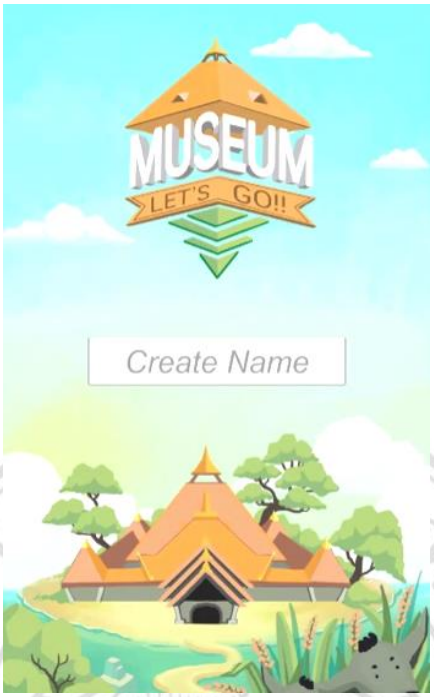

Figure 5.14 The title page of Game based learning application

The goal of Game based learning games is to find all locations and accumulate the most number of points. The time spent and the number of general knowledge information used includes to the total number of points gained in the game, and the player with the lowest score wins. Game based learning games can be played individually, played as anytime and anywhere. In the team work mode, players share single mobile device. One potential pedagogical advantage to organizing games that involved players working as a team is that sharing a phone promotes group discussion.

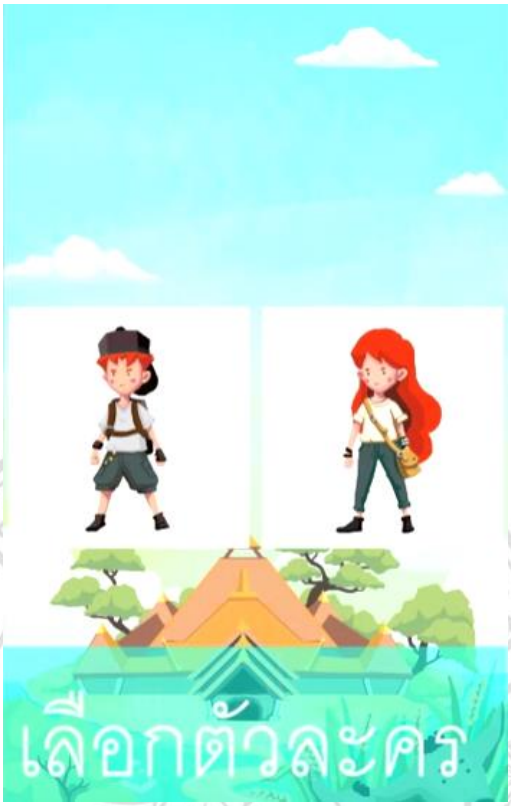

Figure 5.15 The select of characters of Game based learning application

When the game begins, the application will display a map, with the icons on the left side and the distance to the next location written in red numbers in the top right corner. The player can choose what gender they want to be. The numbers of point on whole map will change decrease, either if they are moving away from the next location or increase if moving towards the target location. When the mobile device is within 30 meters, the numbers change from red to green. The user can then access the monster for the location by using the menu option available at the bottom of the screen. The monster is then displayed as an overlay to the map. When the monster is clicked, the icon for the location that is displayed in the progression bar and the game pauses. Then a question will appear at the center of the screen. This will allow the team time to discuss with each other what to do next, without the fear of losing points in the game.

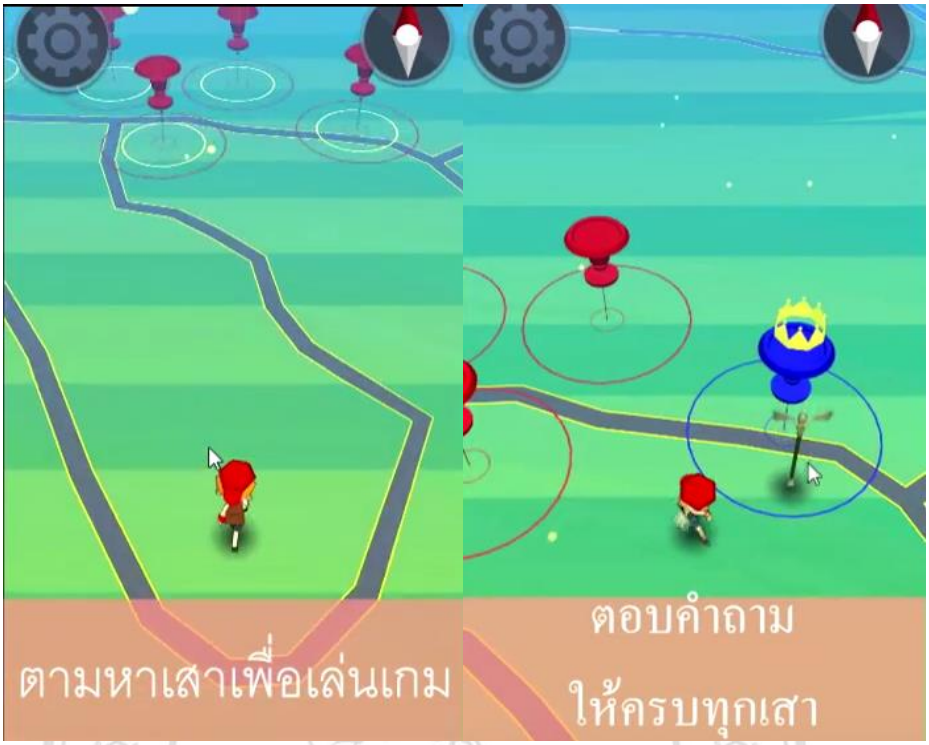

Figure 5.16 The game mechanic of Game based learning application

To help the players find locations, they can point out the tower of learning and use monitor mapping, whereby they can used to lead them in the right direction. The monitor map is available on the right side menu of gameplay, but is also for the Game based learning game and a total of number points for learning is available for each location. It was decided that the use of a map to plan to direct the points in order to gain a score. This was done to lead the group towards relying on discussion, it provided the hints system. Each hint used knowledge information to explain for that question but if the player spent time using with hinting too much it would deducts the number of points from the total score, and the game designer decide the number of points that would be added to the total score for each hint used. Hints can be accessed through the menu system that is located at the bottom of the screen.

5.7.2.2 Mobile interface

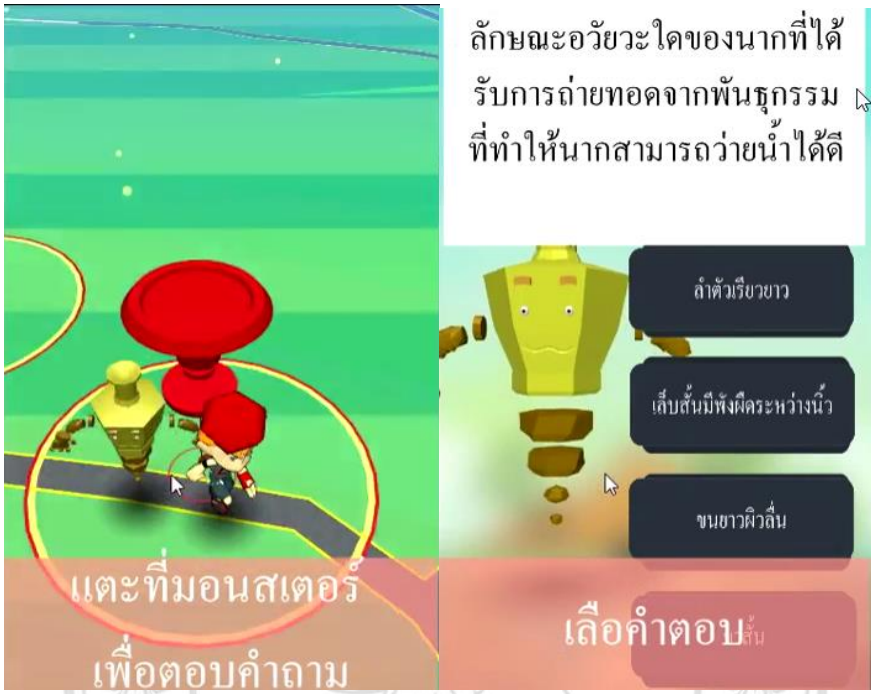

Figure 5.17 The game mechanic of answer-question of Game based learning

Figure 5.18 The concept art of monster for Game based learning .

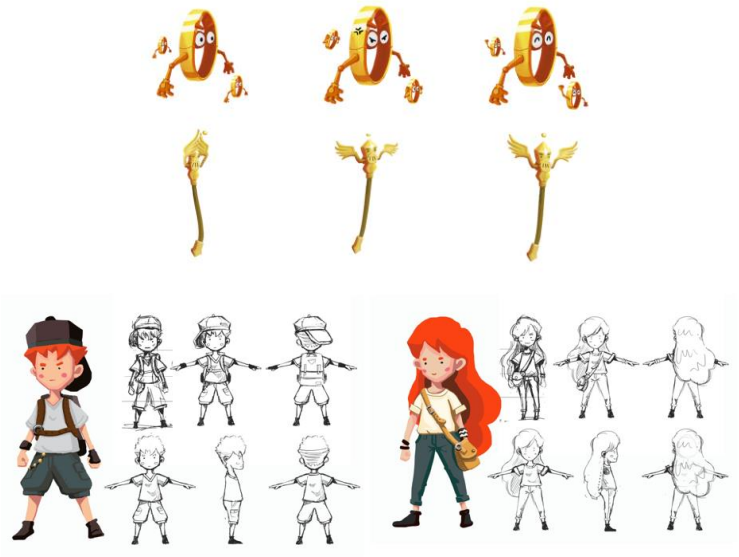

Figure 5.19 The concept art of monster and characters for Game based learning.

Figure 5.20 The monitor map of Game based learning in Chiangmai Zoo

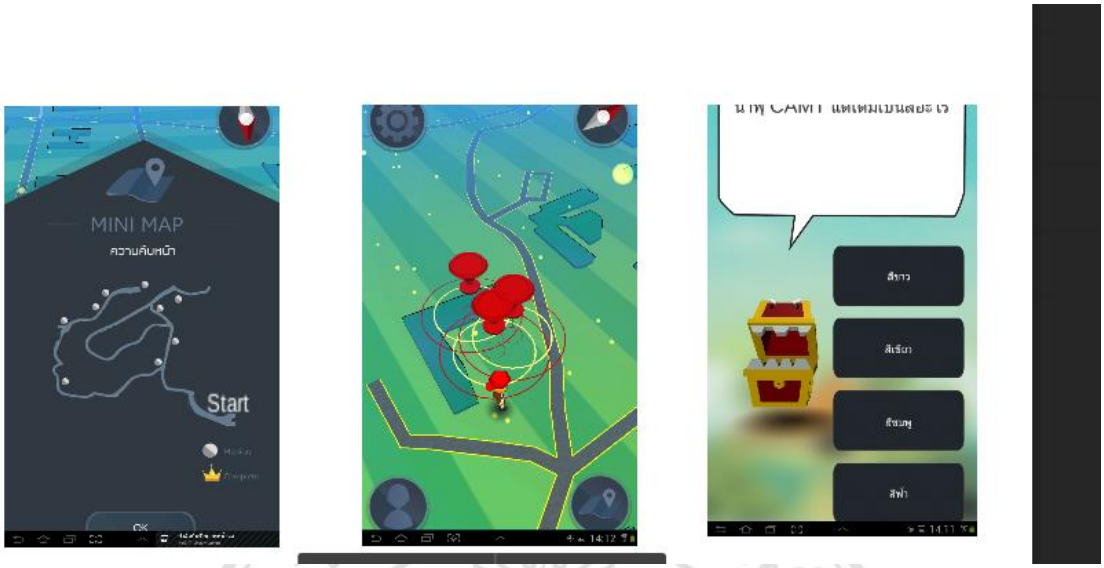

Figure 5.21 The overall of gameplay of Game based learning

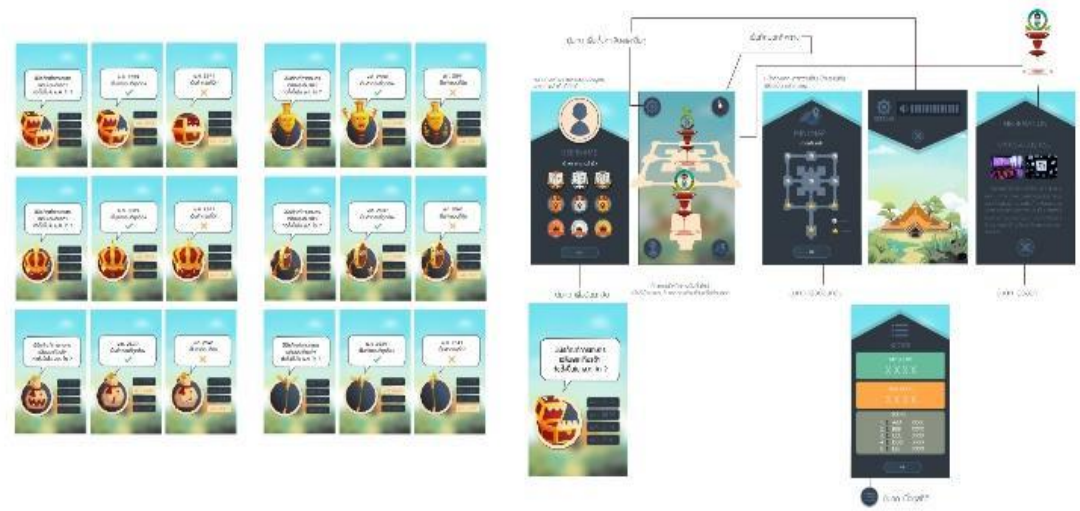

Figure 5.22 The overall of gameplay of Game based learning

5.8 Conclusion

In this chapter, it was proposed that the Game design document for outdoor learning and Prototype of learning game was based on requirement from Cholprathanpateak school and Chiangmai Zoo and consisted of three parts. From the previous chapter, the KA-document,KC-document and KM-document which were included from the traditional game design document based on Chris Taylor. Lastly, the implementation was based on the GDD with name Museum GO. The platform was into

two sides: backend and Frontend which was used to be comprised with the mobile. The side of backend consists of the server architecture based on e-learning platform standard with use to the phpmyadmin to store the trail description and the gamification. As for the algorithms used to generate maps, the exchanging of the data of GIS from the Google map server the need of the license key to exchange data by POST method with game development environments, that support multiple (target) platforms. In the following chapter, the game would be tested following the research framework.