DEVELOPING SELF-TEST MOBILE APPLICATION FOR PROGRAMMING LANGUAGE: CASE STUDY ON JAVA LANGUAGE

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES
OF
NEAR EAST UNIVERSITY

By
MODUPE DAMILOLA LAWAL

In Partial Fulfillment of the Requirements for
The Degree of Master of Science
in
Computer Information Systems

NICOSIA, 2018

DEVELOPING SELF-TEST MOBILE APPLICATION FOR PROGRAMMING LANGUAGE: CASE STUDY ON JAVA LANGUAGE

A THESIS SUBMITTED TO THE GRADUATE SCHOOL OF APPLIED SCIENCES
OF
NEAR EAST UNIVERSITY

By
MODUPE DAMILOLA LAWAL

In Partial Fulfillment of the Requirements for
The Degree of Master of Science
in
Computer Information Systems

NICOSIA, 2018

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Modupe Damilola Lawal

Signature:

Date:

To my parents…

ACKNOWLEDGEMENTS

ABSTRACT

The current generation spends so much time on computer devices, which has the capacity to develop various applications that works efficiently and the only way to communicate and instruct the devices is by having the knowledge of programming languages. Learning programming languages teaches students how to think creatively, it also makes them smarter in that they have to think of steps and ways to instruct the computer. One of the most popular programming languages presently is the Java programming language, it runs on over 3 billion devices and it also has platform independency it allows developers to write ones and run anywhere. Java language experts are of high demand because of its features. Using mobile devices for learning Java languages attempt to make it simple and easy for low level learners, with the mobile devices the learner can self-test each anytime and place. This Thesis aims to develop a self-test puzzle mobile application for learning Java language called Java Tutor. Puzzle is been used in learning Java programming to make it more engaging, because the use of puzzle is proposed to speeding up learning capacity. The Java Tutor self-test puzzle is contains 3 stages the beginners, intermediate and advanced, each of the stage consists of 3 levels beginners 3 level, intermediate 3 levels, advanced level 3. The levels comprises of 3 question in which is chosen at random, when a question is being answered another question follows. The question are solved using a drag and drop approach to rearrange the codes correctly. With the help of the Java Tutor self-test puzzle mobile application learners who desire the knowledge of Java Language will be able to acquire knowledge and with it mobility it can be used anywhere.

Keywords: Java programming; programming languages; puzzle; self-test; mobile application

ÖZET

Günümüzde bilgisayar kullanımına harcanan zaman her geçen gün artmaktadır. Bilgisayarlar ile iletişim kurabilmek ise programlama dilleri bilgisi gerektirmemektedir. Programlama dilleri öğrencilerin yaratıcı düşünme becerilerinin geliştirilmesini sağlamaktadır. Java, en popüler programlama dillerinden biri olan ve 3.000.000.000 cihazın üzerinde çalışır durumda olan bir programlama dilidir. Java dilini mobil cihazlar ile kullanmak programlama dillerine yeni başlayan öğrenciler için basit ve kolaydır. Bu tez, Java Tutor adı verilen, Java programlama dilini öğretmeyi amaçlayan bir mobil uygulama geliştirmeyi amaçlamaktadır. Java programlama dilini öğrenmenin daha ilgi çekici hale getirilebilmesi için uygulama bulmaca olarak tasarlanmıştır. Uygulama, başlangıç, orta ve ileri düzey olmak üzere 3 seviyeden oluşmaktadır. Her seviye sürükle-bırak yöntemiyle çalışan ve rastgele sıra ile öğrencinin karşısına çıkan 3 sorudan oluşmaktadır. Java Tutor uygulaması Java programlama dilini öğrenmeyi mümkün kılmaktadır.

Anahtar Kelimeler: Java programlama; programlama dilleri; bulmaca; kendi kendine test; mobil uygulama

TABLE OF CONTENTS

CHAPTER 5: METHODOLOGY

CHAPTER 6: CONCLUSION AND RECOMMENDATIONS

APPENDICES

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

HCI: Human Computer Interaction

GPS: Global Positioning System

MOOC: Massive Open Online Course

RISE: Research in Software Engineering

RPG: Role Playing Game

CHAPTER 1

INTRODUCTION

In this section, the research problem, aim, importance of the research, and the research limitations is been discussed.

1.1 Overview

Regardless, of various research made in the past, there is still no perfect or interesting idea that would fascinate learners and enhance the learning process. Recently, there have been new ways of learning that connects with the new generation, style and human computer interaction (HCI). In 80s and 90s, researchers stated that computer could be used as an intellectual tool for learning.

The use of puzzle is designed as a mechanism for speeding up learning ability. In a puzzle the learner is meant to arrange scramble piece together in an intelligent manner, so as to solve the puzzle. A puzzle is a problem or a game that test the learner knowledge. Valen (2012) described technologies has advancing quick, education also is changing and teachers have to embrace the change. Game has been used for education for a long time like the old chess game which was used to learn strategies of war during the civil war (Barab et al., 2009). The game has been introduced into the classroom as a tool to be used by the teachers but not to take out the teacher (Bushnell, 2009 & Prenksy, 2004). Hall (2003) proposed the use of puzzle teaching Computer courses for beginner and high level courses. He observed that the concept aids to hold up broader area of individual learning style than doing the traditional methods. The puzzle are mainly identified with game playing, fun and entertainment and as a result can be less overwhelming for the learners as a learning tool.

In Computer Science, learning programming is a fundamental requirement in their curriculum (Denning, 2000). Presently, Computer programming is a common and recognized part of the undergraduate curriculum. In the Computer Science Education Community there was a controversy and still an ongoing intense research on the most presentable programming paradigm

1

(Decker & Hirshfield, 1994; Purgh, LaLande & Thomas, 1987; Bergin, 2000), language (Allen, 1998; Duke et al, 2000), programming environment (Hu, 2004) and philosophical approach (Stein, 1998; Fincher, 1999) to the teaching of introductory programming. Trying to learn a new spoken language is in some ways similar to learning to program, because in both the syntax must be learnt e.g. where the colons or semi-colons should be placed and the semantics e.g. structure. However, programming languages have been said to be difficult which makes it more challenging for novice students and as the highest dropout rate (Robin et al., 2003). The problems teacher has to deal with is that a lot of students are not interested to learn programming language. Most students now have a different way to learn, once they are not being entranced when learning they lose attention or focus (Feldgen & Clua, 2004). Zyda et al. (2008), pointed out students get disillusioned in Computer Science mostly when they see many off point examples that are not interesting or have nothing to do with the reality. Students enhance in learning with good games like learning from mistakes, learning by doing, learning based on task, learning by questioning (Din, 2006).

This study aims to develop a self-test mobile application to teach and learn Java programming language using a drag and drop technology to enable the learners to solve the exercises and with the help of a voice instructor to understand the rules of the test. Basically providing a play and learn environment for learner who are interested in learning Java and have been having difficulties with the traditional learning to learn with easy. Learning programming languages has been known as a very challenging task for most individual (Eagle & Barnes, 2009). It is broadly believed that to change a novice to an expert it takes extra time or less ten years (Winslow, 1996; Robins, Rountree & Rountree, 2003).

1.2 The Problem of Study

This section gives a view of various challenges learners/students tend to encounter in learning programming languages which gives edge to the development of this application:

- Research has shown that teaching or learning programming language is really difficult for instructors or tutors. It often leads to students dropping the course because the students been taught have low motivation and emotional disengaged, the problem is as a result of complications of the programming language (Wilson, 2002: Beaubouef & Mason, 2005).

2

- The teachers taking programming courses tends to get tired of repeating the lectures over and over just because the learners have difficulties understanding since each learners has their own learning capacity (Winslow, 1996).

- A research done by Roslina and Nazli (2009) explains that learning and teaching of programming language are boring to learners, teachers tends to give explanations that are complex for the learners, the form of teaching can also be boring to the learners and the learners don't get enough exercises to learn the subject more effectively in the class.

- The traditional teaching technique depends on lectures and teaching of programming syntax, and doesn't succeed in appealing students or inspiring learners to get them interested in programming. Furthermore, the programming languages that is chosen for the lectures are usually professional in form like C#, C, C++, and Java they have complex syntaxes, which makes it harder for beginners (Wilson, 2002: Beaubouef & Mason, 2005). Also, student have difficulties with the concept how to create solution for a problem, spilt into simpler codes and finding mistakes and challenges in understanding even the easy concepts like data types, variables (Dalal et al., 2009).

1.3 The Aim of the Study

The aim of the study is to develop self-test mobile application for Android users which is specifically designed for learners/ novice, who desire to learn Java programming language to improve their learning knowledge in the introductory concepts.

1.4 The Importance of the Study

The application being developed tends to help learners most especially novice to understand Java programming, it is basically learning without difficulty. The mobility of the self-test puzzle mobile application also saves time on accessing files, instead of having to read different books taking all the time it's easy to access using the mobile phone to learn. The developed mobile application is quite different from previous application been developed, the application covers the basic aspects of Java programming and is less complex than other applications.

- The challenge of the Thesis is to develop a self-test mobile application that can solve the learner problem, a puzzle that enables learner comprehend basic concepts of Java.
- The previously developed application for teaching and learning programming language just solve the problems of teachers and learners however, in this present study the mobile application being developed to improve learning will be using game contents such as drop & drag technology, rewards, immediate feedbacks, and voice instructor to help student and help increase motivation to learn using the developed mobile application.

1.5 The Limitations of the Study

This study has the following limitations:

- This study is limited by the period that begins from September 2017 until May 2018.
- It's limited to an Android platform and teaching just one of many programming language which is Java code, in future can be changed as preferred.
- It's limited to Java basics concepts such as Java program execution, declarations, operators, expression and statement.

1.6 Overview of the Thesis

In this Thesis there are six chapters and references, it's summarized as follows:

Chapter one contains a brief introduction about the subject of matter, which is the puzzle game a mobile application and also the aim, importance of the study, the limitations of the study and the overview of the Thesis.

Chapter two consists of two parts the first presents the concepts of games, how it being introduced into learning and an overview of the puzzle mobile application in programming languages, different reviews on developed applications using puzzle style to learn programming and also the main topic which deals with Java programming.

Chapter three gives details on the technologies involved in this study. In the study we have few technology been used like mobile technology because it deals with Android application and reviews on past developed application to teach programming language.

Chapter four gives details on the developed Java Tutor mobile application for learning Java programming languages using Android. System requirement of the whole study, Software development life cycle, System architecture, flowchart, and the operating system requirements.

Chapter five gives description of the implementations of the system been developed showing the methodology used when analyzing the data and the results obtained from the study.

Chapter six contain the concluding statements and recommendation made to help improve the developed mobile application.

CHAPTER 2

RELATED RESEARCH

In this chapter, the reviews existing theories and technologies to develop the application in view were been presented. It gives details on what game is all about and gives reviews on previously developed self-test puzzle mobile applications how it's been able to help the students.

2.1 Related Research

Adams (2005) made a hypothesis that game help learners to comprehend it doesn't educate learners by themselves, the idea is that game are not like the teachers. Games are learning tools that help students to investigate and comprehend issues, it prepares learners for different circumstances still depend on how it's been translated and requires the teacher's explanations (Ulicsak, 2010). There are various types of game: card game, board game, puzzle game and video game.

Games and puzzle have a past together as far back many years. A puzzle can be either question or problem which perform two features: it designed to entertain the learner and a clear solution. Hill (2003) assumed that making a work puzzle-like it will be more entertaining. It will aid to make sure that the learner will work on a problem with means adequate to acquire the effective routine learning. Michalewicz (2008) said that in education a puzzle perform four principles:

Simplicity: Educational puzzles should be stress-free to state and stress-free to remember. This is also very essential, as easy-to-remember puzzles surge the chance that the solution method, including the universal mathematical problem-solving standard, is also remembered.

Eureka factor: The puzzle will give the learner hard time solving but at end an interesting solution. A puzzle should be motivating because its end result is counter-intuitive: the learner often use instinct to start their search for the solution and this tactic can lead them off track. Eventually a Eureka! Moment is reached (Martin Gardner's Aha!), when the right route to answering the puzzle is acknowledged.

Generality: Educational puzzles should describe some general mathematical problem-solving values. This is of key importance. Most people agree that problem solving, like any other skill, can only be learned by cautious exercise, i.e., by solving problems. However, this activity must be sustained by strategies provided by an instructor. These general strategies would allow for solving new problems in the future.

Entertainment: The puzzle should be entertaining and engaging, the puzzle can to complicated but should be entertaining to solve.

Badger et al. (2012) suggested that a puzzle is a problem that is complicated and requires a solution using creativity or solution in an unexpected way. Unraveling out a puzzle often ends in a eureka (excitement if find the solution) moment, will make the learner glad, and sometimes finding the solution can be hard and complicating but the progress is usually entertaining. Puzzles that are good are logically perplexing and in a mathematics teaching context the best puzzle are those which have both standard solution and a complex exercise (De bono, 1967).

Generally, games are play that individuals relate to teach rules, problem solving, goals and interaction all given as a story. Games captures the basic need to learn by giving motivation, inspiration, and entertainment all when the learning is happening. A lot of educational games are inadequately constructed and are not fun also, many games even ones used for educational purposes don't have useful learning materials. There are few qualities a good educational game should have which includes: interesting storyline, constant challenge, feedback, rewards and flexibility.

Sergiy (2017) study investigated the influence of the recurring use of puzzle as an instructional strategy in the teaching and learning of engineering mathematics. The objective of applying the puzzle to the teaching and learning mathematics was to engage students who were having complication in the course, in their words boring to heighten their broad thinking skill and lateral thinking out of the box. A lot of students taking the engineering and mathematics courses drop out because the course is too complex or boring, there was a term such as academic disinterest and emotional disentanglement. Puzzles can engage the student feelings, ingenuity and interest and also develop their abstract understanding. The idea of a puzzle means no routine, non-standard, unstructured question introduced in an entertaining way. Sergiy evaluated student's attitude towards the instructional strategy by distributing questionnaires with puzzles, the puzzles offered

didn't require special knowledge to the students taking engineering mathematics. The result show 98% of the accepted that puzzle increases their problem solving skills.

Egenfeldt-Nielsen (2007) discovered there are three generation of learning games. The first generation includes drill and practice game which adopts the behaviorist technique to order the player. The second generation builds the inspiration of the learners by combining player concentration, focal point, interest, and illusion. They require the player to create wide comprehension in conformity with the cognitivist technique while the third generation depends on the constructivist technique which means the player makes information through encounters and communication with social groups. In the behaviorist technique switch in behavior is believed to show some learning is taking place. The technique expresses that compensating or rewarding players for a specific conduct urges them to carry on similarly in other circumstance. The reward boost behavior. On the other hand, if behavior is penalized, the player is more averse to repeat it. In behaviorism, a person can learn not to get thing done and additionally to get things done (Carlton, 2012). The code hunt programming game demonstrates an example of the behaviorist technique and its breach between gameplay and educational idea. The player in this game win point for discovering missing code fragments which is a good way of learning programming languages. Cognitivism technique are mostly used by the educational game which endeavor to incorporate learning and game encounter and consequently build up inherent inspiration which is individual self-inspiration to seek after a task for which no reward is given. For the cognitivist technique it is critical to exhibit the learning content in various ways, while considering restriction and possibilities of the individual mind by giving various important contextualized exercises, visualization, control, feedback etc. Cognitivism concentrates more on skill been learned than content, and in addition meta-skill like critical thinking which have gotten much consideration from researchers. The constructionist technique of teaching its essential that the learners intensely develop understanding by using external achievement. The educational games adopting the constructivist technique are mostly called micro worlds and sandboxes, more simulations than games. By the means of direction of communication with objects, the learners find their connection and properties. The concentration again is not on the content but on the skill like problem solving and critical thinking, and creativity. Constructivism basically intent to turn the learners into developers of content by allowing them to design game, with that acquire understanding about programming (Egenfeldt-Nielsen, 2006).

2.2 Mobile Applications on Learning Programming Languages

The various researches shown below gives reviews on past developed application to teach programming languages that has been developed on different platform such as web based, pc based and also mobile based.

Generally, programming is been believed by students to be complex to teach especially the beginner students (Muratet et al., 2009). Many students at the beginning hope for an immediate understanding in a very early time of learning process while the programming concepts require a longer period of time to understand and later on the students can see improvements (Anderson & McLoughlin, 2007).

In Computer Science course programming is one of the most important subject, but teaching programming is not an easy task or learning either it's as a result the subject been complicated in nature that it require special understanding of the concepts. To improve on this problem developing games to boost student learning and skill in programming is a good one, like: crossword games is played on to teach basic of programming. The crosswords is a word puzzle game, the objective is to fill in letters in the platform which will form words by unlocking clues that gets the answers. The clues are programming questions in the crossword game.

Bayliss (2009) proposed choosing a game first approach to teaching the basic parts of programming. In teaching programming languages the level of course assignments and contents are most emphasized on, with the course assignments it is possible to learn as the students become interested and also the visual components in games already developed show students their mistakes in the code and direct them.

Dale and Patricia (2006) made use of a web-based tool used to build puzzle that provides practice of basic programming principles in an entertaining puzzle like format called the parson's programming puzzle. The parson's programming puzzles consist of a set of drag and drop styles exercises which gives learners good learning practice on Turbo Pascal. The parson's puzzle are in a drag and drop style. In each exercise the learner in given selections of code fragments which should be arranged in correct order, the fragments are drag able some contain multiple line codes, the learners drags the choices to the indicated answer locations and by clicking the button "check" the feedback shows and gives the performance rate of the learner. The parson's programming puzzles is a web-based application and it's used in class in a networked computer lab also accessed

outside the class using internet. It was designed using the hot potatoes construction tool. Hot potatoes set consist of 6 applications allows users to create interactive crossword, matching and ordering, multiple-choice and fill-in-gap exercise for the web. The hot potatoes made the drag and drop exercises construction really easy. This application was made as a result to help learners because programming exercises can be very tedious and difficult to motivate learners. Researcher such as Kearsley & Sheniderman, 1999 state the importance of task engagement in learning.

O'kelly and Gibson (2006) the IBM developed and released a game for learning programming Robocode in 2001. It was first started by Mathew A. Nelson for personal use in 2000 which became more efficient and well qualified for general use when he introduced it to IBM. The game is aimed to teach programming language using Java programming language, in the game the players try to program a robot in order to fight another robot in the virtual environment. The Robocode game are been used in teaching programming languages in universities, it is easy to understand and fun to learn. The teaching concepts of the Robocode are the introductory concepts of a structured programming language.

Long (2007) study examined one game the IBM Rodocode. Investigated the characteristics of the Robocode game. A survey was conducted on Robocode players. After the examinations on what motivate individual, results shows fun (enjoyable) was the frequent reason which came up and other reasons was to win the game, win the prize. Also, examined reasons that made Robocode fun one was been able to solve problems yourself, secondly been able to be innovational. The game is started by the players creating a robot using Java. The Robocode framework has some rules the robot has to take and the players then compete in the internet based group, there each robot search and eradicate other robots while shielding itself. The robots which wins are the ones with best plans. The game allows players learn and apply concepts of object oriented programming like: well-arranged classes, extension, and using inheritance (Bonakdarian & White, 2004). The game was utilized in the classroom environment to examine the student learning reaction. Another study shows that the Robocode implemented in the classroom the student were delighted in the project and gave some really innovative results (Bierre, Ventura & Egert, 2006).

Chang and Chou (2008) in their study applied an educational game to motivate learners which is interactive called Bomberman to develop C programming language course, the game allows the

learners write/read codes and also able to view learning materials. The game contains challenges and storyline which makes it fun, the learning environment created by the game engages learners to spend time practicing with the inspiration to win the game. Few researches believed that the problem was with the teaching approach not the type of program also stated that syntax and language was a small part of programming. The game is a maze computer based game first designed by (Hudson, 1983).

Roslina et al. (2010) in their search to enhance learner's motivation and learning perception concerning programming used an educational game as their learning mode applying five constructs interface, motivation, perceptive progress, and attitudes. Two online educational games were developed crossword and duck shooting, the simple game field was used so the learners will not have to spend much time understanding how to play rather they can easily play and learn fast. The game specifically gives exercises on C programming language and looping was the main topic in addition few basic programming concepts like data types, variables and input-output statements. The game contains levels in which the learners unlocks as he/she plays the game. The game gives the learners exercises that make them have to evaluate correctly and the system gives a feedback, the game also provides learners with hints and clues to learn which can help with the learning contents.

Muratet et al. (2010) developed a game called Prog & play, which is a web based and real time strategy game called kernel panic. The kernel panic utilizes Computer Science analogy such as pointers, units, and bits. In which student write the programs to control units in a battlefield, the task are given by clicking on a map like view with the mouse. The player gives instructions through a program and the program can compose a set of commands and send it to the game after the command is received it executes the commands and changes the game state. The student can choose the programming language they would prefer among Ada, C, Java, C++, OCaml and Compalgo (interpreted language). This study concluded that Prog & play can be interesting for more than just practicing programming but also for student studying Computer Science.

Frederic et al. (2012) developed the game called Codin game. Its gives challenges to be solved, allows players to play with programming with increasingly difficult puzzles and the player as to

code an object to interact in the platform in order to win. It's a multiplayer video gaming contests and supports single player mode players can keep track of results. The codes the players execute results into an animation, it gives the direction in which he or she is coding to. This game used different languages at least 25+ to control the object in the environment such as C++, C, Java, Pascal, Php, Perl, Go, Bash etc (http://en.m.wikipedia.org/wiki/codinGame).

Khenissi et al. (2013) described in a paper a learning game for education the game called Instructions right place game. This game allows learners enhance knowledge from the use of drag and drop technology to construct programs in a fascinating way. The game was developed due to the fact that programming languages are complex to teach and the learners often have problems of low motivation also the difference between learners and teachers generation, teachers were taught in a pre-digital generation and the style of education of the past while students are taught in the digital world are used to the computer, game and internet. Using the training of the past will be boring of the student that they won't have interest in it. The instruction right place game allow the teacher interact, the teacher can add name of the game and can allocate a game to any level the teacher can also write question and the learners will see it. The game provides a learner model where the learners mistake and errors is been retained and the reasons for the errors. The game organize the use of the drag and drop technology, instructions and use of avatars, the instruction right place game breaks down complex programming task into fragments and instruct the learners with steps on how to form the program. The game was developed using CSS, Javascript and HTML it also an online based application. It also gives immediate feedback that helps learners allowing them to construct good program and gain good record.

Tillmann et al. (2013) handles a platform called Pex4fun. This platform, grading of traditional Massive Open Online Course (MOOCs) assignments been switched to an automated grading assignment system based on the symbolic execution, it's designed for both students and teachers. The pex4fun platform is a game based learning tool and teaching environment, in which teachers can organize interactive games for students. The students can learn programming by playing programming games in pex4fun also having programming duels with each other.

Tillmann et al. (2013) Research in Software Engineering (RISE) developed a game based learning application called Codehunt in 2014. Codehunt is known as a serious game the learners have to

write codes to continue in levels, it's a web based application which consists of various levels where the learner understanding are been tested. In Codehunt puzzles are used and the players have to examine by means of tips given. Codehunt uses the test/clue generation of Pex, a white-box test generation tool, the Pex executes a directed search to find possible execution. In solving a puzzle in the game is the physical process of operating search based test generation the test data to be generated by player is the code of the player. The objective of the game is to figure out what the code should do and fix it to get all the test correct.

Stuart Perks (2014) used pedagogical approaches and learning styles to design an Android game to teach C programming called C Rocks. The main of the study was to create a learning tool using various learning styles not just an entertaining game. Studies found that the new generation spend a lot of time playing computer games from 15 above (Prensky, 2014 & Hudson, 2013). It was also discovered that students don't wish to spend time studying and getting information they just want to get instant results. Prensky (2014) found that playing games can make people concentrate for a long time, it seen that technology has change the way people learn applying game to the classroom can be helpful for the students. The game involves the learners with task that needs to be completed, by exploring the levels and try work on the tasks using the constructivist approach also the game deals with only few topics in C programming it gives a feedback which encourages the learners to find more information on it.

Ibrahim et al. (2014) approached solving the issues student facing in learning programming languages by introducing the Scratch Programming Environment (SPE). In their work, they made students create very simple and basic games using the SPE. It was established (by dividing a group of students into two subgroups where a group was taught programming language based on pascal programming language using the conventional method and the other using the SPE platform) that the approach significantly boosted the motivation of students who used the SPE towards learning programming language compared to their counterparts. After this experiment was conducted, students were asked about their interest in programming language and 65% showed interest after using the scratch programming environment. This method will not only increase student's motivation for learning programming it can encourage the new age youth to have more interest in the educational system.

Ragab et al. (2015) in their quest to solve the challenges that university students face in learning programming came up with the platform iPlayCode which provides an exciting way to learn languages such as Java, Objective C, and Java for Android etc. It was developed using Xcode 5.0.2 and cocos2d-x development tool and adobe Photoshop graphic design. The aim of the framework is to help develop games that can be played by students who aim to improve and surpass their present programming skill using their mobile devices.

Ocay et al. (2016) conducted a study which was to develop an interactive mobile game based learning application for object oriented programming subjects. The application called JavaLite was developed to assist the Urdaneta city University-College of Computer Science in teaching programming courses like Java. The old process of teaching used by teachers of the Urdaneta city University in Computer Science in teaching programming courses was a combination of sharing power points presentations and using board works when delivering the lecturers to the students but the process has been known to be tiresome on the teachers part it not easy to prepare the notes and lessons also time consuming. The developed system contains lesson notes, browsing e-dictionary feature and playing test games. The lesson notes is a learning module the student will read the notes afterwards will be directed to play test which consists of two phases: debugging and word twist games ones the students completes them successfully more lesson will be unlocked. In the debugging test the students are given codes to debug they will examine program snippets in unity with the language. In the word twist game the student a given programming words which a twisted/scattered to be arranged correctly. In conclusion the features added to the new system was really helpful in attaining students learning in object oriented programming it increased the interest of the student.

Yassine et al. (2017) conducted a study to find educational approaches and gameplay techniques comprised in advancement of serious game for teaching programming languages. An educational game was developed called "Perobo" used to teach most difficult part of C programming which is pointers, it also focused on taxonomy scheme to describe the learning levels. The study used educational approaches and gameplay techniques to teach C programming, the game play represents the manner in which the learner interact with the game, its mains to solve the problem of abstract concepts like pointers using playing styles that achieve real element. Most common gameplay techniques: drag & drop, point & click. The drag & drop is best for beginners by using

a mouse or finger to move the fragments of element to achieve results. The point & click involves interaction with elements of the game also best for beginners it requires the learner to use sense (see and hear) to interact. The learner will explore the game environment and thinking clearly for the answers to point and click. The game make the learner know the definition related to pointers as variables in C programming, understand types of pointer and functions also learn syntax of a pointer. The Perobo game raise thinking using the drag & drop and point & click gameplay. The game still requires immediate feedback channel to encourage the learners.

2.3 Summary

There is a missing gap in the literature of the applications for teaching and learning Java programming language, a lot of literature found to teach and learn programming languages but non-specify on teaching beginners the basics. In the past mobile application the questions weren't categorized into stages and levels which helps students determine their level of ability with the introductory concepts of Java, previous applications didn't classify the rewards given to the students on each questions, also didn't use the voice instructors.

CHAPTER 3

THEORETICAL FRAMEWORK

This chapter comprises the basic concepts of mobile game based learning application and the overall mobile application development. And also some basic concepts on Android, programming languages, learning styles and game based learning.

3.1 Mobile Technology

Mobile technology is a technology that allows mobility, it involves the use of mobile phone, mobile computing, and portable electronic devices. These technologies are portable and can allow users to perform different task e.g. cell phones, iPad, laptop and global positioning system (GPS) and more (Kendrick, 2013). The evolution of mobile technology provides great effect to Information and Communications Technology (ICT) world as well as to electronic learning community. It allows services that can gain access to and used anywhere and anytime.

The use of mobile technology such mobile devices is rapidly spreading and making lives better, it makes communication easier and also work related assignments such sending files around the world (Anastasios, 2005). Mobiles devices are allowed to use different type of communication technology including Wi-Fi, Bluetooth, 3G, dial-up services and the mobile device can be networked to a home or office Internet anytime (Kendrick, 2013).

3.1.1 Advantages of Mobile Technology

Using mobile technology is a technology that is portable like cell phones, tablets, laptop also improves communication all over the world. It also has many benefits:

- Mobile technology has made it possible to communicate with people around by just connecting to the internet you can reach to people across the world.

- Mobile technology like the mobile device can be well entertained in the classroom to the desktop computers make accessing information easier, better and more comfortable to study (Asbere, 2013).

- Mobile technology is said to help university students in researches and allows them get quick feedback from supervisors (Macwan, 2017).

- No matter what device you are using, it is to make work less stressful (Miangah & Nezarat, 2012). Helps students become more efficient (Cavus & Ibrahim, 2009).

- It allows lecturers to create more exciting way to educate students. Now lecturers can use videos, pictures and graphics to teach students (Macwan, 2017).

- Mobile technology are now used in the lecture hall, helps students who are unable to retain lessons to be able to record and ongoing lecture and also lectures can create an online discussion forum to encourage the students to talk about topics (Ozdamli & Cavus, 2011).

3.1.2 Disadvantages of Mobile Technology

Using mobile technology in education also has its negative impact on the students.

- In using Internet there are privacy issues. It's easy to get accounts hacked when using internet so you need to check privacy.

- It makes students lose interest in studying. All information now can be accessed online either using computer or mobile data, student slowly gave poor reading habit and start skipping classes thinking that they have got all information on Internet (Macwan, 2017).

- Using mobile devices maybe seen as a tool of distraction for some students, many get carried away by the social media activities and forget their self, sometimes leads to accidents pressing the devices while walking (Kendrick, 2013).

- Few of the mobile devices are expensive.

3.2 Mobile Devices

Mobile device is a computing device portable to be handle and operated using the hand. They are mainly called handheld computer e.g. tablets, smartphones, e-readers, and music player with smart capabilities are all mobile devices. These mobile devices are connected to the Internet and interconnected with other devices like headphones via Bluetooth, Wi-Fi, cellular networks in order to perform task like calls, text, and also be used to play games. There are different types, styles and models of the mobile devices available to the public, they come with various technology smartphone, PDA, camera phone and touch screens (Priya, 2011). Mobile phones are not just a mere gadget for communication, it is also used as a guide and a tool for building up and upgrading education. Mobile gadgets become more productive in every aspect of human life. Mobile device enhance the skills, knowledge and creativity of individuals. Different operating systems run on the mobile devices that permit third party applications to be installed on it (Hanson, 2011).

- *Smartphones:* Smartphone has the Internet and Wi-Fi access and also cellular communication capabilities which makes it widely used, it's a powerful version of a cellphone. It has so many features making calls, text messaging, voicemail and can connect to the Internet which makes it almost same has a computer same features. The most popular smartphones are IOS and Android which is growing rapidly. The smartphone is tending to replace the use of devices like laptops, camera, and music players because it has both capabilities in one devices, very portable and user friendly (Hanson, 2011).

- *Tablets***:** It's like a computer but in a smaller and portable size, the main difference with the computer is tablets is without keyboard they have touch screen allowing you to type on a keyboard and with your finger serving as a mouse. The tablets doesn't really do everything computer will be able to do most individual might still need computers to use some programs (Anthony, 2013).

- *E-readers***:** Is also called E-book readers they are like tablets used for reading electronic books, these are books that a downloaded from online sources. Common e-reader lines are Amazon Kindle, Kobo, Noble Nook, and Barnes &amp. E-reader mostly use e-ink shows that is simpler to browse than a CRT screen (Hanson, 2011).

3.3 Operating System

Operating system is a system software that handles the computer hardware and software assets and offers shared services for computer programs. The operating system also manages the mobile devices and is called mobile operating system. Mobile operating system includes few examples: Google Android, Apple IOS and Microsoft's Windows phone Operating System (Vangie, 2011). Some of the mobile OS have no restrictions on what can be downloaded or who can download the software it known as an open source software

3.3.1 Google Android

The mobile operating system called google Android and it was built by google based on linux kernel and devised for specially touchscreen smartphones and tablets. It's an open source operating system meaning no restriction on what can be downloaded on the mobile devices.

Android established by Andy Rubin in October 2003 and later procured by Google in 2005. Android is a free Linux based platform and is an open software stack with an operating system, middleware, and applications. It was created by Google and released on November 5, 2007 for mobile platforms (Alabaster, 2013). Android is prevalent with technology companies they need an instant, minimal effort. (Stallman & Richard, 2011).

3.3.1.1 Architecture of Android Operating System

Android is an open source mobile operating system which is built on linux with Java platform. It's a self-possessed of diverse software things, including Android operating system architecture or software stack, Android runtime, native libraries, linux kernel and applications:

- *Linux kernel:* It is the basis of the whole Android operating system and google has made certain variations on it (Singh, 2012). The linux family of operating system is established on kernel and distributed on computer systems like personal computers and servers, mostly in the form of linux distributions. The primary software are process management, memory management and device management. The hardware is operated by Android operating system consists of various hardware devices with properties like virtual memory,

networking, drivers and power management controlled by linux kernel (Meyer & David, 2010).

- *Android runtime:* It is an application runtime environment used by the Android operating system. The Android runtime and dalvik were been created for Android, the Android runtime executes the dalvik executable format and Dex bytecode specification (Andrei, 2014).

3.3.2 IOS

IOS is a mobile operating system developed by Apple. This operating system powers many mobile devices like iPhone, iPad and iPod touch. Initially introduced 2007 for the iPhone, iOS has been prolonged to help other Apple gadgets like the iPod Touch and the iPad. In January 2017, Apple's App Store comprises in excess 2.2 million iOS applications, 1 million of which are local for iPads. The indicated mobile application have jointly been downloaded in excess of 130 billion times (Sinicki & Adam, 2016).

3.3.3 Windows phone

Windows phone is a mobile Operating System developed by Microsoft for smartphones. Windows phone employs certain technology and tools, which are used in the station based application development such as the frameworks Silverlight, development environment and .Net compact (Ziegler & Chris, 2010). Windows phone is well known for its tile based interface based on Microsoft metro design language which can modified at will.

3.4 Programming Languages

Programming language is a set of rules given to instruct the computer to perform a given task (Aaby & Anthony, 2004). Programming language consist of three groups: machine language, high level language and assembly level language. High level languages are languages that allow programmers to write code in a more user friendly context and has high level of abstraction from

the computer, such as BASIC, C, C++, COBOL, Java, and Pascal each languages with their set of keywords and syntax (Vangie, 2010). Machine language are set of instruction codes in binary form which represent the operations on the computer. Assembly language is a low level programming language for a programmable device. Programming languages can be used to create programs that implement specific algorithms (Pierce & Benjamin, 2002).

To each of these languages have certain features that make it appropriate to varying degrees for the implementation of certain types of software on computer. Programming is a perplexing, difficult process and the source of difficulty does not seem to be the syntax or understanding of concepts, but debugging the program especially for novice students (Ala-Mutka, 2004). Programming languages are applied in every area in the field of computing. But Khenissi et al. (2013) debated that teaching and learning programming languages are undeniably not a straightforward task for teachers and students (Khenissi, Essalmi, & Jemni, 2013). All programming languages have some primitive building obstructs for the depiction of data and the procedures or alterations connected to them. The primitives are distinct by syntactic and semantic guidelines which describe their structure and importance individually (Aaby & Anthony, 2004):

- *Syntax:* Is the programming language surface form. Programming languages are typically purely textual, apply sequences of text containing words, number and punctuation, just like writing normally. But some other languages that are more graphical in nature, use visual relationship between symbols to specify a program. The syntax of a language defines likely groupings of symbols that form a syntactically correct program.
- *Semantics:* Is opposed form of syntax. There are static semantic and dynamic semantics: Static semantics defines restriction on the structure of valid texts that are hard or impossible to express in the standard syntactic form. While Dynamic semantics expresses how and when the different construct of the language ought to produce a program behaviour.

3.4.1 Importance of Learning Programming Languages

In this present age, everyone is surrounded by technology and almost 90% of the world population use a smartphone in their daily life (Simon, 2017). But there is a different in knowing how to use the program running in a smartphone and knowing how they work, understanding how it works gives more knowledge about the digital world (Priya, 2011).

- In this new era, everything is computer related even the young ones are involved with computer technology there is no way escaping the use of computer, one can easily communicate with this computer by learning programming language (Jaffer, 2016 & Husain et al., 2013).

- Having the knowledge of programming languages gives the capability to solve problems differently and with a logical thinking (Maloney et al., 2004).

- Programming can make students more productive, efficient, and effective (Jaffar, 2016).

- Being a programming expert will guarantee a job in the ICT world (Jaffar, 2016).

3.4.2 Difficulties of Programming Language

Learning programming language can be interesting and a rewarding skill. Currently in this technology age there are more request for programmers which increases the student's interest to learn programming. However, learning programming is difficult to novice. Winslow (1996) said it demands roughly 10 years to transform a novice into expert in programming. An expert in programming have efficient knowledge of problem solving strategies and schemas (Guindon, 1990). But novice has no knowledge in programming like an experts. Soloway and Spohrer (1989) said the novices have difficulties understanding constructs in programming like loops, arrays, variables and testing of codes. Melki (2013) observed the students showed difficulties in the abstraction, the memory concept, and a misunderstanding between the equality operators in mathematics which is sculptured into the thoughts of students during their long years in schools with the new assignment operator in programming. Studies shows that learners have it hard with the first stage of learning programming. It was discovered that learners have problems in writing, designing and reading simple programming codes (McCracken, 2001). Finding out causes that learner learning difficulty observed that the most difficult topics encountered by students were memory related concepts, the learner aren't efficient in generating a strong rational model of memory effort during program execution. The learners encounter more problems which are: learner lack skill in the techniques of analysis problem, don't understand and masters the programming syntax and constructs, unproductive use of representation techniques for problem solving.

3.4.3 Importance of Java Language

There are many reasons why learning Java language is important to students, some reasons are listed below (Mayur, 2016):

- Java language is piece of the main 5 most prominent programming language, its runs on about 3 billion devices which has made Java programmers of great demand in various telecommunication companies.
- Learning Java language gives students the basic skills to learn any other object oriented language.
- Java is easy to learn and it also versatile.
- Java is one of the most in-demand language and will be used in the future.
- Java is platform independent, write code ones and can run anywhere.

3.5 Mobile Application

The propelled period of information and communication system, individuals are acquainted to utilize pc and pc application. There is a worldwide progressive effect of mobile application. With the use of mobile application advanced nation are becoming facilitate and people, society of emerging nation are elevating each other and creating a new kind of information technology infrastructure (Priya, 2017). These days, large number of individuals utilize mobile application to communicate with friends, surf Internet, edit document and save, entertainment etc. Mobile phone user can get facility of mobile application everywhere, individuals can do many things of his day by day life and business life (Ngai, 2007). The mobile application run in a mobile environment which usability hang on numerous factors like screen resolution, hardware limitations, expensive data usage, connectivity issues. Most recent couple of year's mobile companies are attempting to improve mobile device with larger screen resolution, larger storage, effective connectivity which gives better environment for present day mobile application (Eric, 2010).

3.6 Puzzle

Puzzle can be define has either a problem or a question which is perplexing, it test inventiveness, dexterity and skill of the learner. A puzzle is usually associated with the game playing, fun and leisure and therefore engaging learners. Badger et al. (2012) defines puzzle has a problem that is

complicated and either has a solution that requiring considerable cleverness and perhaps an imaginative thinking solution. Puzzle is often considered a mode of entertainment, but then again puzzle can be seen as a mathematical or logical problems. One is the important aspect of a puzzle is to understand what is been learnt by solving the puzzle and how can the knowledge be applied to solve other problems. Puzzle are educational, however they demonstrate suitable problem solving procedures in a very entertaining way. Puzzle aids learner to learn problem solving skill by the three methods by experience, which is noble as learner can only learn by solving problems, by imitation which is easy it helps apply techniques and principles and by reflection puzzle inspires us to reflect on what is been learnt?, how it's been learnt?, how what learnt be used?. There are various types of puzzles such as: logical puzzles, mathematical puzzles, lateral thinking puzzles, meta-puzzles, mechanical puzzles, tangram, Sokoban, chess puzzles, jigsaw puzzle, word puzzle, sliding puzzle, drag and drop puzzle, and puzzle box (Michalewicz, 2008).

3.6.1 Important of Puzzle

The use of puzzle is an introduced design used to engage learners learning capacity. There are various important of puzzle to the learner few are mentioned below:

- Puzzle permits learner to learn problem solving skills by experiencing, by imitating, and reflecting (Sergiy, 2017).
- Puzzles emphasis on making learners think about framing and solving unstructured problem (Michalewicz, 2008).
- Puzzles are assumed to increase motivation and interest.
- Puzzles aid learners to comprehend what they have learnt by solving the puzzle and in what manner they can apply the knowledge to other problems (Falkner, 2012).
- Solving puzzles associate to the creative thinking required for solving advanced real-life problems (Marzocchi, 2016).
- Solving puzzles is capable to be connected to the development of professional skills.
- The importance of puzzle drives further than the courses done in schools for the reason that the lesson learnt can be applied to any other course (Dale, 2006).

3.6.2 Why Puzzle

The motivation behind using the puzzle strategy based on experience, that puzzle is under-exploited in the teaching of programming languages and problem solving. And it's understood that the results of learners of productive puzzle based courses include an increased confidence in problem solving (Sergiy, 2017).

3.6.3 Importance of Puzzle in Learning Programming Languages

The puzzle designed as a tool to speed up learning process, the approach helps persuade learners that:

- Programming languages can be help learners knowledge in problem solving.
- Programming language is useful and interesting to learn.
- Learning programming language is very important and can be used in day to day living.
- Programming languages are not as hard as they seem, it's easy.

The few points stated are important to the learners due to the fact that they don't understand the significe of the courses and they mostly don't see the relations between the courses taught and the real world (Michalewicz, 2008).

3.7 Learning by Doing

Learning by doing is an active learning method for teaching, it's a means of learning from experience resulting right from learners own actions, as compared with learning from watching others perform, reading others instructions or listening to others instruction. Another definition of learning by doing, it about inducing been engaged in an activity and due to the process of doing this activity, learning about things like: how the activity works, how to find the activity, what the activity makes learners think, and what doing this activity enables the learner (Ludovic, Gossiaux, Rauch & Tabiou, 2005).

3.8 Universal Instructional Design (UID)

Universal instructional design is a procedure that embroils bearing in mind the imaginable necessities of the learners when designing and delivering information (Elias, 2011). It also means recognizing and eradicating unnecessary obstacles to teaching and learning while preserving academic rigidity.

Universal instructional design is mostly about truly universal thinking, it more than just ease of access to replicating on how to maximize learning for students and learners liking while minimizing the requirement for distinct accommodations (Burgstahler, 2001). The universal instructional design is constant and supports accomplishment of the university guiding principles of learners centeredness as well as learning objectives. It has got upright principles to be used in mobile learning because it consistent with universally standard principles of moral teaching which have been confirmed to boost learning for all students (Conell, 1997).

The universal instructional design principles is a outline of instructional resources and scheme that causes the learning aims attainable by individual with varied difference in their skills to hear, see, speak, read, write, organize, engage and reminisce (Elias, 2010). Universal design for learning is attained by aid of adaptable curricular skills. These substitutions are manufactured into the instructional design and operating systems of educational materials. Universal instructional design principles have been established in demand to create products and services suitable to the varied range of divergence (Burgstahler, 2005). Conell et al. (1997) stated that universal instructional design principles were been established to construct litheness of procedure into both the instructional design and operating systems of educations material to be proper to the extensive range of students.

The Table 3.1 below shows the recommendations of universal instructional principles generally for mobile learning (Burgstahler, 2001):

Table 3.1: Universal instructional design principles recommendations for mobile learning (Elias, 2010)

Universal Instructional Design Principles	Mobile Learning Recommendations
1. Equitable use	• Deliver content in the easiest imaginable format • Use cloud-computing file storage
2. Flexible use	• Package content in small pieces • Consider unconventional assignment options • Be flexible in presentation
3. Simple and intuitive	• Create simple code • open source software • Simple interface
4. Perceptible information	• Be explicit • Readily perceived • Descriptors
5. Tolerance for error	• Support learning methods • Issue warning sound and text
6. Low physical and technical effort	• Consider issues of physical effort • Minimize unnecessary requirement
7. Learning space	• Learning space that allows both students and instructional methods

- *Equitable use:* Means the course content used be accessible and fair. To follow in mobile learning requires building content and tasks which can gain access to on an extensive range of devices. Such is making the course content easily accessible to any student also providing easy access to the mobile learning application using Android is cheap and open source.

- *Flexible use:* Providing flexibility in use. This principles requires the course to allow wide variety of students skills, likings, schedules and selections in method of use. Multiple ways of get into and relating with material and signifying their knowledge.

- *Simple and intuitive:* The design is in clear and straightforward mode, consistent for the students. Tools are intuitive easy to understand or grasped by intuition. All distractions or complexity that will detract from learning should be removed.

- *Perceptible information:* Universal Instructional Design does not deduce that all students are mentally or cognitively capable to access all media in the same way. Hence, it search to maximize the simplicity of each elected medium of communication, and also passes information over multiple passages.

- *Tolerance for error:* Universal instructional design principles also reduce dangers and opposing consequences of errors in software operation by designing learning environments with a tolerance for error.

- *Low physical and technical effort:* Universal instructional design identifies that students will be of different range of ages, backgrounds, physical characteristics, and personal conditions. This principle demands for bearing in mind the physical effort required to complete the course and removing any that is pointless or not appropriate to the learning procedure. The learning ought to be about the material not about attaining the material.

- *Learning space:* Universal Instructional Design knows that learning occurs in virtual as well as physical space. These spaces must be available and accommodate all learners. The design of the spaces should also support the type of learning been facilitated (Elias, 2010).

CHAPTER 4

DEVELOPED MOBILE APPLICATION

This chapter provides discussion on the design methodology and implementation of the self-test mobile application. The system architecture is to describe each component and explain the methods used to implementing.

4.1 Software Development Cycle

The procedure of developing computer software has remains formulated by different development methodologies. Software development methodology is an outline that is used to manage, plan, and control the procedures of the development of software system (Bassil, 2012). Formally, Software development cycle is used in several fields such as Software Engineering, Computer Science, Mechanical Engineering and Applied Engineering. Software development cycle consists of various models: Waterfall, Spiral, Agile and Rapid application development are few of the most successful models. These models all comprise of a series of phases or step that developers have to follow and completed so as to get results. Waterfall model is one of the first software development life cycle models consist of five steps: Analysis, design, implementation, testing and maintenance. As result of the success of Waterfall model numerous software development firms and industries have adopted it as their prone development framework and software development life cycle to plan and maintain their resources. Waterfall model is a traditional approach to mobile application development taking methodical and systematic route (Royce, 1970). It is used when a developer is sure about is objectives or scope, and can visualize the requirement specifications, timing. The Waterfall model is a sequential development procedure in which growth is known when it's flowing increasingly downwards through the steps. The Waterfall model was used in this study because its popular, used for long term projects and also the developer knows what is required and the objectives of the study. It consists of five steps namely:

- Analysis
- Design

- Implementation
- Testing
- Maintenance

This application is a demo design to prove that self-test mobile application can actually motivate student to learn Java programming language. It's an Android based application specially designed for novices student between age 16 above.
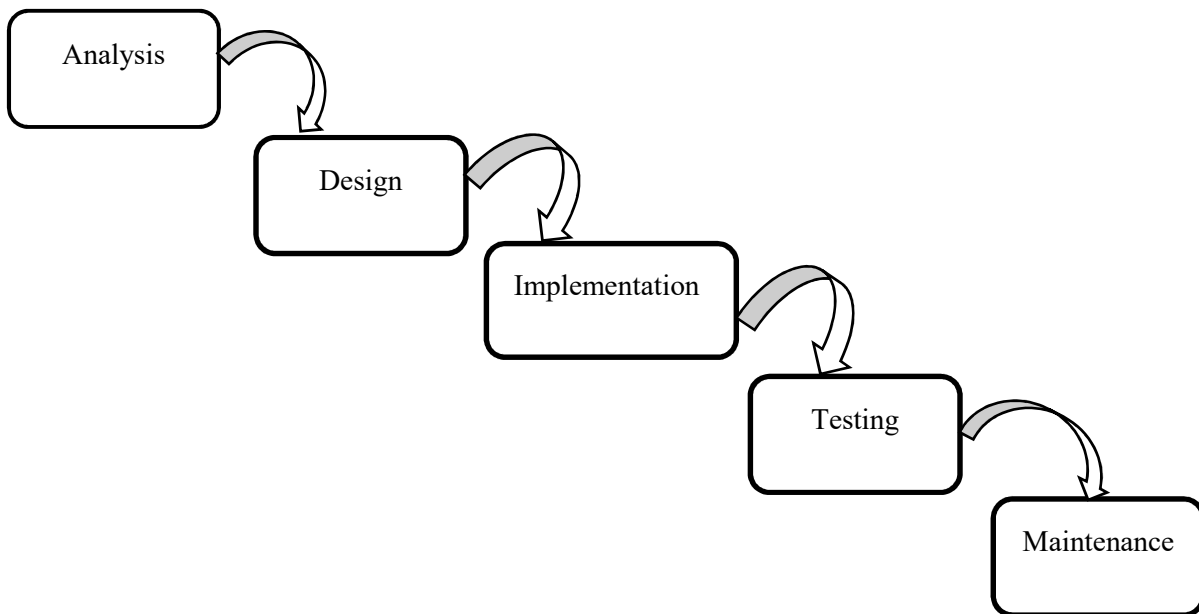
Figure 4.1: Software development life cycle using Waterfall model (Bassil, 2012)

4.1.1 Analysis

The system analysis is the first stage in the Waterfall design model, it knows as the software requirement specification is a clear and complete description of the actions of the software to be

developed. Analysis of the developed application is basically what is expected of the application and the process of development of the application.

It connects system and business analysts to describe both functional and non-functional requirements. Usually, functional requirements are described by aid of use cases that terms the users' interactions with the software. They comprise such requirements as purpose, scope, perspective, functions, software attributes, user characteristics, functionalities specifications, interface requirements, and database requirements. The non-functional requirements indicate to the several principles, restraints, limits, and requirements enforced on the design and task of the software slightly than on specific behaviours. It consist of such properties as reliability, scalability, testability, availability, maintainability, performance, and quality standards.

4.1.1.1 Functional Requirements

The application is developed following specific software requirement. The functional requirements shows how the Java Tutor self-test mobile application for Java language should work to different inputs or behave when it been operated on, basically giving details on the application activities.

- The self-test application should be able to allow learners find new learning mode.
- The learner should be able to enjoy the test exercises.
- The learners should be able to use the application whenever or wherever.
- The learners should be able to see through the score record their performance on the self-test to rate the progress.

- *Welcome page*: Is the splash screen page, on this page the name of the application will appears which is Java Tutor.
- *Login page /sign in*: This page provides a platform for sign up if users is new and sign in if user is existing in the application. The login page is important as it required to store users score record at each test.
- *Introduction*: this page come right after the login page, it gives an explanation on the objective of the application.
- *Home page:* After the splash screen page displays a new page will appear which is home page it contains two buttons. The first button named "I want to learn more about Java

programming" when clicked links the student to a web page that gives tutorials on Java and any other the learner wish to learn. The second button named "take the test" when clicked links the student to the instruction page.

- *Instruction page:* Gives full instruction on how to play the game, it will include voice instructor on how to play the game also has one button which is "take the test" links the student to the test page.

- *Test page:* After the instruction page the test page appears when the learner clicks on take the test. This page contains of three different stages which has three different levels each. The first stage is the beginner stage the student must start from that stage and get the answers correctly to move to the next stage because other stages are locked. The beginner stage consist of three levels each with three questions to be answered correctly. When a learner clicks level 1, three questions are been displayed after which the student will play to move to next level. The second stage is the intermediate stage can only be unlocked after the student finishes with the beginner stage, the intermediate stage consists of three questions if the learner fails any question of the three question he will have to start that same questions until he or she gets it right. The third stage is the advanced level also consist of three different questions. After finishing the test the overall scores is been displayed with the username.

4.1.1.2 Non-Functional Requirement

These are requirement that specifies principles in which the operations of the software can be judge or the properties that is built upon of a software to have.

- **Performance:** The developed application will be quick to execute and have a feedback that has a fast response time.

- *Availability:* The developed application will be available for students age 16 and above, it also an open source there by free on google play.

- *Reliability:* The developed application will be reliable in performing any of its operations.

- *Usability:* The develop application will be simple and easy to use and also easy to understand.

- *Maintenance:* The application should allow update and modification of the score record and users.

- *Security:* The application grant access to Android phone users to install.

4.1.2 Design

This phase is the process of planning and problem solving. At the design stage the requirements of the application starts becoming reality and the application is designed and program is being developed. The developer defines the plan that consists of software architecture design, conceptual schema, concept design, and graphical user interface design.

4.1.2.1 System Architecture

The system architecture gives detailed information on the structure of the application and how each features connects. The proposed self-test mobile application for learning and teaching Java will run on an Android platform, the mobile application is been developed for beginners willing to learn Java it important to make sure the application is easy to understand.

The Java Tutor mobile application is built around a very simple but fully functional architecture which employs easy to implement technologies to achieve the outlined aim and objectives (in summary, encouraging students to learn programming through gaming).
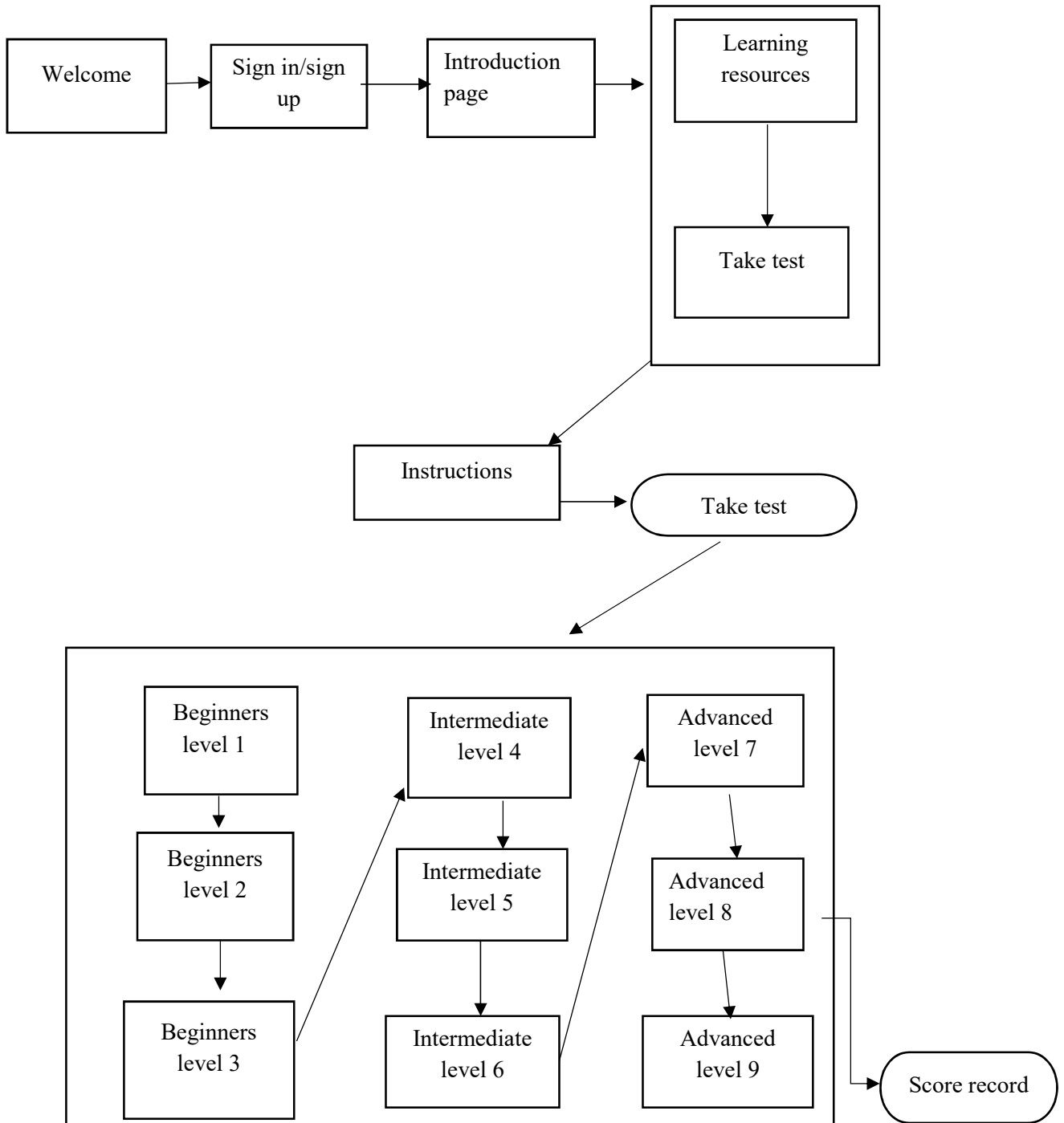
Figure 4.2: Architecture showing the block diagram of the developed mobile application

The figure 4.2 shows the way the application works. The first screen is the welcome page, introducing learners to the application, the second screen is the sign in page which links to the login page where the learner are requested to inputs email and password, and the database uses it to store the scores of the student. Following that, is the introduction page on this page the main and objective of the developed application is explained. Next is the home page where there are options to the learning resources and the test quiz, the test options links to the instruction page after which the test menu which contains 3 stages beginners, intermediate and advance all consists of 3 levels each, each levels got 3 questions which is picked in random. The test starts from beginner level, a student can't click into any level because all levels are lock until the first level is completed and passed by the students then the level get unlocked as they progress.

4.1.2.2 Universal Instructional Design Principles

The original seven principles of universal instructional design principles used in learning environments listed below (Burgstahler, 2001):

- Principle 1: Equitable use
- Principle 2: Flexibility in use
- Principle 3: Simple and intuitive
- Principle 4: Perceptible information
- Principle 5: Tolerance for error
- Principle 6: Low physical effort
- Principle 7: Learning space and size for approach

These principles were not actually developed for mobile learning environments but are equally significant to them. These principles for designing online learning is more advanced importance when designing mobile learning.

- Why Universal instructional design?

Studies discovered that different reasons can shake a learner's ability to learn, like a learners preferred styles of learning. Some learners can learn when listening while some working in groups, few using creative thinking (Elias, 2011). Some psychological factors can impact learning like stress, having interest in other things than studies and anxiety, there are also physical barriers that impact learning such as disability e.g. blind, deaf and handicap (Palmer, 2002). The universal

instructional design as an approach studies the likely needs of all learners in the design and delivery of instruction. The design offers approaches that eliminate any barriers and deliver flexibility to allow students to access learning in ways that is seems clear to them (Burgstahler, 2007).

4.1.2.3 Design Principles

The principles of the design are the building blocks of the proposed application. They are what makes up the design.

- *Login/ Authentication*: The login page allows the application store the learner's information and use it to store record. The learners are asked for their email and password in order to sign in to the application. For example: email: selftest@gmail.com password: ******* without the appropriate character it will be invalid but ones the correct characters are inputted the learners get access to the application data.

- *Levels*: The test contains 3 locked stages namely beginners, intermediate and advanced level each of this stages comprises of 3 levels in total 9 levels. The learners must go through each stages one after the other to unlock the levels. At each levels there are 3 questions which shuffles automatically at any play.

- *Rewards:* Adding rewards like scores in the self-test puzzle makes the puzzle more exciting for learners to monitor self-performance. The beginner's level 1, intermediate level 4 and advanced level 7 are scored 5 points for each questions because these levels contain simple exercises in introduction to the stages. The beginners level 2, intermediate level 5 and advanced level 8 are scored 10 points for each questions because these levels contain mid difficult exercises which means more fragments to arrange. The beginners level 3, intermediate level 6 and advanced level 9 are scored 15points for each exercises because these level are more difficult more fragments and complex codes. The rewards pop-up ones the learner gives correct arrangement but its incorrect the learners losses 2 points after 4 attempts. The score record pops up after beginners level 3 to see the progress level and the learner will see if he/she is got low or high mark to proceed or start again. The score record also show up after intermediate level 6 to show the progress score of intermediate level. The score record shows after the advanced level 9 gives final total score. The rewards can

also be viewed from the test activity page and test menu. After each stages are completed the score record appears to show the learner progress.

- *Attempts/ clue:* The attempts makes the scoring system organized and also makes the test a little bit perplexing for the learners. The test contains 27 questions and each questions the learners are giving 4 attempts which means after 4 arrangement without correct arrangement the correct answers pop-up to give the learners clue on how to answer the question again but the learners automatically losses 2 points.

- *Audio:* The application provides both voice help instructor and notes for the learner. Its gives the instruction on how to take the self-test and also the objective of the application. It gives a voice ones the learner is progressing in the test.

- *Feedback:* The developed self-test mobile application gives the learner immediate feedback, if the learner drags four times and doesn't get the answer it will be a fail 2points and if the correct answers is given the score will pop-up.

- *Drag and drop:* The learner takes the test by dragging and dropping the code to rearrange it. The questions in the application are separated into fragments that are mixed up and the learner are meant to fix them up in the proper order.

- *Rules:* The puzzle as it rules the learner should follow to complete the self-test. The learner has to start the test from the first level before the next level will be unlocked.

4.1.2.4 Java Tutor Self-Test Application Mechanics

The Java Tutor self-test mobile application has three mechanisms to improve learning efficiency, consists: Incentive mechanisms, Competition mechanisms and Drag-drop mechanisms.

- *Incentive mechanisms*: Generally means a bonus or reward something that motivate or encourages. Most mobile learning application like educational game or quiz rewards progress of the learners and penalize failing. As a result this test application has a penalty mechanism. The learners will be deducted 2 points when make the wrong arrangements of the codes after given 4 attempts and for the correct answers the learners get 5 points as an award which increases through the levels.

- *Competition mechanisms*: The application allows room for competition with other learners as it provides the score record of all the learners which pop-up at different stages

for any learner and shows the score list of each of the learners, it also indicates the learners the level of their performance and therefore required to rate the learning process. The learners find it easy to reminisce the information and find more facts within the problem.

- ***Drag-drop mechanisms***: The drag and drop is actually being used as a gameplay technique modified into educational games. The importance of this mechanism is that it is appropriate for beginners that desire to learn. The drag and drop require on mobile phone just using the finger on touch screen and drag the fragments to the correct line to generate a result. For example, the application been developed in of this study Java Tutor the learner has to drag and drop pieces of code and put them in a right order to solve the problem. The aim of using this mechanisms is give the learner a realistic environment with stimulus drawing the learner.

4.1.2.5 System Activity Sequence

Activities that can be engaged in by the system actors are:

Step1: Launch App

Step2: Display Login Page

Step3: Username

Step5: Display home page

Step6: User access the learning materials

Step7: User takes the test

Step8: Test solutions are evaluated and ranked accordingly by the system

Step9: User repeats steps 6 and 7 (optionally)

Step10: User exits the app
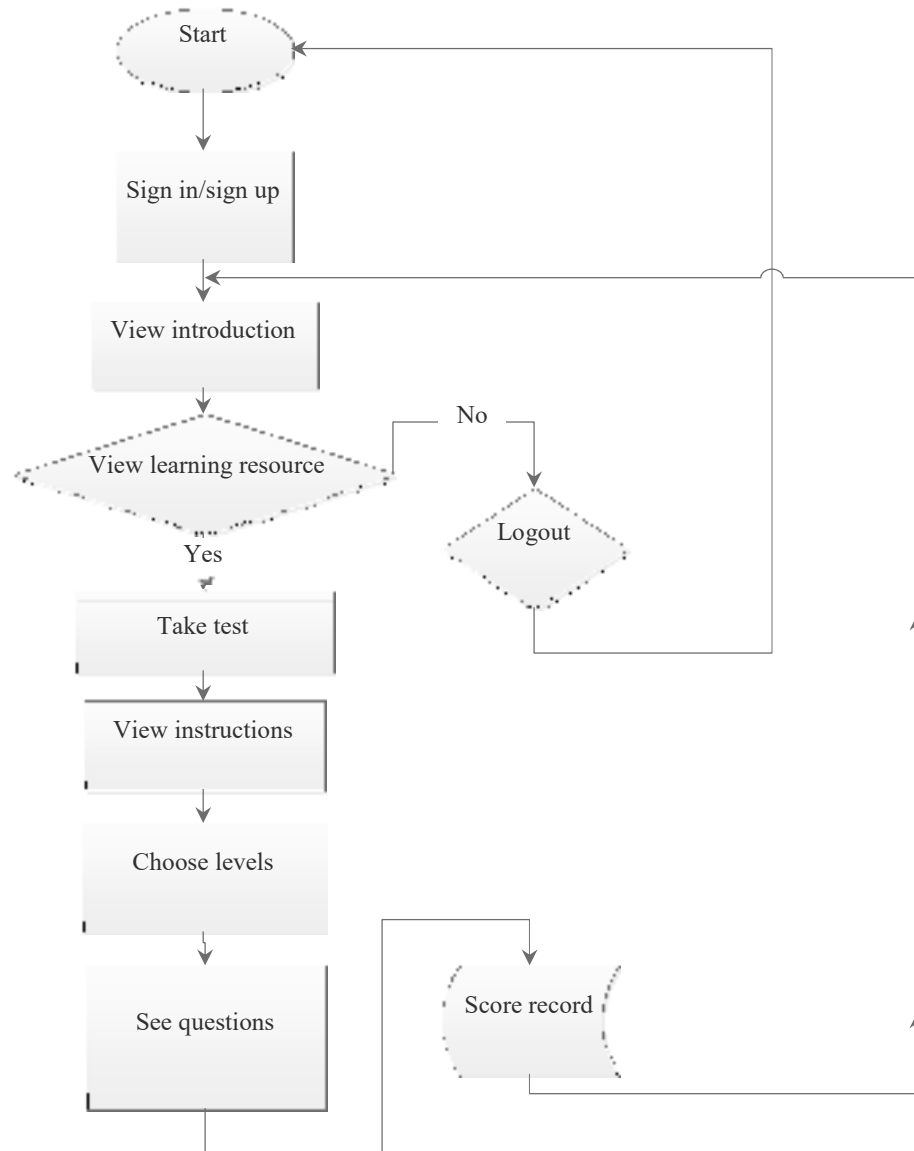
Figure 4.3: Flow-chart of the developed mobile application

4.1.2.6 Use Case Diagram

The use case diagram demonstrate a given presentation image of the feasibilities that can be performed while utilizing a specific application from the learners point of view. It establishes the association among learners and self-test puzzle mobile application based on the design.
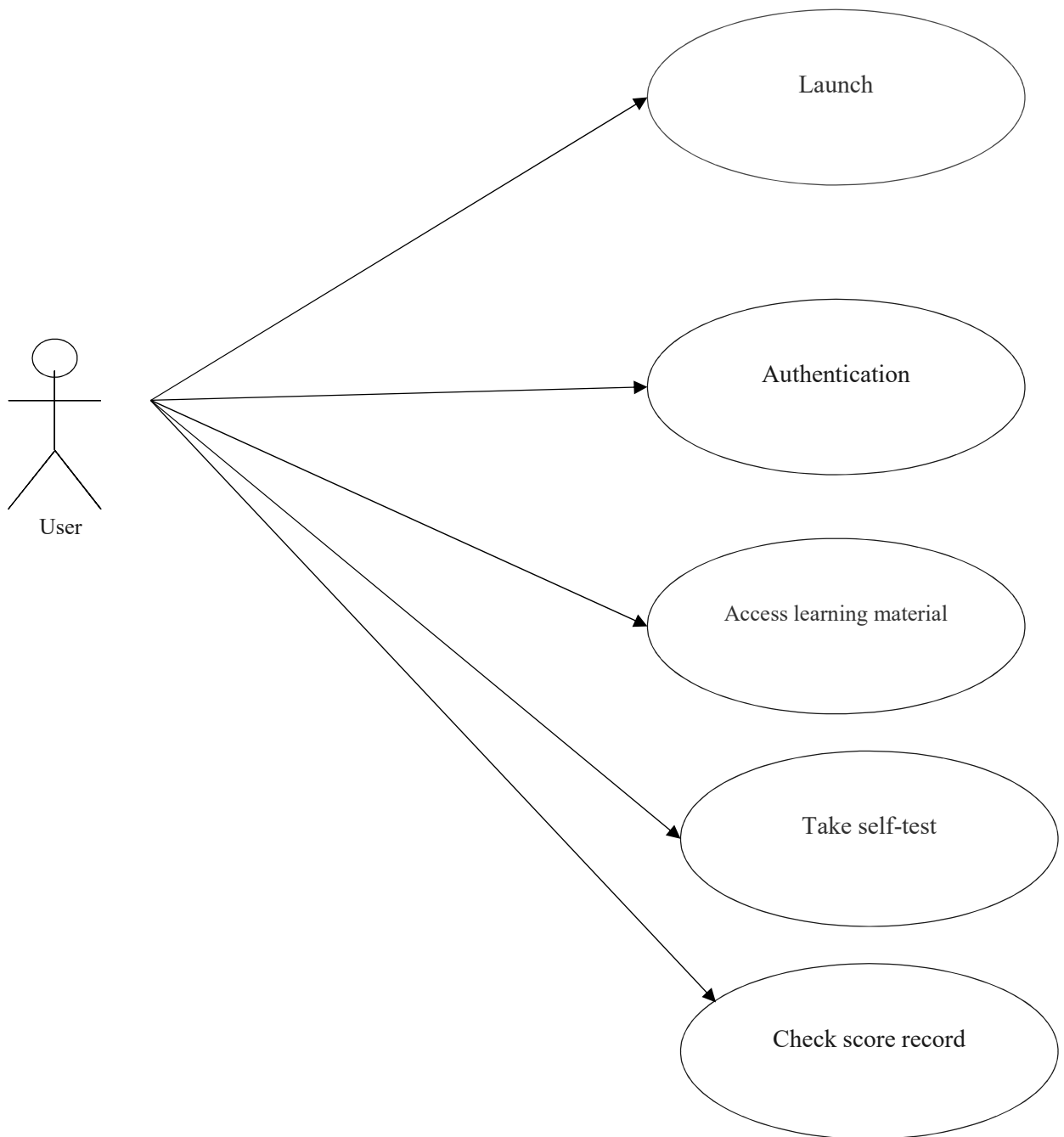
Figure 4.4: Use-case diagram of the developed application

4.1.3 Implementation

The development of the application happens after the application has undergone the design stage. The development deals with the programming, designing the whole inner structure, documenting and verifying all in creating the application.

4.1.3.1 Android Studio

Android studio is an integrated development environment for google Android operating system, built on JetBrains inteliJ IDEA software and designed specifically for Android development. It is used instead of the Eclipse Android Development Tools (ADT) as a basic IDE for native Android application development.

4.1.3.2 Database

Firebase real-time database is a cloud hosted database. It stores and sync with a No SQL cloud database. It uses a real-time to sync data to all users and still operates when the application goes offline. Firebase is a web and mobile development platform. Firebase is a back-end service and a back-end service is importance for application, it's possible to make your own backend but it consumes time. Firebase assist to ease those effort and pains by storing data and making it easy to access that data from an application quick. It manages the application users and what they can be allowed to access. There are few services the firebase database provide to the clients such as:

- *Real-time:* The firebase real-time database uses data synchronization every time data changes, if a device is connected it receives update within milliseconds. The service provides application developers an API that consents application data to be synchronized through to clients and stored on firebase cloud.
- *Firebase auth:* Firebase auth mean firebase authentication is a service that can authenticate users using only client side code and also includes a user management system where developers will permit user authentication with email and password login stored with firebase.
- *Accessible from client's devices:* The database can be retrieved straight from a mobile device or web browser it doesn't require an application server to get access.

4.1.3.3 Java Programming Languages

In developing a mobile application, it can be written in different languages depending on the mobile phone platform. In this thesis the developed system was written in Java programming language. Java is a very popular programming language commonly used from Android applications. It consist of the Java runtime environment which has the Java virtual machine and API as its components.

4.1.3.4 Application Questions

The questions used in this game based mobile learning application for learning Java were gotten from two webpages www.beginnersbook.com this is where the Java code and Questions where gotten from. The beginners book is a book special for beginners, in the web page there are lots of tutorials for students it gives opportunities to learn various programming languages. It's also an open source website so anyone can use it. The second web page is www.tutorialspoint.com this is where the learning resources in the application is linked to. It good for students because it gives notes on different subjects and there are also video tutorials. These two website were chosen after frequent discussion with the supervisor of the department and after making careful search online for tutorials that a beginner can find easy and gives a step by step explanations on what coding is all about. The tutorialpoint webpage was preferred because it easy to understand, it gives details on the basics of programming languages with examples and it will be very useful for beginners. In addition, before taking information and learning materials from these site, an email was sent as to request for the permission of the author to access the material and the permission was granted. In Appendix C shows the reply of the author of the tutorialpoint website.

The app is designed to focus on Java Basics. The Java Tutor mobile application contains 3 stages that rates learners performance beginners, intermediate and advance in this stages the learners a given questions from the easy to perplexing, the stages each contains 3 levels each for instance beginner level 1, level 2 and level 3 which level 1 got 3 questions, level 2 got 3 questions and level 3 3 questions. The questions are selected into the levels by the level of difficulty and simplicity. These basic concepts include:

1. The structure of a Java Program - Few questions on this topic is given in the beginner level 1/ level 2 and for learners to understand programming languages it starts with the structures. The structure is very important in learning codes without it there is no understand.

2. Java Program Execution – Beginners level 3 intermediate deals with questions on the execution of the programs

3. Declarations – Intermediate level 1 to level 2 deals with exercises on declaration of the program ones the learner understands the program structure the program begins to make more sense.

4. Operators – Intermediate level 3 to advance level 1 deals with exercises on operators. Understanding how to work with operators is very important in solving arithmetic problems.

5. Expressions – Advance level 2 deals with exercises on expressions which is very much essential in Java program.

6. Statement – Advance level 3 deals with exercises on statements of a program after the learner gains knowledge on the structure execution declaration and operators it easy to understand the statement in a program.

The exercises/questions in the application were basically picked according to the steps of understanding a Java program piece by piece for a beginner has to start from the beginners which is the structure of a program, it's the only way the learner can understand what is been taught start with the basic little by little to the complex problems.

- *Java Program Structure:* Java source code file is usually considered a compilation unit. This units are made up of classes and sometimes packages depending on the scale of the application been built. A Java class contain data and method members-these shows the state and behaviour of the various objects in the program.
- *Java Program Execution:* The Java compiler translates Java source file into machine-independent byte code. Each class defined as public has its byte code placed in a separate file, this is to enable the Java runtime system easily locate it.
- *Declarations:* Java variable may refer to an object, an array, or an item of primitive type. They are defined using the following simple syntax:

'TypeName, variableName' e.g. (i)   int a;   //defines an integer

   (ii) int [] b; // defines a reference to array of ints

   (iii) Vector v; // reference to a Vector object

   - *Operators:* Operators enables programmers to establish arithmetic, relational, logical etc. relationships between various objects in their programs.

The operators, in order of highest to lowest priority, are:

-- ++ + - ~ ! (TypeName), * / %, + -, << >> >>>

< > <= >= instance of, == !=, & ,^, | .&&

||, ?:, = *= /= %= += -= <<= >>= >>>= &= ^= |=.

   - *Expressions:* Expressions are essential building blocks of any Java program, usually created to produce a new value, although sometimes an expression simply assigns a value to a variable. Expressions are built using values, variables, operators and method calls. While an expression frequently produces a result, it doesn't always. There are three types of expressions in Java:

Those that produce a value, i.e. the result of (1 + 1)

Those that assign a variable, for example (v = 10)

Those that have no result but might have a "side effect" because an expression can include a wide range of elements such as method invocations or increment operators that modify the state (i.e. memory) of a program.

   - *Java Statement:* A statement in Java forms a complete command to be executed and can include one or more expressions

4.1.3.5 Application Features

This application was developed to encourage and improve the general acceptance and efficiency of learners of programming language. This would be achieved if the application is user friendly and intuitive enough for the users.

The features of the developed system are outlined below:

- *Welcome/home page*: This is a simple interface that display the name of the application to the user. This screen fades after the application is completely loaded into the mobile device memory.

- *Login*: This screen provides the platform for students to input their username so as to store their scores as they play.

- *The Option Screen:* This provides the navigational guide for the user of the application. Here, the user is presented with the options of either accessing the relevant Java programming resources or taking the quiz like programming challenge straightaway.

- *The Instruction Screen*: This screen provides instruction on how to interact with the application. It provides information such as: How to play the puzzle, How to reset the game and the implication of using the Reset option during the game session. Also on this page is the button to start the game.

- *The Question Screen*: The question screen serves as the challenge to be accomplished by the user. After reading this, the user will be well informed as to whether he has the knowledge to take on the quiz game or will need to search for more information relating to the task at hand.

4.1.4 Application Testing

The testing is after the development the application is already developed completely. The developed application is tested to see if the codes are working properly and check for error also get them fixed and the installation and debugging.

The developed application was tested using integration testing, where the application is installed and launched the individual units are joined together and tested as a group. The aim of the testing is to find errors in the interaction between units and the check functionality and reliability of the developed application.

4.1.4.1 Unit Testing

To understand the core mechanism of the learning application and confirm the operations are performed as expected. Unit testing is used to test numerous functions in the applications. In performing this test 5 random individuals volunteered to test each components of the application.

Table 4.2: The Integration Testing Results of the developed application

Test unit	Test objective	Test case	Input	Expected result	Actual output	Pass/ fail	Comments
Unit 1: login	Test if learners can login with username	Test case for adding username	Adding name	Added username	Added username	Pass	Login username was successful and displayed
Unit 2: learning resources	Test for the learning resources	Test case for searching learning resources	I want to learn more about Java	Show webpage tutorialpoints	Webpage showing tutorials	Pass	Searching learning resources successful
Unit 3: view instruction	Test for instructions	Test case for instruction to solve the test	Instruction	Instruction page	Instruction page	Pass	The instruction page successfully displayed
Unit 4: view test menu	Test for test menu page	Test case test menu page	Test menu	Test menu:3 stages with 3 levels	Test levels	pass	The test menu display successfully
Unit 5: solve test	Test for solve the test	Test case solve test	Beginner Level 1	Level 1 Question 1	Show test question	pass	Question successful displayed
Unit6:drag and drop	test if learner can drag and drop	test case drag and drop code	Drag and drop code	Drag and drop	Learner drag and drop	pass	The code drag and drop successful
Unit 5: show correct solution	Test if learners fail the correct solution shows	Test case show correct solution if fail	Failed test	Correctly arranged codes	Correctly arranged codes	Pass	Correct code show after failing successful

4.1.5 Maintenance

After the application is being tested, the developed application is checked for modification to make sure everything is working well after the development and the improve performance. The developed application is made to improve the learning knowledge of the learner. The maintenance is conducted to improve the application and also make corrections changing the errors from the developed application. After the unit testing was done on the developed application few features were added to improve the efficiencies of the application such as: adding more voice instructor, adding total score to the test pages so learners can find scores quick without going back to the menu page, on the score record the highest score text was made bold to be clear to the learners, the score record was added after each stages were completed from beginners to advance stage helps learners to rate their self after each stage is completed. In addition, a survey was conducted to test the Java Tutor mobile application and to also find out the learners opinions about the developed mobile application.

4.2 Research Schedule

The schedule has to do with the study period. The study start in September 2017 and was concluded in May 2018. The Gantt charts is used in the scheduling process and give detail on the appointment steps. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project.

ID	Task Mode	Task Name	Duration	Start	Finish
1		**Thesis research**	**173 days**	**Fri 9/1/17**	**Tue 5/1/18**
2		thesis proposal	20 days	Mon 8/21/17	Fri 9/15/1
3		chapter 2: literature review	7 days	Mon 10/2/17	Tue 10/10/
4		finalizing related works	4 days	Wed 10/11/17	Mon 10/16/
5		chapter 3: framework	7 days	Tue 10/17/17	Wed 10/25/
6		application developmen	30 days	Thu 10/26/17	Wed 12/6/1
7		chapter 4: developed mobile application	15 days	Thu 12/7/17	Wed 12/27/
8		developmen	16 days	Thu 12/2	Thu 1/
9		chapter 5: implementa	30 days	Fri 1/19/18	Thu 3/1/18
10		testing and maintenanc	17 days	Fri 3/2/18	Mon 3/26/1
11		review the thesis write up	12 days	Tue 3/27/18	Wed 4/11/1

Legend:

Task	Inactive Summary	External Tasks
Split	Manual Task	External Milestone
Milestone	Duration-only	Deadline
Summary	Manual Summary Rollup	Progress
Project Summary	Manual Summary	Manual Progress
Inactive Task	Start-only	
Inactive Milestone	Finish-only	

Project: Project1. thesis
Date: Wed 5/16/18

Page 1

(a) Gannt chart of thesis research schedule

49

ID	Task	Task Name	Duration	Start	Finish
2		preparation of	10 days	Thu 4/12/18	Wed 4/25/1
11		distributing questionnaries	2 days	Thu 4/26/18	Fri 4/27/1
12		Analysing survey	2 days	Sat 4/28/18	Mon 4/30/1
13		Jury defense	1 day	Tue 5/1/18	Tue 5/1/18

(b) Gannt chart of thesis research schedule

Figure 4.5: Gannt chart schedule of the develop application

4.3 The Developed System

The application is installed into an Android smartphone named Samsung galaxy S6 with an Android version 7.0 and the screenshot were taken from the smartphone.

4.3.1 Welcome Page

When the application starts, it launches the splash screen in which the name of the application appears on it. Figure 4.6 shows the startup page of the developed system.

Figure 4.6: Screenshot of the start page

4.3.2 Login Page

The login page is immediately after the splash screen on this page the learner will be required to input their information to be able to login as an existing user or signup as a new user the learner input a email, a username and password data will be stored in the cloud as the learner test and get their scores the cloud database store the score with using their usernames Figure 4.7 a, b, c shows the login and signup pages. When the store record pop-up the username of the learner and the score will appear with other learners score data.

(a) Screenshot of login page

(b) Screenshot of signing in page

(c) Screenshot of Signup page

(d) Screenshot of user invalid details.

Figure 4.7: Shows the screenshot of all the authentication patterns

- ***Invalid sign in:*** Figure 4.7 (d) shows the authentication page requires users that are ready signed up to provide a valid email and password which was used during the sign up process. If the user enter an incorrect string or number it the application can't be accessed and won't be logged in.

4.3.3 Introduction Page

The introduction page comes immediately after the learners are fully signed in, it basically gives a brief intro on the objective of the application. Figure 4.8 gives the learners hint on what the application is expected to achieve and what the entails.

Figure 4.8: Screenshot of the Introduction page

4.3.4 Home Page

The home screen is after the introduction screen, this is where the learner can get linked to the learning resources and also partake of the self-test to evaluate the knowledge on the subject on this page the leaner can find the logout option. Figure 4.9 shows the home page of the application with the link to the learning resources and test page also the logout link if done.

Figure 4.9: Screenshot of the home page

4.3.5 The Instruction Page

The instruction page follows immediately after the home page on this page are rules on how to go about the test. Figure 4.10 shows the instruction page where the learners get a tip on how to solve the exercises.

Figure 4.10: Screenshot of the instruction page

4.3.6 The Test Page

This test page comes right after the instruction page, on this page the player has 9 levels to complete from beginner to advanced level. The learner have to start from the beginner's level 1 before going into any other level. Figure 4.11 (a) shows the levels that are in the test and Figure 4.11 (b) shows the first test question.

   (a) Shows the stages   (b) Shows the test page

Figure 4.11: Screenshots of the test page and sample question

The learner can view score easy

In beginners level 1 the rewards are 5points each.

(a)Screenshot of beginner level question 1   (b) Score after solving question 1

Figure 4.12: Screenshots Showing beginner level 1 question 1 and score

- *Screenshot of test question:* Figure 4.12 a, b shows the test exercise which the learners have to solving to get their rewards. It also shows the rewards given after the exercise was solved correctly.
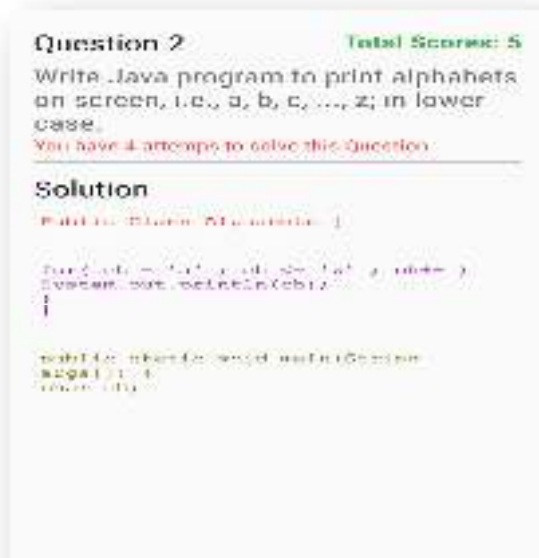
(a) Screenshot of Question 2 level 1   (b) Screenshot of score of Question 2

Figure 4.13: Screenshots showing beginner level 1 question 2 and score after solving

- Figure 4.13 a shows the screenshots of test page of question 2 level 1, it shows the questions and the solution which the learners have to arrange in order to answer. The figure 4.13 b shows the reward gained after getting the correct order in the solution.

(a) Screenshot of question 3 level 1   (b) Screenshot shows the rewards of question 3

Figure 4.14: Screenshots Showing beginner level 1 question 3 and rewards

- Figure 4.14 a shows the screenshots of test page of question 3 level 1, it shows the questions and the solution which the learners have to arrange in order to answer. The figure 4.14 b shows the reward gained after getting the correct order in the solution.

Question 1 Total Scores: 15

Write a Java program to print an American flag on the screen.

You have 4 attempts to solve this question

Solution

(a) Screenshot of question 1 level 2 (b) Screenshot of level 2 rewards

Figure 4.15: Screenshots showing beginner level 2 question 1 and reward

- Figure 4.15 a shows the screenshots of test page of question 1 level 2, it shows the questions and the solution which the learners have to arrange in order to answer. The figure 4.15 b shows the reward gained after getting the correct order in the solution.

(a) Screenshot of question 2 rewards   (b) Screenshot of question 3 level 2
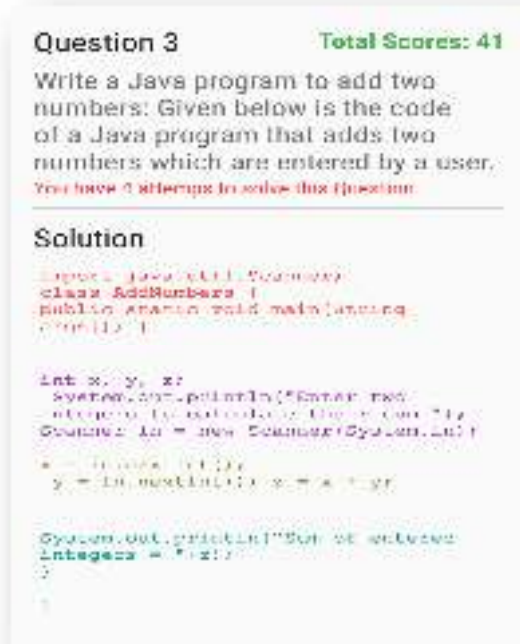
Figure 4.16: Screenshots showing score after solving question 2 and question 3 level 2

- The figure 4.16 a shows the reward gained after getting the correct order in the solution of question 2 level 2. Figure 4.16 b shows the screenshots of test page of question 3 level 2, it shows the questions and the solution which the learners have to arrange in order to answer
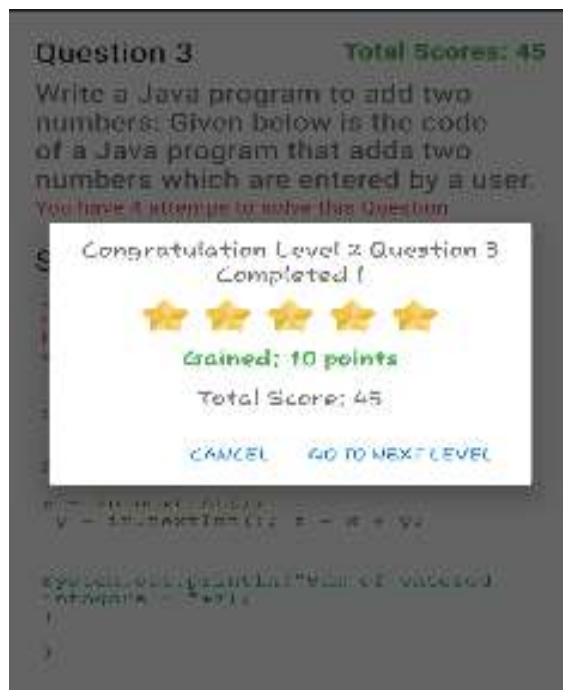
(a) Screenshot of question 3 rewards   (b) Screenshot of level 3 question 1

Figure 4.17: Screenshot**s** Showing the score after solving question 3 and question 1 level 3

- Figure 4.17 a shows the reward gained after getting the correct order in question 3 level 3. Figure 4.17 b shows the screenshots of test page of question 1 level 3, it shows the questions and the solution which the learners have to arrange in order to answer.

Level 3 rewards for each correct answer is 15points

(a) Screenshot of rewards question 2 level 3

(b) Screenshot of rewards level 3 question 3

Figure 4.18: Screenshots showing score after solving question 2 and question 3 level 3

(a) Screenshot of question 3 level 3
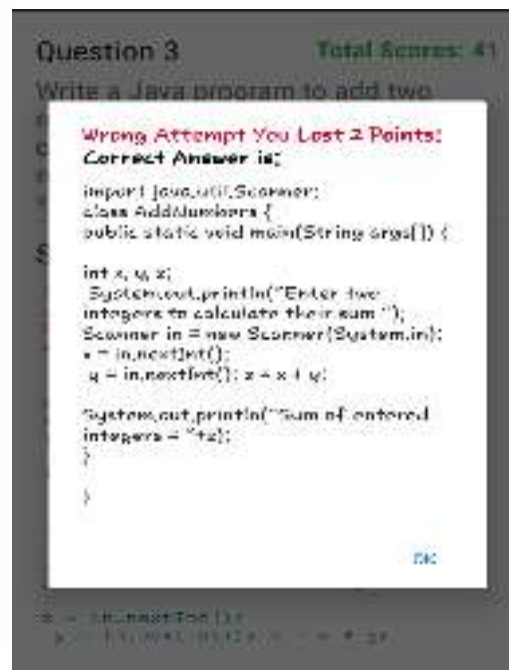
(b) Screenshot showing wrong attempt

Figure 4.19: Screenshots showing level 3 question 3 and the wrong attempt deduction

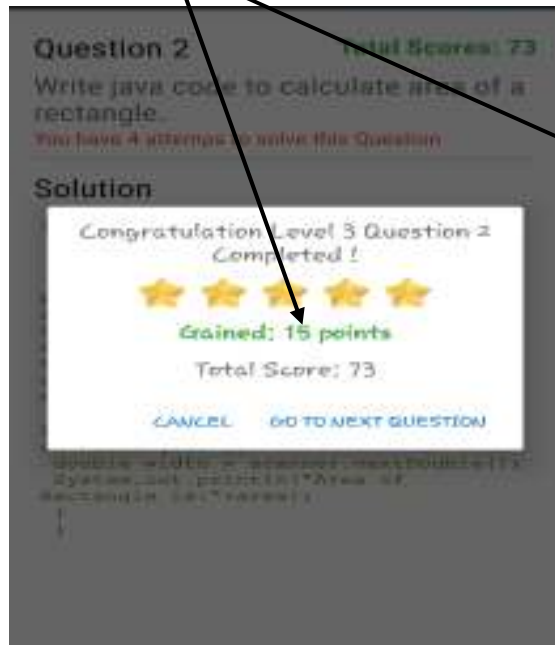- *Wrong attempts:* The learner is allowed only 4 attempts to answer one question. After the 4th attempt the learner loss 2 points from the gained points. Figure 4.19 (b) the learner is given chance to solve again after viewing the correct order in which the answer should be.

The learner with highest score is first on list with green colour

	Record	
1	Damm	Scores 255
2	Kashif	Scores 255
3	Shayan Malekloo	Scores 255
4	mo	Scores 170
5	Shylet	Scores 98
6	Ephraim	Scores 98
7	dam	Scores 88
8	miracle	Scores 88
9	Tatenda	Scores 76

	Record	
1	Damm	Scores 255
2	mo	Scores 170
3	Ephraim	Scores 98
4	dam	Scores 88
5	miracle	Scores 88
6	Adam	Scores 0
7	users	Scores 0
8	ssp	Scores 0
9	sheraz	Scores 0

(a) Screenshot of score record   (b) Screenshot showing the highest score

Figure 4.20: Screenshots showing the score record and the highest score

- *Score record:* Figure 4.20 shows the score record of the learners that used the application. The score and stored by the name and score on the list and the list ranges from highest to lowest. The highest learner get to be first on list and appears in green so the learners will know the difference.

64

CHAPTER 5

METHODOLOGY

5.1 Settings

This experimental study has been carried out at the Near East University, Department of Computer Information Systems during the Spring 2018 semester. In order to learn Java programming language, an Android based application has been developed by the author of the study called Java Tutor Self-test puzzle mobile application.

5.2 Participants

Questionnaires were distributed to the participant in order to get their opinion about the Java Tutor mobile application. There were about 32 random willing students that took part in the survey to use the application. There were 23 male in total and 9 females students of the Computer Information System department. Each participants are of different gender and nationality. Table 5.1 shows the frequency list of the participant's gender.

Table 5.1: Frequency of male and female participants

Gender	Frequency	Percent
Male	23	71.9
Female	9	28.1
Total	32	100.0

The Table 5.1 above shows the frequency of 23 males and 9 females with the percentage of to be 71.9% female 28.1%.

Table 5.2: Frequency list of the participants Nationality

Nationality	Frequency	Percent
Cyprus	2	6.3
Iranian	6	18.8
Libyan	3	9.4
Nigeria	11	34.4
Pakistan	3	9.4
Palestine	1	3.1
Zimbabwean	6	18.8
Total	32	100.0

The above Table 5.2 shows the participants based on their nationality with their frequency and the percent. The participant were from different nationality based on the percentage, Cyprus was 6.3%, Libyan was 9.4%, Nigeria was 34.4% participant was much higher than others, Pakistan was 9.4%, Palestine was 3.1%, Iranian and Zimbabwean was 18.8% had a same percentage of participants.

5.3 Data Collection Tools

To get an evaluation about the Java Tutor mobile application to find the effectiveness 32 students volunteer took part in the survey. The students were given the application to download and register as a learner and also take part in the test before given their opinions on the survey questionnaires. The questionnaire consist of three sections: Section 1 is about the demographic information, Section 2 is the general opinion section to determine if the application got positive response and opinion about the features and Section 3 is the recommendation towards the application to know what the students will advise to improve in the application. The survey was conducted in Near East University in the Computer Information System department.

Section 1: Demographic Information: This section the students are requested to filling information like gender, nationality and department.

Section 2: General Opinion: This section aims to investigate the participant's opinion about the developed application and its features. The section consist of 16 items aim to determine the opinion on the features and the applications in general, the questionnaires was developed by the author of this study after carefully reviewing the features and discussion made with the Thesis supervisor. The internal consistency of the questionnaire was found to be .92 using Cronbach Alpha which proves the questionnaires is consistent since α is greater than 0.70 (Sipahi, Turtkoru & Cinko, 2010), 16 questions were carefully prepared and given to the students to answer. The questionnaires was formed in 5-point likert scale type questions, consisting of 16 items, with 1 being a response to strongly disagree, 2: disagree, 3: Neutral, 4: agree and 5: strongly agree.

Section 3: Recommendation towards the Java Tutor Mobile Application: The aim of this section is for participant to give their recommendation about the Java Tutor mobile application. This section consists of 5 carefully prepared questions developed by the author of this study to find out what improvements or modifications the students suggest to be added into the Java Tutor mobile application. The question was in form of a Yes or No answers.

5.4 Data Analysis

In order to find out the opinions and recommendations of learners towards the Java Tutor mobile puzzle application, the questionnaires were used to collect the data, which was analyzed using SPSS. The questionnaires contained 21 questions which were prepared by the author of this study on developing Java Tutor mobile puzzle application to teach and learn Java programming languages. The Descriptive Analysis used to find the mean and standard deviation of the survey.

5.5 Procedure

This survey was carried out at Near East University at the Computer Information System department. The experimental study was carried out with 32 random volunteer's students and will to participate after explaining the objectives of the investigation which is for them to partake in testing the developed mobile application and get how or what they feel towards it. Before the investigation, the students were asked what type of mobile phone they used because the Java Tutor mobile application is developed for Android users. They were asked to install the Java Tutor mobile application on their Android mobile phones, register as a learner and partake in the test. The learners were given a week/ 7 days to use the developed application after a week, the questionnaires were prepared and shared to the students that have used the developed mobile application to find out their opinions and recommendations on the Java Tutor mobile application.

5.6 Results

Two major goals of the investigation study:

- To observe the students opinions about Java Tutor mobile application and obtain feedback.
- To obtain from the students their recommendations about the Java Tutor mobile application.

The results derived from the investigation are shown above:

A. *General opinions towards the use of the Java tutor mobile application*

According to the result from the survey given in Table 5.3 the students had very high positive opinion about the Java Tutor mobile application. The result of the mean item explained as follows: item 15 " *I found the scoring mechanism used in the application to spark the desire to be efficient and progress in the self-test*" Mean = 4.38 got the high opinion scale, item 10 "*I found the Java*

Tutor mobile application to be user friendly" Mean= 4.34, item 11 "*The interface of the Java Tutor mobile application is understandable*" Mean= 4.34   same mean because student find the application simple to understand and enjoyable to use and item 2 "*The Java Tutor mobile application was easy to use*" Mean= 4.31. This results were gotten because the students were satisfied with the application and efficient. Results on the opinion about the features: item 12 "*The voice instructor was really helpful when using the Java Tutor mobile application*" Mean = 4.00, item 13 "*I found the feedback in the Java Tutor mobile application were immediately displayed and really helpful*" Mean=4.16, item 14 "*I found the test rewards to be motivating to improve more on the solution*" Mean=4.22, item 16 "*I found the number of attempts given to me satisfying to answer the exercises*" Mean= 4.19 shows that student understood the aim of the features and it helped motivate and make the learning interactive to them in using the application.

Table 5.3: The scale of the learner's general opinion towards the use of the application

	Items	Mean	Std.D
1.	I found the Java Tutor mobile application enjoyable.	4.28	1.02
2.	The Java Tutor mobile application was easy to use.	4.31	.47
3.	The Java Tutor mobile application was flexible.	3.97	.69
4.	I learnt a lot about Java codes with the help of the Java Tutor mobile application.	3.84	1.05
5.	I found the Java Tutor mobile application used in learning java codes.	4.19	1.06
6.	I would like similar application used in learning other courses.	3.78	1.36
7.	Using the Java Tutor mobile application with the drag and drop technology has motivated me.	4.25	1.19
8.	I will recommend the Java Tutor mobile application to my friends wish to learn Java.	4.25	.92
9.	I found the Java Tutor mobile application easier than the normal teaching method.	4.03	1.03
10.	I found the Java Tutor mobile application to be user friendly.	4.34	.87
11.	The interface of the Java Tutor mobile application is understandable.	4.34	.55
12.	The voice instructor was really helpful when using the Java Tutor mobile application.	4.00	1.19
13.	I found the feedback in the Java Tutor mobile application were immediately displayed and really helpful.	4.16	.72
14.	I found the test rewards to be motivating to improve more on the solution.	4.22	.91
15.	I found the scoring mechanism used in the application to spark the desire to be efficient and progress in the self-test.	4.38	.75
16.	I found the number of attempts given to me satisfying to answer the exercises.	4.19	.82
	Average mean and std.deviation	4.16	0.91

Note: Scoring: 5= Strongly agree, 1= Strongly disagree.

B. *General recommendation towards the Java Tutor mobile application*

It's simply their opinion on what they think should be added to the application and how it help them. We have asked some questions to determine the student's recommendation on the application. In response to the question and results "*In your opinion do you think you can write a simple Java code after using the Java Tutor mobile application?*", the majority of students 59.4% picked YES which implies the students on average feel they can write a simple code in Java after using the developed application, "*In opinion do you think the application need more animations*", the majority 65.5% picked YES which hint at more than half the students feel there should be more animation added to the developed application so as to increase the fun, "*In your opinion do you think the learning resources were helpful in understanding different programming languages?*", 71.9% picked YES which huge number of students during the survey really enjoyed using the learning resources to help understand programming languages, "*What do you rate your Java programming skill after using Java Tutor mobile application?*", 71.9% picked average which means the students feel that after using the developed application they can understand Java programming on an average level is a learning process and "*Would you like the Java Tutor mobile application to be supported by chat and forums?*", 71.9% picked YES which means the student feel there should be a form of communication with the other students.

5.7 Discussion

The main of the discussion is to decipher and explain the importance of the discoveries in light of what was previously known about the research problem being investigated. The primary focus of the study is to develop a self-test mobile application for learning and teaching Java programming languages using puzzle, if the puzzle is suitable for learning Java programming language. In the earlier researches viewed suggest that there are benefits to using puzzle strategy for learning Java like increasing learning capacity also linked to development of professional skills (Falkner, 2010). Few studies showed that puzzle strategy could be an entertaining platform because puzzle is actually associated with fun and recreation (Michalewicz, 2008).

In a study found by Sergiy (2017), about the influence of using puzzle as an instructional strategy to teaching and learning engineering and mathematics. Almost all students (98%) believed that solving puzzles enhances their problem-solving skills. The huge majority of the

students stated views on the difference between a puzzle and a routine problem similar to the authors of books/articles on puzzle-based learning. Frederic et al. (2012) develop a game using puzzle strategy to help learners in understanding programming languages like C++, C and pascal. In the study it gives challenges to be solved using codes to solve also using video game platform. Another study Tillmann et al. (2013) developed a game codehunt uses a test/clue generator of Pex, the Pex generates possible solution to the test puzzle. The application helps learners to learn Java and C++ by write codes to continue in levels.

In this study to increase learners ability to learn using self-test puzzle, it strategy is using the drag and drop technology, also it includes an audio instructor and it's mainly to teach Java programming language on the introductory concept.

The Table 5.4 shows the difference between the new learning application and the applications mentioned in the literature. Each functionalities are marked differently the symbol √ means the function is available in the application and if **X** means the function is not available.

Table 5.4: Difference between new learning application Java Tutor mobile application and the learning applications in literature studied

Learning application / Functionality	Prog & play	Bomberman	IRPG	Codin game	Iplaycode	Pexfun	Java Tutor
Self-explanation	X	X	✓	X	✓	✓	✓
Cloud based	✓	X	X	X	X	✓	✓
Immediate feedback	✓	✓	✓	X	✓	X	✓
Drag & drop technology	X	X	✓	X	X	X	✓
Voice helper	X	X	X	X	X	X	✓

The Table 5.4 indicates that Java Tutor mobile application contains majority of functionalities. Java Tutor use the cloud based database as a functionalities to help student sign in to their workspace and also view their performance on each levels this helps to boost their confidence level. The Java Tutor mobile application has got immediate feedback that aids learners and task them to build program properly and acquire good score. In Java Tutor mobile application the feedbacks are given after 4 attempts have been made. Also the Java Tutor mobile application combines the use of the drag and drop, self-explanation, and voice helper but in the other learning application the functions are either absent or limited.

Table 5.5: List of applications platforms and their features for programming languages

Authors/ Developers	Name of games	Instructional strategies	Programming Language	Database/ programming language design	Content	Platform
IBM (2001)	Robocode	Constructivism	Java &.Net	Java based, intelliJ Idea, NetBeans & visual studio	Animation	Web based
Muratet et al. (2010)	Prog & play	Constructivism	Ada, C, Java	Real time strategy	Animation	Mobile application
Frederic Desmoulins et al. (2012)	Codın Game	Role playing	C, c++, c#, Java, Javascript, pascal, perl, php, python, ruby	Java,sql	Animation,Puzzle	Web based, Mobile application **(IOS)**
Ocay et al. (2013)	JavaLite	Experimental	Object oriented programming	Java, Sql lite	Quiz, debugging game, word twist, lesson notes	Mobile application **(ANDROID)**
Microsoft research. Tillmann et al. (2013)	Pex4fun	Experimental	C#, visual basic, f#	Cloud application, Pex generator	Puzzle generator	Web based
Tillmann et al. (2014)	Code Hunt	Experimental	C#, Java	Microsoft cloud	Puzzle, animation	Web based
Ragab Ihnissi & Joan Lu (2015)	Iplaycode	Direct instructions	Java, C, C#, C++,	Xcode 5.0.2, adobe Photoshop IOS	Quiz challenge	Mobile application **(IOS)**
Under development (2018)	Self-test puzzle	Learning by doing	Java	Java, firebase database, Android studio	Puzzle challenge, voice instructor, Rewards, drag and drop method, learning resources	Mobile application **(ANDROID)**

Table 5.5 gives a description on various applications for learning programming till date and their features: the learning strategy adopted, the type of programming languages being taught and the learning content.

CHAPTER 6

CONCLUSION AND RECOMMENDATION

6.1 Conclusion

The survey done on the Java Tutor mobile application had a positive result and in support of the developed application. The participant communicated their satisfaction and pleasure of learning using the mobile application. The approach of the develop Java Tutor mobile application is very suitable approach that encourages and motivates learners because of its puzzle drag and drop strategy combined with the audio and selectivity Java programming language which make this self-test mobile application puzzle unique, motivating and engaging for interested learner at the introductory level. Self-test puzzle generate a practicable educational context which additional improves interactive learning. It promotes the performance of learning and make it entrancing and fun.

The aim of the study has been achieved and the developed Java Tutor mobile application can be used anytime and anywhere to learn Java programming language. Furthermore, in the result of Java Tutor mobile application after being used by the student's shows that it was user friendly and entrancing to use. The Java Tutor mobile application will be introduced in the next Fall Term, Computer information System department during the Java course as a learning tool for the students to aid engage and motivate their learning skill.

6.2 Recommendations

The general goal of this study has been to add to using mobile learning anytime and anyplace and also help motivate students in learning Java programming language. Even though the teaching of Java programming using the puzzle drag and drop technology has been successful, the developed mobile application can be improved in several ways:

- The developed mobile application can also be used a learning tool for instructor in the classroom for teaching.
- I will recommend that more questions should be add to the self-test in order to efficiency improve the learners ability.

- In the result of the survey the student also 65.5% recommended that the application requires more animations, the learners feel by adding more animations it will make the developed mobile application more relaxing and 71.9% of the learners thinks chat and forums included in it to help the students communicate with each other about the scores and their performances.

C. The developed Android self-test puzzle was based on just Java programming language, therefore is a need for other programming languages that teaches the basics such as C++, C, Python etc.

REFERENCES

Aaby, Anthony (2004). Introduction to Programming Languages. *ACM SIGCSE Bulletin. In Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education.* (pp.38-51), California. U.S.A.

Adams, E (2005). The Designer's Notebook: Educational games Dnt have to stink! Retrieved May 3, 2005 form http://www.gamasutra.com/view/feature/2190/the_desginers_notebook_php/

Andrei, F (2014). A closer look at Android runtime (ART) in Android L. Retrieved July 5, 2014 from http://www.anandtech.com/show/8231/a-closer-look-at -android-runtime-art-in-andorid

Anastasios A., Economides R., & Nick Nikolaou., T (2005). Evaluation of hand held devices for mobile learning. *International Journal of Engineering Education, 3(1),* 34-23.

Allen, R.K., Bluff, K. & Oppenheim, A. (1998). Jumping into Java: Object-oriented software development for the masses. *In Proceedings of the 3rd Australasian Conference on Computer Science Education,* (pp. 165-172). Brisbane, Australia.

Alanne A., & Kari Y. (2015). An overview of game-based learning in building services engineering education*, European Journal of Engineering Education. 6(7).* 45-50.

Ala-Mutka, K., (2004).Problem in learning teaching programming. *Education and Information Technologies, 7(1),* 55-66.

Alabaster, Jay (2013). Android founder: we aimed to make a camera OS. PC World. International Data Group. Retrieved May 9, 2017 from http://www.androidfounder.com/view/feature

Alaeeddine, Y., Driss, C., Mohammed, B., & Ahmed, T., (2017). A serious game for learning C programming language concepts using solo taxonomy. *International Journal of Education Technology. 12(3).* 23-34.

Barab, S., Scott, B., Siyahhan, S., Goldstone, R., Ingram-Goble, A., Zuiker, S., & Warren, S. (2009). Transformational play as a curricular scaffold: Using videogames to support science education. *International Journal of Science Education Technology, 18,* 305-320.

Barab, S.A., & Dede, C. (2007). Games and immersive participatory simulations for science education: An emerging type of curricula. *Journal of Science Education and Technology*, *16(1),* 1-3.

Bodner, G. M. (1986*).* Constructivism: A theory of knowledge. *Journal of Chemical Education, 63,* 873-878.

Bonar, J., & Soloway, E. (1989). Preprogramming knowledge: A major source of misconceptions in novice programmers. *In E. Soloway & J.C. Spohrer, Studying the Novice Programmer.* (pp. 324–353*).*

Bierre, K., Ventura, P., Phelps, A. & Egert, C. (2006). Motivating OOP by blowing things up: An exercise in cooperation and competition in an introductory Java-programming course. *In Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education.* (pp. 38-45).

Bonakdarian, E. & White, L. (2004). Robocode throughout the curriculum. *Journal of Computing Science in Colleges, 19(3),* 12-15.

Badger, M., Sangwin, C.J., Ventura-Medina, E. and Thomas, C.R. (2012) A guide to puzzle-based learning in STEM subjects. University of Birmingham. Retrieved November 1, 2013 from http://www.heacademy.ac.uk/resources/detail/disciplines/stem/guide_to_PzBL

Bushnell, N. (2009). Presentation at Game Based Learning. *In Proceeding of the International Conference on Software and Engineering.* (pp. 19–20). London.

Bishop, J., Horspool, R., Xie, T., Tillmann, N., & de Halleux, J., (2015). Code Hunt: Experience with coding contests at scale*, In Proceeding of the International Computer Science and Engineering.* (pp. 122-125). Finland.

Brooks, R.E. (1990). Categories of programming knowledge and their application. *International Journal of Man-Machine Studies, 33,* 241–246.

Burgstahler, S. (2001). Equal access: Universal design of instruction methods. Do-IT, University of Washington. Retrieved April, 2001 from http://www.washington.edu/doit/Brochures/PDF/equal_access_udi.pdf

Burgstahler, S. (2005). Universal Design of Instruction: Definition, Principles, and Examples. Retrieved April 13, 2005 from http://www.washington.edu/doit/Brochures/Academics/instruction.html

Cavus, N, & Ibrahim, D. (2009). M-learning: An experiment in using SMS to support learning new English Language words. *British Journal of Educational Technology. 40(1),* 78-91.

Chen, C., Wang, K & Lin, Y. (2015). The Comparison of Solitary and Collaborative Modes of Game-Based Learning on Student's Science Learning and Motivation. *Journal of Educational Technology & Society, 8(7),* 12-22.

David A. Schmidt, (1994). *The structure of typed programming languages, Massachusetts Institute of Technology Press* (pp. 32-36).

Depradine, C.A (2011). Using Gaming to Improve Advanced Programming Skills. *Journal of Caribbean Teaching Scholar. 1(2),* pp. 93–113.

Deterding, S., Khaled, R., Nacke, L., & Dixon, D. (2011). Gamification: towards a definition. Conference: CHI 2011 Gamification workshop. *In Proceedings of the International Computer Human Interaction Education.* (pp. 34-38). Boston.

Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining gamifications. *Journal of the Educational Technology & Society. 3(2).* 56-58.

Denning, P.J. (2000). Computer Science: The Discipline. Encyclopaedia of Computer Science. Retrieved May 25, 2006 https://www.computer.science.the discipline.com/view/documents

De Bono, E. (1967) New think: the use of lateral thinking. London: Jonathan Cape. *Journal of the Engineering Education. 8(4).* 60-65.

Din, H.W. (2006). Play to Learn: Exploring Online Education Games in Museums. In: *International Conference on Computer Graphics and Interactive Techniques, 13 (*pp.45-48). Boston, Massachusetts.

Deek, F.P., Kimmel, H., & McHugh, J.A. (1998). Pedagogical changes in the delivery of the first-course in Computer Science: Problem solving, then programming. *Journal of Engineering Education, 87,* 313–320.

Decker, R. and Hirshfield, S. (1994): The Top 10 Reasons why object-oriented programming cant be taught in CSI. *In Proceeding of the twenty-fifth SIGCSE Symposium Computer Science Education.* (pp. 51-55). Arizona, USA.

Duke, R., Slzman, E., Buemeister, J., Poon, J. and Murray, L. (2000): teaching programming to beginners choosing the language is just the first step. *In Proceedings of the Australasian Conference on Computing Education,* (pp. 79-86). Melbourne Australia.

Dochev, D., & Histrov, I., (2007), Mobile Learning Applications: Ubiquitous Characteristics and Technological Solutions, Retrieved June 20, 2007 from http://www.scrbd.com/mobile learning

Eagle, M., & Tiffany B. (2009). Experimental evaluation of an educational game for improved learning in introductory computing. *In Proceeding of the International of the Computer Science Education. 41,* 321.

Egenfeldt-Nielsen, S. (2007). Educational Potential of Computer Games. *Continuum International Journal of the Education Technology. 5(1),* 70-79.

Egenfeldt-Nielsen, S. (2006). Overview of research on the educational use of video games. Digital Kompetanse. *International Journal of the Education Technology. Vol. 1*, p.184 - 213.

Elisa, T. (2010). Universal instructional design principles for moodle. International review of research in open and distance learning. Retrieved July 18, 2010 from http://www.irrodl.org/index.php/irrodi/article/view/869

Elisa, T. (2011). Universal instructional design principles for mobile learning. International review of research in open and distance learning. Retrieved March 12, 2011 from http://www.irrodl.org/index.php/irrodi/article/view/965/1675

Eric, S. (2010). Flurry: 22% of Recent Mobile Applications Starts Targeting iPad. Retrieved Friday April 02, 2010 from http://www.macrumors.com/2010/04/02/flurry-22

Falkner N., Sooriamurthi R., & Michalewicz Z. (2012). Teaching puzzle-based learning: development of basic concepts. *Teaching Mathematics Computer Science.10 (1)*.183–204.

Fathi E, Leila J., Ben A., & Mohamed J. (2007). A MultiParameters Personalization Approach of LearningScenario. *The Proceeding of the 7th International Engineering and Education.* (pp. 5-7). Nigata, Japan.

Fairclough, C., & Cunningham, P. (2003). A multiplayer case based story engine. *In Proceedings of the 4th International Conference on Intelligent Games and Simulation* (GAME-ON), pp. 41-46.

Feldgen, M., & Clua, O. (2004). Games as a Motivation for Freshman to Learn Programming. *In: 34th International Journal of the Engineering Education Conference, Vol. 3,* pp.11–16.

Fingas, J. (2012). ComScore: Android tops 52 percent of US smartphone share, iPhone cracks the 33 percent mark. Retrieved March 11, 2017 from https://www.engadget.com/2012/09/04/comscore-android-tops-52-percent-of-us-smartphone-share/

Franzwa, C., Tang, Y., & Johnson, A., (2013). "Serious Game Design: Motivating Students through a Balance of Fun and Learning," Games and Virtual Worlds for Serious Applications. *The 5th International Journal of the Engineering and Education Conference*, *6(4).* pp. 1-7.

Garris, R., Ahlers, R., & Driskell, J. E. (2002). Games, motivation, and learning: a research and practice model. Simulation & Gaming. *International Journal of the Engineering and Education Conference*, *33*, 441–467.

Gundotra, V. (2013). Just back from a whirlwind trip to Asia visiting our…". Plus.google.com. Retrieved September 3, 2013 from http://www.plus.google.com/

Gee, J. P. (2008). Learning and Games. The Ecology of Games: Connecting Youth, Games, and Learning. *Massachusetts Institute of Technology Press. 5,* (pp. 21 - 40).

Hill, J., Ray, C., Blair, J & Caver, C. (2003): Puzzles and games: addressing different learning styles in teaching operating systems concepts. *In Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education,* pp. 23-35. Reno, Nevada, USA.

Hu, Minjie (2004): Teaching Novices programming with core language and Dynamic visualisation. *In Proceedings of the NACCQ*, (pp. 2-6). Christchurch, New Zealand.

Honey, M. A. & Hilton, M. L. (Eds). (2011). Learning science through computer games and simulations. Committee on Science Learning: Computer Games, Simulations, and Education. Board on Science Education, *Division of Behavioral and Social Sciences and Education.* (pp.34-50). Washington, D.C

Hanson, C.W. (2011). Chapter 2: Mobile devices in 2011. *Journal of the Library technology Reports. 47 (2).* 11-23.

Herz, J. (1997). Joystick Nation: How Videogames Ate Our Quarters, Won Our Hearts, and Rewired Our Minds, Little Brown & Company. Retrieved May 2, 1997 from http://www.videogames,joysticknation/games

Hwang, G. J., Sung, H. Y., Hung, C. M., & Huang, I. (2012a). Development of a personalized educational computer game based on students' learning styles. *Educational Technology Research and Development, 60(4),* 623–638.

Jaffar, A. (2016). The importance of computer programming. Retrieved April 5, 2016. From http;//www.linkedin.com

Jiayi, Z., & Joan, L. (2014). Using mobile serious games for learning Programming. *The Fourth International Conference on Advanced Communications and Computation. 6,* 4-12.

Khenissi, F.,Essalmi, &M. Jemni, (2013). Presentation of Learning Game for Programming Languages Education, Advanced Learning Technologies (ICALT). *IEEE 13thInternational Conference on. Information Engineering Education.* (pp. 324-326).

Kirriemuir, J., & McFalance, A. (2004). Literature review in games and learning. Report 8. Bristol: NESTA, Futurelab. *International Journal of the Technology of the Education.* 70-77.

Kirikkaya, E. B., Iseri, S., & Vurkaya, G. (2010). A board game about space and solar system for primary school students. *Turkish Online Journal of Education Technology, 9*(2) 1-13.

Kendrick, J. (2013). Mobile Technology: The Amazing Impact On Our Live. Retrieved April 30, 2013 from zdnet.com: http://www.google.com.cy/amp/ww.zdnet.com/mobile technology

Kolb, D. (1984). Experiential learning: Experience as the source of learning and development. *Creative Education. 5(12).* 45-48.

Ludovic, B., Pol-Benard, G., Carl-Philippe, R., & Safouana, T. (2005). Learning by doing: a teaching method for active learning in scientific graduate education. *European Journal of Engineering Education, 30(1),* 105-119.

Liu, M., (2014). Motivating Students to Learn Using a Game-Based Approach: Gaming and Education. Retrieved June 2014 from http://www.edb.utexas.edu/liu/files/Liu_Gaming-andEducation.pdf

Lee, N, (2006), Beyond Barbie and MortalKombat: new perspectives on gender, games and computing. *ACM Computers in Entertainment, 4(2).* 50-57.

Long J. (2007). Just fun: using programming games in software programming training and education. A field study of IBM Robocode community. *Journal of Information Technology Education. Vol (6).* 67-77.

Lahtinen, E., Ala-Mutka, K., &Järvinen, H., (2005) A Study of the Difficulties of Novice Programmers. *In Proceedings of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education.* pp. 14–18. New-York. U.K.

Michalewicz, Z. & Michalewicz, M. (2008) Puzzle-based learning: introduction to critical thinking, mathematics, and problem solving. *International Journal of The Mathematics Education And Science Technology. 34(2).* 55-67

Marzocchi, AS. (2012). Using the Tower of Hanoi puzzle to infuse your mathematics classroom with Computer Science concepts. *International Journal of the Mathematics Education and Science Technology. 47(5).* pp 814–821.

Maloney, J., Burd, L., Kafai, Y., Rust, N., silverman, B., & Ressnick, M. (2004). A sneak preview. *In 2nd International Conference on Creating Connecting, and Collaborating Through Computing.* (pp. 104-109).

Meyer & David (2010). Linux developer explains Android kernel code removal. *ZDNet*. CBS Interactive. Retrieved March 11, 2017 from https://www.zdnet.com/article/linux-developer-explains-android-kernel-code-removal/

Miangah, T. M. & Nezarat, A. (2012). Mobile-assisted language learning. *International Journal of Distributed and Parallel Systems, 3(1),* 309-319.

Miroslav, M., Miloš, M,. & Dusan, S. (2013). Game based learning and mobile learning. *International Journal of the Education Technology*. (pp. 45-55)

Muratet, M., Torguet, P., Viallet, F., & Jessel, J. (2010). Experimental feedback on Prog&Play: a serious game for programming practice. *International Journal of the Education Technology*. *3(1).* (pp. 35-55)

Malone, T. W., & Lepper, M. R. (1987). Making Learning Fun: A Taxonomy of Intrinsic Motivations for Learning. Aptitude, Learning, and Instruction. *Conative and Affective Process Analyses. 3,* 34-37.

Macwan, U. (2017). The benefit of mobile technology in education. Retrieved March 8, 2017 from finextra: http:/www.finextra.com/blogposting/the-benefits-of-mobile-technology-in-edcuation

Ngai, E., & Gunasekaran, A. (2007) A review for mobile commerce research and applications. Decision Support Systems. *In Proceedings of the Conference on Computing Education*. (pp.3-15).

Ozdemli, F, & Cavus, N. (2011). Basic elements and characteristics of mobile learning. *Procedia-Social and Behavioral Science*, (pp. 937-942).

Ocay, A., Gallares, R., & Agas, K. (2016). JavaLite: A mobile game based learning application for Object oriented programming. In *Proceedings of the 8th Australasian Conference on Computing Education*. (pp 222-244).

Parsons, D. and Haden, P. (2006). Parson's programming puzzles: a fun and effective learning tool for first programming courses. *In Proceedings of the 8th Australasian Conference on Computing Education.* (pp.157-163). Australia.

Pierce & Benjamin. (2002). Types and Programming Languages. *In Proceeding of the Computer & Informatics.* pp. 339. New York.

Plass, J. L., Perlin, K., & Nordlinger, J. (2010). The games for learning institute: Research on design patterns for effective educational games. *Paper presented at the Game Developers Conference, San Francisco.* (pp. 45-50), CA.

Prensky M. (2001).Digital game based learning. *Paper presented at the Game Developers Conference,* (67-69). San Francisco, CA.

Prensky, M. (2004). Presentation at NCBL eLearning Summit. Retrieved October 5, 2005 from http://www.netbtechsummits.org/summits2/presentations/5.3.prensky.pdf

Prensky, M. (2004). Digital Game-Based Learning. . *Paper presented at the Game Developers Conference,* pp (3- 8). San Francisco, CA.

Prensky, M. (2006). Don't bother me, mom, I'm learning! How computer and video games are preparing your kids for 21st century success and how you can help. *International Journal Of Human-Computer Studies.* pp (10-12).

Priya, V. (2017). What is a mobile devices. Retrieved October 10, 2017. From https://www.lifewire.com/what-is-a-mobile-device/

Qian, Meihua, and Karen R. Clark (2016). Game based Learning and 21st century skills: A review of recent research. *International Journal of the Computers in Human Behavior,* pp (12 -15).

Roslina I., Rasimah C. M. Yusoff, Hasiah M. Omar & Azizah Jaafar (2011). Students Perceptions of Using Educational Games to Learn Introductory Programming. *International Journal of the Computer and Information Science.* Pp (35-45).

Roslina, I., & Azizah, J. (2009). Educational Games Design Framework: Combination of Game Design, Pedagogy and Content Modelling. *International Conference on Electrical Engineering and Informatics. 6(7).* 56-59.

Roslina, I., & Nazli, Y. (2009). Development and effectiveness of educational games for learning Introductory programming*. Curriculum Teaching and Learning Research Report.* 4(3). 67-69. Malaysia, Skudai, Johor.

Roslina, I., Rasimah, C. M. Y., & Azizah, J. (2008). Computer Games Playing Activities: Habits of Universiti Teknologi Malaysia Student. *Paper presented at the International Conference on IT and Multimedia (*ICIMU), pp. 45-50. Bangi, Selangor

Reiber, L. P., Smith, L. & Noah, D. (1998). The value of serious play*. Educational Technology, 38(6),* 29–37.

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

Rountree, N., Rountree, J., & Robins, A. (2002). Identifying the danger zones: Predictors of success and failure in a CS1 course. *In International Journal of the Education Technology. 34*, 121–124.

Simon, K (2017). Digital in 2017: Global overview. Retrieved 24 January 2017. https://wearesocial.com/special-reports/digital-in-2017-global-overview.

Salen, K. & Zimmerman, E. (2004). *Rules of Play: Game Design Fundamentals. The Masschusetts Institute of Technology Press.* (pp 23-27).

Sedighian, A. & Sedighian K. (1996*).* Can Educational Computer Games Help Educators learn About The Psychology of Learning mathematics in Children? *18th Annual Meeting of the International Group for the Psychology of Mathematics Education,* (pp 23-26). *F*lorida, USA.

Shaffer, D. W., Squire, K. R., Halverson, R. & Gee, J. P. (2005). Video games and the future of learning. *The 10th European Conference on Games Based Learning. 87(2),* 104-111.

Sheil, B. (1980). Teaching Procedural Literacy. *In Proceedings of the ACM 1980 Annual Conference.* (pp. 125 – 126). New York.

Sinicki & Adam (2016). Developing for Android vs developing for iOS – in 5 rounds . *Android Authority.* Retrieved June 21, 2017 from https://www.androidauthority.com/developing-for-android-vs-ios-697304/

Soloway, E., Ehrlich, K., Bonar, J., & Greenspan, J. (1983). What do novices know about programming? *In B. Shneiderman & A. Badre (Eds.), Directions in Human-Computer Interactions. (*pp. 27–54).

Shaffer, D. W., Halverson, R., Squire, K. R., & Gee, J. P. (2005). Video games and the future of learning. Madison: University of Wisconsin–Madison, *Wisconsin Centre for Education Research*. Pp. 56-58.

Stein, Lynn Andrea (1998): What we have swept under the rug: Radically rethinking CSI. *Computer Science* education 8(2): 118-129.

Siti, H., Hamid, A., & Leong Yu, F. (2007). Learn Programming by using mobile Edutainment game Approach. *The First IEEE International Workshop on Digital Game and Intelligent Toy Enhanced Learning.* 5(7) 56-59.

Stallman & Richard (2011). Is Android really free software? . London: The Guardian. Retrieved September 9, 2012 from https://www.theguardian.com/technology/2011/sep/19/android-free-software-stallman

StatCounter"StatCounter. (2016). Global Stats - Browser, OS, Search Engine including Mobile Usage Share". Retrieved November 30, 2016 from https://www.zdnet.com/article/mobile-and-tablet-internet-usage-exceeds-desktop-for-first-time-statcounter/

Thomas C., Badger M., & Esther Ventura-Medina E, (2013). Puzzle-based learning of mathematics in engineering. *Journal of the Engineering Education. 8(1):*122–134

Tillmann, N., de Halleux, J., Xie, T., & Bishop, J. (2014). Code hunt: gamifying teaching and learning of Computer Science at scale. *In: Proceedings of the First ACM Conference on Learning*. 221–222.

Tillmann, N., De Halleux, J., Xie, T., SumitGulwani, L., & Bishop, J. (2013). Teaching and learning programming and software engineering via interactive gaming. In: 35th *International Conference on Software Engineering* (ICSE). 1117–1126, San Francisco, CA,

Ulicsak, M. (2010). Games in Education: Serious Games Context. *International Journal of The Computer Education.* (pp. 50-57).

Vangie, B. (2011). Mobile operating system. Retrieved August 10, 2011. From https://www.webopedia.com/DidYouKnow/Hardware_Software/mobile-operating-systems-mobile-os-explained.html

Valen, J. (2012). *Mobile-Based Teaching Resource Management System* for Urdaneta City University. A master's project study, University of the Cordilleras, (pp. 30-35). Baguio City.

Villalta, M., Gajardo, I., Nussbaum, M., Andreu, J. J., Echeverrı, A., & Plass, J. (2011). Design guidelines for classroom multiplayer presential games (CMPG*). Computers & Education, 57,* 2039–2053.

Vos, N., van der Meijden, H. & Denessen, E. (2011). Effects of constructing versus playing an educational game on student motivation and deep learning strategy use. *Computers & Education, 56,* 127–137.

Wen-Chih C., Yu-Min C., Kuen-Chi C., (2011).Game-based Collaborative Learning System. *Journal of Convergence Information Technology.* (pp. 34-37).

Wai-Tak, W., & Yu-Min, C., (2007). An interactive Bomberman Game based teaching/ learning tool for introductory C programming. *International Conference on Technologies for E-learning and Digital Edutainment*. Pp 433-444.

Winslow, L.E. (1996). Programming pedagogy – A psychological Overview. *In The Proceedings of the International Conference on Engineering & MIS, 28*, 17-22.

Yarow. J (2014). This Chart Shows Google's Incredible Domination Of The World's Computing Platforms . Retrieved March 23, 2014 from http://www.businessinsider.com/androids-share-of-the-computing-market-2014-3

Ziegler & Chris (2010). Microsoft talks windows phone 7 series development ahead of GDC: Silverlight, XNA, and no backward compatibility. Retrieved October 27, 2011 from https://www.engadget.com/2010/03/04/microsoft-talks-windows-phone-7-series-development-ahead-of-gdc/

APPENDICES

APPENDIX A

QUESTIONNAIRE

INVESTIGATING THE OPINIONS OF LEARNERS ABOUT THE JAVA TUTOR MOBILE APPLICATION DEVELOPED TO TEACH AND LEARN JAVA PROGRAMMING LANGUAGE

As a part of the Master thesis research at the Near East University, I am conducting a survey that investigate the attitude of learners towards the use of the Java Tutor mobile application for learning Java programming language. You are expected to choose the most appropriate options applicable to you. Any information obtained will be solely used for analysis in the research report and will remain confidential.

LAWAL MODUPE DAMILOLA (MASTERS STUDENT)

SECTION 1: DEMOGRAPHIC INFORMATION

1. Gender   a. Male   b. Female

2. Nationality   ………………………..

3. Department …………………………

SECTION 2: GENERAL OPINION TOWARDS THE USE OF THE JAVA TUTOR MOBILE APPLICATION

ITEMS	Strongly agree	Agree	Neutral	Disagree	Strong Disagree
1. I found the Java Tutor mobile application enjoyable.					
2. The Java Tutor mobile application was easy to use.					
3. The Java Tutor mobile application was flexible					
4. I learnt a lot about Java codes with the help of the Java Tutor mobile application.					
5. I found the Java Tutor mobile application very useful for learning Java codes.					
6. I would like similar applications used in learning other courses.					
7. Using the Java Tutor mobile application with the drag & drop technology has motivated me.					
8. I will recommend the Java Tutor mobile application to my friends who wish to learn Java.					
9. I found the use of Java Tutor mobile application easier than the normal teaching method.					
10. I found the Java Tutor mobile application to be user friendly.					
11. The interface of the Java Tutor mobile application is understandable.					
12. The voice instructor was really helpful when using the Java Tutor mobile application.					
13. I found the feedback in the Java Tutor mobile application were immediately displayed and really helpful.					
14. I found the test rewards to be motivating to improve more on the solutions.					
15. I found the scoring mechanism used in the application to spark the desire to efficient and progress in the self-test.					
16. I found the number of attempts given to me satisfying to answer the exercises.					

SECTION 3: RECOMMENDATIONS TOWARDS THE JAVA TUTOR MOBILE APPLICATION

17. In your opinion do you think you can write a simple Java code after using the Java Tutor mobile application?
    a. Yes   b. No

18. In your opinion do you think the application need more animations?
    a. Yes   b. No

19. In your opinion do you think the learning resources were helpful in understanding different programming languages?
    a. Yes   b. No

20. What do you rate your Java programming skill after using Java Tutor mobile application?

    a.   Very little   b. Average   c. Above average

21. Would you like the Java Tutor mobile application to be supported by chat and forums?

    a.   Yes   b. No

THE SELF-TEST CODE QUESTION & ANSWERS

Beginners level 1

QUESTION**:** Write a Java program to print the sum (addition), multiply, subtract, divide and remainder of two numbers.

Description:

Challenge:[public class Exercise1 {

```
    public static void main(String[] args) {
    Scanner in = new Scanner(System.in);


    ,System.out.print("Input first number: ");
    int num1 = in.nextInt();
        System.out.print("Input second number: ");
        int num2 = in.nextInt();
        System.out.println(num1 + " + " + num2 + " = " +
        (num1 + num2));


    ,System.out.println(num1 + " - " + num2 + " = " +
     (num1 - num2));
    System.out.println(num1 + " x " + num2 + " = " +
     (num1 * num2));
    System.out.println(num1 + " / " + num2 + " = " +
     (num1 / num2));
    System.out.println(num1 + " mod " + num2 + " = " +
     (num1 % num2));
}

}]
```

Solution: [line2, line3, line1]

Grade: [0-5]

Attempts: 3

Beginner level 1

Question: Write Java program to print alphabets on screen, i.e., a, b, c, ..., z; in lower case.

Challenge:

```
Public Class Alphabets {
public static void main(String args[]) {
char ch;

 for( ch = 'a' ; ch <= 'z' ; ch++ ) System.out.println(ch);
}
}
```

Attempt: 3

Beginner level 1

Question: Java program finds if a number is odd or even. If the number is divisible by two then it will be even, otherwise, it is odd.

Challenge:

```
import java.util.Scanner;
 class CheckEvenOdd {
public static void main(String args[]) {
int num;

System.out.println("Enter an Integer number:");
 num Scanner input = new Scanner(System.in);

 num = input.nextInt();
 if ( num % 2 == 0 ) System.out.println("Entered number is even");
else
```

System.out.println("Entered number is odd");

}

 }

Attempts: 3

Beginner level 2

Question: Write a Java program to print an American flag on the screen.

Description:

Challenge: [public static void main(String[] args)

  {

    System.out.println("* * * * * * ===================================");

    System.out.println(" * * * * * ==================================");

    System.out.println("* * * * * * ===================================");

    System.out.println(" * * * * * ==================================");

    System.out.println("* * * * * * ===================================");

 System.out.println (" * * * * * ==================================");

    System.out.println ("* * * * * * ==================================");

    System.out.println (" * * * * * ==================================");

    System.out.println ("* * * * * * ==================================");

    System.out.println ("===");

    System.out.println ("===");

    System.out.println("==");

    System.out.println("==");

    System.out.println("==");

    System.out.println("==");

  }

}

Solution: [line2,line3,line1]

Grade: [0-3]

Attempts: 3

Beginner level 2

Question: Java program to print multiplication table of a number entered by a user using a for loop.

Chelllenge:

```
import java.util.Scanner;
 class MultiplicationTable {
public static void main(String args[]) {
 int n, c;

System.out.println("Enter an integer to print it's multiplication table");
Scanner in = new Scanner(System.in);

n = in.nextInt();

 System.out.println("Multiplication table of "+n+" is :-");
for ( c = 1 ; c <= 10 ; c++ ) System.out.println(n+"*"+c+" = "+(n*c));


}
 }
```

Beginner level 2

Question: Write a Java program to add two numbers: Given below is the code of a Java program that adds two numbers which are entered by a user.

Challenge:

```java
import java.util.Scanner;
class AddNumbers {

public static void main(String args[]) {
 int x, y, z;

 System.out.println("Enter two integers to calculate their sum ");
Scanner in = new Scanner(System.in);

x = in.nextInt();

 y = in.nextInt(); z = x + y;
System.out.println("Sum of entered integers = "+z);
}


}
```

Beginners level 3

Question: Write a Java program to print a face.

Description:

Challenge:

```java
public class Exercise16 {
   public static void main(String[] args)
   {
      System.out.println(" +\"\"\"\"\"+ ");

    System.out.println("[| o o |]");
    System.out.println(" |  ^  | ");



    System.out.println(" | '-' | ");
    System.out.println(" +-----+ ");
   }
```

}

]

Solution:[line2,line3,line1]

Grade:[0-3]

Attempts:3

Beginner level 3

Question: Write java code to calculate area of a rectangle.

Challenge:

```java
import java.util.Scanner;
 class AreaOfRectangle {
 public static void main (String[] args) {

Scanner scanner = new Scanner(System.in);

 System.out.println("Enter the length of Rectangle:");
        double length = scanner.nextDouble();

System.out.println("Enter the width of Rectangle:");
 double width = scanner.nextDouble();
 System.out.println("Area of Rectangle is:"+area);
 }

 }
```

Beginner level 3

Question: WRITE Java program to calculate area of a square.

Challenge:

```java
import java.util.Scanner;
class SquareArea{
```

```java
public static void main (String[] args) {
System.out.println("Enter Side of Square:");
 Scanner scanner = new Scanner(System.in);

double side =scanner.nextDouble();
 area = side*side; System.out.println("Area of Square is: "+area);
}

}
```

Intermediate level 1

Question: Write Java code to calculate the area and circumference of a circle.
Challenge:

```java
import java.util.Scanner;
class Circle
{
static Scanner sc = new Scanner(System.in);
public static void main(String args[]) {
System.out.print("Enter the radius: ");

 double radius = sc.nextDouble();
//Area = or PI*radius*radius double area = Math.PI  (radius  radius);
 System.out.println("The area of circle is: " + area);
//Circumference = 2*PI*radius double circumference= Math.PI * 2*radius;

System.out.println( "The circumference of the circle is:"+circumference) ;
}

}
```

Intermediate level 1
Question: Write a Java method to find the smallest number among three numbers.
Description:

Test Data:

Input the first number: 25

Input the Second number: 37

Input the third number: 29

Challenge:

```java
import java.util.Scanner;
public class Exercise1 {

public static void main(String[] args)
   {
Scanner in = new Scanner(System.in);
System.out.print("Input the first number: ");


    double x = in.nextDouble();
    System.out.print("Input the Second number: ");
    double y = in.nextDouble();
    System.out.print("Input the third number: ");
    double z = in.nextDouble();



System.out.print("The smallest value is " + smallest(x, y, z)+"\n" );
}

public static double smallest(double x, double y, double z)
{
return Math.min(Math.min(x, y), z);
}


]
```

Solution:[line2,line3,line1]

Grade:[0-3]

Attempts:3

Intermediate level 1

Question: Write a Java code to compare two numbers.

Challenge:

```
public class CompareTwoNumbers {
    public static void main(String[] args){
            int num1 = 324;
            int num2 = 234;
            if(num1 > num2){
  System.out.println(num1 + " is greater than " + num2);
            }
            else if(num1 < num2){
 System.out.println(num1 + " is less than "+ num2);
            }
        else{
                System.out.println(num1 + " is equal to "+ num2);
            }
    }
}
```

Intermediate level 2

Question: Write a Java program to display the speed, in meters per second, kilometers per hour and miles per hour.

User Input: Distance (in meters) and the time was taken (as three numbers: hours, minutes, seconds).

Note: 1 mile = 1609 meters

Description: Test Data

Input distance in meters: 2500

Input hour: 5

Input minutes: 56

Input seconds: 23

Challenge:[

```java
import java.util.Scanner;
public class Exercise7 {

    public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);

float timeSeconds;
float mps,kph, mph;

System.out.print("Input distance in meters: ");
float distance = scanner.nextFloat();
||
System.out.print("Input hour: ");
float hr = scanner.nextFloat();

System.out.print("Input minutes: ");
float min = scanner.nextFloat();

System.out.print("Input seconds: ");
float sec = scanner.nextFloat();

timeSeconds = (hr*3600) + (min*60) + sec;
mps = distance / timeSeconds;
kph = ( distance/1000.0f ) / ( timeSeconds/3600.0f );
mph = kph / 1.609f;
```

```
||
System.out.println("Your speed in meters/second is "+mps);
System.out.println("Your speed in km/h is "+kph);
System.out.println("Your speed in miles/h is "+mph);


scanner.close();
}
}


]
```

Solution: [line2,line3,line1]

Grade: [0-3]

Attempts:3

Intermediate level 2

Question: Write a Java program to display student grade.

```
import java.util.Scanner;


public class JavaProgram
{
    public static void main(String args[])
    {
        int mark[] = new int[5];
        int i;
        float sum=0, avg;
        Scanner scan = new Scanner(System.in);


        System.out.print("Enter Marks Obtained in 5 Subjects : ");
        for(i=0; i<5; i++)
        {
            mark[i] = scan.nextInt();
```

```java
        sum = sum + mark[i];

    }


    avg = sum/5;


    System.out.print("Your Grade is ");
    if(avg>80)
    {
        System.out.print("A");
    }
    else if(avg>60 && avg<=80)
    {
        System.out.print("B");
    }
    else if(avg>40 && avg<=60)
    {
        System.out.print("C");
    }
    else
    {
        System.out.print("D");
    }
    }
}
```

Intermediate level 2

Question: Write a Java code to find the largest of three numbers.

Challenge:

```java
import java.util.Scanner;

class LargestOfThreeNumbers
```

```java
{
  public static void main(String args[])
  {
    int x, y, z;

    System.out.println("Enter three integers ");

    Scanner in = new Scanner(System.in);

    x = in.nextInt();

    y = in.nextInt();

    z = in.nextInt();

    if ( x > y && x > z )

      System.out.println("First number is largest.");

    else if ( y > x && y > z )

      System.out.println("Second number is largest.");

    else if ( z > x && z > y )

      System.out.println("Third number is largest.");

    else

      System.out.println("Entered numbers are not distinct.");

  }

}
```

Intermediate level 3

Question: Write a Java program to print 'Hello' on screen and then print your name on a separate line

Description:

Challenge:

```java
public class Exercise1 {
 ||
 public static void main(String[] args) {
 ||
   System.out.println("Hello\n****** *******!");
 }

 }
 ]
```

Solution:[line2,line3,line1]

Grade:[0-3]

Attempts:3

Intermediate level 3

Question: Write a Java code to display date and time.

Challenge:

```java
import java.util.*;

class GetCurrentDateAndTime

{

  public static void main(String args[])

  {

    int day, month, year;

    int second, minute, hour;

    GregorianCalendar date = new GregorianCalendar();

    day = date.get(Calendar.DAY_OF_MONTH);

    month = date.get(Calendar.MONTH);
```

year = date.get(Calendar.YEAR);

second = date.get(Calendar.SECOND);

minute = date.get(Calendar.MINUTE);

hour = date.get(Calendar.HOUR);

System.out.println("Current date is "+day+"/"+(month+1)+"/"+year);

System.out.println("Current time is "+hour+" : "+minute+" : "+second);

  }

}

Intermediate level 3

Question: Write a Java program to generate random numbers: This code generates random numbers in range 0 to 100.

Challenge:

```java
import java.util.*;

class RandomNumbers {

  public static void main(String[] args) {

    int c;

    Random t = new Random();

    for (c = 1; c <= 10; c++) {

      System.out.println(t.nextInt(100));

    }

  }

}
```

Question: Write a Java program to print the sum of two numbers.

Description: In mathematics, summation (capital Greek sigma symbol: ?) is the addition of a sequence of numbers;

The result is their sum or total. The numbers to be summed may be integers, rational numbers, real numbers, or complex numbers.

Challenge:[

```
public class Exercise2 {
||
 public static void main(String[] args) {
||
   System.out.println(24+26);
 }


}
]
```

Solution:[line2,line3,line1]

Grade:[0-3]

Attempts:3

Question: Write a Java program that takes two numbers as input and display the product of two numbers.

Description:

Challenge:[

```
import java.util.Scanner;


public class Exercise5 {

 public static void main(String[] args) {
  Scanner in = new Scanner(System.in);
```

```
||
  System.out.print("Input first number: ");
int num1 = in.nextInt();


System.out.print("Input second number: ");
int num2 = in.nextInt();
 ||
  System.out.println(num1 + " x " + num2 + " = " + num1 * num2);
}


}
]
```

Solution:[line2,line3,line1]

Grade:[0-3]

Attempts:3

Advanced level 1

Question: Write the java to find substrings of a strings

```java
import java.util.Scanner;

class SubstringsOfAString
{
  public static void main(String args[])
  {
    String string, sub;
    int i, c, length;
    Scanner in = new Scanner(System.in);
    System.out.println("Enter a string to print it's all substrings");
```

```java
    string  = in.nextLine();

    length = string.length();

    System.out.println("Substrings of \""+string+"\" are :-");

    for( c = 0 ; c < length ; c++ )

    {

      for( i = 1 ; i <= length - c ; i++ )

      {

        sub = string.substring(c, c+i);

        System.out.println(sub);

      }

    }

  }

}
```

Advanced level 1

Question: Write a Java program that print reverse of a number.

```java
import java.util.Scanner;

class ReverseNumber

{

  public static void main(String args[])

  {

    int n, reverse = 0;

    System.out.println("Enter the number to reverse");

    Scanner in = new Scanner(System.in);
```

```
    n = in.nextInt();

    while( n != 0 )

    {

     reverse = reverse * 10;

      reverse = reverse + n%10;

      n = n/10;

    }

    System.out.println("Reverse of entered number is "+reverse);

  }

}
```

Advanced level 2

Question: Write Java program to swap two numbers using a temporary variable.

```
import java.util.Scanner;

class SwapNumbers

{

  public static void main(String args[])

  {

    int x, y, temp;

    System.out.println("Enter x and y");

    Scanner in = new Scanner(System.in);

    x = in.nextInt();

    y = in.nextInt();

    System.out.println("Before Swapping\nx = "+x+"\ny = "+y);
```

```java
        temp = x;

        x = y;

        y = temp;

        System.out.println("After Swapping\nx = "+x+"\ny = "+y);

    }

}
```

Advanced level 2

Question: Write java program to print floydtriangle.

```java
import java.util.Scanner;

class FloydTriangle

{

    public static void main(String args[])

    {

    int n, num = 1, c, d;

        Scanner in = new Scanner(System.in);

        System.out.println("Enter the number of rows of floyd's triangle you want");

        n = in.nextInt();

        System.out.println("Floyd's triangle :-");

        for ( c = 1 ; c <= n ; c++ )

        {

            for ( d = 1 ; d <= c ; d++ )

            {

                System.out.print(num+" ");
```

```
        num++;

    }

    System.out.println();

  }

 }

}
```

Advanced level 3

Question: Write a Java program to convert Fahrenheit to Celsius:

```
import java.util.*;

class FahrenheitToCelsius {

  public static void main(String[] args) {

    float temperatue;

    Scanner in = new Scanner(System.in);

    System.out.println("Enter temperatue in Fahrenheit");

    temperatue = in.nextInt();

    temperatue = ((temperatue - 32)*5)/9;

    System.out.println("Temperatue in Celsius = " + temperatue);

  }

}
```

Advanced level 3

Question: Write a Java Program to enter two number and perform the addition, subtraction, multiplication and division.

```
 import java.util.Scanner;
```

```java
public class JavaProgram
{
    public static void main(String args[])
    {
        int a, b, res;
        Scanner scan = new Scanner(System.in);

        System.out.print("Enter Two Numbers : ");
        a = scan.nextInt();
        b = scan.nextInt();
        res = a + b;
        System.out.println("Addition = " +res);
        res = a - b;
        System.out.println("Subtraction = " +res);
        res = a * b;
        System.out.println("Multiplication = " +res);
        res = a / b;
        System.out.println("Division = " +res);
    }
}
```
(http://beginnersbook.com)

APPENDIX C

PERMISSION FROM ONLINE LEARNING SOURCE

Shows the mail sent to the web page www.tutorialpoint.com asking consent to use the examples on the application.

From: Mohtashim M
Sent: Thursday, February 15, 2018 7:41 PM
To: mo.damilola@gmail.com
Cc: contact@tutorialspoint.com
Subject: Re:

Sure you can use. Don't forget to give due credit to the website.

Kind Regards
Mohammad Mohtashim
www.tutorialspoint.com

On Thu, Feb 15, 2018 at 8:38 PM, <mo.damilola@gmail.com> wrote:

Good day

Please I would like to use few contents on the web site for my master thesis application I find it really educative. Please I need a reply I would be glad. Thanks

Sent from Mail for Windows 10

APPENDIX D

SOURCE CODE SPECIAL PARTS OF THE DEVELOPED APPLICATION (Java)

```java
package com.worldfinder.javatutor;

import android.content.Intent;
import android.content.SharedPreferences;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class EnterNameActivity extends AppCompatActivity {
    EditText name;
    TextView save;
    SharedPreferences.Editor editor;
    SharedPreferences sharedpreferences;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_enter_name);


        name = (EditText) findViewById(R.id.name);

        save = (TextView) findViewById(R.id.save);

        save.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                String Namee = name.getText().toString().trim();

                if (Namee.equals("")) {
                    Toast.makeText(EnterNameActivity.this, "Please Enter Your Name!",
Toast.LENGTH_SHORT).show();
                } else {

                    editor = getSharedPreferences("MY_PREFS_NAME", MODE_PRIVATE).edit();
                    editor.putString("UserName", Namee);
                    editor.commit();

                    Intent intent = new Intent(EnterNameActivity.this, OptionsActivity.class);
                    startActivity(intent);


                }
```

```java
            }
        });


    }
}


package com.worldfinder.javatutor;

import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.media.Ringtone;
import android.media.RingtoneManager;
import android.net.Uri;
import android.support.v4.util.Pair;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.woxthebox.draglistview.DragListView;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;
import java.util.Random;

public class Question_1_Activity extends AppCompatActivity {


    DragListView mDragListView;
    private ArrayList<Pair<Long, String>> mItemArray;

    int[] answerArray;
    SharedPreferences.Editor editor;
    SharedPreferences sharedpreferences;
    int attemptCounter = 0;

    TextView questionDetailTv;

    String Question1 = "Write a Java program to print the sum (addition), multiply, subtract, divide and
remainder of two numbers.";
    String Question2 = "Write Java program to print alphabets on screen, i.e., a, b, c, ..., z; in lower case.";
    String Question3 = "Write Java program finds if a number is odd or even. If the number is divisible by two
```

then it will be even, otherwise, it is odd.";
   int[] checkArray;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_question_1_);

     final AppCompatActivity activity = **this**;

```
//      Button nextButton = findViewById(R.id.next);
//      nextButton.setVisibility(View.INVISIBLE);
//      nextButton.setOnClickListener(new View.OnClickListener() {
//          @Override
//          public void onClick(View view) {
//              gotoNextQuestion();
//          }
//      });
```

     questionDetailTv = (TextView) findViewById(R.id.questionDetailTv);

     int min = 1;
     int max = 3;

     Random r = **new** Random();
     int NumberRandm = r.nextInt(max - min + 1) + min;
     Log.d(**"RandomNumber"**, NumberRandm + **""**);

     switch (NumberRandm) {
       case 1:
         questionDetailTv.setText(Question1);
         checkArray = **new int**[5];
         answerArray = **new int**[]{0, 2, 3, 1, 4};
         Question1Content();
         break;
       case 2:
         questionDetailTv.setText(Question2);
         checkArray = **new int**[3];
         answerArray = **new int**[]{0, 1, 2};
         Question2Content();
         break;
       case 3:
         questionDetailTv.setText(Question3);
         checkArray = **new int**[5];
         answerArray = **new int**[]{0, 1, 2, 3, 4};
         Question3Content();
         break;
     }

     mDragListView = (DragListView) findViewById(R.id.question1_list);
     mDragListView.setDragListListener(**new** DragListView.DragListListener() {
       @Override
       public void onItemDragStarted(**int** position) {
         //Toast.makeText(getApplicationContext(), "Start - position: " + position,

```java
Toast.LENGTH_SHORT).show();
    }

    @Override
    public void onItemDragging(int itemPosition, float x, float y) {

    }

    @Override
    public void onItemDragEnded(int fromPosition, int toPosition) {
        if (fromPosition != toPosition) {
            //Toast.makeText(getApplicationContext(), "End - position: " + toPosition,
Toast.LENGTH_SHORT).show();


            for (int i = 0; i < mDragListView.getAdapter().getItemList().size(); i++) {
                int v = (int) mDragListView.getAdapter().getItemId(i);
                checkArray[i] = v;
                //System.out.println(v);
            }

            System.out.println(checkArray);
            System.out.println(answerArray);

            Log.d("checkArray", checkArray + "");
            Log.d("answerArray", answerArray + "");


            attemptCounter = attemptCounter + 1;

            if (attemptCounter == 4) {

                SoundPlayer.playSound(activity, R.raw.attempts_sound);
                Toast.makeText(getApplicationContext(), "Attempts Exceeded Try Again!",
Toast.LENGTH_SHORT).show();
                finish();
            } else if (Arrays.equals(checkArray, answerArray)) {
                //System.out.println("Cool Java Coder");
                // Toast.makeText(getApplicationContext(), "Cool Java Coder", Toast.LENGTH_SHORT).show();
                SoundPlayer.playSound(activity, R.raw.gettinggoodsound);

                editor = getSharedPreferences("MY_PREFS_NAME", MODE_PRIVATE).edit();
                editor.putString("level1Scrore", "5");
                editor.putString("Level1", "done");
                editor.commit();

                AlertCustom();

            }
        }
    }
});


//}
```

```java
        mDragListView.setLayoutManager(new LinearLayoutManager(getApplicationContext()));
        ItemAdapter listAdapter = new ItemAdapter(mItemArray, R.layout.list_item, R.id.text, false);

        mDragListView.setAdapter(listAdapter, false);
        mDragListView.setCanDragHorizontally(false);
    }

    private void Question1Content() {
        mItemArray = new ArrayList<>();

        mItemArray.add(new Pair<>((long) 0, "public class Exercise1 {\n" +
            "\tpublic static void main(String[] args) {"));

        mItemArray.add(new Pair<>((long) 1, "System.out.println(num1 + \" + \" + num2 + \" = \" +  (num1 +
num2));\n" +
            "\t \tSystem.out.println(num1 + \" - \" + num2 + \" = \" +  (num1 - num2));\n" +
            "\t \tSystem.out.println(num1 + \" x \" + num2 + \" = \" +  (num1 * num2));"));

        mItemArray.add(new Pair<>((long) 2, "Scanner in = new Scanner(System.in);\n" +
            "\t\tSystem.out.print(Input first number:);\n" +
            " \t \tint num1 = in.nextInt();"));

        mItemArray.add(new Pair<>((long) 3, "System.out.print(Input second number:);\n" +
            "     int num2 = in.nextInt();"));

        mItemArray.add(new Pair<>((long) 4, "     System.out.println(num1 + \" / \" + num2 + \" = \" +  (num1 /
num2));\n" +
            "\t \tSystem.out.println(num1 + \" mod \" + num2 + \" = \" +  (num1 % num2));\n" +
            "    }\n" +
            "}"));
    }

    private void Question2Content() {
        mItemArray = new ArrayList<>();

        mItemArray.add(new Pair<>((long) 0, "Public Class Alphabets { \n"));


        mItemArray.add(new Pair<>((long) 2, "for( ch = 'a' ; ch <= 'z' ; ch++ ) System.out.println(ch); \n" +
            "} \n" +
            "}\n"));
        mItemArray.add(new Pair<>((long) 1, "public static void main(String args[]) { \n" +
            "char ch; "));


    }

    private void Question3Content() {
        mItemArray = new ArrayList<>();

        mItemArray.add(new Pair<>((long) 1, "public static void main(String args[]) { \n" +
            "int num; "));

        mItemArray.add(new Pair<>((long) 0, "import java.util.Scanner;\n" +
            " class CheckEvenOdd { "));
```

```java
    mItemArray.add(new Pair<>((long) 2, "System.out.println(\"Enter an Integer number:\"); \n" +
        " num Scanner input = new Scanner(System.in);"));

    mItemArray.add(new Pair<>((long) 4, "System.out.println(\"Entered number is odd\"); \n" +
        "}\n" +
        " }\n"));

    mItemArray.add(new Pair<>((long) 3, "num = input.nextInt(); \n" +
        " if ( num % 2 == 0 ) System.out.println(\"Entered number is even\"); \n" +
        "else \n"));


}


private void gotoNextQuestion() {

    Intent intent = new Intent(Question_1_Activity.this, Question_2_Activity.class);
    intent.putExtra("score", 5);
    startActivity(intent);
    finish();

}

public void AlertCustom() {
    AlertDialog.Builder dialogBuilder = new AlertDialog.Builder(Question_1_Activity.this);
    LayoutInflater inflater = this.getLayoutInflater();
    final View dialogView = inflater.inflate(R.layout.custom_dialog, null);
    dialogBuilder.setView(dialogView);
    dialogBuilder.setCancelable(false);

    TextView levelText = (TextView) dialogView.findViewById(R.id.levelText);


    levelText.setText("Configuration Level " + 1 + " Completed !");

    TextView scroreText = (TextView) dialogView.findViewById(R.id.scroreText);


    sharedpreferences = getSharedPreferences("MY_PREFS_NAME", Context.MODE_PRIVATE);
    String levelScore = sharedpreferences.getString("level1Scrore", "");
    TextView totlaScoreTv = (TextView) dialogView.findViewById(R.id.totlaScoreTv);
    scroreText.setText("Your Score: " + levelScore);
    totlaScoreTv.setText("Total Score: " + totoalScoreCount());


//      dialogBuilder.setTitle("Custom dialog");
//      dialogBuilder.setMessage("Enter text below");
    dialogBuilder.setPositiveButton("Next Level", new DialogInterface.OnClickListener() {
      public void onClick(DialogInterface dialog, int whichButton) {
        //do something with edt.getText().toString();
        gotoNextQuestion();
      }
    });
    dialogBuilder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
```

```java
        public void onClick(DialogInterface dialog, int whichButton) {
            //pass
        }
    });
    AlertDialog b = dialogBuilder.create();
    b.show();
}


private String totoalScoreCount() {

    String level1Scrore = sharedpreferences.getString("level1Scrore", "0");
    String level2Scrore = sharedpreferences.getString("level2Scrore", "0");
    String level3Scrore = sharedpreferences.getString("level3Scrore", "0");
    String level4Scrore = sharedpreferences.getString("level4Scrore", "0");
    String level5Scrore = sharedpreferences.getString("level5Scrore", "0");
    String level6Scrore = sharedpreferences.getString("level6Scrore", "0");
    String level7Scrore = sharedpreferences.getString("level7Scrore", "0");
    String level8Scrore = sharedpreferences.getString("level8Scrore", "0");
    String level9Scrore = sharedpreferences.getString("level9Scrore", "0");



    int TotalScore =
        Integer.valueOf(level1Scrore)
            + Integer.valueOf(level2Scrore)
            + Integer.valueOf(level3Scrore)
            + Integer.valueOf(level4Scrore)
            + Integer.valueOf(level5Scrore)
            + Integer.valueOf(level6Scrore)
            + Integer.valueOf(level7Scrore)
            + Integer.valueOf(level8Scrore)
            + Integer.valueOf(level9Scrore);


    return String.valueOf(TotalScore);
}



}




package com.worldfinder.javatutor;

import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.media.Ringtone;
import android.media.RingtoneManager;
import android.net.Uri;
import android.support.v4.util.Pair;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.LinearLayoutManager;
import android.util.Log;
```

```java
import android.view.LayoutInflater;
import android.view.View;
import android.widget.Button;
```