

**A Comprehensive Study on Application Development Software Systems**

Aaron Don M. Africa, Gerard Ching, Kerwin Go, Rency Evidente, Jason Uy

Department of Electronics and Communications Engineering  
De La Salle University, Manila  
2401 Taft Ave., Malate, Manila 1004,  
Philippines, aaron.africa@dlsu.edu.ph

**ABSTRACT**

The research will discuss the topic of computer software, more specifically the development of applications using offline and online software. The paper will select two sample software as its main topics of discussion, namely Android Studio and MIT's Application Inventor. The paper will also discuss how these applications developing software are used, and how one can optimize his/her experience creating apps with the given software. MIT Application Inventor was chosen because of its user-friendly interface which allows the user to focus on learning the logic of Java. Android Studio is a practical application development program to learn because of the ubiquity of Android operating systems on mobile devices.

**Key words:** Application, Block Programming, Android Studio, MIT App Inventor, Algorithm.

**1. INTRODUCTION**

For this research, the group will attempt to create a manual in which the reader will be able to grasp entry-level concepts for coding and navigating application development software [1]. The group, as college students, have been training and will continue to train, in all facets of computer studies, both hardware, and software. Part of the group's studies in computer software is developing programs to perform increasingly complicated tasks based on what they have learned in their programming subjects. In these programming subjects, the group would be required to use various software specifically designed to read certain coding language and output it as a program. Such a process is called application development, and the programs that allow the user to create other programs are called application development software.

As an example, the mobile app market has become more and more popular mostly thanks to the rapid advancement of computer technology in recent years. And as a consequence, the value of application development as a skill is also on the rise. Despite these advancements, utilizing and navigating application development software can still be intimidating to many people. By conducting this research, the group aims to create a guide on how to effectively utilize application development software. The guide will mainly focus on the application development software Dev C++, but the topics

discussed can apply to other application development software as well, more details will be discussed in the "Methodology" section of the research. The target audience of this research are people who want to learn to program but cannot get past learning the application development software's user interface. The specific audiences that will benefit most from this research shall be discussed in the "Significance of the Study" section of this research.

**2. BACKGROUND OF THE STUDY**

The realm of application developing has quickly been gaining ground since the beginning of the age of the internet. ComboApp, is a solutions agency that "helps startups, mid-sized companies, and established enterprises make their mark in the competitive mobile and digital marketplace (ComboApp, 2018)". The company has recently conducted a survey which highlights key facts about the industry of application development as a whole. The survey encapsulates the multiple aspects of application development. Included in the survey are the app developers' concerns, challenges, and sources of motivation. The survey provides a meta-picture of the world of application development.

There were 3237 participants in the survey. Included are top management personnel, middle management personnel, and the app developers themselves. Majority of the app developers (~74%) choose to work with iOS and Android platforms. When asked for the primary motivator for developing apps, the main reason was for profit (taking up 60% of all responses tallied). Meanwhile, the attempt to attract more customers to buy a certain product and 'brand recognition' ranked second in the list of motivators to develop applications (28%). For seasoned app developers working in companies, the biggest challenge they face is marketing their application and getting it to stand out in the market. Meanwhile, for the greater mass of aspiring app developers, not knowing how to code at all is the biggest roadblock to actually making an application.

With many more computer and app enthusiasts interested in making apps, but not aware of how to, the emergence of app developing websites has sought out to fill in the demand. Applications such as Android Studio and MIT App Maker offers users a wide plethora of options, ranging from visual to

audio effects [2]. Some apps even integrate an intuitive block drag-and-drop system which makes it easy and fun for developers to concretize the app of their dreams [3]. The following sections will outline the functions of both Android Studio and MIT App Developer.

### 3. STATEMENT OF THE PROBLEM

App development is really important in this day, and by having this skill to create applications that will help him/her be more efficient will give a big advantage to that person, but not all people are able to understand coding because of the complicated codes, and algorithms that professional manual show to the user [4]. Not only casual users will have a disadvantage, but also students who can't code properly because of the lack of proper materials and professors who will simplify the code for the students for them to know the logic of it, and as a result, this will limit the potential of the students because the students will also have a hard time app developing and making their innovative idea come to life [5]. Based on an article entitled "The problem of programming and how to fix it." written by Jonathan Edwards (2018), says that without computer scientist to simplify and to make coding accessible to everyone and also making applications that will simplify the process of programming [6]. The group of students believes that anyone can code and create applications to make their lives better with the use of simpler learning materials and also software that will make the process of making applications easier, in which the group will discuss in following sections of the paper.

### 4. SIGNIFICANCE OF THE STUDY

Programming has allowed users to create new interactive web applications, make a positive impact on programming career, open up new areas of opportunity, showcase creativity in problem solving or ideas, develop new video games, enhanced graphics, and resolve technical problems [7]. This study aims to establish an understanding of different types of programming languages, demonstrate how programming works and explain the importance of programming when creating software and applications. Furthermore, the results of the study could be of great benefit to the following:

#### 4.1. Students

The data will provide students some knowledge on how to create an application through learning the difference of program languages and understanding the basic programming concepts. With this, the students will be able to innovate, calculate, create, or analyze processes in a machine, specifically the software and hardware of a computer. This study may also be used as reference data to provide them a background or idea when programming.

#### 4.2. Aspiring application developers

The information presented will enable them to understand individual step to create a working program, learn how to analyze technical problems, and have a certain mindset on

approaching different types of computer language. Furthermore, this study may aid them in how to use software programs effectively for their daily career lives.

### 5. DESCRIPTION OF THE SYSTEM

Android studio is an application creating software which can be downloaded online. It is mainly run using the Java programming language. Similarly, many of the Android software which is available in the market are also coded using Java. Android's API (Application Programming Interface) -- the set of routines and instructions for building software apps--is also heavily run using Java. Although Android applications can also be programmed using C or C++, its developer Google does not recommend it for certain reasons. Meanwhile, the MIT Application Inventor is an online software which allows people to create applications of their own with flexibility and ease of usage. Users do not need any knowledge in any of the traditional programming languages (such as C++, Python, etc.) to be able to create applications.

### 6. METHODOLOGY

The researchers will attempt to create a manual aiming to introduce the basic knowledge and logic of coding to people who are interested in learning to code. For this purpose, the researchers have chosen to use Android Studio and MIT App Inventor as the focus of their discussion.

MIT App Inventor was chosen is because of its "Block Programming" feature, which uses pre-made formats of the syntax used in Java programming the user can simply add, insert, delete as they wish. This is suited for learning the logic of programming because it will put less of a focus on learning the syntax, and more on when and where to use said syntax. This also lessens the possibility of syntax errors to easily identify logic errors [8].

The mobile app market is one of the fastest-growing in the world and Android is one of the most widely used mobile operating systems [9]. For these reasons, the researchers thought it would be beneficial to select Android Studio as a topic of discussion. Additionally, Android Studio uses Java as a programming language, making it fairly easy to use and understand after learning basic syntax and logic.

### 7. THEORETICAL CONSIDERATION

There appear to be no theoretical considerations required for the user to either utilize the software or create the application itself [10]. The only thing a user might be interested to learn is the logic which goes in any programming endeavor. The use of conditional statements, switch statements, and other similar functions are important to create dynamic applications [11]. Without these inputs, a program might be too simple, or it might simply be a no-input application. In any programming language, whether it is block-based or the traditional style of

coding, using logical operators is paramount to optimizing the possibilities of programming [12,13].

## 8. DATA AND RESULTS

### 8.1. Android Studio

Android studio is an application created by Google for app developers to use to create apps for the Android platform. The applications give the developers to customize apps and create algorithms for the applications, and this application can also create simulations of the app that can help the developer on how the app will work and behave in the platform [14,15].

To operate the Android studio, the developer/user should be able to download the app from the site: "<https://developer.android.com/studio>". This site will the user select the version that the computer can handle. After this step, the user must install and agree to all the policies that the application will ask the user [16]. Then, the application will take a while to compile and unzip all the files to be able to run it successfully. After that, the application will open and will ask the developer or programmer on what type of layout that the developer wants to use [17]. After choosing the layout that the developer will use, the system will redirect the developer to a window, where the user can input the project name, package name, and save location. The developer should also choose the language that they want to use, this can be Kotlin or Java. After filling up all the necessary questions and blanks [18]. The program will redirect the user to the programming window where all the codes are at. The Java or Kotlin codes which will contain the plug-ins and widgets will be located at the "mainactivity.java" page, and all the physical designs and interface will be in the "mainactivity.xml" file. For the designs of the fonts, background, and etc, these codes can be found in the values folder that can be found in the right side of the app [19,20].

After coding all the plug-ins and commands that you want to program to have, click the "run app" button at the top, and the program will ask you to add a device where the user wants to test the app in, and if the user doesn't have device, the app can simulate the app running on a real smartphone [21]. The application will notify the programmer as its building the app if there is a problem in the code because it can't run the code if it has a mistake in it. After that the programmer/developer can see application work in a phone and can also check if there are any mistakes in it that the developer doesn't want. The developer can run the build repeatedly because the app will just keep on updating the file in the developer's test phone [22].

### 8.2. MIT App Developer

To start, one would need to go to the link, "[www.appinventor.mit.edu/hour-of-code.html](http://www.appinventor.mit.edu/hour-of-code.html)". After clicking the CREATE button, the MIT App Developer site will prompt users to create an account, which can be linked to your google account. First time 'inventors' would not have any projects shown on the menu page, so one would need to

press the 'new project' button to create a new project. When naming a project, it is important to note that MIT App Inventor does not permit spaces, but characters can be separated by special characters, such as underscores [23,24].

Next, the user will be prompted to the Display Window, where the application's display interface can be modified and beautified. Textboxes, images, labels, sliders, and list displays are just some of the visual objects which can be added by the programmer. MIT App Inventor uses a drag and drop interface. If one wants to add images, for instance, then he/she would need to drag the image tab to the main screen. The main screen resembles the screen of a mobile phone, and for good reason. The size of the image on the main screen is proportionate to its size when the app is running on an actual mobile phone. The user can also program his/her display blocks using the programming section. This section is where users can hard code commands which go with the displays on their apps. Unlike conventional app developing processes, the commands here are all code blocks. Instead of having to know the appropriate syntax which goes to every command, loop, or conditional statement, one can simply drag and drop these blocks [25].

MIT has a function called 'AI companion' which allows users to connect their created apps to their mobile phones. One would first need to load the MIT Inventor App on the Google Play store. Once the app is loaded, the user can use AI companion to load a 6 character code and a QR code on the computer for the mobile device to either scan the code or input the 6 character code. After that, the user can simulate the app on his/her own device, and possibly on a friend's device as well [26].

## 9. ANALYSIS OF DATA

### 9.1. Android Studio

Android studio is software where it uses xml and java to create an app. First, xml or Extensive mark-up language is the language used to make the design of the app, and in it is all the codes that are related to the color of the widgets and also the background of the app (Whatis.com). Second, Java and Kotlin, Java is the high-level language that is used by the programmer/developer to create commands, widgets, and algorithms that the application needs to run on [27]. Java is a really important part of app developing because, without it, the application won't run as well as the compiler. Kotlin, on the other hand, is another programming language that is created by JetBrains, the language is compatible in most platforms, and it works seamlessly besides Java and Android (Programiz.com.). Android studio is an application where the programmer has the potential to create any type of application without any limits, and that with it, the developer can create unlimited type of applications that can run on Android smoothly because of the emulators that have, and also the flexibility to use Kotlin or Java, and XML to create innovative applications [28]. The application is constantly updated, bringing the developers more options on customizing their

layout, as well as, creating more innovative interfaces for the users to use.

## 9.2. MIT App Inventor

MIT App Inventor is a cloud-based website which allows users to utilize a block programming language to create their own applications. Aspiring app inventors need not know or have a background in programming languages like C, C++, Python, Java, and the like [29,30]. The App Inventor offers a drag and drop interface for both the display and programming phases of the app. Therefore, what app developers have in mind, they can realize in a very easy and efficient manner. According to the Massachusetts Institute of Technology (or MIT), the very creators of this app inventor, "...it is an intuitive, visual programming environment that allows everyone – even children – to build fully functional apps for smartphones and tablets in less than 30 minutes..." First-time users will not find it difficult to learn the software, as there are multiple Youtube tutorials online to teach them the ropes of app making and using the Inventor. This system can also be integrated in PLC [31,32]. A thing that should be considered is the database configuration. It should pattern the studies of [33,34,35,36]. The program structure of the system should pattern the studies of [37,38,39].

## 10. CONCLUSION

In conclusion, this research covered some concepts in programming languages, the basic idea of what programming is, the process of translating a program into machine language, the importance of learning computer programs, and how computer languages work. The researchers have introduced a number of applications that could help students or aspiring application developers to create numerous apps. In addition, the researchers have also emphasized that with programming languages, users will be able to manipulate data stored in files and manipulated data held in RAM. Hence, users can perform a variety of data processing tasks.

## 11. RECOMMENDATIONS

The researchers recommend users to be familiar with relational operators, logical operators, and other staple concepts related to any programming language. This is vital for being able to create robust and flexible apps, whatever their function may be.

The researchers also recommend the users take time to practice what they have learned in coding to form a deeper understanding and increase its retention time. With this, it would help the user to retain the knowledge they have learned from this research and know what to do when encountering certain problems and specialize in certain topics.

As a final recommendation, the researchers recommend the users to also study other programming languages to increase their ability to express ideas in the form of performing different types of languages and not limit themselves from creating programs that do not support programming languages

used as examples in this manual. Furthermore, this would improve their background on choosing languages that would fit for the program, improve their skills in order to quickly adapt to industry, take new opportunities and become a versatile developer.

## REFERENCES

- [1] J. Bucanek, "Boom! App," Learn iOS App Development, pp. 17–55, 2013.  
[https://doi.org/10.1007/978-1-4302-5063-0\\_2](https://doi.org/10.1007/978-1-4302-5063-0_2)
- [2] J. Bucanek, "Learn iOS App Development," 2013.  
<https://doi.org/10.1007/978-1-4302-5063-0>
- [3] W. Ellis and E. Lodi, "The Decision Block," Structured Programming Using Turbo BASIC, pp. 75–92, 1988.  
<https://doi.org/10.1016/B978-0-12-237459-3.50009-7>
- [4] T. Hagos, "Android Studio Introduction and Setup," Learn Android Studio 3 with Kotlin, pp. 157–171, 2018.  
[https://doi.org/10.1007/978-1-4842-3907-0\\_8](https://doi.org/10.1007/978-1-4842-3907-0_8)
- [5] T. Hagos, "Android Studio," Learn Android Studio 3, pp. 5–17, 2018.
- [6] D. Sheppard, "Turning a Real App into a PWA," Beginning Progressive Web App Development, pp. 175–208, 2017.
- [7] M. Sherman and F. Martin, "Learning analytics for the assessment of interaction with App Inventor," 2015 IEEE Blocks and Beyond Workshop (Blocks and Beyond), 2015.
- [8] M. Sherman, "User models of reasoning and understanding in app inventor," Proceedings of the tenth annual conference on International computing education research - ICER 14, 2014.  
<https://doi.org/10.1145/2632320.2632340>
- [9] A. Bakir, "Welcome to iOS Media App Development," Beginning iOS Media App Development, pp. 1–22, 2014.
- [10] D.D.N. Abinoja and L.Y. Materum, "BIC-based optimization of the identification of multipath propagation clusters in MIMO wireless systems," ISAP 2016 - International Symposium on Antennas and Propagation. pp. 428-429, 2016.
- [11] P. Hanpinitzak, K. Saito, J. Takada, M. Kim, and L. Materum, "Clustering method based on scatterer locations for indoor dynamic MIMO channel," 2016 10th European Conference on Antennas and Propagation. 2016.  
<https://doi.org/10.30534/ijatcse/2019/32832019>
- [12] M.R.M. Bailon and L. Materum, "International roaming services optimization using private blockchain and smart contracts," International Journal of Advanced Trends in Computer Science and Engineering. Vol. 8, No. 3, pp. 544-550, 2019.
- [13] A. Celebre, A. Dubouzet, I. Medina, A. Surposa and RC. Gustilo, "Home automation using raspberry Pi through Siri enabled mobile devices," 8th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management, HNICEM 2015, 2015.
- [14] N. Yenikalaycı, D. Çelikler, and Z. Aksan, "The Development of Mit App Inventor Mobile Applications

- Related To The Display Of Molecule Geometries With Vsepr Theory,” INTED2018 Proceedings, 2018.
- [15] “Networking,” Beginning Android® Programming with Android Studio, pp. 351–379, 2017.
- [16] “Foundation Android,” Wearable Android™, pp. 35–35, 2015.
- [17] “Activities, Fragments, and Intents,” Beginning Android® Programming with Android Studio, pp. 47–99, 2017.
- [18] F. M. F. Matter, “Front Matter,” E-Journal Menara Perkebunan, vol. 85, no. 1, 2017.
- [19] N. Kuh, “Provisioning Our App for Development,” Foundation iPhone App Development, pp. 21–39, 2012.
- [20] “Android Application Development with Android Studio,” Expert Android® Studio, pp. 45–79, 2017.  
<https://doi.org/10.1002/9781119419310.ch3>
- [21] K. Roy, “App inventor for android,” Proceedings of the 43rd ACM technical symposium on Computer Science Education - SIGCSE 12, 2012.
- [22] “Using Android NDK with Android Studio,” Expert Android® Studio, pp. 309–330, 2017.
- [23] K. Perdikuri, “Students Experiences from the use of MIT App Inventor in classroom,” Proceedings of the 18th Panhellenic Conference on Informatics - PCI 14, 2014.
- [24] B. Mackellar, “App inventor for android in a healthcare IT course,” Proceedings of the 13th annual conference on Information technology education - SIGITE 12, 2012.  
<https://doi.org/10.1145/2380552.2380621>
- [25] P. C. Hutchinson, Extensions to a block-structured programming language to support processing of symbolic data and dynamic arrays. Ann Arbor, MI: University Microfilms International, 1982.
- [26] A. Gerber and C. Craig, “Navigating Android Studio,” Learn Android Studio, pp. 27–44, 2015.
- [27] “Android SDK,” Wearable Android™, pp. 87–109, 2015.
- [28] W. Jackson, “Exploring Android App Development: The Lingo of Android and Building Your First Hello World App!” Learn Android App Development, pp. 31–54, 2013.  
[https://doi.org/10.1007/978-1-4302-5747-9\\_2](https://doi.org/10.1007/978-1-4302-5747-9_2)
- [29] N. Kuh, “iPhone App Navigation, Views, and View Controllers,” Foundation iPhone App Development, pp. 111–142, 2012.
- [30] N.J.C. Libatique, G.L. Tangonan, R.C. Gustilo, W.K.G. Seah, C. Pineda, M.L. Guico, G. Abrajano, R. Ching, J.L. Zamora, A. Espinosa, A.C. Valera, R. Lamac, H. Dy, J. Pusta, E.M. Trono, A. Gimpaya, J.R.S. Luis, S.J. Gonzales, and A.T. Lotho, “Design of a tropical rain - Disaster alarm system: A new approach based on wireless sensor networks and acoustic rain rate measurements,” IEEE Instrumentation and Measurement Technology Conference. pp. 1341-1345, 2009.
- [31] L. Materum, “Stochastic tapped delay line based one-sided beamformed channel impulse response models of LoS and reflected waves at 62.5 GHz in a conference room environment,” Journal of Telecommunication, Electronic and Computer Engineering. Vol. 9, Nos. 1-5, pp. 33-38, 2017.
- [32] A. Dulay, R. Sze, A. Tan, R. Yap, and L. Materum, “Development of a wideband PLC channel emulator with random noise scenarios,” Journal of Telecommunication, Electronic and Computer Engineering. Vol. 10, Nos. 1-9, pp. 153-159, 2018.
- [33] A. Africa, C. Alcantara, M. Lagula, A. Latina Jr. and C. Te, “Mobile phone graphical user interface (GUI) for appliance remote control: An SMS-Based electronic appliance monitoring and control system.” International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE). Vol. 8, No. 3, pp. 487-494, 2019.  
<https://doi.org/10.30534/ijatcse/2019/23832019>
- [34] A. Africa, and C. Uy, “Development of a cost-efficient waste bin management system with mobile monitoring and tracking.” International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE). Vol. 8, No. 2, pp. 319-327, 2019.  
<https://doi.org/10.30534/ijatcse/2019/35822019>
- [35] J. Sashi Banu, J. Sastry, B. Devi and V. Prakash, “Career Guidance through TIC-TAC-TOE Game.” International Journal of Emerging Trends in Engineering Research (IJETER). Vol. 7, No. 6, pp. 25-31, 2019.  
<https://doi.org/10.30534/ijeter/2019/01762019>
- [36] E. Panganiban, “Microcontroller-based Wearable Blood Pressure Monitoring Device with GPS and SMS Feature through Mobile App.” International Journal of Emerging Trends in Engineering Research (IJETER). Vol. 7, No. 6, pp. 32-35, 2019.  
<https://doi.org/10.30534/ijeter/2019/02762019>
- [37] A. Africa, “A Rough Set Based Solar Powered Flood Water Purification System with a Fuzzy Logic Model.” ARPN Journal of Engineering and Applied Sciences. Vol. 12, No. 3, pp.638-647, 2017.
- [38] A. Africa and J. Velasco, “Development of a Urine Strip Analyzer using Artificial Neural Network using an Android Phone.” ARPN Journal of Engineering and Applied Sciences. Vol. 12, No. 6, pp. 1706-1712, 2017.
- [39] A. Africa, S. Bautista, F. Lardizabal, J. Patron, and A. Santos, “Minimizing Passenger Congestion in Train Stations through Radio Frequency Identification (RFID) coupled with Database Monitoring System.” ARPN Journal of Engineering and Applied Sciences. Vol. 12, No. 9, pp. 2863-2869, 2017.