USER INTERFACE AND INTERACTIVITY DESIGN GUIDELINES OF ALGORITHM VISUALIZATION ON MOBILE PLATFORM

AHMAD AFFANDI SUPLI

DOCTOR OF PHILOSOPHY
UNIVERSITI UTARA MALAYSIA
2019

Permission to Use

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences
UUM College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok

Abstrak

Visualisasi Algoritma (AV) adalah alat pedagogi yang dapat membantu pelajar melihat proses algoritma secara animasi. Para pelajar boleh menonton dan memerhatikannya melalui penjelasan animasi dinamik. Kajian terdahulu menunjukkan bahawa kajian mudah alih AV masih terhad. Pelaksanaan AV di platform mudah alih masih dianggap sebagai trend baharu yang bermula pada tahun 2013. Di samping itu, garis panduan reka bentuk komprehensif bagi AV dari segi mereka bentuk antara muka pengguna (UI) dan faktor interaktiviti adalah masih terhad dan dibincangkan secara berasingan dalam kajian terdahulu. Banyak bukti dalam kajian empirikal terdahulu menunjukkan kepelbagaian strategi interaktiviti dan aspek reka bentuk UI adalah dua aspek untuk membuat alat AV yang berkesan kepada pelajar. Dalam konteks ini, kajian ini mencadangkan garis panduan berhubung reka bentuk AV pada platform mudah alih (AVOMP) yang berfungsi sebagai pendekatan sistematik yang merangkumi cadangan asas untuk pereka, pemaju, dan pensyarah bagi menghasilkan AVOMP berdasarkan dua aspek, iaitu reka bentuk UI dan interaktiviti. Oleh itu, bagi memastikan matlamat utama tercapai, beberapa sub-tujuan telah dibentuk iaitu: (a) untuk mengenal pasti cadangan yang sesuai untuk aspek reka bentuk UI dan interaktiviti AVOMP, (b) untuk membangunkan garis panduan reka bentuk AVOMP berdasarkan cadangan yang dikenal pasti iaitu reka bentuk UI dan interaktiviti, (c) untuk mengesahkan garis panduan reka bentuk yang dibangunkan AVOMP dari segi "kegunaan" melalui semakan pakar, dan (d) untuk mengukur keberkesanan AV pada platform mudah alih yang melaksanakan garis panduan reka bentuk yang dicadangkan melalui prototaip. Kajian ini menggunakan kaedah Penyelidikan Sains Reka Bentuk sebagai rangka kerja proses penyelidikan. Aktiviti pembinaan garis panduan reka bentuk AVOMP meliputi sorotan karya dan kajian perbandingan. Garis panduan reka bentuk yang dicadangkan disahkan melalui semakan pakar yang melibatkan 16 orang pakar. Keputusan dari pengujian hipotesis menyimpulkan bahawa garis panduan reka bentuk AVOMP yang dicadangkan menunjukkan keputusan signifikan kerana ia mempunyai kualiti untuk dijadikan garis panduan bagi pembangun atau pereka bentuk dan membangunkan AVOMP. Selain itu, penilaian keberkesanan prototaip AVOMP daripada 35 orang peserta melalui eksperimen makmal berasaskan ujian Taksonomi Bloom menunjukkan bahawa terdapat perbezaan yang signifikan antara pendekatan manual penyusunan pembelajaran algoritma (Pra-ujian) dan pendekatan manual penyusunan pembelajaran algoritma (Pasca-ujian). Oleh itu, kajian ini memberi sumbangan melalui tiga aspek iaitu sumbangan dari segi artifak, empirikal, dan juga teori. Dalam aspek sumbangan artifak, kajian ini menghasilkan garis panduan reka bentuk AVOMP yang terdiri daripada aspek reka bentuk UI dan interaktiviti serta susunan prototaip AVOMP algoritma. Manakala sumbangan dari segi empirikal pula menunjukkan hasil keberkesanan aplikasi AVOMP. Akhir sekali, aspek teori menyumbang kepada keberkesanan garis panduan reka bentuk AVOMP yang dibangunkan secara berstruktur dan komprehensif dibentuk dengan gabungan sekumpulan teori dan kajian empirikal dari dua aspek, iaitu reka bentuk UI dan interaktiviti.

Kata kunci: Visualisasi algoritma, platform mudah alih, garis panduan reka bentuk, taksonomi penglibatan, reka bentuk UI.

Abstract

Algorithm Visualization (AV) is a pedagogical tool that can help learners to see the animation of the step-by-step process of an algorithm. Students can watch and observe through the elaboration of dynamic animation. Previous studies show that AV mobile study is still limited. AV implementation on the mobile platform is still considered as a new trend which started in 2013. In addition, comprehensive design guidelines of AV in terms of designing user-interface (UI) and interactivity factors are still limited and discussed separately in previous studies. Even though, much evidence in previous empirical studies show that various interactivity strategies and UI design aspects are two imperative aspects to make an effective AV tool for learners. Within this context, this study proposes AV on mobile platforms (AVOMP) design guidelines that serve as a systematic approach. It includes the fundamental recommendations for designers, developer, and lecturers to produce AVOMP which are based on two aspects, namely UI design and interactivity aspects. Hence, in order to accomplish the main aim, a number of sub-objectives were formed: (a) to identify the appropriate recommendations for UI design and interactivity aspects of AVOMP, (b) to develop the design guidelines of AVOMP based on the identified recommendations of UI design and interactivity, (c) to validate the developed design guidelines of AVOMP in terms of "usefulness" through expert reviews, and (d) to measure the effectiveness of AV on mobile platform that implements the proposed design guidelines through prototype. This study adopted the Design Science Research methodology as the framework of the research process. Activities of AVOMP design guidelines construction include a literature review and a comparative study. The proposed design guidelines were validated through expert reviews, which involved 16 experts. Results of the hypothesis testing concludes that the proposed AVOMP design guidelines are significantly perceived as having quality in serving as a guideline for developers or designers to design and develop AVOMP. Moreover, the evaluation of the effectiveness of the AVOMP prototype from 35 participants through laboratory experiments based on the bloom taxonomy test shows that there is a significant difference between students learning sorting algorithms using the manual approach (Pre-Test) and the AVOMP app (Post-Test). Hence, this study makes three major contributions, namely artefact, empirical, and theoretical. In terms of artifact, this study yields AVOMP design guidelines that are comprised of UI design and interactivity aspects as well as AVOMP prototype of sorting algorithms. Meanwhile, empirical contribution shows the result of the effectiveness of AVOMP apps. Finally, the theoretical aspect contributes the novelty of the developed design guidelines of AVOMP that are structured and comprehensively formed with a combination of a bunch of theories and empirical studies of two aspects, which are UI design and interactivity.

Keywords: Algorithm visualization, mobile platform, design guidelines, engagement taxonomy, UI design.

Acknowledgement

I wish to express my deepest and warmest thanks first and foremost to Allah S.W.T, The Almightiest and Merciful that I shall ever be indebted to for all His bounties upon me. Deeply grateful to our Prophet Muhammad S.A.W. of great messenger who has given light to mankind so that we can identify the true meaning of Islam in our life. I am greatly indebted and would like to thank to so many wonderful people for their contributions and spontaneous assistance in so many ways in completing this thesis.

First, I would like to thank to my supervisor, Prof. Dr. Norshuhada Bt Shiratuddin for her supervision on critical assessment, supervision, invaluable advice, compassion, understanding, patience, kindness and encouragement throughout my study. Without her support, invaluable knowledge, efforts and time, it would be difficult to complete this research. I would like to also thank to my co-supervisor, Assoc. Prof. Dr. Syamsul Bahrin Zaibon for his guidance and supervision.

I wish and extend my deepest gratitude to my parents, Mr. Supli Effendi Rahim, Mdm. Nurhayati Damiri, and to my parent-in-law, Azharuddin Sahil and Sri Rahayu, for their endless support, patient, and prayers. I would also wish to thank to my sisters, Ika, Mia, Nurul, Rani, Zulfah and Iin, for their encouragement and spiritual support. Also many thanks to my wife Novia Zahrah who have fully support, provided exceptional guidance, and encourage me so that I wouldn't give up, be positive and be motivated to complete this research. Thanks for being beside me all my life.

Finally, special thanks to all my colleagues and lectures for their motivational spirit.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

DSA	Data Structure and Algorithm
AV	Algorithm Visualization
PV	Program Visualization
UI	User Interface
HCI	Human Computer Interaction
AVOMP	Algorithm Visualization on Mobile Platform
G	Guideline
ET	Engagement Taxonomy
EET	Extend Engagement Taxonomy
2DET	Two-dimensional Engagement Taxonomy
CLT	Cognitive Load Theory
GUI	Graphical User Interface
IS	Information System
IT	Information Technology
DSR	Design Science Research
ITM	Iterative Triangulation Methodology
KA	Knowledge Acquisition
CS	Computer Science
SE	Software Engineering
UUM	Universiti utara Malaysia

List of Appendices

CHAPTER ONE
INTRODUCTION

1.1 Overview

This introductory chapter consists of a background of study which discusses on issues that lead to the motivation of the study, explanation of the problem background, research gap, and then research objective formulation. This section also elaborates the scope and limitations of the study, significance of the study, theoretical and research framework, as well as operational definitions of terms used throughout the study.

1.2 Motivation of Study

Learning Data Structures and Algorithm (DSA) course in understanding the algorithm concept is a really challenging task in computer science education (Bäck, Fogel, & Michalewicz, 2018; Crescenzi, Malizia, Verri, Díaz, & Aedo, 2012; Osman & Elmusharaf, 2014). DSA is an important subject for students in computer science education, but it is mostly hard to learn due to the abstract nature (Sadikan & Yassin, 2012). In general, the students have difficulty in order to depict visually the step by step process that is involved in the algorithm.

According to Patel (2014), there are four main problems faced by lecturers and tutors to teach this course: first is the low motivation of students. The second problem is how to explain DSA concept, which can be tricky and abstract to students. Third, in order to do the assignment, the cooperation between students is still low. Fourth, the implementation of algorithms to implement in real life situation setting is hard to do.

Initially, the learning process of this course relies on conventional methods, such as textbook and power point. In particular, the use of images is utilized in order to describe transitions and processes that occur in an algorithm. However, instructors find it difficult to describe it well between the image and its represented line of an algorithm (del Vado Vírseda, 2010). One of the main reasons is because students could not grasp the concept of dynamic movement which is complicated in an algorithm when taught with static images. In addition, there is a possibility when tutors teach them so fast (Halim, Koh, Loh, & Halim, 2012). Another way is to draw the sequence of algorithm animation on the board. This method may have an advantage so that students can understand the concept, but it is time-consuming.

Thus, to address this issue, instructors and tutors have turned to utilize algorithm visualization (AV) technology (Shaffer et al., 2010). AV is the systematic system that could animate or visualize the step by step process of an algorithm that is being displayed. The students can learn throughout the elaboration of an algorithm with dynamic visualization or animation.

Initially, AV was presented passively to students as movie animation only. However, researchers such as Hundhausen, Douglas, and Stasko (2002) concluded from their result that passive AV gives little impact towards students understanding. In AV system, there is a need to also interact with an AV system. Therefore, the interactivity has to be built in AV system, so that a conducive of an active learning environment can be achieved to improve students' motivation. Many AV studies (Adamchik, 2011; Burguillo, 2010; Moons & De Backer, 2013; Sadikan & Yassin, 2013; Segura, Pita, del Vado Vírseda, Saiz, & Soler, 2008) confirmed that interactivity is the most important elements that must be taken into account for engaging learners with the system, otherwise the animation will have no significant effect on effective learning of educational setting.

According to Naps et al. (2002), there are six levels of student interaction with AV: no viewing, viewing, responding, changing, constructing, and presenting. No viewing interaction (provides plain description) is the lowest level to interact with AV, whereas the presenting is the highest level. This suggests in interactivity itself, the variation of interactivity concept (mix of different engagement) is also required so that students can engage and be motivated more in learning the process of an algorithm (see Figure 1.1) (Alhosban & Hamad, 2011; Karavirta & Shaffer, 2013; Sorva, Karavirta, & Malmi, 2013). They can be in the form of problem-solving exercises, game concept activity, mouse viewing activity, etc.

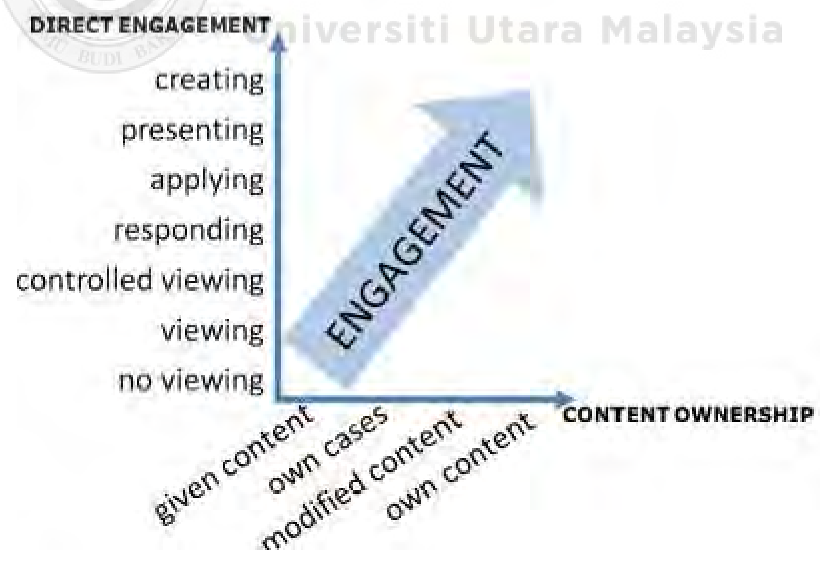


Figure 1.1. The two-dimensional engagement taxonomy (Sorva et al., 2013)

In the meantime, the UI design aspect confirms on how the AV interface and its content should be arranged based on human perspective design guideline. This has to be explored from the theories of Multimedia learning and Educational Technologies (Lee & Rößling, 2010).

Moreover, apart from all above considerations, many studies have conducted research on AV system, however, there is still lack of AV study that is conducted on mobile platform (Patel, 2014). The detail explanation about this issue is elaborated in section 1.3. Thus, this study takes an initiative to conduct a study of AV system as to improve the learning environment, specifically on mobile platform. The next sub-section further justifies the significance of mobile learning for this study.

1.2.1 Current State of Mobile Devices in Education

There are some reasons why the mobile platform is nowadays mostly selected for improving students' learning performance compared with other platforms, such as desktop platform and web platform. According to Duin, Anklesaria, and Nater (2012), mobile devices are presently turning into a prevailing technology with noteworthy potential as learning instruments. Since tablets and smartphone are really advantageous, ubiquitous, and invaluable for web technology; the use of them by students is quickly growing. Eventually, the learning affordances of mobile device incorporate, expediency, immediacy, flexibility, accessibility and portability (Crescente & Lee, 2011; Terras & Ramsay, 2012) cross-context learning, interactivity, individuality and network accommodation (Eliasson, Pargmann, Nouri, Spikol, & Ramberg, 2013b; Park, 2011).

Moreover, Mobile learning encourages peer to peer and collaborative learning both in virtually and individually, allowing students and gatherings to create and share artifacts of learning free of time and space (Attwell, 2012; Eliasson et al., 2013b; Wang & Shen, 2012)

The studies by Dahlstrom, Walker, and Dziuban (2013) showed that college students are really enthusiastic towards the use of mobile devices as a learning tool and also hope more from their lectures to always use mobile devices. Another study by Ciampa (2014) stated that students in the 6^{th} grade are successfully motivated using mobile device technology (tablet) during the learning process in the class.

Furthermore, the investigation conducted by Hwang and Wu (2014) who reviewed 7 well-known Social Science Citation Index (SSCI) of publication journals regarding the influence of mobile application technology in enhancing learning concludes that mobile learning is promising in enhancing the learners' learning accomplishment, interest, and motivation. According to their survey (2008-2012), it is discovered that tablet PCs and smartphone are gradually becoming the most used as mobile learning device adopted application (see Figure 1.2). The red line shows the inclination of smartphone within 5 years has surpassed the other devices as learning environment

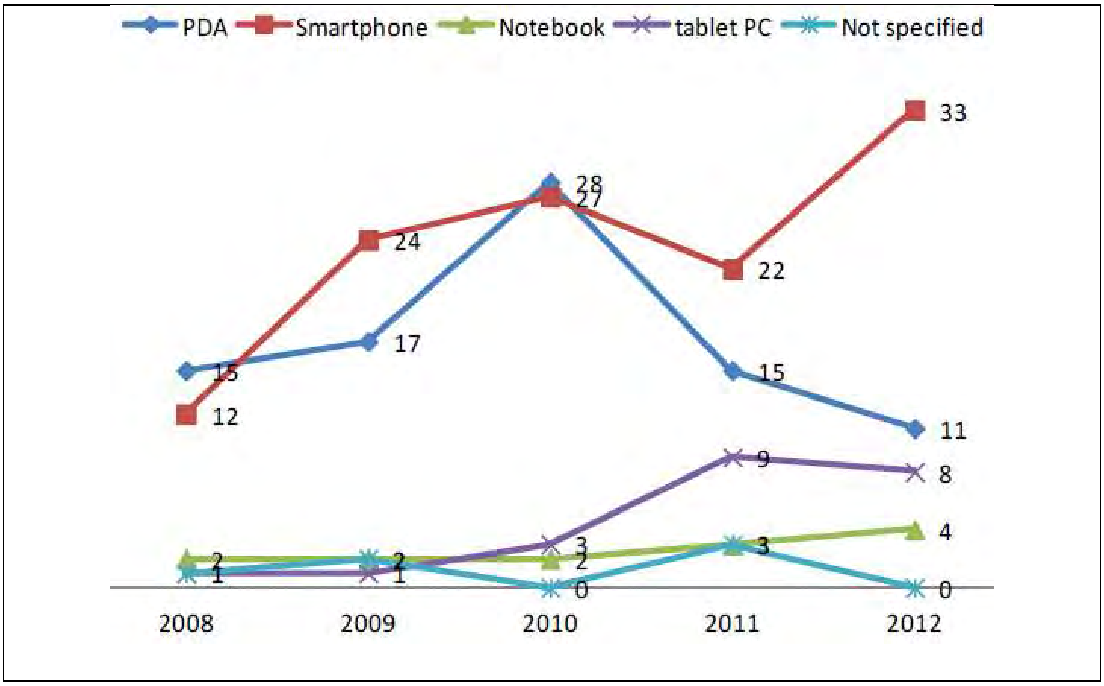


Figure 1.2. The use of devices for learning environment in 2008–2012 (Hwang and Wu, 2014).

In addition, according to the study by Crompton and Burke (2018) that 24 published studies from year 2010 until 2016 about the use of mobile device in higher education learning in Africa found out that mobile learning within higher education institutions in Africa increased student and lecturer collaboration and, provide dinstant communication, increased student participation and engagement, facilitating authentic learning and reflective practice, as well as fostering learning communities.

Moreover, As far as learning advancements are concerned, some studies have confirmed through quantitative analysis with pre and post-test that proved mobile learning yields significant learning improvement (Basoglu & Akdemir, 2010; Chen & Chang, 2011; Delgado-Almonte, Andreu, & Pedraja-Rejas, 2010; Hwang, Shi, & Chu, 2011; Liu, Geurtz, Karam, Navarrete, & Scordino, 2013). Next, it is explained, the trend used in Malaysia's universities for teaching DSA subject, as another motivation that makes this research is conducted.

1.2.2 Current State of DSA Teaching Aid in Malaysia

According to Patel (2014) who surveyed AV studies in the literature of 48 papers from 2001 to 2013, he reported that a lot of efforts have been conducted across the world to use AV as a teaching tool in teaching data structures and algorithm (DSA). On the other hand, in Malaysia itself particularly, the use of AV as teaching aids for DSA in the computer science field is still rare (Ramli & Habib, 2013).

This has been reported by Ramli and Habib (2013) in their survey from 19 public universities in Malaysia (UITM, IIUM, UM, UPM, UKM, UTM, USM, UUM, UPNM, UTeM, UMP, USIM, UPSI, UTHM, UniSZA, UMT, UniMAP, UNIMAS and UMS), in which 51 respondents are computer science faculty lectures that have 2 to 20 years experiences in teaching algorithm and programming. They found out that PowerPoint presentation is the most used as teaching aid in computer science education faculty with 94.1%, then the second highest is using Exercises and Solution (92.2%), and the third highest is textbook (90.2%) (see Figure 1.3).

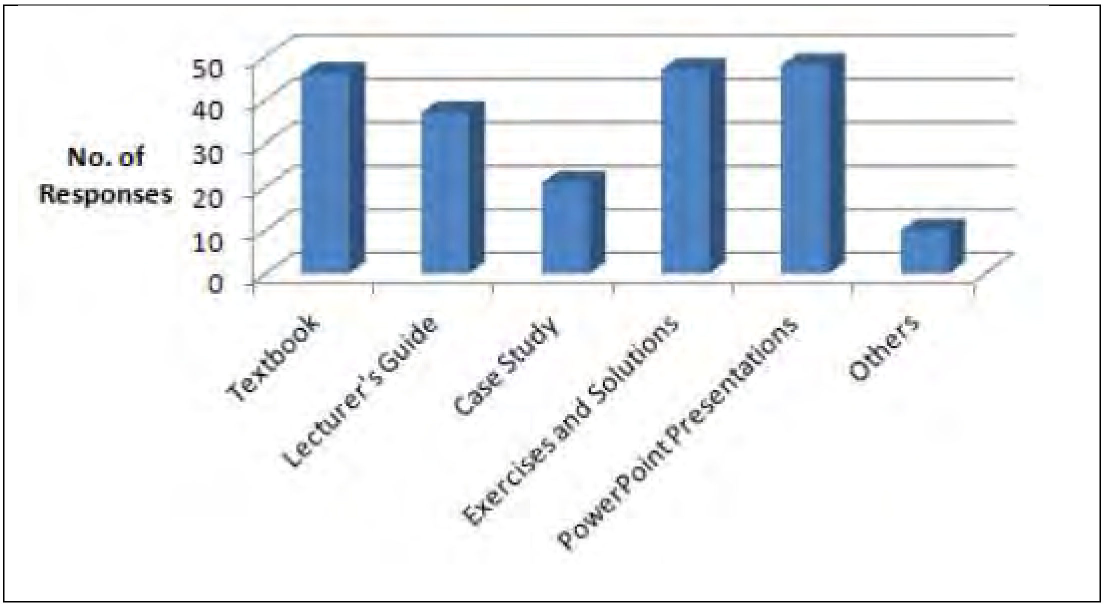


Figure 1.3. Teaching aid in computer science education faculty (Patel, 2014)

Therefore, this study has the momentous motivation of AV system for Malaysia education particularly (matching with the Malaysia university syllabus), and to enhance the design guideline of the body of knowledge in the field of algorithm visualization generally. That means to bring meaningful learning process using mobile technology that can boost motivation and engage students in learning DSA course.

1.2.3 The Benefits of AV System

As mentioned before, AV can help tutors or lecturers to explain how the algorithm works by showing the dynamic animation of step by step line of codes. This is not only can help in terms of efficient use of time in the class, but also the learners can gain better insight in learning algorithm (Segura et al., 2008). Not to mention, to create engaging learning circumstances and motivate students, AV system is highly necessary (Halim et al., 2012). This study attempts to conduct AV on mobile platform so that the learning setting that is personable, smart, easy to use, mobile, and engaging can be achieved.

1.3 Problem Statement

In order to develop an AV system in which students are expected to learn effectively and be more engaging, several design guidelines have been defined across the literatures (Alhosban & Hamad, 2011; Lee & Rößling, 2011; Meolic & Dogša, 2014; Osman & Elmusharaf, 2014; Rößling & Naps, 2002). However, there is still lack of investigation regarding design guidelines in AV study that is specifically designed on mobile platform (see Section 2.9). Prior studies mostly choose desktop platform (Osman & Elmusharaf, 2014; Sadikan & Yassin, 2012) or web platform (Alhosban & Hamad, 2011; Foutsitzis & Demetriadis, 2013). Only two of AV studies (Meolic, 2013; Meolic & Dogša, 2014) that were applied to mobile platform. This is due to the study of mobile learning of algorithm visualization is still relatively new, which just started in 2013 (Bäck et al., 2018; Pathania & Singh, 2014).

Furthermore, according to Lee and Rößling (2010) and Cooper, Shaffer, Edwards, and Ponce (2014), in order to make an effective AV system, there are two most important aspects that have to be considered by AV designers, namely, interactivity and the user interface design. Interactivity is a function that confirms student's engagements through students input or interaction (Segura et al., 2008). Meanwhile, the user interface design is about the arrangement of AV layout, consisting of symbol systems' components, such as graphics, texts, sounds, animation, icons, and so on. Cooper et al. (2014) asserted that poor design of AV system will affect towards low effectiveness.

The former aspect is based on the learning theory of constructivism (Hadjerrouit, 2005) that advocated the more students directly act and manipulate towards learning materials, the more physiological entanglement and mental efforts, and consequently the better the learning performance. In addition, as mentioned before (see section 1.2), to be active learning only is not sufficient; the AV system should have the variety of interactivity. Particularly, to include more interactive functions based on engagement taxonomy and know how to arrange or apply them and identify the relationships between one interactive feature and others are considerably important pertaining to make students more engage towards learning material; and thus it will impact to their educational performance (Sorva et al., 2013).

Equally, the latter aspect's arrangement is also crucial to be taken into account for the success of effective learning in software visualization (Zhang, 2012). Particularly, how the manipulation and interplay upon these components of AV user interface design that could either alleviate the cognitive load of students advantageously or vice versa (Lee & Rößling, 2010).

In regard to this study, however, in general, the previous AV design guidelines mostly elaborated those two aspects separately and also lack of comprehensive practical recommendation of user interface (UI) design and interactivity for AV on mobile platform specifically. Most of UI design aspects in previous AV studies are prepared for desktop and website, whereby mobile platform has different constraints such as small screen and features such as touch screen (Karam, 2015). Moreover, in terms of interactivity factor, two existing AV design guidelines on mobile platform, as elaborated in section 2.9, employed the lowest level of interactivity (viewing level) in their AV guidelines to demonstrate sorting algorithms as a case study.

Both just used step through navigation method only. The inclusions of interactive functions, such as pause, stop, next, and backward buttons are utilized to demonstrate step by step process of an algorithm. It is indeed that the app shows the animation for each line of codes, but this way only is in contrast with the previous aforementioned deliberation that suggests a variety of interactivity must be included in an AV system; otherwise, the impact of learning will be not significant. Specifically, this AV system can lead students to have a problem in terms of low engagement with AV system itself, and thus will effect on its effectiveness (Patel, 2014).

Therefore, this study proposes to develop design guidelines that can help AV designers developing an effective AV system on mobile platform based on two main perspectives, namely, UI design and interactivity that are appropriate on mobile platform. These proposed design guidelines are prescriptive in which they provide useful recommendation of Do's and Don'ts, which are derived and grounded from a collection of relevant theories and empirical studies (Leavitt & Shneiderman, 2006; O'Malley et al., 2005; Shitkova, Holler, Heide, Clever, & Becker, 2015).

Leavitt and Shneiderman (2006) and Shitkova et al. (2015) in addition stated that guidelines ought to be presented complete with an example(s) (illustration) and justification. The justification can be from the combination of relevant theories and empirical studies (O'Malley et al., 2005). These relevant empirical studies are so important to strengthen and justify the proposed guidelines through examining what the result of their experiments and recommendations (Shitkova et al., 2015).

All in all, it is highly important in this study to investigate further about the recommendation of interactivity (comprising what kind of interactivity features and its relationship one another) based on engagement taxonomy concepts and relevant experimental research of AV interactivity (Rößling & Naps, 2002; Sorva et al., 2013) as well as UI design that are appropriate on mobile platform. UI design has to be examined in an effort to see the interrelationship between components of interface and the effects of their juxtaposition and its presence, which either could improve or obstruct toward effective learning (Lee & Rößling, 2010).

Based on the discussed problem, the following research gaps are concluded:

i) The design guidelines of AV development specifically designed on mobile platform are still lacking and insufficiently explored.

ii) The UI design aspect that should be taken into account when developing AV is not clearly identified in the existing design guidelines, especially for AV on mobile platform.

iii) There is still lack of AV studies on mobile platform that applies the interactivity based on engagement taxonomy.

iv) The previous mobile AV design guidelines still show indication of the low effectiveness of AV system, which is characterized by low interactivity level (low engagement) and lack of comprehensive guidelines for UI.

In nutshell, in this study, design guidelines are defined as sets of recommendations towards good practice in designing AV on mobile platform. They are intended to provide clear instructions to designers and developers on how to adopt specific principles. Therefore, based on the aforementioned discussion, the recommendations itself are divided into two aspects, which are UI design and interactivity. The details of how the design guidelines are constructed based on those recommendations can be seen further in chapter 2, section 2.12.

1.4 Research Question

Based on the problem statement discussed previously, the following research questions are formed:

1. What are the appropriate recommendations for UI design and Interactivity of AV on mobile platform?
2. How to construct the design guidelines of AV on mobile platform based on the identified recommendations of UI design and interactivity?
3. How to validate the proposed design guidelines of AVOMP through expert review and prototyping?
4. Is the AV on mobile platform that implements the proposed design guidelines effective?

1.5 Research Objectives

The main aim of this study is to propose the design guidelines of AV on mobile platforms (AVOMP). Thus, to achieve the main aim, the following specific objectives are planned:

1. To identify the appropriate recommendations for UI design and Interactivity aspects of AVOMP.
2. To construct the design guidelines of AVOMP based on the identified recommendations of UI design and interactivity.
3. To validate the proposed design guidelines of AVOMP through expert review and prototyping.
4. To measure the effectiveness of the AV on mobile platform that implements the proposed design guidelines.

1.6 Research Scopes

The research scopes are described as follows:

1. The study is based in Malaysia, thus the syllabus of DSA courses is referred to the standard of Malaysian university, which focuses on three topics of DSA, which are: List, Stack, and Queue as a case study.
2. The study involves Information and Technology (IT) and Computer Sciences students, who take DSA course.
3. The prototype in this study emphasizes the implementation on mobile platform (mobile application).

4. The study chooses sorting algorithms as part of DSA sub-topic in the prototype (the justification can be seen at section 2.2.1 in Chapter 2).
5. Pre-Experimental Design with one group pre-test post-test study is used in the study.
6. There are 4 levels of Bloom Taxonomy applied in this study: Remembering, Understanding, Applying, and Analyzing (the justification can be seen at section 3.3.3.2.2 in Chapter 3).

1.7 Contribution of Study

The following points describe the significance of this study:

1. Artifact

A. The Design Guidelines of AVOMP

The main contribution of this study is the design guidelines of AVOMP. The design guidelines of AVOMP reveal and facilitate new insight and outcomes as they will be considered as new references for AV developers, AV designers, and tutors who want to develop AV on mobile platform with comprehensive combination of UI design and interactivity aspect. This is due to the previous studies have not comprehensively touched these two combinations yet as one in general for AV and mobile platform specifically. Therefore, these design guidelines provide new and inclusive ways which enable AV developers or designers to implement them to any mobile platforms (iOS, android, windows mobile, etc.) and also are reusable in any DSA topics, algorithm subject, artificial intelligent (AI), and any other subjects that require algorithms to be visualized.

B. AVOMP Prototype

Novel AVOMP prototype, which is as a result of the implementation of the design guidelines of AVOMP, exhibits the new possible future because it will lead to accomplish the new variant way for students to learn algorithm more profoundly and effectively as it is enriched with various of interactivity techniques (viewing, responding, changing, and constructing) and is designed systematically and comprehensively.

2. Empirical

The empirical contribution of this study is in form of quantitative data which follows from scientific study through laboratory experiment. The empirical data are measured based on the effectiveness of AVOMP app that implements the developed design guidelines of AVOMP. The findings reveal the new insight about students' performance (effectiveness) in learning three sorting algorithms (bubble sorting, insertion sorting, and selection sorting) using AVOMP app. The empirical methods used to measure AVOMP app effectiveness include experimental studies (using Pre-Test and Post-test, comparing studying using manual approach vs AVOMP app) and a survey questionnaire (consists of 10 statements) is also distributed to capture the students' opinion towards the prototype (using 5 likert scale). In detail, the empirical contributions are analyzed and compared based on 4 levels of bloom taxonomy (knowledge, analysis, understanding, and apply) and each sorting algorithm.

3. Theoretical

The theoretical contribution of this study comprises the novelty of the developed design guidelines of AVOMP that are structured and comprehensively formed with a combination of a bunch of theories and empirical studies of 2 aspects, which are UI design and Interactivity that have yet to be explored previously. So that, they will give essential features to significantly advance the understanding of AV developers or designers by providing inherently reusable constructs to help them developing AV on mobile platform in any topics of learning algorithm.

Therefore, in detail, the study conducts the content analysis and comparative analysis to find the appropriate recommendations for UI design through some theories, such as color theory, UI design guidelines and principles, user experience design, mobile design principle, gestalt principle, symbol system theory, cognitive science theory, multimedia learning theory and so forth. Also, this study also highly emphasizes the interactivity guidelines, so that it is necessary to explore the relevant theories (Engagement Taxonomy (ET), Extended Engagement Taxonomy (EET), Two-Dimensional Engagement Taxonomy (2DET), Usability of mobile learning, Constructivism Learning Theory, Discovery Learning Theory) and studies of interactivity; more specifically: in terms of what kind of interactivity features though EET engagement taxonomy , how their hierarchical structures, and so on.

4. **Comparative Analysis of Existing Studies**

The comparative analysis concerning existing studies will be conducted in order to identify the generic components and underlying theories for proposing design guidelines of AVOMP in this study. The outcomes can be useful for future researchers to broaden their insight in the field of AV study.

1.8 Research Framework

To accomplish the objectives of the study, this research comprises five phases as portrayed in Figure 1.4. In this study, the research problem, research scope, and research gap are identified through reviewing from existing design guidelines of AV. Then, in the suggestion phase, in order to identify appropriate UI design and interactivity theories, the literature review and content analysis towards related concepts, such as constructivism learning theory, interactivity based on engagement taxonomy, cognitive load theory, gestalt theory, color, design guideline for mobile learning and so on are conducted.

Then, those components are constructed into a first draft of the proposed design guidelines of AVOMP. The reviewing by experts in the proposed guidelines is performed before proceeding to the agreement of the final proposed design guidelines of AVOMP. The next phase, which is called the evaluation, is taken place to create the working prototype based on the proposed design guidelines of AVOMP. This prototype is used as validation stage of proposed design guidelines of AVOMP and then will be assessed in terms of its effectiveness. To achieve the robustness of desired outcomes, all phases and stages are executed iteratively according to design science research methodology, which will be discussed in chapter 3. Finally, the conclusion of the study is drawn by answering research objectives and research questions.

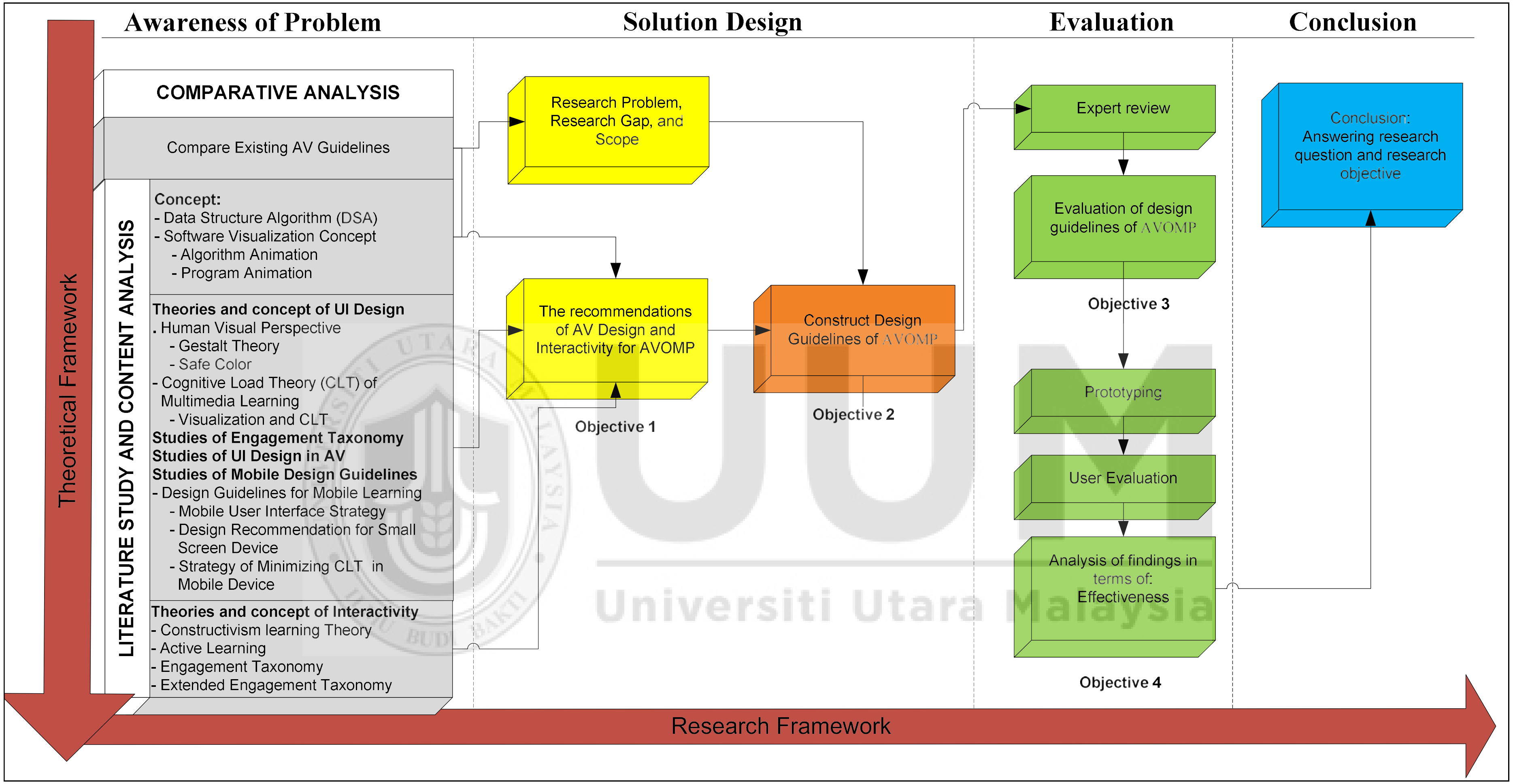


Figure 1.4. Theoretical and research framework

1.9 Operational Definition of Terms

This section explains the terminologies pertaining to this study which direct to the operational terminologies that are involved in this study.

a) Data Structure and Algorithm (DSA)

DSA is one of the courses taught in Computer Science Education. DSA is a fundamental course that contains the abstract concept of learning how algorithms work within a variety of data structure types, such as: single linked list, stack, queue, and trees. The further explanation of what data structure is, and its types are discussed in section 2.1.

b) Algorithm Visualization (AV)

AV is a system or tool that can visualize the step by step process in an algorithm. AV is prepared to comprehend student toward the high level of abstraction in algorithm concept.

c) UI Design

UI Design refers to the arrangement the AV layout that consists of the combination symbol systems (texts, animations, pictures, icons, graphics, etc.). This needs to follow the relevant theories so that students will not be distracted from the wrong implementation of the user interface.

d) Interactivity

Interactivity is a function in which a student is prompted to be interactive through input or activities within a system.

e) Design Guidelines of AVOMP

Design Guidelines are the useful set encompassing the Do's and Don'ts advice. In this study, the design guidelines of AVOMP contains the useful sets from two main perspectives: UI design and interactivity.

f) Mobile Platform

The mobile platform is a platform that is based on mobile device utilization. Meaning, this study emphasizes the use of AV on the mobile application.

g) Pseudocode

is an artificial and informal language that helps programmers develop algorithms. It is a "text-based" detail (algorithmic) design tool.

1.10 Summary

This section summarizes that chapter one has an initial point to propose design guidelines of AVOMP, by organizing the sequences of motivation, the problem of statement, research questions and research objectives. The design guidelines of AVOMP comprise two main facets, namely, UI design and interactivity. To achieve a successful development of the design guidelines, interactive processes are conducted based on design research design. Next chapter discusses the relevant concepts and theories as foundations in this study.

CHAPTER TWO
LITERATURE REVIEW

2.1 Software Visualization

As mentioned at the beginning, the conventional technique of visualization to complete the textbooks and lectures in teaching DSA is using drawing pictures. The time-consuming nature of drawing may also limit the number of visual examples that a lecturer can expound in a lecture. Thus, to assist, computing educators have come up with software visualization. Within software visualization itself, two broad subfields can be identified (see Figure 2.1), namely: Algorithm visualization (AV) systems visualize general algorithms (e.g., quicksort, binary tree operations), at a high level of abstraction, while program visualization (PV) systems visualize concrete programs, usually at a lower level. Within program visualization, some visualizations represent program code (e.g., dependencies or code evolution) while others illustrate the runtime dynamics of programs.

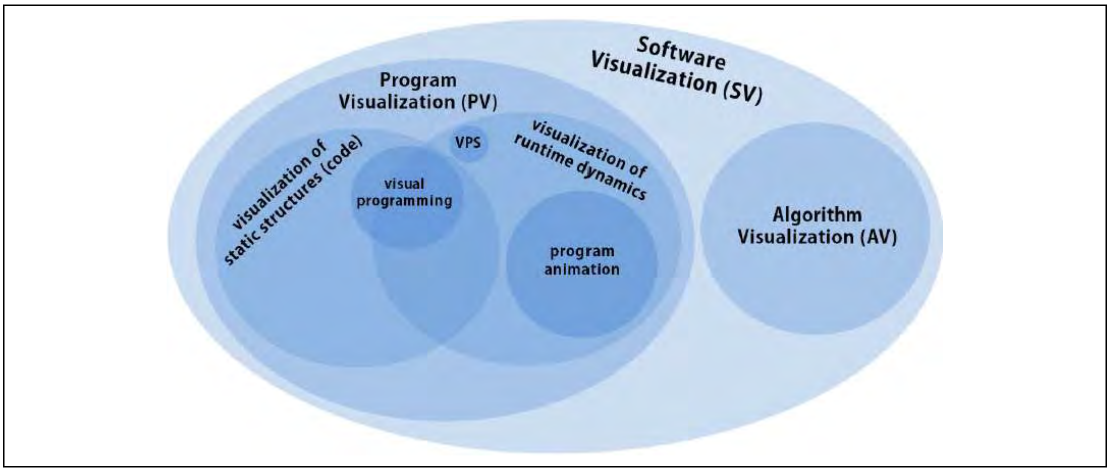


Figure 2.1. Software visualization subfields (Sorva et al., 2013)

It has to be noted that this study focuses on AV as part of software visualization's subfield. The next sub-sections elaborated these two terms.

2.1.1 Program Visualization

Program Visualization (PV) systems produce direct representations of programming structures and/or program execution phases (e.g., values of variables, internal program structures, method frames, data structures, objects etc.). BlueJ (see Figure 2.2) (Kölling, 2008) and Jeliot (see Figure 2.3) (Moreno & Joy, 2007) are the most popular systems developed for object-oriented programming. In BlueJ, students interact with classes by creating objects and calling their methods through a visual interface. The system is excellent for an introductory course in object-oriented programming with Java. However, it is not an algorithm animation system, and cannot be effectual in visualizing abstract data types and operations on them. Jeliot animates automatically the object-oriented structures of the source code written by the student, but visualizations are restricted to a subset of Java source code.

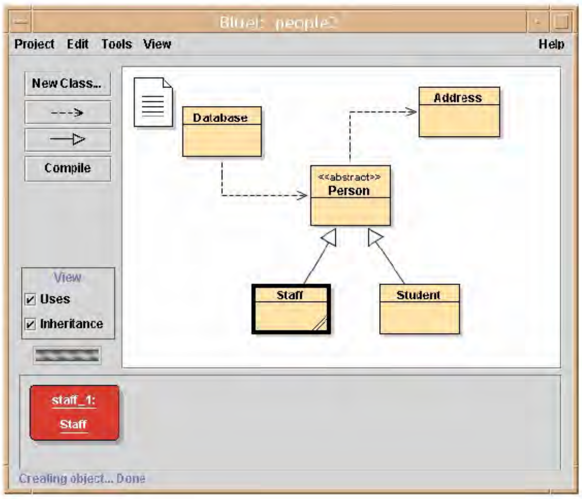


Figure 2.2 BlueJ System (Kölling, 2008)

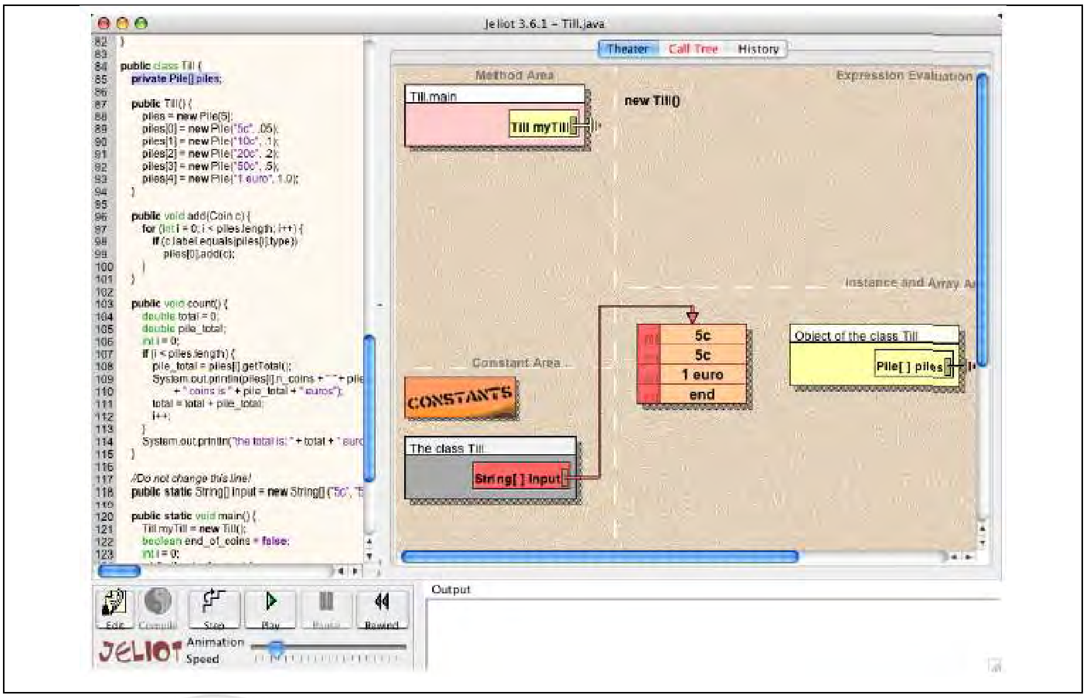


Figure 2.3. Jeliot 3 system (Moreno & Joy, 2007)

2.1.2 Algorithm Visualization (AV)

Algorithm Visualization (AV) or it is also called algorithm animation (AA) is the way to visualize the process of an algorithm in terms of operation in the algorithm as well as data (Wen, 2013). It shows the first stage of the algorithm as an image in which it will be animated accordingly based on a step by step process in the algorithm itself. This animation allows students to have a better insight towards a particular algorithm that is being presented. In other words, the logic behind an algorithm can be shown clearly from dynamic transitions of processes.

Initially in 1984, the first acknowledge tool to develop algorithm animation named BALSA (Wen, 2013). Afterward, many researchers and developers created more than hundreds of systems regarding algorithm animation, which was then used and implemented by many educators in their teaching. It is highly believed that algorithm visualization can offer a potent support of static teaching method (textbook) into having more illustration in form of animation (Hansen & Narayanan, 2013).

The example of types of the algorithm can be seen in the histogram (see Figure 2.4). This histogram is based on a survey by Shaffer, Cooper, and Edwards (2007). They listed down the names of algorithm animation systems that have been disseminated over the internet.

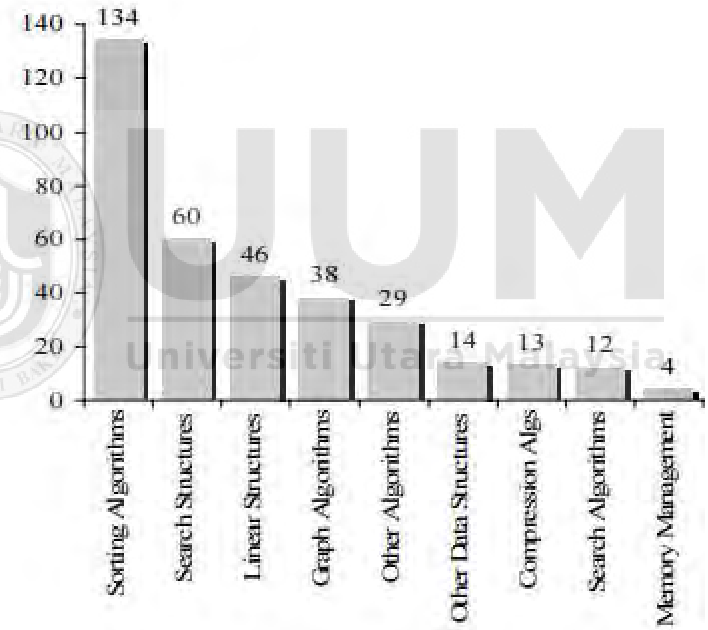


Figure 2.4. Histogram of algorithm types (Shaffer et al., 2007).

However, there are still pro and cons among researchers about the effectiveness of involving AV into a class. Most of the researchers (Karavirta, Korhonen, Malmi, & Naps, 2010; Shaffer et al., 2010; Urquiza-Fuentes & Velázquez-Iturbide, 2009b) stated that there are no significant effects towards students' performance in order to get better understanding of algorithm and engagement, if the interactivity is not included in AV system.

2.1.3 Implication of Algorithm Visualization

In nutshell, it has to be noted that this study focuses on AV that visualizes the higher level of abstraction of algorithms, while PV is more on programming structures. Particularly, AV is an approach that is used by educators or learner to help them understand the concept of DSA subject. To further understand what should be included in AV system as to help learners comprehend DSA subject, it is important to also understand what the DSA and its elements are. Thus, next section discusses DSA in more detail.

2.2 Data Structures and Algorithm (DSA)

DSA course is a theoretically requesting course in which many students of computer science face in their early course. This course is considered as one of the topics that have complexity in term of concepts and is difficult to grasp by many students. DSA is an imperative study of students in IT and ICT, but it is difficult to learn their abstract concept. The difficulty comes from the algorithm concept as well as its implementation (Osman & Elmusharaf, 2014).

According to Patel (2014), there are several problems faced by tutors or lecturers to deliver this course towards students:

i. The low motivation of student, the students cannot see the real practical implementation and the use of DSA for their future.

ii. The abstract nature of DSA concepts has made students confused to learn since the concept can be tricky in its explanation.

iii. It is hard to implement the algorithms in a real-life setting, so this has made the assignment become more difficult.

However, these all problems should not be an obstacle for learners. This is due to the importance of DSA course cannot be neglected, as it is a fundamental knowledge that every student of computer science education must have. According to Lister, Box, Morrison, Tenenberg, and Westbrook (2004), here the reasons why learning/teaching DSA subject is important:

1. **Developing the way of thinking**: DSA is explained as a transportation to develop human's mind in terms of problem-solving skills. The DSA is designed like a puzzle to be solved. Both of the processes to find the solution of the puzzle is as much as important to the solution itself.
2. **Developing programming skills**: by implementing DSA in a programming language, students can enhance their ability in programming skills, in particular, their handiness with pointers and recursion. Meaning that students can test their understanding in a programming language.
3. **Learning how the algorithm works**; nowadays the library to serve the implementation of DSA is available. Thus, if the student can understand what is going on behind the codes, they can adjust or modify the codes to fulfill their needs. The students have to be convinced that in order to master the DSA, they have to know what the processes behind the curtain.

4. **Component thinking**: This highlights the learning of students towards learning the component principle, such as the re-use of code and black box interaction. This means that students do not need to re-create the code, instead just re-use the existed components.

Thus, as a fundamental course in the computer science field, as mentioned earlier, the researchers have been attempting to find solutions by using visualization approach. The next section explains further what the visualization approach is all about. It has to be distinguished the term between program visualization (PV) and algorithm visualization (AV).

2.2.1 Data Structure and Sorting Algorithms

Data structure itself is defined as containers in a computer that store data in various style and organized functions (Mehlhorn, 2013). Whereas, the step by step process to solve the problem in the data structure, such as sort, insert, remove, search, is called algorithm (Goodrich & Tamassia, 2008). Then, those algorithms are usually presented in a straightforward way as text-based information language which is called pseudocode (Patil, 2012).

There are some sub-topics tackled in DSA, such as linked list, stack, queue, sorting, etc., but as a scope of this study, the author opts sorting algorithms which are presented as a case study. This is due to sorting algorithms are one of the most difficult sub-topic in DSA (Sutopo, 2011) .

Sorting is a process that arranges a collection of data into either descending or ascending order, which can be lexicographical, numerical, or any user-defined order (Mehlhorn, 2013).

Here, some types of sorting algorithms:

a. **Bubble sorting**: is a sorting algorithm that conceptually imitates the movement of bubbles that gradually rise in a glass of soda so that the elements of data move to their appropriate position in the array. Bubble sorting algorithm works through a list, comparing pairs of elements starting from the first element (leftmost) to the last element (rightmost). If the value of left one of the pairs is less than the right one, then the elements will not be switched. On the other way around, the elements position will be switched. This cycle will repeat from left to right for several passes based on the number of elements in the array. See the following example on how the bubble sorting algorithm working as well as its pseudo code (Figure 2.5).

Example A: 5, 12, 3, 9, 16

Pass 1

● **5**, **12**, 3, 9, 16

○ The list stays the same because 5 is less than 12.

● 5, **3**, **12**, 9, 16

○ 3 and 12 are switched because 3 is less than 12

● 5, 3, **9**, **12**, 16

○ 9 and 12 are switched since 9 is less than 12

● 5, 3, 9, **12**, **16**

○ 12 and 16 do not switch because 12 is less than 16

Pass 2

● **3**, **5**, 9, 12, 16

○ 3 is less than 5, so they switch

● 3, **5**, **9**, 12, 16

○ 5 is less than 9 so they remain in the same places

● 3, 5, **9**, **12**, 16

○ 12 is greater than 9 so they do not switch places

● 3, 5, 9, **12**, **16**

○ 12 and 16 are in numerical order so they don't switch

Pass 3

● **3**, **5**, 9, 12, 16

○ 3 is less than 5, so they do not switch

● 3, **5**, **9**, 12, 16

○ 5 is less than 9 so they remain in the same places

● 3, 5, **9**, **12**, 16

○ 12 is greater than 9 so they do not switch places

● 3, 5, 9, **12**, **16**

○ 12 and 16 are in numerical order so they don't switch

```
void bubblesort(Comparable[] A) {
  for (int i=0; i<A.length-1; i++) // Insert i'th record
    for (int j=1; j<A.length-i; j++)
      if (A[j-1].compareTo(A[j]) > 0)
        swap(A, j-1, j);
}
```

Figure 2.5. The pseudo code of bubble sorting algorithm

b. Insertion sorting: is a sorting algorithm that chooses the second element of the array to be compared with first element of the array. If the first is greater than the second element, then the swap is taken place, otherwise it will remain in the same position. The iteration will continue to go to the third element in which it will be compared to the previous elements, in this case the second and the first elements. Overall, it will be compared in backward movement, and will be arranged like the cards in the hand in ascending order. This process will be continued throughout the array until it reaches the last element. See the following example on how the insertion sorting algorithm working (Figure 2.6) as well as its pseudo code (Figure 2.7).

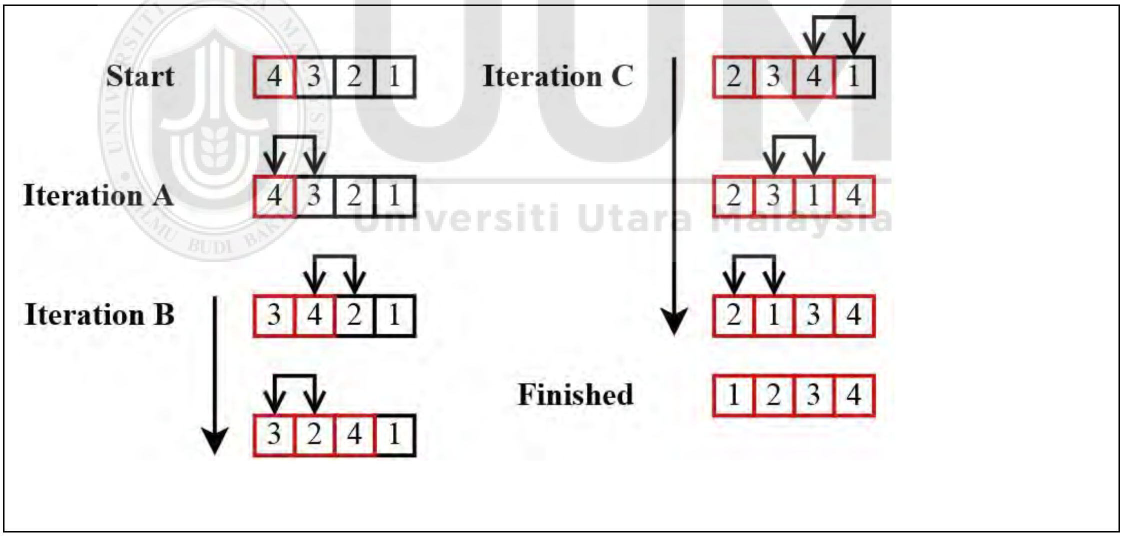


Figure 2.6. The illustration of insertion sorting algorithm processes

```
void inssort(Comparable[] A) {
  for (int i=1; i<A.length; i++) // Insert i'th record
    for (int j=i; (j>0) && (A[j].compareTo(A[j-1]) < 0); j--)
      swap(A, j, j-1);
}
```

Figure 2.7. The pseudo code of insertion sorting algorithm

c. Selection sorting: is a sorting algorithm that goes throughout the all elements of the array in order to find the lowest value to be positioned to the first element of the array. Then, it continues to scan through the whole elements to find the second lowest to be positioned to the second element of the array. It will go on and on to the next elements until it will be sorted in ascending order. See the following example on how the selection sorting algorithm working (Figure 2.8) as well as its pseudo code (Figure 2.9).

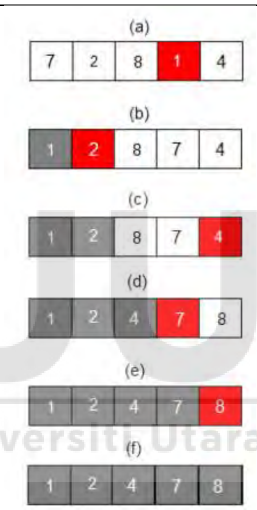


Figure 2.8. The illustration of selection sorting algorithm processes

```
void selsort(Comparable[] A) {
  for (int i=0; i<A.length-1; i++) {
    int bigindex = 0;
    for (int j=1; j<A.length-i; j++)
      if (A[j].compareTo(A[bigindex])
        bigindex = j;
    swap(A, bigindex, A.length-i-1);
  }
}
```

Figure 2.9. The pseudo code of selection sorting algorithm processes

2.2.2 The Implication of Data Structure

The section has discussed the DSA course as one of the essential courses in Computer Science faculty. The problems faced by students, the importance of this course and several types of data structure are then elaborated. The implication of this study, it is important to know what kind of elements that has to be inserted into the learning material. The arrangement of the elements, such as pseudocode and its representative visualization, have to be taken into account in this study. Moreover, to ensure the effective learning on AV, it is also noticed that interactivity factor is essential. Meaning, to be an active leaner is more encouraged than passive learner. Thus, next sections present the related theory and concepts that ground interactivity as one of the important facets in AV system.

2.3 Constructivism Learning Theory

The basic assumption of constructivism theory is that learning or understanding processes are based on human experiences itself, in order to be actively engaged with learning material (Büyükduman & Şirin, 2010). Therefore, the human can use their own construction of mentality through experiences. In other words, Constructivism necessitates the learner to construct their own knowledge through activities or interactions with learning material such as "creating, constructing, developing and inventing (Guney & Al, 2012).

Some are the basic principles of constructivism (Cooperstein & Kocevar-Weidinger, 2004):

1. Searching for meaning is the key to learning. Therefore, it is a necessity that learning needs individual perception.

2. The requirement to understand the whole picture and small pictures (parts). Thus, the main concept has a big role in the learning process.

3. The focus is positioned to the application of knowledge as opposed to learning skill of considering the content of facts.

4. The social aspect is a vital part to form the constructivist learning view. People need to learn from one another.

2.3.1 The Implication of Constructivism Learning Theory

With regards to this study, the constructivist theory impacts the AV learners through the meaningful learning content, interesting learning activities, and interesting experience of interaction. Meaning, to be an active learner is important by prompting students to learn through their own experiences using AV system. Next section explains further the strategy of interactivity aspect on AV system, which called engagement taxonomy.

2.4 Engagement Taxonomy (ET)

Engagement is a users' willingness to invest the effort to explore further and gain more information from the visualization (Boy, Detienne, & Fekete, 2015; Haroz, Kosara, & Franconeri, 2015). Moere, Tomitsch, Wimmer, Christoph, and Grechenig (2012) used the term engagement as the (perceived) effectiveness of visualizations, measured by higher participation/use of visualizations.

Increasing evidence confirm the hypothesis that student engagement is the key to the educational effectiveness of AVs (Halim et al., 2012; Hundhausen et al., 2002; Naps et al., 2002; Segura et al., 2008). The different types of engagement encountered in visualization technology have been categorized in the engagement Taxonomy (Naps et al., 2002; Rößling & Freisleben, 2002). The taxonomy defines six levels of engagement between a student and software visualization:

1. **No Viewing**: this indicates the case when the visualization is not viewed by students
2. **Viewing**: the student passively views visualization. The ability to control the animation speed or move step-by-step backward and forward. Most visualization systems support this minimal level of engagement.
3. **Responding**: the student must respond to questions about the content while viewing visualization. These are most often in form of pop-up questions where the student is required to select or type the correct answer. This type of engagement is used, for example, in Animal (Rößling & Freisleben, 2002) and JHAVE (Naps, 2005).
4. **Changing**: the student must change the visualization by, for example, providing input data to the algorithm. For example, the JHAVE system supports this in some visualization.

5. **Constructing**: the student can construct visualization. A variation on this approach is taken in MA&DA (Krebs, Lauer, Ottmann, & Trahasch, 2005) and TRAKLA2, where the student is given a DSA and is expected to simulate the algorithm. That is, they need to imitate the steps of an algorithm by controlling the visualization. This approach is also called visual algorithm simulation.
6. **Presenting** - The student is presenting a visualization to others

many studies agreed that "more is better": a mix of different engagement levels leads to better learning (Sorva et al., 2013). Therefore, the possibility of creating engaging material can be considered the most important feature of an AV system.

2.4.1 Extend Engagement Taxonomy (EET)

Myller et al. (2009) investigated the impact of engagement on collaborative learning. They extended the engagement taxonomy of Naps et al. (2002) to better capture the differences between student behaviors. They added four engagement levels (see Table 2.1), including *Controlled Viewing* and *Entering Input*, falling between the original levels of *Viewing* and *Responding* in ET.

They claimed that their extended engagement taxonomy (EET) can also be used to guide successful collaboration among students when using visualization tools. To prove their claim, Myller et al. (2009) conducted an experiment in which students were asked to work in pairs during the lab session. The students interacted with the program visualization tools BlueJ (Sanders et al., 2001) and Jeliot (Levy et al., 2003) at different EET levels. They observed and recorded all students' communication during the experiments. They found that EET levels are positively correlated with the amount of interaction. Interaction is one of the essential ingredients for a successful computer-supported collaboration along with coordination and motivation (Meier et al., 2007).

Table 2.1

The Additional Four Engagements of EET

No	Engagement Type	Description
1	Controlled Viewing (The extension of viewing level)	The visualization is viewed, and the students control the visualization, for example by selecting objects to inspect or by changing the speed of the animation.
2	Entering Input (The extension of responding level)	The student enters input into a program or parameters to a method before or during their execution.
3	Modifying	Modification of the visualization is carried out before it is viewed, for example, by changing source code or an input set.
4	Reviewing	Visualizations are viewed for the purpose of providing comments, suggestions, and feedback on the visualization itself or on the program or algorithm.

Myller et al. (2009)

2.4.2 Implication of Engagement Taxonomy

The ET and EET are used to ensure the active circumstances in learning with software visualization. In short, to include more activities or features based on these taxonomies are highly important. These taxonomies are created to make sure the better engagement of students with a visualization system. One has to note, that these taxonomies are created for visualization technology in which AV and PV can adapt these taxonomies, but of course, they are still different in terms of its implementation form. However, even though this study only focuses on AV, but related visualization studies that specifically used engagement taxonomy is still relevant to gather evidence of important interactivity aspects. Therefore, next section reports some empirical studies from both AV and PV that specifically related to the implementation of ET and EET.

2.5 Experimental Studies of Engagement Taxonomy

This section discusses some results of empirical studies related to the effects of engagement taxonomy on AV and PV software. The results are not only in terms of learning outcomes but also the behavior of students.

2.5.1 WinHipe AV Tool: Viewers vs. Builders

Urquiza-Fuentes and Velázquez-Iturbide (2007) study showed that students, who are in builders group (BG), are more spending time with WinHipe web-based AV tool (see Figure 2.10) that students in viewers group (VG). In addition, it is also discovered that builder group outperformed the viewer group in application level (see Figure 2.11).

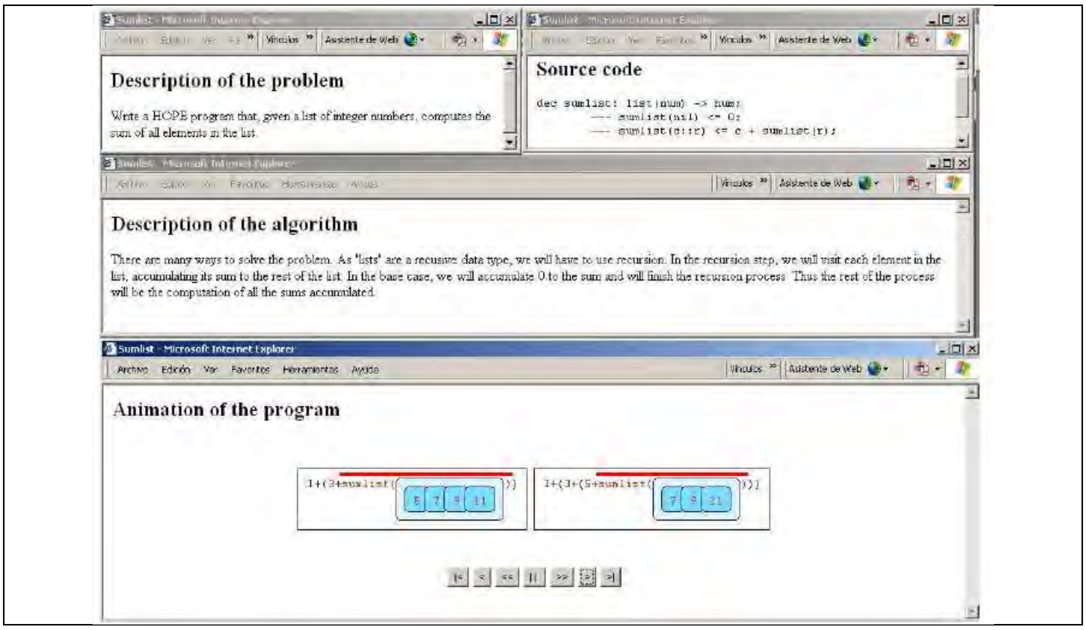
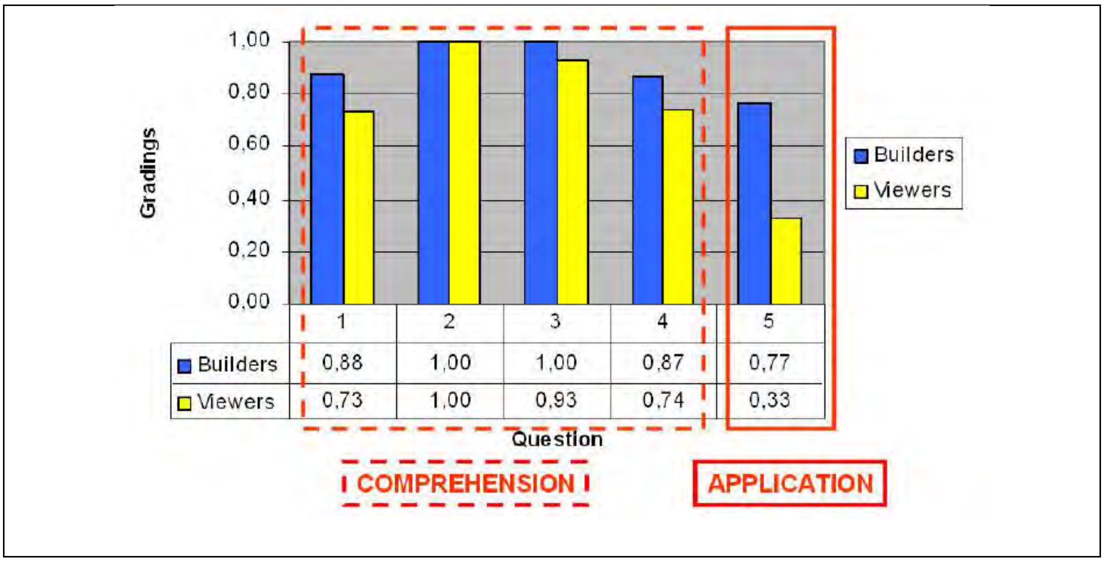


Figure 2.10. WinHipe web-based AV tool (Urquiza-Fuentes and Velázquez-Iturbide, 2007)

There were 15 participants who were involved in the evaluation using WinHipe web-based AV tool. To do this purpose, the WinHipe tool is then created with two forms for each group. The VG group has the WinHipe with viewing feature only, while BG has viewing and constructing features. The t-test evaluation was conducted through four questions that have a different level of comprehension of the algorithm. Specifically, Mann Whitney U test type was chosen. Thus, the results were significant differences (U (Mann Whitney score) = 7.000, $p < .05$) in the questions about the application level. The VG attained an average score of 0.33, whereas the BG gained an average score of 0.77, a 60% of learning improvement (see Figure 2.11).



	1	2	3	4	5
Builders	0,88	1,00	1,00	0,87	0,77
Viewers	0,73	1,00	0,93	0,74	0,33

Figure 2.11. The results of students' grading between builders and viewers group (Urquiza-Fuentes and Velázquez-Iturbide, 2007)

These empirical study results are in line with the study performed by Hübscher-Younger and Narayanan (2003) that found out the students who are constructing group (viewing + constructing) have bigger effort (lead to having more motivation) and better learning outcome than viewing group. Another thing that Urquiza-Fuentes and Velázquez-Iturbide (2007) suggested: that the utilization of narratives and textual contents incorporated with visualization could increase the effectiveness of AV.

2.5.2 Trackla 2 AV Tool: Controlled Viewing vs. Changing

Different with aforementioned study, Laakso, Myller, and Korhonen (2009) compared the learning outcomes of students who collaboratively used Trackla 2 (see Figure 2.12) AV-desktop tool on two different EET levels, namely *controlled viewing* as a control group and *changing* as a treatment group. The control group encompassed viewing and controlled viewing, whereby the treatment group encompassed viewing, controlled viewing, and changing.

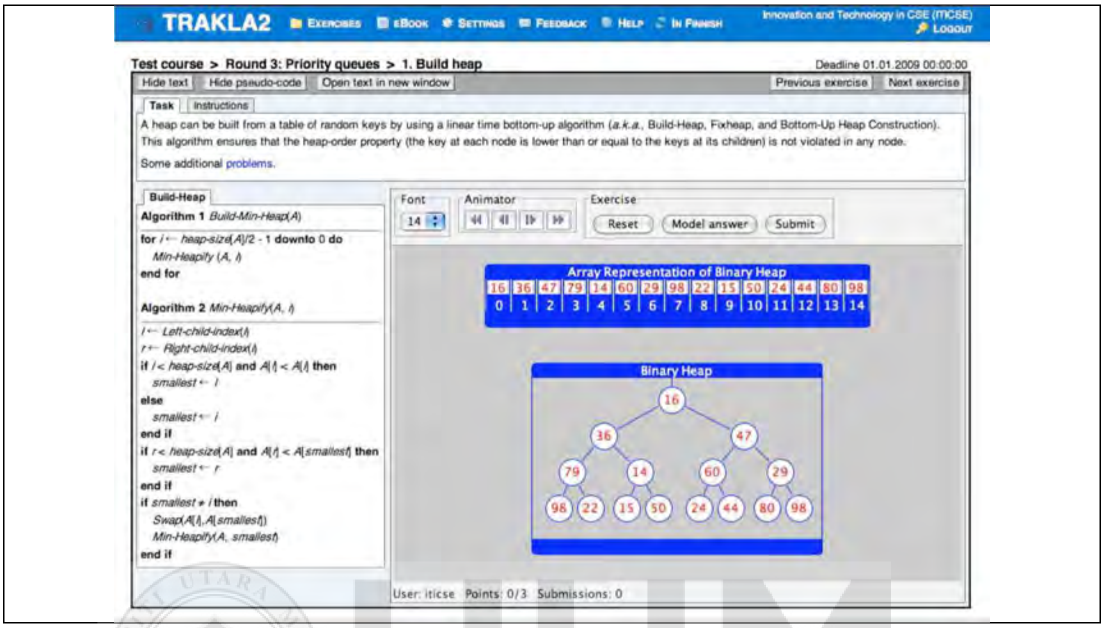


Figure 2.12. Trackla2 AV Tool (Laakso, Myller, and Korhonen, 2009)

To do so, the pre-test and post-test are conducted and the score results for pre-test showed insignificant, which means the preliminary knowledge of students was similar. Meanwhile, based on one-tailed t-test, the post-test between, both group and individual averages were significant, ($t(69) = -1.73$, $p < 0.05$ and $t(31) = -1.97$, $p < 0.05$, respectively). These results are in line with the similar previous studies performed by (Grissom, McNally, & Naps, 2003; Hundhausen & Brown, 2008; Myller, Laakso, & Korhonen, 2007; Naps et al., 2002).

2.5.3 Comparison of Students' Behavior on AV tool (Controlled Viewing vs. Changing)

Continuing from the earlier study in section 2.5.2, Korhonen, Laakso, and Myller (2009) presented another empirical study in which they researched more on students' behavior during the use of Trackla 2. Particularly, they observed the relationship between the amounts of discussion and engagement level of EET by analyzing 20 minutes of recorded audio and video from six groups of students. The same condition as stated before, the control group is *controlled viewing*, and the treatment group is *changing*.

Based on their result (see Figure 2.13 and Figure 2.14), they concluded that the higher level of engagement, the higher students' interaction and communication will be. In other words, the students become more active as the level of engagement increases. The amount of discussion is changed significantly for both control and treatment groups, especially in the state of *controlled viewing* and *viewing*. The students' silence was also dramatically declined as the level of engagement increased.

In addition, their study also observed that the students who have the number of engagement levels (treatment group), the communication and collaboration of students are further improved, which is denoted from the absent silence of the two highest engagement level (see Figure 2.14). On the other hand, the control group who has fewer engagement levels, the amount of silence is also decreased, but smaller. For instance, the *controlled viewing* level has over 30% of silence in control group, whereas it is below 10% of silence in the treatment group. The same significant difference is much more drastic between the groups in the *viewing* level.

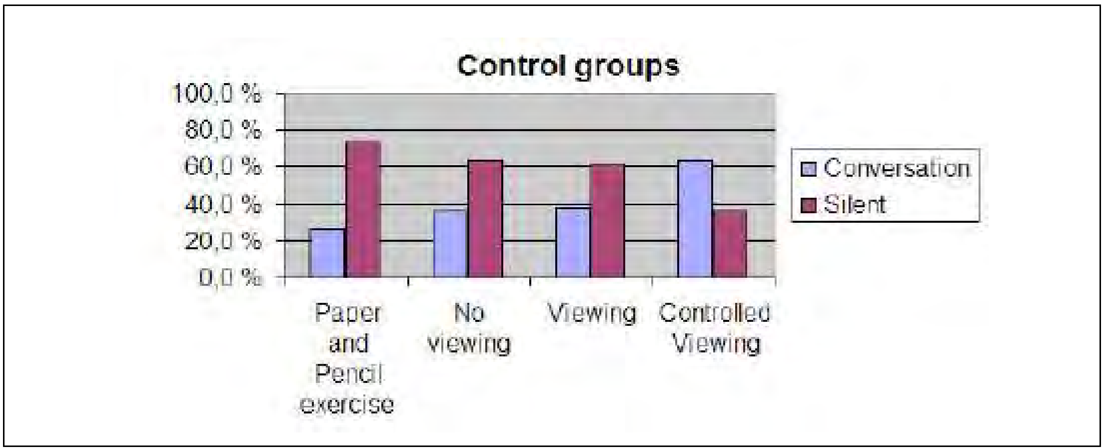


Figure 2.13. The activities' distribution in control groups for all EET-levels (Korhonen, Laakso, and Myller, 2009)

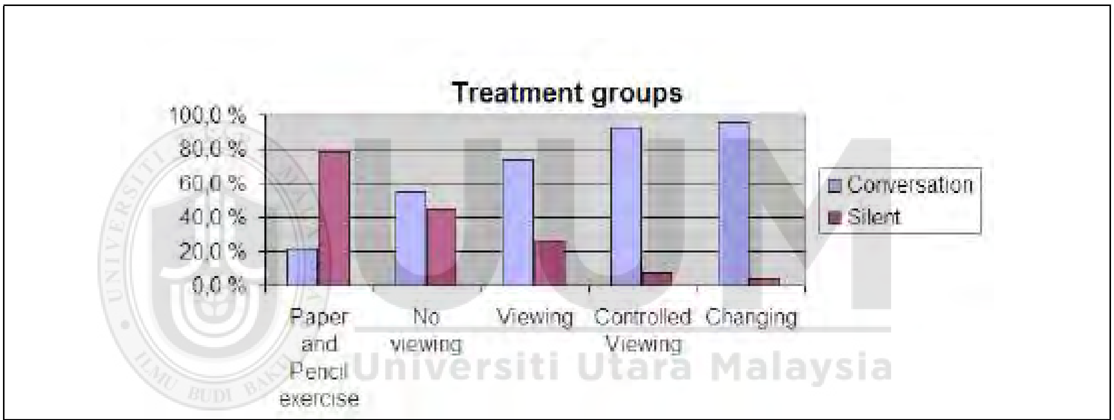


Figure 2.14. The activities' distribution in treatment groups for all EET-levels (Korhonen, Laakso, and Myller, 2009)

2.5.4 PV tool: Behavior Engagement and Problem-Solving Results (Viewing vs. Responding)

The current study by Banerjee, Murthy, and Iyer (2015) integrated PV tool in a class setting and compared the effects of two different engagement levels: *viewing* as a control group (watching visualization with commentary of instructor) and *responding* as the experimental group (predicting within visualization). The results show that *responding* group has the higher behavioral engagement of learning than *viewing* group (see Table 2.2). The analysis was based on Behavioral Observation of Students in Schools (BOSS) terminology by Shapiro (2003) that observed the students' active engagement within the class. In particular BOSS divides active engagement in form of reading aloud, raising hand, writing, discussing to peer, and talking about materials. Meanwhile, passive engagement in BOSS terminologies is reading silently, listening to lectures listening to peer regarding the work, looking only at the material and looking away or around the room. They found that *prediction* group was more engaged actively (23.33 %) than the *viewing* group (9.58 %).

Table 2.2

The Results of Active Behavioral Engagement

	Prediction Group	**Viewing Group**
Total observations	240	240
Frequency of Active Engagement (%)	56 (23.33 %)	23 (9.58 %)
Chi-square results	X^2 (1) = 4.42, p = 0.00	

The problem-solving result achievement from *responding* group is also higher than *viewing* group, it is shown by two-way ANOVA results of problem solving (see Table 2.3).

Table 2.3

Two Way ANOVA: The Rate of Problem Solving (Responding Group)

Dependent Variable: rate of problem solving			
Independent variable	***df***	***F***	**Significance**
Level of engagement	1	23.93	0.00
Prior exposure	1	58.45	0.00
Engagement level * prior exposure	1	38.08	0.00

2.5.5 PV Tool: Learning Results (Responding vs. Constructing)

Another empirical study by Derus and Ali (2015) that compared the use of PV tools in two different engagement levels, namely *responding* and *constructing*. There were 84 students divided into two respective groups: responding group and constructing group. The responding group consists of viewing and responding engagement level, whereas constructing group covers the same, but has two additional engagements, which are changing and constructing.

Initially, the students took the pre-test session within 60 minutes. Then, the 5 times lab-sessions within 3 weeks are conducted. The session is then finished by post-test session. By using t-test, their finding showed that *constructing* group has higher result that *responding* group (see Table 2.4). The difference of the results from both groups also showed statistically significant (see Table 2.5).

Table 2.4

Descriptive Statistic

	Strategy	N	Mean	SD
POST	Responding	42	63.76	7.947
	Costructing	42	68.29	10.013

Table 2.5

Independent T-test Result

	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference
POST	-2.293	82	.024	-4.524	1.973

2.5.6 AlgoVis AV Tool: (Viewing vs. Changing)

This experimental study is performed by Avancena and Nishihara (2015) that attempts to compare the use of two different conditions of AlgoVis tool (see Figure 2.15), namely, group A (AlgoVis 1 with more options) and group B (AlgoVis 2 with limited options). Specifically, AlgoVis 1 has additional changing features, while AlgoVis 2 only can view. Thus, the students with AlgoVis 1 can change the height of arrays in AV tool, while AlgoVis 2 can only see the same height of provided arrays.

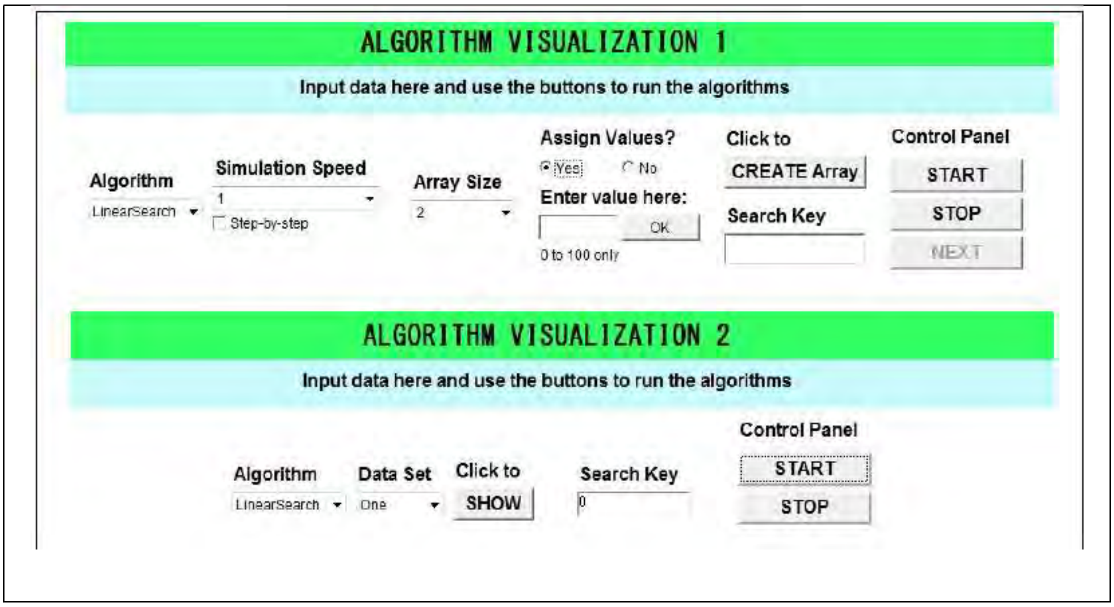


Figure 2.15. AlgoVis1 dan AlgoVis2 control panel (Avancena and Nishihara, 2015)

The pretest and posttest were then conducted in their study. The test itself is divided into three parts. The first part is about identification that requires students to identify the algorithms itself as well as their attributes. The second part is about coding category that requires students to filling the blank space in coding format within a particular algorithm. The third part is about demonstrating that requires students to demonstrate manually step by step of a particular algorithm and also show its results' output. Each part has 10 points, so the maximum for all is 30 points.

After students finished all parts, the *paired-samples t-test* was used to measure the general improvement of all students after using AlgoVis, comparing the scores between pretest and posttest. Then, to compare the scores between group A and B, the independent-samples t-test was used.

The results showed that group A has better mean scores in posttest, even though the pretest showed on the contrary (see Table 2.6). Then, by subtracting the result between mean scores of posttest and pretest, the gains scores resulted. They found that students in group A have better results of gain scores for all part of the test that group B (see Table 2.7).

Table 2.6

Group A and B results (mean scores)

Group	Algorithm Test	N	Mean	Std. Deviation
A	Pretest	17	11.82	1.718
B		15	12.33	2.298
A	Posttest	17	20.06	1.848
B		15	18.60	2.079

Table 2.7

The Gain Scores for Both Groups

Group	AV Tool	Gain Scores (Posttest - Pretest)			
		Total Score	Part I Identification	Part II Complete Code	Part III Simulation
A	AlgoVis1	8.24	3.00	2.29	2.94
B	AlgoVis2	6.27	1.80	2.27	2.20

2.5.7 Implication of Experimental Studies of Engagement Taxonomy

The implication of empirical studies of engagement taxonomy is that they are needed to strengthen and justify each guideline formed in terms of interactivity aspect. Therefore, the related experimental studies about engagement taxonomy are gathered from AV and PV studies. Besides, they are also needed to explore what the appropriate recommendation of interactivity to this study. At the same time, these studies showed that the more interactive functions that system has, the more effective the students' performance will be. Therefore, this study takes initiative to include all active features from engagement taxonomy. Next sections discuss some theories and concept from another important perspective for the proposed design guidelines of AVOMP, which is UI design.

2.6 Perspective of Human Visual

According to Ware (2012), the highest bandwidth belongs to a visual display, which has a channel to transfer information from display to human. In other words, the vision has the biggest impact to get information, compared to all combination of other senses. More than 20 billion neurons of the human brain are together analyzing pattern through visualization display, which has the enormous responsibility for cognitive activity. Therefore, it has to be careful to present visualization display towards the human.

One of the significant features that human vision has is pre-attentive processing capability (Deller, Ebert, Bender, Agne, & Barthel, 2007). This is related to what kind of visualization features that can catch our attention unconsciously, meaning we do not have to pay attention to it. This feature is significantly efficient for the searching method, and as a result, makes lower in cognitive load. The following sections particularize further about the features.

2.6.1 Gestalt Theory in Interactive Design

In the beginning of 20^{th} century, the Gestalt theory was greatly disseminated by three scientists from German, which is Köhler, Koffka, and Wertheimer (Jackson, 2008). In the German language, “gestalt” means “patterns”. So, in the simplest definition, the gestalt principle is the way on how human recognize or identify patterns. The patterns here also mean grouping; this theory explains the scientific method on how human perception works from the tendency of grouping properties in visualization design.

The rational explanation is also explained through this theory that the changes of configuration, spacing, and timing can have profound impacts on the meaning of visualization. Figure 2.16 shows the differences in spacing which results in different interpretation. The design will have a wrong interpretation from users if the designer ignores this law (Graham, 2008).

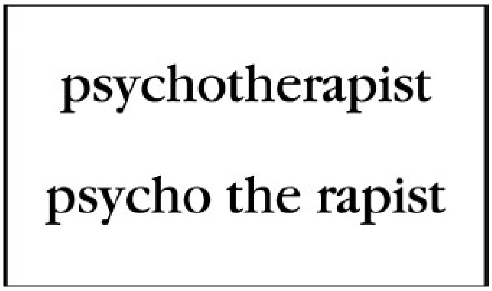


Figure 2.16. The gestalt theory of different spacing is different meaning

In relation to this study, as the focus is in interactive design of AV system, it is important to follow the law of gestalt principle. Gestalt theory hypothesizes that information is assembled into components as per the following standards (Guberman, Maximov, & Pashintsev, 2012):

1. ***Proximity***: this implies that we have a tendency to gather components as per their closeness to each other and the patterns that they structure.

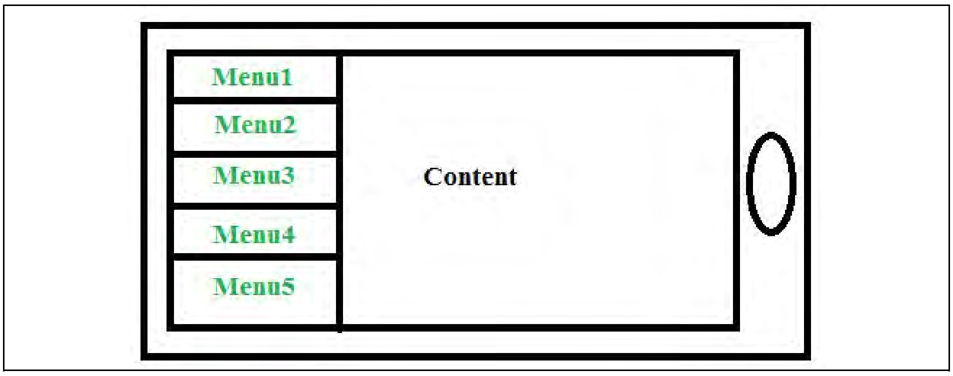


Figure 2.17. The proximity law in mobile app design

Figure 2.17 is the example that shows the menu group (green-text links) is perceived as a group due to its proximity in space and box. It is important to organize the same items into a unified group that is separated by a gap of space or line with other groups, or else the readers will find difficult to read. Another way is to help readers to distinguish between the primary links and secondary links (see Figure 2.18).

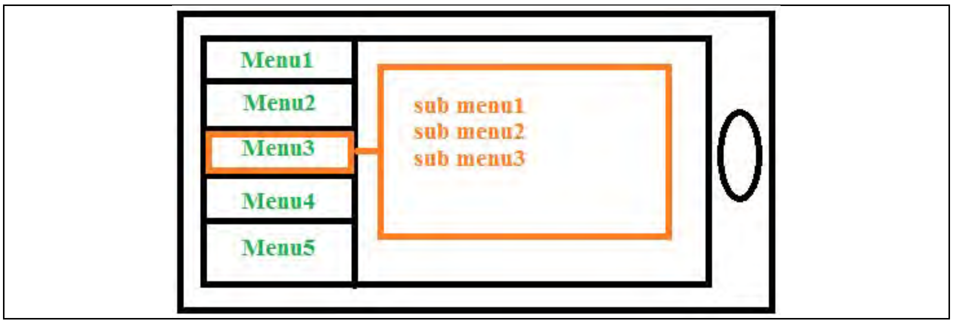


Figure 2.18. The differences primary and secondary links by proximity law

2. ***Similarity***: this suggests that we have a tendency to gather together things that are same in several things. The visual elements with similarity in terms of color, proximity, direction, shape, and size are perceived as the same group (see Figure 2.19), or even they are disconnected in space (see Figure 2.20).

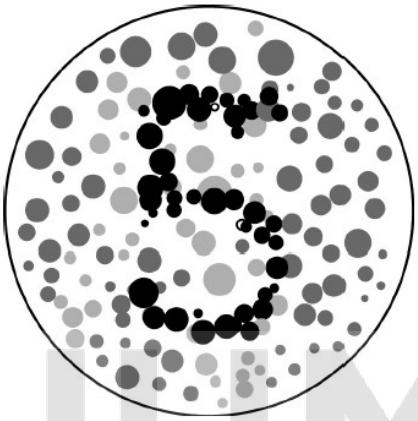


Figure 2.19. The similarity law in color, perceived as a group.

Figure 2.20. The similarity color in space

In relation to interactive design, it is necessary to help users distinguishing the group of the item either conceptually or physically by keeping the similarity law in links, texts, and animated elements. The items that move in a similar way are also counted as a group element (see Figure 2.21). This figure is an example of the motion of expanded link animation. The same movements of the animation of links are indicated as a group.

Figure 2.21. Similarity on motion or movement are perceived as a group

3. ***Closure***: this implies that we have a tendency to gather things together on the off chance that they appear to finish some element. This law explains that human has tendency to close the gap, particularly in a familiar form. Even if the information presented is missing, a human can still observe by filling or closing the gap and see the real form because of the predicted or familiar patterns. In Figure 2.22 the users can still recognize the leaf even after the missing part of it.

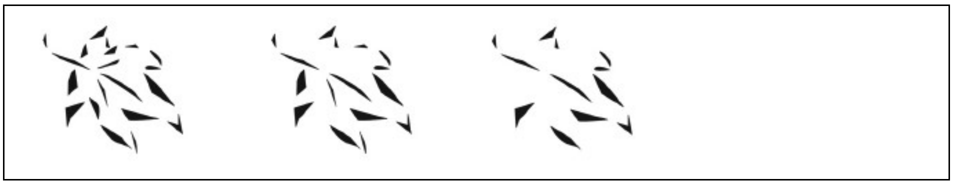


Figure 2.22. The example of closure in leaf animation

Another example of animation movement is in Figure 2.23. In this figure, a human can identify that the ball movement exhibits the shape of a circle, despite the motion is incomplete (just half of the circle or even less).

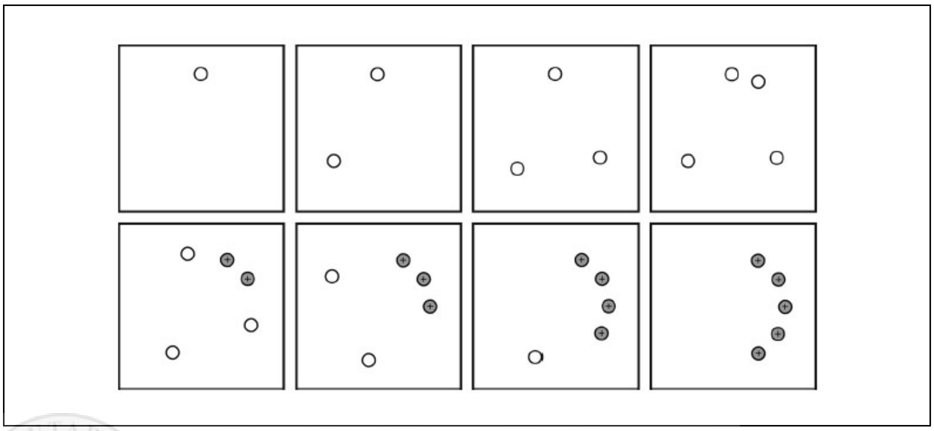


Figure 2.23. The ball movement of closure example

4. ***Continuation***: this principle shows that human vision is more distracted to follow (continuation occurred) the relationship between shapes, such as curve, lines, animation motion (sequence shape), and so forth. This principle is used to direct users' attention to particular content (see Figure 2.24).

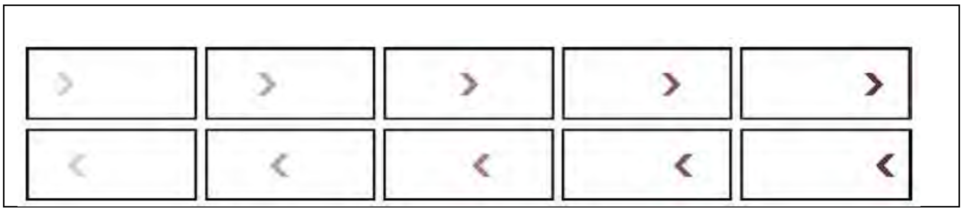


Figure 2.24. The continuation law from faded arrow animation

In conclusion, it is important to include gestalt principle into two-dimensional design or interactive design in this study, to enhance composition and the visual communication, remove miss-interpretation, and organize information well.

2.6.2 Safe Color

The color is one of the features that have value in visualization. The recognition of types or attribute of an object is determined by its color. For instance, the use of color can be as a label, as quantity, as representation, and also as beauty (Stone, 2004). However, the implementation of color must be selected or design carefully. As stated in Tufte's principle in color design is "do not harm" (Stone, 2006). Accordingly, Ware (2012) has listed colors that can be selected safely for labeling information on computer or phone devices (see table 2.8).

Table 2.8

Safe Colors for Computer and Phone Screen

Red	**Green**	**Yellow**	**Blue**
Black	White	Pink	Cyan
Gray	Orange	Brown	Purple

2.6.3 Implication of Perspective of Human Visual

As far as UI design is concerned, the related human visual theories need to be taken into account. The Gestalt theory is discussed to know the tendency on how our visual transfers and processes it to our brain. The mistakes of wrong patterns of UI design, it will lead to a wrong interpretation, and thus will affect on cognitive learning. The same goes with color, it is important to see what kind of appropriate color that does not harm human eyes.

2.7 Cognitive Load Theory

Cognitive load theory (CLT) is an underlying theory that shows how to design a learning material according to the perception of the human brain (Sweller, 2011). The fundamental idea of this theory is that the human brain has a limitation in cognitive capacity, and thus the required learning material has to be optimized to reduce working memory learning process (Paas & Sweller, 2014). The working memory process, is also called short-term memory, has a responsibility to process the retrieved information. CLT strongly declared that the learning process will be hampered if cognitive load material has gone beyond the limit of human cognitive load capacity (Rubel & Wallace, 2013).

Hence, it is very significant to guarantee that material contents of AVOMP that is presented in this study follow the rules of CLT. The followings are three types of cognitive load in instructional design (Leppink, Paas, Van Gog, van Der Vleuten, & Van Merrienboer, 2014):

1. **Extraneous Cognitive Load**: is a cognitive load that is elicited because of extra materials that do not contribute to the learning process, such as movement and sound. This means as a designer of courseware interface, it is very important to ensure that learning materials are presented without inappropriate additional elements. This is due to the students will be much distracted, if the extra elements are added in conveying knowledge. The focus is separated; causing the cognitive load is overloaded.

2. **Intrinsic Cognitive Load**: unlike the extraneous load which is evoked by extra materials, this cognitive load is triggered due to the complexity of the concepts itself.

3. **Germane Cognitive Load**: is cognitive load based on construction scheme. Meaning, students are encouraged and engaged towards the problem-solving scheme. Breaking down a complex problem into a step by step process to be understandable. This involves students to focus on the process of problem-solving that has been broken down in sequence before finally achieving the solution of the problem.

2.7.1 Visualization and CLT

As far as the visualization is concerned in this study, this section provides the strong linkage between visualization and cognitive load theory. Basically, visualization is created in order to represent information as simple as possible by using visual aids (image and graphics), rather than to burden cognitive load with a lot of unstructured information (Block, 2013). This gives the learner deeper and simpler awareness towards the representation of knowledge.

However, even though the visualization had helped to reduce cognitive load itself, there is still a probability in which the environment surrounded could have a negative impact on the learning process. As illustrated in the first part of Figure 2.25 by Tudoreanu (2003), it is shown that users have different channels in learning, which is visualization and source code manipulation. This can give an effect of diversion on cognitive load. Meanwhile, the second illustration incorporates the visualization and source code within direct manipulation method in the same channel, so that users still can see it clearly on what is going on through visualization.

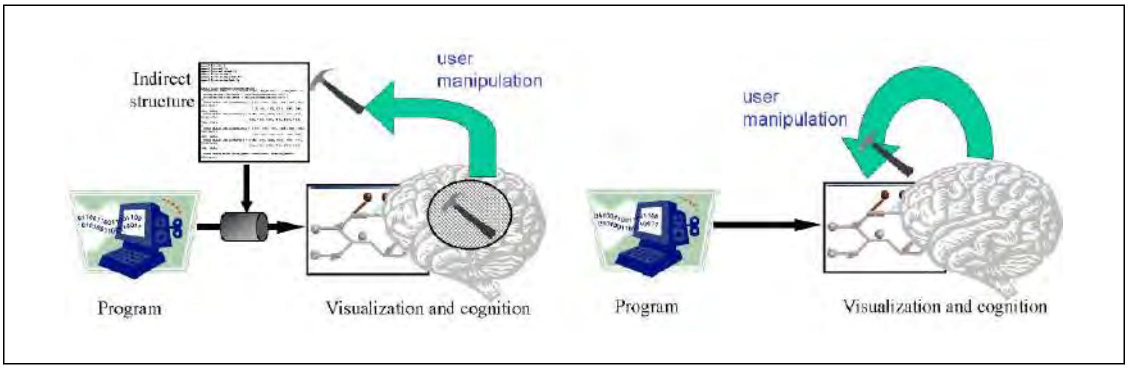


Figure 2.25. Cognitive load illustrations in different ways (Tudoreanu, 2003)

Hence, as discussed earlier, it is very important to identify various factors, so that the negative aspects can be averted in presenting learning material based on cognitive load theory. To keep in mind, that CLT emphasizes on designing learning material as simple as possible so that the system presents visualization without unnecessary or inappropriate extra elements that can turn over being ineffective courseware as learning aid tool. Next, the following sub-section deliberates cognitive load theory of multimedia learning.

2.7.2 CLT of Multimedia Learning

In order to present information regarding the cognitive load theory, Mayer's model shows the use of multimedia materials based on "dual channel model" (see Figure 2.26) (Mayer, 2014). Mayer's model advocates that multimedia materials (words and pictures) are supposed to be presented in two different channels simultaneously, namely verbal channel and pictorial channel (Ando & Ueno, 2008).

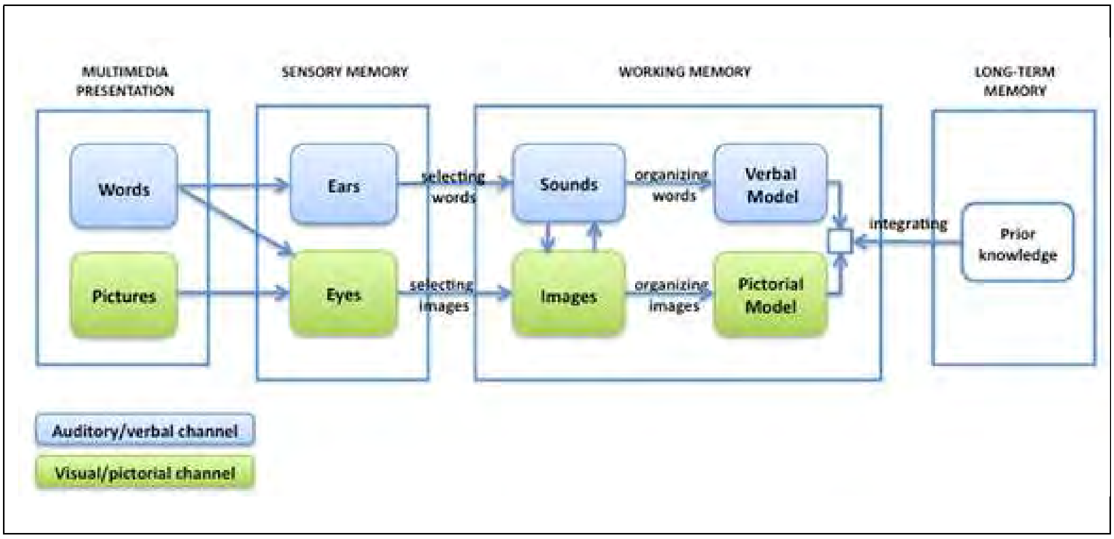


Figure 2.26. Multimedia Learning Model by Mayer (Mayer, 2014)

This model is built by Mayer according to three assumptions, as follows:

1. **Dual-Channel Assumption:**

   First and foremost, multimedia presentation in form of words and pictures is captured in sensory memory, either by ears or eyes. Then, this is due to human verbal channel and the pictorial channel is disjointed in working memory, the presentation of them has to be organized in a parallel way. This way has made the memory can integrate the channels each other with related images and sounds. Specifically, the presentation of images or pictures is supported concurrently with narration (sounds) as an extra explanation. Lastly, the information will be integrated together with prior knowledge within a long-term memory. He advocated that this integration is not only reducing cognitive load but also can enhance or strengthen understanding while learning. This dual channel concept is also used to avoid the split attention effects.

In general, split attention effects can occur if the learning materials which have two different types are displayed in the unsuitable way (Ayres & Cierniak, 2012). The idea is based on the previous discussion, the audio and picture channels should have to be integrated simultaneously, to ensure that both of information is received at the same time without being split. The strategy is to make verbalize instructional audio to be useful as an extra explanation for a given picture. Meanwhile, if the instructional is in form of written text, the text should be presented nearby the given picture, rather than in a different area. This could release working memory to just pay attention towards one focus only.

2. **Limited Capacity Assumption:**

This second assumption stated that each channel has limited capacity at any given time in learning information processing. According to Alhosban and Hamad (2011), working memory can only have approximately seven pieces of information in the average at a certain time. This means that instructional learning material should be designed carefully in terms of the amount of information.

3. **Active Processing Assumption:**

To keep actively engaging learners with the learning environment, this third assumption suggested making them linking, organizing, collecting, and integrating the newly received information with their prior knowledge. This kind of way is able to develop an effective mental model of that information (Hanley, 2010).

2.7.3 Multimedia Learning Principle

In addition to CLT, Mayer (2009) concluded the twelve principles for organization design in a multimedia presentation, as follows:

1. **Coherence Principle:** Students learn better if the irrelevant information is omitted, such as words, images, sounds, which will have no impact in knowledge.

2. **Signaling Principle:** Students learn better if the cue is added in order to inform them further about particular information.

3. **Redundancy Principle:** Students learn better from pictures and narration, either in text form or sound narration; but both of the narration should not be combined on screen at the same time.

4. **Spatial Contiguity Principle:** Students learn better when the corresponding words and pictures are displayed close each other, rather than far on the screen or even page.

5. **Temporal Contiguity Principle:** Students learn better when the presence of corresponding words and pictures are displayed simultaneously, rather than in sequence.

6. **Segmenting Principle:** Students learn better when the multimedia presentation of the lesson is displayed in segments rather than continuously.

7. **Pre-Training Principle:** Students learn better when multimedia lesson shows the names and characteristics of the main element.

8. **Modality Principle:** Students learn better if the animation is explained through narration rather than written text; however, in our case of study. the AV should be explained by pseudocode as one of the elements of the AV itself.

9. **Multimedia Principle:** Students learn better from both images and words, rather than to show words alone.

10. **Personalization Principle:** Students learn better from multimedia presentation if the lesson is presented as conversational style, instead of formal style.

11. **Voice Principle:** Students learn better when the voice narration is spoken in a way that is human-friendly, instead of using machine voice.

12. **Image Principle:** Students will not necessarily learn better from the additional spoken character image.

2.7.4 Implication of Cognitive Load Theory

As this study concerns to create design guidelines of AVOMP, it is required to look up into cognitive science theory that shows how the brain learns something concepts. The perception of the human brain is depicted from CLT. The thing about CLT is to make sure that AVOMP is presented with a scheme and is avoided from a lot of distractions that make human brain overloaded to learn something. In addition, the multimedia learning principle will be also considered as important aspects of this study.

2.8 Mobile Learning

As far as this study focuses on mobile learning, therefore it is important to know the definition as well as it's the content of mobile learning itself. As a concept, mobile learning has various definitions at present (Crescente & Lee, 2011). The different camps contend that mobile learning is a "a free train" or "subset of e-learning," or a "horizontal in the distance learning universe" (Crescente & Lee, 2011; Mostakhdemin-Hosseini, 2009a). While understanding the mobile learning as an idea is still in a beginning stage, usage rapidly got the consideration of K-12, higher organization, and government and business agencies, as it gives numerous affordances that changed the customary idea of learning (Su, Liu, & Lee, 2011). Mobile devices have changed the human being approaches and devour information, and as a consequence, the way human learns (Su et al., 2011). Traxler (2012) proposes that in the arrangement of instruction, mobile learning can be portrayed as a particular project that some recommend could hamper the congruity of the present education system. The concurrence with this recommendation may rely on upon membership for the meaning of mobile learning.

For instance, in the beginning, a number of which define mobile learning as instruction, education, or learning opportunities via content delivery that utilize handheld and mobile learning as the solitary or prevailing innovation (Crescente & Lee, 2011; Demirbilek & Demirel, 2010; Ismail, Baharum, & Idrus, 2010; Kukulska-Hulme et al., 2011; Pimmer, Pachler, & Attwell, 2010). As the idea developed more consideration was paid to the universal affordances of mobile phones (Crescente & Lee, 2011; Son, Park, & Kim, 2011). Uzunboylu and Ozdamli (2011) defined mobile learning as "a sort of model of learning that enables learners to acquire learning materials anytime and anyplace utilizing mobile innovation and the internet". Terras and Ramsay (2012) noticed that the day in and day out access gave by mobile technology permits clients to participate in at whatever time learning and person to person communication. Pea and Maldonado (2006) and Park (2011) included the mobile technology empowers learners to work at one of a kind exercises in ways that were formerly unimaginable. (Crescente & Lee, 2011) further affirmed that mobile learning is universal as far as the now broad accessibility and flexibility of mobile phones, depicting the concept of anywhere.

Attwell et al. (2003) recognized mobile learning opportunities as recently offering contextualized learning access for those with constrained access to customary instructive. Mobile learning surmounts conventional space and time imperatives, while regularly upgrading the connection and circumstance of both formal and casual learning (Najima & Rachida, 2008; Traxler, 2012). Through the emerge of wireless technology, mobile learning can now enhance learning and teaching effectiveness, due to its accessibility and flexibility (Ismail et al., 2010).

Mobile learners can increase their understanding of knowledge with arranged and contextualized rehearsal. In other words, "mobile learning is considered as 'the procedures of coming to learn through discussions over various settings among individuals and personal interactive advancements" (Pimmer et al., 2010; Sharples, Arnedillo-Sánchez, Milrad, & Vavoula, 2009). Pimmer et al (2010) noticed the movement from a technical conception of meaning-making via innovation to an instructive one. They declared that the discussion is moving towards looking at mobile learning through cultural, social, and psychological lenses. Seemingly, in every case when one may argue in the level of debate, mobile learning exists at the crossing point of various key elements (Figure 2.27) (Koole, 2009).

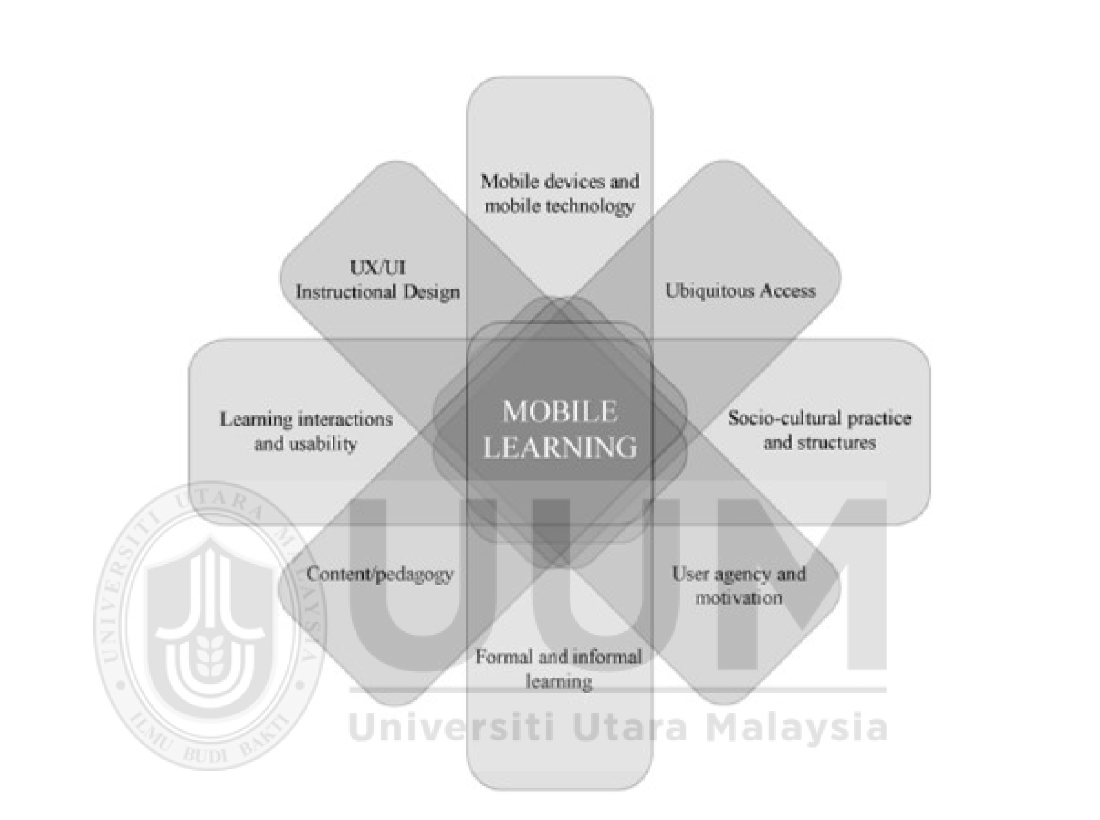


Figure 2.27. Key elements of mobile learning (Koole, 2009)

Based on (Traxler, 2007), mobile learning is happening since mobile phones are changing human comprehensions of community, space, group, and discussion. It permits learners to take part in customized, synergistic, and interactive learning and has empowered instructors to educate and convey in inventive manners through the novel attribute of the gadgets (Demirbilek, 2010).

Some studies noticed that specific component, such small screen display and constrained web access, can exhibit challenges for specific users (Crescente and Lee, 2011; Y. Park, 2011). In spite of these constraints, Crescente and Lee stated that mobile learning "may turn into a mode of decision with learners since present and future eras won't know existence without complex electronic technology" (p. 112).

Mobility grants more prominent authority over learning experiences (Zabel, 2010), accordingly permitting learners to have benefits of gaps of downtime (Valk, Rashid, & Elder, 2010). Mobile learning additionally gives substitute methods of conveyance and ideal security for learners (Crescente and Lee, 2011). Not like customary models of instruction that work by exchanging knowledge from educators to understudies, mobile learning engages learners to take part in the learning procedure and effectively develop their own learning (Valk et al., 2010). Consequently, the learning affordances of mobile learning incorporate flexibility, immediacy, expediency, portability, accessibility (Crescente and Lee, 2011; Terras and Ramsay, 2012), network accommodation, individuality, interactivity, and cross-context learning (Eliasson, Pargmann, Nouri, Spikol, & Ramberg, 2013a; Park, 2011). The conceptions of expediency and immediacy also make conceivable just-for-me, just-in-case, just-enough learning, and just-in-time (Park, 2011; Traxler, 2007). Mobile learning additionally encourages collaborative and peer to peer learning both in individual and virtually, permitting people and gatherings to make and share artifacts of learning free of time and space (Graham Attwell, 2010; Eliasson et al., 2013; M. Wang and Shen, 2012).

As far as learning advancements, some studies have discovered confirmation through quantitative strategies with pre and post-test evaluation that shows mobile learning produces momentous learning advancements (Basoglu & Akdemir, 2010; Chandran, 2010; Che, Lin, Jang, Lien, & Tsai, 2009; Chen & Chang, 2011; Delgado-Almonte et al., 2010; Hwang et al., 2011; Liu, Geurtz, et al., 2013; Wang & Wu, 2011; Wu, Hwang, Tsai, Chen, & Huang, 2011).

2.9 Implication of Mobile Learning

In this study, it is vital to know the definition of mobile learning across literature as well as its influence in learning and teaching at hand. Mobile learning includes flexibility, immediacy, expediency, portability, accessibility. Previous findings also confirmed the significant results by applying mobile learning in the study environment. Therefore, this study applies AV system to mobile learning. Next, previous AV design guidelines are reviewed and critiqued in the following section.

2.10 Mobile UI Design Guidelines

To ensure the successful user experience of AV system, the user interface is one of the main components to be taken into consideration. This section discusses the design guideline perspective, which comes from general design guideline first, and then go to more specifically from user interface guideline in a mobile application. Some studies noticed that specific components, such small screen display, can exhibit challenges for specific users (Crescente and Lee, 2011; Y. Park, 2011). Therefore, the discussion towards this probability problem is important.

In general, Meyers (2004) concluded that the most important design guideline is to make the interface easy to use correctly and hard to use incorrectly. He also stated that the error from the user interface belongs to interface designer, not from interface user. Thus, to achieve the goal of this guideline, there are two points have to be considered:

1. The interface designers should try to explore all possible ways that are able to result in incorrect behavior from the system.
2. They must find solutions to prevent all that errors from occurring.

For instance, the user interface that allows or has the possibility for users to make mistakes activity can be seen in Figure 2.28. This user interface drop-down box is about date input from airplane booking ticket website. The users can mistakenly choose the wrong date, such as 31 of February. Thus, it is important to choose what kind of method or user interface that can remove this kind of inadvertent activity.

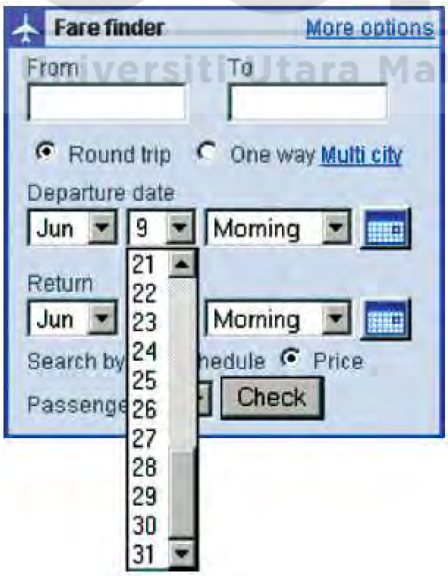


Figure 2.28. The drop-down box of date input

In conjunction with this study, it is highly deliberated to anticipate an AV system on mobile platform from incorrect respond from system and activity from users. The appropriate user interface has to be chosen carefully; otherwise, the users will be guided to incorrect activities.

2.10.1 Mobile User Interface Strategy

According to Seong (2006), there are four usability guidelines of mobile applications interface:

1) Small Screen Display

In regards to design and arrange various information effectively on a small screen display is really challenging tasks (Repokari, Saarela, & Kurki, 2002). The limitation of small screen can affect to other limitations (Repokari et al., 2002; Seong, 2006). For example, the difficulty of reading process can be affected in which the normal eyes' movement pattern is forced to be changed. The content and information showed on the small screen have to consider the smaller text effect as well as the reading speed. This means the information is highly recommended organized into small chunks in order to arrange the information in the more effective way (Seong & Chee, 2006).

According to (Page, 2014); Seong (2006) and Page (2014), the scrolling and touching frequency by users should be decreased. Moreover, the application's screen page should not bigger than the screen of mobile devices.

2) Relevant Information Display

The irrelevant information is able to make novice learners confused (Fling, 2009; Lee & Grice, 2004). The relevant information has to be shown due to the limitations and restrictions resulted from the size of mobile screen devices (Seong, 2006). For the readability aspect, the most important information is recommended to be placed at the right top corner (Neil, 2014). The design of blank or empty spaces has to be arranged properly to prevent the confusions of users (Seong, 2006).

3) Navigation

Consistent navigation structure is really important to determine a success of mobile application on small mobile devices (Seong & Chee, 2006). This way can maintain and support satisfaction of learners when using a mobile application (Seong, 2006). Lee and Grice (2004) and Buchanan et al. (2001) suggest two important aspects to create more fluid navigation, which is reducing the users' touches and converting text input method into the selection of menu list.

4) Consistency

According to Holzinger (2005), one of the most significant aspects of usability principles of a user interface is consistency. In a mobile application, the consistency in designing interface is really an important consideration due to the limitation of mobile screen size (Seong, 2006). Therefore, the location of contents on user interface has to be the same, guarantying the consistency of user interactivity in each page (Seong, 2006). For example, the location of a similar button on the first page has to be the same on other pages.

2.10.2 Design recommendations for Small Screen Device

The notion of some studies claimed that design guidelines are required to guarantee the successful design and development of the mobile-learning application (Al-Zoubi, Alkouz, & Otair, 2008; Crescente & Lee, 2011; Molina, Redondo, Lacave, & Ortega, 2014; Mostakhdemin-Hosseini, 2009b). Designing is currently turning out to be more entangled, as designers of constructional materials should now think about applying as an assortment of design guidelines in aspect of design and assessment of instructional media (Wang & Shen, 2012), such as taking into account users' perspective, learning, communication and architecture of systems (UI/UX).

Crescente and Lee (2011) noticed such guidelines are expected to direct not just in terms of the pedagogical learning styles proficiency, and procedures, additionally the andragogical ones. Wang and Shen (2012) proceeded with, "M-learning must surmount some center difficulties, keeping in mind the end goal to have a significant effect on the overall educational atmosphere". Some studies have drawn closer the topic of designing with regards to learning environment from small screen device (Churchill, 2011; Liu, Geurtz, et al., 2013; Seraj & Wong, 2014), but designing interface has various components, any blend of which may give a distinct result. Here some design recommendations for small screen displays, as follows:

1. **Maximizing space:**

   As far as the small screen design is concerned, the more proportion of the screen used, the more places are accessible for information (Churchill, 2011). Thus, it is suggested to make a mobile application designed in full-screen mode, avoiding the mobile web-browser format (containing headers and footers) that can take over a precious space (Churchill, 2011; Churchill & Hedberg, 2008). Seraj and Wong (2014) added that the application should be fixed in mobile devices display, not to be wider or taller, otherwise it could force learners to make a gesture scrolling up/down or left/right to see the whole content. This is due to the scrolling has been proved to trigger split attention effect (Luong & McLaughlin, 2009), and consequently, it is highly suggested to evade (Churchill, 2011; Churchill & Hedberg, 2008).

This is also in line with finding from (Sanchez & Goolsbee, 2010; Sanchez & Wiley, 2009) that found scrolling while reading content can decrease learners' comprehension as it continually moves up/down through the screen. In contrast, two studies have different findings. Leavitt and Shneiderman (2006) declared scrolling is preferred in particular reading or long text comprehension. Jin (2013) agreed that scrolling method is more effective for expository information. Nevertheless, these two studies were conducted on a larger screen, it is still not clear for relevancy on the small screen.

Besides of constraining or disposing of scrolling, restricting the quantity of taps and/or swipes is likewise suggested for maximizing learning comprehensions (Seraj & Wong, 2014). Conversely, Leavitt and Shneiderman (2006) declared that in instances of long content and perusing for understanding, scrolling was more emphasized to become pages. Jin (2013) took after that scrolling is preferable for explanatory content. Nevertheless, these studies were conducted utilizing bigger screens, and it is not vague regardless of whether this applicable to little screens.

2. Zooming.

Zooming appeared to increase cognitive load, particularly if the zoom is learner-controlled (Maniar, Bennett, Hand, & Allan, 2008). In their study that analyzed three zoom functionalities on small screens to decide cognitive load, Luong and McLaughlin (2009) observed that zooming made higher cognitive load than no zooming. Also, they found that controlled zoom made less cognitive load than the learner-controlled zoom. Contrariwise, Churchill (2011) suggests augmenting zoom usefulness to content in small smartphone properties.

3. Screen orientation.

Some studies advocated utilizing landscape view when developing and designing a mobile application for small screen devices, therefore turning the gadget so it is more extensive than high. Most smartphones permit users to control the perspective of landscape and portrait mode. A study by Sanchez and Branaghan (2011) which looked to assess learner review and complex comprehension when learning from smartphones, outcomes showed that turning the gadget to landscape mode abolished declined performance when contrasted with portrait representation of the same gadget. Designing for landscape orientation was found to improve overall learning and user experience (Churchill, 2011; Churchill & Hedberg, 2008; Sanchez & Branaghan, 2011; Sanchez & Goolsbee, 2010).

This appeared to be particularly valid for lower working memory quantity learners. The advantages of user-controlled adaptive configuration were additionally suggested with the goal that learners could alter for their individual amenity (Sanchez & Branaghan, 2011). On the contrary, it was likewise noticed that multi-touch control of content (including landscape orientation) still place a request on the client to continually control screen (Sanchez & Goolsbee, 2010) making a superfluous load.

4. **Text Size Formatting Recommendations.**

In a study that compared about text size on small and bigger screen presentations, Sanchez and Goolsbee (2010) discovered that smaller textual style delivered preferable entire retention over bigger text style on small screens, to a limited extent since it constrained the need to scroll. Despite the fact that prior studies recommend that between character and between line escalates recall (Chen & Chien, 2005), including these buffers, would raise the length of content on a smart screen device and obligate either more scrolling or pages to explore through. Chen and Chien (2005) prescribes designing mobile learning application with a solitary font style, with a variation of shades, sizes and styles are permissible.

5. **Text length and segmentation.**

Given the requirements of smartphone views, the suggestion to confine the measure of composed content substance is more than once emphasized (Churchill & Hedberg, 2008; Seraj & Wong, 2014; Wang & Shen, 2012). Whenever possible, content ought to be supplanted with pictures, sound, and narration (Bradley, Haynes, Cook, Boyle, & Smith, 2009; Churchill & Hedberg, 2008; Sung & Mayer, 2013). When that is not possible, designing for short, task-centered interactions is appropriate (Churchill, 2011). Furthermore, arranging content by giving meta-information (Churchill & Hedberg, 2008), make a portion for content into smaller which is controllable (Seraj & Wong, 2014), and/or give highlighted point synopses (Wang & Shen, 2012) will help to reduce cognitive overload.

2.10.3 Design Principles for minimizing cognitive load in mobile learning application

There are some recommendations to be taken into account to diminish the cognitive overload's effect (Clark, Nguyen, & Sweller, 2011; Mayer, 2009; Sung & Mayer, 2013). The most important consideration is to abolish irrelevant information**.** In mobile learning application, this means to exclude extraneous contents, such as music background (Brunken, Plass, & Leutner, 2003) and haphazard screen adornment and animation (Fiorella & Mayer, 2014). This is in accordance with the principles by (Mayer, Bove, Bryman, Mars, & Tapangco, 1996; Mayer & Chandler, 2001), specifically that all auditory and visual material is germane to the point of learning. Wherever conceivable, abstain from rehashing information by adding sound to written content. This will inhibit the repetition impact (Mayer & Moreno, 2003; Sweller, 2011).

Regarding split attention effect, incorporating content at whatever point conceivable is recommended (Ayres & Sweller, 2014; Mayer & Moreno, 2002). By integrating two data contents into one, this will reduce split attention (Cierniak, Scheiter, & Gerjets, 2009; Florax & Ploetzner, 2010). Mayer and Moreno (2003) further include that where conceivable, adjusting words to pictures will avert split attention. As far as the mobile device is concerned, instead of using additional equipment as a real object out of mobile learning, it is highly recommended to just fully focus on a mobile device only to prevent over-whelming to the working memory. Otherwise, the split attention will be created as the students move from the mobile device to other additional objects and back again (Liu, Geurtz, et al., 2013).

Another technique to decrease split attention is called cueing or signaling. Cueing is created to inform or direct learner to necessary information as well as to make connections. As found by (Liu, Geurtz, et al., 2013; Liu, Lin, Tsai, & Paas, 2012), that arrow-line cuing has considerably enhanced comprehension in mobile learning. They claimed that cuing makes easier for learners to see essential information and instruction (Liu, Geurtz, et al., 2013). Another form of cuing is segmentation, which is also recommended to reduce cognitive overload (Florax & Ploetzner, 2010).

In other words, segmenting long information into meaningful parts is able to organize information, and thus reducing cognitive load (Kurby & Zacks, 2008; Spanjers, Gog, & Merriënboer, 2012). Another recommended strategy to reduce cognitive load in the case of continuous material is using pauses in learning system (Hassanabadi, Robatjazi, & Savoji, 2011; Spanjers et al., 2012). For instance, Mayer and Chandler (2001) applied this by simply putting arrow button for every edge of slide or segment, which is clickable to move onward. At last, it is very important to provide learners control toward learning system in which it has been proved a lot of times able to reduce cognitive load (Hassanabadi et al., 2011; Scheiter, Schüler, Gerjets, Huk, & Hesse, 2014).

2.10.4 Implication of Mobile UI Design Guidelines

The UI design guidelines of mobile are elaborated from various studies. These collections are applicable to be adjusted for UI design of AVOMP. Moreover, they are also useful as justifications and references for each proposed guideline of AVOMP in terms of UI design facet. Next, the review of previous guidelines of AV studies is presented.

2.11 Previous Guidelines of AV Study

Some artefacts have been produced by AV researchers with the same purpose, which is to help AV developers developing the AV system. Therefore, this section elaborates the review of these artefacts of AV study. The review shows not only the explanation but also the critique on each as the gap of this study.

2.11.1 AV Design Guidelines Based on Constructivism Learning Theory

Lee and Rößling (2011) proposed five design guidelines based on constructivism learning theory as follows:

1. **Experimentation as a platform:** Different with a usual AV system, in this pedagogy, the students are asked initially to solve a problem with the construction of visualization before the official solution is revealed. This guideline emphasizes students' experiment before the tutor will give the answer to particular algorithm later on.
2. **Utilize Real-World model:** This guideline suggests the visualization construction is performed by using real-world model example. For instance: illustrating bubble short algorithm with American football games (see Figure

2.29) (Hundhausen & Douglas, 2002). This guideline stated that students will have a faster understanding when they are learning abstract concept with the depiction of a real-world example.

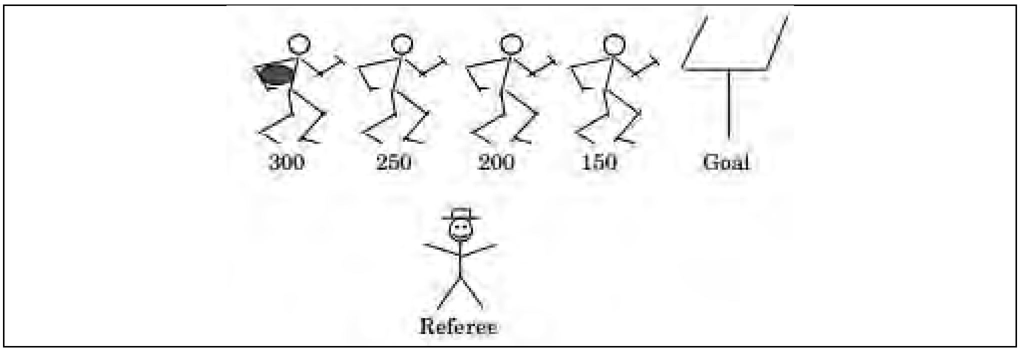


Figure 2.29. Bubble short algorithm in American Football Scenario (Lee and Rößling, 2011)

3. **Direct manipulation:** this guideline enables students to make direct manipulation towards visualization which will generate their own experience feedback for each change they have made.
4. **The inclusion of sound in AV:** the involvement of voice is purposely settled to strengthen students' understanding of an algorithm. The sound narration is also believed able to remove the ambiguity when the textual explanation is not enough for students.
5. **Open access to friends:** The student enables to share their own visualization to be commented by other students. The motivation as an active student is enhanced in this stage, and also will be beneficial for constructing their knowledge.

These design guidelines examine AV system with the inclusion of constructivism theory. The experimentation is the central role of their success for learning AV system. Nevertheless, these guidelines have not yet investigated the possibility for mobile AV system. In addition, the guidelines of UI design are also not deliberated yet.

2.11.2 AV Design Guideline Based on Aural Instruction

As far as the learning of DSA concern, Alhosban and Hamad (2011) suggested a guideline to implement aural instruction (audio instruction) with algorithm animation. He claimed that the audio instruction as additional features is able to enhance the better of students' response time and outcome rather than only to use textual instruction alone. The reason relies on the theory of cognitive load theory (Sweller, 2011) that provided dual channel concept (visualization is explained with voice narration) for better gaining understanding towards algorithm animation that is being presented. The screen shows the DSL system that he created can be seen in Figure 2.30.

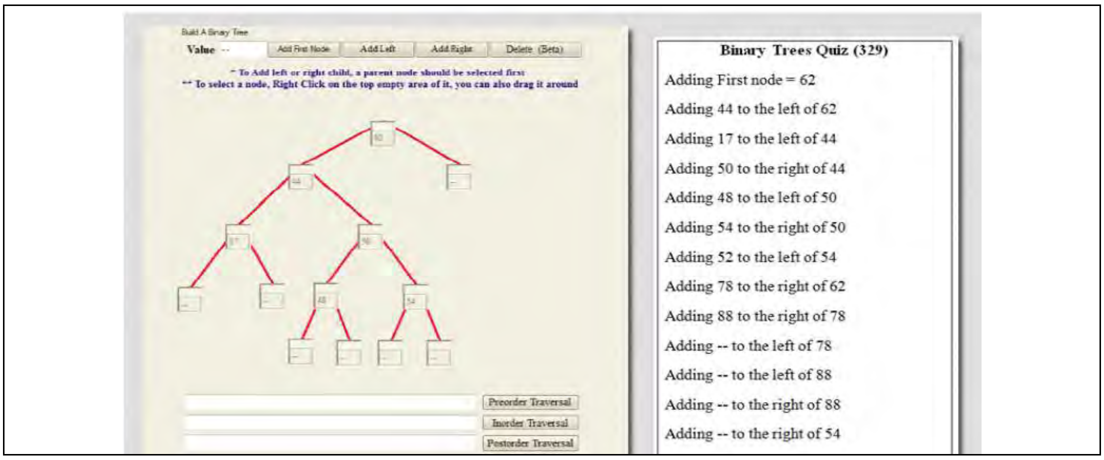


Figure 2.30. DSL of Binary Tree Algorithm (Alhosban and Hamad, 2011).

In particular, they highly suggested that aural instructions need to be associated with algorithm visualization concurrently rather than consecutively as to follow dual channel concepts. This way is able to decrease cognitive load to the lowest point (Alhosban & Hamad, 2011). In terms of interactivity, they suggested that AV learning system should enable students to move or manipulate the objects.

This study concentrated the significant relationship between two main points of view, which are cognitive load theory and aural instruction. In terms of interactivity, they suggest including viewing and changing engagement level. The users are required to give values into a box (see Figure 2.30), and then the system will arrange (showing algorithm animation) based on what users need. This study implements viewing and changing interactivity only, but it neglects the other interactivity strategy: responding and constructing.

2.11.3 AV Design Guidelines Based on Visual LinProg Tool

With regards to the effectiveness of AV system, Lazaridis, Samaras, and Sifaleras (2013) listed some guidelines for AV design based on AV web-based educational tool, named Visual LinProg (see Figure 2.31), as follows:

1. **Pictures based on cognitive load theory**: the integration of pictures as an illustration in AV is important. This is due to the students will easier to remember the picture compare to text only. Particularly, the highlighted pseudo-code is incorporated with picture illustration.

2. **User interactivity**: this suggests as mentioned many times before that user needs to be involved in the interactive design. The simplest is to utilize navigational interactivity, such as play, next, backward, pause buttons, etc. Another way is to provide user-input for users. The users can also insert a desired input they want. For a new user, a predefined input is found more helpful than their own input data.

3. **Proper use of color**: color can be used as an influential way to present information effectively. In AV system, the color can represent the particular states of the algorithm, highlight a current activity, link multiple views, or exhibit the history of user's action. The combination of proper color can strengthen the students' understanding but otherwise can prevent them.

4. **Student's attention**: In order for the novice students to avoid confusion, any dynamic presentation should begin with only a small data set for input. Some researchers support that, at most, seven elements should be used, or even changed, in each step. This statement emanates from research in psychology, which shows that short-term memory can store at most five words or figures, six letters, seven colors, and eight digits. Additionally, a restriction should exist for the number of loops. This means that the student should be able to omit the remaining part of a loop after the comprehension of its functionality. Sometimes, it is convenient to execute only a certain number of steps and to omit the remaining steps once the particular operation has been understood by the student.

5. **Interesting events**: The presentation of one interesting event per step is also an important guideline. One single step in an AV system should not be overloaded with many events/processes. From the students' point of view, each step should contain at most one interesting event; although intermediary steps could still exist, they should not contain any other interesting event. AV systems that are based on step-based execution and thus wait for the student's action in order to present the next step usually have fewer requirements for the control of the execution's speed. This is also could be done by adjusting the execution speed. Finally, some AV systems provide the capability of saving the solution results, the entire dynamic presentation, or just a part of the presentation in cases where a complete reproduction of the dynamic presentation is not feasible. An internal, efficient saving system allows the fast retrieval of an algorithm. Nonetheless, the ability to export just part of a dynamic presentation is also a useful feature.

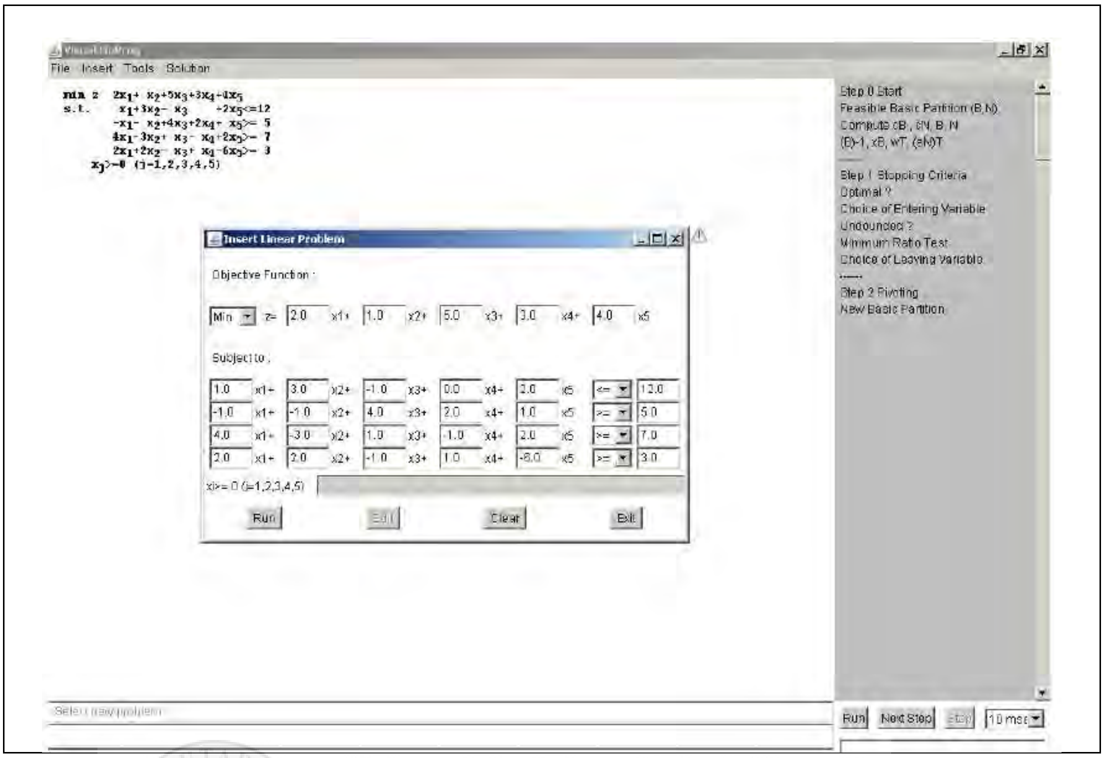


Figure 2.31. Visual LinProg Web-Based Screenshot (Lazaridis, Samaras, and Sifaleras, 2013)

The guidelines above have several key points to have the effectiveness of AV system, such as cognitive load theory, interactivity, colors appropriate, and others. However, the guidelines are designed for desktop version as a web-based system. In addition, the UI design guidelines are not provided sufficiently in the sense that how to arrange the all interface' elements to decrease cognitive load theory. In terms of interactivity, the guidelines provide several ways: viewing and changing level, but still insufficient to explain the other interactivity features.

2.11.4 Design Guidelines of Mobile Platform AV

The study by Meolic (2013) attempted to provide an AV system in mobile platform. In this study, the author provides several guidelines that are supposed to have in AV system on mobile platform:

1. The additional description or explanation, which is not really useful for extra knowledge, is supposed to be omitted.
2. The appropriate colors are supposed to choose wisely, to ensure the clarity and readability.
3. The use of symbols (icon and images) should be limited.
4. The AV system is supposed to have portrait and landscape mode.
5. Pop-up feature to make users can see further information in limited screen
6. Implementation of using swipe gesture of tabs as different screen (this can help users to see additional information in limited screen)
7. The hierarchical presentation is utilized, to optimize the use of showing group items (expand) and hidden items (collapsed)
8. Pseudocode and narrative content can be used in AV system.

Even though the author has provided some design guidelines for AV on mobile platform, but these guidelines are only limited on design guideline in general only, the details to organize the UI based on relevant human perspective theories are overlooked. In perspective of interactivity, the interactivity level is not discussed profoundly. They just focused on demonstrating the animation only (viewing level strategy) on sorting algorithm as a case study. In particular, the demonstration of algorithm animation is controlled by navigation buttons (run, stop, pause, one step, reset, etc.) to see the related pseudocode algorithm line and represented algorithm animation. The figure of his prototype can be seen from the Figure 2.32.

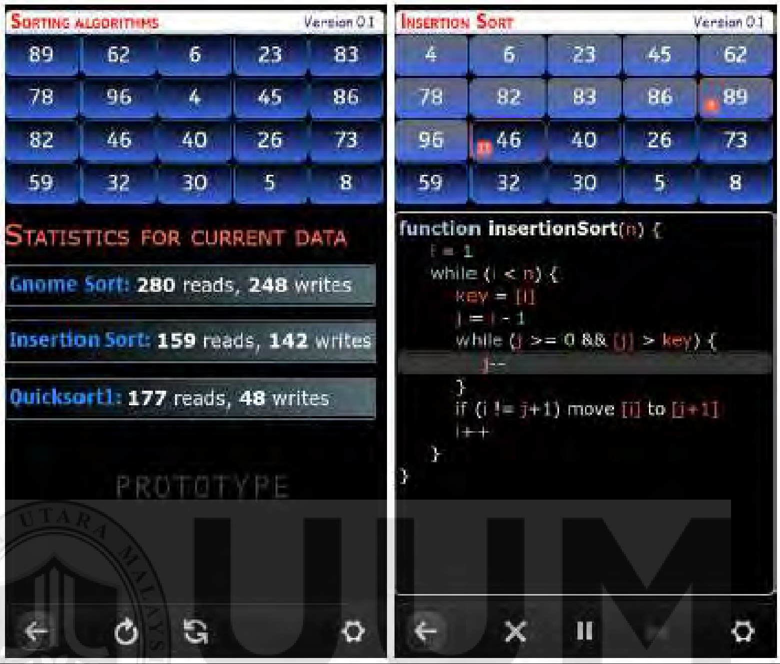


Figure 2.32. Prototype that shows shorting algorithm (Meolic, 2013).

In his prototype, the buttons have been created as a way to interact with the system, but this button will just let learners watch algorithm animation step by step only without further interactive interaction.

2.11.5 Design Guideline of AV Mobile Based on Four Modules

Another attempt by Boticki, Barisic, Martin, and Drljevic (2013) that makes the guidelines in order to develop a mobile AV based on four modules, called Sortko (see Figure 2.33). Sortko is an Android application that is prepared specifically for sorting learning algorithm. Through Sortko, students are prompted to interact with the system by switching (sorting gesture) the element of arrays. The student can press one box of arrays to be a substitute to another box by pressing it.

Figure 2.33. The Sortko mobile app, using switch elements to interact (Boticki, Barisic, Martin, and Drljevic, 2013)

The following are the four modules they made (see Figure 2.34):

1. **The interactive system module**: contains interactive algorithm sorting that verifies whether the student's answer is correct or not after switching the elements (using constructing engagement strategy).
2. **Scaffolding module**: is provided to show help or guide as a fast feedback every time a student makes a mistake. The system will show the logic behind the sorting step so that student can understand well.
3. **Motivational module tracks**: the system has parameters in terms of accuracy, speed, etc. Based on these parameters, this will calculate the results of the student itself in form of a reward point.

4. **The graphical user interface (GUI) module**: the GUI consists of three main views: opening view, sorting view, and the outcome view. All is arranged in horizontal style to maximize the space of use.

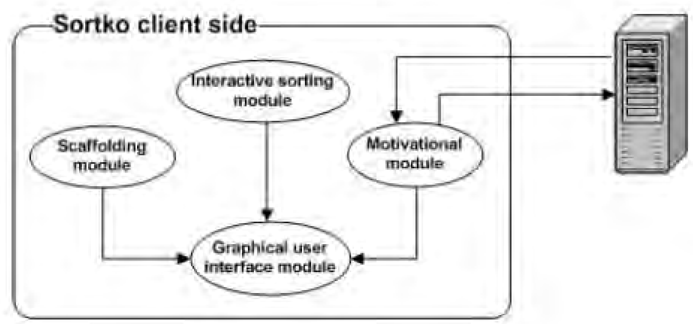


Figure 2.34. Modules in sortko system (Boticki, Barisic, Martin, and Drljevic, 2013)

The modules provide a tactic of interactivity by switching gesture and then improve it by giving feedback to users if he makes mistake. The strategy of multiple views is also provided in GUI module. However, the guidelines are not comprehensive enough about interactivity level engagement as well as the other aspects in designing UI of AV system.

2.11.6 Design Guidelines for Presenting AV

Simonák (2013) declared some guidelines to present AV system for desktop software, as follows:

1. **Screen design and multiple views**: This is arranged for beginner user in order to reduce cognitive load through abstraction. The conversion of the complex portion into simpler forms is highly suggested. For instances, some complicated operation in the algorithm can be hidden or excluded at first in which the basic

concept is displayed first, and a deeper explanation follows based on user control. Another, it can be in form of skipping step refinement approach. It is also so recommended to make subdivision in the screen as functional areas. These divisions contain related information. The different facet of an algorithm is shown in different view in which this way is conceptually easier and simple to construct than united outlooks. The information area could also be included to explain during animation.

2. **Colors**: colors are used to give attention to particular information or data, helping to classify structural elements in order to catch the alterations in time. Color improves attraction, comprehensibility, memorability, and believability. Nevertheless, the guidelines use of color is really not easy to adapt. This is due to the subjectivity of color perception; the complicated design from color can provoke disadvantages or drawbacks. It is much recommended to be more cautious when using colors.

3. **Sound**: Sound is able to decrease the visual clutter of presented user interfaces. Yet, some problems can be raised too when applying sound. For example, the problem happens when all student plays all software in the same room, meaning there is more than one audio output can make confusion. The same goes to sound; it is advocated to be more careful to apply it.

4. **Interaction**: Interactive feature has to become the most important element in visualization, allowing the student to step through the operational algorithm in control, responding, and providing input activity for an algorithm.

These design guidelines provide some ways to design the UI of AV system by considering cognitive load theory, the sound, and color. However, these were provided for the desktop software version. It is important in this study to look up more for arrangement on a mobile app that has a smaller screen. The interactions in these guidelines are more varied, which are controlled viewing, responding, and changing, but are still not comprehensive. It is important in guidelines that are more detailed on how to arrange those interactivity features.

2.11.7 Design Guidelines of AV System by Including PV

Different with all aforementioned guidelines, Osman and Elmusharaf (2014) included PV, which is in form of programming language C++, instead of pseudocode mode. Beside of that, they listed guidelines with respect to design AV system in Desktop version, as follows:

1. Designing four main areas for AV system
    a. **Memory mapping**: this area is in changed for displaying diagrammatical data structure using figures, system-stacks (to see the shift of variables value during iterative looping), and control variables (to see the current value of variables).
    b. **Code display area**: this area to show the PV code in C++ language which is currently being selected by the user. The tracing line will be automatically activated during algorithm execution.

c. **System area message**: this area is responsible to display useful guidance for the user. Such as error message, feedback message, warnings, hints, comments, tips, and etc.

d. User input area: this area embraces the interactive function to change the algorithm visualization by inserting value in the text-box provided.

2. Colors used are based on the recommendation of Ware (2012) that stated there are 12 safe colors that can display for text-information in computer monitors. They are Yellow, Green, Red, Blue, White, Black, Orange, Pink, Gray, Cyan, Purple, and Brown. The depiction of their AV system can be seen in Figure 2.35 below:

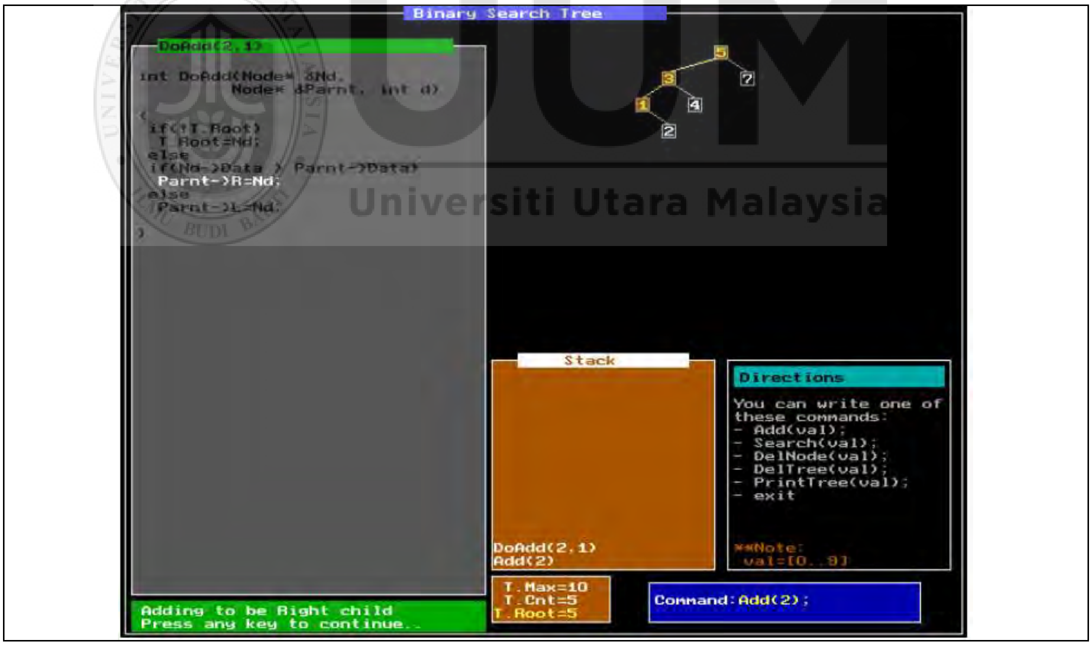


Figure 2.35. The Prototype of AV for binary search tree (Ware, 2012)

The guidelines have expounded the detachment of useful functions into four areas. The inclusion of appropriate colors is also justified. However, the guidelines are applicable only for AV system on the desktop platform. There are too many areas that will not be possibly applied to the mobile platform which has a smaller screen. The adjustment has to be tolerated for AV elements within mobile AV system. The colors selection in this study also was adapted from safe colors for monitor (desktop). The guidelines also use "viewing and changing level" only for its interactive function, and do not address the other engagement strategies.

2.11.8 AV Web-Based Design Guidelines

Another attempt by Vrachnos and Jimoyiannis (2014) that provided guidelines to design AV system on a web-based platform, as follows:

1. To design AV user interface, three main components are purposed:
   a. **Left panel (Source Code Editor):** this area presents an algorithm in pseudocode form. This area also shows highlighted line-code during execution which corresponds to its represented visualization in the right panel (Animation area).
   b. Right panel (Animation Area): the animation of a current executed algorithm is displayed in this area. This area also highlights the critical algorithmic value by showing the smooth transition of the red arrow.
   c. Down Centre Panel (Down Centre Panel): this panel contains four interactive buttons, namely restart, pause, start and step-by-step execution. Boxes input as another interactive strategy is also displayed for the user to modify input data of arrays so that an executed algorithm can be tested with various datasets. A slider is included to control the speed of the animation.

d. **Use Highlighted Line:** it is important to highlight the important stage from pseudocode**.**

e. **Use dynamic arrow:** it is also recommended to use the dynamic arrow to show the movement of values' swapping and changing.

f. **Engagement Strategy:** The experimentation is a key point suggested for learning. In particular, learning the behavior of animation based on the modifying input data and algorithm's code from the user. When giving the processing the input the AV also could detect the error required input, such as index value out of the array bound. The prototype, which is named as DAVE, can be seen in Figure 2.36 below:

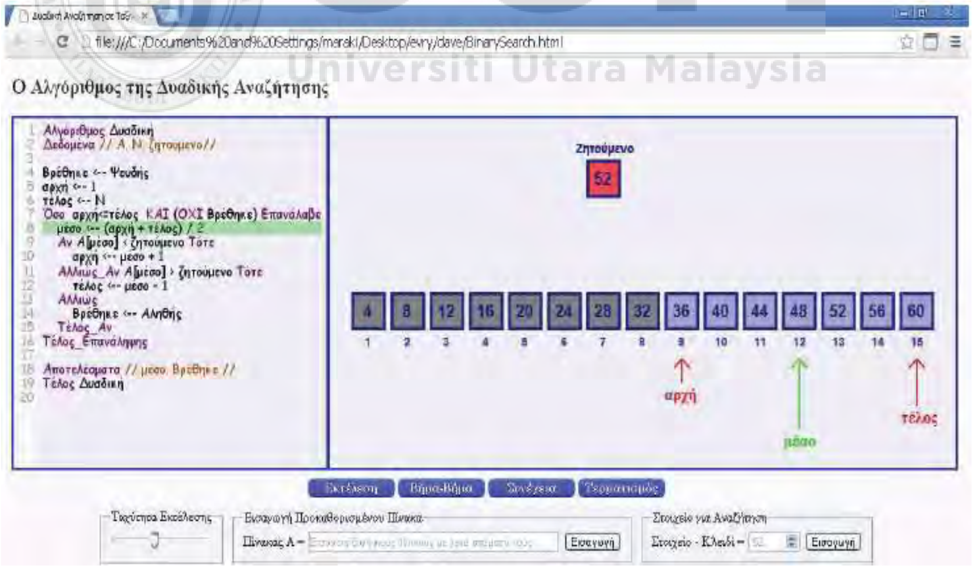


Figure 2.36. The web-based AV system, called DAVE (Vrachnos and Jimoyiannis, 2014)

Unlike the guidelines before, these guidelines have fewer boxes area on the computer screen so that user will be focusing on two areas only in learning behavior of algorithm visualization and its corresponded line of codes. The user beforehand inputs some values on down panel, changing the value of the variable, to get the other visualization effects. These guidelines are also applied for computer only; however, it is worth example to have fewer areas as to be applied for small screen mobile. In terms of interactivity these guidelines provide viewing and changing level but are not adequately offer the other interactive features.

2.11.9 Design Guidelines of Extending AV through Case-Base Comparison

Jonathan, Karnalim, and Ayub (2016) listed several guidelines for extending AV effectiveness based on their heuristic evaluation, as follows:

1. Solving Visualization: This is the common feature of AV which is to show step-by-step visualization's mechanism (viewing activity), showing the logic behind a particular algorithm towards certain set of values.

2. Performance Comparison: This is to compare the algorithm performance based on completeness, optimality, complexity, output, and time-execution. This enables students to learn the characteristic for each algorithm.

3. Input generator: This is to produce the set of values randomly in changing activity level, so that students do not need to prepare their own input.

4. Dynamic changing: Another way is to also provide students with input by hand (text box) in order to see the different behavior of algorithm based on their input.

5. Constructing: is to provide visualization setting (changing the label style of nodes), such as number, alphabet, and roman number.

6. Presenting: is to present to other students about how the behavior of an algorithm during visualization.

The guidelines presented above have defined engagement taxonomy quite comprehensive but is still lacking in terms of UI design. Additionally, the responding activity has yet to be discussed and also, they have miss-interpreted the meaning of constructing activity. They defined constructing as customization of visualization setting, in fact the real meaning of constructing here is to build the visualization itself in order to gain the knowledge of how the algorithm working (Myller, Bednarik, Sutinen, & Ben-Ari, 2009).

2.11.10 Implication Review and Critique on Previous Design Guidelines of AV

The reviews and critiques above clarify that all the reviewed design guidelines have come out with certain advices to ensure the effectiveness of AV system. However, it was found lacking from the existing study that addresses AV on mobile platform, mostly for desktop or website platform. In addition, even though, some guidelines have offered to put more engagement features (viewing, responding, and changing), but still lack of detail explanations on how to arrange or apply them on AV system.

The previous mobile AV guidelines are even more insufficient; due to the engagement strategy used is only viewing or controlled viewing only.

Furthermore, most of guidelines also do not touch the human perspective theory profoundly. Hence, it ought to be noted that this is the research gap that should be the focal point of this study. However, these previous guidelines are served as relevant sources in this study, in conjunction with identifying the appropriate implemented theories or advices to be adjusted on AVOMP. Moreover, unlike desktop platform, mobile screen has smaller size and touch screen interactivity so that to investigate the recommendations from relevant empirical studies and theories are then taken into accounts. Specifically, to identify the kind of recommendations that can be applied in order to arrange important AV design elements (pseudocode, animation, interactive buttons, menu and so on) on mobile screen. Next section presents the definition of guidelines and how they are presented and justified.

2.12 Guidelines

As the main artifact of this study are in form of guidelines. It is imperative to know the definition of guidelines itself. According to O'Malley et al. (2005), guidelines are principles or rules for movement, covering some composites of practitioner, which determine best practice in a particular discipline and the perspective of research based into becoming facets relevant in that discipline.

Researchers (Gong & Tarasewich, 2004; Seong, 2006; Shitkova et al., 2015) agreed that to uncover the guidelines of a particular domain, the identification towards relevant theories, scientific and empirical studies have to be conducted, such as previous guidelines, related studies, framework, principles, etc. Guidelines provide practical suggestion in regards to overcome the various issues that possibly emerge in particular domain (Nilsson, 2008).

The following principles to make guidelines are introduced by O'Malley et al. (2005):

1. **Guidelines are based on theory-informed "do and don'ts"**

   It has to be profoundly investigated that guidelines are according to relevant theories or practice towards the particular domain. Leavitt and Shneiderman (2006) added that guidelines should be presented complete with examples and justification. The example here means graphic illustration so that the users are able to understand it more clearly.

   Accordingly, this study should derive each guideline from relevant theories, some relevant practices of effective learning from existing guidelines of AV system (from the desktop, website, and mobile platform), guidelines or existing published research of designing of mobile learning, and etc.

2. **Guidelines should be validated**

   As each guideline is grounded in either relevant empirical studies or theories, as a result, the guidelines provide some references as justification in a database.

3. **Guidelines will be separated into audiences**

   This principle is about the division of guidelines itself. O'Malley et al. (2005) explained that his guidelines are divided into several audiences in which each guideline belong to some particular audiences, while others have different audiences. They are ranging from students, teacher, professional practitioner, usability engineers, system designers, etc.

From all these considerations, O'Malley et al. (2005) then made a template for guideline's form (see Table 2.9), as follows:

Table 2.9

Guidelines Template

Recommendation	What the guideline says
Audiences	Who the proposed audience is
Basis	Guidelines are derived from
Notes	Considerations and recommendation that need to keep in minds about this guideline
Justification/ Elaboration	Validation or justification of the guideline, and elaboration of contexts in which it could be used

O'Malley et al. (2005)

Another template of the guideline by GSMA (2012), in Table 2.10:

Table 2.10

Guidelines' Template

Guidelines	Implementation	Use Case and Example
What the Guideline Says	• How to do the guidelines • What audiences should prepare or notice • The warning for audiences • What audience has to follow or be careful about	The practical example to make the detail of a guideline.

GSMA (2012)

There are various ways to present guidelines. The other studies for instances (Gong & Tarasewich, 2004; Hong & Kim, 2011; Seong, 2006; Shitkova et al., 2015) do not have a specific structure in table form as aforementioned guidelines, instead, the text-based explanation only that comprises the sequence of explanations. Meaning, each guideline is elaborated with sub-guidelines and explanations completed with justification from empirical studies and theories.

Another example of guideline text-based format from Leavitt and Shneiderman (2006):

Guidelines: what guideline says about?

Comments: the detail explanation of the guideline: recommendation of what users should do and don't and some explanation based on scientific evidence from related studies. This section can also include some results from the experimental research.

Sources: the collection (references) of published researchers that are grounded the guideline.

Example: providing graphical illustration (pictures) with additional descriptions.

2.12.1 Design Guidelines

As this study is to propose the design guidelines, it is important to discuss the definition of design guidelines itself as well as how they are carried out in the study. Design guidelines are sets of recommendations towards good practice in design. They are prepared to provide clear instructions to developers and designers on how to adopt particular principles, such as learnability, intuitiveness, consistency, and efficiency. Design guidelines also provides helpful guidance in terms of achieving a design principle that can be applied for a specific-platform or cross-platform.

For instance, "text should be easy to read" is a design guideline statement that covers between a principle design (such as, "an interface should be easy to utilize") and implementation rule (such as, "font-color: black; background: white; font-size: 14px"). Therefore, a design guideline should be able to guide designers on how to implement a principle in flexible way, without limiting their creativity.

Design guidelines come from various sources. Some have the foundations based on basic common sense, however they are often violated. Others are grounded in human cognition understanding or are from some collections of studies or empirical studies. Others also can be derived from a bunch of theories related to human action. Nevertheless, some guidelines may conflict to each other due to the viewpoints and varied sources or when they are applied under different situation in design (e.g., design for older users vs younger users). Thus, design guidelines are not as generalizable as design principles.

The divergence and deficiency of design guidelines originate from the absence of one binding together interaction theory- such a theory would most likely require a total comprehension of human cognition; a product which designer may keep on struggling on at within a reasonable time-frame. Subsequently, a designer's circumspection must keep on driving the understanding of such guidelines in order to improve the client experience of a product. Meanwhile as stated before in Chapter 1, this study focuses the design guidelines on two perspectives, which are UI design and interactivity. The UI design is about how the UI elements are arranged, placed, designed, colored, etc., whereby interactivity is about how the DSA material should be brought up in terms of its interaction towards users.

2.12.2 Implication of Guidelines

The definitions of guidelines, design guidelines and how to present them are elaborated. The implication of guidelines to this study is important to know how guidelines are derived from as well as the way they are presented. There is no fixed rule or template for constructing guidelines. There are various ways to present guidelines, either in table division form or text-based structure. However, it has to be noted that guidelines are strengthened with the justifications (references: derived from collection of studies and theories) and are illustrated through implementations or examples.

2.13 Conclusion

A number of theories and concepts involving DSA, software visualization, AV, PV, interaction and UI design aspects as well as previous guidelines have been presented and reviewed. The literature recommends including interactivity and UI design in order to create an effective AV system. In addition, this study attempts to integrate AV system on mobile platform. Consequently, to identify the appropriate recommendations from the relevant theories, concepts, and empirical studies that match in this study are important for constructing the purposed design guidelines of AVOMP. Figure 2.37 showed the overall literature overview that has been presented in this chapter.

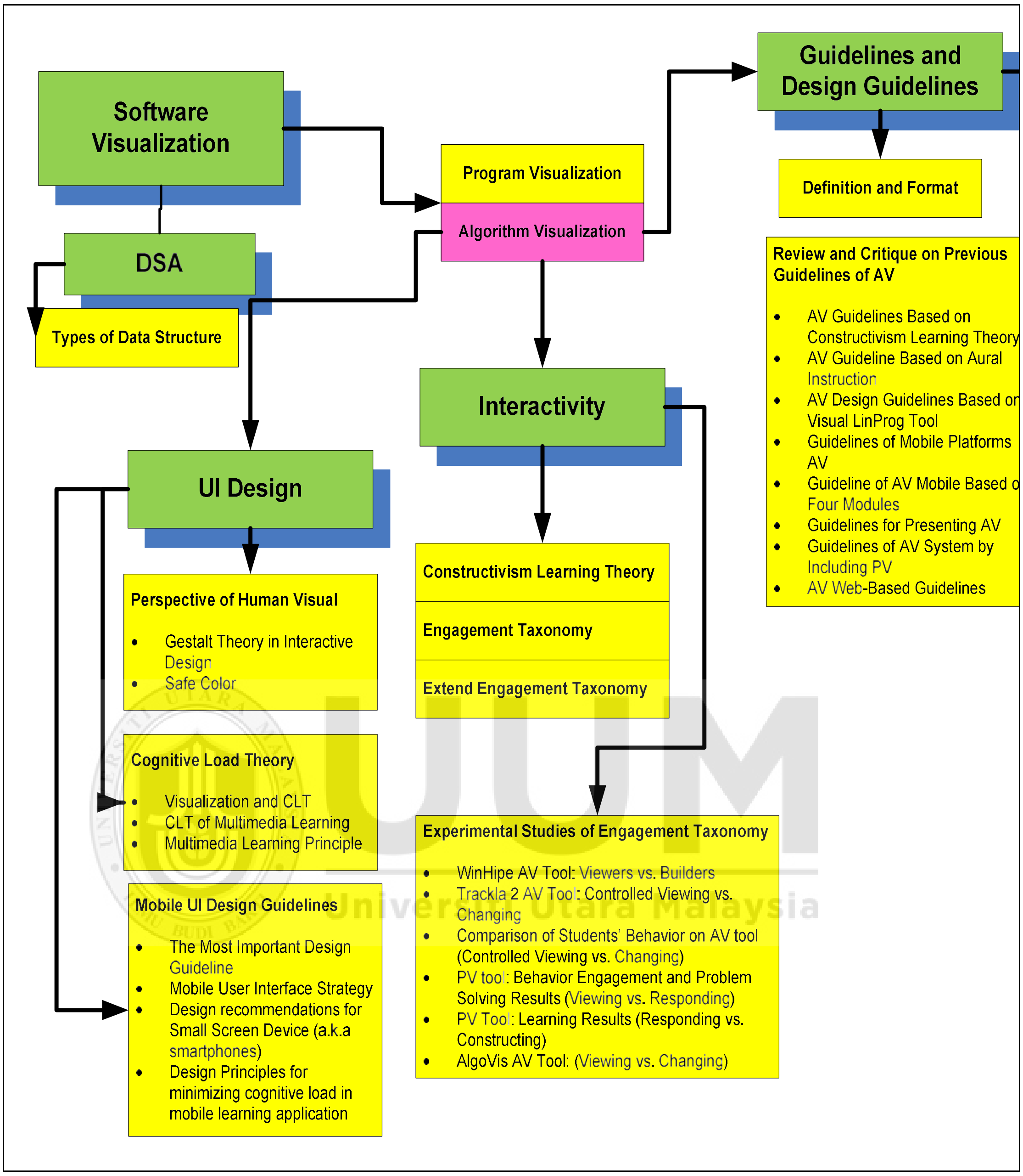


Figure 2.37. The overall overview of literature in chapter 2.

CHAPTER THREE
RESEARCH METHODOLOGY

3.1 Introduction

This chapter discusses the systematic processes in order to achieve the objectives of this study. By adapting design science research information system, the processes are used as the methodological approach. The details of each phase are elaborated in this chapter.

3.2 Design Science Research

Design activities are crucial to most applied disciplines. Research in design has a long history in numerous fields including engineering, architecture, psychology, education, and the fine arts (Cross, 2001). The design science research paradigm is highly relevant to information systems (IS) research because it directly addresses two of the key issues of the discipline: the central, albeit controversial, role of the IT artifact in IS research (Orlikowski & Iacono, 2001).

Design science as conceptualized supports a pragmatic research paradigm that calls for the creation of innovative artifacts to solve real-world problems (Hevner, 2007). Thus, design science research combines a focus on the IT artifact with a high priority on relevance in the application domain. In addition, Norshuhada and Shahizan (2010) state that the valid artifacts in design science research could be in form of algorithm, working prototype, user interfaces, processes, techniques, methodologies, and frameworks. As mention in chapter one, the result of this study is a form of artifact, which is the design guidelines of AVOMP. Therefore, design science research as the selected approach is appropriate in this research.

Design science research (DSR) is one of the research paradigms that creates and evaluates IT artifacts proposed to solve identified organizational problems. Such artifacts are represented in a structured form that may vary from software, formal logic, and rigorous mathematics to informal natural language descriptions. There are roots in engineering and the sciences of the artificial of the design-science paradigm (Simon, 1996). It is fundamentally a problem-solving paradigm. The idea is to create innovations through the analysis, design, implementation, management, and use of information systems that describe the ideas, practices, technical capabilities, and products that able to be effectively and efficiently accomplished (Bansemir, 2013).

Norsuhada and Shahizan (2010) stress that design research is based on Iterative Triangulation Methodology (ITM), where theoretical, development and empirical aspects of research are triangulated to achieve the design research objectives. Many researchers have suggested an appropriate process in design science research. Peffers, Tuunanen, Rothenberger, and Chatterjee (2007) reviewed and evaluated the process for conducting design science research in information systems. Hevner, March, Park, and Ram (2004) also suggested seven steps guidelines for design science research processes.

3.3 Research Methodology Phases

This study conducts three phases as proposed by Offermann, Levina, Schönherr, and Bub (2009): (1) problem identification, (2) solution design and (3) evaluation. Figure 3.1 depicts the activities, which are involved in each phase.

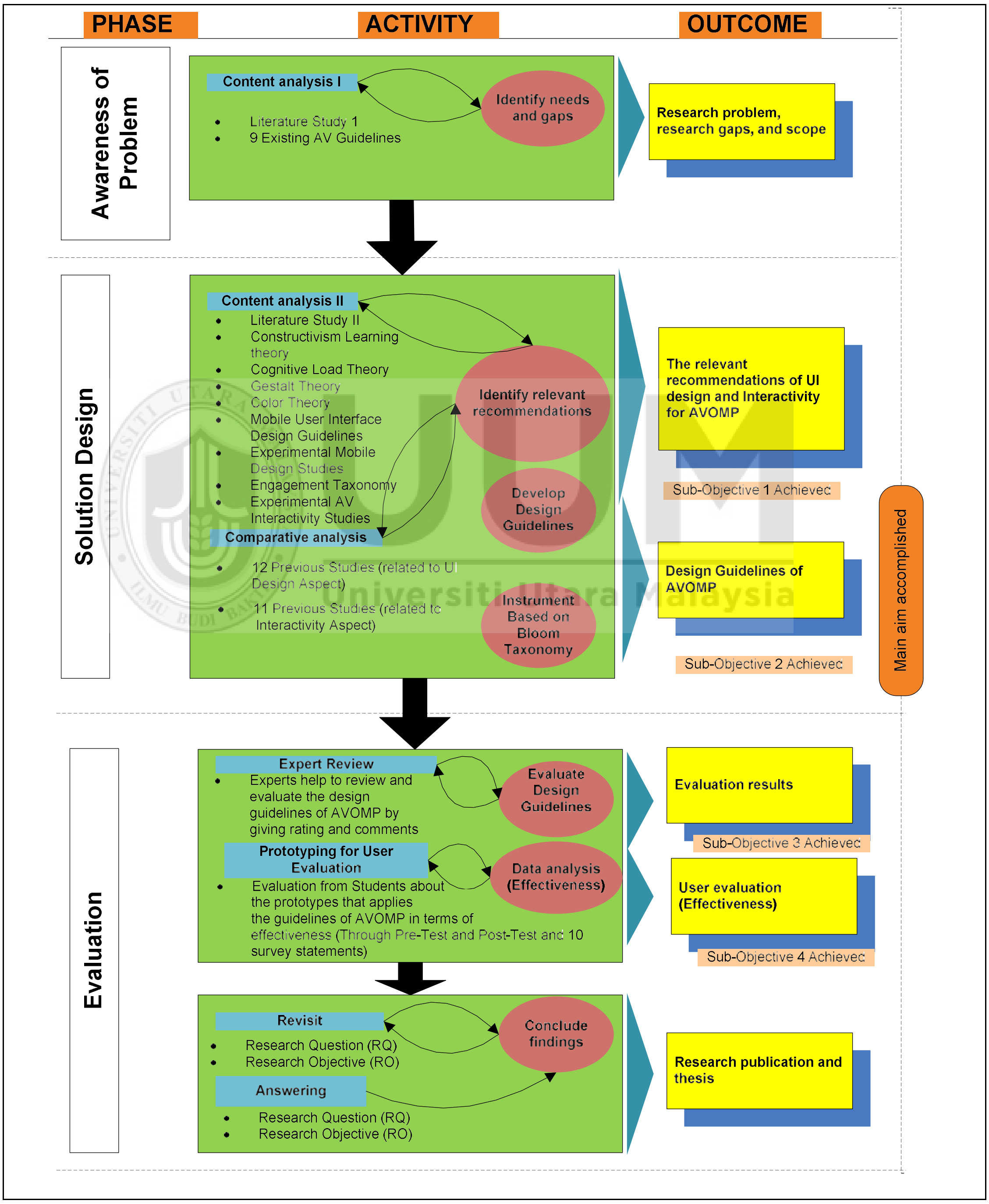


Figure 3.1. Phases of research activities

3.3.1 Phase 1: Awareness of Problem

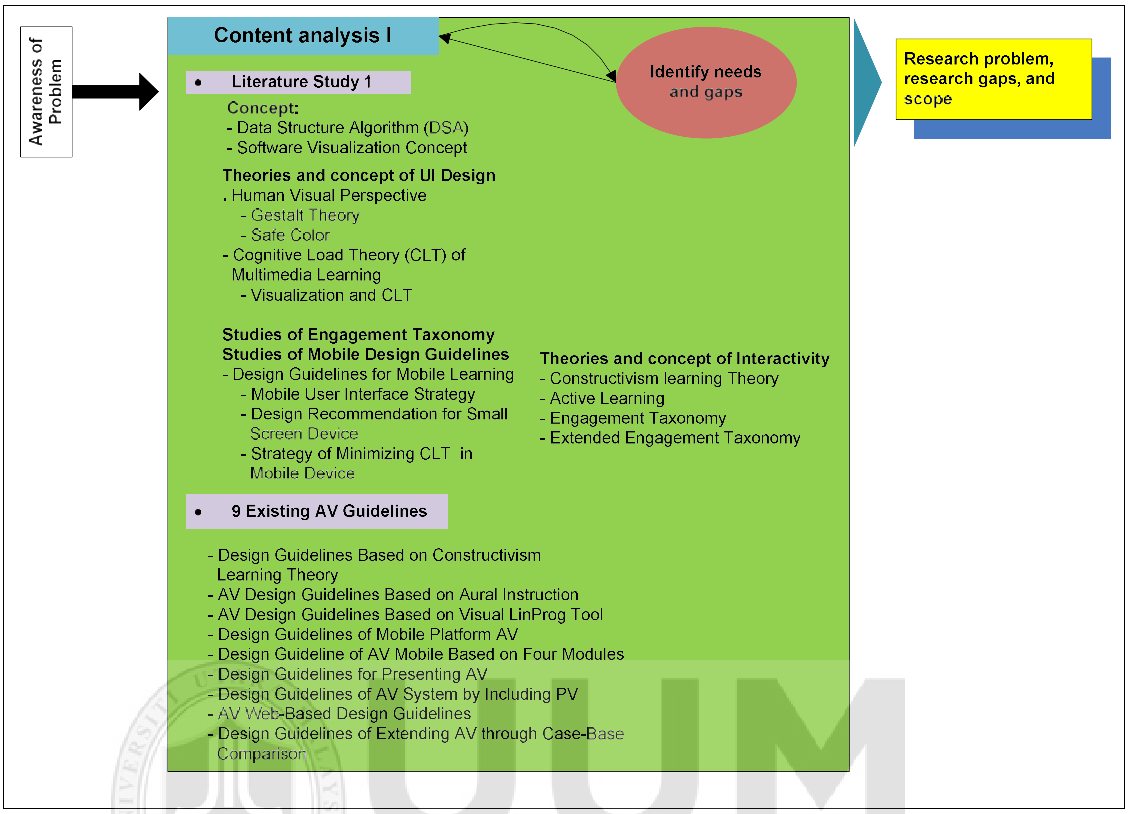


Figure 3.2. Problem identification

The first and foremost phase of this study is to identify problems and gaps. Thus, the content analysis is conducted through related concepts in the literature study. The issues are defined from this process before coming up with the establishment of research gaps. Then, it continues to formulate the research objectives and scopes as elaborated in Chapter 1.

3.3.1.1 Content Analysis I

In this phase, the first content analysis is performed through elicitation of literature. A systematic and comprehensive literature research are conducted from various sources, which are books, articles in journals, proceeding of the national and international conference. At this stage, all related theories and concepts, and 9 existing design guidelines are identified (see Figure 3.2). At this point, the analysis towards problems is identified by reviewing all those previous design guidelines along with the related theories and concepts of this study that have been discussed in Chapter 2.

3.3.2 Phase 2: Solution Design

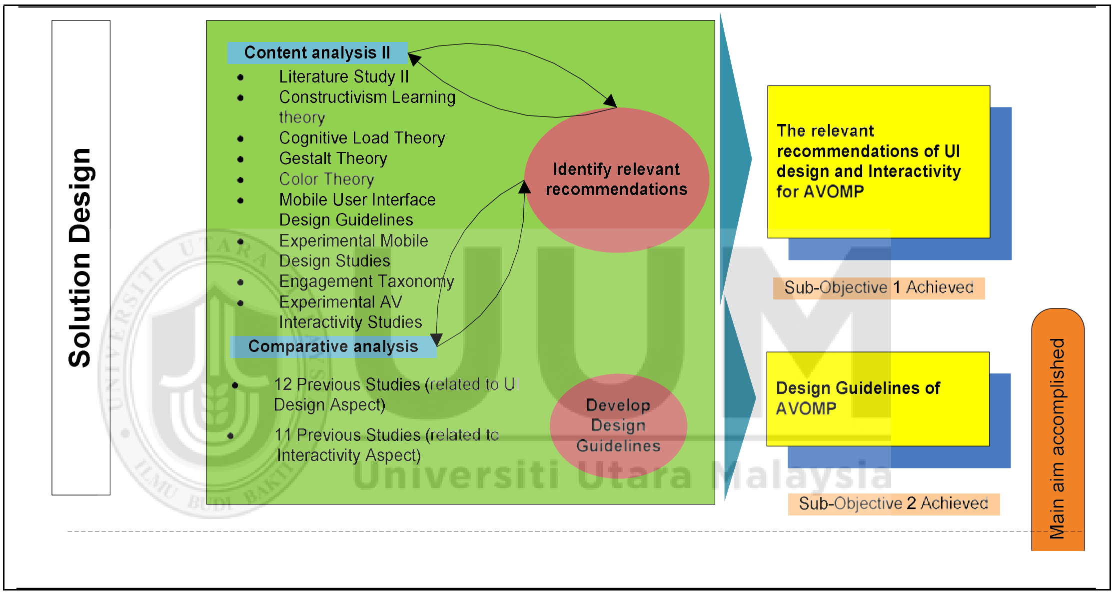


Figure 3.3. Solution design activities

After research problem, gap and scope are identified, the next phase is a solution design. In this phase, in order to accomplish the first sub-objective, content analysis II and comparative analysis are carried out. The comparative analysis in this phase has 2 parts. The first part consists of 12 existing studies in the perspective of UI design are compared and tabulated. The second part consists of 11 existing studies that focus on interactivity aspect. These existing studies are from the studies of existing AV design guidelines, mobile usability studies, and usability principles.

3.3.2.1 Content Analysis II

Content analysis is a way of processing to extract knowledge of the proposed study. According to Syamsul Bahrin (2011), the sources can be from several forms, such as text, audio, video, and others. The main objective of the content analysis II is to collect the concepts and theories before developing the design guidelines of AVOMP. The comprehensive review on existing design guidelines, constructivism learning theory, cognitive load theory, gestalt theory, color theory, mobile user interface design guidelines, experimental research related to mobile UI design and AV interactivity are also involved, in order to identify the relevant recommendations from two main viewpoints: UI design and interactivity for the design guidelines of AVOMP. After that, the next step is called comparative analysis.

3.3.2.2 Comparative Analysis

In identifying the relevant recommendations for the proposed design guidelines of AVOMP in this study, an exploration, comparison, and analysis towards existing studies are necessary (Ariffin, 2009; Siti Mahfuzah, 2011; Syamsul Bahrin, 2011). This study also analyzes and compares 9 guidelines as explained in chapter 2. They are:

1. AV Design Guidelines Based on Constructivism Learning Theory (Lee and Rößling, 2011)
2. AV Design Guideline Based on Aural Instruction (Alhosban and Hamad, 2011)
3. AV Design Guidelines Based on Visual LinProg Tool (Lazaridis, Samaras, and Sifaleras, 2013)
4. Design Guidelines of Mobile Platforms AV (Meolic, 2013)

5. Design Guideline of AV Mobile Based on Four Modules (Boticki, Barisic, Martin, and Drljevic, 2013)
6. Design Guidelines for Presenting AV (Simonák, 2013)
7. Design Guidelines of AV System by Including PV (Osman and Elmusharaf, 2014)
8. AV Web-Based Design Guidelines (Vrachnos and Jimoyiannis, 2014)
9. Design Guidelines of Extending AV through Case-Base Comparison (Jonathan et al., 2016)

These guidelines are systematically compared with other bunch of empirical studies with have conjunction with the UI design and Interactivity aspects. Therefore, as mentioned before, the UI design aspect with 12 studies and Interactivity aspect with 11 studies are compared with the inclusion of the design guidelines mentioned above. The UI aspect specifically includes studies, which are from the study of mobile design guidelines, previous AV design guidelines, UI design guidelines and Usability principle of designing a mobile app (refer to Chapter 4, section 4.3.1 for detail). As for the interactivity aspects, 11 studies are compared, which are from the study of previous AV design guidelines, previous AV studies, and mobile design guidelines (refer to Chapter 4, section 4.3.2 for detail).

As a result, the extracted findings can be produced for the purpose of this study. Next, those results (recommendations) are integrated and updated further to be complete guidelines. In other words, each guideline is established with 4 elements, which are recommendation, basis/sources, explanation, implementation, and illustrations. Consequently, as mentioned before, the inclusion of the existing empirical studies related to UI design and interactivity aspects into each guideline is imperative in this study in order to strengthen and justify its significance (see Chapter 4 for detail).

3.3.3 Phase 3: Evaluation

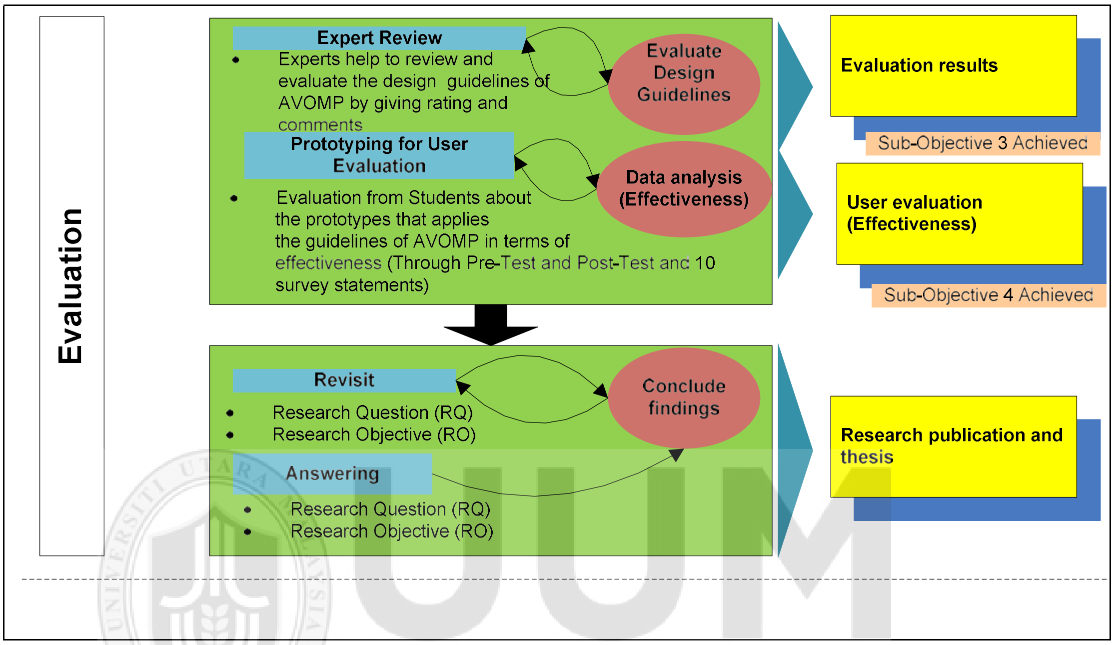


Figure 3.4. Evaluation activities

This phase is in charge to evaluate and validate the research outcomes. In DSR, Norshuhada and Shahizan (2010) recommended several approaches to validate artefacts, namely analysis, experience, example, evaluation, and persuasion. Thus, the validation techniques implemented in this study are by expert review and experimental study in prototype form. The AVOMP design guidelines will be validated through expert evaluation, whereby the prototype will be evaluated in terms of "effectiveness".

3.3.3.1 Expert Review

In order to evaluate the proposed design guidelines expert review are involved in reviewing the proposed design guidelines of AVOMP. This review includes academicians, researches, lecturers as a content expert who are experience in their fields. Therefore, the expert reviewer instrument is created in order to evaluate the design guidelines.

The experts are also allowed to provide helpful comments that will improve the proposed design guidelines. Shneiderman, Plaisant, Cohen, and Jacobs (2010) suggest that having three to five experts participating in reviewing process is adequate. Since this study is based on the research-based guidelines, this study adopts the evaluation method created by Leavitt and Shneiderman (2006). Meanwhile, this study managed to get 16 experts to review the proposed design guidelines.

3.3.3.1.1 Expert Reviewer Instrument

The format of the reviewer instrument is adapted from Nuruladwan Aziz (2015). It contains the following objectives:

1. Evaluate the relevance of the proposed recommendations/guidelines
2. Identify and combine duplicate guidelines
3. Identify and resolve guidelines that conflicted with each other
4. Reword unclear guidelines.

The instrument also contains the important and easy to understand scale for each guideline. They are also required to fill demographic profile, such as level of education and working experiences (in years). They also are encouraged to provide additional comments or advice in the instrument.

In detail, for the first question, the expert is required to verify the relevancy of each guideline with three options (i.e. some are definitely not relevant, or some may be not relevant, or all are relevant). For the second question, the expert is asked to verify the importance scale for each guideline with the anchors set at 'Important (1)' to 'Very Important (5). The third question is the Likert scale of easy to understand with the anchors set at 'strongly disagree (1)' to 'strongly agree (5). For questions four to nine, the experts were required to evaluate the design guidelines by choosing an option "yes or no". Finally, based on their experience, expertise, and point of view, they were expected to write additional comments or advice for overall design guidelines as a whole. The expert reviewer instrument can be seen in Appendix E.

3.3.3.2 User Evaluation through Prototyping

The next evaluation of design guidelines will be continued by implementing them into a working prototype. This kind of way has been applied by several researchers for their evaluation phase, such as Siti Mahfuzah (2011) and Syamsul Bahrin (2011). Offermann et al. (2009) explain that experimental study is a way to evaluate research that can be performed either in laboratory or field experiments. In this study, the experiment is performed in the laboratory.

Thus, as mentioned earlier the user evaluation of effectiveness is selected in this study. The studies (Urquiza-Fuentes & Velázquez-Iturbide, 2007, 2013), (Laakso et al., 2009) and, (Ali & Derus, 2013) agreed in order to measure the effectiveness of software visualization system; it has to be measured from the students' knowledge acquisition (KA). The KA is about the measurement of learning result through DSA test. As mentioned in Chapter 1, this study chooses sorting algorithm (sub-topic in DSA subject) as case study. In detail, the students will be tested through the learning of 3 sorting algorithms, namely Bubble sorting, Inserting Sorting, and Selection Sorting.

3.3.3.2.1 Sampling and Sample Size

As the focus of this study is on particular participants, the purposive sampling is used because of the homogeneous subjects from particular characteristics of a population (Tongco, 2007). The purposing sample focuses on the subjects who have a particular characteristic concerning the objective of the study. Therefore, students of computer science, who take DSA subject are the one who participated in this study.

In addition, Roscoe (1975) suggested the following rules of thumb for determining sample size:

i. Sample sizes larger than 30 and less than 500 are appropriate for most research.

ii. For experimental research with tight experimental controls, successful research is possible with samples as small as 10 to 20 in size.

Therefore, based on the considerations above, this study is aimed for 30 students who take DSA subject.

3.3.3.2.2 Effectiveness measurement

Based on discussion earlier in chapter one and review in section 2.5, this study emphasizes that AVOMP (with more variety of engagement taxonomy and follows UI design recommendations) will have more effectiveness in learning DSA subject. This study employs Pre-Experimental Design with one group Pre-test Post-test study. Pre-experimental design is so named because it follows the basic steps experimental but failed to include the control group. In other words, single groups often researched, but no comparison with non-treatment groups was made (Campbell & Stanley, 2015). In other words, this study only includes one group of participants without involving control group (as a scope of study mentioned in Chapter 1).

The one group Pre-test Post-test study is one of Pre-Experimental Design types that has Pre-Test before applying a treatment, and then lastly the Post-Test is included (Posavac, 2015). To use this design in this study, the comparison of test results is performed between Pre-Test (learning sorting using manual approach: text book) and Post-Test (learning using AVOMP app). In this manner, it can be determined whether changes in results or dependent variables have occurred.

However, this experimental design has a disadvantage if there is a big gap of time interval between Pre-Test and Post-Test which is called maturation of respondents (Campbell & Stanley, 2015) in which the researchers cannot tell whether or not the respondents solely affected by the treatment or got interfered by maturation of time. Thus, this experimental study is taken place in one day only which corresponds to the current lecture time in UUM. This is done due to the time constrain of this study and at the same time is also to minimize the maturation in the learning stage.

The respondents will be tested, concerning on KA measurement. The test will be given on three different topics of DSA (bubble sorting, insertion sorting, and selection sorting) as the scope of this study. The test itself is created and classified based on the levels of Bloom's Taxonomy (see Figure 3.5) (Forehand, 2010).

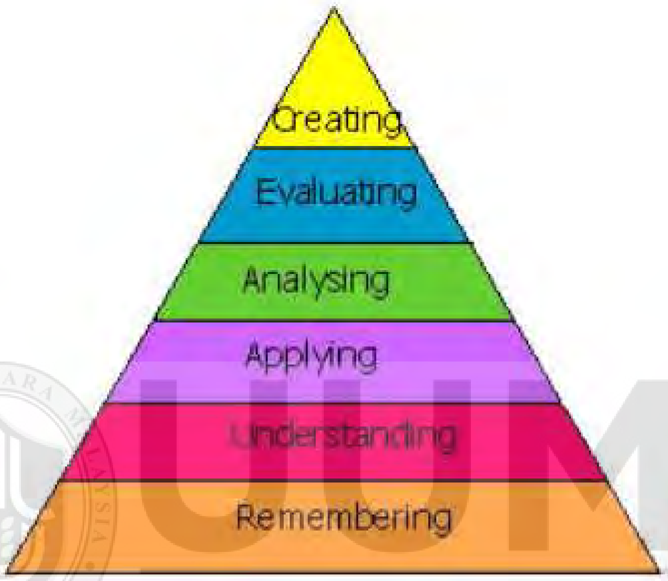


Figure 3.5. Levels of Bloom Taxonomy (Forehand, 2010)

All terms are defined according to Anderson, Krathwohl, and Bloom (2001), as follows:

- **Remembering**: Retrieving, recognizing, and recalling relevant knowledge from long-term memory.
- **Understanding**: Constructing meaning from oral, written, and graphic messages through interpreting, exemplifying, classifying, summarizing, inferring, comparing, and explaining.

- **Applying**: Carrying out or using a procedure through executing or implementing.
- **Analyzing**: Breaking material into constituent parts, determining how the parts relate to one another and to an overall structure or purpose through differentiating, organizing, and attributing.
- **Evaluating**: Making judgments based on criteria and standards through checking and critiquing.
- **Creating**: Putting elements together to form a coherent or functional whole; reorganizing elements into a new pattern or structure through generating, planning, or producing.

However, as the scope of this study, the creating and evaluation level are excluded in this study, considering the time constraint and limitation of students in UUM. This is due to these two levels have the most difficulty and the longest time to be solved (McGee et al., 2016). In addition, it needs more consecutive weeks to achieve those two levels (Selby, 2015).

Then, the experimental study started by asking participants to operate the prototype of AVOMP installed at their own pace, either on their tablet or smartphones. Eventually, the Pre-Test and Post-Test are conducted to see the differences of the learning results of DSA on sorting algorithm sub-topic. Questions shared in both pre-test (see Appendix C) and post-test (see Appendix D) adopt the questionnaire questions from (Ragonis, 2012).

Then, in order to analyze the related Pre-Test and Post-Test, paired sample t test is usually utilized as the first choice to determine whether or not there are significant difference between them. Another option that can be used is using Wilcoxon signed rank test. Therefore, there are 8 paired Pre-Test and Post-Test that are measured and compared in this study, which are:

1. Overall Pre-Test and Post-Test results in general.
2. Bubble Sorting Pre-Test and Post-Test results.
3. Insertion Sorting Pre-Test and Post-Test results.
4. Selection Sorting Pre-Test and Post-Test results.
5. Knowledge Bloom taxonomy level Pre-Test and Post-Test results.
6. Analysis Bloom taxonomy level Pre-Test and Post-Test results.
7. Understand Bloom taxonomy level Pre-Test and Post-Test results.
8. Apply Bloom taxonomy level Pre-Test and Post-Test results.

Then, the survey questionnaire is also distributed to capture the students' opinion towards the prototype (Vrachnos & Jimoyiannis, 2014). The questionnaire adopts the survey questionnaire by Amershi et al. (2005), Giordano and Carlisle (2006), and Jonathan et al. (2016) in which it has 10 statements. The statements number 1 until 3 are about the AVOMP app in general. The statements number 4 and 5 are about viewing activity. The statements number 6 and 7 are about proficiency test. The statements number 8 until 10 are about user interface. The Likert Scale on a five-point scale is applied on it to examine how strongly the learners agree or disagree towards statements with the following anchors: (1) Strongly disagree, (2) Disagree, (3) Neither Agree or Disagree, (4) Agree, (5) Strongly agree.

The followings are the statements given:

1. The AVOMP app is an effective tool for me to learn sorting algorithm.
2. The AVOMP app made sorting algorithm easy to learn
3. The AVOMP makes the lesson and the exercise of sorting algorithm more fun
4. The step by step process of visualization along with highlighted line of pseudocode as well as its explanation in viewing activity help me to understand each algorithm in depth.
5. The features in viewing activity, such as changing numbers to be solved and changing the numbers of arrays, help me to better understand the behavior of each algorithm.
6. The proficiency test by simulating the array visualization help me to understand sorting algorithm better.
7. The hints in proficiency test helps me to solve algorithm test
8. The design layout of AVOMP app is appealing for me to use the app.
9. The layout structure (arrangement, color, shape, consistency, and placement) help to me understand the flow of AVOMP app.
10. All textual contents (description, hints, information, questions, warning, etc.) in AVOMP app are clear and understandable.

3.3.3.2.3 Method of Experimental Study

Basically, this experiment compares the students' performance results on sorting algorithm sub-topic between using manual approach (slides and lecture's note) and prototype (AVOMP application). The flow of both tests is initially started with students learning manually the three sorting algorithms using slides and notes that lecturers have prepared. They have 40 minutes to study and then continue to do pre-test in 40 minutes. After that, they are directed to learn the sorting algorithms using AVOMP application in 40 minutes and then prompted to do the post-test in 40 minutes. To measure the general increase in term of the learning performance of all students, the Wilcoxon test is used instead of using paired-sample t-test (see Chapter 6 in detail) to compare the scores between Pre-test and Post-test.

3.3.3.3 Conclusion

The last phase is a conclusion in which the gathered findings from all previous phases are concluded by revisiting and answering all research questions and research objectives. Last, the contribution is fully given by presenting several publications and a full complete thesis.

3.4 Summary

In conclusion, this chapter has elaborated the research methods, which is adapted in this study. Specifically, the execution of methodology has three main parts: theoretical, development and evaluation. In brief, five phases of design science researched are adapted as follows: (a) awareness of the problem, (b) solution design, (c) evaluation, and (d) conclusion. The details of each phase are also discussed in which the activities are explained.

CHAPTER FOUR
RECOMMENDATION OF UI DESIGN AND INTERACTIVITY ASPECT OF AVOMP

4.1 Introduction

This chapter discusses the first stage of the study which is to identify the appropriate recommendation of AVOMP in terms of UI design and Interactivity aspects by conducting a comparative analysis of some studies. This chapter is especially aimed for answering the first research objective of the study. Hence, in order to extract those recommendations, there are two separated comparative analysis conducted in this phase. In the first comparative analysis, there are 12 studies are compared and categorized to identify the recommendations for UI design aspect. Meanwhile, the second comparative analysis has 11 studies to be compared and classified.

4.2 Related theories in constructing the proposed design guidelines

This study selects several theories that are related to UI design, which are Gestalt Theory, Safe Color Theory, Cognitive Load Theory (CLT) of multimedia learning, and design guidelines and usability for mobile learning. Additionally, theories or basis for interactivity, such as engagement taxonomy (ET), extended engagement taxonomy (EET), the two-dimensional engagement taxonomy (2DET), and constructivism learning theory are also included. These selected theories are really important in order to describe how the different behavior of design and interactivity could result in the way of students perceived learning material and thus result in their learning performance.

For instance, the Gestalt Theory explains that the different sets of design, such as spacing, grouping, and configuring could impact towards human perception (Guberman et al., 2012). In other words, if the design setting does not follow the rule of this theory, the miss-perception will occur due to the ambiguous meaning of visualization design (Jackson, 2008). Another theory that discusses human perception is CLT of multimedia learning. This theory states that human's brain has a limit in terms of cognitive capacity in working memory so that the learning material has to be designed carefully according to instructional design, reducing the cognitive load of students (Paas & Sweller, 2014).

As for interactivity, engagement taxonomy (ET) is about what the interactive features should be included in AV system in order to enhance the engagement of the students learning AV. Likewise, extended engagement taxonomy (EET) is essentially the additional features of ET which is also deliberately created to increase students learning performance. ET and EET are inspired from active learning theory, called constructivism learning theory which emphasizes the more students engage with the learning material, the more the cognitive efforts are made, and consequently affect their learning (Myller et al., 2009; Sorva et al., 2013).

All of these theories are really important as the comprehensive combination of creating AVOMP design guidelines in this study. However, not all recommendation of the theories is embedded, the selection is necessary as to fit the purpose of the study. Other than that, the other related previous empirical studies of both AV studies and mobile learning studies, previous AV design guidelines and so on are also considered in deriving the suggestion and recommendation for each guideline of

AVOMP. Therefore, in the next section, the considerations of the aforementioned theories are taken into account at the first stage before undertaking the construction of design guidelines of AVOMP, while performing comparative analysis in extracting recommendations of UI design and Interactivity.

4.3 UI Design and Interactivity Recommendations

This stage is in conjunction with the first objective of this study which is to find the appropriate recommendation for both UI Design and Interactivity aspects. UI Design is about the arrangement of the design or layout setting and symbol system (background, image, text, sound, icon, color, etc.) (Rogers, Sharp, & Preece, 2011). Meanwhile, the interactivity is about the engagement strategy, comprising several active learning features (viewing, responding, changing, constructing and presenting). Therefore, as mentioned aforementioned, there are two comparative analysis conducted in this phase.

4.3.1 Comparative Analysis of UI Design Aspect

In extracting the recommendation for UI design aspect, in the first part, 12 previous studies are compared in the perspective of UI design, which are the study of mobile design guidelines, previous AV design guidelines, UI design guidelines and Usability principle of designing a mobile app. Table 4.1 lists them all and categorizes based on their similarities.

Table 4.1

Comparative Analysis of Generic UI Design Recommendations

Category	Hujainah et al. (2016)	Shitkova et al. (2015)	(Hoehle, Aljafari, & Venkatesh, 2016)	(Ross & Gao, 2016)	(Sadikan & Yassin, 2013)	(Mirkovic, Bryhni, & Ruland, 2011)	(Lazaridis et al., 2013)	(Meolic, 2013)	(Simonák, 2013)	(Osman & Elmusharaf, 2014)	(Vrachnos & Jimoyiannis, 2014)	(Karam, 2015)
Content	Avoid much content in small screen	Avoid horizontal scrolling	Avoid much content	Avoid long content		Exclude all irrelevant information	Use interesting event in pseudocode	Remove extra explanation	Use multiple view for image and pseudocode	Separation for different elements	Use multiple views for AV elements (segmentation)	Avoid scrolling in small screen
		Use simple Sentence		Information /content needs to be concise and summarized				The use of icons/images should be limited		Show explanation for each step	Provide pseudocode and animation area	eliminating extraneous content
		Emphasis important information								Show pseudocode with highlighted color	Show pseudocode with highlighted line	Create cueing alert for essential information
Navigation Design	Have good navigation system	Make navigation as easy as possible	Make suitable size of fingertip buttons					Use good hierarchal presentation				Use pause strategies
Color			Use appropriate color	Use color friendly basis, not more than 5±2 different color	Use text and color to emphasize	Choose the right color for emphasize	Use proper color to highlight important activity		Use color to give attention	Use save color for displaying text		

Responsiv e-ness	Suitable for various mobile screen		Create responsi veness of the elements			All text and other Interface elements have to be responsive						Make the text responsive to prevent scrolling
Layout Design	Make options and workflow clear and obvious	Keep design consistent, simple, clear, and uniform	Make related elements closed each other	Layout needs to be simple and clear	Use relevant informa tion to describe animati on		Use picture and pseudo code concurrentl y		Put image and pseudocod e close each other			Integrating content whenever possible

The results from the comparative analysis are summarized and tabulated (see Table 4.2). In table 4.1, The UI design aspect is divided into 5 categories, which are content, navigation design, color, responsiveness, and layout design. However, since the "responsiveness" category has the final objective to avoid scrolling in the mobile app based on previous studies (Hoehle et al., 2016; Hujainah et al., 2016) so that it is included as the part of avoiding scrolling recommendation in "content" category. As a result, Table 4.2 presents 4 categories.

Table 4.2 also shows that most of the recommendations are about content's arrangement that advocate learning material in the mobile application has to be summarized to fit the screen size.

Table 4.2

The Details of UI Design recommendations that are classified from comparative analysis.

Category	Recommendations	Sources
Content	Avoid much content in small screen ***Note**: one of ways: using interesting event in pseudocode	Hujainah et al., 2016; Shitkova et al., 2015; Ross & Gao, 2016; Mirkovic et al., 2011; Meolic, 2013; Karam, 2015
	Avoid scrolling of contents	Hujainah et al., 2016; Shitkova et al., 2015; Hoehle et al., 2016; Mirkovic et al., 2011; Karam, 2015
	Give cueing to emphasis essential information ***Note**: for instance, can use highlighted line of pseudocode for each step connecting the student with the corresponding graphical visualization.	Shitkova et al., 2015; Lazaridis et al., 2013; Osman & Elmusharaf, 2014; Vrachnos & Jimoyiannis, 2014; Karam, 2015
	Use multiple views (segmentation)	Simonák, 2013; Osman & Elmusharaf, 2014; Vrachnos & Jimoyiannis, 2014

Navigation Design	Have good design of hierarchal navigation system	Hujainah et al., 2016; Shitkova et al., 2015; Hoehle et al., 2016; Meolic, 2013; Karam, 2015
Color	Use appropriate color	Hoehle et al., 2016; Ross & Gao, 2016; Sadikan & Yassin, 2013; Mirkovic et al., 2011; Lazaridis et al., 2013; Simonák, 2013; Osman & Elmusharaf, 2014
Layout Design	Make layout clear, simple, obvious	Hujainah et al., 2016; Shitkova et al., 2015; Ross & Gao, 2016
	Make the related elements close each other	Hoehle et al., 2016; Sadikan & Yassin, 2013; Lazaridis et al., 2013; Simonák, 2013; Karam, 2015

These generic recommendations were translated into the formation of AVOMP design guidelines. Beforehand, the next sub-section is about the comparative analysis on interactivity aspect.

4.3.2 Comparative Analysis of Interactivity Aspect

This is the second part of the comparative analysis, focusing on extracting recommendations for interactivity aspect of AVOMP design guidelines. In doing so, 11 previous studies are compared, and they are from the study of previous AV design guidelines, previous AV studies, and mobile design guidelines. Table 4.3 lists the findings based on their similarities and engagement taxonomy.

Table 4.3

Comparative Analysis of Generic Interactivity Recommendations

Category	Lee and Rößling (2011)	Alhosban and Hamad (2011)	Lazaridis et al., (2013)	Meolic (2013)	Boticki et al., (2013)	Simonák (2013)	Osman and Elmusharaf (2014)	Vrachnos and Jimoyiannis (2014)	Jonathan et al., (2016)	Hujainah et al. (2016)	Karavirta and Shaffer (2015)
Controlled Viewing			Provide step-through animation	Use navigational buttons for algorithm visualization		Allow student to step through the execution	Displaying step-by-step execution	Displaying step-by-step execution using control buttons	Show step-by-step algorithm mechanism	Provide back option	Show step-by-step algorithm mechanism
				Provide back button		Show information and pseudocode in viewing activity	Show pseudocode to understand the AV	Show pseudocode to understand the AV	Provide back button		Support rewinding button
				Accompany animation with pseudocode and explanation							Show dynamic explanation during visualization
Responding						Respond to relevant questions					Student can respond to relevant questions
Changing		Allow student to give the value to AV system	Users can insert their own values or use generated value for novice			Provide input data for algorithm		Allow modifying input data for algorithm	Provide input generator for random values and also input area for students' value		AV should provide custom input

Constructing	Use constructing approach as experiment learning	Allow student to manipulate the object on the screen			Use swapping or switching gesture for constructing activity		Provide input area to simulate the algorithm, such as add, delete, search			Offer informative feedback	AV should support constructing activity in form of exercises
	Use direct manipulation for student's experiences				Give feedback (guide or info) during constructing activity		Give hints, guides, error message, comments, etc.				Give feedback during constructing activity
Presenting	Present and invite the other students for getting feedbacks and comments								Present in front of the class about how the behavior of visualization		Student can present their skill during viewing activity
Control setting		The student should be able to choose either to use narrative or text only.	AV should be able to provide the execution speed control setting	Pop-up information to see other information or set the environment	Control setting is provided to choose other options	Student should be able to adjust the speed of animation			Provide environment setting (e.g. the visualization style)	Provide visualization setting	Allow customization to the system, such as background, transition speed, etc.
Error Handling							Give error message when users make a mistake	Detect error when student insert the wrong value		Prevent and handle the error (e.g. give error indication/message)	

As displayed in table 4.3, the interactivity aspect is divided into 7 categories, which are viewing, responding, changing, constructing, presenting, control setting and error handling. However, in this study, the presenting is excluded due to this guideline is aimed at students. Presenting is an engagement taxonomy (ET) category that suggests learner presenting their proficiency in viewing and constructing activity in front of other students in order to get others' response and feedback (Wong). Meanwhile, this study only focuses more on design guidelines of algorithm visualization. Therefore, the results from comparative analysis of generic interactivity recommendations (see Table 4.3) for interactivity aspect are summarized and tabulated into 6 categories (without presenting category) (see Table 4.4).

Table 4.3 also shows that most of the recommendations do not implement all engagement taxonomy completely. The only one who implements all is the study from Karavirta and Shaffer (2015). Nevertheless, in general, there is still lack of explanation and implementation from aforementioned studies about how to design the engagement taxonomy adapting on mobile screen (the actual example and arrangement). Additionally, as mentioned in chapter 1, there is still lack of comprehensive studies that discuss both perspectives (UI design and interactivity) generally and on mobile platform specifically. Therefore, it is important in this study to integrate those recommendations into one package. Next section describes further the construction of the design guidelines of AVOMP for each recommendation above.

Table 4.4

The Detail of Interactivity Recommendations That Were Classified from Comparative Analysis

Category	Recommendations	Sources
Controlled Viewing	Displaying step-by-step execution of algorithm visualization using control buttons	Lazaridis et al., 2013; Meolic, 2013; Simonák, 2013; Osman and Elmusharaf, 2014; Vrachnos and Jimoyiannis, 2014; Jonathan et al., 2016; Karavirta and Shaffer, 2015
	Provide rewinding button	Meolic, 2013; Jonathan et al., 2016; Hujainah et al., 2016; Karavirta and Shaffer, 2015
	Show dynamic explanation during visualization (pseudocode and textual explanation)	Meolic, 2013; Simonák, 2013; Vrachnos and Jimoyiannis, 2014; Karavirta and Shaffer, 2015
Responding	Provide relevant questions to be answered by students	Simonák, 2013; Osman and Elmusharaf, 2014; Karavirta and Shaffer, 2015
Changing	Provide modifying input data for algorithm and also can generate random values	Alhosban and Hamad, 2011; Lazaridis et al., 2013; Simonák, 2013; Vrachnos and Jimoyiannis, 2014; Jonathan et al., 2016; Karavirta and Shaffer, 2015
Constructing	AV should provide constructing activity	Lee and Rößling, 2011; Alhosban and Hamad, 2011; Boticki et al., 2013; Osman and Elmusharaf, 2014; Karavirta and Shaffer, 2015
	Give feedback during constructing activity	Boticki et al., 2013; Osman and Elmusharaf, 2014; Karavirta and Shaffer, 2015; Hujainah et al., 2016
Control Setting	Provide environment setting	Alhosban and Hamad, 2011; Lazaridis et al., 2013; Meolic, 2013; Boticki et al., 2013; Osman and Elmusharaf, 2014; Jonathan et al., 2016; Hujainah et al., 2016; Karavirta and Shaffer, 2015

Error Handling	Prevent and Handle the error	Osman and Elmusharaf, 2014; Vrachnos and Jimoyiannis, 2014; Hujainah et al., 2016

4.4 Summary

The recommendations of both aspects, namely UI design and Interactivity have been identified in this chapter through comparative analysis approach. Afterward, the identified recommendations are grounded and constructed more based on relevant theories and empirical studies into comprehensive guidelines. The next chapter explains in detail the development of AVOMP design guidelines.

CHAPTER FIVE
THE CONSTRUCTION OF AVOMP DESIGN GUIDELINES

5.1 Introduction

This chapter discusses the systematic processes in developing the design guidelines of AVOMP in addressing the second research objective and also discussion about expert review to validate the design guidelines is elaborated in addressing the third research objective. Based on the identified recommendations from the previous chapter, each guideline is then elaborated further with 4 structures, which are basis (the source from theories or studies), explanation (as justification), implementation (how to do), and illustrations. The development of AVOMP design guidelines in general involves content analysis, comparative analysis, and expert review. The following sections explain how the theories relate to the development of proposed design guidelines and describe each activity in detail which lead to the proposition of the artifact.

5.2 The Construction of AVOMP Design Guidelines

Each guideline is grounded on relevant theory and empirical studies (Hoehle et al., 2016; Shitkova et al., 2015). Therefore, each guideline is a comprehensive form that contains some components, not only recommendation, but also basis (sources/theories), explanation, implementation, and illustration. In other words, each of recommendations that have been resulted from the previous stage will be clarified and constructed more as the complete guidelines that fulfill those components. Figure 5.1 exhibits the components of each guideline.

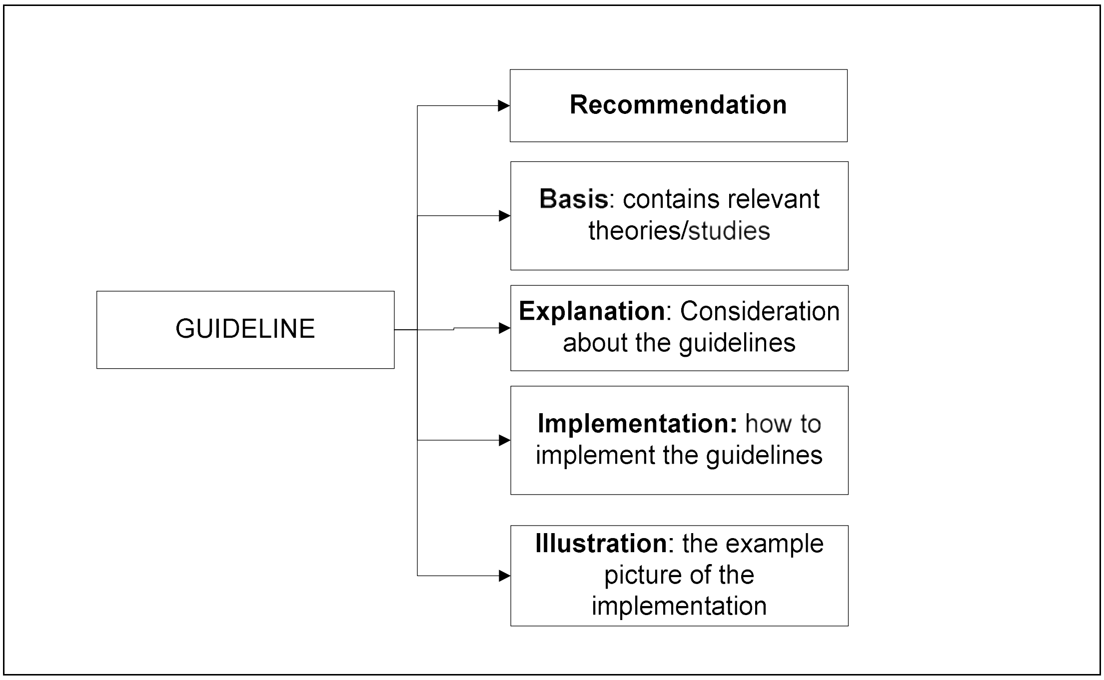


Figure 5.1. Components of AVOMP design guidelines

Therefore, the content analysis towards relevant theories and empirical studies for UI design and interactivity are conducted in order to justify and strengthen each guideline in this study. This content analysis has been carried out through comprehensive reading, searching, and reviewing from previous studies. In detail, it can be seen in Chapter 2, such as Interesting Event approach, Cognitive Load Theory, split attention effect, Usability principles, Constructivism Learning Theory, ET, EET, etc. The next sections present further those terms in the guidelines and its components.

5.2.1 Design Guidelines of AVOMP in Perspective of UI Design Aspect

The table 5.1 below explains in detail about each guideline of UI design aspect. The "G" initial in table 5.1 stands for a guideline.

Table 5.1

UI Design Aspect

No	Category 1: Content

Recommendation: Avoid much content on small screen

Basis: Coherence principle of multimedia learning by Mayer (2002)

Explanation: This advocated from coherence principle by Mayer (2002) that students will learn better when the extraneous materials are removed in which the cognitive overload is minimized. Removing the extraneous content in reducing cognitive load is always becoming the first recommendation from many studies of multimedia learning (Ayres & Paas, 2012; Fiorella & Mayer, 2014; Mayer, 2009; Sung & Mayer, 2013).

Implementation:

1. In AV system itself, the content, such as information, guidance, warning, etc., can be summarized by putting a simpler and direct sentence.
2. This includes abolishing of irrelevant background music (Brunken et al., 2003).
3. Avoid background decorations (Fiorella & Mayer, 2014).
4. Use Interesting Event approach that can summarize the focal point in pseudocode (see Figure A).

G1

5. Avoid adding spoken sound towards textual information since it will create redundancy effect (Mayer & Moreno, 2003; Sweller, 2011).

Illustration:

```
void bubblesort(Comparable[] A) {
  for (int i=0; i<A.length-1; i++)
    for (int j=1; j<A.length-i; j++)
      if (A[j-1].compareTo(A[j]) > 0)
        swap(A, j-1, j);
}
```

Figure A. Interesting Event Example

The figure above shows an example of an interesting event for swapping process. The swapping code usually presented with 3 lines for the full code but the interesting event is useful to summarize it into one-line code to represent the whole process.

Recommendation: Avoid Scrolling

Basis: Split-attention effect in multimedia learning

Explanation: The scrolling while reading learning material could create split attention effect and is strongly suggested to be avoided (Churchill, 2011; Luong & McLaughlin, 2009). A study from Sanchez and Wiley (2009) found that this happens due to the eye needs to allocate the focus many times while the text moves up and down. Sanchez and Goolsbee (2010) also confirmed through their experimental study that scrolling gives negative effects on students' text comprehension, especially using mobile phone with a small screen. This is consistently in line with the study of Sanchez and Branaghan (2011) stated that
G2 scrolling is not appropriate for mobile learning because it will affect the recall issue during the reading text.

Implementation:
1. Do not make the design of mobile app exceed the size of mobile screen
2. Choose important UI elements only (text, image, button, etc.)
3. Set properly the size of AV system elements (pseudocode text size, visualization size, buttons, etc.).
4. Make UI elements to be responsive that can suit and adapt the various size of the screen. This way will prevent from extending the learning material that does not exceed the size of the screen; otherwise, it will force to have scrolling within an app (Sanchez & Goolsbee, 2010).

Recommendation: Give cueing to emphasize the essential information

Basis: Multimedia learning

Explanation: Cueing or also known as signaling has an advantage in terms of decreasing the split-attention effect. This is because the cuing gives direct focus to a focal point of essential information and also is used to make a connection towards a corresponding animation or image (Liu, Lin, & Paas, 2013). The studies (Liu et al., 2012) and (Liu, Lin, et al., 2013) found that cuing in form of arrow line has significantly improved the students learning on science lesson through a mobile application.

G3

Implementation:

1. Highlight the pseudocode line by line during the animation of a particular algorithm (see Figure B).
2. Give cues in visualization, such as arrow signs in showing swapping process, showing the movement of values (see Figure C).

Illustration:

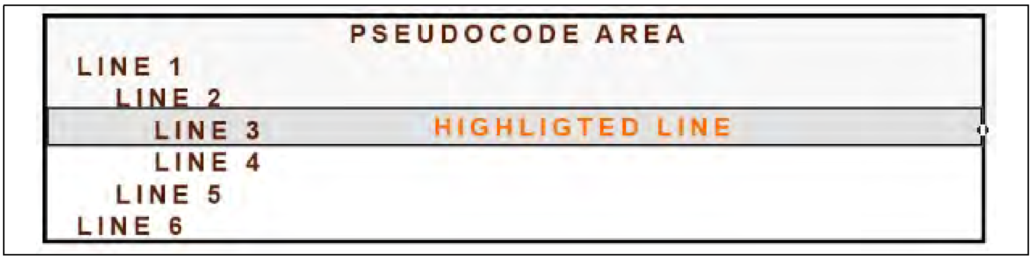


Figure B. Highlighted line of pseudocode during step-by-step animation

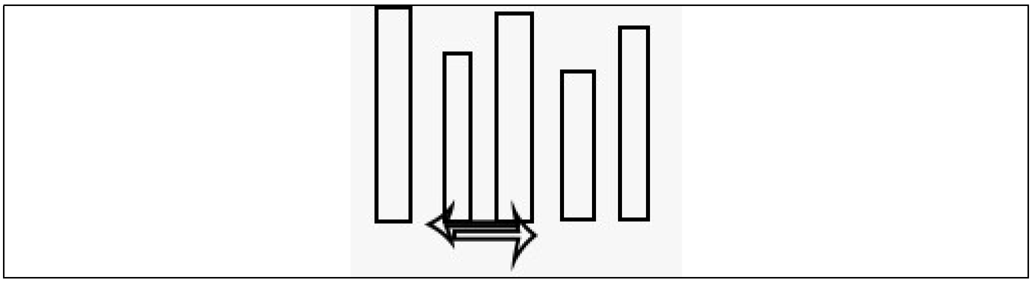


Figure C. Two arrow lines animation show the swapping process

Recommendation: Use multiple views (segmentation).

Basis: Cognitive load theory

Explanation: Segmentation is used where the information is divided into pieces of meaningful information, reducing the cognitive overload since it will signal the learners the organization of the information (Florax & Ploetzner, 2010; Sung & Mayer, 2013). Moreover, segmentation reduces the overwhelming of working memory as it gives pauses effect between segments of dynamic information (such as animation), which improves learning comprehension on learning material (Karam, 2015). Thus, it is also suggested to facilitate the learners with a control panel for segmenting the continuous
G4 material in minimizing superfluous load (Ayres & Paas, 2012; Wong, Leahy, Marcus, & Sweller, 2012). This way will allow students to pause and catch up with the reading material before they can proceed to the next step of new segmented material (Spanjers, Van Gog, & van Merriënboer, 2010).

Implementation:

1. Use line/box/white space as a gap that separates between one element and others.
2. Divide the animation into pieces of information, processed sequentially based on line by line of an algorithm pseudocode with step-by-step navigation buttons (next and backward), rather than using play button (see Figure D).

Illustration:

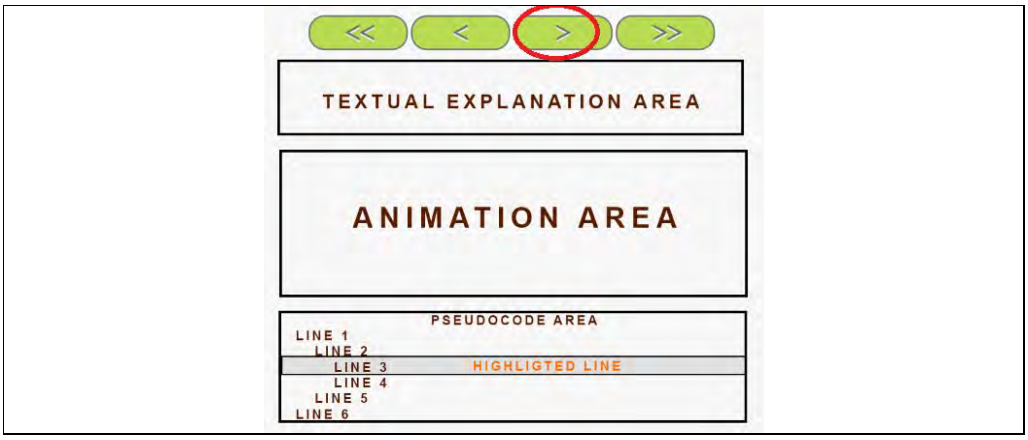


Figure D. The segmentation in animation and area

The illustration 4 above shows that the animation area will be shown segmentally based on the step-by-step process in pseudocode as the student clicks the next arrow button. The figure D also shows that each different area is segmented inside the box (textual explanation area, animation area, and pseudocode area) in order to distinguish between one and the others.

Category 2: Navigation Design

Recommendation: Have a good design of hierarchal navigation system

Basis: Cognitive load theory, Usability principle

G5 **Explanation**: Navigation has to be simple and clear in terms of information and content. The navigation indicates as a cue, helping the learner see the connection between elements (Liu, Lin, et al., 2013). If the learner sees the navigation has ambiguous meaning that leads to the mismatched expectation of user's action then it will lead to the decreased performance level (Ross & Gao, 2016). Another factor that needs to consider is the size of navigation buttons, as a study (Cockburn, Ahlström, & Gutwin, 2012) found from their experimental study that the high error rate occurred due to the size of fingertip does not match with the size of touch buttons. They called this issue as "fat finger" problem.

Implementation:

1. Make the cue (as a breadcrumb or header) on the page that indicates the connection after clicking the main menu buttons
2. The simple design of navigation that has no additional icon or image.
3. Have consistency in terms of the design style of AV navigation buttons as well as its position in many pages
4. Set the size of navigation buttons that fit fingertip size (Hoehle et al., 2016)

and also consider the gap (white space) between each button for small buttons.

Illustration:

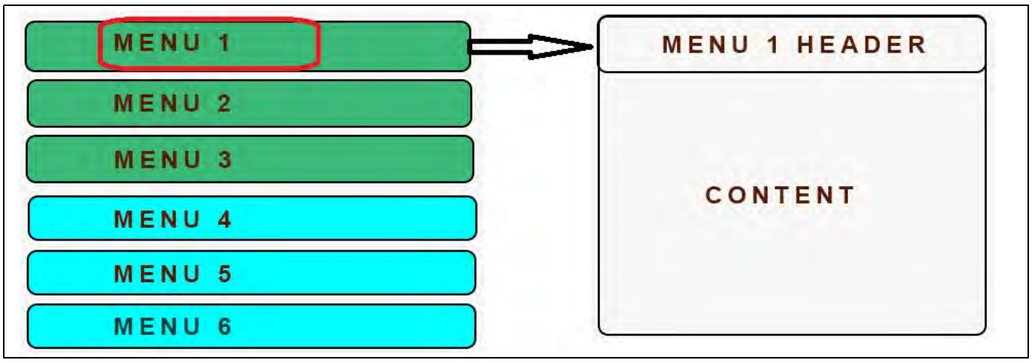


Figure E. Menu 1 header is used to inform a user after clicking button

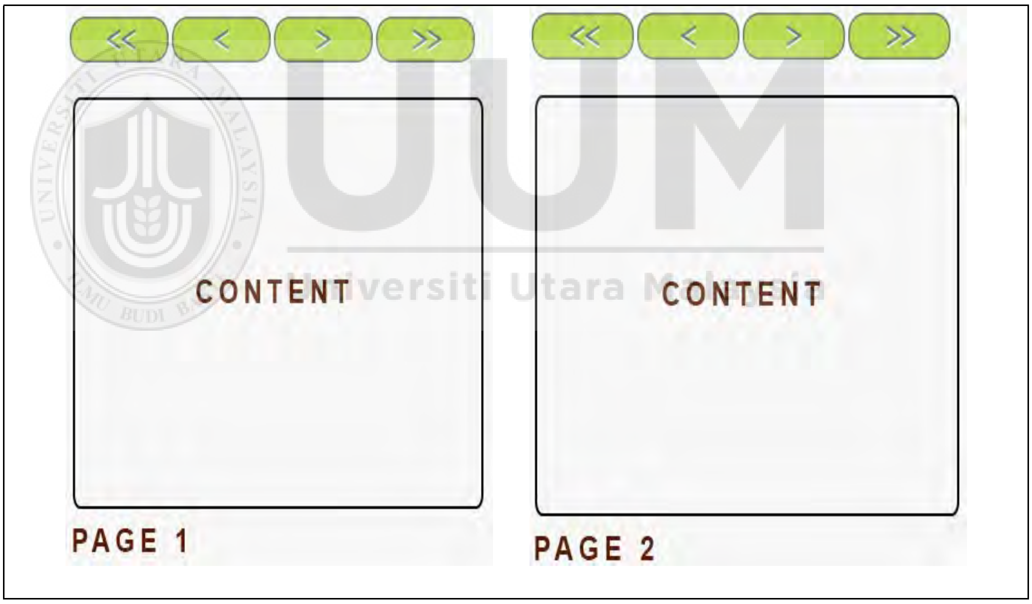


Figure F. The consistency of navigation buttons' position in many pages

Figure F shows the consistency in terms of buttons' position. Additionally, buttons have to fit fingertip size and are separated with a gap of white space in decreasing the miss-click error rates.

Category 3: Color

Recommendation: Use appropriate color

Basis: Visual cuing of cognitive load theory, color theory, usability of mobile application

Explanation: The appropriate use of color could decrease the cognitive load, for instance, the text should have sufficient contrast with the color of the background to meet the readability requirement (Richardson, Drexler, & Delparte, 2014). Color could be also used as cuing element for highlighting the essential information, decreasing the split attention effect (Lazaridis et al., 2013; Mirkovic et al., 2011) and giving guidance to users (Brandse & Tomimatsu, 2014). Color could assist in terms of grouping information, focusing on particular content, making differentiation, and establishing relationships between elements (Hoehle et al., 2016).

Implementation:

G6

1. Use color to group bunch of similar information (see Figure G).
2. Use color in visualization to highlight the dynamic movement of algorithm processes (see Figure H).
3. Use color to highlight the line (as a current activity) in pseudocode during visualization (see Figure H).
4. Avoid utilizing more than 5±2 different colors (especially in pseudocode) because it can overwhelm students' recall in short-term memory (Ros and Gao, 2016)

Illustration:

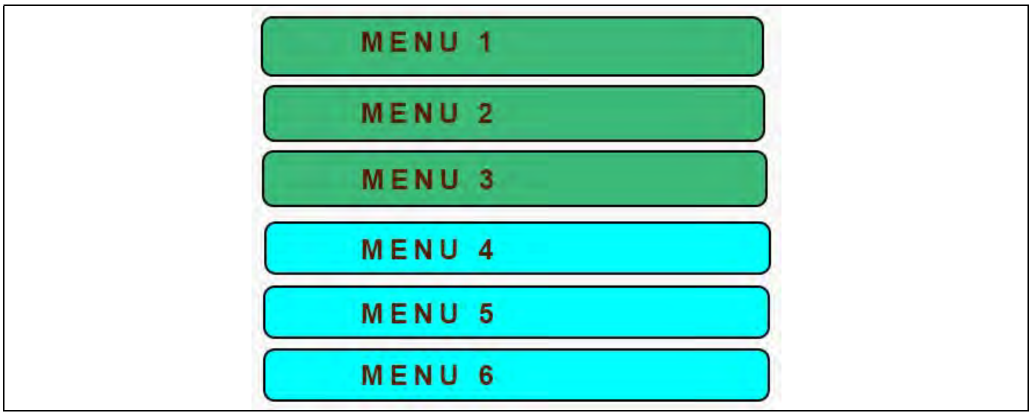


Figure G. The use of color to group the similar information in buttons

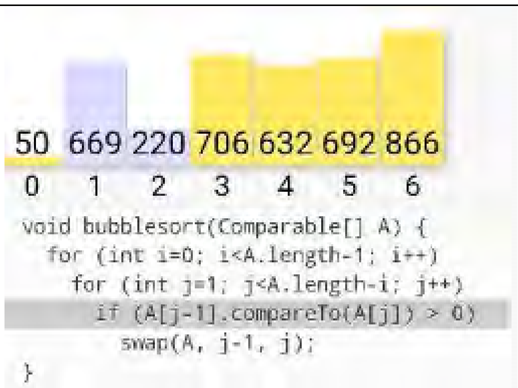


Figure H. The use of color to highlight the dynamic movement in animation and highlight the current line in pseudocode

Figure H is the example of bubble sorting algorithm that shows the use of color to highlight the processed element in both animation and pseudocode parts.

Category 4: Layout Design

Recommendation: Make layout clear, simple and obvious

Basis: Usability of mobile application, Gestalt theory

Explanation: The layout needs to be simple (Wilson, Shortreed, & Landoni, 2004), clear, uniform, and consistent (Nathan & Yeow, 2011; Zollet & Back, 2010). It is also recommended to make information and content organized based on gestalt principle. The gestalt theory of similarity principle, for instance, suggests that the elements that have similarity in characteristic (shape/type/form/style/color) can be grouped in order to make the design clear, obvious, and easy to be recognized (Hoehle et al., 2016). This will make learners easy to grasp the content organization and thus improve their learning experiences.

G7 **Implementation**:

1. Make the design of UI elements uniform and consistent in terms of color, style, shape, and position.
2. Use the similar shape, color and look for navigation buttons in the system.

Illustration:

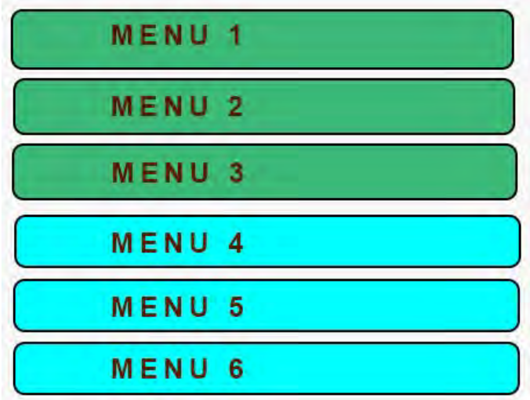


Figure I. The similarity in terms of shape, look and color for menu buttons

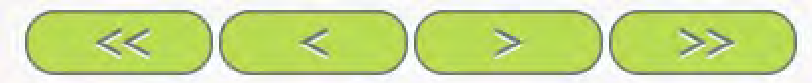


Figure J. The similarity of control buttons of algorithm visualization

There will be many algorithms to be visualized in the system. Therefore, it is important to be consistent to use the same style or model for UI elements for instance control buttons for each page of algorithms, otherwise, the students will need more time to comprehend the learning material.

Recommendation: Make the related element close each other

Basis: Multimedia Learning: Spatial contiguity principle, Gestalt theory

Explanation: Spatial contiguity principle states that whenever possible it is
strongly suggested make two related elements (e.g. image and text) near,
reducing split attention effect of learners (Mayer, 2009). Additionally,
G8 proximity principle of Gestalt theory also suggests making the related element
close to each other, since it will be perceived by learner as one group, and as a
result, will be examined or read together (Graham, 2008; Guberman et al.,
2012).

Implementation:

Make the same elements proximate to each other, such as buttons, visualization, and its pseudo code.

Illustration:

Figure K. Proximity of related elements

Table 5.1 shows the relevant theories and explanation for each recommendation. It is imperative in this study to get a deeper understanding of the theory and its explanation behind each recommendation before forming the comprehensive guidelines of AVOMP. The next sub-section deliberates the explanation for each recommendation of the second aspect, which is interactivity.

5.2.2 Design Guidelines of AVOMP in Perspective of Interactivity Aspect

Table 5.2 Interactivity Aspect

No	Category 1: Controlled Viewing (First engagement taxonomy level)
G9	**Recommendation**: Displaying step-by-step execution of an algorithm using control buttons **Basis**: Cognitive load theory (CLT), Engagement Taxonomy (ET), Extended Engagement Taxonomy (EET) **Explanation and Implementation**: As the core engagement taxonomy, viewing means the leaner can view the animation sequentially. This means the concept of an algorithm is divided into meaningful chunks (in line with G4). This will minimize the cognitive load of students during viewing activity; instead of showing all the codes which will overwhelm the student's working memory. Originally viewing is from ET, listed by Naps et al. (2002), but then has been revised by Myller et al. (2009) as an additional category, which is called controlled viewing. The difference is viewing just only allow the learner to watch the visualization passively, while the controlled viewing provides control setting for the user so that the learners can control the environment of visualization, either changing the animation speed or visual elements (Myller et al., 2009; Sorva et al., 2013). **Implementation**: Provide navigation buttons for controlling the visualization that will be dynamically animated sequentially as the student clicks on it. Illustration:

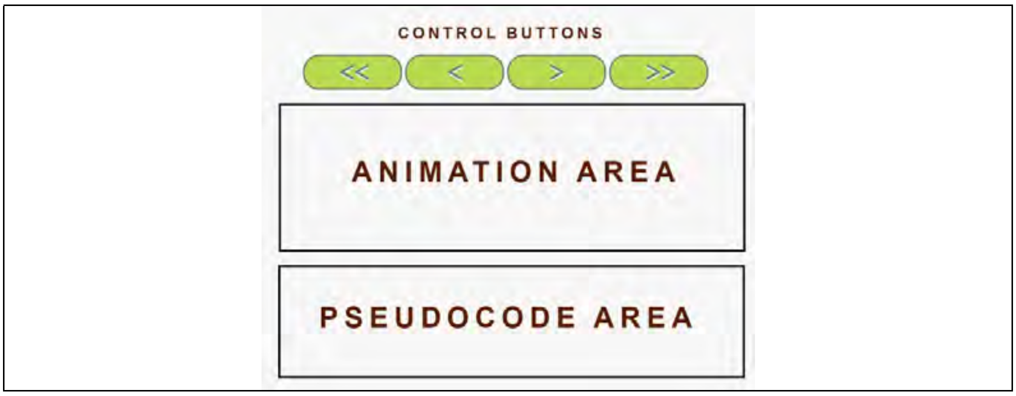


Figure A. The controlled viewing for algorithm visualization

Recommendation: Provide rewinding button

Basis: Extended Engagement Taxonomy (EET)

Recommendation: This suggests AV should provide a backward option during viewing visualization. Some AV studies (Boticki et al., 2013; Osman & Elmusharaf, 2014; Simonák, 2013; Vrachnos & Jimoyiannis, 2014) do not provide this option on their navigation buttons; instead, they just provide forward and play button only. In fact, this option will be useful whenever students feel lost track of the visualization and are still confused (Juett, 2016; Karavirta & Shaffer, 2015). Therefore it is also recommended to put a notification in form of the page number of visualization so that the learners will be notified as they move forward or backward (Karavirta & Shaffer, 2015).

G10

Implementation:

1. Provide rewinding button in visualization
2. Put the number of current steps and its total number of executions.

Illustration:

Figure B. Backward control button of algorithm

Recommendation: Show dynamic explanation during visualization (pseudocode and textual explanation)

G11

Basis: Engagement Taxonomy (ET), Extended Engagement Taxonomy (EET), Temporal contiguity principle of multimedia learning

Explanation: During viewing activity, it is recommended to also show the dynamic pseudocode and textual information that is traced line by line corresponded with its visualization (Meolic & Dogša, 2014; Osman & Elmusharaf, 2014; Simonák, 2013; Vrachnos & Jimoyiannis, 2014). This is supported by temporal contiguity principle of multimedia learning that recommends presenting related elements simultaneously rather than consecutively in order to minimize cognitive load (Mayer, 2009). In addition, Urquiza-Fuentes and Velázquez-Iturbide (2007) suggested: that the utilization of narratives and textual contents incorporated with visualization could increase the effectiveness of AV. This is in line with a study by Urquiza-Fuentes and Velázquez-Iturbide (2009a) that reported involving dynamic textual content during visualization have improved students' knowledge acquisition.

Implementation:

1. Make a highlighted line of pseudocode dynamically changed as the user proceeds to the next step of visualization (see Figure C).

2. Provide textual explanation which is dynamically changed during visualization, informing the process of each step of pseudocode (see Figure C).

Illustration:

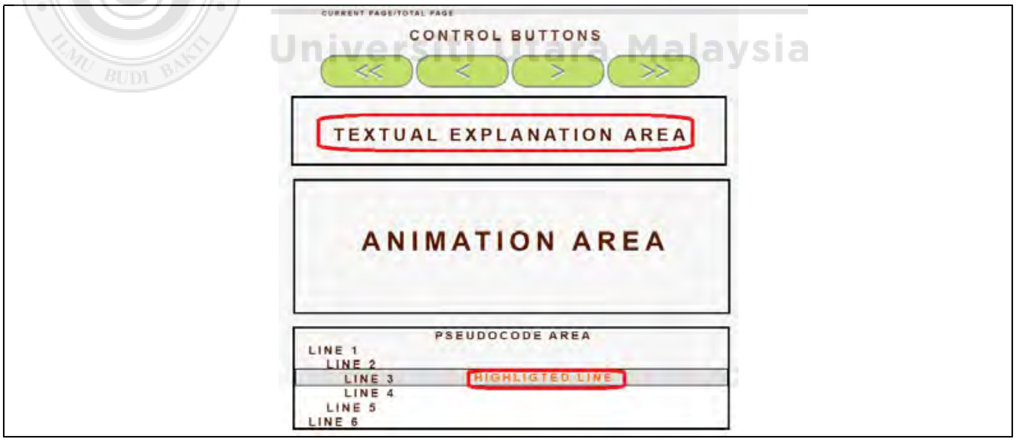


Figure C. The Textual description changes during viewing activity

Category 2: Responding

Recommendation: Provide relevant questions to be answered by students

G12 **Basis**: Engagement Taxonomy (ET)

Explanation: Responding is from ET suggests that AV system should provide relevant question regarding a certain algorithm, either during viewing activity or after it.

Implementation: (**choose one of ways**)

1. Provide the quiz form with multiple choices; the questions are related to a specific algorithm concept.
2. Make Pop-up question during viewing activity, prompting the learner to guess or predict the next move of visualization.

Illustration:

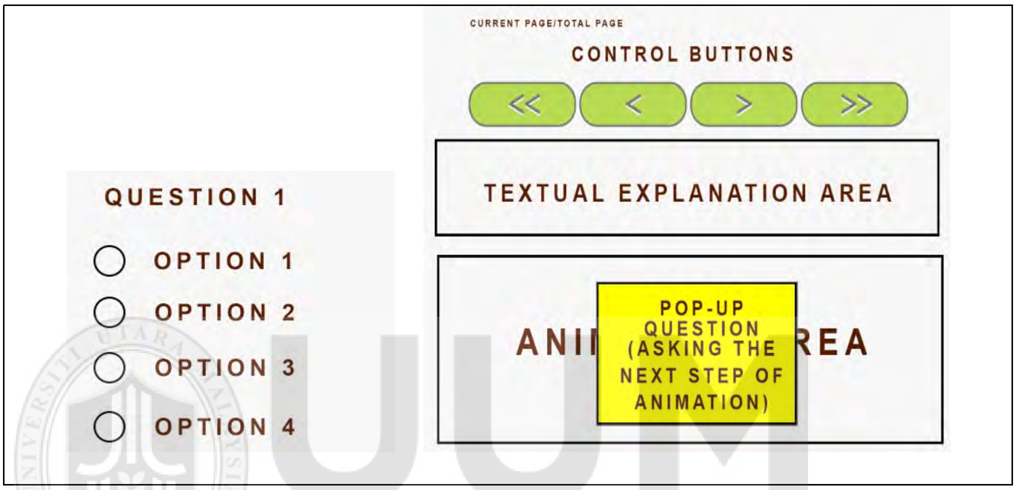


Figure D. Multiple choices and pop-up questions

Category 3: Changing

Recommendation: Provide opportunity to modify input data for algorithm and option to generate the random values

G13

Basis: Engagement Taxonomy (ET), Two-Dimensional Engagement Taxonomy (2DET)

Explanation: Changing is included in ET which means the learner can change the value of variables in order to see the different behavior of a particular algorithm in viewing activity (Jonathan et al., 2016; Naps et al., 2002). In this case, Sorva et al. (2013) listed new terms in the Two-Dimensional Engagement Taxonomy (2DET) that divides the input into two types which are given content and own cases. Given content means the student can learn the AV through predefined/given value. On the other hand, the own cases mean the student learns the AV by modifying the inputs of variables.

Implementation:

1. Provide a button that can generate the random values (Given content by AV system). The button "run" for example in Figure E enables user to generate random numbers to be visualized.
2. Provide a drop-down button for selecting a number of array (see Figure E). This allows user to change the number of array that will be visualized by system.
3. Provide a text box to be inserted by users as their own cases to be visualized by system.

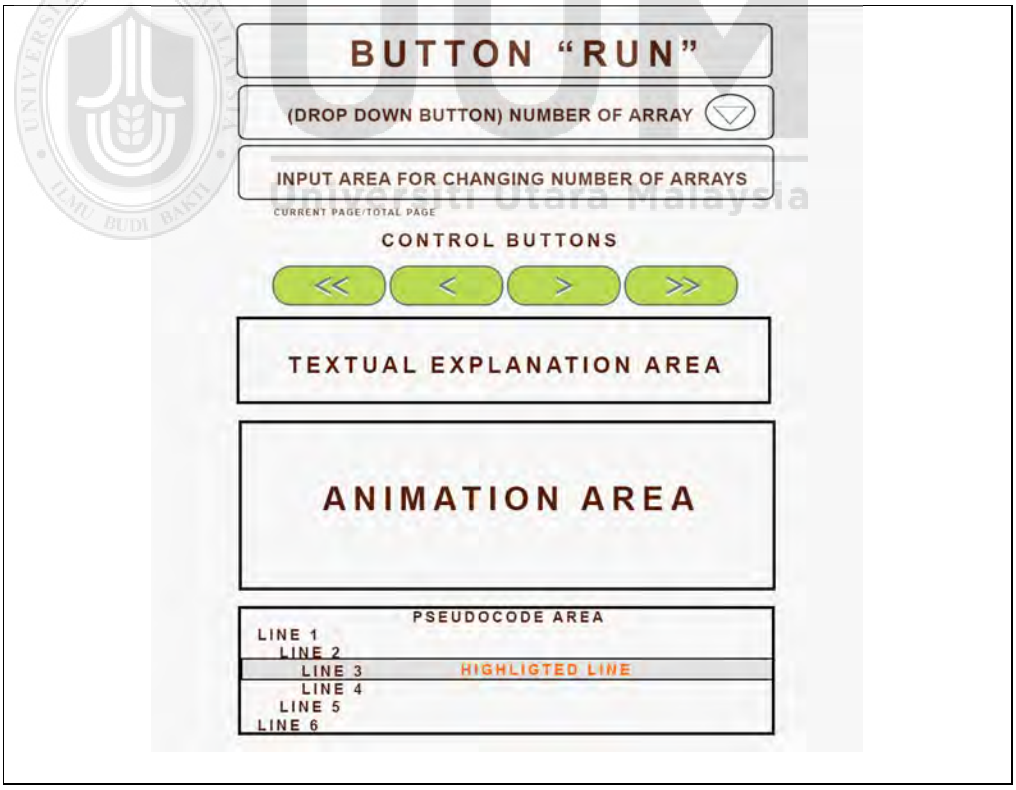


Figure E. Changing features

Category 4: Constructing

Recommendation: AV should provide constructive activity

Basis: Engagement Taxonomy (ET), Two-Dimensional Engagement Taxonomy (2DET), Constructivism Learning Theory

Explanation: Constructing activity is grounded from the constructivism learning theory, which means learning or understanding processes are based on human experiences itself (Büyükduman & Şirin, 2010). In a sense, that constructivism necessitates the learner to construct their own knowledge through activities or interactions with learning material itself such as "creating, constructing, developing and inventing (Guney & Al, 2012). Specifically, constructing activity in ET and EET means the activity that requires the student to learn through simulating the visualization itself, touching or interacting with the visualization elements (Myller et al., 2009; Naps et al., 2002). A study by Hundhausen and Brown (2008) found that learners using their AV system (ALVIS Live!) that consists of developing algorithms and constructing visualizations surpassed the other group using presentation media and traditional text editor. This is also the same with a study (Urquiza-Fuentes & Velázquez-Iturbide, 2007) that compared viewing group and building visualization group on tree traversal algorithm; they found the building group spent more time (more engaging) than viewing group, and thus it affected on better performance result by building group.

G14

Implementation:

Provide exercise activity that requires the student to demonstrate and solve the steps of a specific algorithm based on given questions by clicking or swapping the visual elements (see Figure F).

Illustration:

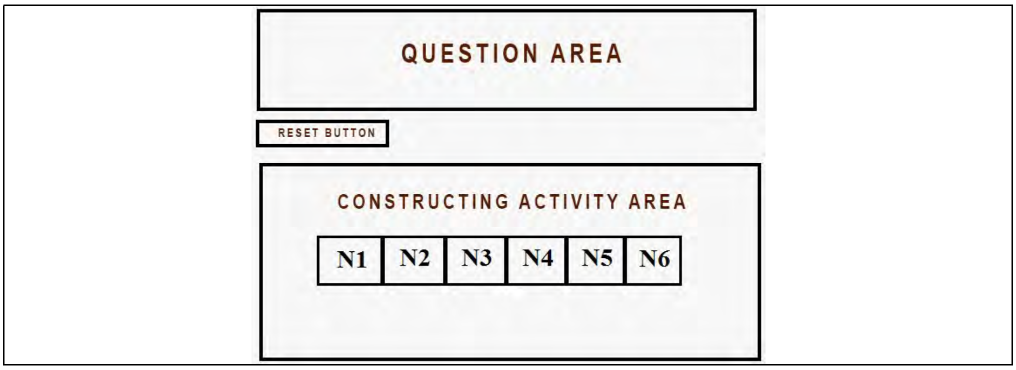


Figure F. Constructing activity example

The Figure F shows a set of numbers (N1-N6) in a case of sorting algorithm activity example. The student will be asked to swap the boxes based on the given question of the particular sorting algorithm. Then need to use sorting or swapping gesture by touching the boxes.

Recommendation: Give feedback during constructive activity

Basis: Engagement Taxonomy (ET), Human Computer Interaction (HCI) principles

Explanation: The availability of the feedback during construction will support students in terms of recognizing the error and also improve their comprehension towards a particular algorithm (Mulvey, 2015). Feedback is also an example of HCI design principle that declares clear feedback in response to student's action could improve the usability of the application (Rogers et al., 2011).

Implementation:

Provide feedback such as buttons that can inform the guidance, error message, correct answer, and info.

G15

Figure G. Feedback buttons for constructing activity

Category 5: Control Setting

Recommendation: Provide environment setting

G16

Basis: Usability of mobile learning, Cognitive load theory

Explanation: This suggests that learners are supposed to have the freedom to control the application, making them feel that they can control the environment. This way will enhance the satisfaction of the learner itself since they feel that they have the flexibility to change the setting anytime they want instead of required to reply the action (Hujainah et al., 2016). Additionally, providing learning-pace setting for users is also highly recommended to be included because it is successfully many times proved by many studies (Savoji, Hassanabadi, & Fasihipour, 2011; Schmidt-Weigand, Kohnert, & Glowalla, 2010; Schüler, Scheiter, & Gerjets, 2013; Spanjers et al., 2012; Sung & Mayer, 2013) in decreasing the cognitive load.

Implementation:

Provide the control panel that entails some settings, such as animation speed of AV, visualization style, background color, etc.

Category 6: Error Handling

Recommendation: Prevent and handle error

Basis: Usability of mobile learning

G17

Explanation: This suggests that the system should be able to prevent the serious error and also can detect the error if the users make a mistake, such as input or value (Hujainah et al., 2016; Osman & Elmusharaf, 2014). Then, the system will provide such simple and constructive instructions to recover.

Implementation:

1. Make a prevention by using disabled UI elements which will be enabled after users achieve a certain condition (e.g. pressing a run button in AV system)
2. Give an alert or warning message to learner if they make a mistake.

5.3 Expert Review

As discussed in Chapter 1, before the proposed AVOMP design guidelines are validated through prototyping, the expert review needs to be conducted to review and evaluate the proposed design guidelines. Therefore, 20 invitations were sent to the identified experts via email communication. Experts in this review stage are nominated based on the following criteria:

1) Have Ph.D. qualifications either in Computer Science (CS) or Human Computer Interaction (HCI) or Instructional Design Expert or Multimedia or Software Engineering (SE) related area **or/and**

2) Have been researching/teaching HCI/SE/CS for at least five years **and**

3) Have at least five years teaching background for one of the following subjects: data structure and analysis (DSA), algorithm, programming, for at least 5 years

Out of the 25 experts, 16 of them approved to take part as experts. This number is sufficient as supported by Keeney and von Winterfeldt (1991), Nielsen (1997) and Chang, Kaasinen, and Kaipainen (2012). Table 5.3 shows the experts' demographic profile.

Table 5.3

Experts' Demographic Profile

Expert	Gender	Education	Field of Expertise	Experiences (Year)	Affiliations
A	Female	PhD	Multimedia	10	Universiti Teknologi Mara (UiTM) Melaka
B	Female	PhD	Multimedia Design and Development	10	Universiti Teknologi Mara (UiTM) Perlis
C	Female	PhD	Multimedia in Education	20	IPG Kampus Sultan Abdul Halim, Sungai Petani
D	Female	PhD	Human Computer Interaction,	20	Universiti Teknologi Mara (UiTM) Perlis

E	Female	PhD	Multimedia Information System, Mobile Learning	17	University of Malaya
F	Female	PhD	Mobile Application Development, Interaction Design	12	Universiti Teknologi Mara (UiTM) Perlis
G	Male	PhD	Educational Technology	13	Universiti Tun Hussein Onn Malaysia
H	Female	PhD	Multimedia	14	Universiti Teknologi Mara (UiTM) Perlis
I	Female	PhD	Software Engineering	18	Universiti Putra Malaysia (UPM)
J	Female	PhD	User Experiences, Usability	22	Universiti Malaysia Pahang (UMP)
K	Female	PhD	Multimedia in Pedagogy and Education, Usability	15	Universiti Teknikal Malaysia Melaka (UTeM)
L	Female	PhD	Multimedia, Human Computer Interaction	20	Universiti Putra Malaysia (UPM)
M	Female	PhD	Human Computer Interaction	7	Universiti Putra Malaysia (UPM)
N	Male	PhD	Software Development	10	Universiti Putra Malaysia (UPM)
O	Female	PhD	Software Engineering, Software Critics	16	Universiti Putra Malaysia (UPM)
P	Female	PhD	Software Engineering	30	Universiti Teknologi Mara (UiTM) Perlis

Table 5.3 exhibits 16 experts with different expertise from academic institutions in Malaysia. This activity is important to establish reviews and comments in this study.

As for the educational background, they are Ph.D. holders. The reviews and comments of these experts from related fields are important in contributing the improvement of AVOMP design guidelines.

5.3.1 Procedure

The review is conducted through email communication. Firstly, the invitation email is sent to the identified experts (see Appendix A). If the expert agreed to be a reviewer, then an official appointment letter from UUM (see Appendix B) and the reviewer instrument (see Appendix E) are sent to them. The opportunity and an ample time are given to them to review the AVOMP design guidelines and fill the questionnaire. On average, the instrument was given back within two or three weeks. The next subsection discusses the findings of the review.

5.3.2 Expert Review Finding

The data were listed in frequency and tabulated in Table 5.4 based on the questions asked in the instrument. It is also plotted in the clustered column charts (Figure 5.2) which provide a straightforward and valuable way to illustrate the different frequency of responses.

Table 5.4

Responses Frequency from Expert Review

Items	Frequency (n = 15)		
	Needs very detail explanations	**Need some explanations**	**Is easy to understand**
Q1: The proposed guidelines are relevant			
Guideline 1		5	11
Guideline 2		1	15
Guideline 3		1	15
Guideline 4	1	2	13
Guideline 5	1	2	13
Guideline 6	1	4	11
Guideline 7	2	1	13
Guideline 8		4	12
Guideline 9	1	1	14
Guideline 10		4	12
Guideline 11	1	2	13
Guideline 12	1	3	12
Guideline 13		3	11
Guideline 14	1	4	11
Guideline 15		5	11
Guideline 16	1	5	10
Guideline 17	1	3	12

Q2: The importance level of guidelines (Important (1) to Very Important (5))

Likert Scale 5	**1**	**2**	**3**	**4**	**5**
Guideline 1			1	4	11
Guideline 2		1	1	5	9
Guideline 3			1	5	10
Guideline 4			1	6	9
Guideline 5			2	3	11
Guideline 6			1	4	11
Guideline 7			1	5	10
Guideline 8		1	2	8	5
Guideline 9	1			6	9
Guideline 10			1	5	10
Guideline 11			2	6	8
Guideline 12			1	9	6
Guideline 13			1	7	8
Guideline 14			1	7	8

Guideline 15			1	4	11
Guideline 16			1	9	6
Guideline 17			1	6	9

Q3: The language and terms of guidelines are easy to understand (Strongly disagree (1) to Strongly Agree (5))

	1	**2**	**3**	**4**	**5**
Guideline 1		1	3	4	8
Guideline 2			2	4	10
Guideline 3			1	5	10
Guideline 4			1	4	11
Guideline 5			3	4	9
Guideline 6			2	6	8
Guideline 7			2	4	10
Guideline 8			1	8	7
Guideline 9		1	1	4	10
Guideline 10			1	5	10
Guideline 11				8	8
Guideline 12			2	5	9
Guideline 13		1	1	6	8
Guideline 14			2	8	6
Guideline 15			1	6	9
Guideline 16			1	9	6
Guideline 17			1	5	10
Yes/No Question				**Yes**	**No**
Q4: The design guidelines for representing the problem are relevant				16	0
Q5: The design guidelines for representing the solution are relevant				16	0
Q6: The connections and flows of all the guidelines are logical				15	1
Q7: The design guidelines are applicable to the development of algorithm visualization on mobile platform				16	0
Q8: Overall, the design guidelines are readable				15	1
Q9: Overall, the design guidelines are applicable				15	1

Note: Q = Question

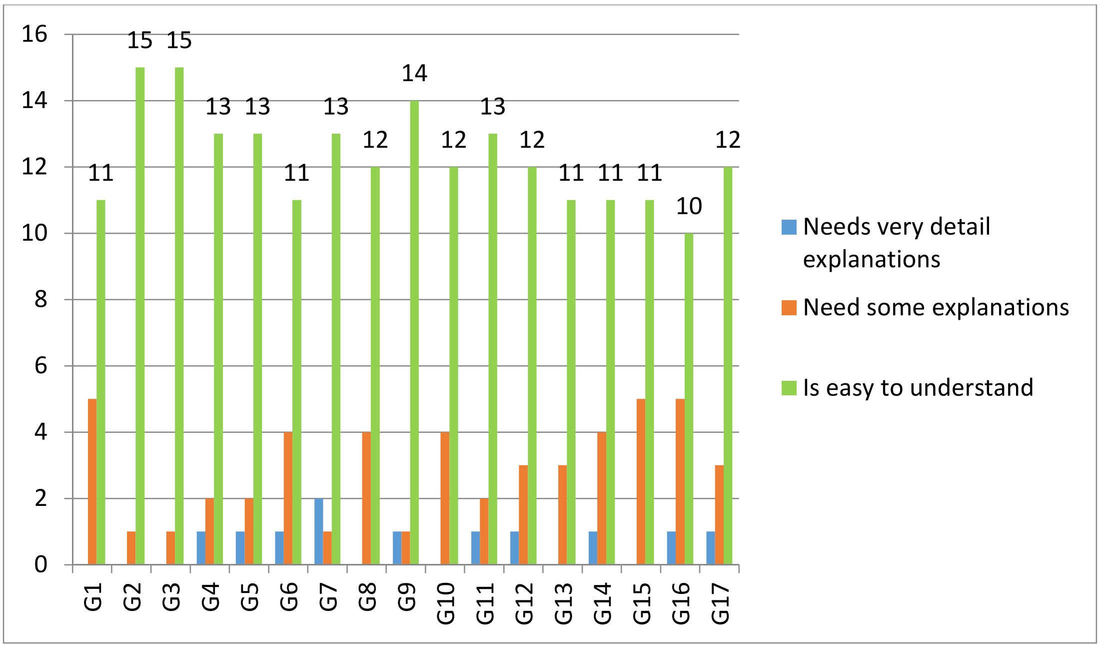


Figure 5.2. Relevancy of the proposed design guidelines of AVOMP

Meanwhile to measure the importance level and easy to understand level, the mean of each guideline is calculated using the formula by (Hujainah et al., 2016), as follows:

M = Σ(RPi* ΣERi) /N
where,

i= is the rate

RP is the Rating Point

ER is the number of the experts who select the rate

N: is the total number of the expert who participated in the evaluation process which is 16 experts.

The mean values for all guidelines are represented in Table 5.5. And Table 5.6.

Table 5.5

The Mean Value of Importance Level

Guidelines	Mean Value
Guideline 1	4.62
Guideline 2	4.37
Guideline 3	4.56
Guideline 4	4.5
Guideline 5	4.56
Guideline 6	4.56
Guideline 7	4.56
Guideline 8	4.06
Guideline 9	4.3
Guideline 10	4.56
Guideline 11	4.37
Guideline 12	4.31
Guideline 13	4.5
Guideline 14	4.5
Guideline 15	4.62
Guideline 16	4.31
Guideline 17	4.87

Table 5.6

The Mean Value of Easy to Understand Level (Term and Language)

Guidelines	Mean Value
Guideline 1	4.18
Guideline 2	4.5
Guideline 3	4.56
Guideline 4	4.62
Guideline 5	4.37
Guideline 6	4.37
Guideline 7	4.62
Guideline 8	4.3
Guideline 9	4.43
Guideline 10	4.56
Guideline 11	4.5
Guideline 12	4.43
Guideline 13	4.41
Guideline 14	4.31
Guideline 15	4.5
Guideline 16	4.31
Guideline 17	4.56

As exhibited in Figure 5.2, majority of the experts agreed that the proposed guidelines are relevant. Also, as depicted at Table 5.5, majority of the experts assessed 4 points and above out of 5 in each guideline in terms of its importance level. The same goes for the mean value of easy to understand level (see Table 5.6), most of experts agreed towards the term and language used in each guideline. In addition, further comments are also provided from all of the experts in this study as presented in Table 5.7. In conveying the clearer meaning, some of the comments were rephrased from the original versions.

Table 5.7

Responses Frequency from Expert Review

Experts	**Comments**
Expert A	1) You should consider on pleasurability aspect. Since mobile apps are interactive learning medium so it is important for the researcher to take this advantage by providing the apps that look different from the text book. You can consider Multimedia Learning principle as the basis and how to make it suitable for students/teenagers 2) G4, G5, G6 could be designed to be more attractive to the user
Expert B	1) G12 and G14 – The description of this guideline is quite unclear. 2) Having the illustrations is very good since they help the evaluator to understand some unclear description of the implementation.
Expert C	1) G6: Implementation (4 Please state your reference (citation), more than 5±2 different colors (can't understand this statement), or maybe not more than 3 colors 2) Set the size of buttons and font size in terms of pixels, a detailed guide for other developers in developing the apps
Expert D	1) Cuing should be specified clearly by stating the types of cuing for example visual/audio/other suitable types of cuing 2) Hierarchy design need to be specified in terms of its structure (refer navigational structure) 3) Constructing activity is not clear in terms of its application. Maybe you can use a different term. 4) must have a clear differentiation between feedback and response in G15 5) Good and applicable guideline to assist the developer (with some knowledge on UI design and algorithm).

Expert E	1) You can enhance the design after evaluating the prototype that has been developed
Expert F	1) There is no guideline regarding providing Help 2) G5: Provide simple hierarchical navigation system Provide straightforward hierarchical navigation system
Expert G	1) When user navigate, it will trigger sound when reaching the end of navigation to alert user the navigation is end 2) It also can trigger a sound when an error occurs. 3) So far, all 17 guidelines provided are suitable for the apps 4) Guideline G1 need to give priority especially when put information regarding source code. Usually, source code needs to be pair with the output or how the code work renders more understanding to the user. So, during source code explanation, avoid too much content. Description can be in a popup agent might help only for user who need more explanation
Expert H	1) You can provide any guidelines for information in multiple pages.
Expert I	1) Need to explain some terms for example "interesting events" (G1) 2) Need to suggest the best or chosen size (g2) 3) Need to describe layout clear? Layout simple? Layout obvious? (G7) 4) Confusing in implementation explanation (G9) 5) Explain more about back movement? How many back steps allowed? What is the difference between >> and > for example (G10) 6) Explain implementation steps. Not detail enough (G11) 7) How the answer should look like; how many attempts (G15) 8) Explain more on implementation steps (G16 & G17) 9) Combination of styles of the text /content does affect the visualization perception as well. Do consider them also. Example, (bold/italics or Highlight/color)
Expert J	1) G13: Explain further the task in this guideline 2) G14: Explain the function of drop-down button
Expert K	1) G6 - Describe good practice of color combination. 2) G9- The 'About Us' could briefly introduce on how the learner is going to learn algorithm in the app. 3) G14 – Provide examples of this exercise activity. 4) G15 – Provide feedback examples. 5) Overall, the guidelines provide a complete, theory supported guidance for learning algorithm visualization on mobile platform. Excellent work.
Expert L	1) It is better if all figures provided as illustration are printed in color for easy understanding. 2) No guidelines seem to be duplicated or in conflict with each other. All have clear explanation and implementation. Perhaps

	you can add more examples to illustrate the implementation.
Expert M	1) Guideline 9 and guideline 16 feels somewhat similar. A more elaborate description may be needed as to not confuse the two. 2) It might be helpful to provide footnotes or links to enable layman users to understand better some of the terminologies. For instance, due to my HCI backgrounds, guideline 1 – 9 appear pretty straightforward to me, but I struggle to understand guidelines 11 and 14. But someone with an education background may find guideline 14 to be clear. 3) the guidelines can be arranged in order of importance and each of them with their respective % of importance 4) Overall, I find the guidelines to be useful
Expert N	1) There are lots of theory of design such as page layout design and CASPER that can be embedded in this mobile design guideline 2) It seems number 9 and 16 is repeatable which can be combined as controlled setting 3) Size is also important which I cannot see in your guideline. 4) Refer to CASPER Principle to get more information. 5) Overall the guidelines are acceptable but are still unclear to see the flow and connection of all guidelines. I recommend you make a big picture that represents the flow and connection of all guidelines.
Expert O	1) In G8 Description: Make the related element close to each other Can be revised as: Group similar elements" 2) In G11 Description, can be revised as "Show dynamic explanation during viewing activity" 3) In G13 Description, can be revised as "Allow modification of input data"
Expert P	1) It is better if the EXPLANATION is divided into 2 parts REASONS: with the theories and concepts OBJECTIVES: what are achieves when the theories are apple. All guidelines G1 to G17 have the objectives but are written as Explanations. IMPLEMENTATION: As it written in the guides 2) Example <u>REASONS</u> Segmentation is useful as it …. <u>OBJECTIVES</u> Segmentation allows user to pause…. <u>IMPLEMENTATION</u> Use of line…

5.3.3 Justification on Experts' Comments

Expert A suggests putting pleasurability aspect in guidelines. Expert A further explains as for pleasurability aspect is to make the G4, G5, and G6 be more attractive to users. This suggestion is important, as it gives enjoyment, pleasure and positive feedback towards users. Expert B comments to give a clearer description on G12 and G14. Expert C suggests giving more detail information about G6, regarding the statement “use more than 5±2 different colors”. Another suggestion is to detail out the size of buttons and text in pixel.

Regarding the clarity, this study also admits the suggestion from Expert D, which is to specify clearly types of cuing guidelines (example), the hierarchy navigational structure, constructing activity in terms of its application and the difference between feedback and response in G15. This is answered by Expert F in terms of hierarchal navigation, which is to provide it simple and straightforward. Another from Expert F is to provide help guidelines.

Expert G suggests to giving trigger sound at the end of navigation button to alert the user. This sound can be also used to give warning when errors occurred. Also, Expert G considers that G1 need to be focused on simple information and provide such a popup agent (just used when needed) that can give more details of the codes (another way to avoid displaying much content). This popup agent could be one of the solutions answering the comments of Expert H that suggest “You can provide any guidelines for information in multiple pages”. The use of tab or swipe features can also be used to settle information in multiple pages.

Other comments comes from Expert I that this study admits to take into account regarding clarity for AVOMP design guidelines, which suggests to explain more the sample of interesting events (G1), the size (G2), layout (G7), implementation explanation (G9), explanation about backward buttons and the different with other buttons (G10), implementation steps (G11), the answer and attempts (G15), implementation steps (G16 & G17) and combination of styles of the text /content that does affect the visualization perception as well, such as (bold/italics or Highlight/color). Other clarity matters are asked by Expert J to explain further the task for G13 and also explain the function of a drop-down button for G14. Expert L gives suggestions to put the good practice of color combination (G6), give an introduction for the learner to use the app (G9), provide an example of exercise activity (G14) and provide feedback examples (G15). Meanwhile, Expert L does not have much suggestion, unless to provide more illustration and make an illustration in color for easy understanding.

Expert M and Expert N suggest combining guideline 9 and 16 in which they found that these two guidelines are somewhat similar. After rechecking and comparing these two guidelines, this study agrees that this review is correct and thus agrees to combine these two guidelines becoming guideline 9. Additionally, Expert M suggests providing footnotes for some terminologies in the guidelines. She also comments to arrange the guidelines in order of importance and each of them with their respective percentage of importance. This study finds the comment of putting importance percentage for each guideline is helpful for designers. However, this study could not take another suggestion to put the guidelines in order of its importance percentage, since it will be confusing for designers to see the flow and connection of the guidelines.

Expert N comments to see CASPER principle for more design theories to be embedded in the guidelines. Expert also added to consider size matter in the guidelines. The last suggestion from Expert N is that it is important to draw the big picture that represents the flow and connections of all guidelines. This study admits that this comment is helpful for this study to make the proposed design guidelines are more understandable for designers. Meanwhile, Expert O comments more on the language of guidelines, to simplify the sentences for G8, so instead of “Make the related element close each other”, can be revised to “Group similar elements”. G11 is suggested to become “Show dynamic explanation during viewing activity”. Lastly, G13 can be revised as “Allow modification of input data”.

The last expert, Expert P suggests dividing explanation part into two parts, which are reasons and objectives. Reasons consist of theories and concepts, while objectives consist of what are achieves when the theories are apple. This study views this comment reasonable. This division will help a designer see clearer into the guidelines.

5.3.4 Reviewed Design Guidelines of AVOMP

In efforts to provide a better impression and enhancing the readability, the Design Guidelines of AVOMP has been revised and redesigned based on the comments from the experts as illustrated in Tables the next sub-sections.

5.3.4.1 Design Guidelines of AVOMP in Perspective of UI Design Aspect

The table 5.8 below explains in detail about each guideline of UI design aspect. The “G” initial in table 5.8 stands for a guideline.

Table 5.8

UI Design Aspect

Category 1: Content

G1

Recommendation: Avoid much content on small screen

Basis: Coherence principle of multimedia learning by Mayer (2002)

Explanation: This is advocated from coherence principle by Mayer (2002) that students will learn better when the extraneous materials are removed, thus minimizing cognitive overload. Removing the extraneous content in reducing cognitive load has been the primary recommendation from numerous studies of multimedia learning (Ayres & Paas, 2012; Fiorella & Mayer, 2014; Mayer, 2009; Sung & Mayer, 2013).

Implementation:

1. In AV system itself, the content, such as information, guidance, warning, etc., can be summarized by putting a simpler and direct sentence.
2. This includes abolishing of irrelevant background music (Brunken et al., 2003).
3. Avoid background decorations (Fiorella & Mayer, 2014).
4. Use popup agent (a button) to provide more information to users or use tab or swipe feature to display long information (multiple pages).
5. Use Interesting Event approach that can summarize the focal point in pseudocode (see Figure A).
6. Avoid adding spoken sound towards textual information since it will create redundancy effect (Mayer & Moreno, 2003; Sweller, 2011).

Illustration:

```
void bubblesort(Comparable[] A) {
  for (int i=0; i<A.length-1; i++)
    for (int j=1; j<A.length-i; j++)
      if (A[j-1].compareTo(A[j]) > 0)
        swap(A, j-1, j);
}
```

Figure A. Interesting Event Example

***Note**: The figure above shows an example of an interesting event for swapping process. The swapping code is usually presented with 3 lines for the full code but the interesting event is useful to summarize it into one-line code to represent the whole process.

G2

Recommendation: Avoid Scrolling

Basis: Split-attention effect in multimedia learning

Explanation: The scrolling while reading learning material could create split attention effect and is strongly suggested to be avoided (Churchill, 2011; Luong & McLaughlin, 2009). A study from Sanchez and Wiley (2009) found that this happens due to the eye needs to allocate the focus many times while the text moves up and down. Sanchez and Goolsbee (2010) also confirmed through their experimental study that scrolling gives negative effects on students' text comprehension, especially using mobile phone with a small screen. This is consistently in line with the study of Sanchez and Branaghan (2011) stated that scrolling is not appropriate for mobile learning because it will affect the recall issue during the reading text.

Implementation:

1. Do not make the design of mobile app exceed the size of mobile screen
2. Choose important UI elements only (text, image, button, etc.)
3. Set properly the size of AV system elements (pseudocode text size, visualization size, buttons, etc.).
4. Make UI elements (pseudocode text size, visualization size, and buttons) to be responsive that can suit and adapt the various size of the screen. This way will prevent from extending the learning material that does not exceed the size of the screen; otherwise, it will force to have scrolling within an app (Sanchez & Goolsbee, 2010).

G3

Recommendation: Use cueing to emphasize the essential information

Basis: Multimedia learning

Explanation: Cueing or also known as signaling has an advantage in terms of decreasing the split-attention effect (Osman & Elmusharaf, 2014; Shitkova et al., 2015). This is because the cuing gives direct focus to a focal point of essential information and also is used to make a connection towards a corresponding animation or image (Lazaridis et al., 2013; Liu, Lin, et al., 2013). The studies (Liu et al., 2012) and (Liu, Lin, et al., 2013) found that cuing in form of arrow line has significantly improved the students learning on science lesson through a mobile application.

Implementation:

1. Highlight the pseudocode line by line during the animation of a particular algorithm (see Figure B).
2. Give cues in visualization, such as arrow signs in showing swapping process, showing the movement of values (see Figure C).
3. Cuing in form of sound can be also used whenever the user navigates at the end of visualization, the sound will be triggered, informing the user.

4. The sound cuing can also be used to give warning to user whenever errors occurred

Illustration:

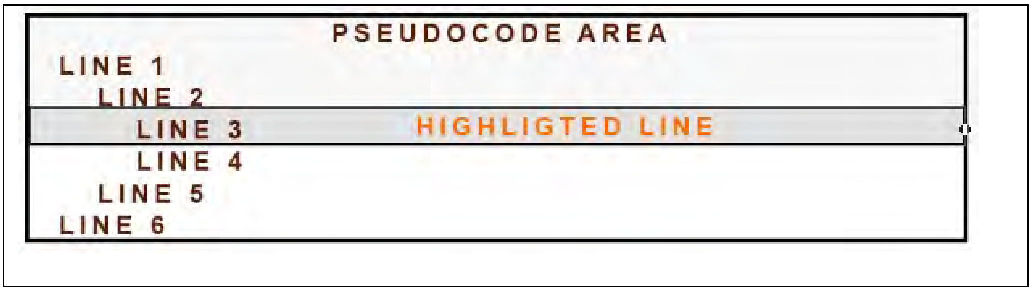


Figure B. Highlighted line of pseudocode during step-by-step animation

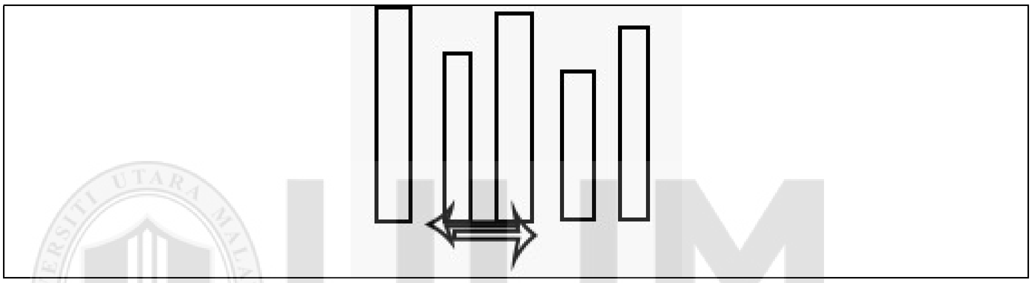


Figure C. Two arrow lines animation show the swapping process

*Note: this guideline number 3 is related to guideline number 9, which is to make cuing (such as arrow animation or highlighted line animation) during viewing activity.

G4

Recommendation: Use multiple views (segmentation).

Basis: Cognitive load theory

Explanation:

Segmentation is used where the information is divided into pieces of meaningful information, reducing the cognitive overload since it will signal the learners the organization of the information (Florax & Ploetzner, 2010; Sung & Mayer, 2013). This segmentation is mainly used to divide algorithm visualization per line of pseudo code, so that each animation representing each line of algorithm codes. This will contribute students understanding, information accessibility, and pleasurability aspects (Aziz, Mutalib, & Sarif, 2017). The animations also need to be brief and stimulate the user to understand the purpose to be delivered.

Moreover, segmentation reduces the overwhelming of working memory as it gives pauses effect between segments of dynamic information (such as animation), which improves learning comprehension on learning material (Karam, 2015). Thus, it is also suggested to facilitate the learners with a control panel for segmenting the continuous material in minimizing superfluous load (Ayres & Paas, 2012; Wong et al., 2012). This way will allow students to pause and catch up with the reading material before they can proceed to the next step of new segmented material (Spanjers et al., 2010).

Implementation:

1. Use line/box/white space as a gap that separates between one element and others (see Figure D).
2. Divide the animation into pieces of information, processed sequentially based on line by line of an algorithm pseudocode with step-by-step navigation buttons (next and backward), rather than using play button. This is related to G9 which is to make segmentation of the visualization in viewing activity.

Illustration:

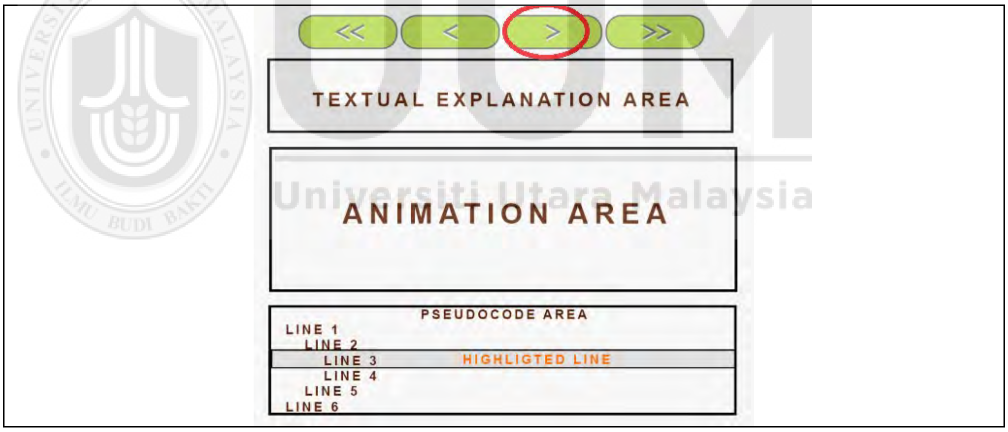


Figure D. The segmentation in animation and area

The figure D above shows that the animation area will be shown segmentally based on the step-by-step process in pseudocode as the student clicks the next arrow button (related to G9 until G11). The figure D also shows that each different area is segmented inside the box (textual explanation area, animation area, and pseudocode area) in order to distinguish between one and the others.

Category 2: Navigation Design

G5

Recommendation: Provide simple and straightforward hierarchical navigation system

Basis: Cognitive load theory, Usability principle

Explanation:

Navigation has to be simple and clear in terms of information and content. The navigation indicates a cue, helping the learner see the connection between elements (Liu, Lin, et al., 2013). If the learner sees the navigation has ambiguous meaning that leads to the mismatched expectation of user's action then it will lead to the decreased performance level (Ross & Gao, 2016). The clear navigation is also helpful for users to keep track of the location in the system and enhance the user pleasurability experiences (Nurulnadwan, Ariffin, & Mahfuzah, 2015). Consistency is another important factor of hierarchical design. This involves the consistency of UI design elements' position and shape in multiple pages. Hujainah et al. (2016); (Shitkova et al., 2015) added that design layout (labels, styles, appearances, and colors) also requires its consistency across multiple platforms and various mobile devices.

Implementation:

1. Make the cue (as a breadcrumb or header) on the page that indicates the connection after clicking the main menu buttons, telling users their location within an app (see Figure E).
2. The simple design of navigation does not have additional icon or image.
3. Have consistency in terms of the design style of AV navigation buttons as well as its position throughout the entire pages of the app.

Illustration:

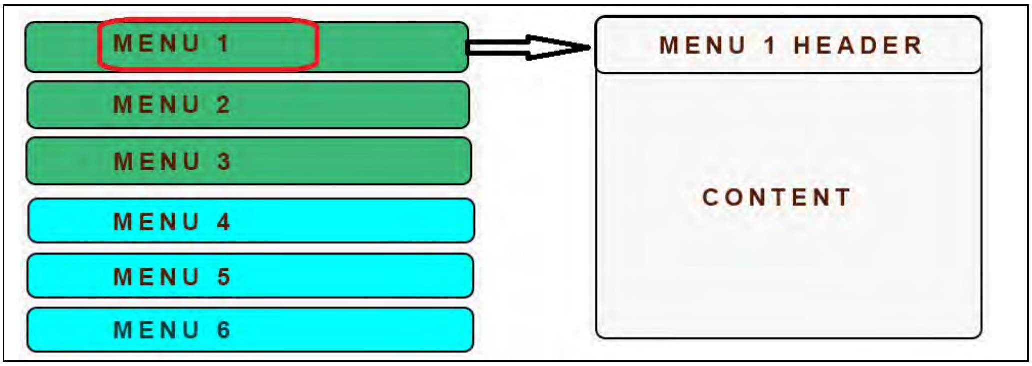


Figure E. Menu 1 header is used to inform a user after clicking button

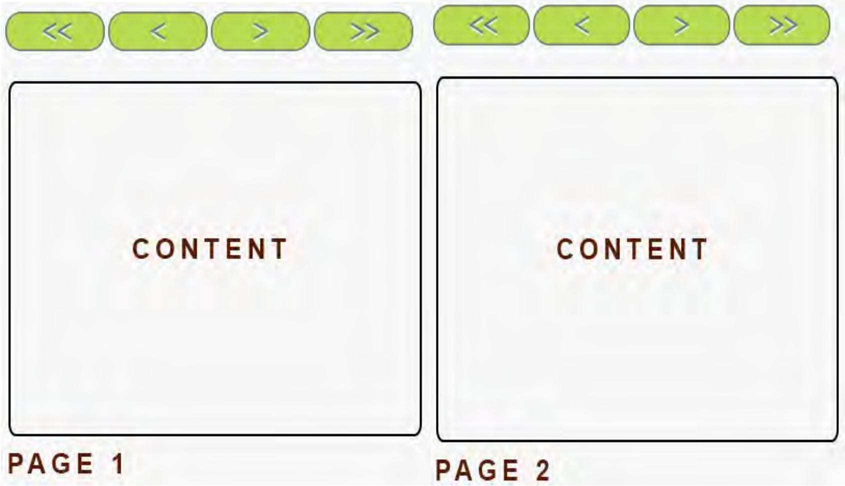


Figure F. The consistency of navigation buttons' position in many pages

Figure F shows the consistency in terms of buttons' position. Additionally, buttons have to fit fingertip size and are separated by a gap of white space in decreasing the miss-click error rates.

Category 3: Color

G6

Recommendation: Use appropriate color

Basis: Visual cuing of cognitive load theory, color theory, usability of mobile application

Explanation: The appropriate use of color could decrease the cognitive load, for instance, the text should have sufficient contrast with the color of the background to meet the readability requirement (Richardson et al., 2014). Color could be also used as cuing element for highlighting the essential information, decreasing the split attention effect (Lazaridis et al., 2013; Mirkovic et al., 2011) and giving guidance to users (Brandse & Tomimatsu, 2014). Color could assist in terms of grouping information, focus on particular content, make differentiation, and establish relationships between elements, as well as giving pleasurability aspect (Hoehle et al., 2016).

To show the difference between the two objects, the software developer should choose the contrasting colors (dark color with bright colors). This way is called a contrast which is one of the most effective ways to add attraction or attention to a design (Shneiderman, 2010).

Implementation:

1. Use color to group bunch of similar information (see Figure G).
2. Use color in visualization to highlight the dynamic movement of algorithm processes (see Figure H). (This related to G9)
3. Use color to highlight the line (as a current activity) in pseudocode during visualization (see Figure H). (This related to G9)
4. Avoid utilizing more than 5±2 different colors (especially in pseudocode) because it can overwhelm students' recall in short-term memory (Ros and Gao, 2016)

Illustration:

Figure G. The use of color to group the similar information in buttons

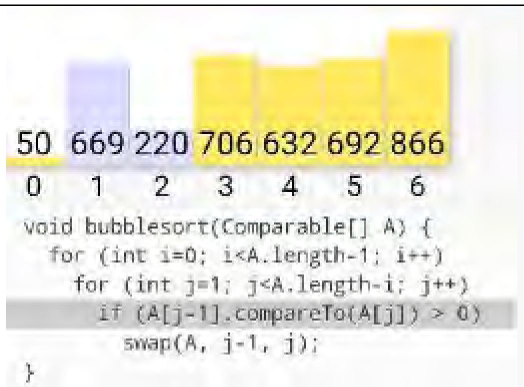


Figure H. The use of color to highlight the dynamic movement in animation and highlight the current line in pseudocode

Figure H is the example of bubble sorting algorithm that shows the use of color to highlight the processed element in both animation and pseudocode parts.

Category 4: Layout Design

Recommendation: Make layout clear, simple and obvious

Basis: Usability of mobile application, Gestalt theory

Explanation: The layout needs to be simple (Wilson et al., 2004), clear, uniform, and consistent (Nathan & Yeow, 2011; Zollet & Back, 2010). It is also recommended to make information and content organized based on gestalt principle. The gestalt theory of similarity principle, for instance, suggests that the elements that have similarity in characteristic (shape/type/form/style/color) can be grouped in order to make the design clear, obvious, and easy to be recognized (Hoehle et al., 2016). This will make learners easy to grasp the content organization and thus affect their learning experiences.

In addition, In the alignment principle, each item to be placed in the software interface or layout should be balanced with the software interface to look attractive (Shneiderman et al., 2010). Software developers should be good at composing items they want to present so they look balanced. In other words, in the principle of Alignment, items need to be structured so that it is not awkward. The items also need to have visually related to each other on every display of the app.

G7 **Implementation**:

1. Make the design of UI elements uniform and consistent in terms of color, style, shape, and position.
2. Use the similar shape, color and look for navigation buttons in the system. (see Figure J)
3. Use the consistent layout and background for each page in the app.
4. Make each item in the layout aligned to make them visually connected and balance (see Figure K).

Illustration:

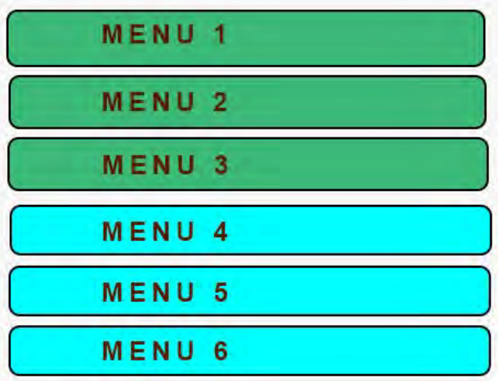


Figure I. The similarity in terms of shape, look and color for menu buttons

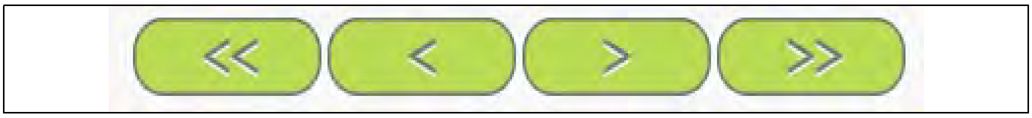


Figure J. The similarity of control buttons of algorithm visualization

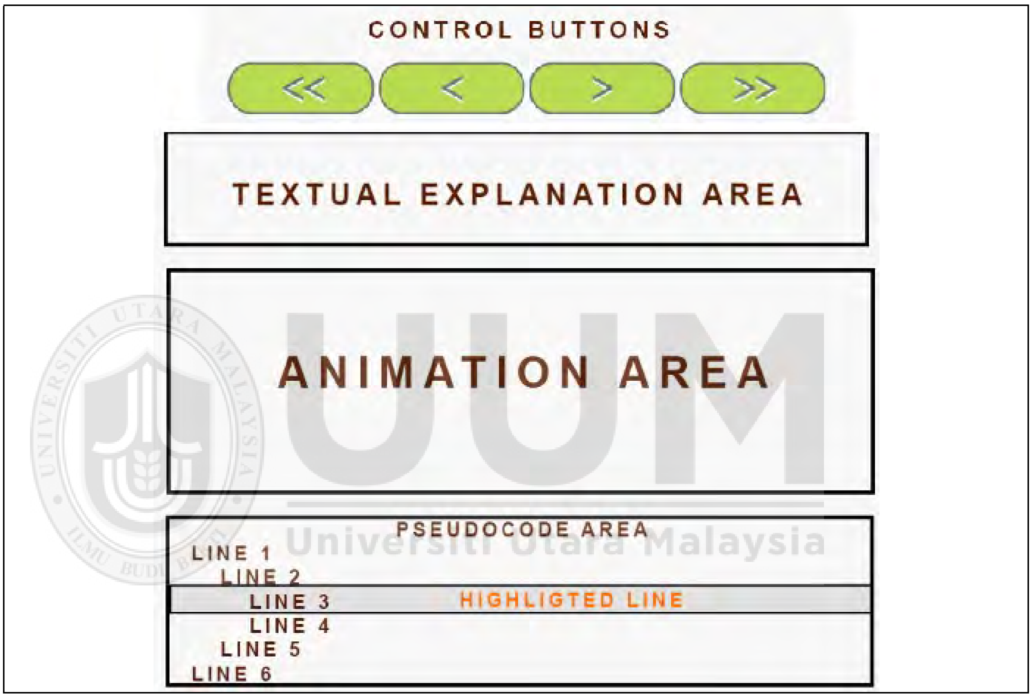


Figure K. The alignment of items or parts in AV layout should be balance.

There will be many algorithms to be visualized in the system. Therefore, it is important to be consistent to use the same style or model for UI elements for instance control buttons for each page of algorithms, otherwise, the students will need more time to comprehend the learning material.

G8

Recommendation: Group similar elements

Basis: Multimedia Learning: Spatial contiguity principle, Gestalt theory

Explanation: Spatial contiguity principle states that whenever possible it is strongly suggested to make two related elements (e.g. image and text) near, reducing split attention effect of learners (Mayer, 2009). Additionally, proximity principle of Gestalt theory also suggests making the related element to close each other, since it will be perceived by the learner as one group, and as a result, will be examined or read together (Graham, 2008; Guberman et al., 2012).

The collection of items makes the user feel comfortable because the related items are seen as a group and are not separate (Shneiderman et al., 2016). Therefore, this element will be a visual unit rather than several separate units.

Implementation:

Make the same elements proximate to each other, such as buttons, visualization, and its pseudo code (see Figure L).

Illustration:

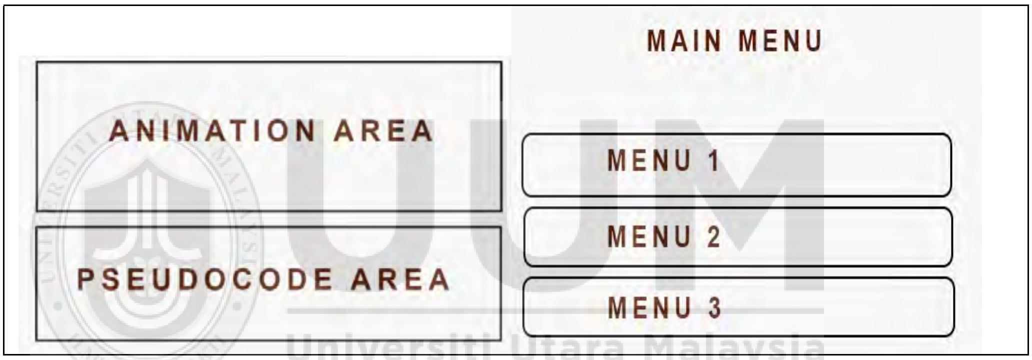


Figure L. Proximity of related elements.

5.3.4.2 Design Guidelines of AVOMP in Perspective of Interactivity Aspect

Table 5.9 Interactivity Aspect

Category 1: Viewing (First engagement taxonomy level)	
G9	**Recommendation**: Displaying step-by-step execution of an algorithm using control buttons (viewing activity) **Basis**: Cognitive load theory (CLT), Engagement Taxonomy (ET), Extended Engagement Taxonomy (EET), Usability of mobile learning **Explanation and Implementation**: As the core engagement taxonomy, viewing means the learner can view the algorithm animation based on pseudo code that is traced line by line sequentially by pressing control buttons. This means the concept of an algorithm is divided into meaningful chunks (refer to G4). This will minimize the cognitive load of students during viewing activity; instead of showing all the codes altogether which will overwhelm the student's working memory. Originally viewing term is from ET, listed by Naps et al. (2002), but then has been revised by Myller et al. (2009) to have another category, which is called controlled viewing. The difference is viewing category allows the learner to watch the visualization without any setting, whereas the controlled viewing provides control setting for user so that the learners can control the environment of visualization (how the visualization is displayed), either changing the animation speed or visual elements during or before the viewing (Myller et al., 2009; Sorva et al., 2013). This control setting suggests that learners are supposed to have the freedom to change the environment of the application, making them feel that they can control the application. This way will enhance the satisfaction of the learner itself since they feel that they have the flexibility to change the setting anytime they want instead of required to reply the action (Hujainah et al., 2016). This is also proved and highly recommended by many studies (Savoji et al., 2011; Schmidt-Weigand et al., 2010; Schüler et al., 2013; Spanjers et al., 2012; Sung & Mayer, 2013) to provide this learning-pace setting for users due it can decrease the cognitive load.

Implementation:

1. Provide navigation/control buttons for controlling the visualization (in animation area) that will be dynamically animated sequentially as the student clicks on it.
2. Provide the control panel that entails some settings, such as animation speed of AV and visualization style (bar or array for example) (see Figure A).

Illustration:

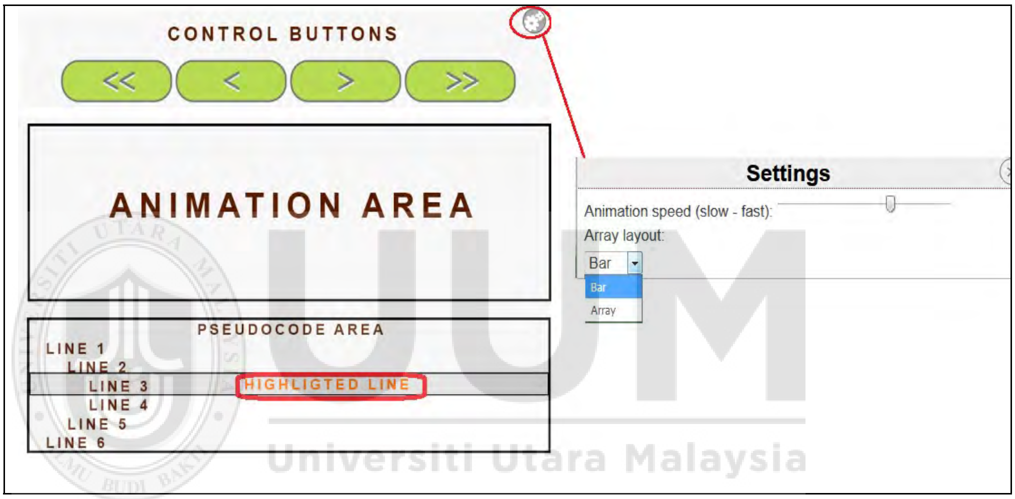


Figure A. The controlled viewing for algorithm visualization

*__Note__:

1. The visualization (in animation area) of an algorithm is processed according to tracing line which moves downward as a user presses the next ">" button.
2. The end ">>" button enables a student to navigate directly to the end of the animation. This is useful for an advanced user to skip the steps ahead.
3. The first-button "<<" button enables a student to navigate back directly to the first step of the animation. This is useful for a user to go back directly to the first animation instead of clicking many times using back-button.

G10

Recommendation: Provide rewinding button

Basis: Extended Engagement Taxonomy (EET)

Explanation: This suggests AV should provide a backward option for viewing activity. Some AV studies (Boticki et al., 2013; Osman & Elmusharaf, 2014; Simonák, 2013; Vrachnos & Jimoyiannis, 2014) do not provide this option on their navigation buttons; instead, they just provide next and play button only. In fact, this option will be used whenever students feel lost track of the visualization and are still confused (Juett, 2016; Karavirta & Shaffer, 2015). It is also recommended to put a notification in form of the page number of visualization (see figure B) so that learners will be notified with the status of the current page as they move forward or backward (Karavirta & Shaffer, 2015).

Implementation:

1. Provide rewinding button in visualization
2. Put the number of current steps and its total number of executions.

Illustration:

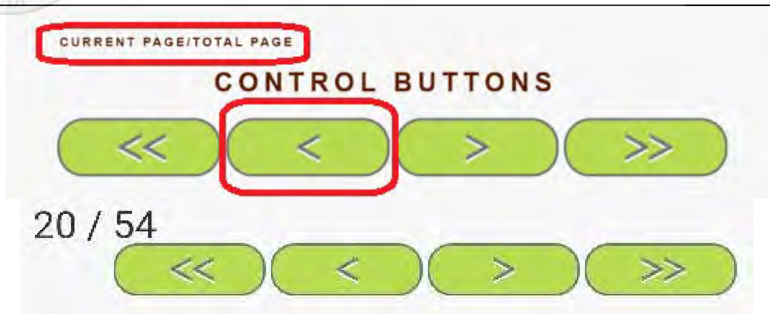


Figure B. Backward control button of algorithm

***Note**:

<u>Keep in mind that backward “<” button is different from first-button “<<”. The backward button allows the user to navigate step by step visualization in a reverse way, whereas first “<<” button, as mentioned before, is to make users jump backward directly to the first step of visualization</u>.

G11	**Recommendation**: Show dynamic explanation during visualization (pseudocode and textual explanation) **Basis**: Engagement Taxonomy (ET), Extended Engagement Taxonomy (EET), Temporal contiguity principle of multimedia learning **Explanation**: During viewing activity, it is recommended to also show the dynamic pseudocode and textual information that is traced line by line corresponded with its visualization (Meolic & Dogša, 2014; Osman & Elmusharaf, 2014; Simonák, 2013; Vrachnos & Jimoyiannis, 2014). This is supported by temporal contiguity principle of multimedia learning that recommends presenting related elements simultaneously rather than consecutively in order to minimize cognitive load (Mayer, 2009). In addition, Urquiza-Fuentes and Velázquez-Iturbide (2007) suggested: that the utilization of narratives and textual contents incorporated with visualization could increase the effectiveness of AV. This is in line with a study by Urquiza-Fuentes and Velázquez-Iturbide (2009a) that reported involving dynamic textual content during visualization have improved students' knowledge acquisition. In the principle of repetition there is a standard interface of the software so that users will be able to interpret and understand easily the instructions provided by the software (Shneiderman et al., 2016). Repetition using different media presentations will help users understand. Therefore, even though, the user is displayed with pseudo code to understand each animation, the repetition media through textual explanation text would be useful to understand animation easily (Karavirta & Shaffer, 2015).

Implementation:

1. Make a highlighted line of pseudocode dynamically changed as the user proceeds to the next step of visualization (see Figure C).
2. Provide textual explanation which is dynamically changed during visualization, informing the process of each step of pseudocode (see Figure C).

Illustration:

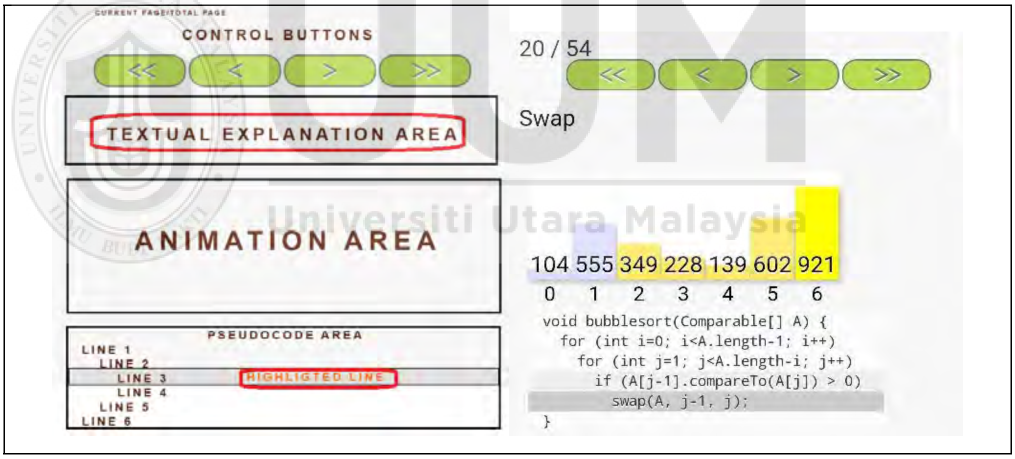


Figure C. The Textual description changes during viewing activity

***Note**: As in the figure above textual explanation is updated automatically when the user traces swap line.

Category 2: Responding

G12 **Recommendation**: Provide relevant questions to be answered by students

Basis: Engagement Taxonomy (ET)

Explanation: Responding is a technique of ET that allows system to display a pop-up question during viewing animation in which it will trigger user to predict the next move or visualization to be displayed (Juett, 2016). This will enhance the experience of student to think a step ahead due to they are encouraged to analyse the next move in visualization (Saltan, 2016). Responding is from ET suggests that AV system should provide relevant questions to be responded regarding a certain algorithm in the system, either during viewing activity or after it (Nikander et al., 2004; Vrachnos & Jimoyiannis, 2014).

Implementation: (**choose one of ways**)

1. Provide the quiz form with multiple choices; the questions are related to a particular algorithm concept. (* this quiz can be displayed after viewing activity) (see Figure D).
2. Make Pop-up question during viewing activity, prompting the learner to guess or predict the next move of visualization.

Illustration:

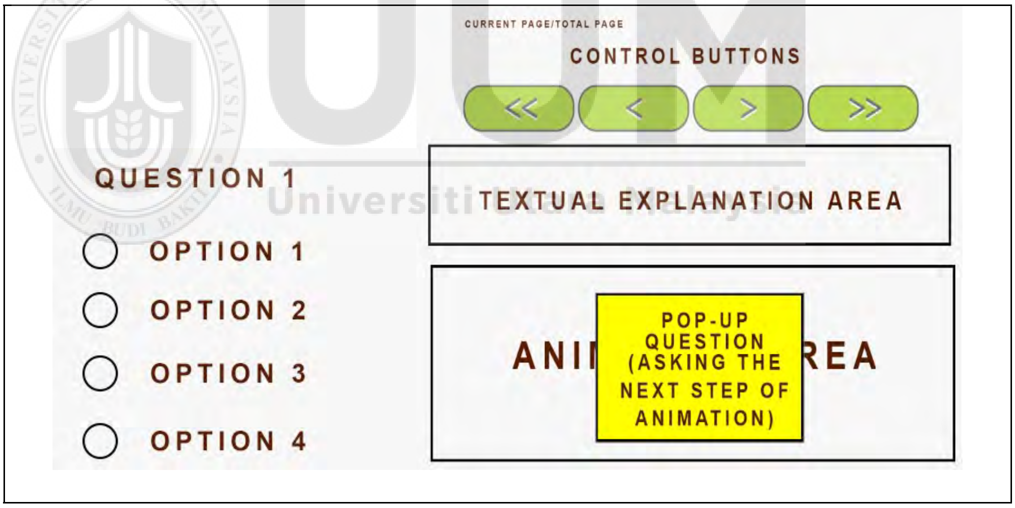


Figure D. Multiple choices and pop-up questions.

Category 3: Changing

G13

Recommendation: Provide modifying input data for algorithm and also can generate the random values

Basis: Engagement Taxonomy (ET), Two-Dimensional Engagement Taxonomy (2DET)

Explanation: Changing is included in ET category which means the learner can change the value of arrays or variables in order to see the different behavior of particular algorithm visualization based on that numbers they set (Jonathan et al., 2016; Naps et al., 2002). This means the user can set certain values before or during doing viewing activity.

In this case, Sorva et al. (2013) listed new terms in the Two-Dimensional Engagement Taxonomy (2DET) that divides the input of changing into two types which are given content and own cases. Given content means the student can learn the AV through predefined/given value (random values from the system). On the other hand, the own cases mean the students learn the AV by inserting the inputs of variables themselves (Karavirta & Shaffer, 2015).

Implementation:

1. Provide a feature that can generate the random values of arrays (Given content by AV system). For instance, the button "run" in Figure E enables the user to generate random numbers by default to be visualized.
2. Provide a feature that allows users to insert or modify their own values of arrays. For example, a text box (in input area) in figure E is provided for users to insert their own values of arrays.
3. Provide a feature that can change the number of arrays (see Figure E) to be visualized by the system. The drop-down button (containing numbers) is an example that can be used to do this activity so that users can select the number of arrays they want to visualize in the system.

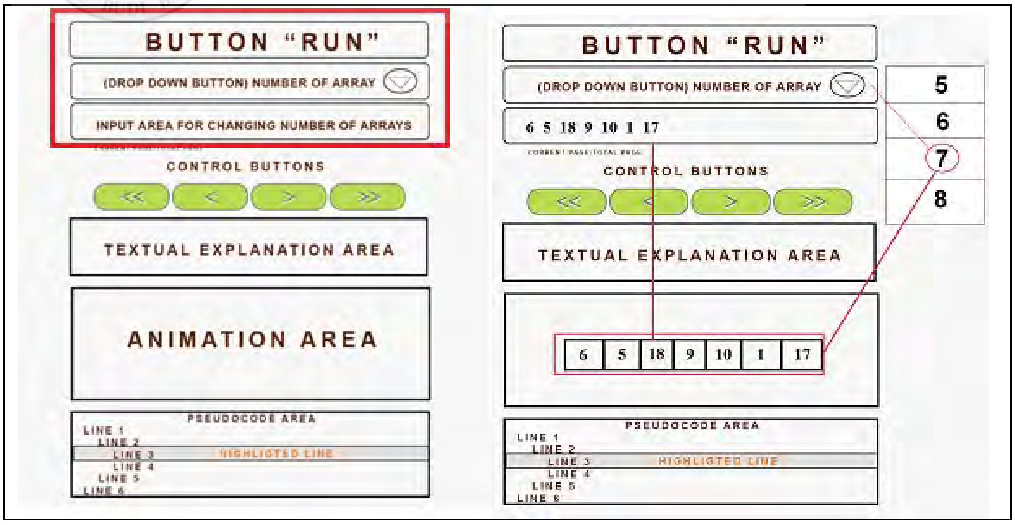


Figure E. Changing features.

Category 4: Constructing	
G14	**Recommendation:** AV should provide constructing activity **Basis**: Engagement Taxonomy (ET), Two-Dimensional Engagement Taxonomy (2DET), Constructivism Learning Theory, Discovery Learning Theory **Explanation**: Constructing activity is grounded from the constructivism learning theory, which means learning or understanding processes are based on human experiences itself (Büyükduman & Şirin, 2010). In a sense, that constructivism necessitates the learner to construct their own knowledge through activities or interactions with learning material itself such as "creating, constructing, developing and inventing (Guney & Al, 2012). Specifically, constructing activity is the activity that requires the student to learn through simulating the visualization itself, touching or interacting with the visualization elements (Myller et al., 2009; Naps et al., 2002). This is also in line as suggested by theory of discovery learning that allowing learners to figure out the learning material through reasoning and experimentation should improve learners' comprehension (Mahmoud & Kamel, 2014). A study by Hundhausen and Brown (2008) found that learners using their AV system (ALVIS Live!) that consists of developing algorithms and constructing visualizations surpassed the other group using presentation media and traditional text editor. This is also the same with a study (Urquiza-Fuentes & Velázquez-Iturbide, 2007) that compared viewing group and building visualization group on tree traversal algorithm; they found the building group spent more time (more engaging) than viewing group, and thus it affected on better performance result by building group. Researchers agree not to set time limits for this activity because construction should have a deeper knowledge of algorithms and should therefore give them more flexibility in time (Sorva et al., 2013). **Implementation**: 1. Provide exercise activity that requires the student to demonstrate and solve the steps of a particular algorithm based on given questions by clicking or swapping the visual elements (see Figure F). 2. Do not put time constraint in this activity. **Illustration**:

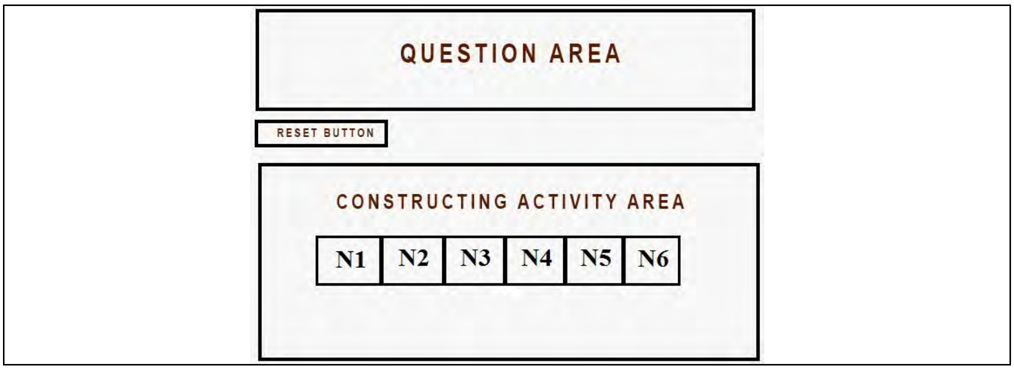


Figure F. Constructing activity example

The Figure F shows a set of numbers (N1-N6) in a case of sorting algorithm activity example. The student will be asked to sort the numbers (N1-N6) in increment sequence by swapping the boxes based on the given question of the particular sorting algorithm. The students can do swap gesture by touching the boxes (N1-N6).

*__Note__: it is up to the designer to set how many attempts the students can try in order to achieve the correct answer.

G15

Recommendation: Give feedback during constructing activity

Basis: Engagement Taxonomy (ET), Human Computer Interaction (HCI) principles

Explanation: The availability of the feedback during constructing activity (G14) will support students in terms of recognizing the error and also improve their comprehension towards a particular algorithm (Mulvey, 2015; Osman & Elmusharaf, 2014). Feedback is also an example of HCI design principle that declares clear feedback in response to student's action could improve the usability of the application (Boticki et al., 2013; Rogers et al., 2011).

Implementation:

Provide feedback in form of text/notification message once (see Figure G) user press button that can inform the guidance, error message, correct answer, and info.

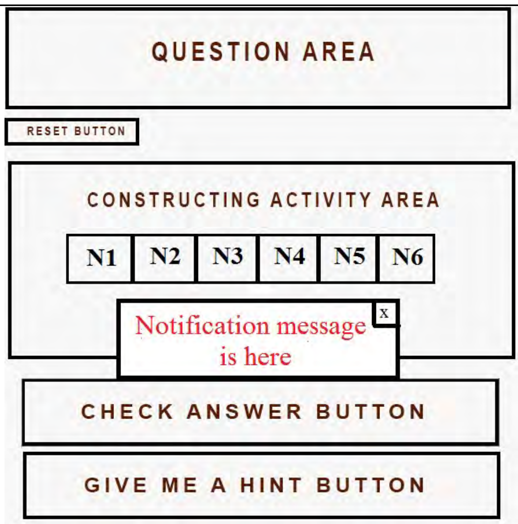


Figure G. Feedback buttons for constructing activity

*__Note__: the students can press the hint button provided whenever needed, and then the hint will be shown to them. Student also will be informed once he presses the check answer button, either the answer is wrong or correct, at the same time giving the guidance of what to do next.

Category 5: Error Handling

G16

Recommendation: Prevent and handle error

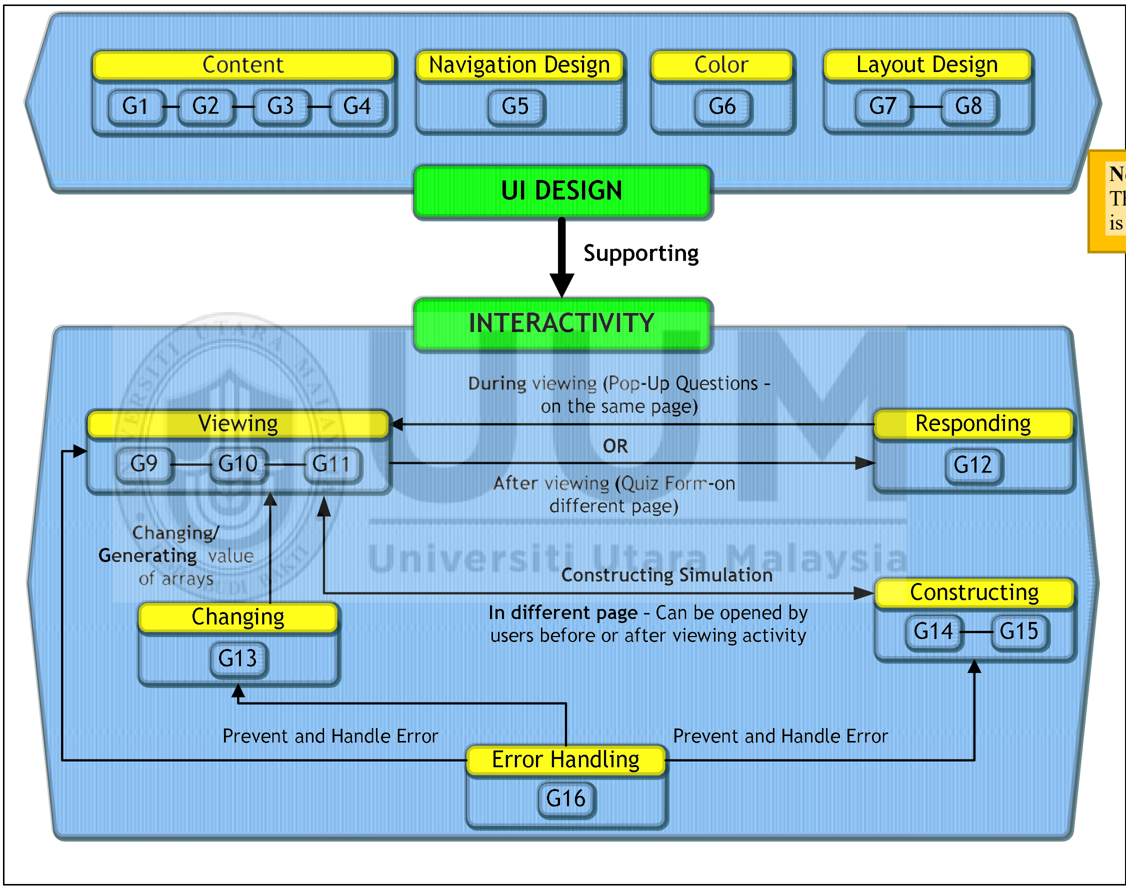
Basis: Usability of mobile learning

Explanation: This suggests that the system should be able to prevent the serious error and also can detect the error if the users make a mistake, such as input or value (Hujainah et al., 2016; Osman & Elmusharaf, 2014). Then, the system will provide such simple and constructive instructions to recover.

Another factor that needs to be considered is the size of navigation buttons, as Cockburn et al. (2012) revealed from their experimental study that the high error rated was caused by the "fat finger" problem. This is mainly attributed to the size of a fingertip that does not match the size of touch buttons. For instance, much experimental research on mobile application usability found that mobile application buttons should be large and appropriate to the size of fingertips (Huber, 2015).

Implementation:

1. Make a prevention by using disabled UI elements which will be enabled after users achieve a certain condition (e.g. pressing a run button in AV system)
2. Give an alert or warning message to learner if they make a mistake.
3. <u>Set the size of navigation buttons that fit fingertip size (Hoehle et al., 2016) and also consider the gap (white space) between each button for a small button.</u>



Note: This Figure is added.

Figure 5.3. The big picture of the AVOMP design guidelines

CHAPTER SIX
PROTOTYPING OF AVOMP DESIGN GUIDELINES

6.1 Overview

The AVOMP prototype is constructed in order validate the usefulness of AVOMP design guideline in answering the third research objective. In this prototyping phase, the sorting algorithms are selected as a case study of data structure and algorithm subject. Particularly, three sorting algorithms are chosen, which are Bubble sorting, Insertion sorting, and Selection sorting. The constructions processes are explained in the next sections.

6.2 The Process Flow of AVOMP

This prototyping phase starts from implementing the proposed AVOMP design guidelines (see Chapter 4). Therefore, the first task to be done is to design the content itself. The arrangement of all UI components, such as buttons, images, text, text boxes, etc. should be carefully designed and built based on UI design and interactivity aspects.

The flow of AVOMP application begins with the user opening the app and the main menu will be displayed (see Figure 6.1). Then, the menu will appear where they fall into two categories, namely viewing and testing. The menu represents bubble sorting, inserting sorting, and selection sorting algorithm. Viewing category consists of the visualization of algorithms (showing step by step process of the selected algorithm), while the testing category displays the algorithm's proficiency test (simulating the visualization of an algorithm).

Users in the viewing category can watch the step by step process as they hit the next “>” button. They can also do backward motions by pressing the back button “<”. Two other buttons (“>>” and “<<”) are also provided to navigate the user towards the beginning and ending of the algorithm process. The viewing category also offers textbox options for users to change or insert their own numbers to be processed by the system so as to improve their understanding by looking at the different numbers being sorted visually by the system.

Meanwhile, the proficiency testing of sorting algorithm permits users to simulate a particular sorting algorithm by doing swapping gesture towards the array boxes provided. Specifically, this is done by tapping two boxes on the screen sequentially in order to swap their positions. The testing category also comes with feedback that serves users with clue, guide, info, and alert. When the user successfully answers the question correctly, a smiley icon will appear indicating the correct answer; the user is then prompted to press the next question button (generating the same question but with different or random numbers). This allows the user to have more experiences with various cases. Conversely, the effect of the shaking button will be triggered if the answer is incorrect; the user is then prompted to try again or press the reset button. The following sections describe in detail the AVOMP application and its implementation guidelines.

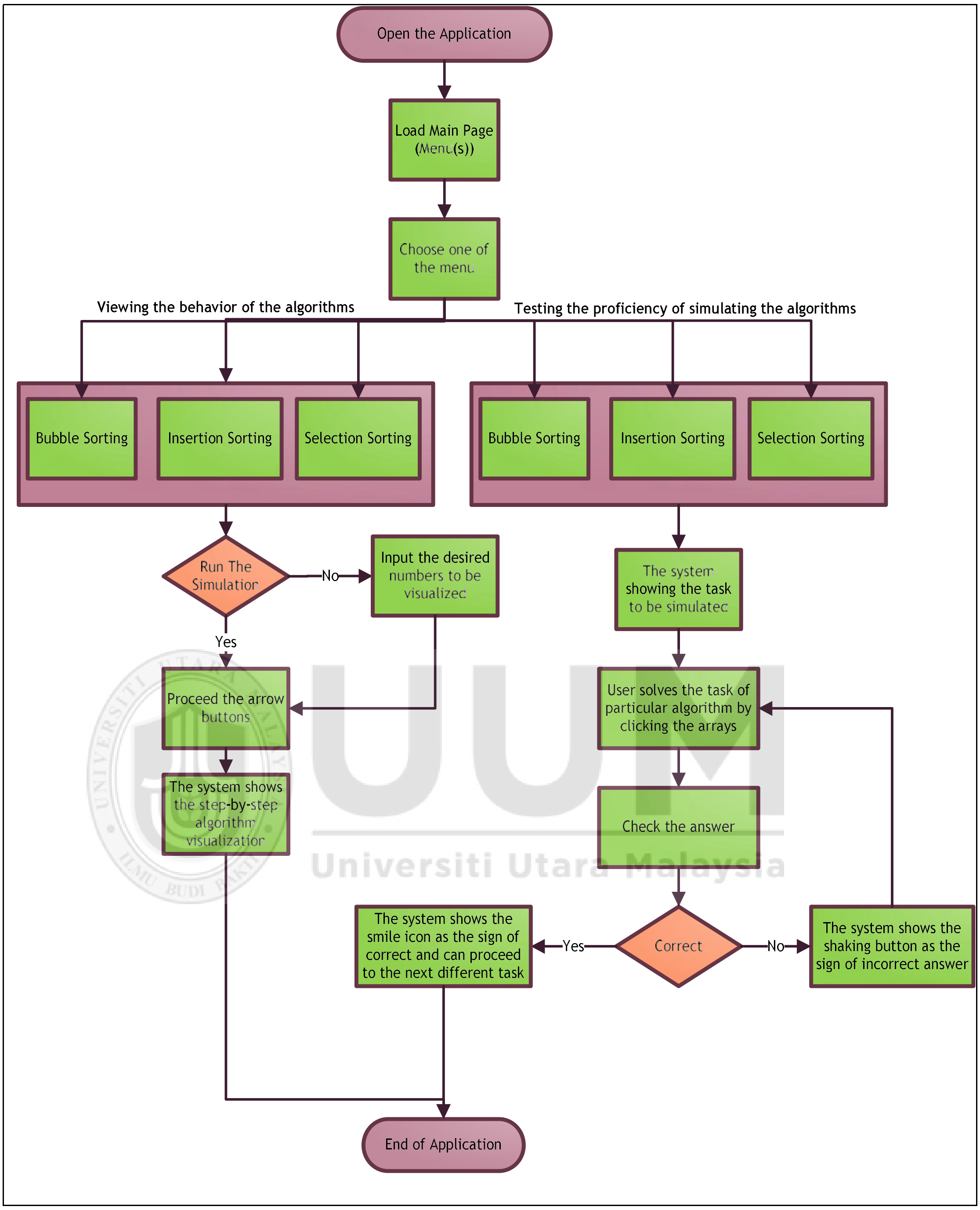


Figure 6.1. Process flow of AVOMP prototype

6.3 The Development of AVOMP

As mentioned earlier in Chapter 4, to achieve an effective learning process of data structures and algorithms, the AVOMP design guidelines have been proposed and reviewed in two aspects, UI design and interactivity. Therefore, all components (text, images, sentences, animations, buttons, colors, etc.) are carefully crafted based on the proposed design guidelines. Here is a screen shot of the prototype as a result of the implementation of the proposed guidelines.

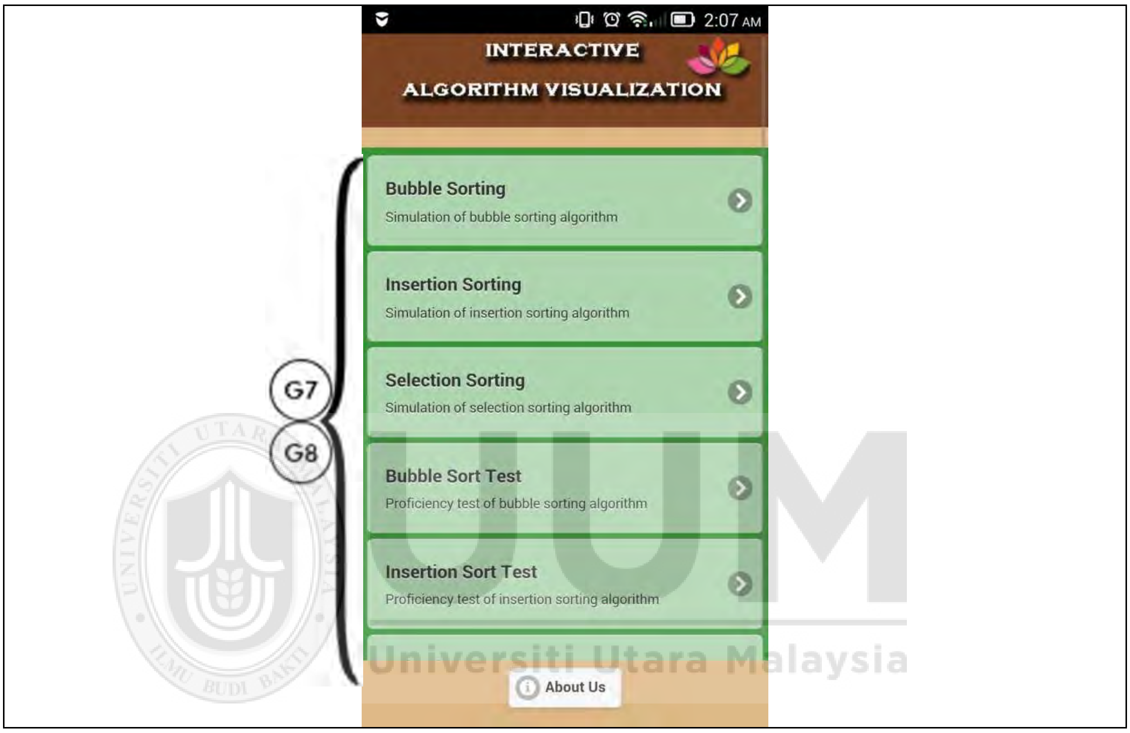


Figure 6.2. The main interface of the prototype

This is the prototype's main interface (see Figure 6.2). Guidelines 7 and 8 are implemented in this interface. Guideline 7 is a principle of similarity to the layout that makes the button design uniform and consistent in terms of shape, color, shape. As such, it makes the design clear and organized (Hoehle et al., 2016).

After the user clicks on bubble sorting menu, for example, the next interface is displayed (see Figure 6.3). Three elements in this interface are displayed, namely "run" button, drop-down button, and text-box. According to Guideline 3 (Changing category), "Run" button is used to generate random values of arrays automatically, if the user does not insert any numbers in the text-box. Meanwhile, if the user inputs a set of numbers in the text-box and then clicks run button, the system will set the arrays' values based on those numbers. Beside of that, another guideline used is guideline 16 (Prevent and handle error) in which the other elements (buttons and visualization) (Figure 6.3) are hidden at the first glance before the user pressing the run button.

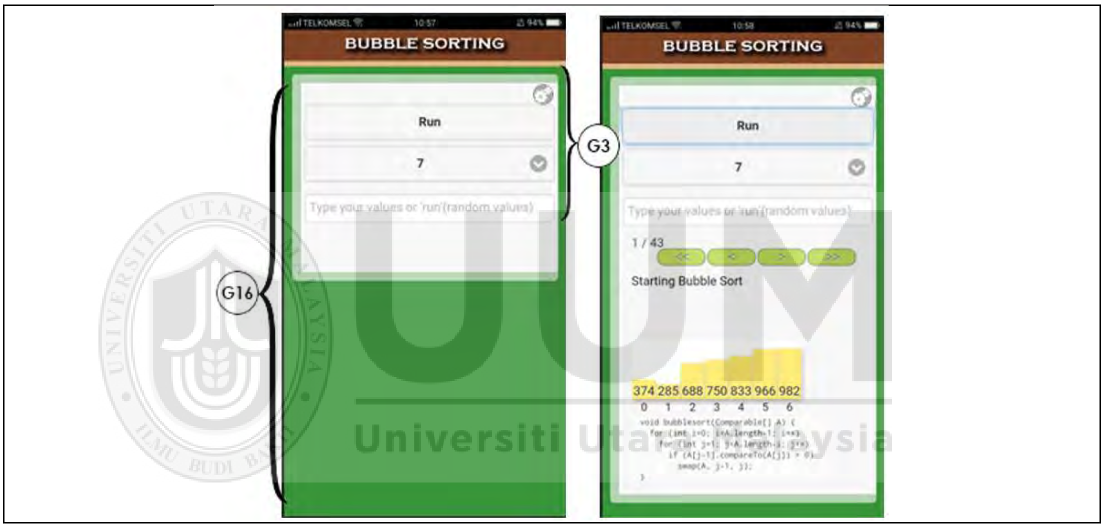


Figure 6.3. The interface of bubble sorting

These hidden objects are useful for making system flow (top to bottom) clear and prevent users from confusions about what to do next in the interface. In other words, showing all elements at first are not recommended due to it could create confusion and increase cognitive load of users since they need to recognize and comprehend all components at the beginning (Hujainah et al., 2016). The last element is a drop-down button which enables users to also change the total number of arrays (see Figure 6.4) they want to view in the visualization.

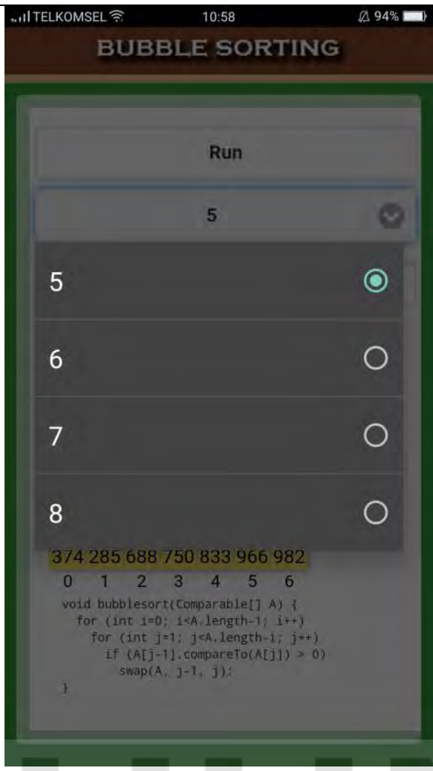


Figure 6.4. The changing drop-down button of bubble sorting

6.3.1 Implemented Guidelines for Viewing Interface

After clicking on one of the viewing activity menus (for instance bubble sorting), other components (buttons, pseudo code, and visualization) will be displayed (see Figure 6.5) as the requirement of viewing activity (Guideline 9). Rewinding button (Guideline 10) is also helpful as it helps learners to retreat when they are still confused with the logic of an algorithm along the way. Therefore, the status number (Guideline 10) of current step is also displayed accordingly, notifying the learners the step taken.

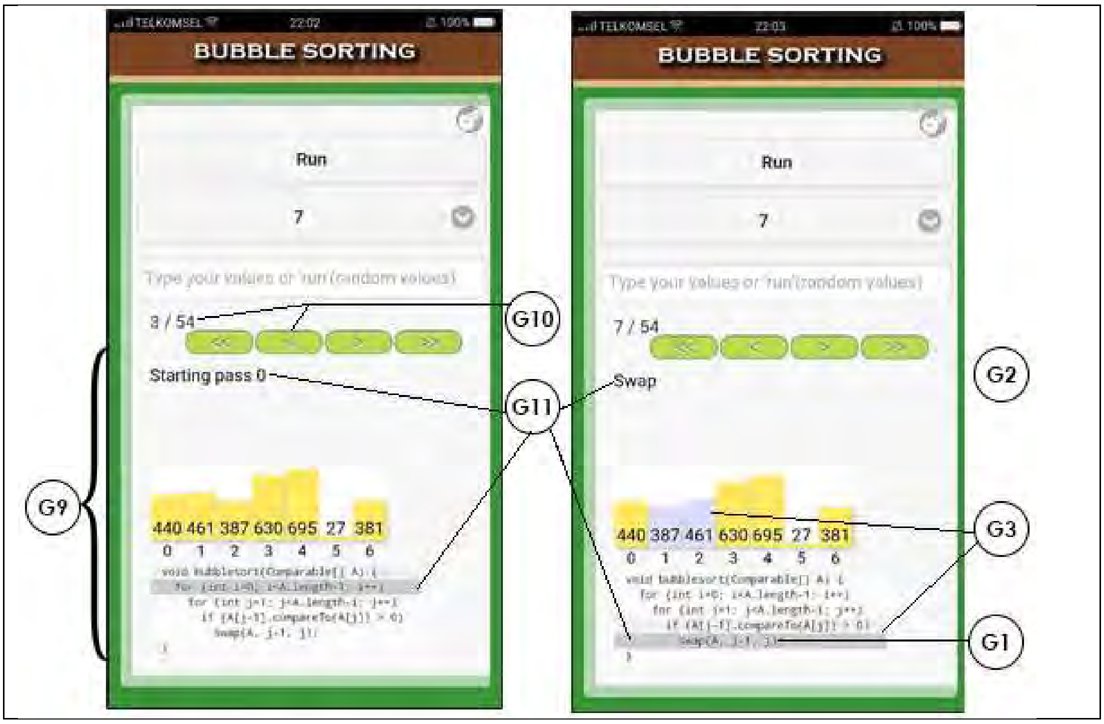


Figure 6.5. The guidelines of viewing interface

Guideline 11 is applied in order to show a dynamic explanation of visualization as the student presses the buttons, which are the downward movement of the highlighted line of pseudo code and alteration of textual information (see Figure 6.5). These are presented simultaneously with its visualization (Mayer, 2009), which is highly recommended to minimize cognitive load and improve students' understanding (Urquiza-Fuentes & Velázquez-Iturbide, 2013).

In Figure 6.5 also displays Guideline 1 is implemented using an interesting event approach in the sense of summarizing the swapping code (usually presented in three lines in detail process) into a single line code only (representing a swapping procedure). This summary can be used to reduce irrelevant content, especially on mobile platforms that have a screen smaller than on desktop platforms. The other ways to apply Guideline 1 are to employ simple and direct expressions for textual information, avoid background decoration, and use popup agent button (showing additional information on hidden page). All of these are also related to compliance with Guideline 2 to avoid scrolling on the phone screen. Otherwise, learning while scrolling can cause split attention effect (Churchill, 2011). Therefore, the selection of UI components, as well as its size, should not exceed the size of the mobile screen. Another way to prevent scrolling is to implement responsive design, which means the size of UI components can flexibly adapt and suit to any other sizes of mobile screen.

Guideline 3 is about cueing or signaling that serves as highlighting the essential information and connecting related objects. In this case, the highlighted line on pseudo code informs user upon the processed step of an algorithm and connects it with its corresponding visualization.

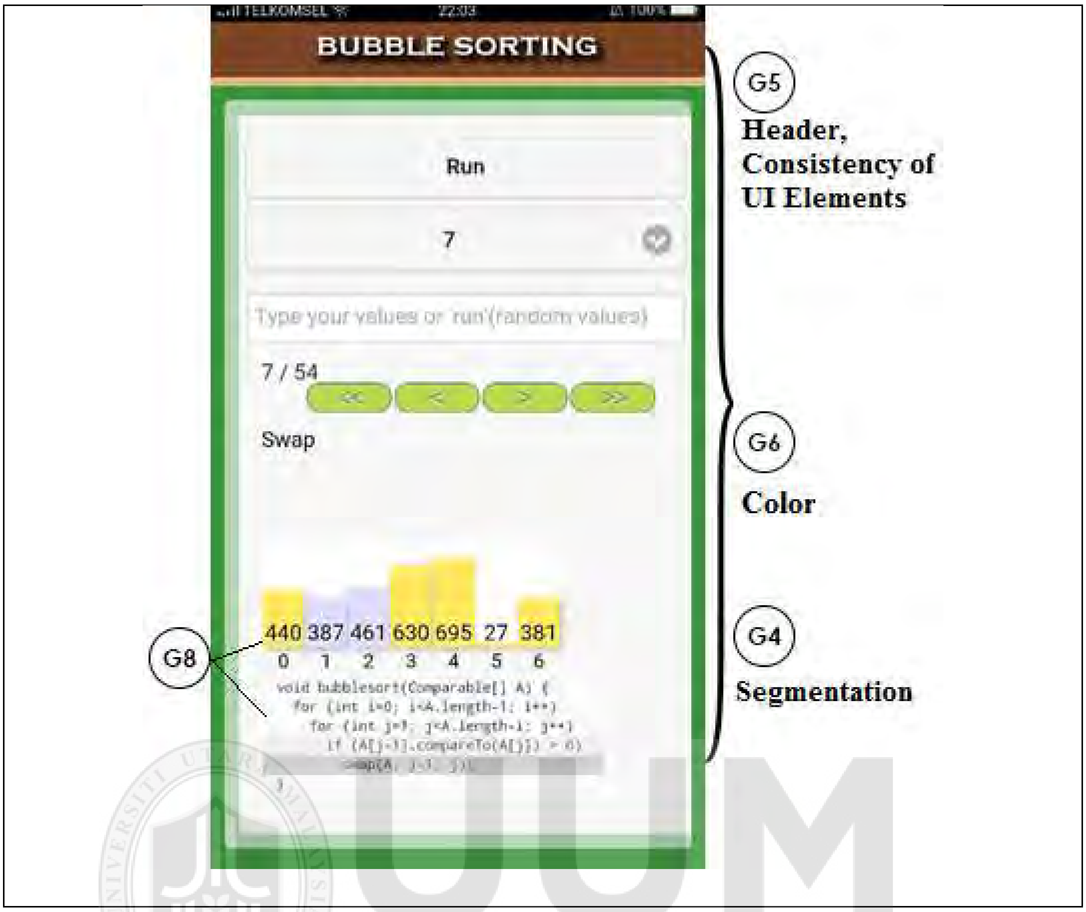


Figure 6.6 The Other Guidelines of Viewing Interface

The Figure 6.6 shows 4 guidelines applied, which are number 4, 5, 6, and 8. Guideline 5 is about providing simple and straightforward hierarchical navigation system. One of examples is header (see Figure 6.6). Header is important to indicate the connection after the user clicks the menu (bubble sorting menu in this case). Another way is to use breadcrumb (showing the navigational aid in the interface) which makes users keep in track of their location in the system. Guideline 5 also informs about the consistency of UI elements (text, button, animation, pseudo code, and text-box) in many pages, such as consistency of its position and design styles. The consistency of design layout makes users have better recall whenever they go to the other pages (Karam, 2015) and as a result increase their satisfaction level (Hujainah et al., 2016).

Guideline 6 is the appropriate use of colors in the user interface. For example, the use of sufficient contrast color between text and background to achieve readability. It is also used as mentioned earlier to highlight important information, group information, and establish relationships between elements. Guideline 4 is for segmentation purposes. The visualization of the algorithm itself is one of the forms of the segmentation methods in which it divides an algorithm into a collection of meaningful information. Theoretically, this segmented information, presented with control button navigation, provides a pause effect for learners to help them digesting information gradually (Liu, Geurtz, et al., 2013). Another thing of segmentation is to separate different elements in which one can use line, box, and white space, helping students in minimizing excessive load.

On the other hand, Guideline 8 is conducted to make the related elements close one another. This is validated by the principle of spatial contiguity principle (Mayer, 2009) and the principle of proximity of gestalt theory (Guberman et al., 2012) that approved the closeness of two related objects would be considered a group by learners thereby reducing the split attention effect.

6.3.2 Control Panel of Controlled Viewing

This is part of the viewing category in Guideline 9 (see Figure 6.7) that supports controlled viewing as a means to resolve how visualizations will appear. It deals with how learners modify the visualization environment; whether related to changes in animation speed or visual element style (Sorva et al., 2013). Therefore, the pop-up control panel is prepared in the prototype, accomplishing the purpose (see Figure 6.7). This makes it easier for learners to change the animation speed and visualization style based on their preference during or after viewing activity. Animation speed adjustments also match the needs of learners; Beginners may require slower speeds to follow visualizations whereas advanced ones can improve speed to save study time. The flexibility of this arrangement is judged to increase the satisfaction of learners because they feel under control of the system (Sung & Mayer, 2013).

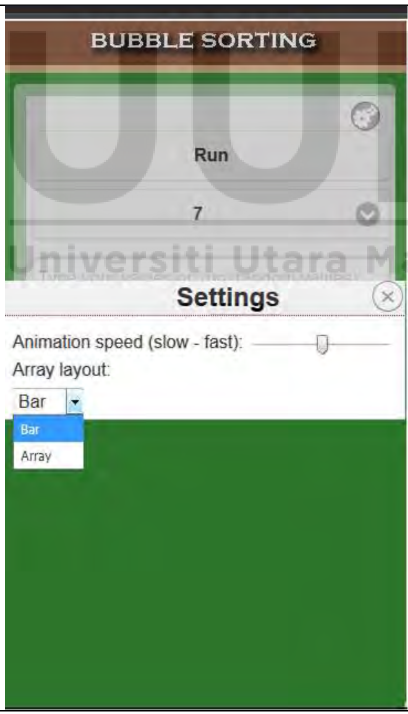


Figure 6.7. Controlled Viewing Settings

6.3.3 Error Handling on AVOMP Interface

Guideline 16 on prevention and error handling applied to the interface. In this case, input-text has been prepared to prevent user input errors (see Figure 6.8). If the user passes the conditions that must be met, then the system will automatically alert in the form of alert messages. Figure 6.8 shows two different cases; the left picture is when the user wants to set his own number (changing activity) which exceeds more than eight digits (as max numbers) so the warning appears on the screen. The right picture is when the user inserts other than the number characters between 0 and 999, a warning will also appear.

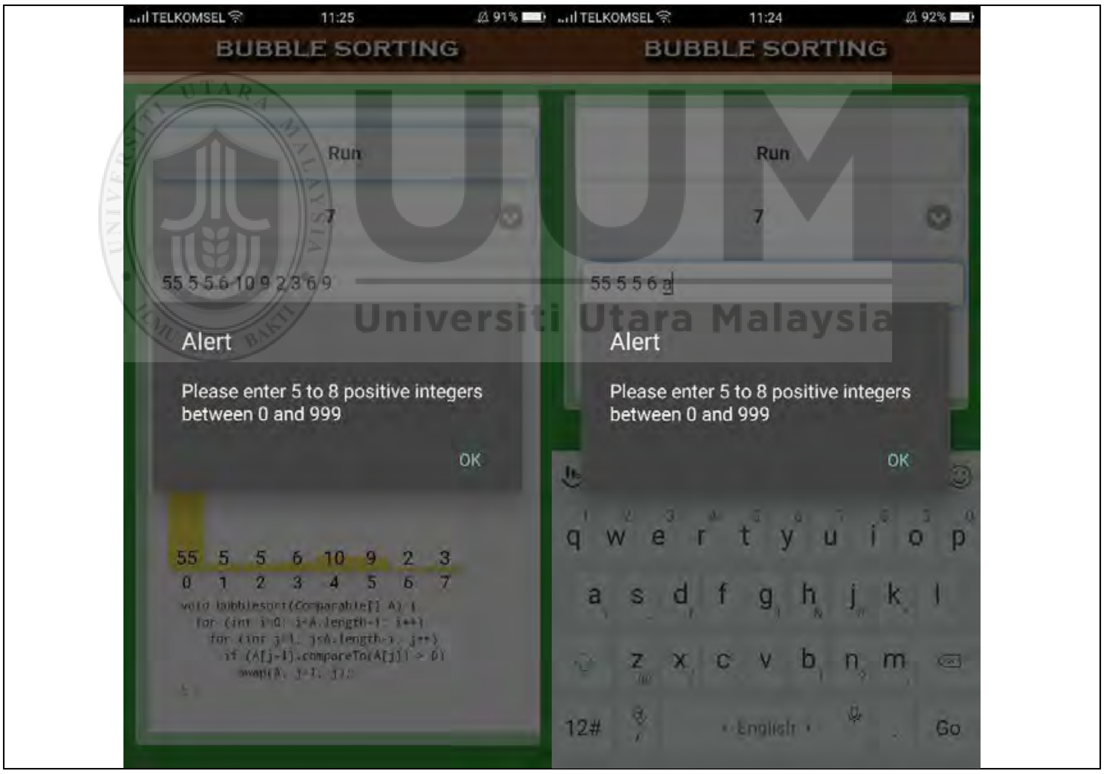


Figure 6.8. Errors handling on AVOMP

Another error prevention on the prototype is settling the size of navigation buttons (see Figure 6.9) that fits fingertip size. This guideline is very important considering that smartphone users use their fingers to navigate menus or buttons. Hence, the buttons should be large and fit the fingertip size (Hoehle et al., 2016). Nonetheless, if by any chance the AV designers want to use small buttons one, it is recommended to create a small white space gap between them.

Figure 6.9. Errors handling on button navigations

6.3.4 Constructing Activity of AVOMP

Guideline 14 is about constructing activity. Constructing activity is built on prototypes as a regard to applying constructivism theory (learning by doing theory) in which it encourages users to test their abilities while simulating algorithmic steps. Users need certain troubleshooting of selected sorting algorithms by performing swapping gesture (touching) on the array interface (see Figure 6.10). This stimulates their cognitive performance as it enriches their experience in solving problems. Figure 6.10 shows that a student needs to sort the numbers incrementally based on insertion algorithm question.

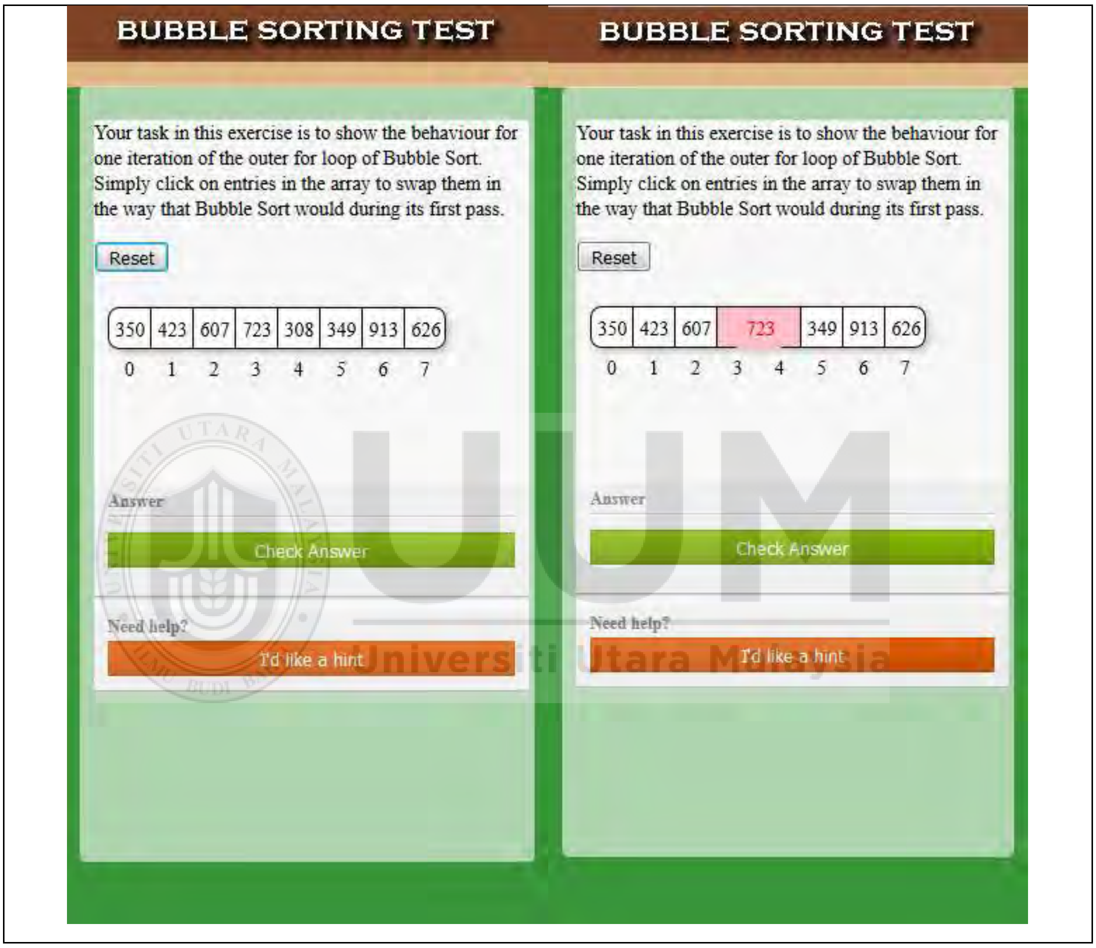


Figure 6.10. Constructing activity on insertion sorting

The user is also equipped with a hint button in which the textual hint will appear on the screen after user presses it (see Figure 6.11). This hint is implemented pursuant to Guideline 15, which provides feedback during construction activities. Another feedback applied to the prototype is checking the user's answer. Once the user is satisfied with the answer, he can then proceed by clicking the "check answer" button.

If the answer is incorrect, then the button will show a shaky effect. While if the answer is correct, then a smiley icon will appear (see Figure 6.12).

Figure 6.11. Textual hint of constructing activity

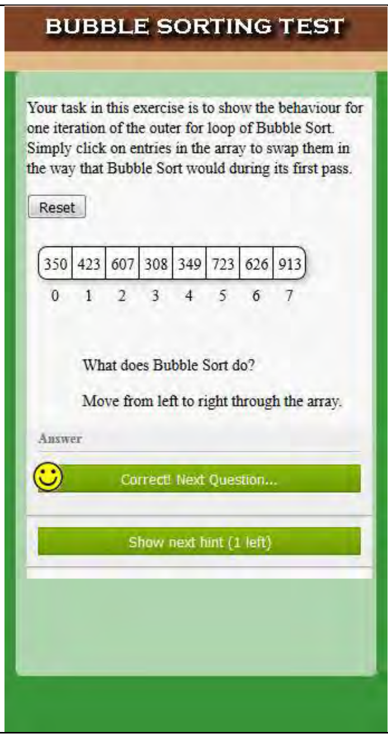


Figure 6.12. Smiley icon feedback of constructing activity

The smiley icon is set as a symbol of positive effect that can evoke pleasurability feeling towards users (Kim, Kang, & Choi, 2014) as they solve the problem correctly and at the same time will encourage their learning development as well (Puteh, Adnan, Ibrahim, Noh, & Che'Ahmad, 2014). As such notes "Correct! Next Question" on the button (see Figure 6.12) will lead the learner to try again facing the same questions but with different set of values that are randomly generated by the system. This feature generates random value indefinitely, so it presents many opportunities for learners to constantly try and understand the nature of each algorithm every time they solve a new problem until finally opening up the understanding in the long run.

6.4 Summary

This chapter essentially elaborates on the construction of AVOMP prototype as the implementation of the purposed AVOMP design guidelines from Chapter 4. Besides, the flow of processes that exist in the AVOMP is also discussed in a succinct manner. To achieve that purposes, the sorting algorithms are selected as case study from data structure and algorithm subject. The prototype is constructed following two key aspects, namely UI design and interactivity. All guidelines are implemented in the development of prototypes which are all combined in a single unit within an application. Detailed description and function of each feature are also provided in this chapter. Next chapter discusses the testing prototype towards IT students who enroll DSA subject as the validation on the effectiveness of AVOMP.

CHAPTER SEVEN
AVOMP EFFECTIVENESS

7.1 Overview

The chapter is in addressing the forth objective of the study which is to measure the effectiveness of the AV on mobile platform that implements the proposed design guidelines. Thus, a series of experimental laboratory experiments were conducted at this stage. This study focuses on measuring the effectiveness of AVOMP application. This is carried out through the controlled experiment whereby pre-test and post-test are involved as a means to measure how far AVOMP application can improve students' comprehension in conjunction with three sorting algorithms, namely bubble sorting, insertion sorting and selection sorting. Additionally, several questions are also given to students to see their subjective opinion about AVOMP application.

7.2 Controlled Experiment

The controlled experiments in this study include pre-test and post-test tests, which are based on the revised Bloom taxonomy level (Forehand, 2010). Essentially, this experiment compares the students' performance results on sorting algorithm sub-topic between using manual approach (slides and lecture's note) and the prototype (AVOMP application). The flow of both tests (see Figure 7.1) is initially started with students learning manually the three sorting algorithms using slides and notes that lecturers have prepared. They have been given 40 minutes to study and then continue to do pre-test in 40 minutes as well. After that, they are directed to learn the sorting algorithms using AVOMP application in 40 minutes and then prompted to do the post-test in 40 minutes. In addition, to better understand the student performance outcomes of the post-test, the study continued by measuring the student's opinion towards the prototype in the next 15 minutes through questionnaire survey.

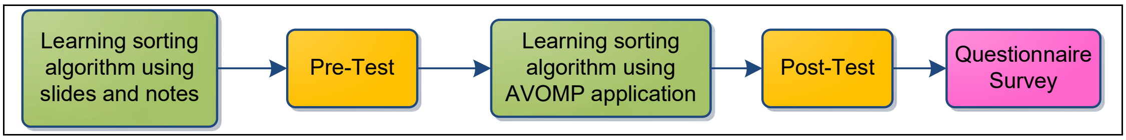


Figure 7.1. The flow of tests

7.2.1 Instrument

The effectiveness evaluation of the AVOMP prototype reflects from the revised Bloom taxonomy levels that consist of 6 levels. Nonetheless, this study only takes 4 out of 6 levels of Bloom taxonomy. The last two levels, the evaluate and create levels, are not involved because they are considered as the most difficult ones in which students require more time to complete the tasks. (Benitti & Sommariva, 2015; Thompson, Luxton-Reilly, Whalley, Hu, & Robbins, 2008). Meanwhile, this study has limitation in terms of lecturing time as it is written as the scope of study in Chapter 1. Both evaluate and create levels need profound explanation to be delivered due to they are classified as the highest cognition level (Bhagyalakshmi & Seshachalam, 2015). Specifically, the evaluate level prompts student to evaluate, review, compare and criticize each of algorithms. The create-level requires the student to develop a new alternative algorithm that has a better solution, which can be also from the combination of all three sorting algorithms they have learned. The followings are the 4 levels used in this study:

i. **Remember**: This is related to recalling and recognizing any materials covered in teaching class. Such as, the definition, process, design patterns of particular algorithms. In this case, this study asks student to recalling the pattern used in each sorting algorithm, such as to find out how many loops are used in the bubble sorting algorithm.

ii. **Understand**: This is related to interpreting, exemplifying, summarizing, classifying, and explaining. Thus, this study requires students to explain the process of pseudo code of each sorting algorithm.

iii. **Apply**: This is related to carrying out or implementing the procedure or method in a certain case. This means the students are required to apply the sorting algorithm into a given numbers.

iv. **Analyze**: This is related to identifying, differentiating and attributing the parts of knowledge and its relation to other parts as a whole. This study takes identifying question that needs student to identify the different between parts of codes

Questions shared in both pre-test (see Appendix C) and post-test (see Appendix D) are the same in terms of difficulty level but different in terms of numbers to be sorted and analyzed as well as the perspective of several questions. Each test contains 11 questions that are categorized based on these 4 levels of Bloom taxonomy. The questions are a mixture of multiple choice and essay (see Figure 7.2 for detail). After the students finish both tests, they are required to fill the last questionnaire survey in order to measure their perspective opinions towards the AVOMP app. This is executed by giving several statements to students to be assessed. The Likert Scale on a five-point scale is applied on it to examine how strongly the learners agree or disagree towards statements with the following anchors: (1) Strongly disagree, (2) Disagree, (3) Neither Agree or Disagree, (4) Agree, (5) Strongly agree. The followings are the statements given:

11. The AVOMP app is an effective tool for me to learn sorting algorithm.
12. The AVOMP app made sorting algorithm easy to learn
13. The AVOMP makes the lesson and the exercise of sorting algorithm more fun
14. The step by step process of visualization along with highlighted line of pseudocode as well as its explanation in viewing activity help me to understand each algorithm in depth.
15. The features in viewing activity, such as changing numbers to be solved and changing the numbers of arrays, help me to better understand the behavior of each algorithm.
16. The proficiency test by simulating the array visualization help me to understand sorting algorithm better.
17. The hints in proficiency test helps me to solve algorithm test
18. The design layout of AVOMP app is appealing for me to use the app.
19. The layout structure (arrangement, color, shape, consistency, and placement) help to me understand the flow of AVOMP app.
20. All textual contents (description, hints, information, questions, warning, etc.) in AVOMP app are clear and understandable.

The statements number 1 until 3 are about the AVOMP app in general. The statements number 4 and 5 are about viewing activity. The statements number 6 and 7 are about proficiency test. The statements number 8 until 10 are about user interface.

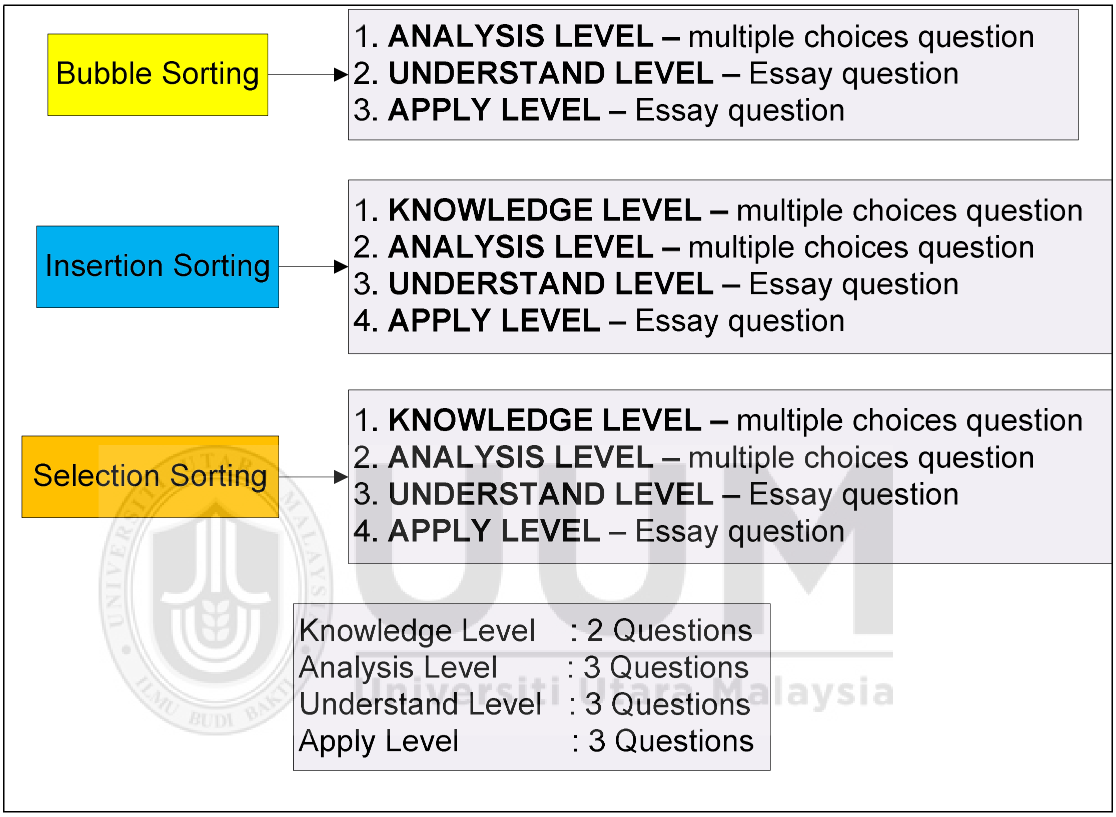


Figure 7.2. The Bloom Taxonomy of questions structure in Pre and Post Tests

7.2.2 Respondents

35 students participated in this experiment, consisting of 24 females and 11 males (see Figure 7.3). They are IT students in the second and third semester of Universiti Utara Malaysia (UUM) who enroll in Data Structured and Algorithm (DSA) subject. The sorting algorithm itself is a DSA last sub topic that lies at the last meeting of the semester in UUM. Meanwhile, this experiment was deliberately conducted right at

the end of this course because it does not want to disrupt students' time in other subjects. It is also expected that they will become more focused and do not require additional time when compared to if this experiment is done outside the hours of the lecture. Therefore, this research runs naturally in accordance with the schedule of sorting algorithm sub topic time that has been prepared in UUM curriculum that are using slide and note provided (hardcopies) (see Figure 7.4) before finally testing them on a pre-test.

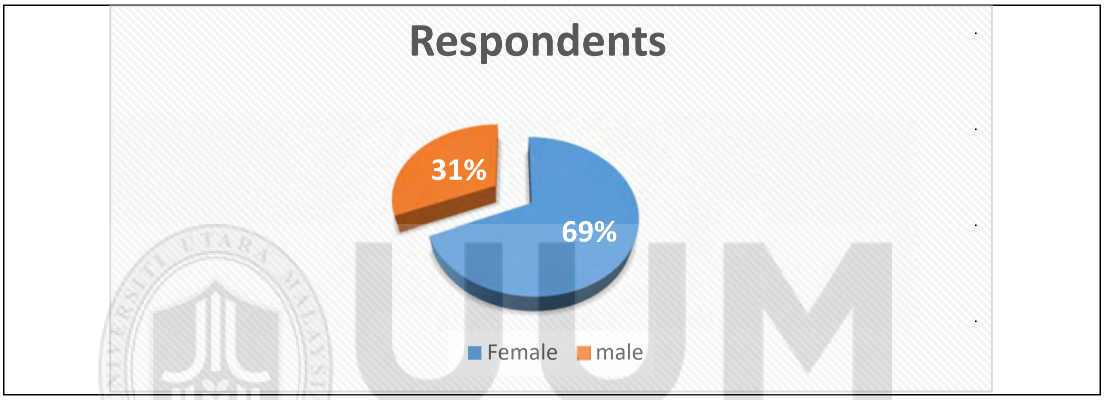


Figure 7.3. The number of participants

7.3 Testing

The experiments were conducted in the classroom, school of computing (SOC), University Utara Malaysia (UUM), involving learning and testing activity. Before manual learning begins (as the first phase on the testing), they are taught on how to view and read algorithms on slides and notes. Then the students are given time to learn sorting algorithms on their own in the first 40 minutes. In this phase, they are allowed to study individually or in groups. Then, after the time of study runs out, pre-test was held in the next 40 minutes.

Then, as noted earlier, after pre-test the experiment continues with the learning of sorting algorithms using the AVOMP application (see Figure 7.4). First and foremost, they students are directed to download the app through GooglePlay store with the keywords: “inavomp”. As the scope of this study, this prototype was developed for android users only, and thus the other users who use iPhone for instance are not able to do so. There were 4 iPhone users and 31 Android users. However, this obstacle is not an issue in this experiment because group learning is allowed, and thus for the iPhone users, they can join and study together with other Android users.

After downloading and installing the app, students are then led and shown on how to use the app including the overall interactive features. They were then given 40 minutes to study the sorting algorithms on their own (individually or group) and were also allowed to use the stationeries (to make notes or scribbles). They were also given the opportunity to ask anything about AVOMP application during the lesson. After the learning time, they are eventually requested to complete the post-test for another 40 minutes.

Figure 7.4. Learning sorting algorithm using AVOMP application

7.4 Results and Discussion

In order to analyze the paired sample, in this case, the related Pre-Test and Post-Test, paired sample t test is usually utilized to determine whether or not there are significant difference between them. The requirement of using the paired sample t test is to have data which is normally distributed. Therefore, prior to that, the normality test should be performed. There are 4 types of normality test, namely

Shapiro Wilk, Kolmogorov Smirnov, Lilliefors, Anderson Darling (Yap & Sim, 2011). In terms of the number of samples, if there are 50 or less respondents, Shapiro Wilk is used, whereas if there are more than 50 respondents, then Anderson Darling is used (Lantz, Andersson, & Manfredsson, 2016). Meanwhile, Lilliefors and Kolmogorov Smirnov tests should be used in samples less than 100 (Corder & Foreman, 2014). Thus, this study opts the Shapiro Wilk to perform the normality test as the data is 35 respondents, which is less than 50 respondents.

In conjunction with this study, 8 paired data (the pre-test and post test result) are categorized and tested (see Figure 7.5). They are overall data, bubble sorting data, insertion sorting data, selection sorting data, knowledge bloom taxonomy data, analysis bloom taxonomy data, understand bloom taxonomy data, and apply bloom taxonomy data.

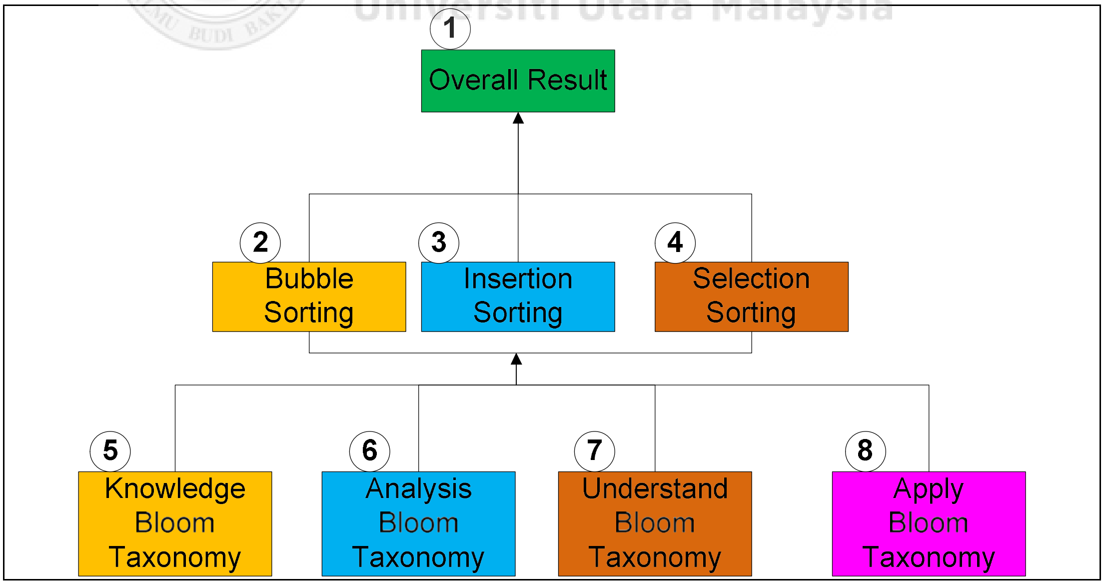


Figure 7.5. 8 Paired Data (Pre-Test and Post-Test).

The normality of distributed data using Shapiro Wilk can be concluded by checking the significant (sig) value of each category. On condition that if the sig. value is more than 0.05, this means data is distributed normally, otherwise data is not distributed normally (Park, 2015). The following are the results of their normality test using Shapiro Wilk:

Table 7.1

Normality Test Using Shapiro-Wilk.

	Shapiro-Wilk		
	Statistic	**df**	**Sig.**
Overall Result Pre-Test	.948	35	**.096**
Overall Result Post-Test	.825	35	.000
Bubble Sorting Pre-Test	.916	35	.011
Bubble Sorting Post-Test	.526	35	.000
Insertion Sorting Pre-Test	.970	35	**.445**
Insertion Sorting Post-Test	.648	35	.000
Selection Sorting Pre-Test	.938	35	.048
Selection Sorting Post-Test	.809	35	.000
Knowledge Pre-Test	.753	35	.000
Knowledge Post-Test	.517	35	.000
Analysis Pre-Test	.890	35	.002
Analysis Post-Test	.678	35	.000
Understand Pre-Test	.914	35	.010
Understand Post-Test	.490	35	.000
Apply Pre-Test	.953	35	.143
Apply Post-Test	.709	35	.000

Based on the result of normality test from Table 7.1, only Overall Result Pre-Test and Insertion Sorting Pre-Test that have value more than 0.05, while the rests are less than 0.05. This it can be concluded that almost all data are not distributed normally. As a result, sample t test cannot be employed to this type of data.

Consequently, an alternative way is taken, which is called Wilcoxon test. Wilcoxon test is often used as a substitute to paired sample t test (Yap & Sim, 2011). This is performed since the data are not normally distributed through the normality test in which the data do not meet the requirements in testing parametric statistics, especially paired sample t test. Therefore, it is necessary to be done by the researcher so that the collected research data can still be tested or analyzed by doing non-parametric statistic method (Marx, Backes, Meese, Lenhof, & Keller, 2016).

Wilcoxon test or also called Wilcoxon signed rank test is part of non-parametric static method. Because it is part of non-parametric statistics, the Wilcoxon test does not require normal distributed research data (Park, 2015). Thus, it can be said that the use of Wilcoxon test as a substitute for paired sample t test is the most appropriate step when the data is not normally distributed (Yap & Sim, 2011). Therefore, Wilcoxon test is used in this study analyzing all 8-paired data mentioned previously.

The decision-making basis used in the Wilcoxon test is as follows:

1. If the value of **Asymp.Sig. (2-tailed)** is smaller than 0.05, then H1 is accepted (meaning there is a significant difference between Pre-Test and Post-Test).

2. On the other hand, if the value of **Asysmp.Sign (2-tailed)** is greater than 0.05, then H1 is rejected (meaning there is no a significant difference between Pre-Test and Post-Test).

The elaboration of all analyzed data can be seen in the next sections.

7.4.1 Overall Pre-Test and Post-Test Results

After the experiment, the next phase is to examine the results of the students' test. Hence, the first thing to do is to compare the overall result between Pre-Test and Post-Test in general, which is shown in Figure 7.6.

Table 7.2

Descriptive Statistic of Overall Result.

	N	Mean	Std. Deviation	Minimum	Maximum
Pre-Test	35	61.10	18.082	28	90
Post-Test	35	93.05	7.589	78	100

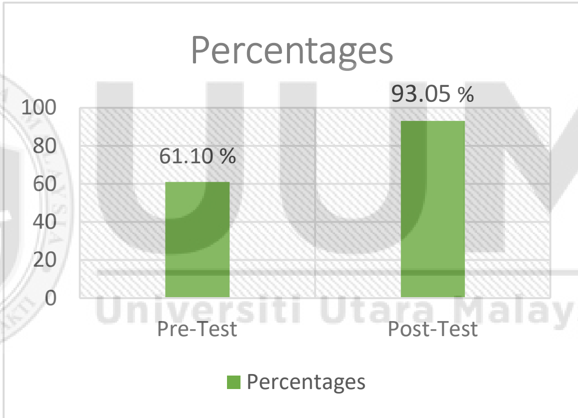


Figure 7.6. Pre-test and post-test of the overall result

Statistically, it appears that the mean of Post-Test score is 93.05 which is greater than the Pre-Test score (61.10) for overall result in general. the Wilcoxon Signed Rank Test in the next table explains in more detail the magnitude of this difference whether it is statistically significant or not.

Table 7.3

Wilcoxon Signed Ranks Test of Overall Result.

		N	Mean Rank	Sum of Ranks
Result_PostTest - Result_PreTest	Negative Ranks	0[a]	.00	.00
	Positive Ranks	35[b]	18.00	630.00
	Ties	0[c]		
	Total	35		

a. Result_PostTest < Result_PreTest

b. Result_PostTest > Result_PreTest

c. Result_PostTest = Result_PreTest

The Rank Table 7.3 provides some data on the comparison of participants' Before (Pre) and After (Post) for overall score. Based on the table's legend, there are 35 positive data (N) which means that all students experience an increase in sorting algorithm learning outcomes from the Pre-Test value to the Post-Test value. Thus, Mean Rank or average increase is equal to 18.00, while the positive rank or the Sum of Rank is 630.00. Accordingly, none of them have lower Pre-Test result and the same result between Pre-Test and Post-Test.

Table 7.4

Test Statistic of Overall Result

	Result_PostTest - Result_PreTest
Z	-5.162[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Based on the results of the Wilcoxon Signed Rank Test calculation (see Table 7.4), the Z value obtained is -5.162 with p-value (**Asymp. Sig 2 tailed**) equals to 0.000 is less than 0.05, so the hypothesis is to accept H1 or that means there is a significant difference between Pre-Test and Post-Test result in general.

In general, students' achievement increases significantly from Pre-test result which is from 61.10 % to 93.05 % in Post-test result. However, even though in average the Pre-Test result is 61.10 % which is categorized as B (see Table 7.5) according to grading schemed initiated by Malaysia Qualification Agency (MQA, 2018), there were 16 out of 35 students (45.71 %) got grades below 51 (see detail at Table 7.6) which means they are categorized into C grade and below.

Table 7.5

Grading scheme by Malaysia Qualification Agency (MQA, 2018)

Grade	Marks	GPA
A+	85-100	4.00
A	80-84	4.00
A-	75-79	3.67
B+	70-74	3.33
B	65-69	3.00
B-	60-64	2.67
C+	55-59	2.33
C	50-54	2.00
C-	45-49	1.67
D+	40-44	1.33
D	45-40	1.00
E	39-30	0.67
F	<30	

Table 7.6

Pre-Test Result of Every Student

No.	Bubble Sorting Pre-Test	Insertion Sorting Pre-Test	Selection Sorting Pre-Test	Average
1	50	60	40	50
2	80	65	75	73.33
3	0	30	55	28.33

4	55	90	65	70
5	0	45	65	36.67
6	55	35	40	43.33
7	95	80	90	88.33
8	50	70	75	65
9	70	90	85	81.67
10	40	55	45	46.67
11	90	80	75	81.67
12	15	60	40	38.33
13	75	75	85	78.33
14	40	35	50	41.67
15	90	85	95	90
16	60	70	55	61.67
17	0	40	55	31.67
18	80	90	85	85
19	70	65	85	73.33
20	30	55	40	41.67
21	75	100	90	88.33
22	40	45	45	43.33
23	50	30	50	43.33
24	70	65	75	70
25	80	80	90	83.33
26	75	80	55	70
27	60	60	70	63.33
28	40	65	55	53.33
29	55	50	40	48.33
30	55	65	80	66.67
31	40	65	50	51.67
32	40	55	65	53.33
33	85	70	55	70
34	65	90	80	78.33
35	50	50	45	48.3333

The following are the number of students who got grades 50 and below:

C (50 – 54) = 4 Students

C- (45 – 49) = 3 Students

D+ (40 – 44) = 5 Students

D (45 – 40) = 0 Students

E (39 – 30) = 3 Students

F (< 30) = 1 Student

7.4.2 Each Sorting Pre-Test and Post-Test Comparison

This section discusses the result of each sorting between Pre-Test and Post-Test. This involves Bubble sorting data, Insertion sorting data, and Selection sorting data. The next sub-sections detail the analysis result using Wilcoxon test.

7.4.2.1 Bubble Sorting Comparison

Table 7.7

Descriptive Statistic of Bubble Sorting Result.

	N	Mean	Std. Deviation	Minimum	Maximum
Pretest bubble sorting	35	64.14	17.843	40	95
Protest bubble sorting	35	95.14	9.586	75	100

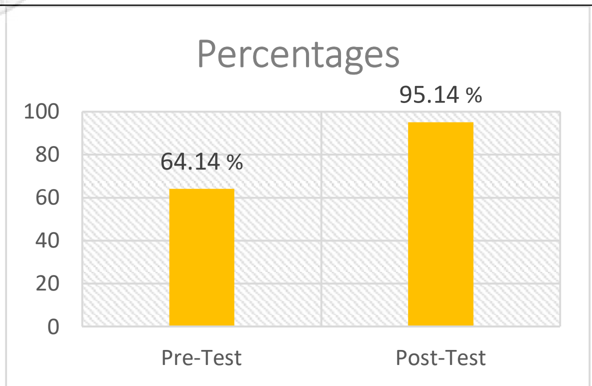


Figure 7.7. Pre-Test and Post-Test of Bubble Sorting Result

Statistically, based on Bubble sorting data (see Table 7.7), it appears that the mean Post-Test score (95.14%) is greater than the pretest score (64.14%). the Wilcoxon

Signed Rank Test in the next table explains in more detail the magnitude of this difference whether it is statistically significant or not.

Table 7.8

Wilcoxon Signed Ranks Test of Bubble Sorting Result.

		N	Mean Rank	Sum of Ranks
BubbleSort_PostTest - Bubble Sorting	Negative Ranks	0[a]	.00	.00
	Positive Ranks	34[b]	17.50	595.00
	Ties	1[c]		
	Total	35		

a. BubbleSort_PostTest < Bubble Sorting

b. BubbleSort_PostTest > Bubble Sorting

c. BubbleSort_PostTest = Bubble Sorting

The Ranks Table 7.8 provides some data on the comparison of participants' Before (Pre) and After (Post) for Bubble Sorting Score. Based on the table's legend (Table 6.8), there are 34 positive data (N) which means that 34 students experience an increase in sorting algorithm learning outcomes from the Pre-Test value to the Post-Test value. Thus, Mean Rank or average increase is equal to 17.50, while the positive rank or the Sum of Rank is 595.00. None of them have lower Pre-Test result. Only 1 participant has no change in his Post-Test Score.

Table 7.9 Test Statistic of Bubble Sorting Result

	BubbleSort_PostTest - Bubble Sorting
Z	-5.095[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Meanwhile, based on the result of the Wilcoxon Signed Rank Test calculation (see Table 7.9), the Z value obtained is -5.095 with p value (**Asymp. Sig 2 tailed**) equal to 0.000 is less than 0.05, so the hypothesis is to accept H1 or that means there is a significant difference between Pre-Test and Post-Test results for Bubble Sorting algorithm.

7.4.2.2 Insertion Sorting Pre-Test and Post-Test Comparison

Table 7.10

Descriptive Statistic of Insertion Sorting Result.

	N	Mean	Std. Deviation	Minimum	Maximum
PreTest_Insertion	35	64.14	18.530	30	100
PostTest_Insertion	35	95.57	7.453	75	100

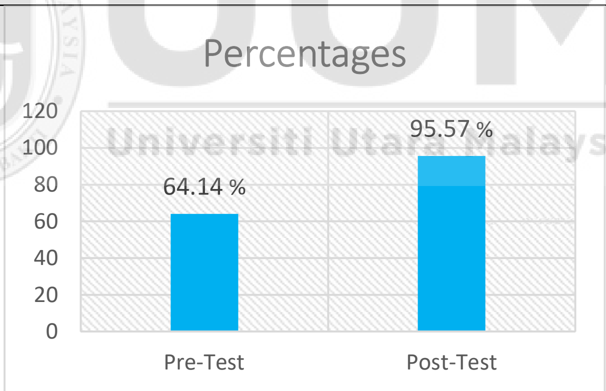


Figure 7.8 Pre-Test and Post-Test of Insertion Sorting Result

Statistically, it appears that the mean posttest score (95.57%) is greater than the pretest score (64.14%) for Insertion sorting (see Table 7.10). The Wilcoxon Signed Rank Test in the next table explains in more detail the magnitude of this difference whether it is statistically significant or not

Table 7.11

Wilcoxon Signed Ranks Test of Insertion Sorting Result

		N	Mean Rank	Sum of Ranks
PostTest_Insertion - PreTest_Insertion	Negative Ranks	0[a]	.00	.00
	Positive Ranks	34[b]	17.50	595.00
	Ties	1[c]		
	Total	35		

a. PostTest_Insertion < PreTest_Insertion

b. PostTest_Insertion > PreTest_Insertion

c. PostTest_Insertion = PreTest_Insertion

The Ranks Table 7.11 provides some data on the comparison of participants' Before (Pre) and After (Post) Insertion sorting Score. The same with Bubble sorting result, based on the table's legend, there are 34 positive data (N) which means that 34 students experience an increase in sorting algorithm learning outcomes from the Pre-Test value to the Post-Test value. Thus, Mean Rank or average increase is equal to 17.50, while the positive rank or the Sum of Rank is 595.00. None of them have lower Pre-Test result. Only 1 participant has no change in his Post-Test Score.

Table 7.12

Test Statistic of Insertion Sorting Result

	PostTest_Insertion - PreTest_Insertion
Z	-5.096[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Based on the results of the Wilcoxon Signed Rank Test calculation (see Table 7.12), the Z value obtained is -5.096 with p value (**Asymp. Sig 2 tailed**) equal to 0.000 is less than 0.05, so the hypothesis is to accept H1 or that means there is a significant difference between Pre-Test and Post-Test results for Insertion Sorting algorithm.

7.4.2.3 Selection Sorting Pre-Test and Post-Test Comparison

Table 7.13

Descriptive Statistic of Selection Sorting Result

	N	Mean	Std. Deviation	Minimum	Maximum
PreTest Selection	35	53.86	24.226	0	90
PostTest Selection Sorting	35	88.43	13.437	60	100

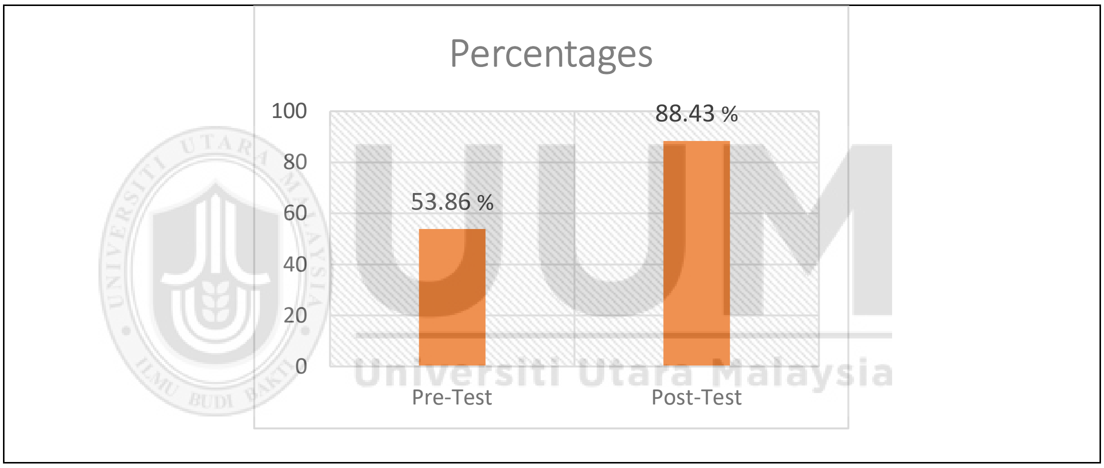


Figure 7.9 Pre-Test and Post-Test of Selection Sorting Result

Statistically, it appears that the mean Post-Test score (88.44%) is greater than the Pre-Test score (53.86%) for Selection sorting result (see Table 7.13). the Wilcoxon Signed Rank Test in the next table explains in more detail the magnitude of this difference whether it is statistically significant or not.

Table 7.14

Wilcoxon Signed Ranks Test of Selection Sorting Result

		N	Mean Rank	Sum of Ranks
PostTest Selection Sorting - PreTest Selection	Negative Ranks	0[a]	.00	.00
	Positive Ranks	35[b]	18.00	630.00
	Ties	0[c]		
	Total	35		

a. PostTest Selection Sorting < PreTest Selection

b. PostTest Selection Sorting > PreTest Selection

c. PostTest Selection Sorting = PreTest Selection

The Ranks Table 7.14 provides some data on the comparison of participants' Before (Pre) and After (Post) for Selection Sorting Score. Based on the table's legend, there are 35 positive data (N) which means that all students experience an increase in sorting algorithm learning outcomes from the Pre-Test value to the Post-Test value. Thus, Mean Rank or average increase is equal to 18.00, while the positive rank or the Sum of Rank is 630.00. None of them have lower Pre-Test results or same result between Pre-Test result and Post-test result.

Table 7.15

Test Statistic of Selection Sorting Result

	PostTest Selection Sorting - PreTest Selection
Z	-5.167[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Based on the results of the Wilcoxon Signed Rank Test calculation (see Table 7.15), the Z value obtained is -5.167 with p value (**Asymp. Sig 2 tailed**) equal to 0.000 is less than 0.05, so the hypothesis is to accept H1 or that means there is a significant difference between Pre-Test and Post-Test results on Selection Sorting algorithm.

7.4.2.4 Summary of All Sorting Pre-Test and Post-Test Comparison

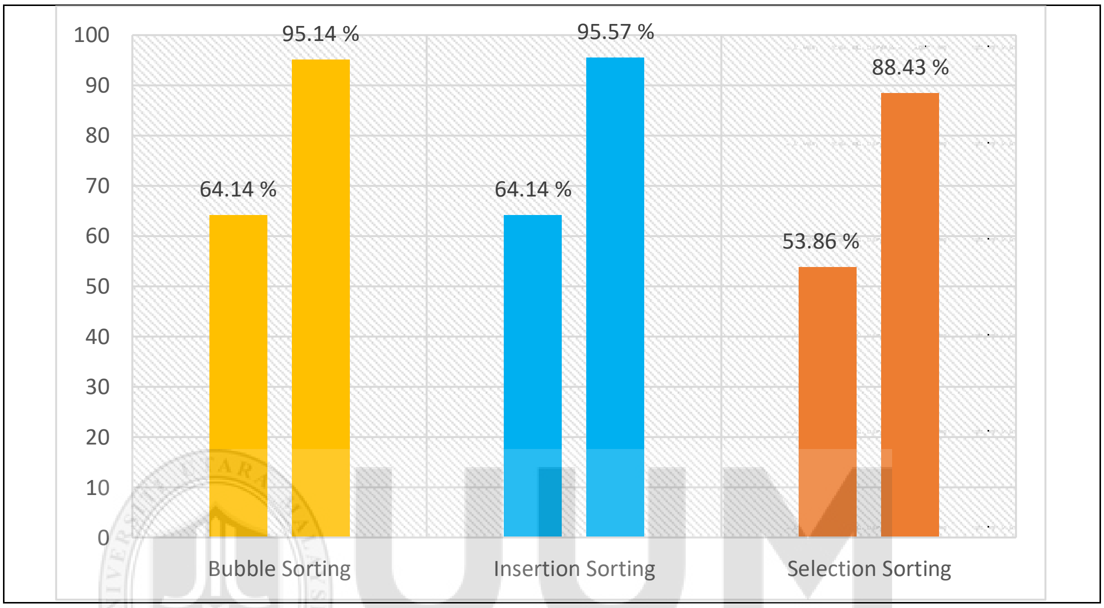


Figure 7.10. Pre-Test and Post-Test of All Sorting Result

The bar chart (see Figure 7.10) demonstrates the differences between Pre-Test and Post-Test results of three sorting algorithms, namely Bubble sorting, Insertion sorting, and Selection sorting. It is measured in percentages. Generally, it can be seen that all Post-Test are increased in all sorting.

The Pre-Test results of Bubble sorting and Insertion sorting have the same value, which is 64.14%, whereas Selection sorting got the smallest value of the Pre-Test, which is 53.86%. Meanwhile, the Post-test results portray Insertion sorting as the highest value, which is 95.57%. The second highest of Post-test is Bubble sorting which has slight difference only compared to Insertion sorting, which is 95.14%. The smallest Post-test value is Selection sorting, which is 88.43%. Nevertheless, Selection sorting has the biggest increment value from Pre-test to Post-test, which is from 53.86% to 88.43%. This has 34.57 difference, and as a result the increment is 64.18%. On the other hand, the increment values for Bubble sorting and Insertion sorting have almost the same values. The Bubble sorting is from 64.14% to 95.14%, which means 31% difference and thus the increment is 48.33%. The Insertion sorting is from 64.14% to 95.57%, which means 34.43% difference, and thus the increment is 49%.

7.4.3 Each Bloom Taxonomy Level Pre-Test and Post-Test Comparison

This section discusses the result of each bloom taxonomy level between Pre-Test and Post-Test. This involves Knowledge Bloom Taxonomy, Analysis Bloom Taxonomy, Understanding Bloom Taxonomy and Apply Bloom Taxonomy. The next sub-sections detail the analysis result using Wilcoxon test.

7.4.3.1 Knowledge Bloom Taxonomy Level

Table 7.16

Descriptive Statistic of Knowledge Bloom Taxonomy Result

	N	Mean	Std. Deviation	Minimum	Maximum
PRETESTknowledge	35	68.57	36.553	0	100
POSTESTknowledge	35	88.57	24.512	0	100

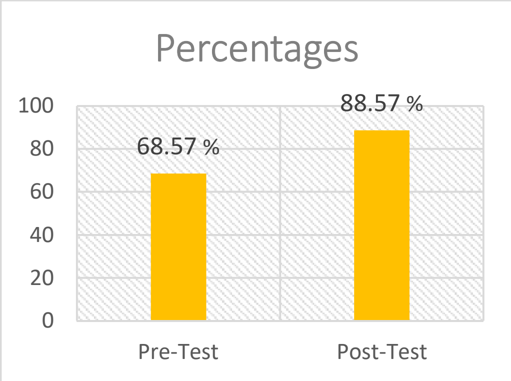


Figure 7.11. Pre-Test and Post-Test of Knowledge Bloom Taxonomy

Statistically, it appears that the mean posttest score is 88.57 which is greater than the pretest value of 68.57 on Knowledge Bloom Taxonomy level (see Table 7.16). the Wilcoxon Signed Rank Test in the next table explains in more detail the magnitude of this difference whether it is statistically significant or not.

Table 7.17

Wilcoxon Signed Ranks Test of Knowledge Bloom Taxonomy Result

		N	Mean Rank	Sum of Ranks
POSTESTknowledge - PRETESTknowledge	Negative Ranks	1[a]	6.50	6.50
	Positive Ranks	13[b]	7.58	98.50
	Ties	21[c]		
	Total	35		

a. POSTESTknowledge < PRETESTknowledge

b. POSTESTknowledge > PRETESTknowledge

c. POSTESTknowledge = PRETESTknowledge

The Ranks Table 7.17 provides some data on the comparison of participants' Before (Pre) and After (Post) for Knowledge Bloom Taxonomy Score. Based on the table's legend, there are 13 positive data (N) which means that 13 students experience an increase in sorting algorithm learning outcomes from the Pre-Test value to the Post-Test value. Thus, Mean Rank or average increase is equal to 7.58, while the positive rank or the Sum of Rank is 98.50. Only 1 student has lower result in Pre-Test result, which has both mean rank and sum of ranks are equal to 6.50. Meanwhile, 21 students have the same result between Pre-Test result and Post-test result.

Table 7.18

Test Statistic of Knowledge Bloom Taxonomy Result

	POSTESTknowledge - PRETESTknowledge
Z	-3.116[b]
Asymp. Sig. (2-tailed)	.002

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Based on the results of the Wilcoxon Signed Rank Test calculation (see Table 7.18), the Z value obtained is -3.116 with p value (**Asymp. Sig 2 tailed**) equal to 0.002 is less than 0.05, so the hypothesis is to accept H1 or that means there is a significant difference between Pre-Test and Post-Test results on Knowledge Bloom Taxonomy level.

7.4.3.2 Analysis Bloom Taxonomy Level

Table 7.19

Descriptive Statistic of Analysis Bloom Taxonomy Result

	N	**Mean**	**Std. Deviation**	**Minimum**	**Maximum**
Analysis Pre-Test	35	59.22	33.832	.00	100.00
Analysis Post-Test	35	85.97	21.983	27.27	100.00

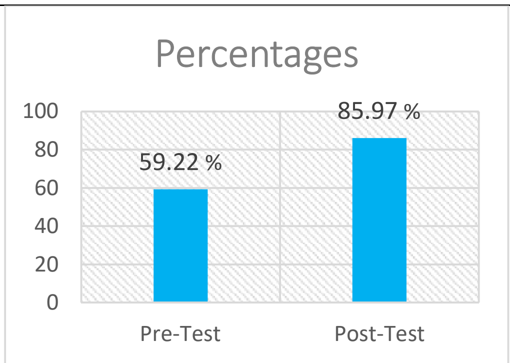


Figure 7.12. Pre-Test and Post-Test of Analysis Bloom Taxonomy

Statistically, it appears that the mean posttest score is 85.97 which is greater than the pretest value of 59.22 on Analysis Bloom Taxonomy level (see Table 7.19). the Wilcoxon Signed Rank Test in the next table explains in more detail the magnitude of this difference whether it is statistically significant or not.

Table 7.20

Wilcoxon Signed Ranks Test of Analysis Bloom Taxonomy Result

		N	Mean Rank	Sum of Ranks
Analysis Post-Test - Analysis Pre-Test	Negative Ranks	4[a]	9.00	36.00
	Positive Ranks	22[b]	14.32	315.00
	Ties	9[c]		
	Total	35		

a. Analysis Post-Test < Analysis Pre-Test

b. Analysis Post-Test > Analysis Pre-Test

c. Analysis Post-Test = Analysis Pre-Test

The Ranks Table 7.20 provides some data on the comparison of participants' Before (Pre) and After (Post) for Analysis Bloom Taxonomy Score. Based on the table's legend, 4 students have lower result in Pre-Test result in which the Mean Rank is 9.00 and the Sum of Ranks is 36.00. However, there are 22 positive data (N) which means that 22 students experience an increase in sorting algorithm learning outcomes from the Pre-Test value to the Post-Test value. Thus, Mean Rank or average increase is equal to 14.32, while the positive rank or the Sum of Rank is 315.00. Meanwhile, 9 students have the same result between Pre-Test result and Post-test result.

Table 7.21

Test Statistic of Analysis Bloom Taxonomy Result

	Analysis Post-Test - Analysis Pre-Test
Z	-3.566[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Based on the results of the Wilcoxon Signed Rank Test calculation (see Table 7.21), the Z value obtained is -3.566 with p value (**Asymp. Sig 2 tailed**) equal to 0.000 is less than the 0.05, so the hypothesis is to accept H1 or that means there is a significant difference between Pre-Test and Post-Test results on Analysis Bloom Taxonomy level.

7.4.3.3 Understand Bloom Taxonomy Level

Table 7.22

Descriptive Statistic of Understand Bloom Taxonomy Result

	N	Mean	Std. Deviation	Minimum	Maximum
Understand Pre-Test	35	66.32	13.995	31.58	84.21
Understand Post-Test	35	97.29	6.303	68.42	100.00

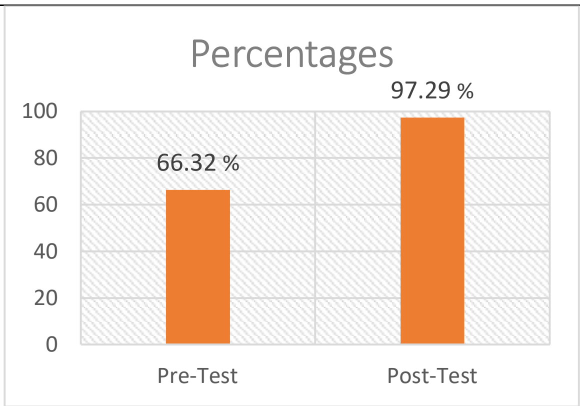


Figure 7.13. Pre-Test and Post-Test of Understand Bloom Taxonomy

Statistically, it appears that the mean posttest score is 97.29 which is greater than the pretest value of 66.32 on Understanding Bloom Taxonomy Level (see Table 7.22). the Wilcoxon Signed Rank Test in the next table explains in more detail the magnitude of this difference whether it is statistically significant or not.

Table 7.23

Wilcoxon Signed Ranks Test of Understand Bloom Taxonomy Result

		N	Mean Rank	Sum of Ranks
Understand Post-Test - Understand Pre-Test	Negative Ranks	0[a]	.00	.00
	Positive Ranks	35[b]	18.00	630.00
	Ties	0[c]		
	Total	35		

a. Understand Post-Test < Understand Pre-Test

b. Understand Post-Test > Understand Pre-Test

c. Understand Post-Test = Understand Pre-Test

The Ranks Table 7.23 provides some data on the comparison of participants' Before (Pre) and After (Post) for Understand Bloom Taxonomy Score. Based on the table's legend, there are 35 positive data (N) which means that all students experience an increase in sorting algorithm learning outcomes from the Pre-Test value to the Post-

Test value. Thus, Mean Rank or average increase is equal to 18.00, while the positive rank or the Sum of Rank is 630.00. Therefore, none of them have lower Pre-Test results or same result between Pre-Test result and Post-test result.

Table 7.24

Test Statistic of Understand Bloom Taxonomy Result

	Understand Post-Test - Understand Pre-Test
Z	-5.177[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Based on the results of the Wilcoxon Signed Rank Test calculation (see Table 7.24), the Z value obtained is -5.177 with p value (**Asymp. Sig 2 tailed**) equal to 0.000 is less 0.05, so the hypothesis is to accept H1 or that means there is a significant difference between Pre-Test and Post-Test results on Understand Bloom Taxonomy level.

7.4.3.4 Apply Bloom Taxonomy Level

Table 7.25

Descriptive Statistic of Apply Bloom Taxonomy Result

	N	Mean	Std. Deviation	Minimum	Maximum
Apply Pre-Test	35	55.00	24.396	.00	100.00
Apply Post-Test	35	94.05	8.532	66.67	100.00

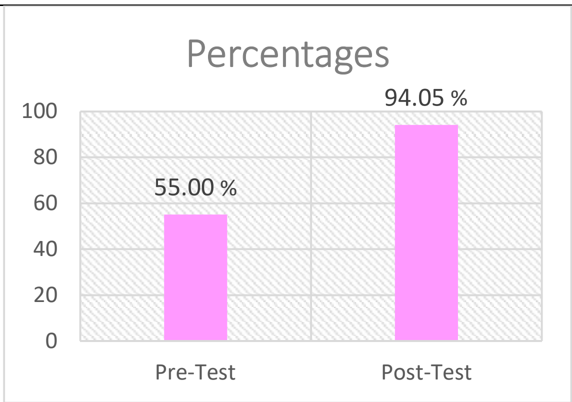


Figure 7.14. Pre-test and post-test of apply bloom taxonomy

Statistically, it appears that the mean posttest score is 94.05 which is greater than the pretest value of 55.00 on Apply Bloom Taxonomy Level (see Table 7.25). the Wilcoxon Signed Rank Test in the next table explains in more detail the magnitude of this difference whether it is statistically significant or not.

Table 7.26

Wilcoxon Signed Ranks Test of Apply Bloom Taxonomy Result

		N	Mean Rank	Sum of Ranks
Apply Post-Test - Apply Pre-Test	Negative Ranks	0[a]	.00	.00
	Positive Ranks	32[b]	16.50	528.00
	Ties	3[c]		
	Total	35		

a. Apply Post-Test < Apply Pre-Test

b. Apply Post-Test > Apply Pre-Test

c. Apply Post-Test = Apply Pre-Test

The Ranks Table 7.26 provides some data on the comparison of participants' Before (Pre) and After (Post) for Apply Bloom Taxonomy Score. Based on the table's legend, there are 32 positive data (N) which means that 32 students experience an increase in sorting algorithm learning outcomes from the Pre-Test value to the Post-

Test value. Thus, Mean Rank or average increase is equal to 16.50, while the positive rank or the Sum of Rank is 528.00. Meanwhile, none of them have lower result on the Pre-Test. However, 3 students have the same results between Pre-Test and Post-Test

Table 7.27

Test Statistic of Apply Bloom Taxonomy Result

	Apply Post-Test - Apply Pre-Test
Z	-4.944^{b}
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Based on the results of the Wilcoxon Signed Rank Test calculation (see table 7.27), the Z value obtained is -4.944 with p value (**Asymp. Sig 2 tailed**) equal to 0.000 is less than 0.05, so the hypothesis is to accept H1 or that means there is a significant difference between Pre-Test and Post-Test results on Apply Bloom Taxonomy level.

7.4.3.5 Summary of Bloom Taxonomy Pre-Test and Post-Test Comparison

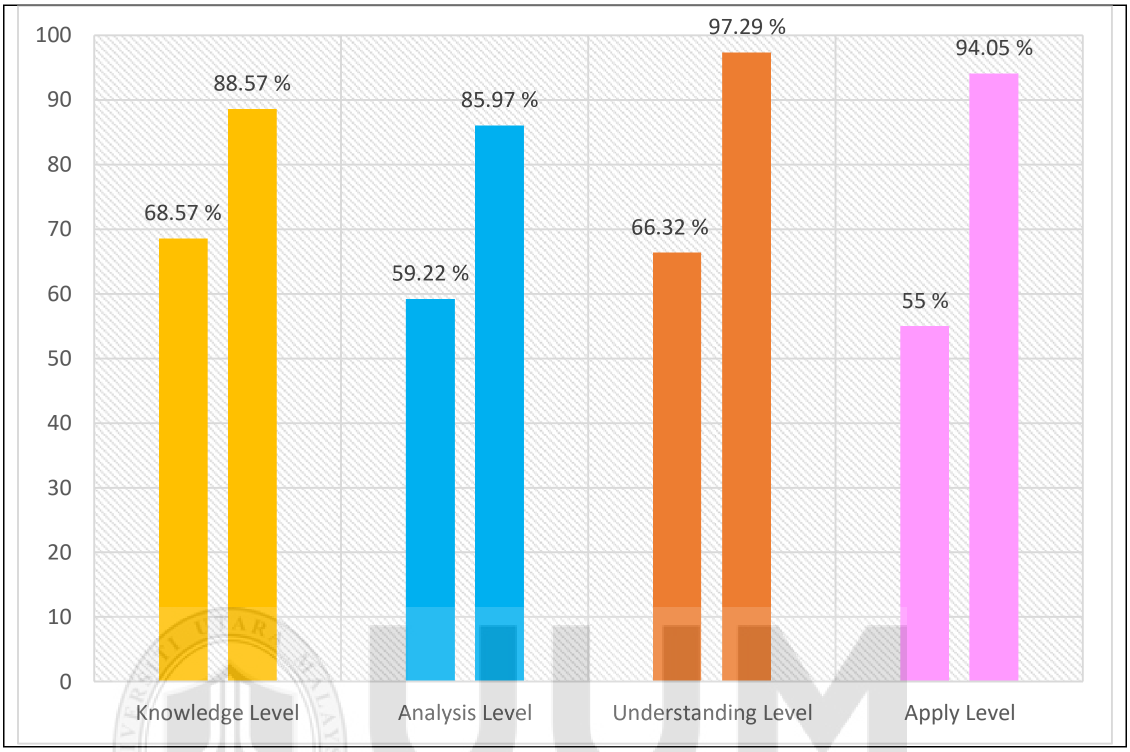


Figure 7.15. Pre-test and post-test of 4 levels of bloom taxonomy results.

The bar chart (see Figure 7.15) demonstrates the differences between Pre-Test and Post-Test results of 4 Bloom Taxonomy levels, namely knowledge level, analysis level, understanding level and apply level. It is measured in percentages. Generally, it can be seen that all Post-Test are increased in all levels.

The knowledge level of Pre-Test shows the highest result, which is 68.57 %, whereas apply level has the lowest score of Pre-Test which is 55 %. Meanwhile, the highest score of Post-Test is understanding level, which is 97.27 % and the lowest score of Post-Test is Analysis level, which is 85.97 %.

However, even though the analysis Pre-Test is the lowest one, but has the biggest increment of all levels, which is from 55% to 94.05%. It has 39.05% difference and as a result the increment is 71%. On the other hand, the knowledge level of Pretest is the highest, but has the smallest increment, which is from 68.57% to 88.57%. It has 20% difference, which means 29.17% increment. Meanwhile, understanding level is the second highest increment, which is from 66.32% to 97.29%. This has 30.97% difference, which means 46.70% increment. The third position is analysis level that has slight difference on its increment compared to understanding level, which is 45.17% increment. It starts from 59.22% to 85.97%, which means 26.75% difference.

7.4.4 Survey Questionnaire Result

As mentioned earlier, after Pre-and-Post tests, the students are directed to grade the statements based on their own opinions. The following figures are the results of it:

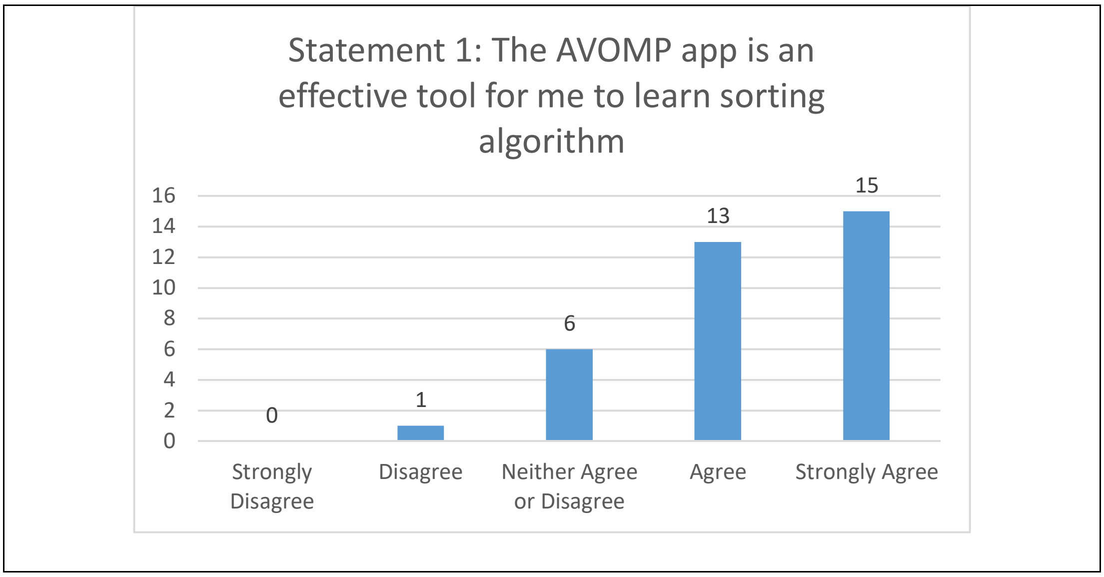


Figure 7.16. Survey statement 1 results.

As seen in Figure 7.16 (statement 1 results), most of students evaluated AVOMP app as an effective tool to learn sorting algorithm, indicated by either agree (13 students) or strongly agree (15 students). Furthermore, the Figure 7.17 (statement 2 results) shows the highest point is at agree scale (18 students) that perceived AVOMP app made sorting algorithm learning easy to learn. 11 students strongly agree on it, whereby the rest are neither agree or disagree (5 students) and disagree is 1 student only.

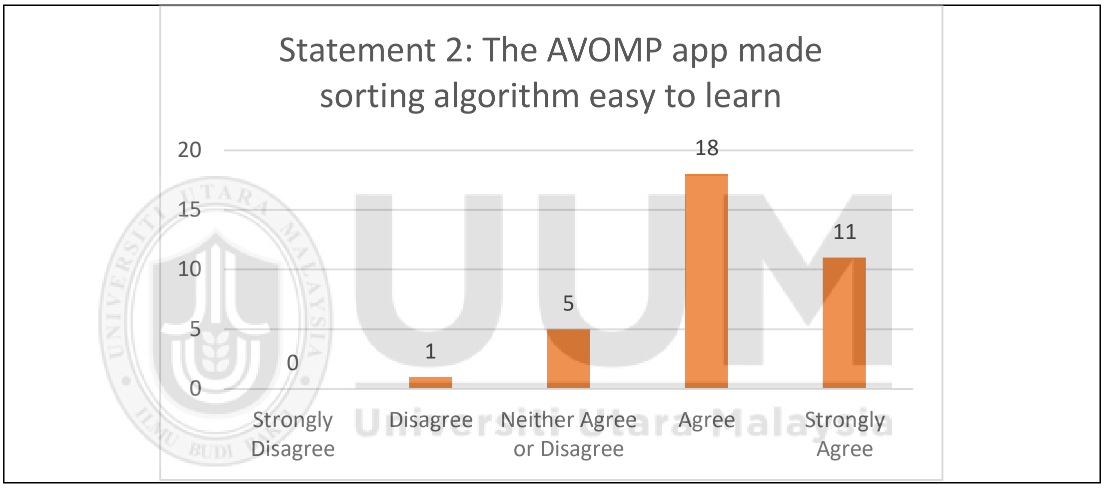


Figure 7.17. Survey statement 2 results.

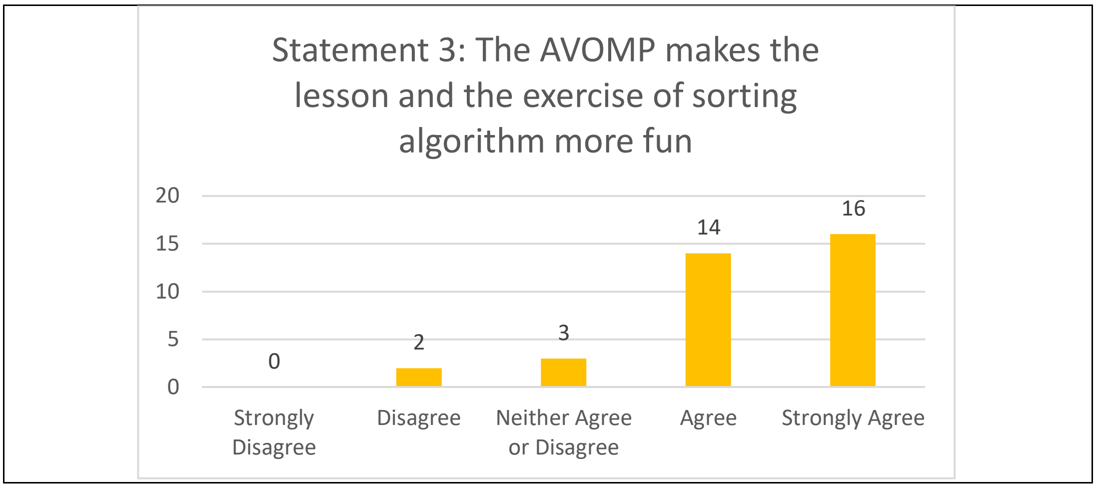


Figure 7.18. Survey statement 3 results.

Figure 7.18 (statement 3 results) still shows the same trend in which most of students either agree (14 students) or strongly agree (16 students) that AVOMP app makes the lesson and the exercise of sorting algorithm more fun. Figure 7.19 (statement 4 results) illustrates that students have the highest point at agree scale (19 students) and the second highest at strongly agree (11 students) towards the statement: step by step process of visualization with highlighted pseudocode line and its explanation help them to understand a sorting algorithm in depth. This supports the proposed guidelines that emphasize the segmentation for continues material (by dividing it into step by step animation supported by explanation) and also give cuing by highlighting the associated pseudocode line will give better insight in learning course material due to reduction of superfluous load and split attention effect.

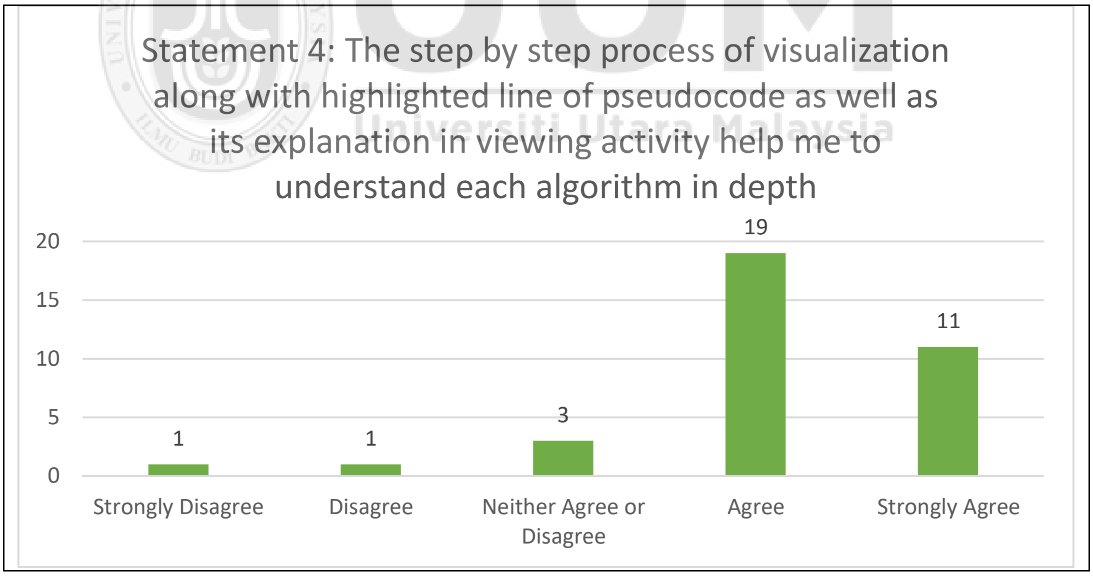


Figure 7.19. Survey statement 4 results.

Figure 7.20 (statement 5 results) demonstrates most of students strongly agree (20 students) towards changing features in AVOMP app which help them to better understand the behavior of each sorting algorithm. This explains the proposed guidelines that the more students view on how the system solves the different numbers (different cases) to be sorted, the more they will understand the behavior of a sorting algorithm.

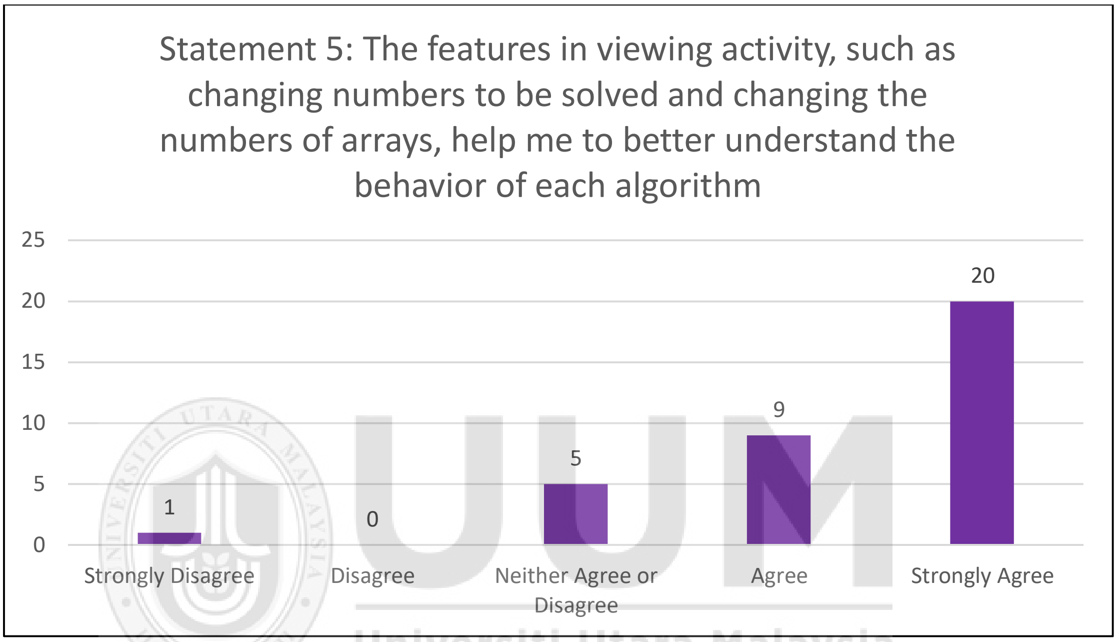


Figure 7.20. Survey statement 5 results.

On the other hand, Figure 7.21 (statement 6 results) interestingly depicts the highest strongly agree point (26 students) among all other statements. Most of students strongly agree that proficiency test helps them to understand sorting algorithms better. This supports the pedagogical approach in the proposed guidelines that the hands-on experience (as the highest interactivity level), inspired by constructivism theory, with the notion learning by doing will construct and strengthen students' knowledge through activities or interactions. Not only that, the AVOMP app also provides reset button which can generate the other random numbers to be solved by students; and hence it gives them more experiences. Additionally, the Figure 7.22

(statement 7 results) demonstrates the hints help students to solve proficiency test, indicating by 18 students agree and 11 students strongly agree on it.

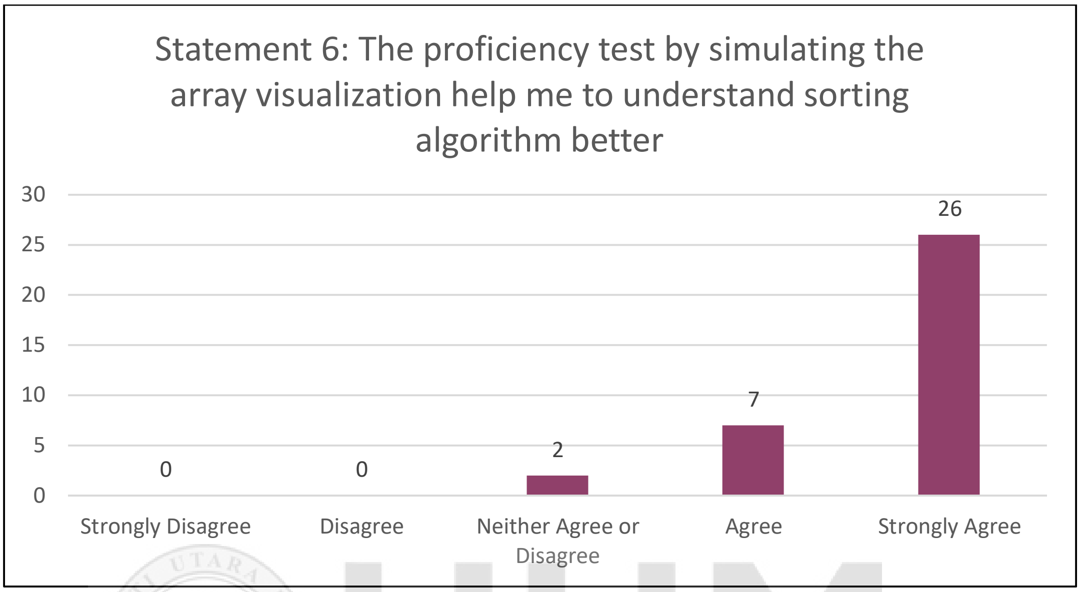


Figure 7.21. Survey statement 6 results

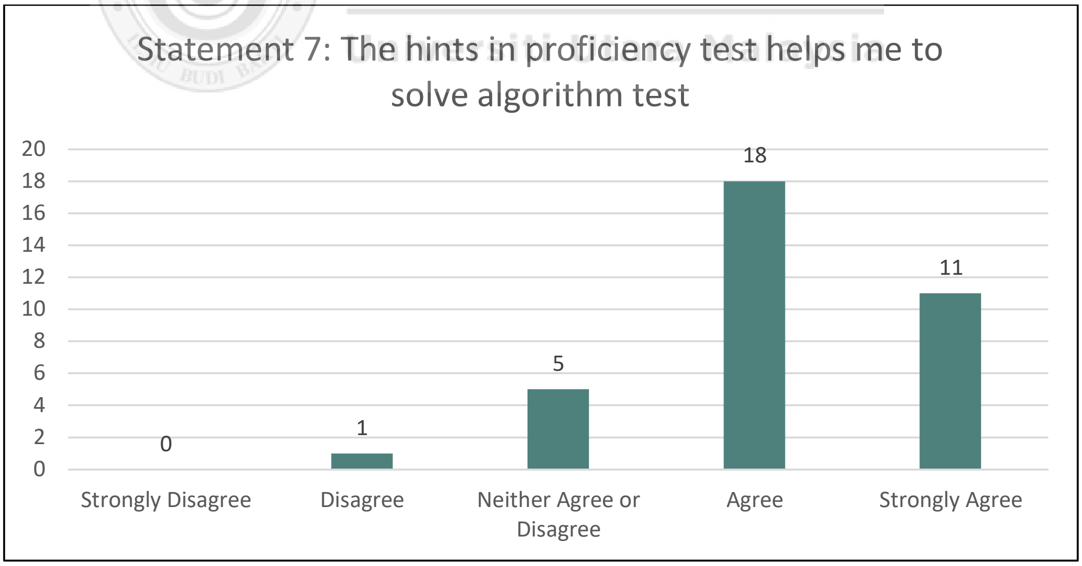


Figure 7.22. Survey statement 7 results

Meanwhile, in the perspective of UI design of AVOMP app, Figure 7.23 (statement 8 results) illustrates that students bolster the notion that an appealing design layout of an app would be able to motivate them using the app, indicating by most of students either strongly agree (14 students) or agree (10 students).

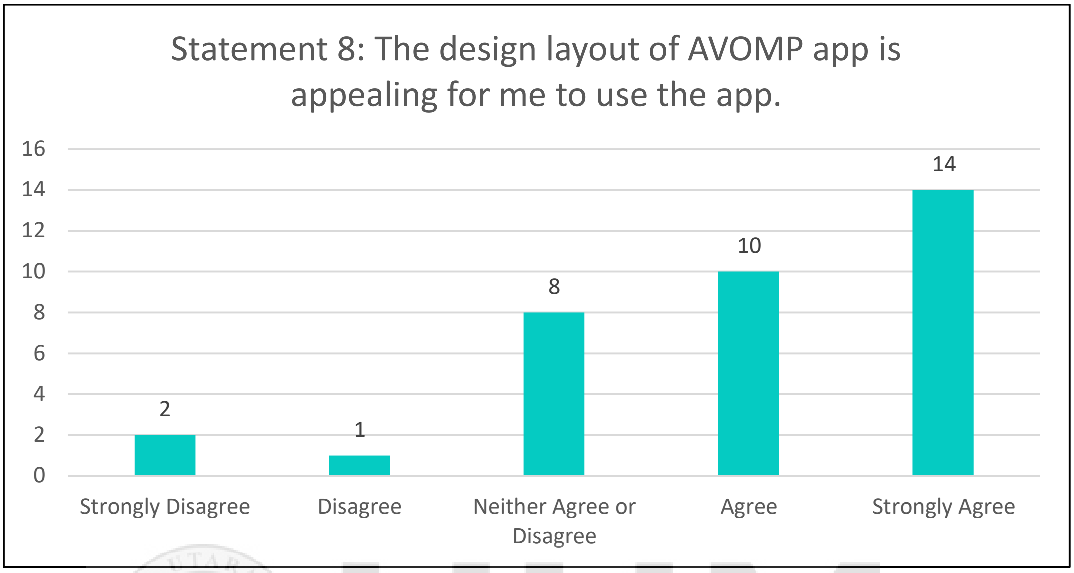


Figure 7.23. Survey statement 8 results.

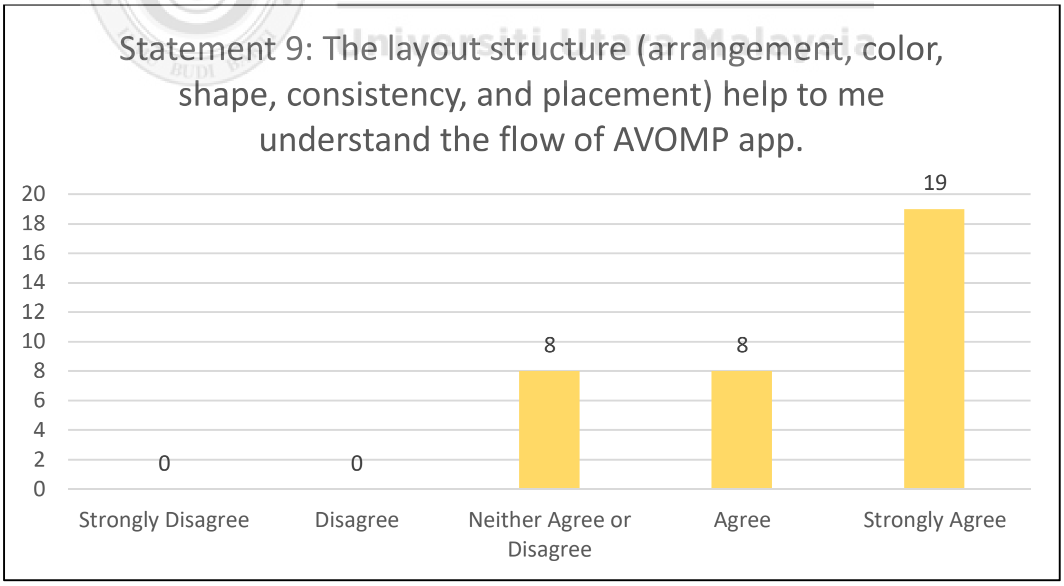


Figure 7.24. Survey statement 9 results.

Still related to UI design, Figure 7.24 (statement 9 results) gives another prove that layout structure is one of the most imperative facets to be taken into account since it effects on how students perceiving the flow and the usage of AVOMP app as a whole. There are 19 students strongly agree about this statement whereas the other students share the same points in agree and neither agree or disagree points, which are 8 students respectively. Finally, the last statement in Figure 7.25 (statement 10 results) demonstrates 18 students agree and 6 students strongly agree towards the textual contents in AVOMP app are clear and understandable. However, there 10 students show neither agree or disagree in this regard and 1 student disagree.

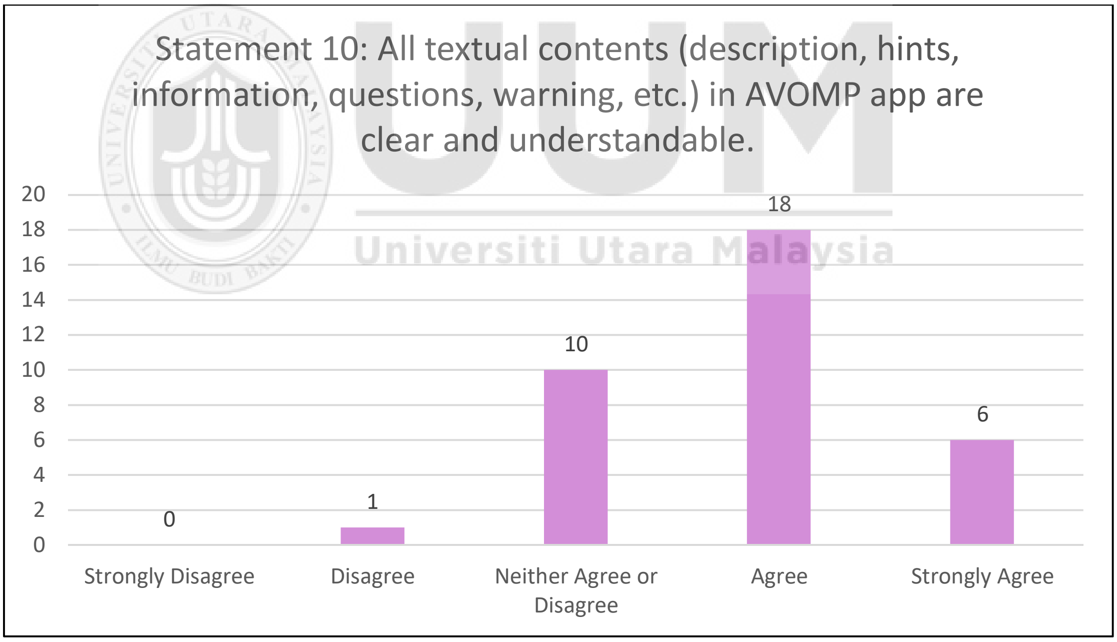


Figure 7.25. Survey statement 10 results

7.4.4.1 Summary of Survey Questionnaire Results

Table 7.28

Summary of Survey Questionnaire Results

		statement 1	statement 2	statement 3	statement 4	statement 5
N	Valid	35	35	35	35	35
	Missing	0	0	0	0	0
Mean		**4.2000**	**4.0857**	**4.2571**	**4.0857**	**4.3429**
Std. Deviation		.83314	.85307	.85209	.88688	.93755
Minimum		2.00	1.00	2.00	1.00	1.00
Maximum		5.00	5.00	5.00	5.00	5.00

		statement 6	statement 7	statement 8	statement 9	statement 10
N	Valid	35	35	35	35	35
	Missing	0	0	0	0	0
Mean		**4.6857**	**4.1143**	**3.9429**	**4.3143**	**3.8286**
Std. Deviation		0.58266	.75815	1.13611	.83213	.74698
Minimum		3	2.00	1.00	3.00	2.00
Maximum		5	5.00	5.00	5.00	5.00

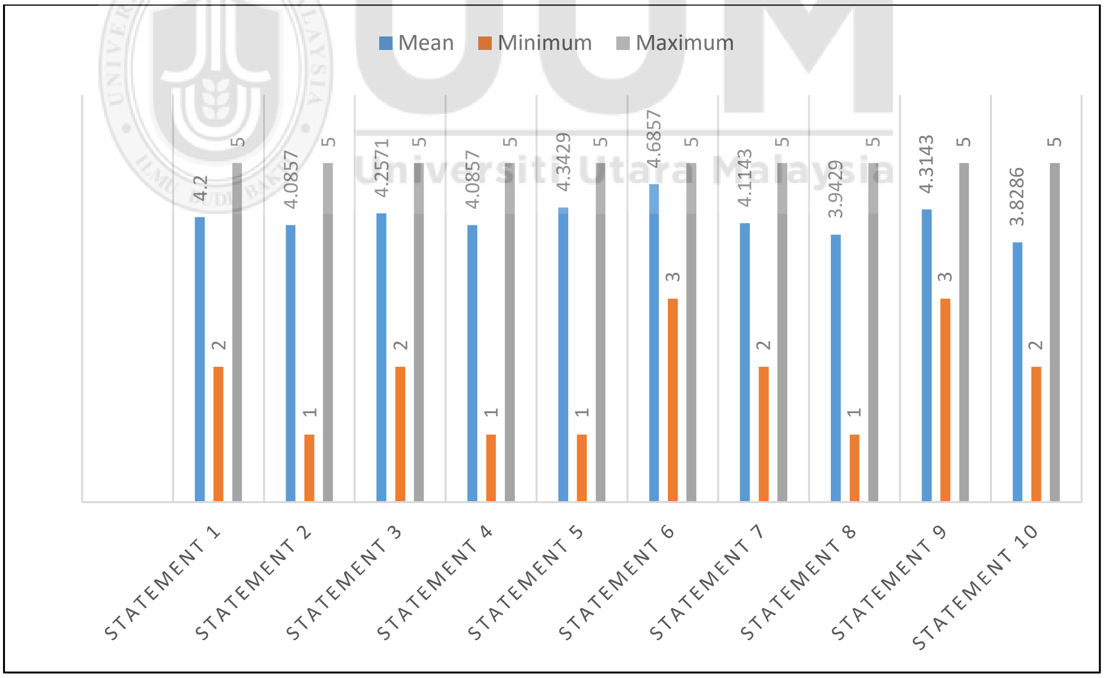


Figure 7.25. Summary of Survey Questionnaire Results

It can be seen from Table 7.28 and Figure 7.25, as far as the effectiveness of the prototype is concerned, statement number 1, 4, 5, and 8 recorded the lowest score of

1 but with a small number of frequencies. In contrast, all ten statements recorded the highest score of 5. As a result, this experiment shows a very impressive mean value of overall survey questionnaire result. The mean value for eight statements that are greater than 4 indicated that the AVOMP is accepted to be an effective tool for students to learn sorting algorithm. Meanwhile, the other two statements (8 and 10) are having slightly different only below 4 points.

7.5 Summary

This chapter reports and concludes that AVOMP application can be a potential tool that can improve students' comprehension in conjunction with three sorting algorithms, namely bubble sorting, insertion sorting and selection sorting based on the controlled experiment (Pre-test and Post-test). Then, several questions are also used to measure students' subjective opinion about AVOMP application. The results show that in general all Post-Test results are increased significantly in the perspective of all sorting algorithms and 4 bloom taxonomy levels. The results survey questionnaire also shows that most of the students are collectively fallen into highly agree and agree Likert scales on all 10 statements provided in which it interprets their subjective opinions towards the effectiveness of the prototype.

CHAPTER EIGHT
CONCLUSION

8.1 Overview

This study was carried out based on these four research questions;

1. What are the appropriate recommendations for UI design and Interactivity of AV on mobile platform?
2. How to construct the design guidelines of AV on mobile platform based on the identified recommendations of UI design and interactivity?
3. How to evaluate the proposed design guidelines of AV on mobile platform?
4. How to measure the effectiveness of the AV on mobile platform that implements the proposed design guidelines through prototype?

Also, this study formed a main aim to meet its expectation, which is to propose a design guidelines of algorithm visualization on mobile platform that could improve students' learning result in learning DSA subject. To accomplish that, four objectives were outlined:

1. To identify the appropriate recommendations for UI design and Interactivity aspects of AVOMP.
2. To develop the design guidelines of AVOMP based on the identified recommendations of UI design and interactivity.
3. To evaluate the proposed design guidelines of AV on mobile platform.

4. To measure the effectiveness of the AV on mobile platform that implements the proposed design guidelines through prototype.

The next sections elaborate the solutions proposed for each research question and the overall conclusions of this study.

8.2 Research Question 1

What are the appropriate recommendations for UI design and Interactivity of AV on mobile platform?

The main purpose of this activity was to determine the appropriate recommendations for UI Design and Interactivity of AV on mobile platform. A content analysis and comparative analysis were carried out to identify them. Firstly, to extract the recommendation of UI design aspect, 11 existing studies pertaining to UI design are compared. They are from the study of mobile design guidelines, previous AV design guidelines, UI design guidelines and usability principle of designing a mobile app. As a result, 8 recommendations of UI design aspect were found and divided into 4 categories, namely content, navigation design, color, and layout design.

Secondly, the comparative analysis is then continued in order to extract the recommendation for interactivity aspect of AVOMP design guidelines. Henceforth, 11 previous studies are compared; and they are from the study of previous AV design guidelines, empirical AV studies (related to engagement taxonomy), and mobile design guidelines. Subsequently, 8 recommendations of interactivity aspect were identified and divided into 5 categories, which are viewing, responding, changing, constructing, control setting and error handling. The summary of those recommendations can be seen in Table 8.1.

Table 8.1

Summary of AVOMP Recommendations

UI Design Aspect	
Category 1: Content	
Recommendation 1	Avoid much content on small screen
Recommendation 2	Avoid Scrolling
Recommendation 3	Use cueing to emphasize the essential information
Recommendation 4	Use multiple views (segmentation)
Category 2: Navigation Design	
Recommendation 5	Provide simple and straightforward hierarchical navigation system
Category 3: Color	
Recommendation 6	Use appropriate color
Category 4: Layout Design	
Recommendation 7	Make layout clear, simple and obvious
Recommendation 8	Group similar elements
Interactivity Aspect	
Category 1: Viewing	
Recommendation 9	Displaying step-by-step execution of an algorithm using control buttons (viewing activity)
Recommendation 10	Provide rewinding button
Recommendation 11	Show dynamic explanation during visualization (pseudocode and textual explanation)
Category 2: Responding	
Recommendation 12	Provide relevant questions to be answered by students
Category 3: Changing	
Recommendation 13	Provide modifying input data for algorithm and also can generate the random values
Category 4: Constructing	
Recommendation 14	AV should provide constructing activity
Recommendation 15	Give feedback during constructing activity
Category 5: Error handling	
Recommendation 16	Prevent and handle error

8.3 Research Question 2

How to construct the design guidelines of AV on mobile platform (AVOMP) based on the identified recommendations of UI design and interactivity?

Based on the identified recommendations of UI design and interactivity from the previous stage, the design guidelines are then constructed. The design guidelines in this study is a research-based which means each guideline is grounded from relevant theory and empirical studies. Therefore, in the construction phase, this study formulated the AVOMP design guidelines consists of 5 components for its each guideline: recommendation, basis (sources, theories), explanation, implementation, and illustration.

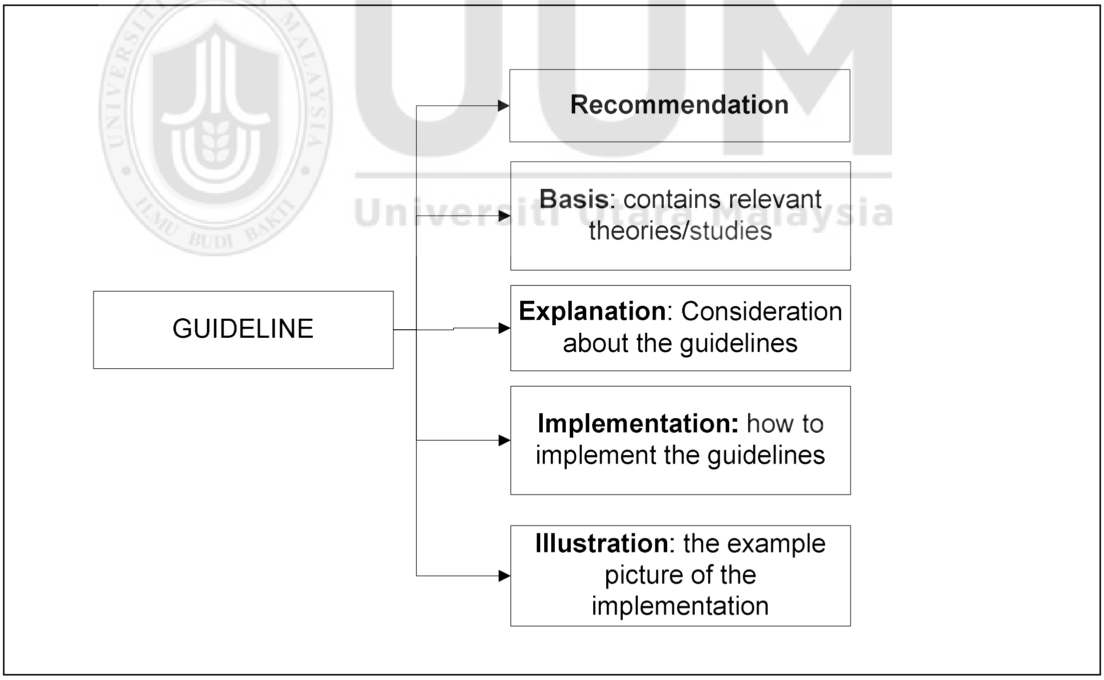


Figure 8.1. AVOMP design guideline components

To fulfil those components, the content analysis towards relevant theories and empirical studies for UI design and interactivity are conducted further in order to justify and strengthen each guideline in this study. The basis component lists out the theories and sources for each guideline. The explanation component describes in detail regarding the guideline, the do and don't, the significance of the recommendation from previous studies, etc. The implementation components list out a set of practical implementations that can be used for AVOMP developers or designers. Last but not least, the illustration component shows the picture to illustrate more the implementation part.

This content analysis has been carried out through comprehensive reading, searching, and reviewing from previous studies and related theories. In detail, it can be seen in Chapter 2, such as Interesting Event approach, Cognitive Load Theory, split attention effect, Usability principles, Constructivism Learning Theory, ET, EET, etc. After that, the guidelines are sent to be reviewed by experts in order to get more useful advices and comments, rewording unclear guidelines, checking its clarity, and connections. Then, based on those comments, the AVOMP design guidelines are revised as a final artifact.

8.4 Research Question 3

How to validate the proposed design guidelines of AVOMP through expert review and prototyping?

In this phase, few activities were conducted. The expert review needs to be conducted to review and validate the proposed design guidelines. Therefore, 20 invitations were sent to the identified experts via email communication. Out of 20 experts, 16 experts with different expertise from academic institutions in Malaysia agree to participate in the study. As for the educational background they are Ph.D.

holders. The reviews and comments of these experts from related fields are important in contributing the improvement of AVOMP design guidelines.

They were given a form to evaluate the design guidelines. Additionally, the experts were also encouraged to write their further comments in the provided instrument. The findings from the expert reviews are discussed in detail in chapter 4. Next, an AVOMP app was developed as the means to validate the design guidelines.

8.5 Research Question 4

Is the AV on mobile platform that implements the proposed design guidelines effective?

The AVOMP design guidelines have been measured for its effectiveness by using a prototype. The prototype is called AVOMP app. There were 35 students of Universiti Utara Malaysia (UUM) participating in this experiment. This was carried out through the controlled experiment whereby pre-test and post-test are involved as a means to measure how far AVOMP application can improve students' comprehension in conjunction with three sorting algorithms, namely bubble sorting, insertion sorting and selection sorting. Additionally, several questions are also given to students to see their subjective opinion about AVOMP application.

The controlled experiments are based on the revised Bloom taxonomy level in 4 levels, which are remember, understand, apply, and analyze. Essentially, this experiment compares the students' performance results on sorting algorithm sub-topic between using manual approach (slides and lecture's note) as Pre-Test and the prototype (AVOMP application) as Post-Test. They were given questions based on the bloom taxonomy levels to be solved in both tests. After the students finish both tests, they are required to fill the last questionnaire survey in order to measure their perspective opinions towards the AVOMP app. This is executed by giving 10 statements to students to be assessed. The overall result shows that AVOMP app is accepted to be an effective tool for students to boost their performance in learning sorting algorithm in DSA topic. The survey questionnaire also shows that most of students either agree or disagree towards the given statements.

8.6 Aim and Objectives: Revisit

The main aim of this study is to propose the design guidelines that can help AV designers, computer science lecturers, tutors and others developing an effective AV system based on two main perspectives, namely, UI design and interactivity that are appropriate on mobile platform. The main aim has been achieved through the completion of the four supporting objectives. The first objective was achieved through the identification of relevant recommendations of UI design and interactivity for AVOMP. The identified recommendations were then classified in several categories. The classification was achieved through rigorous process of content analysis and comparative analysis (see Chapter 4).

The second objective was achieved with the construction of the design guidelines of AVOMP with the complete components (recommendation, basis/source, explanation, and illustration) for each guideline. The conceptual design model was constructed based on findings from series of content analysis and comparative analysis of related theories and empirical studies in UI design and interactivity (see Chapter 5).

The third objective was achieved with the validation of the proposed design guidelines of AVOMP. The validation is made through series of expert reviews (see Chapter 5). The expert reviews are valuable to improve the quality of proposed design guidelines of AVOMP. Finally, the fourth objective was achieved with the evaluation of AVOMP design guidelines in form of prototype to measure its effectiveness through tests (Pre-test and Post-test) based on Bloom taxonomy and survey questionnaire (see Chapter 7).

8.7 Limitations and Recommendations

Based on the carried-out evaluation, there are two areas of limitations this study that may be improved for the next research, which are the design guidelines and AVOMP app prototype.

8.7.1 Design Guidelines of Algorithm Visualization on Mobile Platform

In constructing the AVOMP design guidelines, the recommendations were proposed as the outcomes of various methods implemented throughout the study, which are content analysis and comparatives analysis. A number of AV design guidelines and several collections of studies were based as to derive the common recommendations of AV system. However, the design guidelines used are not exhaustive. The selection also represents the design models for the past ten years (i.e. 2008-2017). Definitely, consideration of more empirical studies of AV system and mobile system could produce more guidelines options.

Furthermore, the evaluation of the design guidelines only focuses on potential AV designers in order to guide them to develop AV system on mobile platform which embed elements that are perceived could boost students' performance in learning DSA subject. Thus, future work could target on the evaluation of the design guidelines for other core subjects, such as object-oriented programming (OOP) and artificial intelligence (AI) subjects.

In addition, due to time constraint, the number of experts and respondents involved were limited. Perhaps with more respondents from different universities may produce other results. In addition, it is highly recommended to have comparative study with the other AV tool in the future. Moreover, this study only attempted to investigate the effectiveness of the proposed design guidelines. Thus, future works could consider measuring each layer in term of their helpfulness or usability.

8.7.2 AVOMP App

The AVOMP app presents three sorting algorithms as DSA sub-topic. The more DSA sub-topics, such as list, stack, queue, etc. could be inserted in the system. The more features of interactivity also ought to be explored. Furthermore, testing the tool to a larger sample and different DSA sub-topics could further enhance the adaptability of the AVOMP design guidelines.

8.8 Conclusion

This study was carried out in a systematic investigation. All relevant recommendations in designing AVOMP design guidelines were considered and validated. From the findings, there are indications that the proposed design guidelines could be utilized to design an effective tool to learn AV on mobile platform.

References

Adamchik, V. (2011). *Data structures and algorithms in pen-based computing environments.* Paper presented at the Global Engineering Education Conference (EDUCON), 2011 IEEE.

Al-Zoubi, A., Alkouz, A., & Otair, M. (2008). Trends and Challenges for Mobile Learning in Jordan. *International Journal of Interactive Mobile Technologies, 2*(2).

Alhosban, F., & Hamad, M. (2011). *The Effectiveness of Aural Instructions with Visualisations in E-Learning Environments.* Durham University.

Ali, A. Z. M., & Derus, S. R. M. (2013). Usability and user satisfaction of hardware-software interfacing visualization kit for novice learning programming. *Procedia-Social and Behavioral Sciences, 103*, 1252-1260.

Amershi, S., Arksey, N., Carenini, G., Conati, C., Mackworth, A., Maclaren, H., & Poole, D. (2005). Fostering Student Learning and Motivation: an interactive educational tool for AI: Citeseer.

Anderson, L. W., Krathwohl, D. R., & Bloom, B. S. (2001). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*: Allyn & Bacon.

Ando, M., & Ueno, M. (2008). *Cognitive load reduction on multimedia e-learning materials.* Paper presented at the Advanced Learning Technologies, 2008. ICALT'08. Eighth IEEE International Conference on.

Ariffin, A. M. (2009). *Conceptual design of reality learning media (RLM) model based on entertaining and fun constructs.* Universiti Utara Malaysia.

Attwell, G. (2012). Work-Based Mobile Learning Environments: Contributing to. *Refining Current Practices in Mobile and Blended Learning: New Applications: New Applications*, 229.

Attwell, G., Guile, D., Angelis, E. d., Scheuermann, F., Frutos, M. B., & Skulimowski, A. M. (2003). The challenge of e-learning in small enterprises: the UK Case Study: Issues for Policy and Practice in Europe.

Avancena, A. T., & Nishihara, A. (2015). Usability and pedagogical assessment of an algorithm learning tool: a case study for an introductory programming course for high school. *Issues in Informing Science & Information Technology, 12*, 21-44.

Ayres, P., & Cierniak, G. (2012). Split-Attention Effect *Encyclopedia of the Sciences of Learning* (pp. 3172-3175): Springer.

Ayres, P., & Paas, F. (2012). Cognitive load theory: New directions and challenges. *Applied Cognitive Psychology, 26*(6), 827-832.

Ayres, P., & Sweller, J. (2014). The split-attention principle in multimedia learning. *The Cambridge handbook of multimedia learning, 2*.

Aziz, N., Mutalib, A. A., & Sarif, S. M. (2017). User experience of interactive assistive courseware for low vision learners (AC4LV): Initial round. *TEM Journal, 6*(3), 488.

Bäck, T., Fogel, D. B., & Michalewicz, Z. (2018). *Evolutionary computation 1: Basic algorithms and operators*: CRC press.

Banerjee, G., Murthy, S., & Iyer, S. (2015). Effect of active learning using program visualization in technology-constrained college classrooms. *Research and Practice in Technology Enhanced Learning, 10*(1), 1.

Bansemir, B. (2013). Research paradigm *Organizational Innovation Communities* (pp. 9-11): Springer.

Basoglu, E. B., & Akdemir, Ö. (2010). A comparison of undergraduate students' English vocabulary learning: Using mobile phones and flash cards. *TOJET: The Turkish Online Journal of Educational Technology, 9*(3).

Benitti, F. B. V., & Sommariva, L. (2015). Evaluation of a game used to teach usability to undergraduate students in computer science. *Journal of Usability Studies, 11*(1), 21-39.

Bhagyalakshmi, H., & Seshachalam, D. (2015). Student performance using Blooms cognition levels: A case study. *Journal of Engineering Education Transformations*, 122-125.

Block, G. (2013). *Reducing Cognitive Load Using Adaptive Uncertainty Visualization.* Nova Southeastern University.

Boticki, I., Barisic, A., Martin, S., & Drljevic, N. (2013). Teaching and learning computer science sorting algorithms with mobile devices: A case study. *Computer Applications in Engineering Education, 21*(S1), E41-E50.

Boy, J., Detienne, F., & Fekete, J.-D. (2015). *Storytelling in Information Visualizations: Does it Engage Users to Explore Data?* Paper presented at the Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems.

Bradley, C., Haynes, R., Cook, J., Boyle, T., & Smith, C. (2009). Design and development of multimedia learning objects for mobile phones. *Mobile learning: Transforming the delivery of education and training*, 157-182.

Brandse, M., & Tomimatsu, K. (2014). *Using color guidance to improve on usability in interactive environments.* Paper presented at the International Conference on Human-Computer Interaction.

Brunken, R., Plass, J. L., & Leutner, D. (2003). Direct measurement of cognitive load in multimedia learning. *Educational Psychologist, 38*(1), 53-61.

Buchanan, G., Farrant, S., Jones, M., Thimbleby, H., Marsden, G., & Pazzani, M. (2001). *Improving mobile internet usability.* Paper presented at the Proceedings of the 10th international conference on World Wide Web.

Burguillo, J. C. (2010). Using game theory and competition-based learning to stimulate student motivation and performance. *Computers & Education, 55*(2), 566-575.

Büyükduman, İ., & Şirin, S. (2010). Learning portfolio (LP) to enhance constructivism and student autonomy. *Procedia-Social and Behavioral Sciences, 3*, 55-61.

Campbell, D. T., & Stanley, J. C. (2015). *Experimental and quasi-experimental designs for research*: Ravenio Books.

Chandran, S. (2010). E-education in multicultural setting: The success of mobile learning. *World Academy of Science, Engineering and Technology, International Journal of Social, Behavioral, Educational, Economic, Business and Industrial Engineering, 4*(10), 2049-2053.

Chang, T.-R., Kaasinen, E., & Kaipainen, K. (2012). *Persuasive design in mobile applications for mental well-being: multidisciplinary expert review.* Paper presented at the International Conference on Wireless Mobile Communication and Healthcare.

Che, P.-C., Lin, H.-Y., Jang, H.-C., Lien, Y.-N., & Tsai, T.-C. (2009). A study of English mobile learning applications at National Chengchi University. *International Journal of Distance Education Technologies (IJDET), 7*(4), 38-60.

Chen, C.-H., & Chien, Y.-H. (2005). Reading Chinese text on a small screen with RSVP. *Displays, 26*(3), 103-108.

Chen, I.-J., & Chang, C.-C. (2011). Content presentation modes in mobile language listening tasks: English proficiency as a moderator. *Computer Assisted Language Learning, 24*(5), 451-470.

Churchill, D. (2011). Conceptual Model Learning Objects and Design Recommendations for Small Screens. *Educational Technology & Society, 14*(1), 203-216.

Churchill, D., & Hedberg, J. (2008). Learning object design considerations for small-screen handheld devices. *Computers & Education, 50*(3), 881-893.

Ciampa, K. (2014). Learning in a mobile age: an investigation of student motivation. *Journal of Computer Assisted Learning, 30*(1), 82-96.

Cierniak, G., Scheiter, K., & Gerjets, P. (2009). Explaining the split-attention effect: Is the reduction of extraneous cognitive load accompanied by an increase in germane cognitive load? *Computers in Human Behavior, 25*(2), 315-324.

Clark, R. C., Nguyen, F., & Sweller, J. (2011). *Efficiency in learning: Evidence-based guidelines to manage cognitive load*: John Wiley & Sons.

Cockburn, A., Ahlström, D., & Gutwin, C. (2012). Understanding performance in touch selections: Tap, drag and radial pointing drag with finger, stylus and mouse. *International Journal of Human-Computer Studies, 70*(3), 218-233.

Cooper, M. L., Shaffer, C. A., Edwards, S. H., & Ponce, S. P. (2014). Open source software and the algorithm visualization community. *Science of Computer Programming, 88*, 82-91.

Cooperstein, S. E., & Kocevar-Weidinger, E. (2004). Beyond active learning: a constructivist approach to learning. *Reference Services Review, 32*(2), 141-148.

Corder, G. W., & Foreman, D. I. (2014). *Nonparametric statistics: A step-by-step approach*: John Wiley & Sons.

Crescente, M. L., & Lee, D. (2011). Critical issues of m-learning: design models, adoption processes, and future trends. *Journal of the Chinese Institute of Industrial Engineers, 28*(2), 111-123.

Crescenzi, P., Malizia, A., Verri, M. C., Díaz, P., & Aedo, I. (2012). Integrating Algorithm Visualization Video into a First-Year Algorithm and Data Structure Course. *Educational Technology & Society, 15*(2), 115-124.

Crompton, H., & Burke, D. (2018). The use of mobile learning in higher education: A systematic review. *Computers & Education, 123*, 53-64.

Cross, N. (2001). Designerly ways of knowing: Design discipline versus design science. *Design issues, 17*(3), 49-55.

Dahlstrom, E., Walker, J., & Dziuban, C. (2013). ECAR study of undergraduate students and information technology: 2013.

del Vado Vírseda, R. (2010). *A visualization tool for tutoring the interactive learning of data structures and algorithmic schemes. NOTE FROM ACM: It has been determined that the author of this article plagiarized the contents from a previously published paper. Therefore ACM has shut off access to this paper.* Paper presented at the Proceedings of the 41st ACM technical symposium on Computer science education.

Delgado-Almonte, M., Andreu, H. B., & Pedraja-Rejas, L. (2010). Information technologies in higher education: Lessons learned in industrial engineering. *Educational Technology & Society, 13*(4), 140-154.

Deller, M., Ebert, A., Bender, M., Agne, S., & Barthel, H. (2007). *Preattentive visualization of information relevance.* Paper presented at the Proceedings of the international workshop on Human-centered multimedia.

Demirbilek, M., & Demirel, S. (2010). Investigating attitudes of adult educators towards educational mobile media and games in eight European countries. *Journal of Information Technology Education, 9*(1), 235-247.

Derus, S. R. M., & Ali, A. Z. M. (2015). *Utilizing program visualization in learning hardware programming: Effects of engagement level.* Paper presented at the Information and Communication Technology Convergence (ICTC), 2015 International Conference on.

Duin, A. H., Anklesaria, F., & Nater, E. (2012). Cultivating Change in the Academy: 50+ Stories from the Digital Frontlines at the University of Minnesota in 2012.

Eliasson, J., Pargmann, T. C., Nouri, J., Spikol, D., & Ramberg, R. (2013a). Mobile devices as support rather than distraction for mobile learners: Evaluating guidelines for design *Innovations in Mobile Educational Technologies and Applications* (pp. 61-76): IGI Global.

Eliasson, J., Pargmann, T. C., Nouri, J., Spikol, D., & Ramberg, R. (2013b). Mobile devices as support rather than distraction for mobile learners: Evaluating guidelines for design. *Innovations in mobile educational technologies and applications*.

Fiorella, L., & Mayer, R. E. (2014). Role of expectations and explanations in learning by teaching. *Contemporary Educational Psychology, 39*(2), 75-85.

Fling, B. (2009). *Mobile Design and Development: Practical concepts and techniques for creating mobile sites and web apps*: " O'Reilly Media, Inc.".

Florax, M., & Ploetzner, R. (2010). What contributes to the split-attention effect? The role of text segmentation, picture labelling, and spatial proximity. *Learning and Instruction, 20*(3), 216-224.

Forehand, M. (2010). Bloom's taxonomy. *Emerging perspectives on learning, teaching, and technology*, 41-47.

Foutsitzis, C. G., & Demetriadis, S. (2013). Scripted collaboration to leverage the impact of algorithm visualization tools in online learning: Results from two small scale studies. *International Journal of e-Collaboration (IJeC), 9*(1), 42-56.

Giordano, J. C., & Carlisle, M. (2006). *Toward a more effective visualization tool to teach novice programmers.* Paper presented at the Proceedings of the 7th conference on Information technology education.

Gong, J., & Tarasewich, P. (2004). *Guidelines for handheld mobile device interface design.* Paper presented at the Proceedings of DSI 2004 Annual Meeting.

Goodrich, M. T., & Tamassia, R. (2008). *Data structures and algorithms in Java*: John Wiley & Sons.

Graham, L. (2008). Gestalt theory in interactive media design. *Journal of Humanities & Social Sciences, 2*(1).

Grissom, S., McNally, M. F., & Naps, T. (2003). *Algorithm visualization in CS education: comparing levels of student engagement.* Paper presented at the Proceedings of the 2003 ACM symposium on Software visualization.

Guberman, S., Maximov, V. V., & Pashintsev, A. (2012). Gestalt and Image Understanding. *Gestalt Theory, 34*(2), 143.

Guney, A., & Al, S. (2012). Effective learning environments in relation to different learning theories. *Procedia-Social and Behavioral Sciences, 46*, 2334-2338.

Hadjerrouit, S. (2005). Constructivism as guiding philosophy for software engineering education. *ACM SIGCSE Bulletin, 37*(4), 45-49.

Halim, S., Koh, Z. C., Loh, V., & Halim, F. (2012). Learning Algorithms with Unified and Interactive Web-Based Visualization. *Olympiads in Informatics, 6*, 53-68.

Hanley, M. (2010). A Media Model for E-Learning Content: Project Lifecycle 4. *E-Learning Curve Blog*.

Hansen, S. R., & Narayanan, N. H. (2013). *On the role of animated analogies in algorithm visualizations.* Paper presented at the Proceedings of the Fourth International Conference of The Learning Sciences.

Haroz, S., Kosara, R., & Franconeri, S. L. (2015). *Isotype visualization: Working memory, performance, and engagement with pictographs.* Paper presented at the Proceedings of the 33rd annual ACM conference on human factors in computing systems.

Hassanabadi, H., Robatjazi, E. S., & Savoji, A. P. (2011). Cognitive consequences of segmentation and modality methods in learning from instructional animations. *Procedia-Social and Behavioral Sciences, 30*, 1481-1487.

Hevner, A. R. (2007). A three cycle view of design science research. *Scandinavian journal of information systems, 19*(2), 4.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly, 28*(1), 75-105.

Hoehle, H., Aljafari, R., & Venkatesh, V. (2016). Leveraging Microsoft′ s mobile usability guidelines: Conceptualizing and developing scales for mobile application usability. *International Journal of Human-Computer Studies, 89*, 35-53.

Holzinger, A. (2005). Usability engineering methods for software developers. *Communications of the ACM, 48*(1), 71-74.

Hong, S., & Kim, S. C. (2011). *Mobile web usability: developing guidelines for mobile web via smart phones.* Paper presented at the International Conference of Design, User Experience, and Usability.

Huber, P. (2015). Inaccurate input on touch devices relating to the fingertip. *Media Informatics Proseminar on" Interactive Surfaces, 31*.

Hübscher-Younger, T., & Narayanan, N. H. (2003). *Dancing hamsters and marble statues: characterizing student visualizations of algorithms.* Paper presented at the Proceedings of the 2003 ACM symposium on Software visualization.

Hujainah, F., Dahlan, H., Al-Haimi, B., Hujainah, A., Al-Bashiri, H., & Abdulgabber, M. A. (2016). New Usability Guidelines with Implementation Ways of Mobile Learning Application based on Mobile Learning Usability Attributes. *Indian Journal of Science and Technology, 9*(37).

Hundhausen, C., & Douglas, S. (2002). A language and system for constructing and presenting low fidelity algorithm visualizations *Software Visualization* (pp. 227-240): Springer.

Hundhausen, C. D., & Brown, J. L. (2008). Designing, visualizing, and discussing algorithms within a CS 1 studio experience: An empirical study. *Computers & Education, 50*(1), 301-326.

Hundhausen, C. D., Douglas, S. A., & Stasko, J. T. (2002). A meta-study of algorithm visualization effectiveness. *Journal of Visual Languages & Computing, 13*(3), 259-290.

Hwang, G.-J., & Wu, P.-H. (2014). Applications, impacts and trends of mobile technology-enhanced learning: a review of 2008–2012 publications in selected SSCI journals. *International Journal of Mobile Learning and Organisation, 8*(2), 83-95.

Hwang, G. J., Shi, Y. R., & Chu, H. C. (2011). A concept map approach to developing collaborative Mindtools for context-aware ubiquitous learning. *British Journal of Educational Technology, 42*(5), 778-789.

Ismail, I., Baharum, H., & Idrus, R. (2010). Simplistic is the ingredient for mobile learning. *International Journal of Interactive Mobile Technologies, 4*(3), 4-8.

Jackson, I. (2008). Gestalt-A Learning Theory for Graphic Design Education. *International Journal of Art & Design Education, 27*(1), 63-69.

Jin, S.-H. (2013). Visual design guidelines for improving learning from dynamic and interactive digital text. *Computers & Education, 63*, 248-258.

Jonathan, F. C., Karnalim, O., & Ayub, M. (2016). *Extending The Effectiveness of Algorithm Visualization with Performance Comparison through Evaluation-integrated Development.* Paper presented at the Seminar Nasional Aplikasi Teknologi Informasi (SNATI).

Juett, J. A. (2016). Using Program Visualization to Illuminate the Notional Machine.

Karam, A. M. (2015). *A comparison of the effects of mobile device display size and orientation, and text segmentation on learning, cognitive load, and user perception in a higher education chemistry course.*

Karavirta, V., Korhonen, A., Malmi, L., & Naps, T. (2010). A comprehensive taxonomy of algorithm animation languages. *Journal of Visual Languages & Computing, 21*(1), 1-22.

Karavirta, V., & Shaffer, C. (2015). Creating Engaging Online Learning Material with the JSAV JavaScript Algorithm Visualization Library.

Karavirta, V., & Shaffer, C. A. (2013). *JSAV: the JavaScript algorithm visualization library.* Paper presented at the Proceedings of the 18th ACM conference on Innovation and technology in computer science education.

Keeney, R. L., & von Winterfeldt, D. (1991). Eliciting probabilities from experts in complex technical problems. *IEEE Transactions on engineering management, 38*(3), 191-201.

Kim, J., Kang, P., & Choi, I. (2014). Pleasure now, meaning later: Temporal dynamics between pleasure and meaning. *Journal of Experimental Social Psychology, 55*, 262-270.

Kölling, M. (2008). Using BlueJ to introduce programming *Reflections on the Teaching of Programming* (pp. 98-115): Springer.

Koole, M. L. (2009). A model for framing mobile learning. *Mobile learning: Transforming the delivery of education and training, 1*(2), 25-47.

Korhonen, A., Laakso, M.-J., & Myller, N. (2009). HOW DOES ALGORITHM VISUALIZATION AFFECT COLLABORATION?

Krebs, M., Lauer, T., Ottmann, T., & Trahasch, S. (2005). *Student-built algorithm visualizations for assessment: flexible generation, feedback and grading.* Paper presented at the ACM SIGCSE Bulletin.

Kukulska-Hulme, A., Pettit, J., Bradley, L., Carvalho, A. A., Herrington, A., Kennedy, D., & Walker, A. (2011). Mature students using mobile devices in life and learning.

Kurby, C. A., & Zacks, J. M. (2008). Segmentation in the perception and memory of events. *Trends in cognitive sciences, 12*(2), 72-79.

Laakso, M.-J., Myller, N., & Korhonen, A. (2009). Comparing Learning Performance of Students Using Algorithm Visualizations Collaboratively on Different Engagement Levels. *Educational Technology & Society, 12*(2), 267-282.

Lantz, B., Andersson, R., & Manfredsson, P. (2016). Preliminary Tests of Normality When Comparing Three Independent Samples. *Journal of Modern Applied Statistical Methods, 15*(2), 11.

Lazaridis, V., Samaras, N., & Sifaleras, A. (2013). An empirical study on factors influencing the effectiveness of algorithm visualization. *Computer Applications in Engineering Education, 21*(3), 410-420.

Leavitt, M. O., & Shneiderman, B. (2006). Research-based web design & usability guidelines. *US Department of Health and Human Services*.

Lee, K. B., & Grice, R. A. (2004). *Developing a new usability testing method for mobile devices.* Paper presented at the Professional Communication Conference, 2004. IPCC 2004. Proceedings. International.

Lee, M.-H., & Rößling, G. (2010). *Integrating categories of algorithm learning objective into algorithm visualization design: a proposal.* Paper presented at the Proceedings of the fifteenth annual conference on Innovation and technology in computer science education.

Lee, M.-H., & Rößling, G. (2011). *Constructivist and Constructionist Approaches to Constructing Algorithm Visualizations: A Proposal.* Paper presented at the 2011 11th IEEE International Conference on Advanced Learning Technologies.

Leppink, J., Paas, F., Van Gog, T., van Der Vleuten, C. P., & Van Merrienboer, J. J. (2014). Effects of pairs of problems and examples on task performance and different types of cognitive load. *Learning and Instruction, 30*, 32-42.

Lister, R., Box, I., Morrison, B., Tenenberg, J., & Westbrook, D. S. (2004). *The dimensions of variation in the teaching of data structures.* Paper presented at the ACM SIGCSE Bulletin.

Liu, M., Geurtz, R., Karam, A., Navarrete, C., & Scordino, R. (2013). Research on mobile learning in adult education. *On the move: Mobile learning for development. Charlotte, NC: Information Age Publishing*.

Liu, T.-C., Lin, Y.-C., Tsai, M.-J., & Paas, F. (2012). Split-attention and redundancy effects on mobile learning in physical environments. *Computers & Education, 58*(1), 172-180.

Liu, T. C., Lin, Y. C., & Paas, F. (2013). Effects of cues and real objects on learning in a mobile device supported environment. *British Journal of Educational Technology, 44*(3), 386-399.

Luong, M. G., & McLaughlin, A. C. (2009). *Bar Graphs and Small Screens: Mitigating Cognitive Load in Mobile Visualizations.* Paper presented at the Proceedings of the Human Factors and Ergonomics Society Annual Meeting.

Mahmoud, A. K. A., & Kamel, A. (2014). The effect of using discovery learning strategy in teaching grammatical rules to first year general secondary student on developing their achievement and metacognitive skills. *International Journal of Innovation and Scientific Research, 5*(2), 146-153.

Maniar, N., Bennett, E., Hand, S., & Allan, G. (2008). The effect of mobile phone screen size on video based learning. *Journal of software, 3*(4), 51-61.

Marx, A., Backes, C., Meese, E., Lenhof, H.-P., & Keller, A. (2016). EDISON-WMW: exact dynamic programing solution of the Wilcoxon–Mann–Whitney test. *Genomics, proteomics & bioinformatics, 14*(1), 55-61.

Mayer, R., & Moreno, R. (2002). Aids to computer-based multimedia learning. Learning and Introduction, 12, 107-119.

Mayer, R. E. (2002). Multimedia learning. *Psychology of Learning and Motivation, 41*, 85-139.

Mayer, R. E. (2009). *Multimedia learning*: Cambridge university press.

Mayer, R. E. (2014). Cognitive theory of multimedia learning. *The Cambridge handbook of multimedia learning, 43*.

Mayer, R. E., Bove, W., Bryman, A., Mars, R., & Tapangco, L. (1996). When less is more: Meaningful learning from visual and verbal summaries of science textbook lessons. *Journal of educational psychology, 88*(1), 64.

Mayer, R. E., & Chandler, P. (2001). When learning is just a click away: Does simple user interaction foster deeper understanding of multimedia messages? *Journal of educational psychology, 93*(2), 390.

Mayer, R. E., & Moreno, R. (2003). Nine ways to reduce cognitive load in multimedia learning. *Educational psychologist, 38*(1), 43-52.

McGee, S., McGee-Tekula, R., Duck, J., White, T., Greenberg, R. I., Dettori, L., . . . Rasmussen, A. (2016). *Does a taste of computing increase Computer Science enrollment?* Paper presented at the Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT), 2016.

Mehlhorn, K. (2013). *Data structures and algorithms 1: Sorting and searching* (Vol. 1): Springer Science & Business Media.

Meolic, R. (2013). *Demonstration of Sorting Algorithms on Mobile Platforms.* Paper presented at the CSEDU.

Meolic, R., & Dogša, T. (2014). A C++ App for Demonstration of Sorting Algorithms on Mobile Platforms. *International Journal of Interactive Mobile Technologies (iJIM), 8*(1), pp. 40-45.

Meyers, S. (2004). The most important design guideline?[user interfaces]. *Software, IEEE, 21*(4), 14-16.

Mirkovic, J., Bryhni, H., & Ruland, C. M. (2011). Designing user friendly mobile application to assist cancer patients in illness management.

Moere, A. V., Tomitsch, M., Wimmer, C., Christoph, B., & Grechenig, T. (2012). Evaluating the effect of style in information visualization. *IEEE transactions on visualization and computer graphics, 18*(12), 2739-2748.

Molina, A. I., Redondo, M. A., Lacave, C., & Ortega, M. (2014). Assessing the effectiveness of new devices for accessing learning materials: An empirical analysis based on eye tracking and learner subjective perception. *Computers in Human Behavior, 31*, 475-490.

Moons, J., & De Backer, C. (2013). The design and pilot evaluation of an interactive learning environment for introductory programming influenced by cognitive load theory and constructivism. *Computers & Education, 60*(1), 368-384.

Moreno, A., & Joy, M. S. (2007). Jeliot 3 in a demanding educational setting. *Electronic Notes in Theoretical Computer Science, 178*, 51-59.

Mostakhdemin-Hosseini, A. (2009a). Analysis of Pedagogical Considerations of M-Learning in Smart Devices. *International Journal of Interactive Mobile Technologies, 3*(4).

Mostakhdemin-Hosseini, A. (2009b). Analysis of Pedagogical Considerations of M-Learning in Smart Devices. *iJIM, 3*(4), 33-34.

MQA, M. Q. A. (2018). *GUIDELINES: MALAYSIAN QUALIFICATION STATEMENT (MQS).*

Mulvey, M. (2015). *Effects of visualization on algorithm comprehension.* The University of Wisconsin-Milwaukee.

Myller, N., Bednarik, R., Sutinen, E., & Ben-Ari, M. (2009). Extending the engagement taxonomy: Software visualization and collaborative learning. *ACM Transactions on Computing Education (TOCE), 9*(1), 7.

Myller, N., Laakso, M., & Korhonen, A. (2007). *Analyzing engagement taxonomy in collaborative algorithm visualization.* Paper presented at the ACM SIGCSE Bulletin.

Najima, D., & Rachida, A. (2008). An Adaptation of E-learning Standards to M-learning. *International Journal of Interactive Mobile Technologies, 2*(3).

Naps, T. L. (2005). Jhavé: Supporting algorithm visualization. *Computer Graphics and Applications, IEEE, 25*(5), 49-55.

Naps, T. L., Rößling, G., Almstrum, V., Dann, W., Fleischer, R., Hundhausen, C., . . . Rodger, S. (2002). *Exploring the role of visualization and engagement in computer science education.* Paper presented at the ACM Sigcse Bulletin.

Nathan, R. J., & Yeow, P. H. (2011). Crucial web usability factors of 36 industries for students: a large-scale empirical study. *Electronic Commerce Research, 11*(2), 151-180.

Neil, T. (2014). *Mobile design pattern gallery: UI patterns for smartphone apps*: " O'Reilly Media, Inc.".

Nielsen, J. (1997). The use and misuse of focus groups. *IEEE software, 14*(1), 94-95.

Nikander, J., Korhonen, A., Seppälä, O., Karavirta, V., Silvasti, P., & Malmi, L. (2004). Visual algorithm simulation exercise system with automatic assessment: TRAKLA2. *Informatics in Education-An International Journal*(Vol 3_2), 267-288.

Nilsson, E. G. (2008). Design guidelines for mobile applications. *SINTEF ICT, Jun*.

Norshuhada, S., & Shahizan, H. (2010). Design Research in Software Development: Constructing and Linking Research Questions, Objectives, Methods and Outcomes: Sintok: Penerbit Universiti Utara Malaysia.

Norsuhada, S., & Shahizan, H. (2010). *Design Research in Software Development: Constructing, Linking Research Questions, Objective, Methods and Outcomes* (U. U. Malaysia Ed.). Sintok: Universiti Utara Malaysia.

Nurulnadwan, A., Ariffin, M., & Mahfuzah, S. S. (2015). Expert review conceptual design and development model of assistive courseware for young low vision (AC4LV) learners. *Int. J. Conceptions Manag. Soc. Sci, 3*(2), 35-39.

O'Malley, C., Vavoula, G., Glew, J., Taylor, J., Sharples, M., Lefrere, P., . . . Waycott, J. (2005). Guidelines for learning/teaching/tutoring in a mobile environment.

Offermann, P., Levina, O., Schönherr, M., & Bub, U. (2009). *Outline of a design science research process.* Paper presented at the Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology.

Orlikowski, W. J., & Iacono, C. S. (2001). Research commentary: Desperately seeking the" it" in it research—a call to theorizing the it artifact. *Information systems research, 12*(2), 121-134.

Osman, W. I., & Elmusharaf, M. M. (2014). Effectiveness of Combining Algorithm and Program Animation: A Case Study with Data Structure Course. *Issues in Informing Science and Information Technology, 11*.

Paas, F., & Sweller, J. (2014). 2 Implications of Cognitive Load Theory for Multimedia Learning. *The Cambridge Handbook of Multimedia Learning*, 27.

Page, T. (2014). Application-based mobile devices in design education. *International Journal of Mobile Learning and Organisation, 8*(2), 96-111.

Park, H. M. (2015). Univariate analysis and normality test using SAS, Stata, and SPSS.

Park, Y. (2011). A pedagogical framework for mobile learning: Categorizing educational applications of mobile technologies into four types. *The International Review of Research in Open and Distributed Learning, 12*(2), 78-102.

Patel, S. (2014). A Literature Review on Tools for Learning Data Structures.

Pathania, U., & Singh, A. (2014). Visualization Tools of Data Structures Algorithms – A Survey. *International Journal of Advanced Research in Computer Science and Software Engineering, 4*(3), 338-341.

Patil, V. H. (2012). *Data Structures Using C++*: Oxford University Press, Inc.

Pea, R. D., & Maldonado, H. (2006). WILD for learning: Interacting through new computing devices anytime, anywhere. *The Cambridge handbook of the learning sciences*, 852-886.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of management information systems, 24*(3), 45-77.

Pimmer, C., Pachler, N., & Attwell, G. (2010). Towards work-based mobile learning: what we can learn from the fields of work-based learning and mobile learning. *International journal of mobile and blended learning, 2*(4), 1-18.

Posavac, E. J. (2015). *Program evaluation: Methods and case studies*: Routledge.

Puteh, M., Adnan, M., Ibrahim, M. H., Noh, N. M., & Che'Ahmad, C. N. (2014). An analysis of comfortable teaching and learning environment: Community response to climate change in school. *Procedia-Social and Behavioral Sciences, 116*, 285-290.

Ragonis, N. (2012). Type of questions–the case of computer science. *Olympiads in Informatics, 6*, 115-132.

Ramli, N., & Habib, A. (2013). A SURVEY OF PROGRAMMING LANGUAGE COURSES IN MALAYSIA'S PUBLIC UNIVERSITIES. *INTED2013 Proceedings*, 4086-4095.

Repokari, L., Saarela, T., & Kurki, I. (2002). *Visual search on a mobile phone display.* Paper presented at the Proceedings of the 2002 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology.

Richardson, R. T., Drexler, T. L., & Delparte, D. M. (2014). Color and contrast in E-Learning design: A review of the literature and recommendations for instructional designers and web developers. *Journal of Online Learning and Teaching, 10*(4), 657.

Rogers, Y., Sharp, H., & Preece, J. (2011). *Interaction design: beyond human-computer interaction*: John Wiley & Sons.

Ross, J., & Gao, J. (2016). Overcoming the language barrier in mobile user interface design: A case study on a mobile health app. *arXiv preprint arXiv:1605.04693*.

Rößling, G., & Freisleben, B. (2002). ANIMAL: A system for supporting multiple roles in algorithm animation. *Journal of Visual Languages & Computing, 13*(3), 341-354.

Rößling, G., & Naps, T. L. (2002). A testbed for pedagogical requirements in algorithm visualizations. *ACM SIGCSE Bulletin, 34*(3), 96-100.

Rubel, C., & Wallace, M. (2013). *Cognitive Load and Online Learning-When is Working Memory Capacity Exceeded?* Paper presented at the World Conference on Educational Multimedia, Hypermedia and Telecommunications.

Sadikan, S. F. N., & Yassin, S. F. M. (2012). Role of Interactive Computer Programming Courseware in Facilitating Teaching and Learning Process Based on Perception of Students in Bangi, Selangor, Malaysia. *Learning, 3*, 0.51.

Sadikan, S. F. N., & Yassin, S. F. M. (2013). A Conceptual Framework for C Programming Learning Through Basic Game Development Courseware. *International Journal of Scientific & Engginering Research, 4*(8).

Saltan, F. (2016). The Impact of Online Algorithm Visualization on ICT Students' Achievements in Introduction to Programming Course. *Journal of Education and Learning, 6*(1), 184.

Sanchez, C. A., & Branaghan, R. J. (2011). Turning to learn: Screen orientation and reasoning with small devices. *Computers in Human Behavior, 27*(2), 793-797.

Sanchez, C. A., & Goolsbee, J. Z. (2010). Character size and reading to remember from small displays. *Computers & Education, 55*(3), 1056-1062.

Sanchez, C. A., & Wiley, J. (2009). To scroll or not to scroll: Scrolling, working memory capacity, and comprehending complex texts. *Human Factors: The Journal of the Human Factors and Ergonomics Society, 51*(5), 730-738.

Savoji, A. P., Hassanabadi, H., & Fasihipour, Z. (2011). The modality effect in learner-paced multimedia learning. *Procedia-Social and Behavioral Sciences, 30*, 1488-1493.

Scheiter, K., Schüler, A., Gerjets, P., Huk, T., & Hesse, F. W. (2014). Extending multimedia research: How do prerequisite knowledge and reading comprehension affect learning from text and pictures. *Computers in Human Behavior, 31*, 73-84.

Schmidt-Weigand, F., Kohnert, A., & Glowalla, U. (2010). A closer look at split visual attention in system-and self-paced instruction in multimedia learning. *Learning and instruction, 20*(2), 100-110.

Schüler, A., Scheiter, K., & Gerjets, P. (2013). Is spoken text always better? Investigating the modality and redundancy effect with longer text presentation. *Computers in Human Behavior, 29*(4), 1590-1601.

Segura, C., Pita, I., del Vado Vírseda, R., Saiz, A. I., & Soler, P. (2008). Interactive Learning of Data Structures and Algorithmic Schemes *Computational Science–ICCS 2008* (pp. 800-809): Springer.

Selby, C. C. (2015). *Relationships: computational thinking, pedagogy of programming, and Bloom's Taxonomy.* Paper presented at the Proceedings of the Workshop in Primary and Secondary Computing Education.

Seong, D. S. K. (2006). *Usability guidelines for designing mobile learning portals.* Paper presented at the Proceedings of the 3rd international conference on Mobile technology, applications & systems.

Seong, D. S. K., & Chee, C. F. (2006). Navigational patterns on usable mobile news portals. *Journal of Internet Technology, 7*(3), 239-245.

Seraj, M., & Wong, C. Y. (2014). Impacts of Different Mobile User Interfaces on Students' Satisfaction for Learning Dijkstra's Shortest Path Algorithm. *iJIM, 8*(4), 24-30.

Shaffer, C. A., Cooper, M., & Edwards, S. H. (2007). *Algorithm visualization: a report on the state of the field.* Paper presented at the ACM SIGCSE Bulletin.

Shaffer, C. A., Cooper, M. L., Alon, A. J. D., Akbar, M., Stewart, M., Ponce, S., & Edwards, S. H. (2010). Algorithm visualization: The state of the field. *ACM Transactions on Computing Education (TOCE), 10*(3), 9.

Shapiro, E. (2003). Behavioral observation of students in schools (BOSS). *Computer Software. San Antonio, TX: Psychological Corporation*.

Sharples, M., Arnedillo-Sánchez, I., Milrad, M., & Vavoula, G. (2009). *Mobile learning*: Springer.

Shitkova, M., Holler, J., Heide, T., Clever, N., & Becker, J. (2015). *Towards Usability Guidelines for Mobile Websites and Applications.* Paper presented at the Wirtschaftsinformatik.

Shneiderman, B. (2010). *Designing the user interface: strategies for effective human-computer interaction*: Pearson Education India.

Shneiderman, B., Plaisant, C., Cohen, M., & Jacobs, S. (2010). Designing the User Interface: Strategies for Effective Human-Computer Interaction, Addison-Wesley Publ. Co. *Reading, MA, USA*.

Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2016). *Designing the user interface: strategies for effective human-computer interaction*: Pearson.

Simon, H. (1996). *The Sciences of Artificial* (3rd edn ed.). Cambridge, MA: MIT Press.

Simonák, S. (2013). Algorithm Visualization Using the VizAlgo Platform. *Acta Electrotechnica et Informatica, 13*(2), 54.

Siti Mahfuzah, S. (2011). *Conceptual Design Model of Computerized Personal-Decision AID (ComPDA).* Universiti Utara Malaysia.

Son, C., Park, S., & Kim, M. (2011). Linear Text vs. Non-Linear Hypertext in Handheld Computers: Effects on Declarative and Structural Knowledge, and Learner Motivation. *Journal of Interactive Learning Research, 22*(2), 241.

Sorva, J., Karavirta, V., & Malmi, L. (2013). A review of generic program visualization systems for introductory programming education. *ACM Transactions on Computing Education (TOCE), 13*(4), 15.

Spanjers, I. A., Gog, T., & Merriënboer, J. J. (2012). Segmentation of worked examples: Effects on cognitive load and learning. *Applied Cognitive Psychology, 26*(3), 352-358.

Spanjers, I. A., Van Gog, T., & van Merriënboer, J. J. (2010). A theoretical analysis of how segmentation of dynamic visualizations optimizes students' learning. *Educational Psychology Review, 22*(4), 411-423.

Stone, M. (2006). Color in information display. *Vis06*.

Stone, M. C. (2004). *Color in information display principles, perception, and models.* Paper presented at the ACM SIGGRAPH 2004 Course Notes.

Su, K.-W., Liu, C.-L., & Lee, C.-C. (2011). A mobile flight case learning system for ATC miscommunications. *Safety science, 49*(10), 1331-1339.

Sung, E., & Mayer, R. E. (2013). Online multimedia learning with mobile devices and desktop computers: An experimental test of Clark's methods-not-media hypothesis. *Computers in Human Behavior, 29*(3), 639-647.

Sutopo, H. (2011). Selection sorting algorithm visualization using flash. *The International Journal of Multimedia & Its Applications (IJMA), 3*(1), 22-35.

Sweller, J. (2011). Cognitive load theory. *The psychology of learning and motivation: Cognition in education, 55*, 37-76.

Syamsul Bahrin, Z. (2011). *Mobile game-based learning (mGBL) engineering model.* Universiti Utara Malaysia.

Terras, M. M., & Ramsay, J. (2012). The five central psychological challenges facing effective mobile learning. *British Journal of Educational Technology, 43*(5), 820-832.

Thompson, E., Luxton-Reilly, A., Whalley, J. L., Hu, M., & Robbins, P. (2008). *Bloom's taxonomy for CS assessment.* Paper presented at the Proceedings of the tenth conference on Australasian computing education-Volume 78.

Tongco, M. D. C. (2007). Purposive sampling as a tool for informant selection.

Traxler, J. (2007). Defining, Discussing and Evaluating Mobile Learning: The moving finger writes and having writ. *The International Review of Research in Open and Distributed Learning, 8*(2).

Traxler, J. (2012). Sustaining mobile learning and its institutions. *Refining Current Practices in Mobile and Blended Learning: New Applications*, 1-9.

Tudoreanu, M. E. (2003). *Designing effective program visualization tools for reducing user's cognitive effort.* Paper presented at the Proceedings of the 2003 ACM symposium on Software visualization.

Urquiza-Fuentes, J., & Velázquez-Iturbide, J. A. (2009a). Pedagogical effectiveness of engagement levels–a survey of successful experiences. *Electronic Notes in Theoretical Computer Science, 224*, 169-178.

Urquiza-Fuentes, J., & Velázquez-Iturbide, J. Á. (2007). An evaluation of the effortless approach to build algorithm animations with WinHIPE. *Electronic Notes in Theoretical Computer Science, 178*, 3-13.

Urquiza-Fuentes, J., & Velázquez-Iturbide, J. Á. (2009b). A survey of successful evaluations of program visualization and algorithm animation systems. *ACM Transactions on Computing Education (TOCE), 9*(2), 9.

Urquiza-Fuentes, J., & Velázquez-Iturbide, J. Á. (2013). Toward the effective use of educational program animations: The roles of student's engagement and topic complexity. *Computers & Education, 67*, 178-192.

Uzunboylu, H., & Ozdamli, F. (2011). Teacher perception for m-learning: scale development and teachers' perceptions. *Journal of Computer Assisted Learning, 27*(6), 544-556.

Valk, J.-H., Rashid, A. T., & Elder, L. (2010). Using mobile phones to improve educational outcomes: An analysis of evidence from Asia. *The International Review of Research in Open and Distributed Learning, 11*(1), 117-140.

Vrachnos, E., & Jimoyiannis, A. (2014). Design and evaluation of a web-based dynamic algorithm visualization environment for novices. *Procedia Computer Science, 27*, 229-239.

Wang, M., & Shen, R. (2012). Message design for mobile learning: Learning theories, human cognition and design principles. *British Journal of Educational Technology, 43*(4), 561-575.

Wang, S.-L., & Wu, C.-Y. (2011). Application of context-aware and personalized recommendation to implement an adaptive ubiquitous learning system. *Expert Systems with applications, 38*(9), 10831-10838.

Ware, C. (2012). *Information visualization: perception for design*: Elsevier.

Wen, G. Z. (2013). *The Algorithm Animation of Genetic Algorithm of Travelling Salesman Problem.* Paper presented at the Applied Mechanics and Materials.

Wilson, R., Shortreed, J., & Landoni, M. (2004). *A study into the usability of e-encyclopaedias.* Paper presented at the Proceedings of the 2004 ACM symposium on Applied Computing.

Wong, A., Leahy, W., Marcus, N., & Sweller, J. (2012). Cognitive load theory, the transient information effect and e-learning. *Learning and Instruction, 22*(6), 449-457.

Wong, L. Program Visualization: Effect Of Viewing Vs. Responding On Student Learning.

Wu, P.-H., Hwang, G.-J., Tsai, C.-C., Chen, Y.-C., & Huang, Y.-M. (2011). A pilot study on conducting mobile learning activities for clinical nursing courses based on the repertory grid approach. *Nurse Education Today, 31*(8), e8-e15.

Yap, B. W., & Sim, C. H. (2011). Comparisons of various types of normality tests. *Journal of Statistical Computation and Simulation, 81*(12), 2141-2155.

Zabel, T. W. (2010). Viability, Advantages and Design Methodologies of M-Learning Delivery. *Online Submission*.

Zhang, K. (2012). *Software Visualization: From Theory to Practice* (Vol. 734): Springer Science & Business Media.

Zollet, R., & Back, A. (2010). Website usability for internet banking.

APPENDIX A

Sample of Invitation Email to Experts

Expert Review for AVOMP Design Guidelines 4 Yahoo/Terkirim

ahmad Affandi <ahmad_affandi123@yahoo.com> 12 Nov 2017 jam 01.06
Kepada: jhy@ukm.edu.my, fadzilah@ukm.edu.my, tsmeriam@ukm.edu.my, ainuddin@um.edu.my, amir@um.edu.my dan 27 lainnya...

***Note: it is highly recommended to download and install the app, keyword: "inavomp" on google play to understand better about the initial prototype of this study (with case study: sorting algorithm),**

Assalamualaikum.. Selamat Sejahtera

Nama saya Ahmad Affandi Supli dan sekarang ini sedang mengikuti pengajian PhD pengkhususan Multimedia di Sekolah Teknologi Multimedia dan Komunikasi (SMMTC),Universiti Utara Malaysia (UUM). Dibawah penyeliaan Prof. Dr. Shuhada Shiratuddin. Sehubungan itu saya berminat dengan kepakaran Prof/Dr. dalam software engineering, multimedia learning, application and data structure algorithm. Saya percaya dengan kepakaran yang Prof./Dr. miliki ianya amat bersesuaian dengan bidang kajian saya.

Oleh itu disini saya berharap Prof/Dr. boleh menjadi pakar rujuk untuk menilai design guidelines yang saya bangunkan dinamakan sebagai "**Design Guidelines of Algorithm Visualization on Mobile Platform (AVOMP)**". Untuk makluman, AVOMP design guidelines adalah design guidelines reka bentuk bagi aplikasi multimedia yang direka khusus untuk belajar dan mengajar **data structure and algorithm (DSA)** subject.
Jika sekiranya Prof./Dr. bersetuju, saya akan menghantar borang persetujuan dan surat pelantikan rasmi daripada pihak UUM diikuti dengan lakaran design guidelines yang dicadangkan bersama-sama dengan instrumen (soal selidik). Saya percaya ia akan mengambil masa hanya kira-kira 20 hingga 30 minit untuk anda menilai design guidelines tersebut.

Persetujuan dari pihak Prof./Dr. amat-amat diharapkan dan segala kerjasama diucapkan berbanyak terima kasih.

Sekian..

Ahmad Affandi Supli (900946)
Sekolah Teknologi Multimedia dan Komunikasi (SMMTC)
Universiti Utara Malaysia (UUM)
0175019375

APPENDIX B

Official Appointment Letter by Dean

AWANG HAD SALLEH
GRADUATE SCHOOL OF ARTS AND SCIENCES
UUM College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM SINTOK
KEDAH DARULAMAN
MALAYSIA

MUAFAKAT KEDAH

UUM/CAS/AHSGS/900946
29 October 2017

Aznoora Osman (Dr.)
Senior Lecturer
Computer Science Department
Faculty of Computer and Mathematical Sciences
UiTM Perlis Branch

Madam,

APPOINTMENT AS EXPERT REVIEWER FOR "ALGORITHM VISUALIZATION ON MOBILE PLATFORM" DESIGN GUIDELINES

With regard to the above, it is my pleasure to appoint you as an expert reviewer for the following PhD candidate:

Student Name:	**Ahmad Affandi Supli**
Matric No:	**900946**
School:	**School of Multimedia Technology and Communication, Universiti Utara Malaysia**
Research Title:	**Design Guidelines of Algorithm Visualization on Mobile Platform**
Supervisor:	**Prof. Dr. Norshuhada Shiratuddin**

Please give your expert opinion by completing the review form. Your cooperation, time and assistance are greatly appreciated.

Thank you.

Sincerely yours,

PROF. DR. NORSHUHADA SHIRATUDDIN
Dekan
Awang Had Salleh Graduate School of Arts and Sciences
Universiti Utara Malaysia

APPENDIX C

Pre-Test Instrument

Bubble Sort

1. (**analysis**) What is the maximum number of comparisons if there are 5 elements in array x?

A. 10
B. 2
C. 5
D. 25

2. (**understand**) Explain this code in plain English (what it does)

```
void bubblesort(Comparable[] A) {
  for (int i=0; i<A.length-1; i++) // Insert i'th record
    for (int j=1; j<A.length-i; j++)
      if (A[j-1].compareTo(A[j]) > 0)
        swap(A, j-1, j);
}
```

3. Sort this with bubble sort? 15,20,10,18,13 (List the output for each Pass) (**apply**)

Insertion Sorting

1. (**Knowledge**) What operation does the Insertion Sort use to move numbers from the unsorted section to the sorted section of the list?

A. Finding the minimum value
B. Swapping
C. Finding out a pivot value
D. None of the above

2. (**Analysis**) Consider the following lists of partially sorted numbers. The double bars represent the sort marker. How many comparisons and swaps are needed to sort the next number. [**1 3 4 8 9 || 5 2**]

- A. 2 comparisons, 3 swaps
- B. 3 comparisons, 2 swaps
- C. 4 comparisons, 3 swaps
- D. 3 comparisons, 4 swaps

3. (**understand**) Explain this code in plain English (what it does)

```
void inssort(Comparable[] A) {
  for (int i=1; i<A.length; i++) // Insert i'th record
    for (int j=i; (j>0) && (A[j].compareTo(A[j-1]) < 0); j--)
      swap(A, j, j-1);
}
```

4. (**apply**)Sort this with Insertion sort? 15,20,10,18,13 (List the output for each Pass)

Selection Sort

1. (**knowledge**) Which one of the following is the first step in a selection sort algorithm (based on question number 3)?

A. The minimum value in the list is found

B. The maximum value in the list is found

C. Adjacent elements are swapped

2. (**analysis**) How many passes/scans will go through a list of 10 elements?
a. A. 11
b. B. 9
c. C. 20
d. D. 25

3. (**understand**) Explain this code in plain English (what it does)

```
void selsort(Comparable[] A) {
  for (int i=0; i<A.length-1; i++) {
    int bigindex = 0;
    for (int j=1; j<A.length-i; j++)
      if (A[j].compareTo(A[bigindex]))
        bigindex = j;
    swap(A, bigindex, A.length-i-1);
  }
}
```


4. (**apply**)Sort this with Selection sort 15,20,10,18,13 (List the output for each Pass)?

APPENDIX D

Post-Test Instrument

Bubble Sorting

1. (**analysis**) What is the max. number of comparisons that can take place when a bubble sort is implemented? Assume there are n elements in the array?

A. (1/2)(n-1)
B. (1/2)n(n-1)
C. (1/4)n(n-1)
D. None of the above

2. (**understand**) Explain this code in plain English (what it does)

```
void bubblesort(Comparable[] A) {
  for (int i=0; i<A.length-1; i++) // Insert i'th record
    for (int j=1; j<A.length-i; j++)
      if (A[j-1].compareTo(A[j]) > 0)
        swap(A, j-1, j);
}
```

3. Sort this with bubble sort? 18,98,70,11,25 (List the output for each Pass) (**apply**)

Insertion Sorting

1. (**Knowledge**) What is the first operation of Insertion Sort in the nested for loop?

A. Swapping
B. Finding the minimum value
C. Finding out a pivot value
D. None of the above

2. (**analysis**) Consider the following lists of partially sorted numbers. The double bars represent the sort marker. How many comparisons and swaps are needed to sort the next number. [**1 3 4 5 8 9 || 2**]

a. A. 5 comparisons, 4 swaps
b. B. 4 comparisons, 5 swaps
c. C. 6 comparisons, 5 swaps
d. D. 5 comparisons, 6 swaps

3. (**understand**) Explain this code in plain English (what it does)

```
void inssort(Comparable[] A) {
  for (int i=1; i<A.length; i++) // Insert i'th record
    for (int j=i; (j>0) && (A[j].compareTo(A[j-1]) < 0); j--)
      swap(A, j, j-1);
}
```

4. (**apply**) Sort this with Insertion sort? 18,98,70,11,25 (List the output for each Pass)

Selection Sorting

1. (**knowledge**) In a selection sort structure, there is/are?
   a. A. Two separate for loops
   b. B. Three for loops, all separate
   c. C. Two for loops, one nested in the other
   d. D. A for loop nested inside a while loop

2. (**analysis**) How many passes (or "scans") will there be through a list being sorted using a selection sort?

   e. A. Array_size*2
   f. B. Array_size+1
   g. C. Array_size-1
   h. D. None of the above

3. (**understand**) Explain this code in plain English (what it does)

```
void selsort(Comparable[] A) {
  for (int i=0; i<A.length-1; i++) {
    int bigindex = 0;
    for (int j=1; j<A.length-i; j++)
      if (A[j].compareTo(A[bigindex]))
        bigindex = j;
    swap(A, bigindex, A.length-i-1);
  }
}
```


4. (**apply**) Sort this with Selection sort? 18,98,70,11,25 (List the output for each Pass)

APPENDIX E

Expert Review Instrument

***Note: the initial prototype of this study can be seen in google play, keyword: "inavomp"**

INSTRUMENTS DEVELOPMENT

AVOMP Design Guidelines evaluation will involve 2 types of instrument. This phase is conducted to achieve 3rd and 4th objectives of this study:

1. To identify the suitable recommendations for AVOMP Design Guidelines.
2. To construct design guidelines of algorithm visualization on mobile platform (AVOMP).
3. To validate AVOMP design guidelines in term of "applicability" through expert review.
4. To measure students' effectiveness of the AVOMP design guidelines through prototype.

Instrument	Purpose	Participant
1. Expert Review Form	To evaluate the **validity** of the AVOMP Design Guidelines in term of **"applicability".**	Experts (practitioners & academicians)
2. Sorting Test	To measure students' effectiveness. (Results learning performance)	IT Students (who enrolled the Data Structure and Algorithm (DSA) subject)

INSTRUMENT FOR EXPERT REVIEW:

DESIGN GUIDELINES OF ALGORITHM VISUALIZATION ON MOBILE PLATFORM (AVOMP).

Dear Prof/Dr./Sir/Mdm

EXPERT REVIEW ON DESIGN GUIDELINES OF ALGORITHM VISUALIZATION ON MOBILE PLATFORM.

My name is Ahmad Affandi Supli and currently doing PHD study in multimedia of studies at University Utara Malaysia (UUM). I am delighted to inform that you have been selected to participate in this research on reason as follow;

- Your qualifications is either in HCI or CS or SE related areas and/or.
- You have been studying/researching/teaching HCI/SE/CS for at least five years.
- You have been teaching one of the following subjects: data structure and analysis (DSA), algorithm, programming, for at least 5 years

My PhD research propose the "**Design Guidelines of Algorithm Visualization on Mobile Platform (AVOMP) "**. These design guidelines are proposed with the aim to assist developers/designers/lecturers by understanding what kind of knowledge they may need in order to achieve a successful design step in developing algorithm visualization applications. Therefore, the objective of this expert review is to:

1. Evaluate the relevant of the proposed recommendations/guidelines
2. Identify and combine duplicate guidelines
3. Identify and resolve guidelines that conflicted with each other
4. Reword unclear guidelines.

The information given will be treated as confidential and only be used as a research purposes which will be anonymously reported in academic publications.

Please feel free to contact me by email (ahmad_affandi123@yahoo.com) for any queries or my supervisor email (shuhada@uum.edu.my).

Your kind cooperation and assistance are highly appreciated.

Instruction:

Please go through the figure of AVOMP design guidelines. Based on your expertise, please provide feedback and suggestion in a space provided. Please refer to **Appendix A** for detail explanations of each terminology.

EXPERT REVIEW FORM

EXPERT REVIEW DETAILS

Name* : ______________________________

Age : ______________________________

Gender : **Male** ☐ **Female** ☐

Organization/ Instituition/ : ______________________________

Experience* : ______________________________ **[years]**

******Compulsory*

ITEMS FOR REVIEW

Based on the proposed AVOMP Design Guidelines (as depicted in the given hand-out), please tick (√) your choice.

Guidelines (G)	Needs very detail explanations	Need some explanations	Is easy to understand
G1			
G2			
G3			
G4			
G5			
G6			
G7			
G8			
G9			
G10			
G11			
G12			
G13			
G14			
G15			
G16			
G17			

***Guideline (G).**

please tick (√) your choice.
Likert importance scale with the anchors set at **’Important (1)’ to ’Very Important (5)**.’

Guidelines (G)	1	2	3	4	5
G1					
G2					
G3					
G4					
G5					
G6					
G7					
G8					
G9					
G10					
G11					
G12					
G13					
G14					
G15					
G16					
G17					

The language and terms in each guideline

Likert scale of easy to understand with the anchors set at **’Strongly disagree(1)’ to ’Strongly agree(5)**.’

Guidelines (G)	1	2	3	4	5
G1					
G2					
G3					
G4					
G5					
G6					
G7					
G8					
G9					
G10					
G11					
G12					
G13					
G14					

G15					
G16					
G17					

1	The design guidelines for representing the problem are relevant.	Yes		No	
2	The design guidelines for representing the solution are relevant.	Yes		No	
3	The connections and flows of all the guidelines are logical	Yes		No	
4	The design guidelines are applicable to the development of algorithm visualization on mobile platform.	Yes		No	
5	Overall, the design guidelines are readable.	Yes		No	
6	Overall, the design guidelines are applicable.	Yes		No	

10. Would you suggest to add any relevant guidelines or additional recommendations in these design guidelines? Please describe your suggestion.

11. Would you suggest to remove any explanation or guideline? Please describe your suggestion.

12. Would you suggest any updates or improvements related to the AVOMP design guidelines in term of combining duplicate guidelines, reword unclear guidelines, implementation, and resolve guidelines that conflicted with each other? Please explain your suggestion?

13. Could the design guidelines be made more applicable in a prototype development? How?

14. Please write any further comments below:
