WEBCODER MOBILE APPLICATION FOR LEARNING WEB PROGRAMMING FOR PRIMARY SCHOOL CHILDREN

By

Nurul Aida binti Azman Shah
16420

Dissertation submitted in partial fulfilment
of the requirements for the
Bachelor of Technology (Hons)
Business Information System

FYP 2 MAY 2015

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Webcoder Mobile Application for Learning Web Programming for Primary School Children

By

Nurul Aida binti Azman Shah

A project dissertation to the
Information System Programme
Universiti Teknologi PETRONAS
In partial fulfillment of the requirement for the
BACHELOR OF TECHNOLOGY (Hons)
(Business Information System)

Approved by,

_____

(Dr Ahmad Sobri Bin Hashim)
UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK
July 2015

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

_____

NURUL AIDA BINTI AZMAN SHAH

ABSTRACT

WebCoder mobile application is to be used on smartphones with Android operating systems. This mobile application is aimed to teach children range from 8 to 12 years old to learn HTML codes. The problems identified behind this project are difficulties faced by tertiary education students in learning programming and to fulfill the important need of programming in today's world. This project is aimed to design an instructional design model for teaching web programming to school children, to develop and android-based application that comprises the designed instructional design model and to evaluate the project in terms of user acceptance. The development of this project is using ADDIE lifecycle which is an approach to Instructional Design Model consisting analysis, design, development, implementation and evaluation. On top of that, this project has developed an instructional design model that covers the relevant elements for a teaching application. The development is done by using Phonegap cross platform tool. The flow of the application is designed and presented in a flowchart, and the graphical user interface (GUI) is designed according to suitability. Functionality test carried out shows a good result which means this project has nearly met success. From the User acceptance testing (UAT) performed, most of the feedbacks reflect good results in all three aspects which are performance and presentation of the application, as well as its effectiveness towards users. These results show positive feedbacks from potential end users. With a total of 28 weeks of development; starting from selection of project topic up to the testing and implementation, it is believed that the usage of mobile phones will be more beneficial especially among children, in terms of educational purposes. It will also put Malaysia on the same level with other developed countries in terms of education where programming is taught in early ages.

ACKNOWLEDGEMENT

First and foremost, I would like to express my deepest gratitude to Allah who gave me the strength to face all of the challenges and problems throughout the final year journey until I managed to complete this project.

I would like to offer my sincere gratitude to my supervisor, Dr Ahmad Sobri Bin Hashim who has guided and assisted me throughout my Final Year Project with patience and knowledge whilst allowing me to work in my own way. Although with a tight schedule as a lecturer and coordinator of Final Year Project course, he still offers his valuable time to assist me in completing the whole project.

Besides, special acknowledgement to the Computer Information Sciences (CIS) department for managing and give us, the students opportunity to carry out this individual project as well as to enhance our knowledge and skills.

Last but not least, I would like to thank my parents, family members and friends who always give me moral support and advices along the completion of my final year project. Without their support, I could not make it this far in completing the whole final year project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND OF STUDY

As observed today, mobile phones are important device that people bring along everywhere, throughout their day. Even in their sleep, people are still using mobile phones for alarm so much so alarm clocks have lost their significance. This attachment with smart phones is not only seen with adults, but also with children. Society pointed out the disadvantages of this condition, especially for children (Johnson & Kristonis, 2007). Therefore, this project focuses on developing a learning mobile application for children, to turn that negative idea into a positive one.

Programming is writing specific language into a machine e.g. computers and smart phones to ensure they run a specific program according to instructions. There are many languages of programming including C++, C#, Java, Python, et cetera, which most of them needs to be compiled on their own platform. Generally in Malaysia, students start learning programming in their tertiary education. Only students majoring in IT fields will be learning programming in their syllabus. However, initiatively, there are many students learn programming on their own, using online sources. In Malaysia, children are not exposed to programming languages in early ages. During primary and secondary school, students in Malaysian public schools are only taught about the hardware, computer security, basis of software and networking; leaving programming on the surface (Ahmad Tazali, 2011). Nevertheless, in developed countries, children are exposed to programming earlier through certain programs in school to introduce them. As early as kindergarten ages, they have tried basic programming codes which are helpful for familiarization.

Teaching programming requires the utilization of ICT hardware like computers, keyboard and CPU. However, with the smart phones phenomenon today, it would be a great opportunity to apply programming learning through smart phones (Buchegger, n.d.). Instead of children wasting their time with useless games, they will be replaced with educational mobile apps. The advantages of learning programming with mobile application are including its convenience to the learner and also the teacher; in terms of mobility. Children will also be more attracted and interested to learn because they always thought that mobile phones contain all the best games they want to play. Plus, monitoring and evaluation of students' performance will be easier with mobile phones.

WebCoder is a project that will be developed to help children to learn HTML web programming language through a mobile app. This project is designed as an after-school activity because the modules are less formal. The activities are created similar to games to attract children. So, children will play games while learning. HTML language is chosen because it is one of the basic languages to be learnt in programming. Plus, the syntax is simple and does not require a compiler to be executed. The development of this mobile application will help reducing the concerns of parents who worry how addicted are their children with games. This project will promote education in a fun way.

1.2 PROBLEM STATEMENT

The project is analyzed based on the problems identified which are stated below.

1.2.1 Lack of Instructional Design Model Study for Teaching Programming to School Children

Programming teaching is usually done in class or through online websites. In class, a student must be present in order to learn that subject and they will be able to run the programming codes when they do the tutorial. This method is usually done in formal teaching in schools or universities. Through website, there would be notes and tutorials that show the users to do certain programs for users to try. As far as this research goes, there is lack of instructional design model found for teaching programming to children.

1.2.2 Programming Language is Difficult to Learn

Programming subjects are difficult to learn and to score, especially when the learning process starts late, i.e. in tertiary education. In Malaysia, IT students often complain on how difficult it is writing programs as their syllabus only touches the basics. Ala-Mutka (2004) mentioned in her study at Tampere University of Technology, Finland, learning to program is generally considered difficult and programming courses often have high dropout rates. It has even been said, that it takes about 10 years for a novice to become an expert programmer. The difficulties they face lower their interest on programming subjects. This is why this project is meant for children, to create interest and curiosity to them, hence to increase the understanding about programming as they learn through.

Early start will ensure children to familiarize this subject to be learnt in the future. A research done by University of Wollongong found that early start will encourage the active participation of children and adults in a range of learning experiences (Brown & Lysaght, 2012). Plus, when children are familiar with programming codes, when

they are learning in class during tertiary education, the teacher will not face difficulties to explain the basis of programming because the students have understood them. Thus, it is believed early start to programming will be beneficial to the learner and also the teacher.

1.2.3 Lack of Awareness on the Importance of Programming among Students

Today, it is undeniable that technology has taken a big role in the world's development. When talking about technology, the common thoughts that will appear to mind are computers and smart phones. However, behind those running devices, there are millions lines of codes that enable those machines to work as what human desires. These lines of codes are written by programmers, which the job vacancies are plenty around the world (Weinberger, 2015).

Developed countries like the United States of America and Germany have already started programming teaching to children as early as primary school level. However, in research done for this project, there is no finding that programming is taught to primary level education. It is believed that if programming subjects are listed as important as Mathematics and Science, the children will be able to apply critical thinking when learning codes (Robins, Rountree & Rountree, 2003). Apart of that, learning programming will empower children's skills on problem solving and arithmetic. Plus, they are exciting because children will be able to see the outcome of their codes.

1.3 OBJECTIVE

The objectives of this project include:

1. To design instructional design model for teaching programming to school children.
2. To develop android-based application called WebCoder that incorporates the designed instructional design model.
3. To evaluate the application in terms of user acceptance.

1.4 PROJECT SCOPE

To fulfill the project objectives and ensure the project is achievable all the necessary requirements has been identified. Definition of the project scope is important to complete this project within the time frame. The scopes of this project are broken down into 3 items:

i. Platform: Running on Android platform
ii. Target user: Children range from 10 to 14 years old
iii. Content: Modules on chapters needed to be learnt for HTML web programming and activities to enhance the users' understanding on subject matter. The chapters will be guided by the Instructional Design Model created, according to suitability of users.

1.5 RELEVANCY OF PROJECT

The development of this mobile application will contribute to Malaysian community in terms of education and the importance of programming in the industry. The rise of awareness will help to reduce the high number of overseas outsourcing as well as external labor. Therefore, WebCoder is relevant to be developed as it will help to create awareness for target users on importance of Information, Communication and Technology (ICT).

1.6 FEASIBILITY OF PROJECT WITHIN THE SCOPE AND TIME FRAME

Prior to the initial phase of the study, the feasibility study is required in order to ensure the research is possible and smooth the development stage of this study. Feasibility study is separated in 5 categories as below:

Technical Feasibility

From the author technical point of view, this study is feasible. The technical part of this study begins in the prototypes cycle stage which consists of 3 sub-stages such as develop, demonstrate and refine where this study is expected to developed a mobile application for target users who owned vehicles and any workshops in Malaysia. It consists of two major functions which help the target users to learn programming. Technically, this mobile application is possible to develop as the author has the programming knowledge and learn from the internet.

Economic Feasibility

It is expected that the development cost for this mobile application will be low. As the author already equip with good hardware and software, no direct material cost will be incur and keep the cost for the development low.

Organizational Feasibility

In the pre-research, the study shows that there is some weakness the author identified on the user's side in determines the right lubricant for specific type of vehicles. With the increasing trend of using smart phone throughout the world including to children, the author would expect that this study and development of this mobile application will be supported by the target users.

Scope Feasibility

The area of research covers in Malaysia, but due to the time constraints on data collection, this study is focused on Seri Iskandar, Perak. Besides, the focus on the research is the target users' behavior in using smart phones in their ample time. In order to gather this information, the author has conducted a survey and observes target users from randomly from 8 to 10 years old.

Time Feasibility

The feasibility of my project within the time frame by focusing on the requirement planning and prototyping as well as to start at least develops a preliminary interface. Further with development phase, the development of this mobile application will be using web-based language through Phone Gap. With the month provided, it is expected every activities can be done according to the timeline.

CHAPTER 2

LITERATURE REVIEW

2.1 PROGRAMMING IN MALAYSIA

Programming in Malaysia is commonly taught in universities and pre-universities, after secondary school completed. Programming is not considered an important subject for every student to understand well although today we are living in a world full of Information Technology (IT). IT has become one of the basic needs in life; therefore it is essential for the society to at least know the use of one simple line of code.

However, the new curriculum structure of Malaysia education has implemented basic IT subjects to be taught in primary school. However, this subject is not further brought to secondary school unless the students selected IT as major subjects. One more shortage in IT education system in Malaysia, not many teachers is IT experts, and many are not even computer users. They are only assigned to as one of the standards of procedure. According to Asan (2003), many teachers lacked a functional computer literacy foundation upon which to build new technology and skills. Analysis of teachers' knowledge of computer technologies revealed low levels of technical knowledge, as well as some interesting perceptions of the role of some specific computer-related items. For most teachers, the use of computers and related technologies had not been a routine part of their own educational environment.

2.2 COMPARISON OF PROGRAMMING LESSONS IN OTHER COUNTRIES

Different countries have different ways in including Computer Sciences subjects in their education system. From a study conducted by Jones (2011), he discovered that there is increasing clarity that "Computer Science" means a lot more than "Learn to program in Java or C++". Programming is central to computing, but the underlying principles of algorithms, data structures, and computational thinking skills are both more fundamental and more durable. That is why, the idea to start teaching programming from early age is beneficial because as the children grow up and start focusing on the certain subject, and their time will not be wasted to start learning programming from scratch. They can already focus on more crucial Computer Science subjects. Jones also found that Scotland, India, South Korea and Greece had started formal computing education in secondary schools.

Department of Information and Computer Education of National Taiwan Normal University in Taipei, Taiwan had carried out a study on Teaching Computer Programming in Elementary Schools that reveals that children can learn and they enjoy learning computer programming and most parents also support teaching of computer programming in elementary schools. It is proven that programming is not as inaccessible and incomprehensible to children as many educators think it is (Lin, Yen, Yang, & Chen, 2005). The results of the research indicate that children can learn and they enjoy learning computer programming. Most parents also express positive opinions about their children's programming learning experience.

2.3 TEACHING PROGRAMMING TO CHILDREN

This idea to teach computer programming to children has been questioned for a long time; a study carried out in 1975 has been done by Weyer and Cannara. It was not a new idea that children should learn how to program a computer, so that they too might access its unparalleled power as a tool for thinking (Weyer & Camara, 1975).

The olden studies believe that computer programming potentially construct interactive environment to enhance children's interest and learning in mathematics. Department of Information and Computer Education of National Taiwan Normal University in Taipei, Taiwan had carried out a study on Teaching Computer Programming in Elementary Schools. The outcomes of this study reveal that children can learn and they enjoy learning computer programming and most parents also support teaching of computer programming in elementary schools.

This idea arises because as observed in universities in Malaysia, even Computer Sciences students face difficulties to learn programming at first. The reasons stated by the students are because they have no basic skills in programming, and only started learning programming at the age of at least 18, which is an age to master a subject, not to learn new things. Ala-Mutka (2004) mentioned in her study at Tampere University of Technology, Finland, learning to program is generally considered difficult and programming courses often have high dropout rates. It has even been said, that it takes about 10 years for a novice to become an expert programmer. An online research from University of Wollongong found that early start will encourage the active participation of children and adults in a range of learning experiences (Brown & Lysaght, 2012). This will increase the probability of success in individuals and the whole community.

It is agreed that computer programming delivers a number of benefits, so it is a great idea to develop programming learning method as mobile apps. Whether to apply programming learning at school, included in curriculum syllabus or as an afterschool program, this idea is believed to assist teachers and parents to help their children to learn programming. Web-based educational modules are plenty on the net, but observing today's environment where mobility is important, a mobile application has to be developed for children. These days, even children own their own personal tablets or mobile phones. The ever-changing digital age provides guidance for early childhood educators about the use of technology and interactive media in ways that can optimize opportunities for young children's cognitive, social, emotional, physical, and linguistic development. In this position statement, the definition of

technology tools encompasses a broad range of digital devices such as computers, tablets, multitouch screens, interactive whiteboards, mobile devices, cameras, DVD and music players, audio recorders, electronic toys, games, e-book readers, and older analog devices still being used such as tape recorders, VCRs, VHS tapes, record and cassette players, light tables, projectors, and microscopes (Radich, 2012). Therefore, developing an educational and interactive application would help them from being too preoccupied with the neglecting games.

According to (Kahn, 1999), some children, when introduced to something new and complex, will get excited and explore because they enjoy exploration and are good at it. Others are much more timid and will explore only if coached or guided. Others ask for instructions and follow them meticulously. Some children will carefully watch a demonstration, while others are impatient to try things themselves. In his study, Kahn found out that many educators today question the value of teaching programming to children. It is hard and there are now so many other things children can do with computers. Furthermore, programming can be a very empowering and creative experience. Children who can program can turn computers into electronic games, simulators, art or music generators, databases, animations, robot controllers, and the literally millions of other things that professional programmers have turned computers into.

Kahn also listed out a number of techniques which are suitable to be implemented to teach children computer programming. These include safe self-revealing environments, puzzle sequences as tutorials, pictorial instructions and other ways to learn, specifically traditional methods like lectures and homework assignments. This project is emphasizing the self-revealing environments. A self-revealing environment is designed so that an inquisitive explorer can discover what objects exist and how they behave. Exploratory learning is best supported by an environment that is safe and self-revealing. An environment is safe to explore if novice actions will not cause any permanent damage. It is very difficult to make a completely self-revealing environment. A self-revealing environment should be incremental, and the environment may feel open-ended and rich but is

designed so that certain objects or actions can be discovered only after others have been mastered. This helps reduce confusion and frustration that often results from the initial explorations of a rich and complex environment.

There are many individuals attempting to make learning programming as simple and attractive as possible. On 2004, Yakov Fain wrote a book, “Java Programming for Kids, Parents and Grandparents” where complete tutorial of Java programming is written completely and directly. Briggs wrote “Python for Kids” on 2013, to teach HTML from the installing part to the program samples. Previously on 2007, Briggs also has written “Snake Wrangling for Kids, Learning to Program with Python” but the book is wordier and too descriptive, more suitable for older students.

2.4 APPROACHES OF ONLINE TEACHING

Today, online learning is the most accessible pathway to the new knowledge economy and related jobs for the majority of working people. To be effective for the next generation, online learning has to include mobile learning, e-gaming, online communities, and learning management systems that engage each user (Anderson, 2008). In the same book, the co-author Ally defines online learning as the use of the Internet to access learning materials; to interact with the content, instructor, and other learners; and to obtain support during the learning process, in order to acquire knowledge, to construct personal meaning, and to grow from the learning experience.

Ally listed out the strategies to attention for online learning:

- Important information should be placed in the center of the screen for reading, and learners must be able to read from left to right.
- Information critical for learning should be highlighted to focus learners’ attention.
- Learners should be told why they should take the lesson, so that they can attend to the information throughout the lesson.

- The difficulty level of the material must match the cognitive level of the learner, so that the learner can both attend to and relate to the material. Links to both simpler and more complicated materials can be used to accommodate learners at different knowledge levels.

Online learning should include a variety of learning activities to help students achieve the lesson's learning outcome and to cater for their individual needs (Anderson, 2008). Examples of learning activities for children include viewing visuals or video materials. Appropriate application exercises should be embedded throughout the online lesson to establish the relevance of the materials. Practice activities, with feedback, should be included to allow learners to monitor how they are performing, so that they know their ability and performance.

Teaching and learning with mobile technologies is beginning to make a breakthrough from small-scale pilots to large departmental and institutional implementations. This section presents both key issues for educators and technical developers, and research informed guidelines as to how these can be addressed. Compared to desktop technology, learning and teaching with mobile technology presents significant new challenges including the context, mobility and informality (University of Birmingham, n.d.).

As this project is developed for mobile devices, it must follow certain guidelines to make sure the subjects are received by the user. Furthermore, this application will be used by primary school children thus it requires suitable approaches to teach them a new and complex subject, web programming. These includes the arrangement of content must be in bottom-up approach, where they will be taught in smaller pieces, in detail, before collecting those pieces into one working program.

2.5 PROGRAMMING LANGUAGES

The top programming languages available like Java, Ruby, Python, C/C++ and JavaScript are among the essential languages need to be mastered by IT students. However, there are a few more programming languages made for simplicity and convenience. For example, Scratch was released on 2007, developed by the Lifelong Kindergarten group at the MIT Media Lab (Chiang, n.d.). Here is how Scratch website describes Scratch.

"Scratch is a new programming language that makes it easy to create your own interactive stories, animations, games, music, and art -- and share your creations on the web."

The current version of Scratch is intended for students ages 8-16. According to Resnick et al. (2009) where the development put high priority on:

1. Diversity – supporting many different types of projects (stories, games, animations, simulations), so that people with widely varying interests can all work on projects that they care deeply about.
2. Personalization – making it easy for people to personalize their Scratch projects by importing photos and music clips, recording voices, creating graphics.

2.6 HTML PROGRAMMING

HTML is HyperText Markup Language that is used to design documents for the World Wide Web. Author's project is developed for HTML learning. Compared to Scratch, it is intentionally created for kids, but this language will not be used for real computer programs. HTML is used in widely so it is believed that it would give more advantages to the learner. HTML language does not even need a compiler to be executed so the results of the written codes can be seen as soon as the codes are ready.

WebCoder aims to prove that the usage of smart phones by children can also bring educational advantages. The advantages are including increasing awareness of parents and society on how important is teaching programming language to children, as it gives benefits to the children and also to the nation in the future. It will also make programming as a compulsory subject to be learnt by everyone because of its significance to today's world. Learning programming can increase children's skills in terms of their critical thinking and arithmetic. Based on the researches, I believe that this mobile application project is achievable because the content of the learning syllabus is based on reliable resources and interactive module will be included for suitability of the age group selected.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 RESEARCH METHODOLOGY

Research methodology involves the process in collecting information and data. This process is done for the purpose of decision making. The methodology may include publication research, interviews, surveys and other research techniques. In IT projects, project development also involves System Development Life Cycle (SDLC). Figure 1 below shows the main common SDLC.

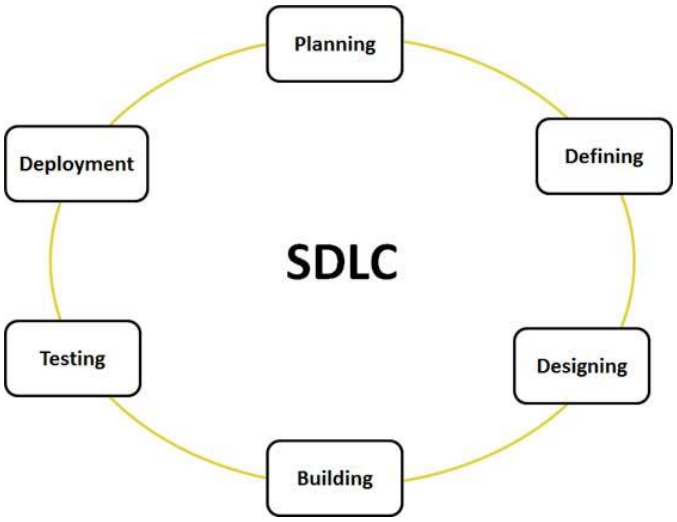


FIGURE 1. System Development Life Cycle

There are a number of SDLC models including Waterfall Model, Iterative Model, Spiral Model, V-Model and Big Bang Model. In this project, the author is using ADDIE lifecycle. ADDIE model is used because it provides a method for a good decision making in order to determine the who, what, when, where, why, and how of

a project. The concept of this approach is to obtain an overall view of project development. The benefits of using this approach are the end product is more likely to meet the project requirements. This approach also helps to ensure that no development activity is started before a necessary pre-activity is completed. ADDIE contains five phases which are Analyse, Design, Develop, Implement and Evaluate. Each phase is characterized by a set of activities and a project output in the form of a tangible deliverable. The deliverable for one phase is the input for the next. Diagram below shows the basic sequence of ADDIE based on Florida State University.

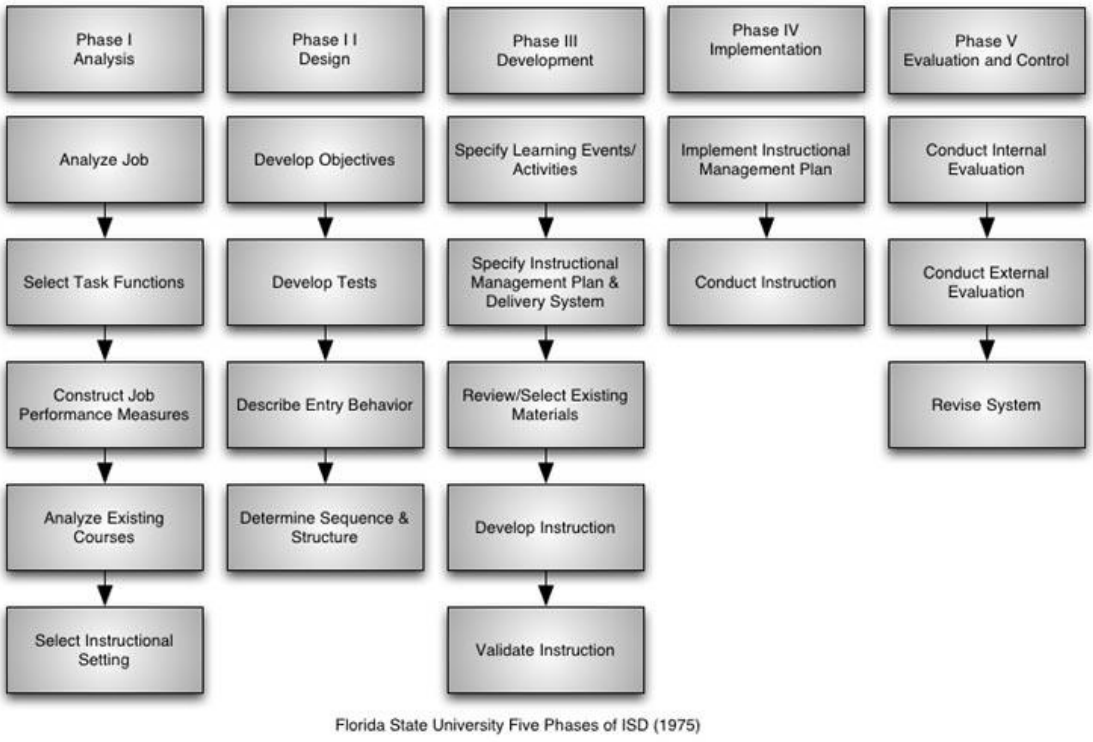


FIGURE 2. Addie diagram

Figure 2 shows a linear process of ADDIE lifecycle. However, ADDIE is not like the common waterfall model, where each step must be depending on the step before as pre-requisite. ADDIE lifecycle is more dynamic where the steps can be carried out according to relevancy. Thus, time constraint is not a barrier to develop the project in a short time using this ADDIE approach. Details of ADDIE approach in this project are described as follows.

3.1.1 Analysis Phase

Based on Figure 4 on Analysis Phase, for the input, a target user is identified which are school children range from 8 to 12 years old. Based on the problem statements found, which mostly regarding programming learning, Opportunities of mobile phone usage are also included as the input because target group is suitable. After analysis phase, the outputs are content specification which are made suitable for target group, syllabus which are obtained from various sources, learning tools which are modules and activities in the application and other application requirements.

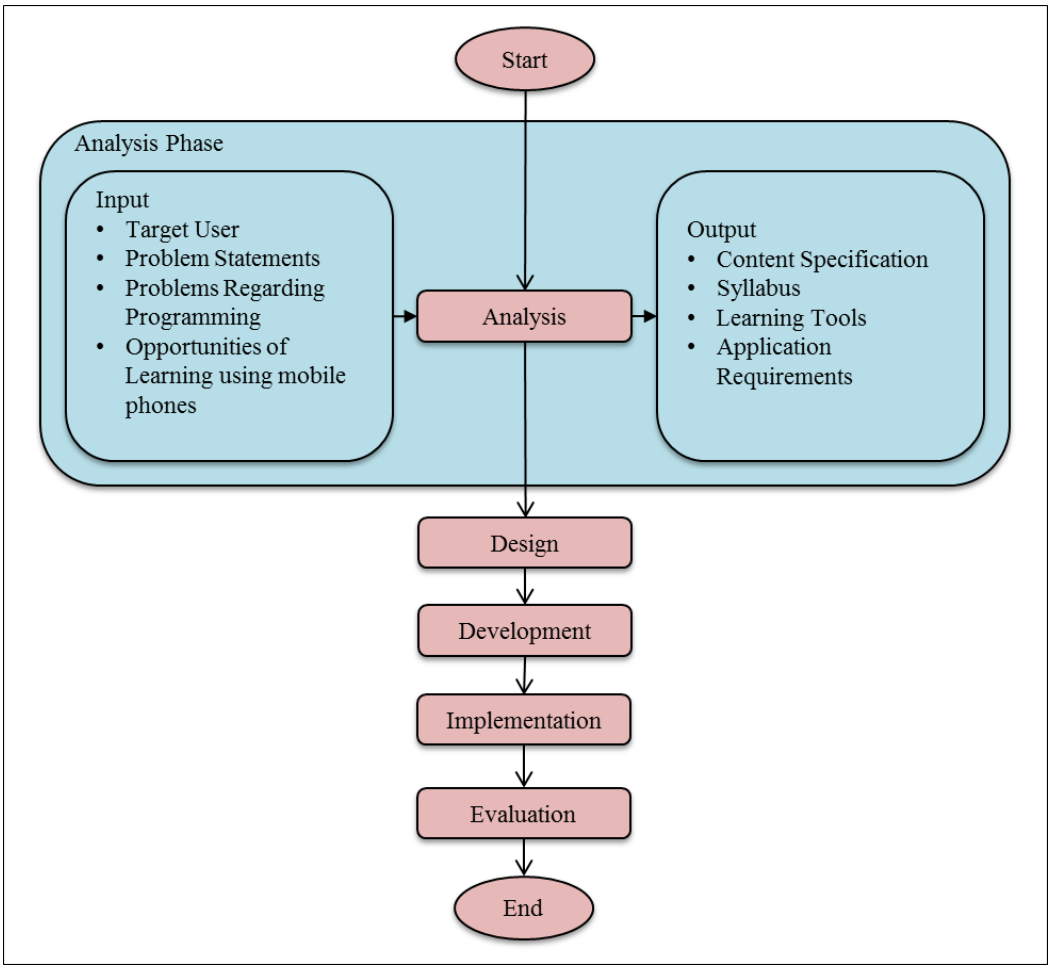


FIGURE 3. Analysis Phase

3.1.2 Design Phase

In Figure 5, Design Phase is illustrated. From the subject content, the separate designs of content, structure, interface and prototype is set as input and after designing phase, the whole application design application and flowchart are drawn.

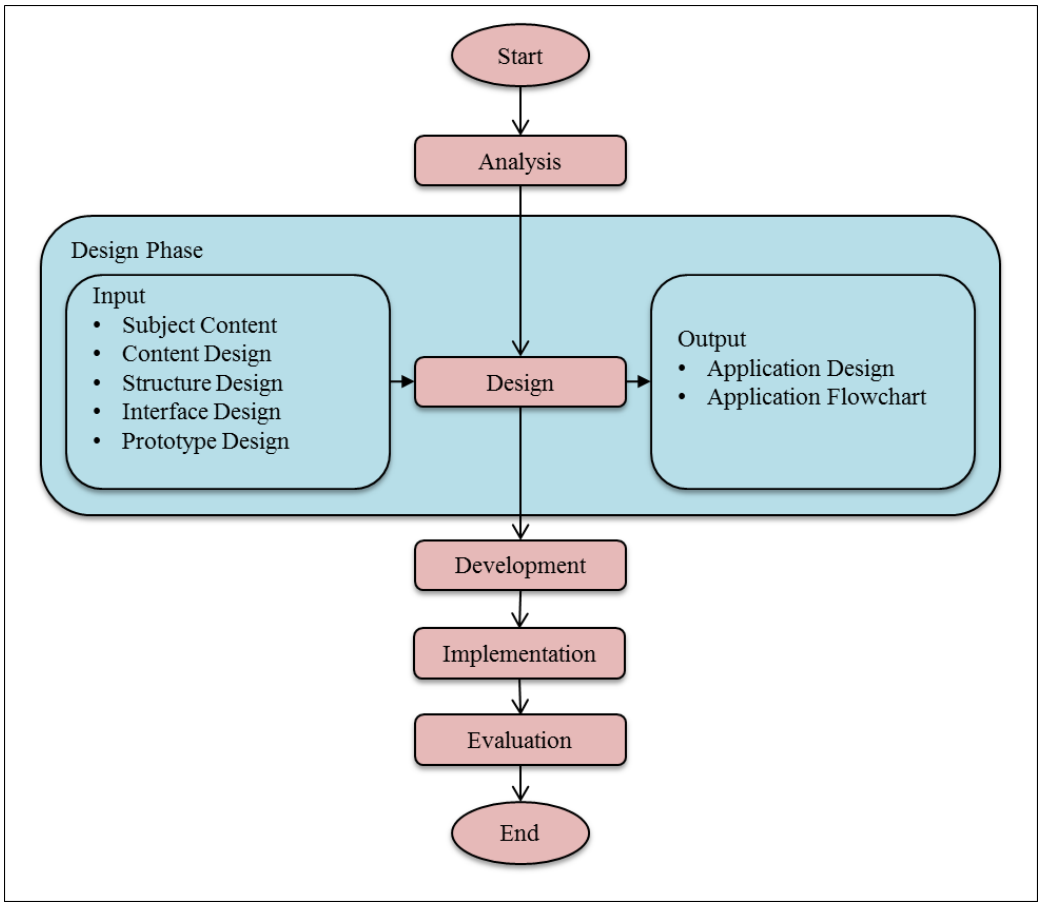


FIGURE 4. Design Phase

3.1.3 Development Phase

Figure 6 below shows the development phase in ADDIE lifecycle of this project. The inputs of development phase are flowchart and application designs which are obtained from design phase. Development is done using App Inventor, a tool to develop mobile application using an online platform. The output would be the application itself.

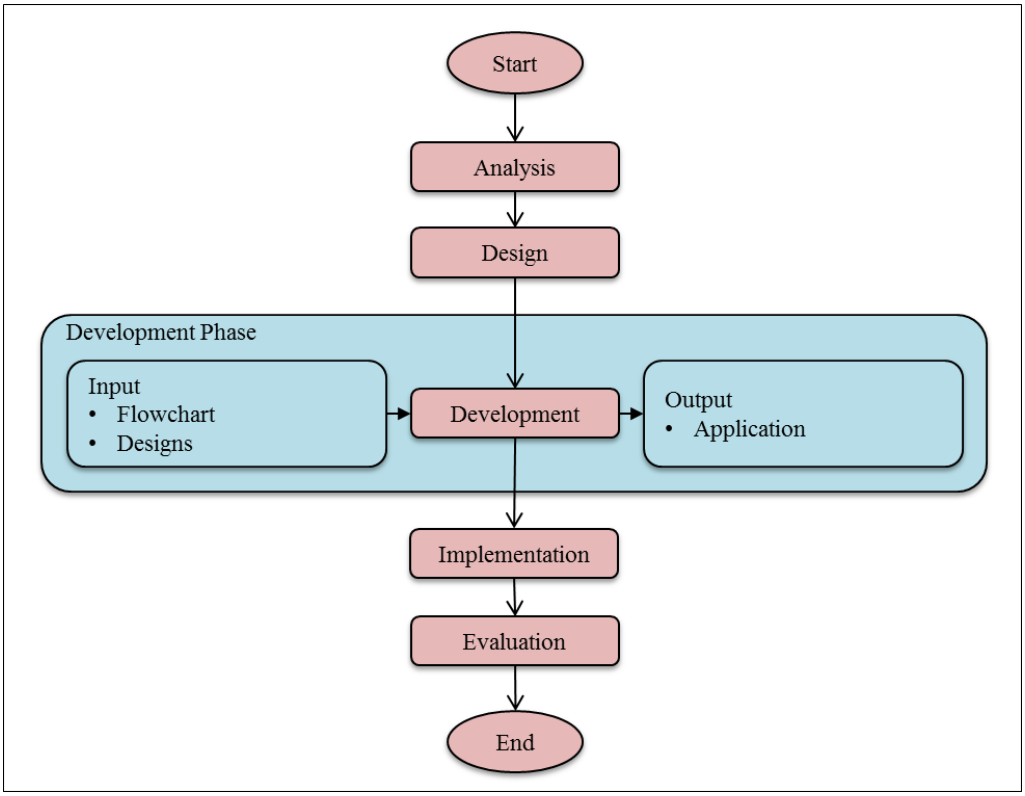


FIGURE 5. Development Phase

3.1.4 Implementation Phase

Implementation phase of this project is demonstrated in Figure 7 below. The developed application will become the input in this phase. This application will be uploaded into Google Play Store as this application is running on Android platform. After that, usability data is obtained as the output and this will allow improvements to be made according to the data.

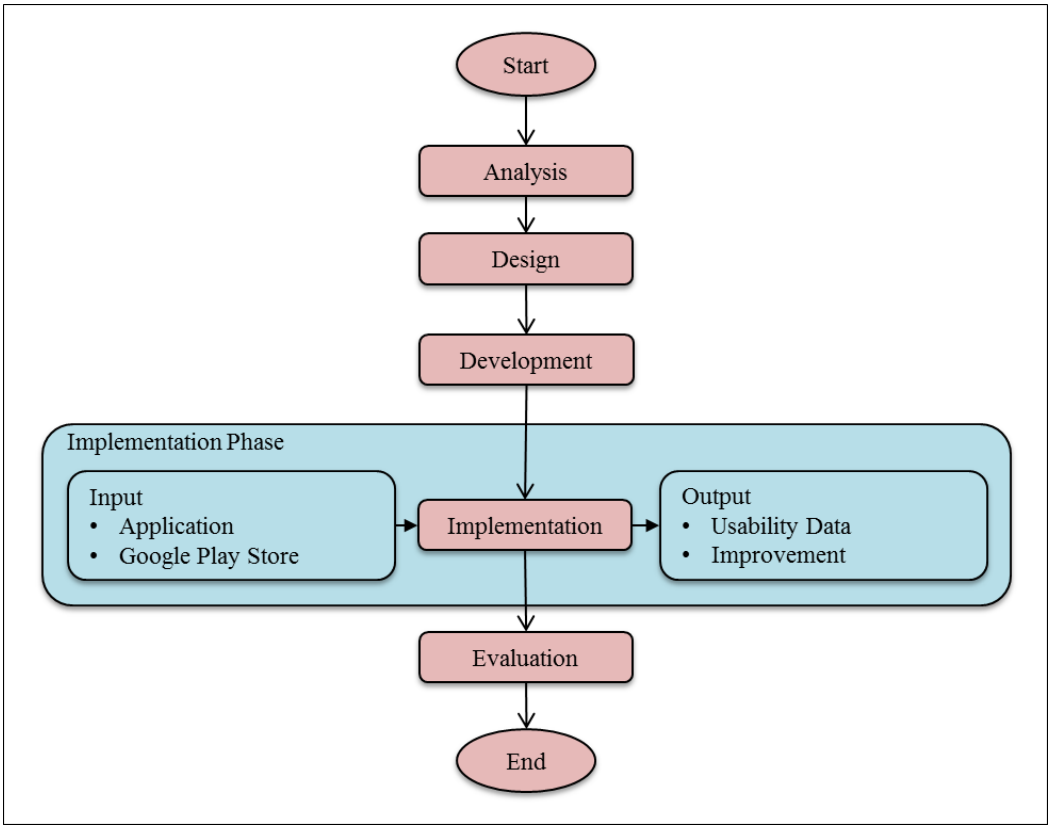


FIGURE 6. Implementation Phase

3.1.5 Evaluation Phase

As shown in Figure 8, user acceptance testing will be carried out to make sure target users can learn a new subject using a new way. User acceptance testing is done to make sure users can use the application easily. This stage of testing will produce output for usability and effectiveness of the project.

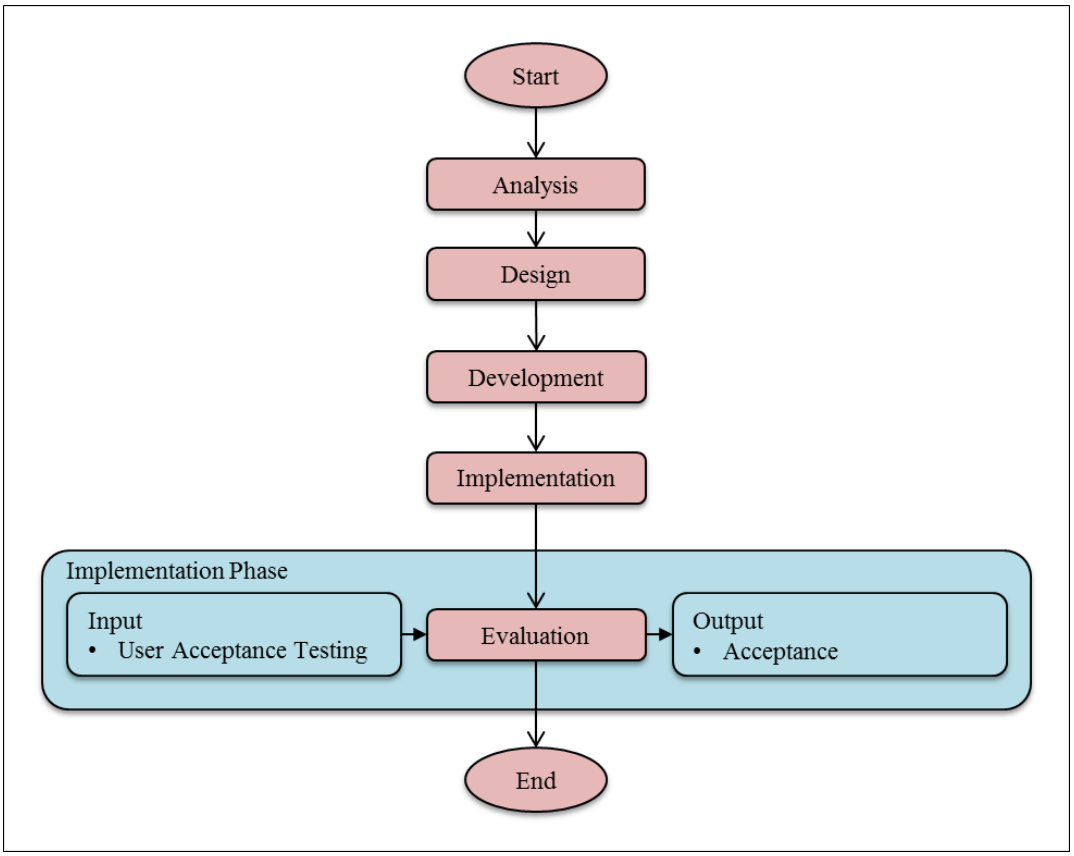


FIGURE 7. Implementation Phase

3.2 SYSTEM ARCHITECTURE

Systems Architecture is the structural and conceptual model of the system developed, that includes the connection among the components of the systems. In this project, the components include the database, the graphical user interface and the network. Diagram below shows the project's system architecture.

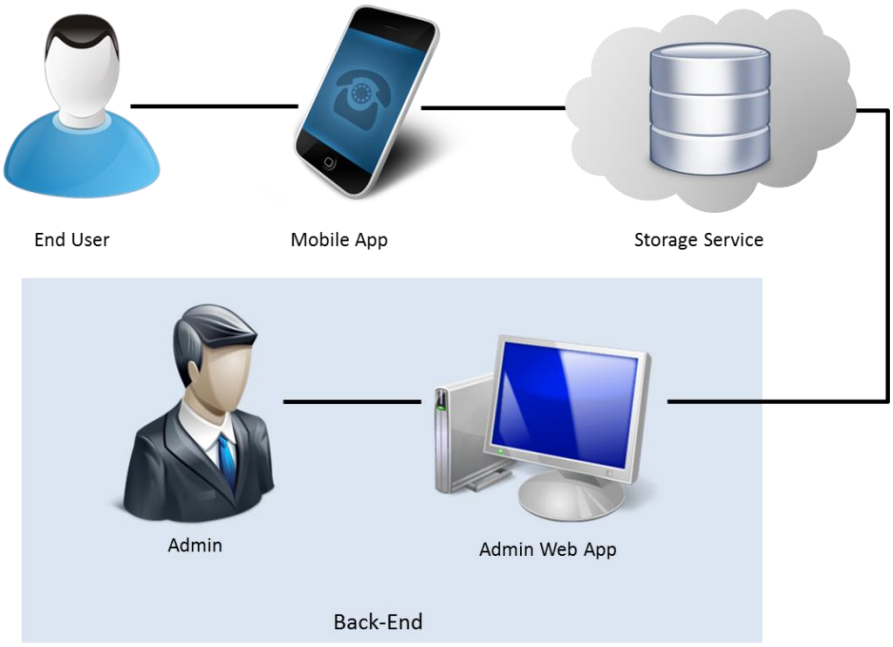


FIGURE 8. System Architecture

3.3 GANTT CHART AND KEY MILESTONES

TABLE 1. Gantt Chart and Key Milestone

Task	Week																											
	FYP1														FYP2													
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Requirement Planning																												
Selection of Project Topic		☐																										
Identification of Problem			☐																									
Define Objectives of Project																												
Study on Project Background																												
Preliminary Research Work																												
Literature Review						☐																						
Prototyping																												
Designing Architecture																												
Designing Flowchart																												
Designing ADDIE Diagram																												
Construct Flowchart																												
Construct ADDIE Diagram																												
Tools and Equipment																												
Submission of Draft Interim Report																												
Submission of Interim Report											☐																	

Prototype Development																												
Proposal Defence																												
Developing User Interface																												
Coding Mobile Application																												
Testing																												
User Acceptance Test																												
Improvement of Prototype																												
Implementation																												
Viva																												
Project Submission																												

Gantt chart Key Milestones

3.4 TOOLS

Tools needed include:

a) Hardware
   - Smartphone or tablet running on Android Operation System platform
   - PC with Windows OS

b) Software
   - Microsoft Office
   - Phonegap installed in PC and smartphone to connect with the built application

c) Development tools
   - Phonegap build: used to build mobile applications for mobile devices using JavaScript, HTML5 and CSS3

d) Development tools
   - HTML
   - CSS
   - JavaScript

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 INSTRUCTIONAL DESIGN MODEL

The development of the project is implemented using ADDIE lifecycle, as mentioned in Chapter 3, which the output is represented in instructional design model (IDM). The IDM involves five elements that work as the reference for the development phase. The elements are objective, source, perpetual navigation, learning approach, pedagogical approach and interactivity.

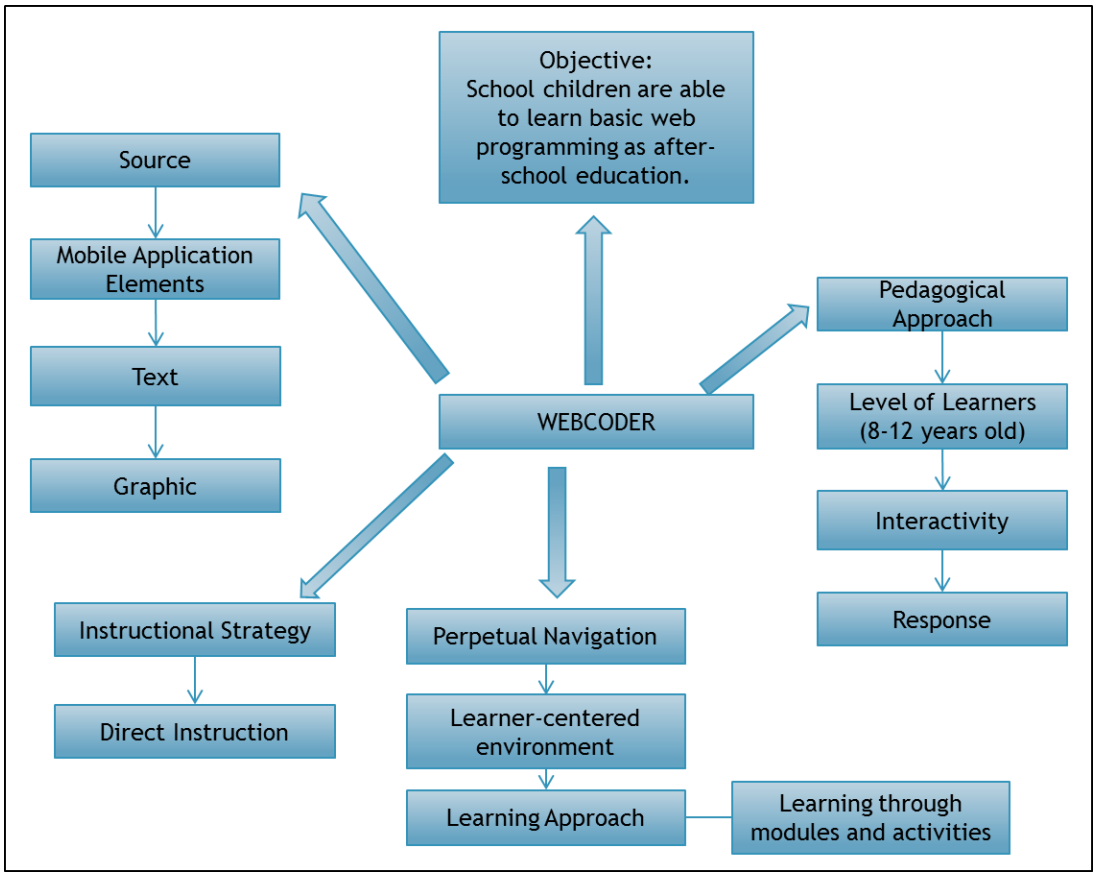


FIGURE 9. Instructional Design Model

Based on Figure 9, the IDM of this project consists of five elements. First, the objective is to allow school children to learn programming language not as part of school syllabus but as an alternative to after-school activities that uses mobile phone as the medium. Secondly, the source of this project is from mobile application elements. Mobile application pages are not as big as desktop-size applications. Therefore, there are limitations for mobile applications as well as the source or learning medium is also limited. There are only two learning sources which are text and graphics. Texts are used to give the explanation regarding the topic while graphics are mostly shown in the activity part. Next, the instructional strategy for this project is direct instruction which is received by the users when they start doing the activity option. This is shown in the activity page, where the instructions are given to make the users understand what they can do with the application.

Next, perpetual navigation means continuous navigation experienced by the user. This element need learner-centered environment for their effective environment when they are learning programming. In activity option, the difficulties of questions are made accordingly so that the users can adapt. After each topic in the tutorial, the users are allowed to play an activity to enhance their comprehension on the subject. The learning approach is as mentioned earlier. There are two approaches namely tutorials; to explain on the subject matter and activities; to exercise users' understanding. Finally, the pedagogical approach which is defined as the best approach to teaching; consists of level of learners which has been mentioned previously; 8-12 years old, interactivity which can be shown with keyboard and touch screen to the application and the response; how users respond to the instructions given.

After completed gathering all the required information, the author has identified the requirement for the application. All the requirements are presented in the table below:

TABLE 2. Functionality of Application

Function	Description
Log in	Allows user to provide their username and password for unique identity when they are using the application.
Welcoming bird	To greet individual user based on the unique username that they used.
Tabs	To separate contents of notes according to topics and difficulty level.
Tutorials	To show the content of syllabus, shown in comic strips and colourful graphics.
Exercises	To let the users play activities according to the topics they had learned to increase their comprehension.
Hint	Initially set hidden, but users are allowed to view. To show users the hint to answer the activities.
Answer checking	Results of the exercise appear when submit button is clicked. To let the users know if the answers they provide are correct.

4.2 FLOW OF APPLICATION

Diagram below shows the flowchart of the mobile application from the start to the end.

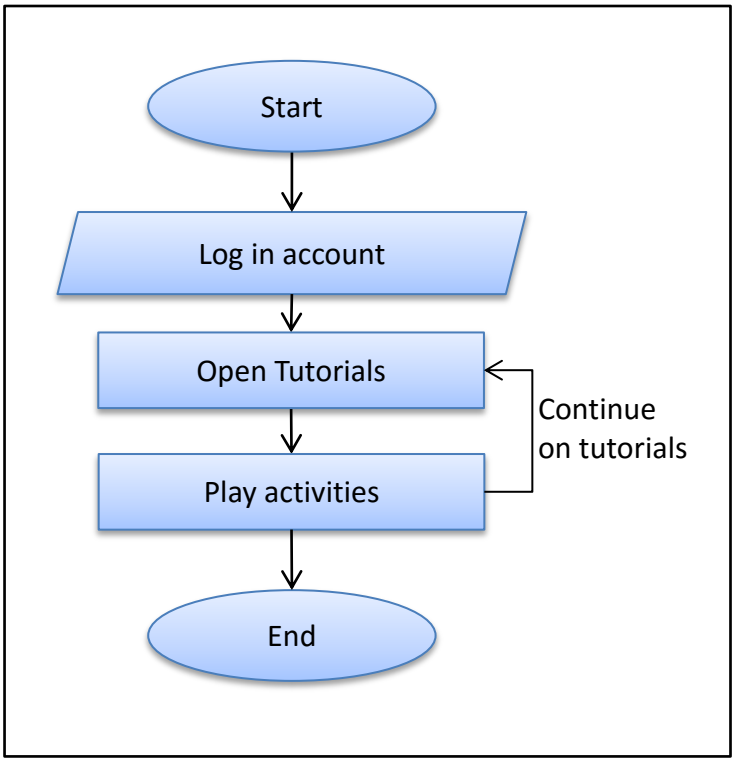


FIGURE 10. Flow of Application

Figure 10 shows the flowchart of the application. The flow of application is designed in a simple way to maintain target users' usability. Based on the diagram above, as soon as the user starts the program, they are required to log into their account. Then, the home screen will appear which allows them to proceed to learn in the tutorials page. As soon as the finished, they are allowed to play activities to increase their understanding in the subject matter. Next, the users can take another tutorial page and move on to different topics available.

4.3 USER INTERFACE

Diagrams below show the user interface of the prototype.

FIGURE 11. Sign in Page

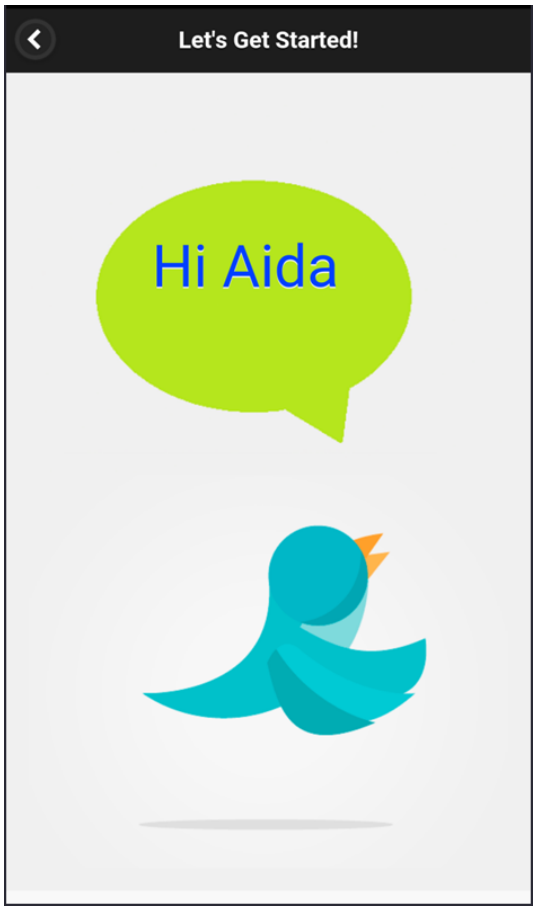


FIGURE 12. Start Page

Figure 11 above shows the Sign in Page, the first page that will appear once a user first open the application. As usual, before clicking the login button, user is required to fill in username and password. As they logged in, user's name will appear to greet and welcome the users. User is required to touch anywhere on the screen to proceed to the next page that contains the main functionality. As seen on diagrams above, the interface assists the children to learn with the use of attractive colours, graphics and animations.

FIGURE 13. Tutorials Page

After user log into their account, they will be brought to the tutorials page, which can be referred at Figure 13. Based on the figure, graphics and colours play an important role as an aid to learning. Tutorial are the place where modules on the subjects are displayed to the users. Users are allowed to choose which chapter that they want to learn that are represented by difficulty levels; one, two or three. Each chapter will provide users with the explanation, syntax and example of that particular subject. Users can enhance their understanding when they started doing exercises in Activities functionality.

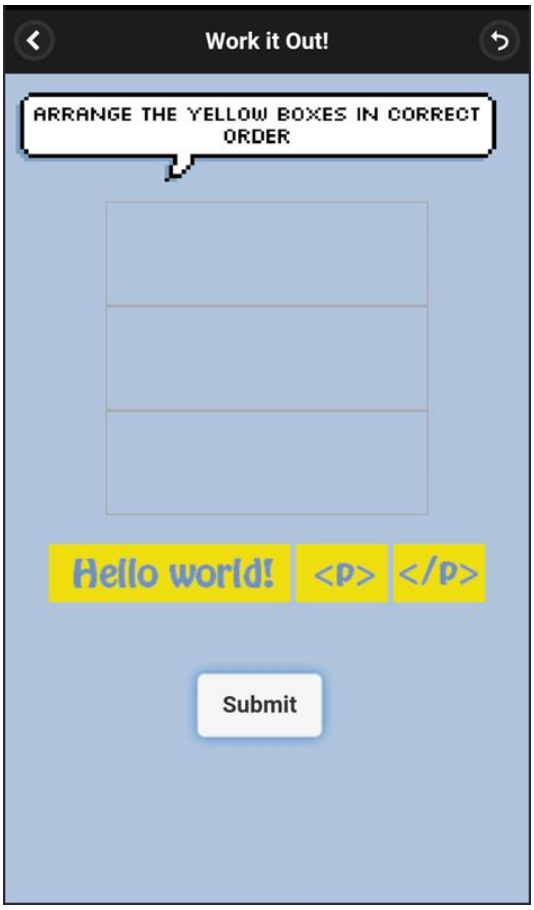


FIGURE 14. Activities Page

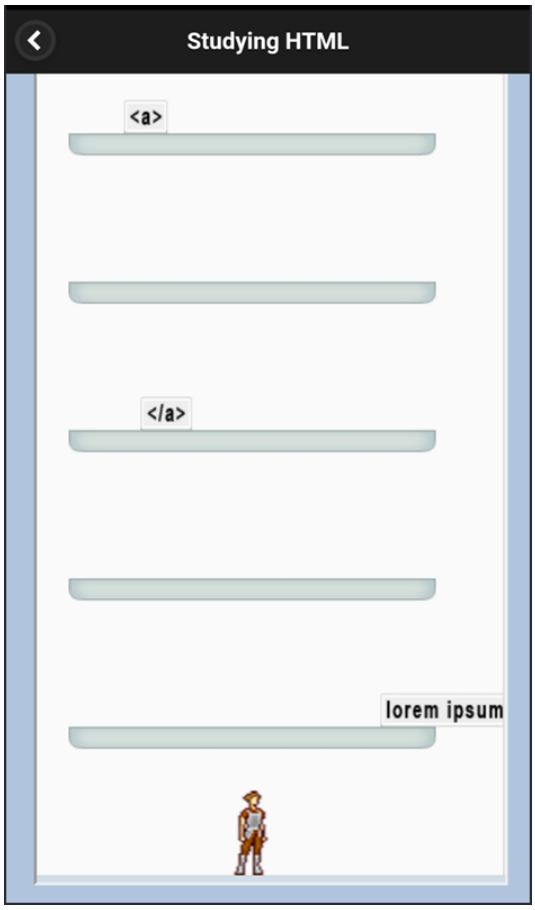


FIGURE 15. Activities Page
(Embedded from external webpage)

As shown in Figure 14, this is one of the pages from activities. The instruction is given directly at the top of the page and the objects in the page are made clearly visible. Meanwhile, Figure 15 shows another page from activities. This page is a graphic game embedded from an external website which the instructions appear before the real games appear.

4.4 FUNCTIONALITY TESTING

The purpose of this testing is to check the functionalities of this mobile application based on the requirement. Table below shows the data of each functions of WebCoder based on testing carried out to 5 candidates who are selected from mobile phone users who are familiar with mobile applications and IT-savvy.

TABLE 3. Functionality Testing Results

Functions	Expected Outcome	Testing Frequency	Testing Result		Remark
			Success	Failure	
Start-up					
"Login" button	Requires user to fill in their username and password before starting the application	5	5	0	
Start page	Application greets the user and displays the username	5	4	1	Javascript responds in different speed in different device
"Click me"	Requires user to touch anywhere on the screen to proceed to the activities page	5	5	0	
Tutorials					
Tabs	Navigates user to different topics of notes	5	4	1	Javascript responds in different speed in different device
Dialog buttons	Navigates user to exercise pages according to the topics	5	5	0	
Exercises					
Drag and drop (Exercise 1)	Allows user to drag the boxes provided to arrange them in correct order	5	3	2	Different devices have limitations on the Javascript
"Hint" button	Shows hidden hint to the user	5	5	0	
Click tags (Exercise 2)	Words from each button will pop up in the arranging box	5	4	1	Different devices have limitations on the Javascript
"Submit" button	Check the results and alert the users	5	3	2	

4.5 USER ACCEPTANCE TESTING

User Acceptance Testing (UAT) is a key feature of project to implement new systems, mobile applications or process. In line with the purpose of UAT which is to ensure that the new systems, mobile applications or process does actually meet the essential user requirement, UAT will be test by our target users for evaluation of WebCoder Mobile Application. 15 people who range from 8 to 12 years are selected randomly. Targeted users will be provided with WebCoder Mobile Application setup on developer's Samsung Galaxy Note 3 to test this application. After the testing, users are required to fill in the questionnaire (shown in Appendix 1-A) guided by a guardian to make sure the children understand the requirements they need to respond. The detailed results are shown in Appendix 1-B. The main features that are being tested by the target users of mobile application as well as the summary of results are shown below.

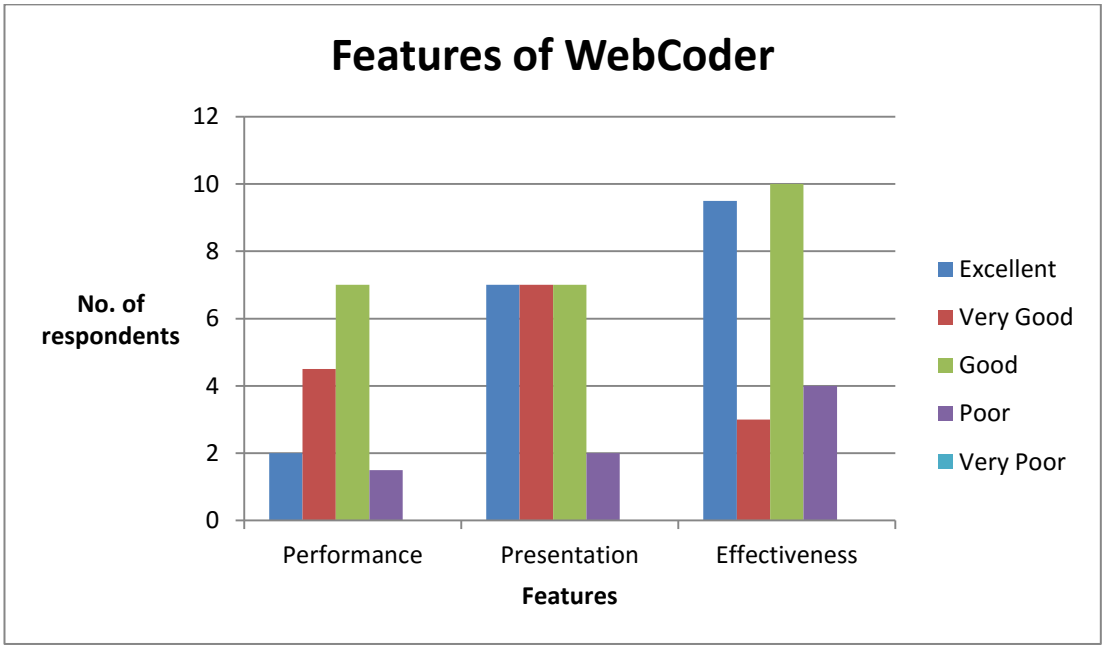


FIGURE 16. Graph of UAT Results

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 CONCLUSION

This project aims to design instructional design model for teaching programming to school children. Thus, this project is to develop android-based application called WebCoder that comprises the designed instructional design model. When the project is done, the application is evaluated for user acceptance. So far, this project has achieved one objective which is developing instructional design model for teaching programming to school children. The interfaces for WebCoder have been designed based on the instructional design that has been developed

WebCoder is designed for children age from 8 to 12 to help them to learn programming after-school through two functionalities which are tutorial notes according to chapters and activities as exercise to enhance users' understanding. This project is developed using instructional design model approach which is ADDIE in order to achieve maximum efficiency and effectiveness. ADDIE lifecycle ensures the project has fully covered all the objectives through five phases, namely analyze, design, develop, implement and evaluate. This project is specifically developed for children in Malaysia to expose them to programming earlier, which will be a positive change to the society. The project's user interface is designed suitable for children to increase their interest in exploring the application. User flowchart when running the system is also designed for usability.

Functionality test carried out shows a good result which means this project has nearly met success. From the UAT performed, most of the feedbacks reflect good results in

all three aspects which are performance and presentation of the application, as well as its effectiveness towards users. Thus, with existence of this application, it is believed that the usage of mobile phones will be more beneficial especially among children, in terms of educational purposes. It will also put Malaysia on the same level with other developed countries where programming is taught in early ages.

5.2 RECOMMENDATION

In the future, this mobile application needs to be improved by adding more functionality in order to add more benefits to its users. Improvements made must also attract more target user group to use this application, while increase its reliability and usability. Future projects might want to expand target user group so that programming learning can be learnt by everyone through this application. Future recommendations to be taken to consideration are as follows:

1. Additional modules for deeper understanding of programming.
   WebCoder only touches the basics. Future projects might consider adding more chapters into the modules and the activities after beginner phase has passed.

2. Other programming languages like C++ into this application series.
   Number of programming languages increase in time, so learning only one type of programming language will not be sufficient if users are interested to programming.

3. Mobile compiler as one of the functionalities.
   Adding compiler as one of the functionalities in the application will allow users to write their own source codes and see the outputs by themselves. This can increase the exploration of users in learning programming.

REFERENCES

Ahmad Tazali, N. (2011). BLOG PANITIA ICT: Learning area 5: Programming. Panitiaict-semekar.blogspot.com. Retrieved 14 July 2015, from http://panitiaict-semekar.blogspot.com/2011/01/learning-area-5-programming.html

Ala-Mutka, K. (2004). Problems in learning and teaching programming–a literature study for developing visualizations in the Codewitz-Minerva project Codewitz Needs Analysis (pp. 1-13). Tampere University of Technology, Finland: Institute of Software Systems, Tampere University of Technology, Finland.

Anderson, T. (2008). Towards a theory of online learning. Theory and practice of online learning, 2, 15-44.

Asan, A. (2003). Computer Technology Awareness by Elementary School Teachers: A Case Study from Turkey. Journal of Information Technology Education, 2, 12.

Bers, M., & Resnick, M. (2014). ScratchJr: Computer programming in early childhood education as a pathway to academic readiness and success: Massachusetts Institute of Technology.

Brown, I., & Lysaght, P. (2012). Early Start: Grand designs for learning.

Buchegger, B. *Using the Mobile Phone in School*. Vienna: www.handywissen.at.

Chiang, J. (n.d.). Shall We Learn Scratch Programming

Fain, Y. (2004). Java Programming for Kids, Parents and Grandparents: Smart Data Processing.

Garrison, D. R., & Cleveland-Innes, M. (2005). Facilitating cognitive presence in online learning: Interaction is not enough. The American Journal of Distance Education, 19(3), 133-148.

Johnson, C., & Kristonis, W. (2007). National School Debate: Banning Cell Phones on Public School Campuses in America. National Forum of Educational Administration And Supervision Journals, 25(4).

Jones, S. (2011). Computing at School. International comparisons (pp. 12).

Kahn, K. (1999). Helping children learn hard things: computer programming with familiar objects and activities The design of children's technology (pp. 223-241): Morgan Kaufmann Publishers Inc.

Lin, J. M.-C., Yen, L.-Y., Yang, M.-C., & Chen, C.-F. (2005). Teaching computer programming in elementary schools: A pilot study National Educational Computing Conference. Taipei, Taiwan: Department of Information and Computer Education National Taiwan Normal University.

Pawan, F., Paulus, T. M., Yalcin, S., & Chang, C.-F. (2003). Online learning: Patterns of engagement and interaction among in-service teachers. Language Learning & Technology, 7(3), 119-140.

Radich, J. (2012). Technology and interactive media as tools in early childhood programs serving children from birth through age 8. 15.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., . . . Silverman, B. (2009). Scratch: programming for all. Communications of the ACM, 52(11), 60-67.

Robins, A., Rountree, J., & Rountree, N. (2003). Learning and Teaching Programming: A Review Discussion. Computer Science Education, 13(2), 137-172.

Salmon, G. (2013). E-tivities: The key to active online learning: Routledge.

University of Birmingham,. Literature Review in Mobile Technologies and Learning. University of Birmingham.

Weinberger, M. (2015). Learn any of these 11 programming languages and you'll always have a job - Business Insider. Business Insider. Retrieved 12 July 2015, from http://www.businessinsider.my/programming-languages-in-highest-demand-2015-6/#tTpXevRefb30kCw3.97

Weyer, S., & Camara, A. (1975). Children Learning Computer Programming: Experiments with Languages, Curricula and Programmable Devices Psychology and

Education Series. Stanford, California: Institute for Mathematical Studies in the Social Sciences, Stanford University.

APPENDICES

APPENDIX 1-A: User Acceptance Testing Questionnaire

1. Age

   ☐ 8 ☐ 9 ☐ 10 ☐ 11 ☐ 12

2. Are you a frequent mobile phone user?

   ☐ Yes ☐ No

3. On what purpose do you usually use a mobile phone?

   ☐ Games ☐ Learning ☐ Others

4. Do you understand the objective of this mobile application?

   ☐ Yes ☐ No

5. Do you see any consistency in the theme used on this mobile application?

   ☐ Yes ☐ No

6. How do you rate the functionality of the mobile application in terms of performance?

   ☐ Excellent ☐ Poor

   ☐ Very Good ☐ Very Poor

   ☐ Good

7. How do you rate the graphical user interface design of this mobile application?

   ☐ Excellent ☐ Poor

   ☐ Very Good ☐ Very Poor

   ☐ Good

8. How do you rate the mobile application in terms of user friendliness?

   ☐ Excellent ☐ Poor

   ☐ Very Good ☐ Very Poor

   ☐ Good

9. How do you rate the operational performance of this mobile application?

   ☐ Excellent ☐ Poor

   ☐ Very Good ☐ Very Poor

   ☐ Good

10. Does this mobile application help users to learn programming?

    ☐ Yes ☐ No

11. Does this application help users practice education at home?

    ☐ Yes ☐ No

12. Does this application help users to understand HTML programming in the tutorials pages?

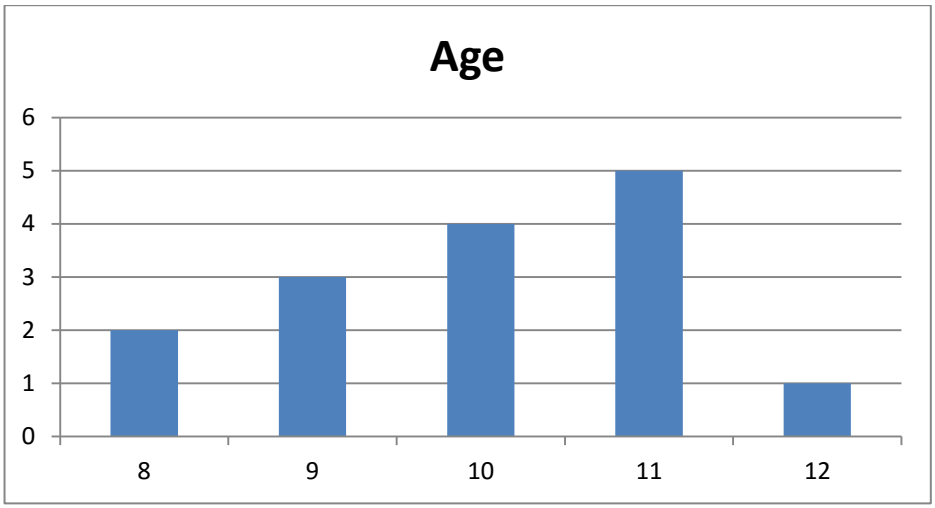
    ☐ Yes ☐ No

13. Does this application help users to increase the comprehension on HTML programming in the exercises pages?

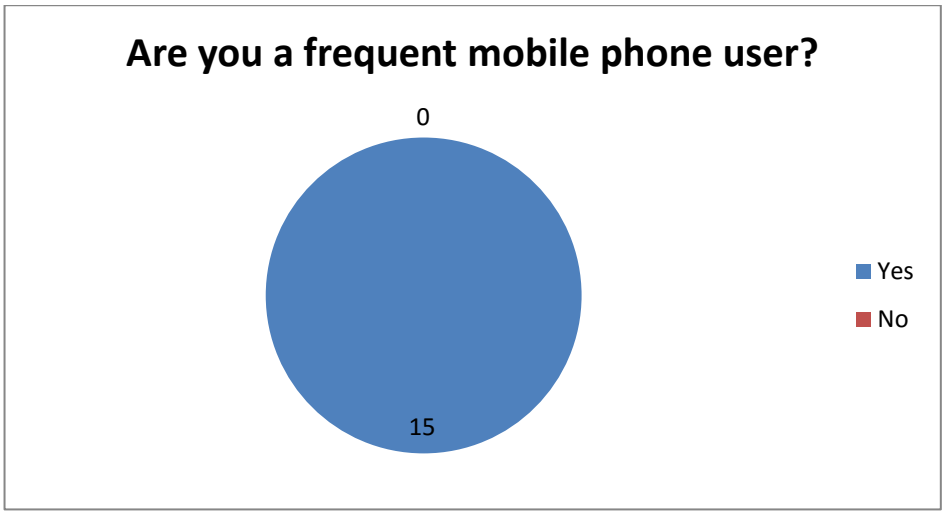
    ☐ Yes ☐ No

APPENDIX 1-B: User Acceptance Testing Results

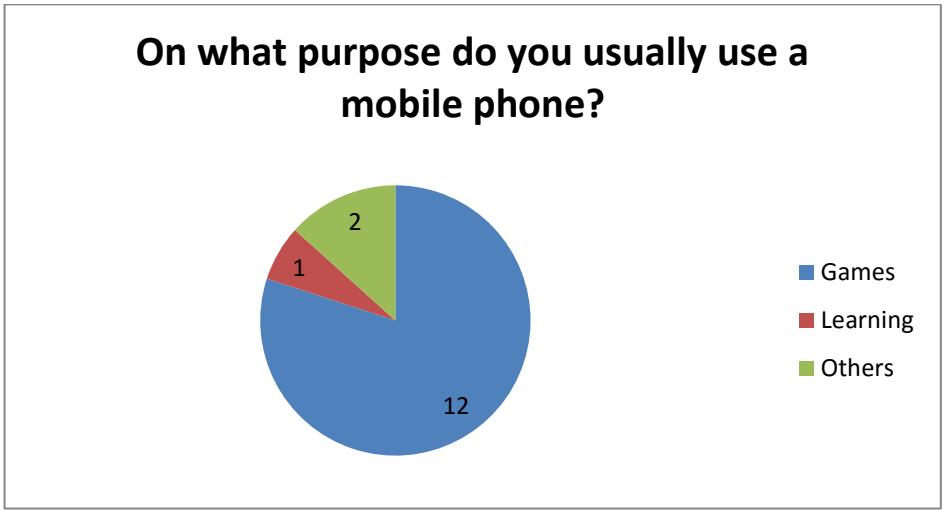
1. Age



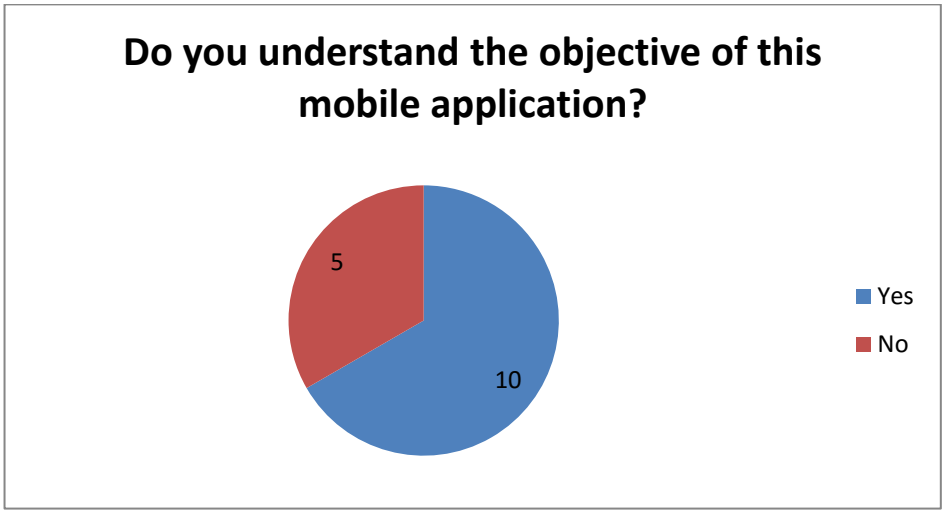
2. Are you a frequent mobile phone user?



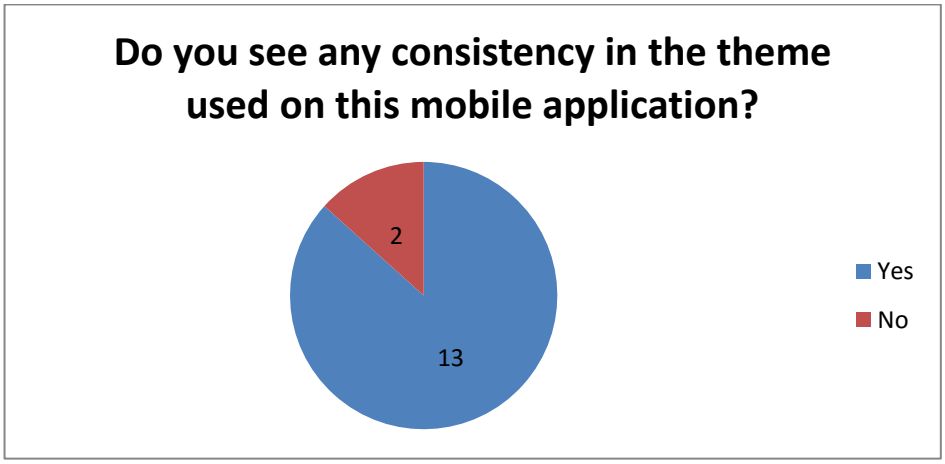
3. On what purpose do you usually use a mobile phone?



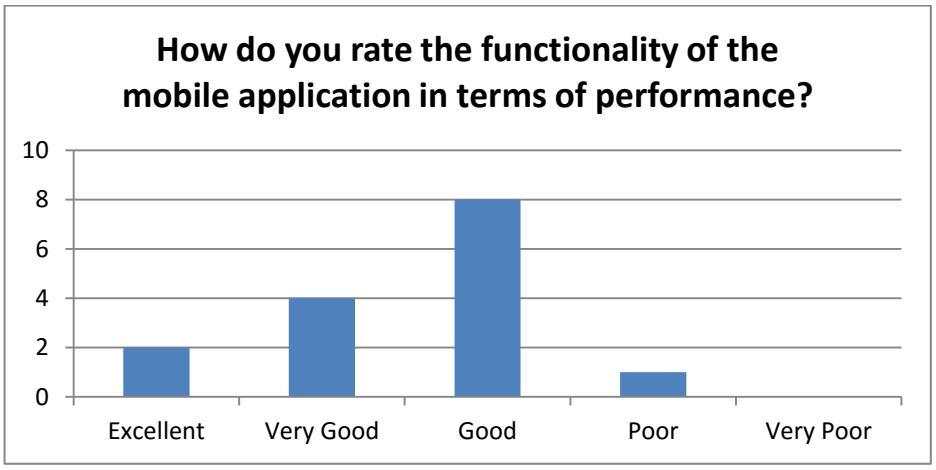
4. Do you understand the objective of this mobile application?



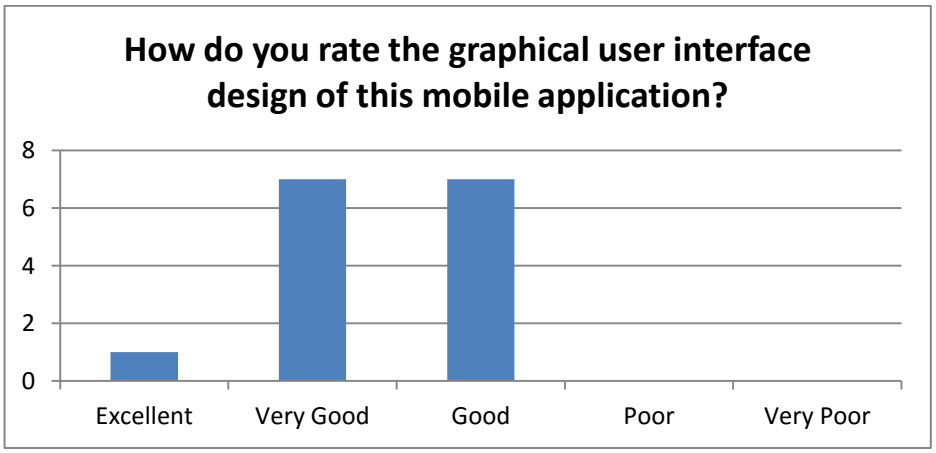
5. Do you see any consistency in the theme used on this mobile application?



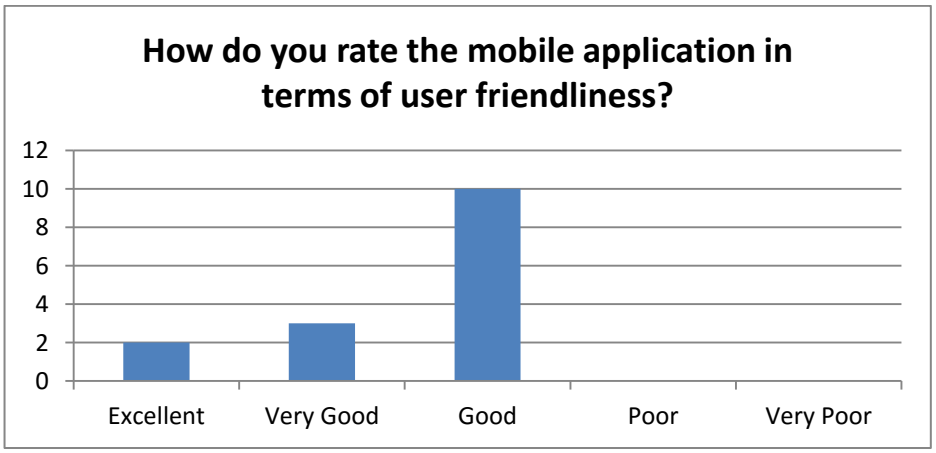
6. How do you rate the functionality of the mobile application in terms of performance?



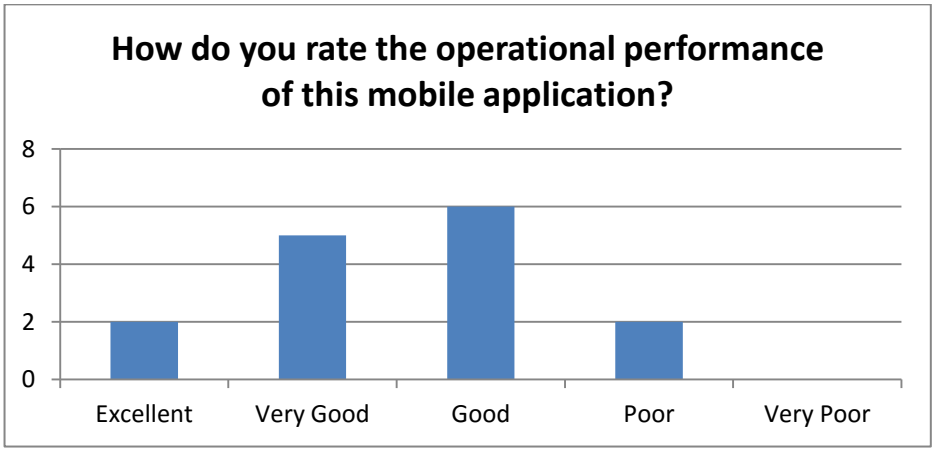
7. How do you rate the graphical user interface design of this mobile application?



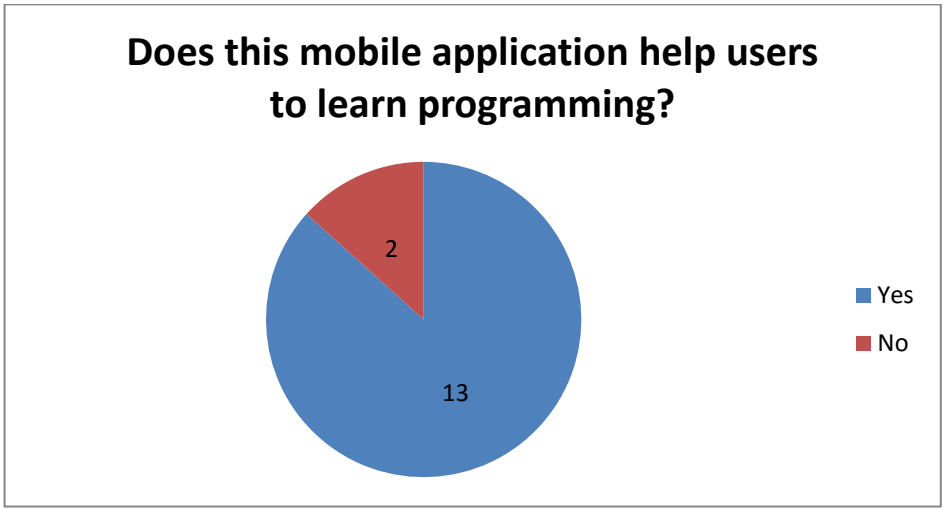
8. How do you rate the mobile application in terms of user friendliness?



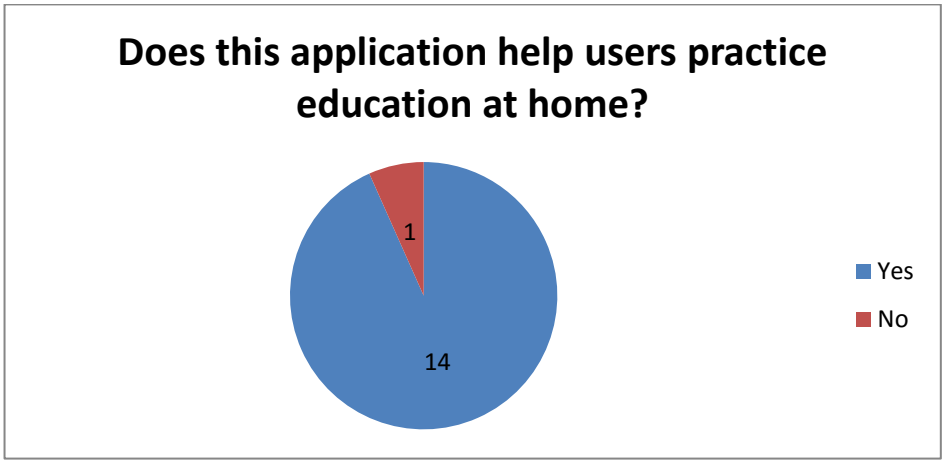
9. How do you rate the operational performance of this mobile application?



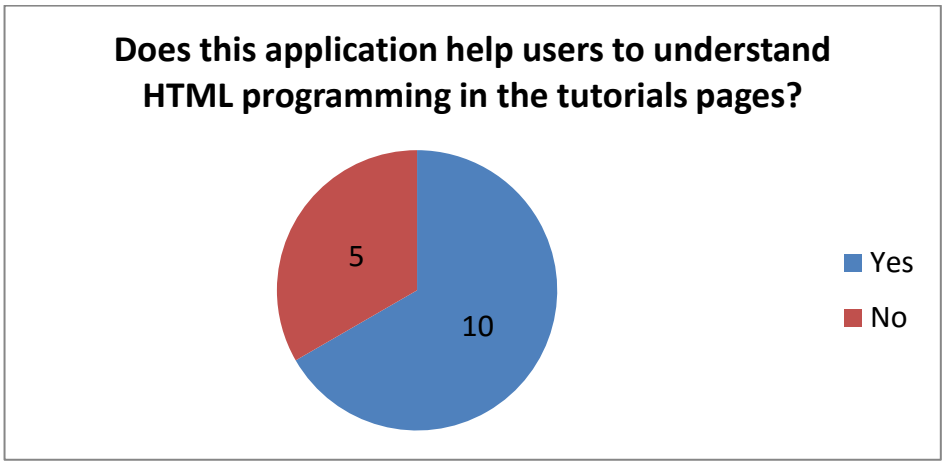
10. Does this mobile application help users to learn programming?



11. Does this application help users practice education at home?



12. Does this application help users to understand HTML programming in the tutorials pages?



13. Does this application help users to increase the comprehension on HTML programming in the exercises pages?