

DEVELOPING MOBILE APPLICATION FOR GENERIC HUMAN-MACHINE COMMUNICATION

By:

LAU DICK MUN

(Matrix no: 125407)

Supervisor:

Assoc. Prof. Ir. Dr. Chin Jeng Feng

April 2018

This dissertation is submitted to

Universiti Sains Malaysia

As partial fulfillment of the requirement to graduate with honors degree in

**BACHELOR OF ENGINEERING (MANUFACTURING ENGINEERING WITH
MANAGEMENT)**



School of Mechanical Engineering

Engineering Campus

Universiti Sains Malaysia

DECLARATION

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

STATEMENT 1

This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by giving explicit references. Bibliography/references are appended.

STATEMENT 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for interlibrary loan, and for the title and summary to be made available outside organizations.

Sign:

(Lau Dick Mun)

Date:

ACKNOWLEDGEMENT

This thesis would not have been possible unless the help from many people. First for all, I would like to thank you Universiti Sains Malaysia (USM) for giving me the chance to complete this four-year course in Manufacturing Engineering with Management and enhance myself with wide knowledge. Sincere gratitude towards the School of Mechanical Engineering for all the instruments and facilities provided.

Next, I owe my deepest gratitude to my supervisor Associate Professor Ir. Dr. Chin Jeng Feng. Without his continuous optimism concerning this work, enthusiasm, encouragement and support this study would hardly have been completed. He has made available his support in a number of ways. I also want to express my warmest gratitude to him who suggested this topic to me. I am deeply grateful to for making it possible to carry this work.

Last but not least, my sense of gratitude to one and all who, directly or indirectly, have lent their helping hand in this thesis. It is a pleasure to thank those who made this thesis possible.

TABLE OF CONTENTS

CHAPTER ONE INTRODUCTION	1
1.1 Overview	1
1.2 Objectives.....	4
1.3 Problem Statement	4
1.4 Scope of The Project	4
CHAPTER 2 LITERATURE REVIEW	6
2.1 Overview	6
2.2 Industry 4.0 Concept	6
2.3 A Tool to Support Remote Maintenance.....	8
2.4 Chatting Application Based on Android Bluetooth	9
2.5 Android Application Development	10
2.6 Development of Android Application for Courier Monitoring System via Network	11
CHAPTER 3 RESEARCH METHODOLOGY	14
3.1 Overview	14
3.2 Software: MIT App Inventor.....	15
3.3 System Architecture	17
3.4 Application Development	18
3.4.1 Main Screen	18
3.4.2 User Registration Screen	21
3.4.3 Friend List Screen.....	22
3.4.4 Chat Screen.....	25
3.4.5 Fusion Table Screen	28
3.5 Google Fusion Table Setup	31

3.6 Application Testing	33
3.6.1 Pilot Test.....	33
3.6.2 Compatibility Test	33
CHAPTER 4 RESULT AND DISCUSSION	35
4.1 System Testing	35
4.2 Android Version Compatibility Testing.....	39
4.3 Discussion	39
CHAPTER 5 CONCLUSION AND FUTUREWORKS	42
5.1 Conclusion.....	42
5.2 Future Work	42
REFERENCES.....	44
APPENDICES	46

APPENDICES

Appendix A: Hourly billing rates of the various instance classes for Google Cloud.....	46
Appendix B: Hourly billing rates of the various computing resources for Google Cloud.	46
Appendix C: Billing for Cloud Datastore operations.	47
Appendix D: Fees for use of the Search API.....	47
Appendix E: Billing for other resources in Google Cloud.	48

LIST OF FIGURES

Figure 1.1: Pictorial explanation of the proposed mobile application and its novel features.	3
Figure 2.1: Design Principles for Industrie 4.0. (Hermann, Pentek, & Otto, 2016)	7
Figure 2.2 The system as call sequence diagram. (Mahajan et al., 2014)	10
Figure 2.3: Procedure to establish connection with MySQL. (Noury et al., 2012)	11
Figure 2.4: Application Methodology to develop Android Application for Courier Monitoring System. (Atletiko, 2017)	12
Figure 2.5 Case diagram. (Atletiko, 2017)	13
Figure 3.1: Activity flow chart of the project.	15
Figure 3.2: The method to build the blocks in MIT App Inventor II.....	16
Figure 3.3: Overview of the architecture of the monitoring system.	17
Figure 3.4: (a) Main screen design (b) Components used (c) Properties of component of the application.....	19
Figure 3.5: Block diagram for Main Screen.	20
Figure 3.6: (a) User Registration screen design (b) Components used (c) Properties of component of the application.....	21
Figure 3.7: Block diagram for User Registration screen.	22
Figure 3.8: (a) Friend List screen design (b) Components used (c) Properties of component of the application.....	23
Figure 3.9: Block diagram for Friend_List screen.....	24
Figure 3.10: a) Chat screen design (b) Components used (c) Properties of component of the application.....	25
Figure 3.11: Components input for Chat screen.	26
Figure 3.12: Block diagram for Chat screen.....	27

Figure 3.13: (a) Fusion Table screen design (b) Components used (c) Properties of component of the application.....	28
Figure 3.14: Components input for Fusion Table screen.....	29
Figure 3.16: URL of Fusion Table sharing setting.	31
Figure 3.17: Table URL for Fusion Table screen.	31
Figure 3.18: Service Account Email ID for Fusion Table screen.....	32
Figure 3.19: The view setup in Google Fusion Table.....	32
Figure 3.20: Method to upload Excel file into Google Fusion Table.	33
Figure 4.1: Screenshots on mobile application (a) Main Screen (b) Friend List Screen (c)Chat Screen. (d) Text box after submitting data (e) Notification after submitting data (f)Display of chart.....	36
Figure 4.2: Screenshots on data tabulation in Google Fusion Table using browser in computer.	38
Figure 4.3: Size of file that generated by Google Fusion Table	38
Figure 4.4: The compatibility testing on android with API version 5.0 Lollipop (old version) using Samsung Galaxy Note 3.....	39

ABSTRAK

Revolusi perindustrian yang seterusnya, Industri 4.0 merupakan pembuatan yang canggih dan meliputi Internet of Things (IoT), teknologi awan dan data besar. Senario ini memerlukan aplikasi masa nyata yang amat penting dalam sistem pemantauan yang efektif dengan serta-merta di mana-mana sahaja pada setiap masa. Sebuah aplikasi sistem pemantauan telah direkakan. Sistem pemantauan yang dicadangkan mampu bertindak sebagai sistem pengawasan berdasarkan platform Android untuk mengakses parameter yang dipantau dengan cepat pada peranti mudah alih di mana-mana sahaja pada bila-bila masa. Aplikasi ini dibina dengan menggunakan MIT App Inventor II dan ia boleh digunakan dalam setiap peranti mudah alih yang dipasang dengan Android. Graphical User Interface (GUI) direkakan untuk mempamerkan maklumat yang dikumpulkan dalam sistem. Antara fungsi utama aplikasi ini ialah mampu memaparkan data, menyimpan data, menjadi sistem peringatan dan membolehkan perkongsian maklumat dalam kalangan pengguna. Selain pemaparan data, aplikasi ini memberikan kemudahan kepada pengguna untuk mengadakan komunikasi antara satu sama lain dan membincangkan isu-isu yang berlaku. Syarat untuk menjalankan aplikasi ini hanya memerlukan rangkaian 3G/4G atau sambungan Wi-Fi melalui sokongan aplikasi yang diberikan oleh sistem Android. Akhir sekali, ujian pada aplikasi ini akan dijalankan untuk memahami keserasiannya dalam versi Android API yang lama dan baharu.

ABSTRACT

As the next industrial revolution, Industry 4.0 is an intelligent production incorporated with Internet of Things Remote (IoT), cloud technology and big data. This scenario requires real time applications which are important to monitoring system effectively, instantly, anywhere and anytime. A monitoring system application is designed and developed. The proposed system acts as the monitoring system based on Android platform to give convenience in accessing monitored parameters quickly on mobile devices anywhere at anytime. The application is built by using MIT App Inventor II and is applicable in any android mobile device. A graphical user interface (GUI) is developed as the monitoring system which exhibits the information gathered from the system. Its main functions are to display information, store data, alert system and allow information sharing among the users. Beside displaying the charts, the application enables the user to have the conversation with each other to communicate and discuss about any issues occurred. The only condition required is the application supportability given by the android system over 3G/4G network or Wi-Fi connection. Finally, a pilot test will be tested on this application to understand its compatibility on old and new version of Android API.

CHAPTER ONE INTRODUCTION

1.1 Overview

In the realm of manufacturing, the advances of science and technology continuously support the development of industrialisation all around the world. Though there is still no universal agreement on what constitutes an industrial revolution. From a technological evolution perspective, there are four stages commonly identified. The first three industrial revolutions took around two centuries, and are the result of, respectively: (1) the introduction of water and steam-powered mechanical manufacturing facilities; (2) the application of electrically-powered mass production technologies through the division of labour; and (3) the use of electronics and information technology (IT) to support further automation of manufacturing. In recent years, along with the increased research attention on the Internet of Things (IoT) and Cyber-Physical Systems (CPS). Governments and industries worldwide have noticed this trend and acted to benefit from what this new industrial revolution wave could provide annually sets billions of incomes aside for the development of technologies.

Previously, humans have to monitor the situation of the machine with their own eyes. Now, the monitoring system can be display on a screen of computer but with wired. In the future, it is believed that the real-time condition of a machine can be sent to a mobile device in the dialogs form and humans can track the task on going and performance of the machine from the mobile device. The data is imported to and exported from a cloud database. This scenario could make “the integration of complex physical machinery and devices with networked sensors and software, used to predict, control and plan for better business and societal outcomes” by Consortium II Fact Sheet (2013) come true. (Mario, Tobias, & Boris, 2015)

Monitoring is done on mobile device such as handpone or tablets. With the development of 3G and 4G technology today, wireless bandwidth is larger, which makes it possible to develop more content-rich. (Abhishek & Pragnesh, 2012) Nourya, et al., had found that the smartphone is a good candidate in monitoring the mobility of the elderly as this technology is already accepted in our daily life. (Noury, Quach, Berenguer, Bouzi, & Teyssier, 2012) Leijdekkers, et al., developed a remote healthcare monitoring system that to are for the elderly and the chronically ill in their homes with care facilities. This remote healthcare monitoring system includes a smart phone, wireless sensors, web servers and IP webcams. The authors centered the window based smart phone by integrates all the devices with it. The smart

phone acts as a receiver that receives all the data sent by the wireless healthcare sensor and further processes the data by sending the data the healthcare server or calls an ambulance during emergency. With the remote healthcare monitoring system, patients are free to move about in their own home and secure in the knowledge that they are being monitored. (Leijdekkers, Gay, & Lawrence, 2007)

Since the monitoring system can be done remotely for commercial, it is possible to introduce in industrial practice as well. Remote monitoring of equipment or structures that are inaccessible to configure within a local network can be designed. Examples of remote condition monitoring include the monitoring of vibrations, temperature of the machines and the unit produced during the production.

Android is an open source phone operation system based on linux platform and it's the first truly open and complete mobile software for mobile terminal. (Sung Wook et al., 2011) Android application can be written using Kotlin, Java and C++ languages. The Android SDK tools compile the code along with any data and resource files into an APK, an Android Package, which is an archive file with an .apk suffix. All the contents of an Android application are contained in one APK file and it is the file that Android-powered devices use to install the application.

Generally, this project is to build a mobile app between machines and mobile phone which allows online display of information, auto decision making and alert system. Figure 1 presents the pictorial diagram depicting the functions of the application and how it is remotely connected to machines. The general idea is for machine to supply a stream of processing related data to mobile device and the proposed application would take advantage of these information for machine monitoring. Notable features of the application online display of information, performance comparison, embedded with a certain level of autonomous decision making and alert system. For online information display, various graphs from the predetermined set in the library can be chosen to present the data. For example, machine hourly production rate can be presented in the form of line graph. Similarly, performance can be compared within a cluster of identical machines or heterogeneous machines. A case in point is to compare the production performance at the frontend and backend of the process value chain. Conditions can be set for the mobile application to alert the user when they are met. Autonomous decision making allows a programmed sequence of tasks to be executed, e.g. send out warning to the related production personnel in the event of machine breakdown. Data analysis can be devised on the mobile

device. A case in point could be the measuring of Overall Equipment Efficiency (OEE) when the data such as breakdown, defect and utilization are collected from the machine. It is applicable in industry and non-industry. Ways of talking is in English version, perhaps other language in the future.

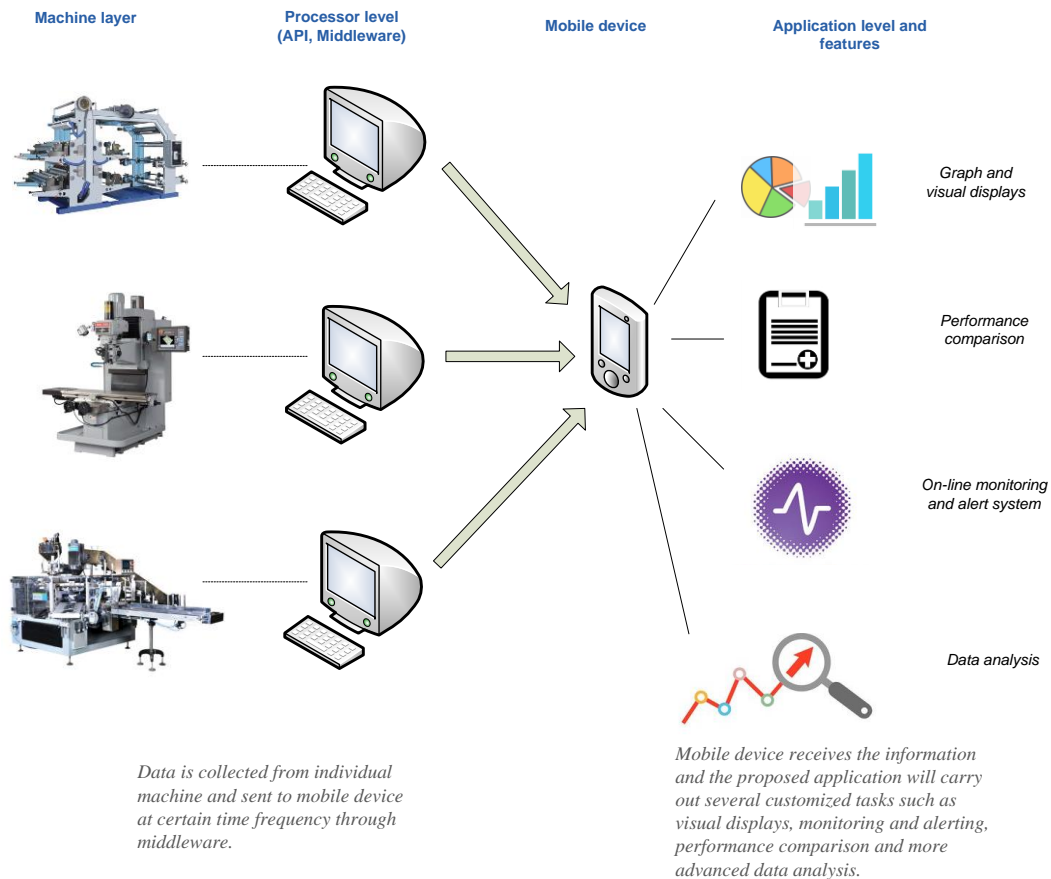


Figure 1.1: Pictorial explanation of the proposed mobile application and its novel features.

In term of architecture, two major components are middleware and mobile application software. The middleware intends to be installed at the processor, such as PC connected to the machine. The middleware would draw data from the machine through appropriate application program interface (API). The middleware also acts as messenger to transmit the data to the designated mobile device. The mobile application software deposits the data and performs a collection of tasks determined by users. Both components would be programmed in different mobile application developer platform including MIT App Inventor.

1.2 Objectives

The objectives of this project are as below:

- a) To design and develop a mobile application that enables the features of real-time machine monitoring.
- b) To validate and demonstrate the operation of mobile application.

1.3 Problem Statement

Among the topics discussed, there are some literatures focusing on the impact from Industry 1.0 to 3.0. However, to accelerate industries towards the next level of technology adaptation which is Industry 4.0, the increasing integration of the internet of everything into the industrial value chain is necessary as a foundation. Therefore, the main objective of the present paper is to entirely discuss the feature needed in the communication between machine and mobile device that aim for monitoring the machine's condition anytime anywhere. To enhance the portability of the remote monitoring system, the device should be wireless. Also, the application of the project should identify the parameter information to be display which able to help in faster monitoring system.

A new remote machine monitoring system should be build. The application needs a user-server system where the server controls the data and information needed to do the required work. The user-server system is important since it centralizes the data and information. Users can only access areas pertained to them as for reasons of security.

1.4 Scope of The Project

The system being designed is with respect to the operators and production manager point of view. The goal is to develop a mobile monitoring system application. Basic approach attempts to develop a smart phone based application using Android that can be used to make the process easier, secure and less error occurred. More efficient information will be achieved through this system with the feature of monitoring system and enable discussion forum communication in two ways. Within the application, users can also update the production information such as the type of production and its quantity anytime anywhere as long as there

is an internet connection. On the other hand, the application can extract and display useful information on the mobile device from the database. Thus, avoiding the necessary for production manager to travel over a distance as to get the updates from the production floor. Any other simple registration will also be made available in this application, replacing the need of paper forms to digital forms. Other authorized users are also allowed to view the data stored in the database.

CHAPTER 2 LITERATURE REVIEW

2.1 Overview

The Internet of Things (IoT) offers both new opportunities and challenges to developers. An open application programming interface (API) could be the heart of an IoT platform. An open API is central to all IoT development because IoT development is at its essence about connectivity. The IoT devices can be linked and the data collect are back into a large big-data repository, and the analytics programs make sense out of all the IoT data.

For consumer-oriented IoT devices and applications, universal language and multi-device capabilities or multi operating system are highly desirable. Universal language capabilities will ensure that a product or service being developed has global potential. Whereas for multi-device compatibility, developing for multi mobile devices is a bigger challenge. It will be successful if consumers can use it on their preferred device.

The android operating system is a trending OS now that is used in android mobile devices. This operating system powers more than billion smartphones and tablets. Besides, android is open source and building on open-source components helps drive down the cost of IoT development, while simultaneously preventing an IoT application from being locked into a proprietary software stack. Most importantly, application that made by android are highly customizable which able to customize the desire monitoring system in this case. System which is based on android platform gives authorized users a medium to remotely access and control monitored parameters quickly on mobile devices using any WiFi or 3G/4G.

2.2 Industry 4.0 Concept

The term “Industry 4.0” originates from a project in the high-tech strategy of the German government, which promotes the computerization of manufacturing. The term was first used in 2011 at the Hanover Fair. In October, the Working Group on Industry 4.0 chaired by Siegfried Dais (Robert Bosch GmbH) and Henning Kagermann (acatech) presented a set of Industry 4.0 implementation recommendations to the German federal government. On 8 April 2013 at the Hannover Fair, the final report of the Working Group Industry 4.0 was presented.

Since industry 4.0 is the vision of the industrial production of the future, it requires nine technologies to transform industrial production from the past. For instances, augmented reality,

additive manufacturing, the cloud, cyber security, Internet of Things (IoT), horizontal and vertical system integration, simulation, autonomous robot and big data and analytics. (Rüßmann et al., 2015)

Industry 4.0 is a trend in the manufacturing domain, based on the concept of “smart factory”. Among other organizational services, Industry 4.0 requires a quick and efficient maintenance service in order to guarantee that companies implement an efficient production system. Enabled through the communication between people, machines, and resources, the fourth industrial revolution is characterized by a paradigm shift from centrally controlled to decentralized production processes. Smart products know their production history, their current and target state, and actively steer themselves through the production process by instructing machines to perform the required manufacturing tasks and ordering conveyors for transportation to the next production stage. (Kagermann, 2015) Thus, well communication can be seen playing a vital role in order to make this factory “smart”.

So, question comes, how can we make this Industry 4.0 comes true? The research by Mario Hermann resulted in four Industrie 4.0 design principles as shown in Figure 2.1.

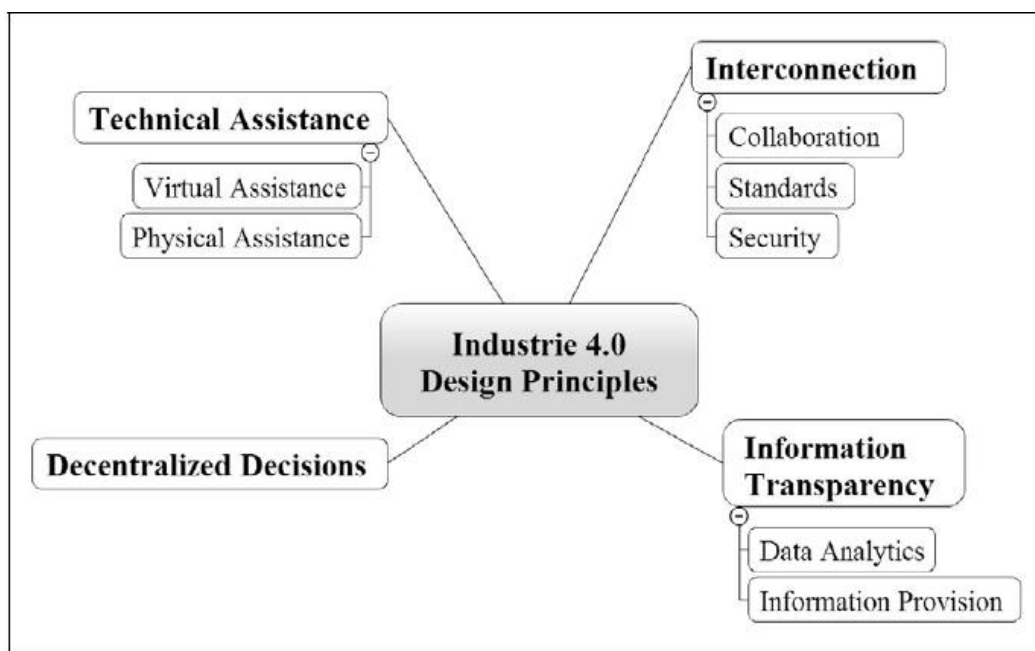


Figure 2.1: Design Principles for Industrie 4.0. (Hermann, Pentek, & Otto, 2016)

The quantitative text analysis and qualitative literature review identified four design principles guiding practitioners and scientists on “how to do” Industrie 4.0 (Gregor, 2006): interconnection, information transparency, decentralized decisions, and technical assistance. It

strikes out that certain design elements are discussed more frequently in practical publications. In particular, human-machine collaboration, data and information security, and decentralized decisions are more often discussed in industry publications. The frequent discussion of the first two design elements highlights the biggest challenges for a successful implementation of Industrie 4.0 from a practitioner's perspective, while decentralized decision making as a key principle of Industrie 4.0 is understood as its most disruptive element and, therefore, explained on a very detailed and extensive level. (Hermann et al., 2016)

2.3 A Tool to Support Remote Maintenance

Based on the analysis of industrial needs to enable remote maintenance performed in years, after the release of the application described in (Bordegoni et al., 2014), a tool is found should include the following features to support remote maintenance effectively:

a) An inspection module

- The maintenance expert should be able to see why and where the failure occurred, to understand how to perform the maintenance activity. Usually, the diagnosis is performed by visually or aurally inspecting the machine. Thus, the remote maintenance system should include a module with a video and audio streaming. Usually, companies describe this requirement such as “the maintenance expert should be able to see and hear through the operator's eye and ears and then operate through his hands”. While this specification seems to be easily reachable thanks to the high availability of cameras and microphones on off-the-shelf mobile technology, goggles and so on, it is important to analyse their specifications to understand that they might not be able to support the diagnosis task. Most of the available mobile technology today might not allow an effective visual inspection of the failed machine parts. Sometimes the failure makes the machine stop working, but sometimes the failure causes unwanted movements into the machine and to detect the failure is necessary to see the machine while is running. The movements might be at high speeds, and this is not compatible with the limitations of the available mobile cameras which work at very limited frame rates. Usually mobile device cameras of the consumer market work at a frame rate of 30 Hz.

- b) The use of universal language to allow people of different languages and cultures to exchange information without misunderstandings.
- In this view, the system could make use of graphic symbols, together with text and audio messages. Indeed, it might happen that the maintenance expert is asked to communicate to local operators, who do not necessarily speak the same language. This is a quite common problem we found in industries. To our knowledge, there are few attempts to find a solution to this problem.
- c) A maintenance recorder module
- The maintenance operation should be recorded, in order to become a potential manual if the same failure occurs on another machine, or to register the maintenance operation for legal issues. (Masoni et al., 2017)

These considerations lead to IoT technologies such as RFID and wireless communications are used for capturing real-time machines statuses. After that, such information is visualized through a graphical dashboard after being processed by various data models and cloud-based services over smart phones. (Zhong, Wang, & Xu, 2017)

2.4 Chatting Application Based on Android Bluetooth

Bluetooth provides the communication on low-cost, low-power basis. Wireless communication can also be done with the help of Bluetooth technology in a mobile communication. Short-range establishment of two-way communication has occurred without any support of the network. Bluetooth is integrated into Android which is a mainstream Smartphone platform as a mean of mobile communication. The Android system provides many Bluetooth APIs for developers to call. (Mahajan, Verma, Erale, Bonde, & Arya, 2014)

This system mainly used the Android system components is the Activity and Service. In the Android system each Activity is an independent process, each Service is an independent process, while the Activity and Service must be communicated with each other, then you need to use the Binder mechanism (Mahajan, Verma, Erale, Bonde, & Arya, 2012). Android Binder mechanism abstract defined by Binder interface. In the upper, its concrete implementation is complete by AIDL. The main program of this system includes three files: BluetoothChat.java BluetoothChatService.java and DeviceListActivity.java, detailed features can be seen below the description shown in Figure 2.2.

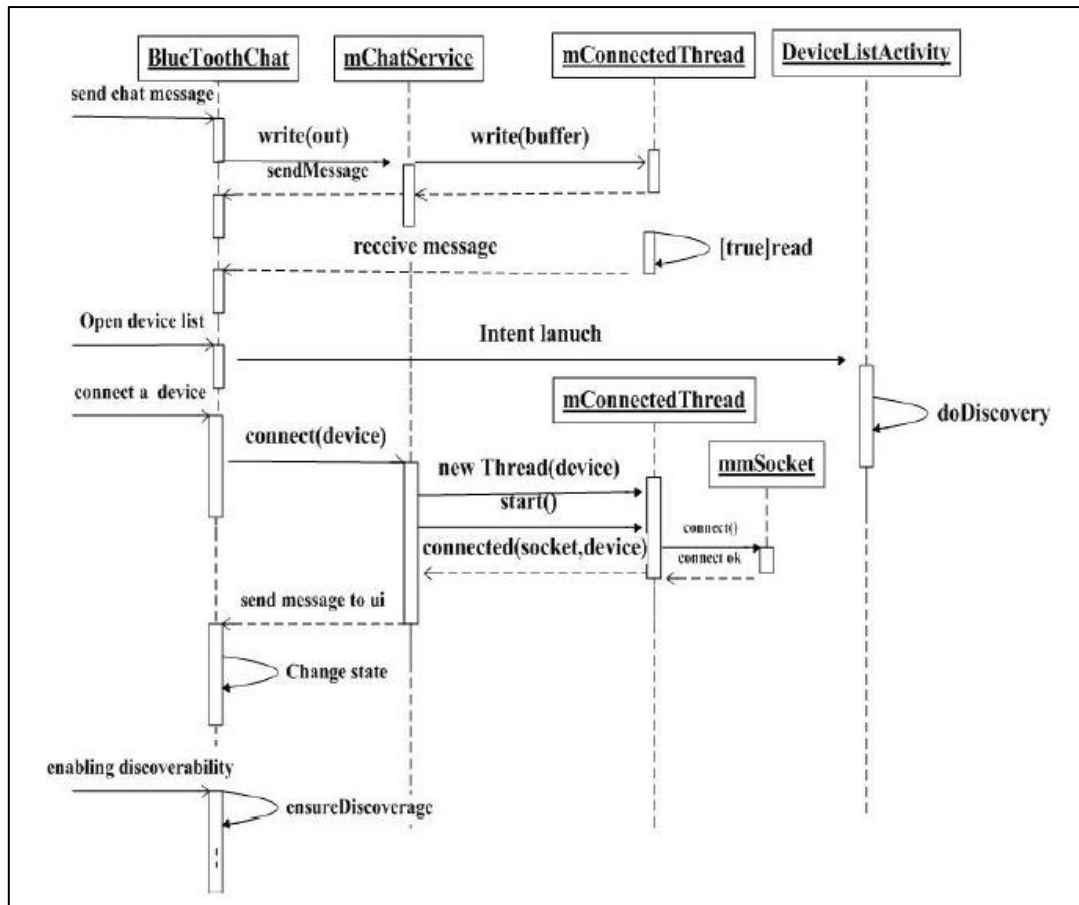


Figure 2.2 The system as call sequence diagram. (Mahajan et al., 2014)

2.5 Android Application Development

Android's open source platform offers compatibility with a wide range of devices, which provide global access to the mobile market. Revolutionize the organization by delivering robust business application for Android phones and tablets and integrate them with enterprise systems.

The foundational skills needed for the management and administration of networked industrial control systems. Many courses help plant administrators, control systems engineers and traditional network engineers understand networking technologies that are needed in today's connected plants and enterprises.

By now, there is a procedure to establish an Android application connection with MySQL as shown in Figure 2.3.

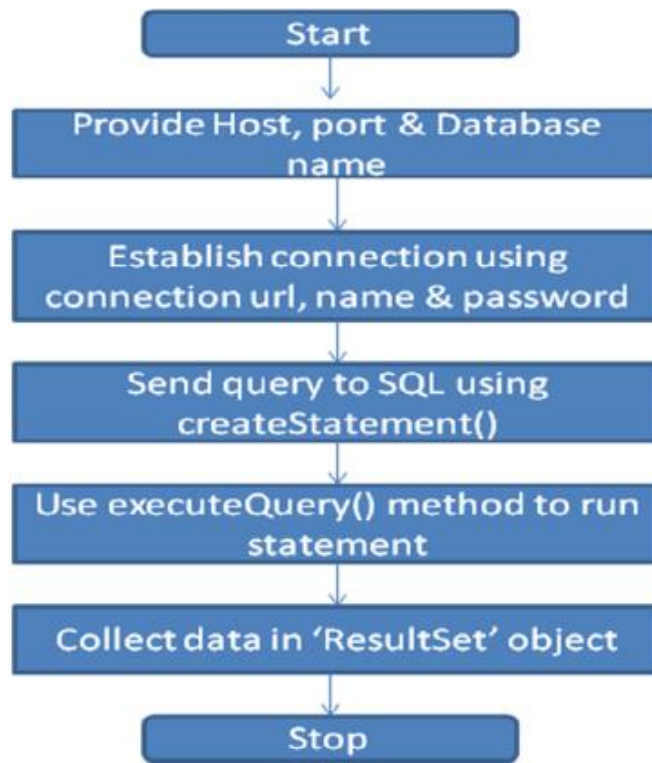


Figure 2.3: Procedure to establish connection with MySQL. (Noury et al., 2012)

2.6 Development of Android Application for Courier Monitoring System via Network

Based on the research conducted by Faizal Johan Atletiko (Atletiko, 2017), he designed and built a courier monitoring system. At the time of the distribution process performed, a problem could occur. Medicine can be ordered when distributors assign some of its salesman for visiting pharmacies or hospitals and offer medicine to be sold. Pharmacies who will order must create a letter and handed it to the salesman. All orders from some pharmacies or hospitals will be entered into the order book manually by the salesman. Afterwards, order book returned to distributor and after going through several process again, the order will be sent to the pharmacy or hospital. That issue is the topic of this research, which at the time of delivery of order, courier could be negligent. Therefore, courier needs to be monitored to determine whether the order has been shipped or not. In this case, Faizal Johan Atletiko designed and build a courier monitoring system for medicine distributor company, salesman, courier. He believed that what he had done could enable the pharmacies or hospitals order medicine quickly and in the right amount with this courier monitoring system. His application can also help

distributors in order to monitor location of the courier so that the business process of both parts would run properly.

The stages involved for him to develop Courier Monitoring System are as shown in Figure 2.4.

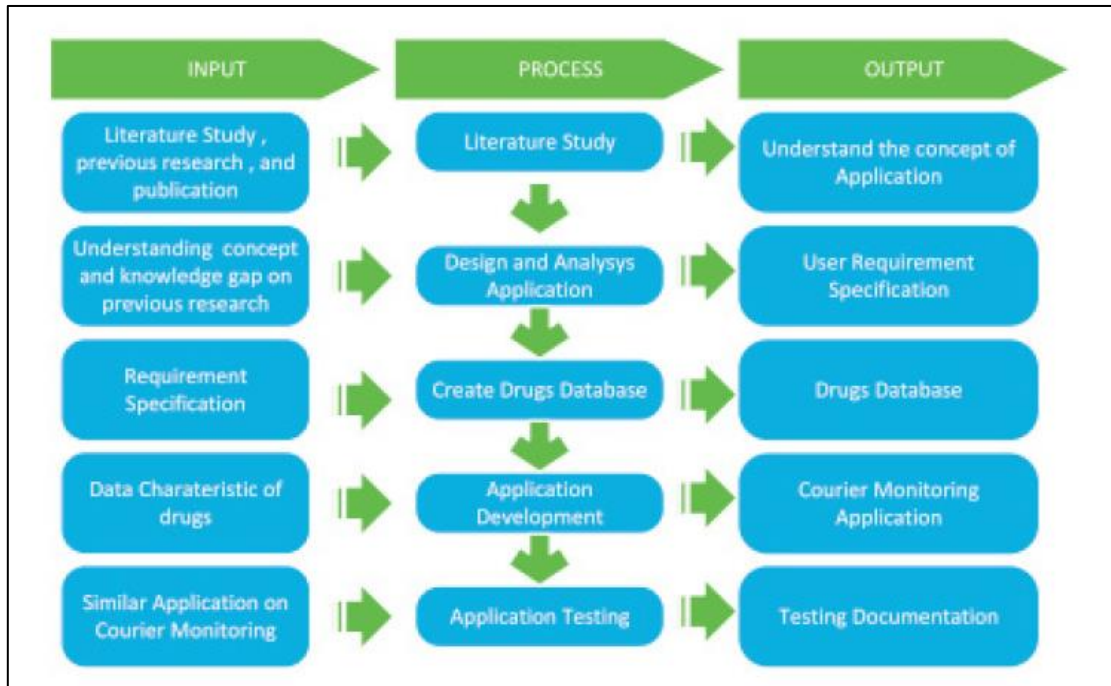


Figure 2.4: Application Methodology to develop Android Application for Courier Monitoring System. (Atletiko, 2017)

After making the sequence diagram, class diagram, method in each Class source code and interface design, he created a database as the list of drugs. The process of collecting data by extracting drug list information on drug distributor company website. So, it is clearly stated that the core of this project is to have a database with full information. This is because without information from database, the user might fail to receive or deliver the correct type and amount of the drug.

Beside the application design, he also created a map for different level of users with their specific role. The map can have found in Figure 2.5. The mapping shows how does the system functionality works.

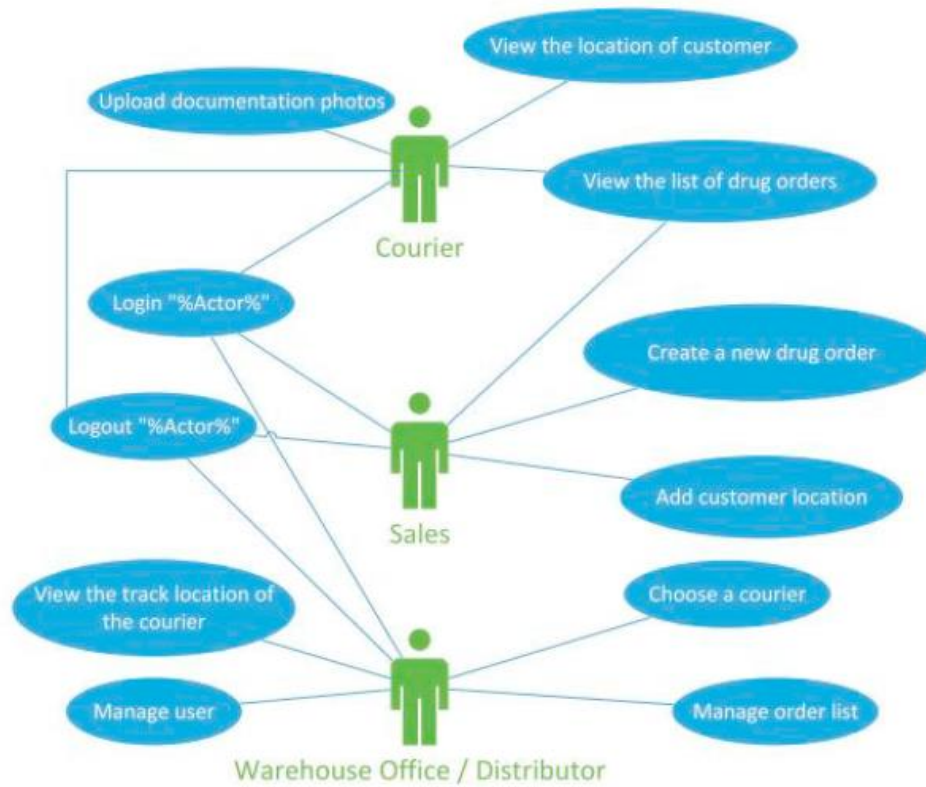


Figure 2.5 Case diagram. (Atletiko, 2017)

Therefore, this final project gives the idea of making an Android machine talk between machine and mobile device to communicate all the possible tasks. Design GUIs that work seamlessly with a range of phones and tablets is the another consideration factor followed by integrate applications with enterprise web and location-based services.

CHAPTER 3 RESEARCH METHODOLOGY

3.1 Overview

The project development is executed in the fashion of a waterfall model. Figure 3.1 respectively depict the process flow as well as the associated time allocation on individual phases. The duration requires to complete each phase would be different naturally due to the complexity and also the learning curve. Time for product testing and pilot run would be expected to be significant and various as the consensus is that a commercializable product need also a strong user experience provider besides fulfilling the functional requirement. Additional time therefore would be allocated as a buffer for mobile application fine-tuning after feedback collected from the users.

The first phase is a conceptualization period whereby extensive literature review (including from non-academic sources) would be carried out. As the objective has already been rather clearly defined, the outcomes of literature review should be specific, for example, compatible list of lean tools suitable at a particular phase of continuous improvement. An acceptable duration for literature review could be three months. This is achievable as a large sum of information has already been accumulated through the previous researches. Stakeholder analysis is to gather user requirement. Market survey and meetings with the industries and Lean practitioners, as parts of market research are held in order to determine the requirements such as data input, contextual factors, level of customization, result presentation and so forth.

In Phase 2, system and interface design would be coming in next, where decisions have to be made on the programming platform, program flow, data storage, fool-proof, guide, user experience which includes the intuitiveness as well as artistic values. Incorporating the data collected from the literature review as well as stakeholder analysis. MIT App Inventor II provides toolboxes to expedite application development. For the machine, a middleware will be built to collect data via machine application program interface (API), transmit information digitally to respective mobile phones. This phase ends with the completion of alpha-version of the mobile application.

In Phase 3, multiple test runs would be carried out through student focus groups. This is to collect the feedback about the performance of the applications and making notes on additional requirements. This step helps remove possible flaws. The application is refined

accordingly. This is followed by pilot run in selected machine. The pilot run is done iteratively, at least for several rounds for the system to converge and to accurately reflect the industry expectation, functionally and cosmetically.

Phase 4 starts with the completion of mobile application, a stable version ready to be mass distributed. This phase includes promoting the application through online platforms and exhibitions whenever opportunity presents itself.

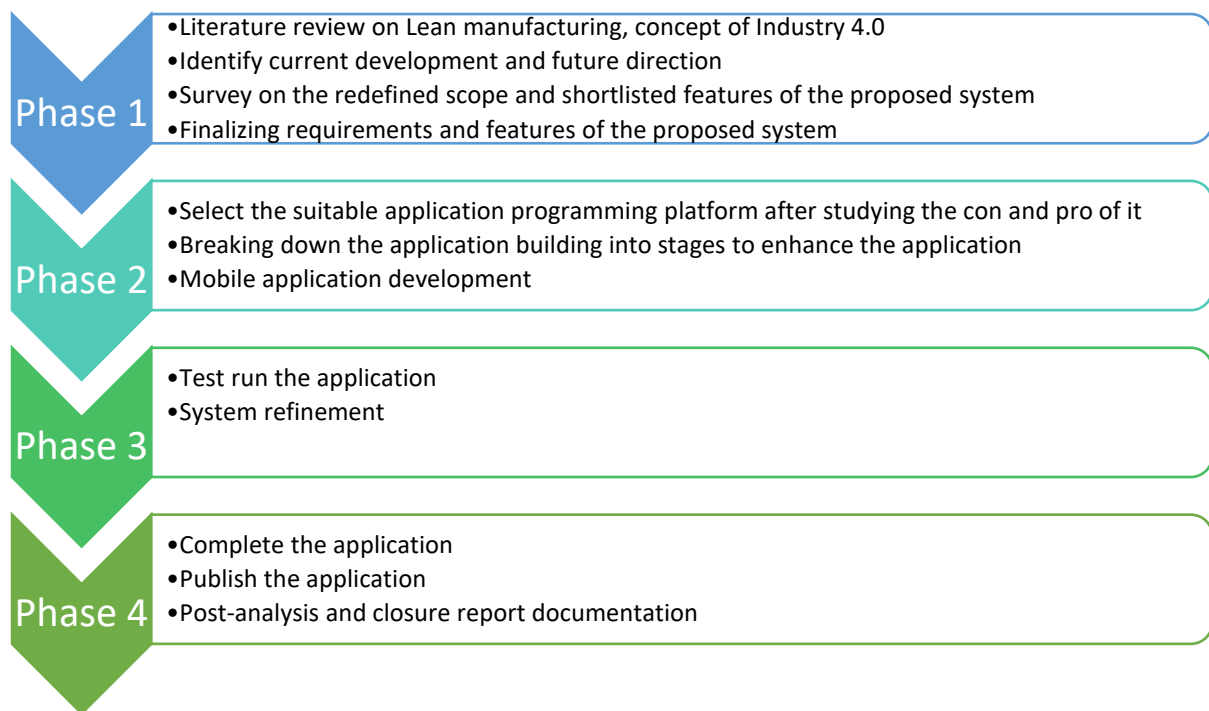


Figure 3.1: Activity flow chart of the project.

3.2 Software: MIT App Inventor

This project will only fully focus on programming. MIT APP Inventor is used as the platform for this project. App Inventor for android is an open source web application originally provided by Google and now maintained by the Massachusetts Institute of Technology (MIT). It allows newcomers to computer programming to create software applications for the Android operating system (OS). It allows users to drag-and-drop visual objects to create an application that can run on Android devices. Its coding is like plugging puzzle pieces together as shown in Figure 3.2. Newcomers can learn to build games, social apps, quizzes and other apps within

hours. One neat thing about APP Inventor is the app can be seen on our phone while building it.

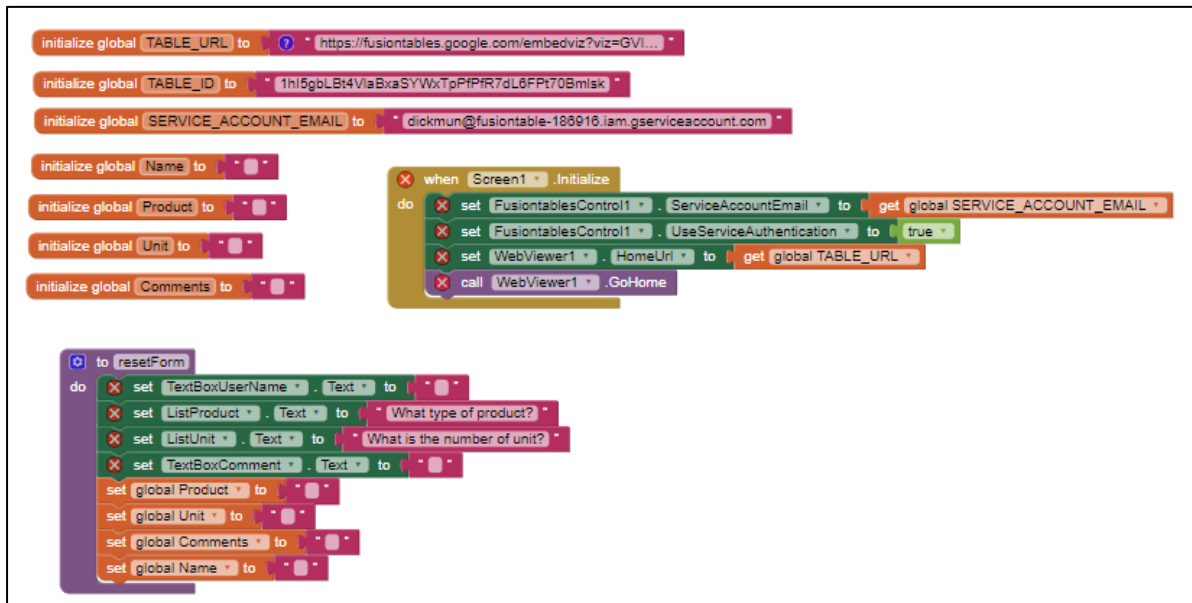


Figure 3.2: The method to build the blocks in MIT App Inventor II.

The codes generated in this project are in blocks type which it is the source code of the language. It describes the syntax tree of App Inventor program in the same way that traditional text programs describe the syntax trees of their languages.

When an aia file (which is really just a .zip file) is saved, the blocks are saved in an XML textual format that describes exactly the same nested tree structure in which the blocks are shown on the screen. So, if you're more comfortable with thinking of text as source code, this XML format is the source code of App Inventor programs.

When blocks program is executed on mobile device, it is not translated to a traditional language like Java/C/JavaScript etc. It is translated to Scheme-like YAIL code and executed in an interpreter on the phone. Ultimately all Android apps are running a version of Java, but App Inventor code is never converted to Java in live development mode. Rather, it's YAIL code that runs in an interpreter implemented on top of Java. In contrast, Android Studio coding will cause error easily if one line of command input is wrong.

3.3 System Architecture

Before creating an application, the architecture of the application has to be determined. Each system does different work. It explains how to perform modelling tasks and how to create flexible analytical models based on real-time data. Figure 3.3 presents an overall architecture of the proposed.

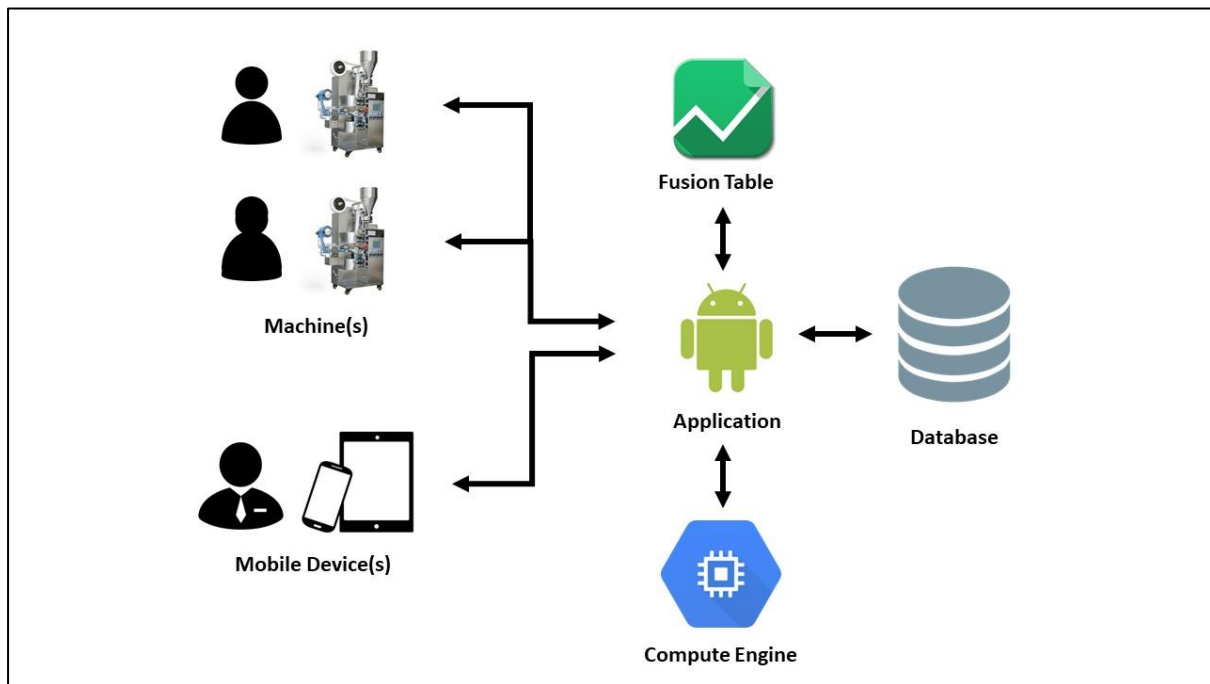


Figure 3.3: Overview of the architecture of the monitoring system.

In this project, all the relationships of the system should be clearly determined and understand. For the requirements analysis, the parameter is collected based on a machine for example punching machine. The process begins with placing the workpiece on the die of punching machine by the operator. Then operator will choose the correct tool and carry out the punching process. After that, the finish product will leave the machine to transfer to other places or processes. The important attribute at this moment is that the user like production manager wants to know the type of product and its unit had produced by the punching machine. Thus, the operator may enter the specific information into an application using either a computer or other devices. This information has to link with the application which installed on mobile device that holding by the user in order to follow up the latest information of the process. Therefore, the user can take whatever further action after knowing the situation of the production in production floor. For example, increase the working time of the operator to speed up the production if the product produced does not achieve the target.

The process of delivery of information that have been recorded also less complicated. With manual data collection by walking to the production floor constantly, a specific time will be lost on traveling.

This application will be divided into four sections as illustrated in Figure 3.2, the first is the “Machine” which controlled by the operator. The information of the product such as quantity of unit produced will be entered through the application. The second section is the “Server”, server that store the incoming data into a database. The third section is the “Compute Engine” which interpret the data and extract the data from server and Google Fusion Table. It is then push the message and display the information for the second user like production manager. The application can be run in a reverse way which mean that the second user can update and change the data in database as well.

Since the objective of this project is to adopt with the Industry 4.0 concept, the application should be included an effective communication tool. Therefore, the chatting tool feature like WhatsApp will be added into this application beside monitoring system.

3.4 Application Development

At this stage, since the software to develop the application which is MIT App Inventor and the overall application architecture had been determined, the following step can be proceeded by building the GUI. Five main screens which are main screen, user registration, friend list, fusion table and chat screen are designed and built.

3.4.1 Main Screen

The application starts with the main screen design by adding in the components and configuring their properties that can be seen in Figure 3.4. A username is a name that identifies someone on a computer system uniquely. Many websites allow users to choose a username so that they can customize their settings or set up an online account. For instance, bank may allow customer to choose a username for accessing your banking information. Usually, a username is almost paired with a password. So does in this project, this username/password combination is referred to as a login and required the user for login this application. Since the application should be used by authorized personnel, hence a user login screen is built to enable the

personnel to sign in as same as normal chatting application like WhatsApp for security purpose. The user's username is appeared on the screen but the password is kept secret. The username and password can contain any numbers or letters with no limit but no spaces. Password may keep the confidential data safe and secure to unauthorized access or modifications.

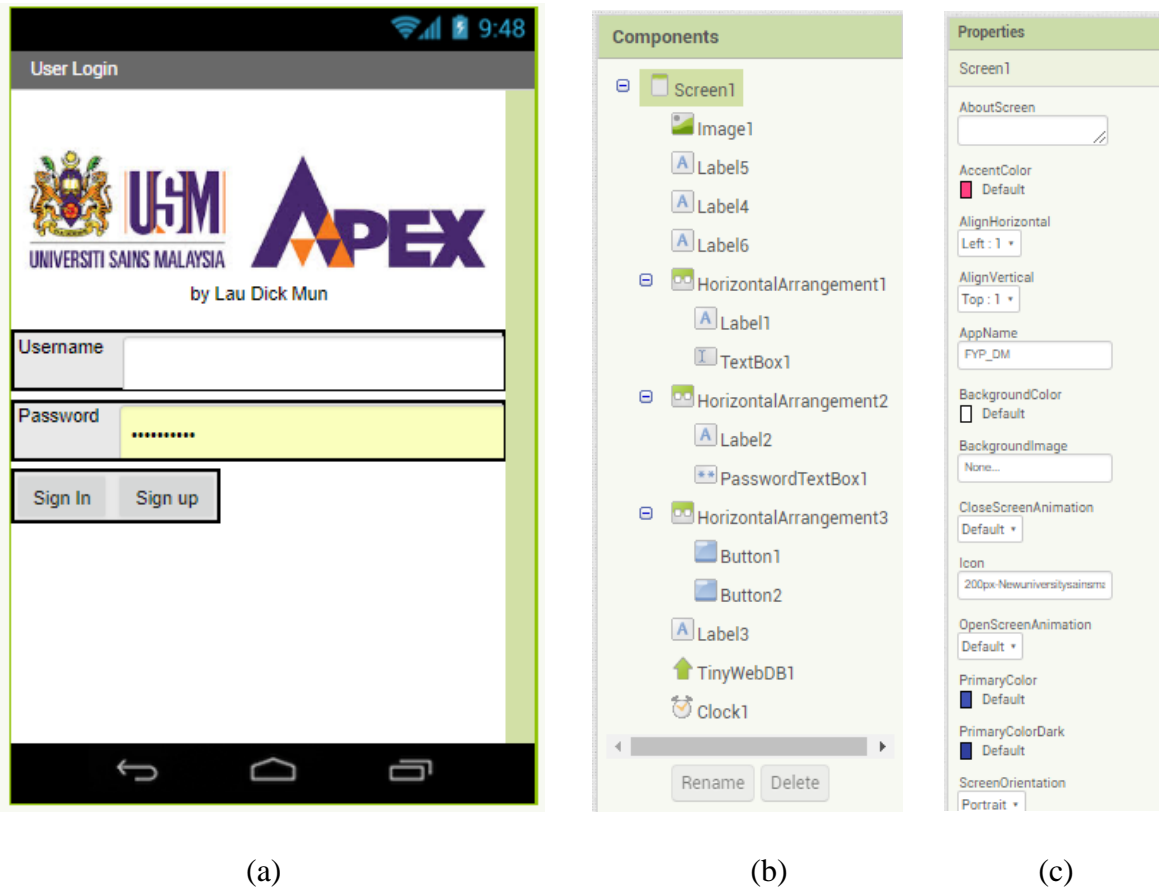


Figure 3.4: (a) Main screen design (b) Components used (c) Properties of component of the application.

At the same time, the person who are in charging the machine can be tracked as the application requires user to register and login with their identities. The server will retain information of all users in the system network. The existing database supports the necessities of the users. Thus, if the system cannot find the username and password from the database, user will be blocking from using this application.

For the block diagram, it is built as Figure 3.5.

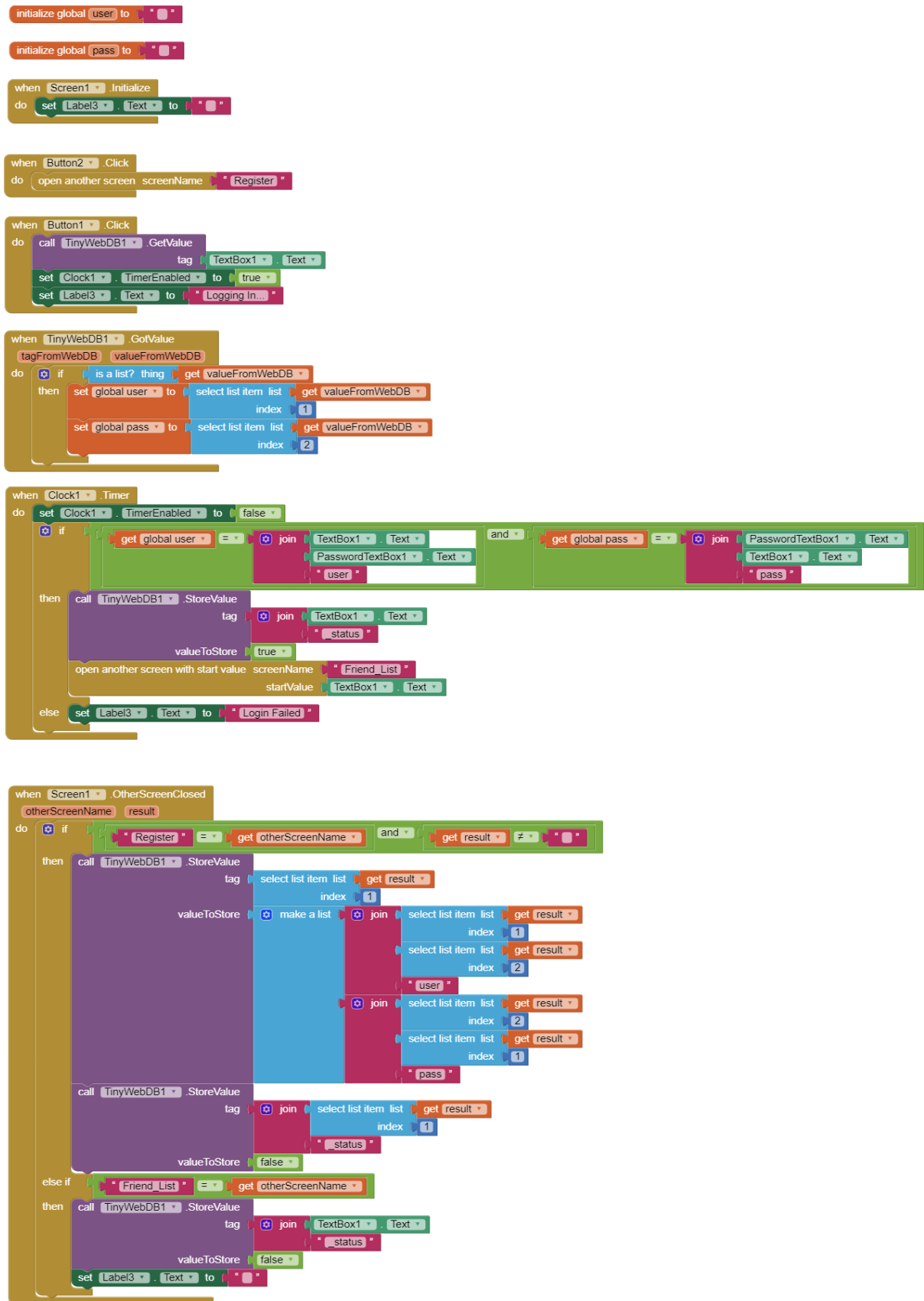


Figure 3.5: Block diagram for Main Screen.

3.4.2 User Registration Screen

Before login to the application, user is required to register himself/herself in the system. Thus, a User Registration screen is designed as Figure 3.6 to sign up with a new username and password by input the components and setting their properties.

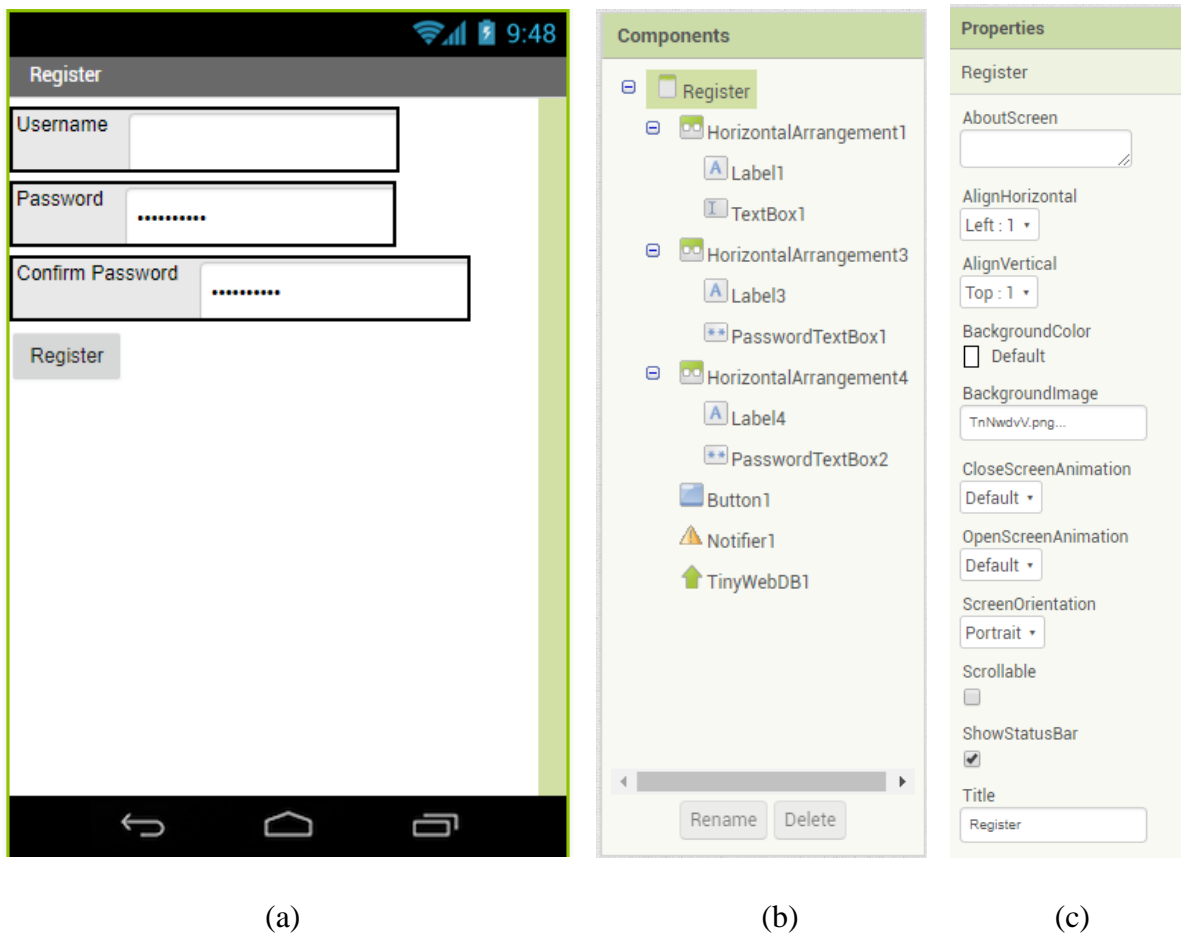


Figure 3.6: (a) User Registration screen design (b) Components used (c) Properties of component of the application.

For the block diagram, it is built as Figure 3.7.

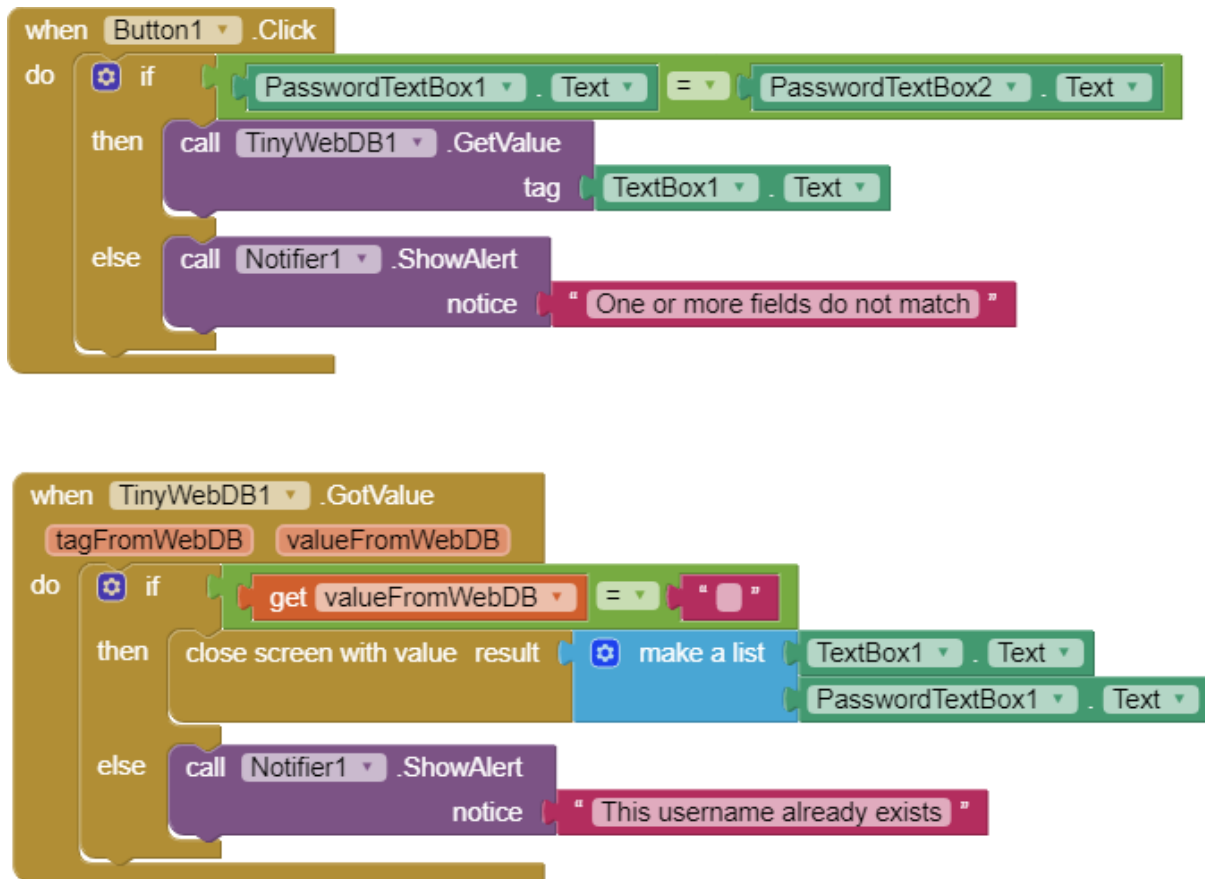
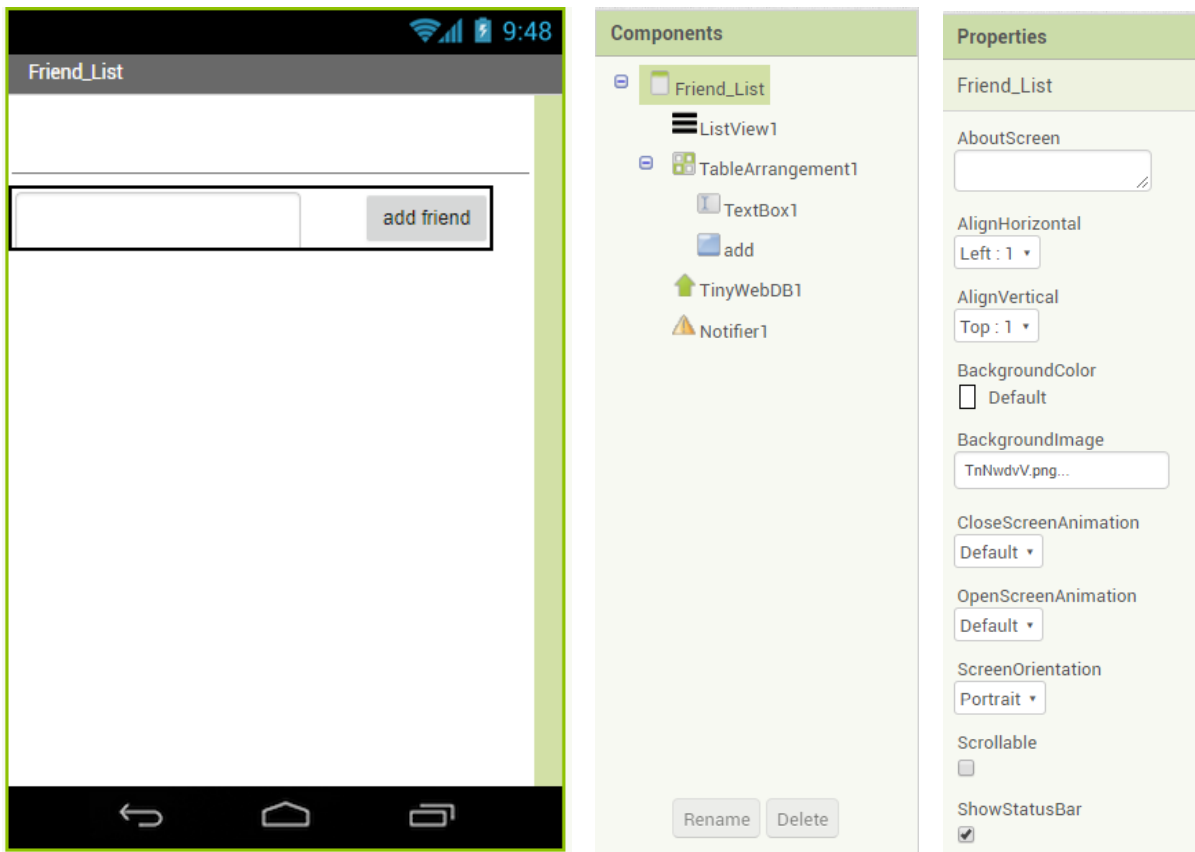


Figure 3.7: Block diagram for User Registration screen.

3.4.3 Friend List Screen

After the user has register an account in the database, he has to add another user as friend to have the conversation with each other. Besides, the user can monitor the activities of right machine or send the message to the right person in the production over the network after have the same channel with another user by adding them as “friend”. Thus, the Friend List screen as shown in Figure 3.8 is designed to let the user to add the specific department or user. Same to previous screen, the specific components are added into the screen and the properties are configured.



(a)

(b)

(c)

Figure 3.8: (a) Friend List screen design (b) Components used (c) Properties of component of the application.

For the block diagram, it is built as Figure 3.9.

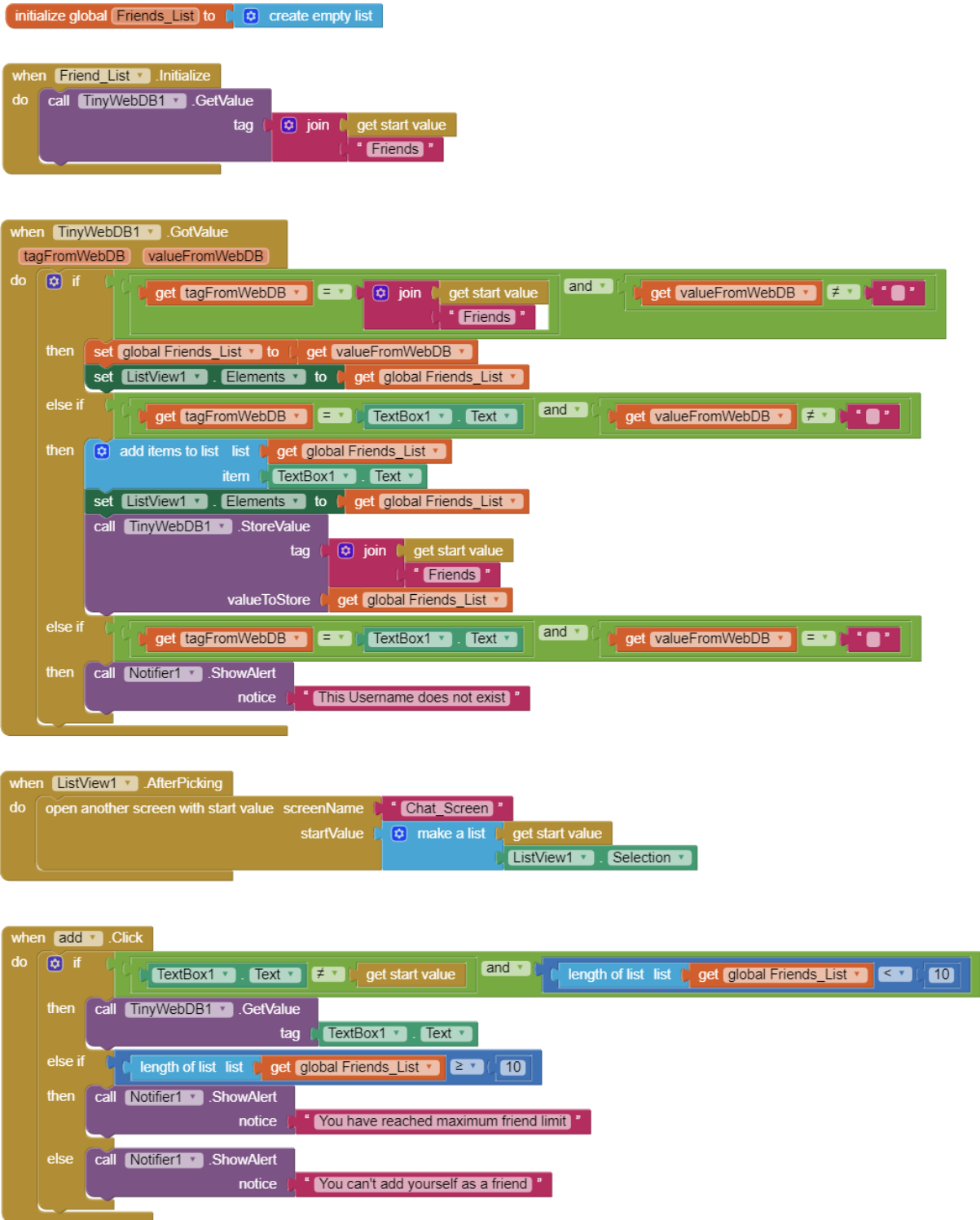


Figure 3.9: Block diagram for Friend_List screen.