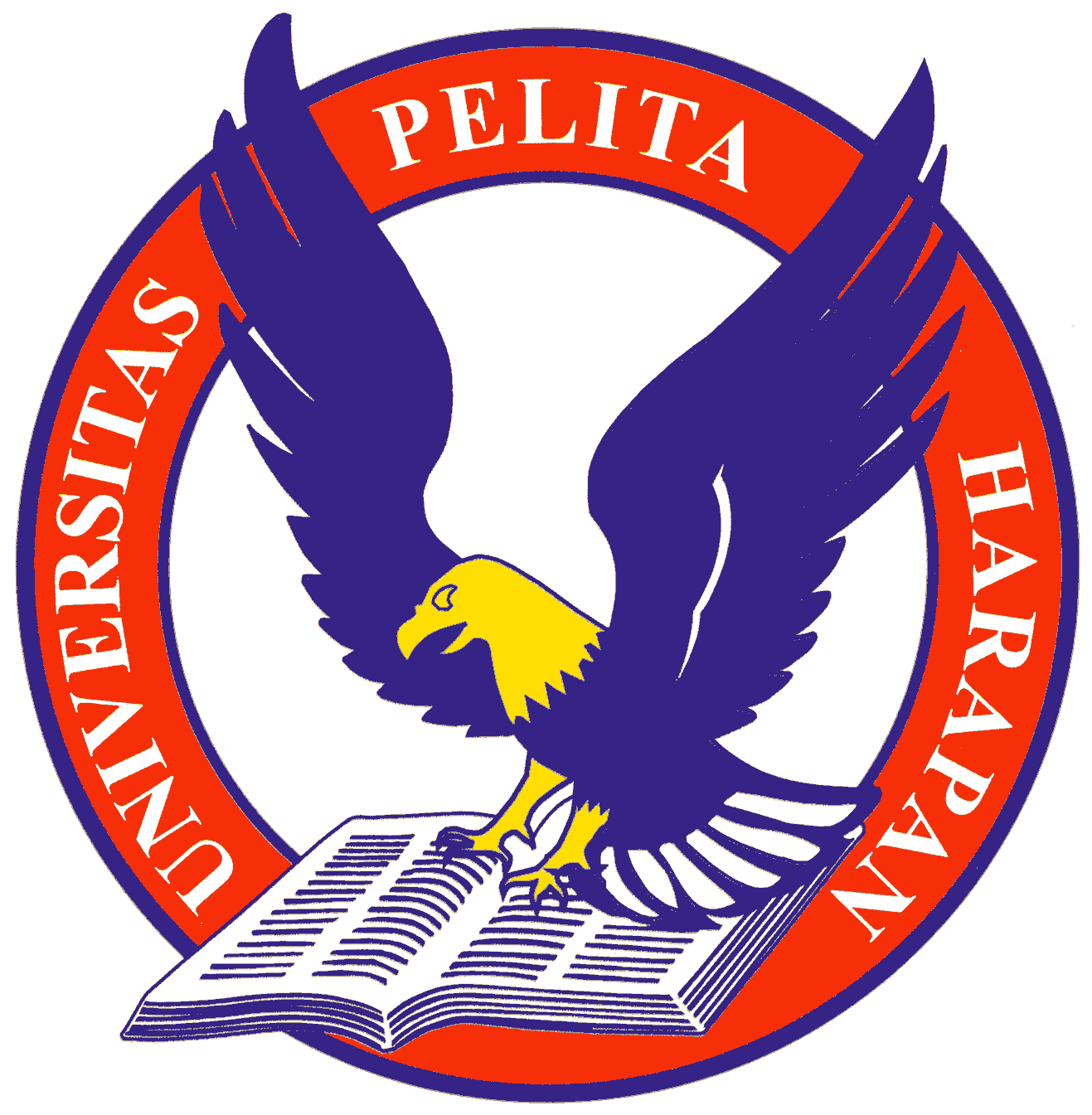
**Mobile Project Report**



Group:

Steven 00000011580 (https://github.com/NamaAkuSteven)

Jason 00000013271 (https://github.com/XtraJos)

**Table of Content**

**Table of Content 2**

**Introduction 3**

**Our Project 3**

**Procedure 4**

**Coding + Explanation 10**

**The Presentation 11**

**The Website 12**

**Limitation Consideration 13**

**Reference 14**

**Introduction**

Human is a social being and that means they need to communicate with one another. Aristotle once said, that human is a social animal, and by any means necessary, needs to interact with one another [1]. It’s a key point why mobile device was created in the first place [2]. As communication becomes major factor for our daily live, mobile device has created an application that manage other person’s number efficiently, which is called Contact. This application is a must in every mobile device. Our project will cover this subject.

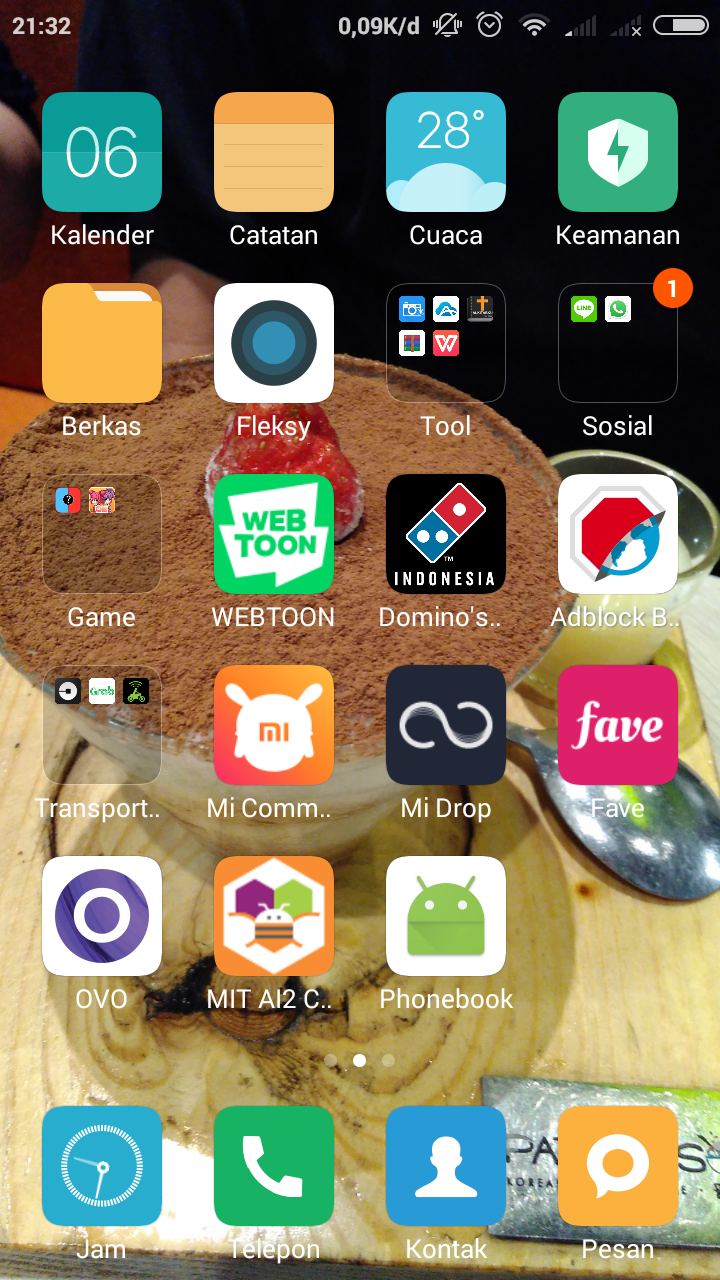
**Our Project**

From our introduction, we have learned the importance of communication. We want to re-create that application that is necessary in every mobile device. When time had passed, we finally decided the name of our project: **Phonebook**. **Phonebook** allows the user to store the name and the number of another person; either it is family, friend, or co-worker. It’s almost the same like the original Contacts, expect **Phonebook** has different design and implementations.

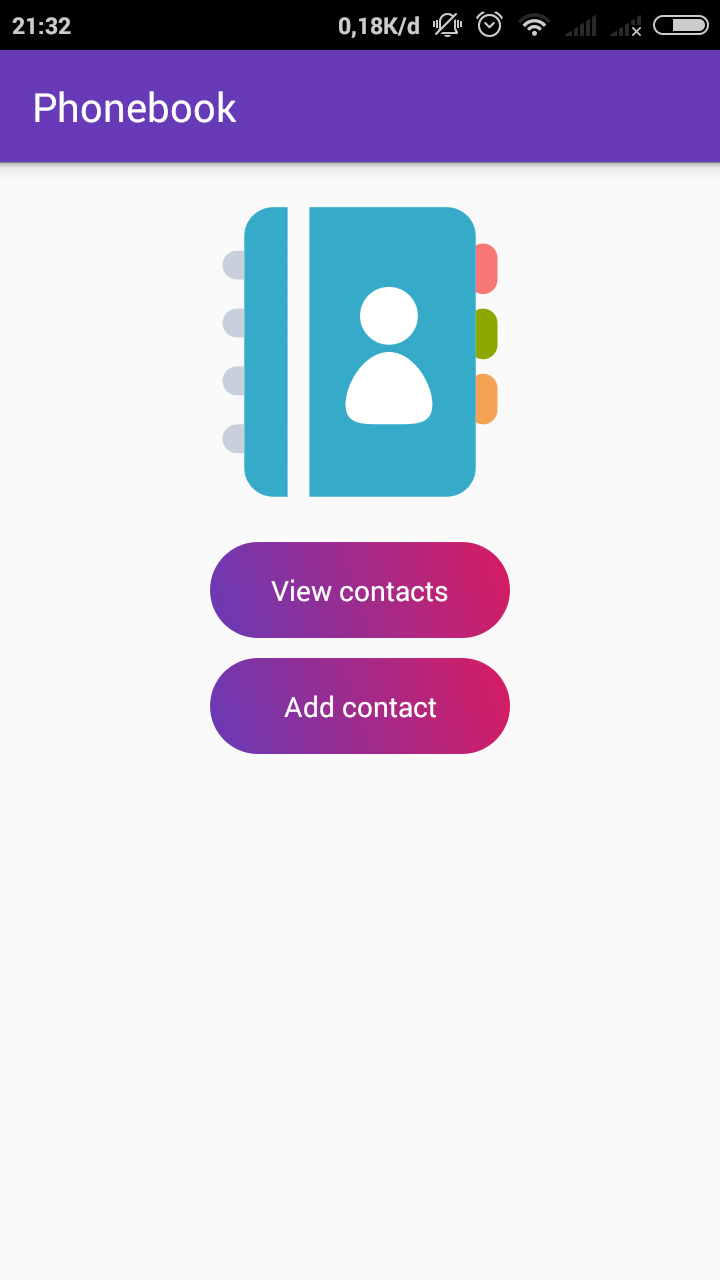
**Procedure**

Below is an explanation (with screenshot) about how the application works:

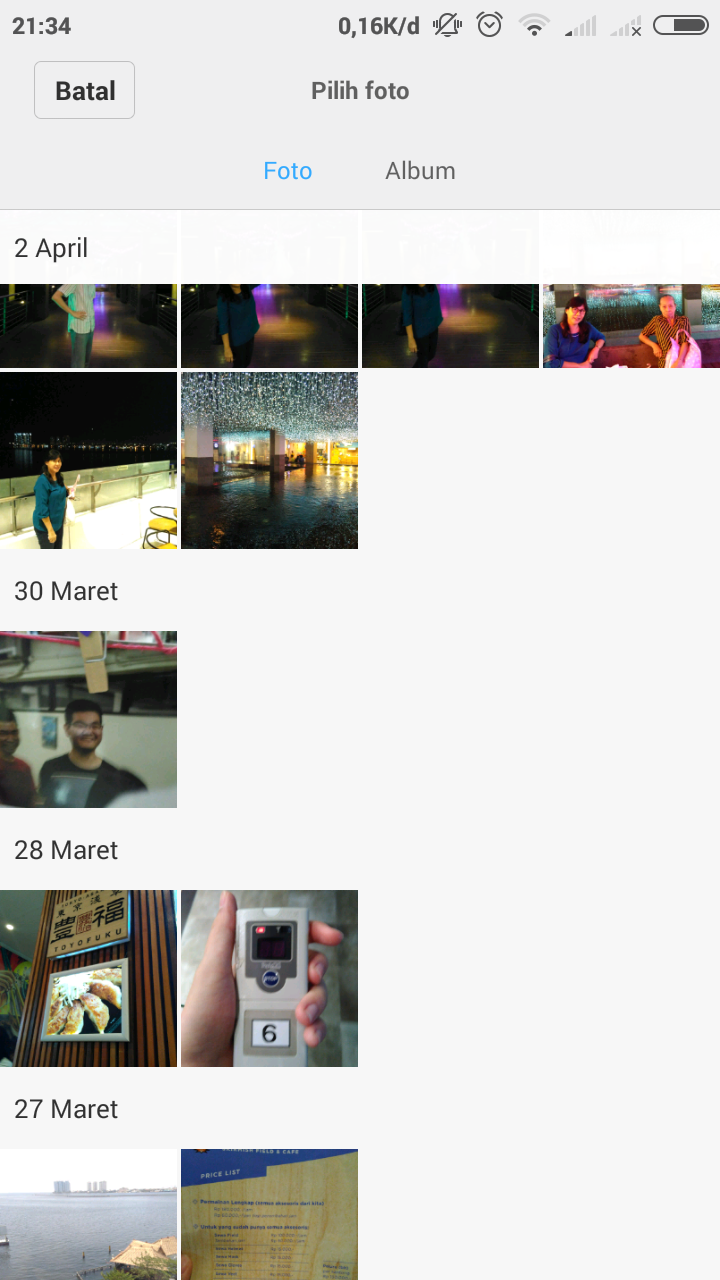
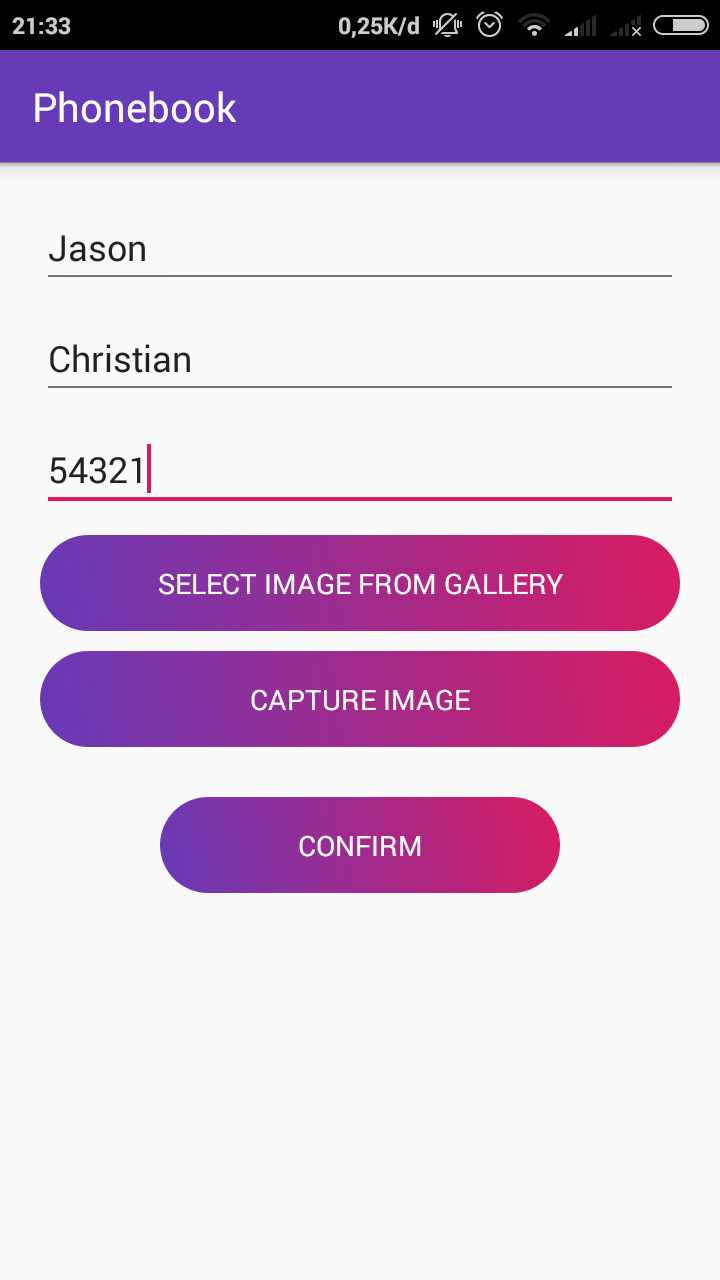
First, choose the **Phonebook** application from the main screen (Make sure that the APK in the computer has been transferred into your mobile device).



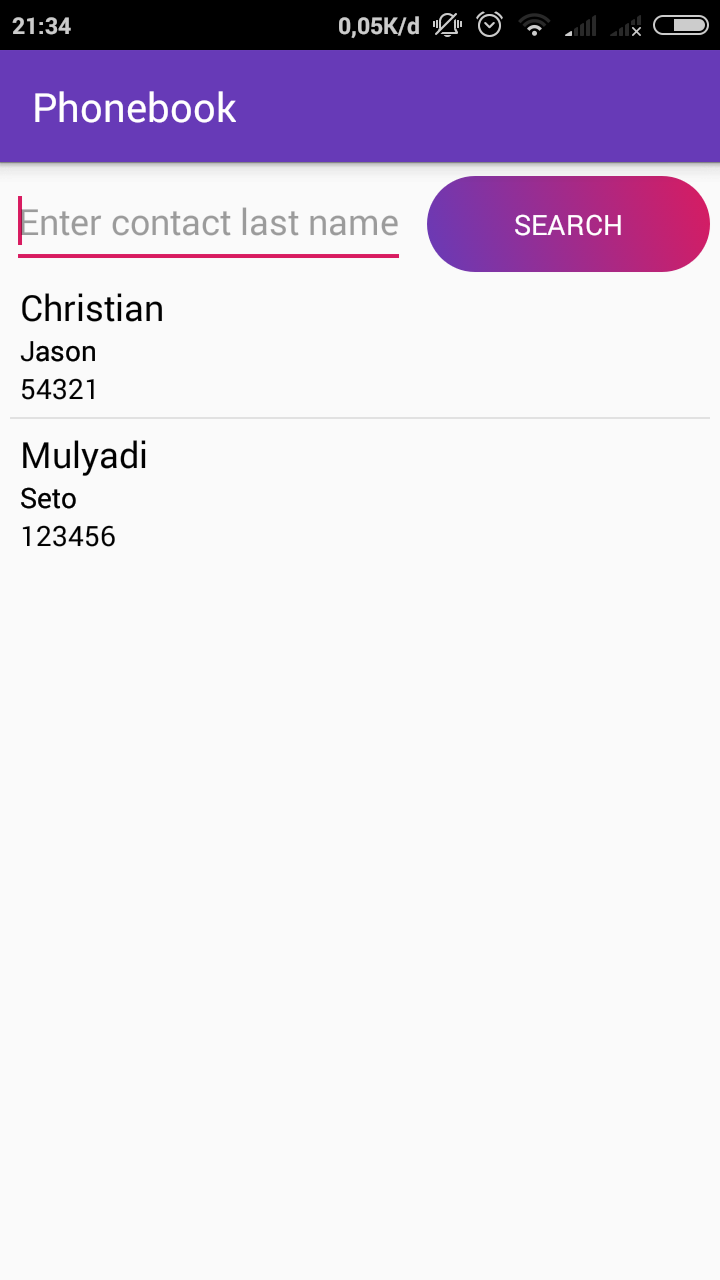
Then, choose ‘Add contact’ Button to add contact.



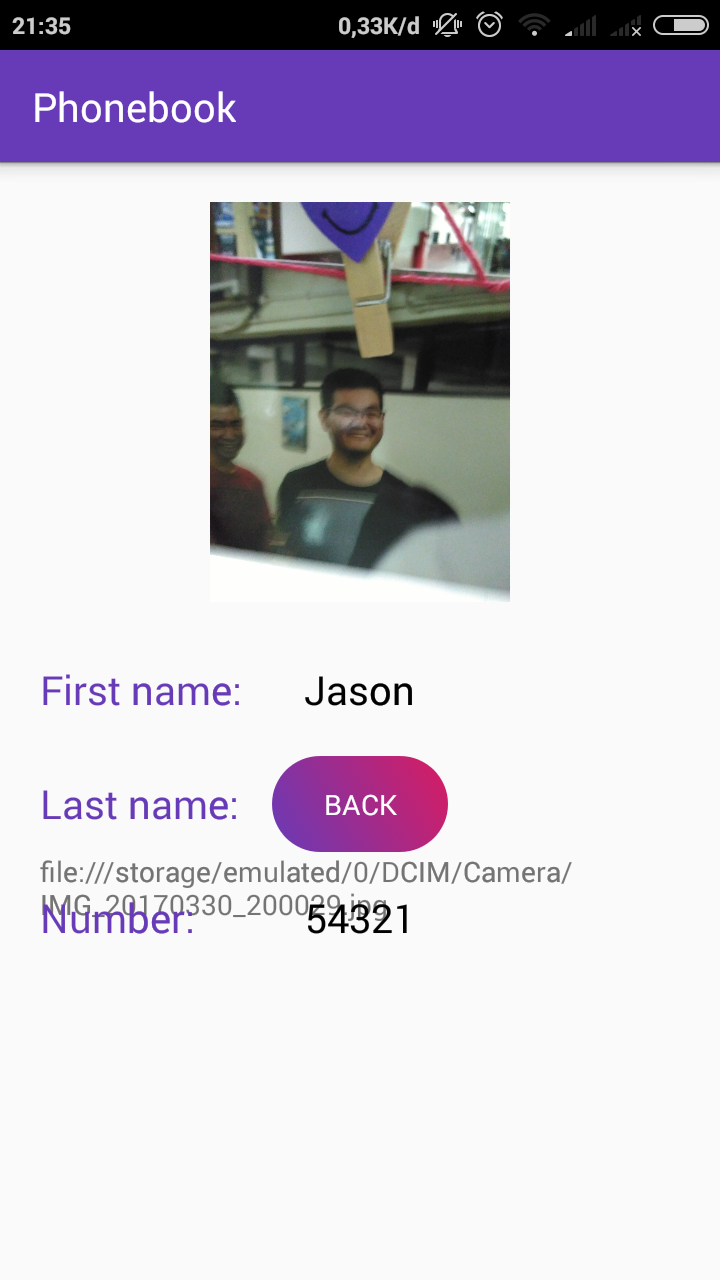
If ‘Add contact’ button is clicked, ‘activty\_add’ screen will be called. In this screen, enter a first name, last name, and phone number of a person. For the image, either choose picture from Gallery or take the current picture with Camera.



After that, go back to the main list. Click ‘View Contacts’ button to check the list. The list has been added with one new contact.



Once the new contact is clicked, it will move to activity\_details screen\* and give you the information about the person full name and phone number, including the picture.



\**When opening ‘activity\_details’ in Steven’s mobile device, it may be messy because the Android version doesn’t match the application’s requirement. The recommended requirement for this application is Android version 5 Lollipop, while Steven’s device is running Android version 4.4 (KitKat).*

**Coding & Explanation**

In this project, there are three main classes, which are ‘**ContactContract.java**’, ‘**AddActivity.java**’, and ‘**DetailsActivity.java**’. The classes can be seen inside the ‘The AndroidManifest’, which means they are connected with one another. The ‘ContactContract.java’ class functions to store table and column name as constants for better readability (Line 10-18). ‘AddActivity.java’ functions to add contacts to the application. There is a code to take image from camera (Line 47-52), take image from gallery (Line 55-62), handle photo from gallery if code is SELECT\_PHOTO or from camera if code is CAPTURE\_PHOTO (Line 75-101), add constraint (Line 113-127), open database in writable mode (Line 129), store values that will be inserted in database with content values (Line 134-136), and insert values and return id of new row (Line138).

‘DetailsActivity.java’ shows user the detailed information about another person first name, last name, phone number, and profile picture. This class able to take all values that are sent from ‘ViewActivity.java’ (Line 43-46), convert image path back to URI\* (Line 48), set image with given URI (Line 49), and see the path image (Line 55), which later will be discussed more in ‘**The Presentation**’ section.

Intent\* plays an important role in this code as in AddActivty.java, intent (Line 144-145) is used to save and trasnfer the data given by the user. So does intent in DetailsActivty.java (Line 38-46).

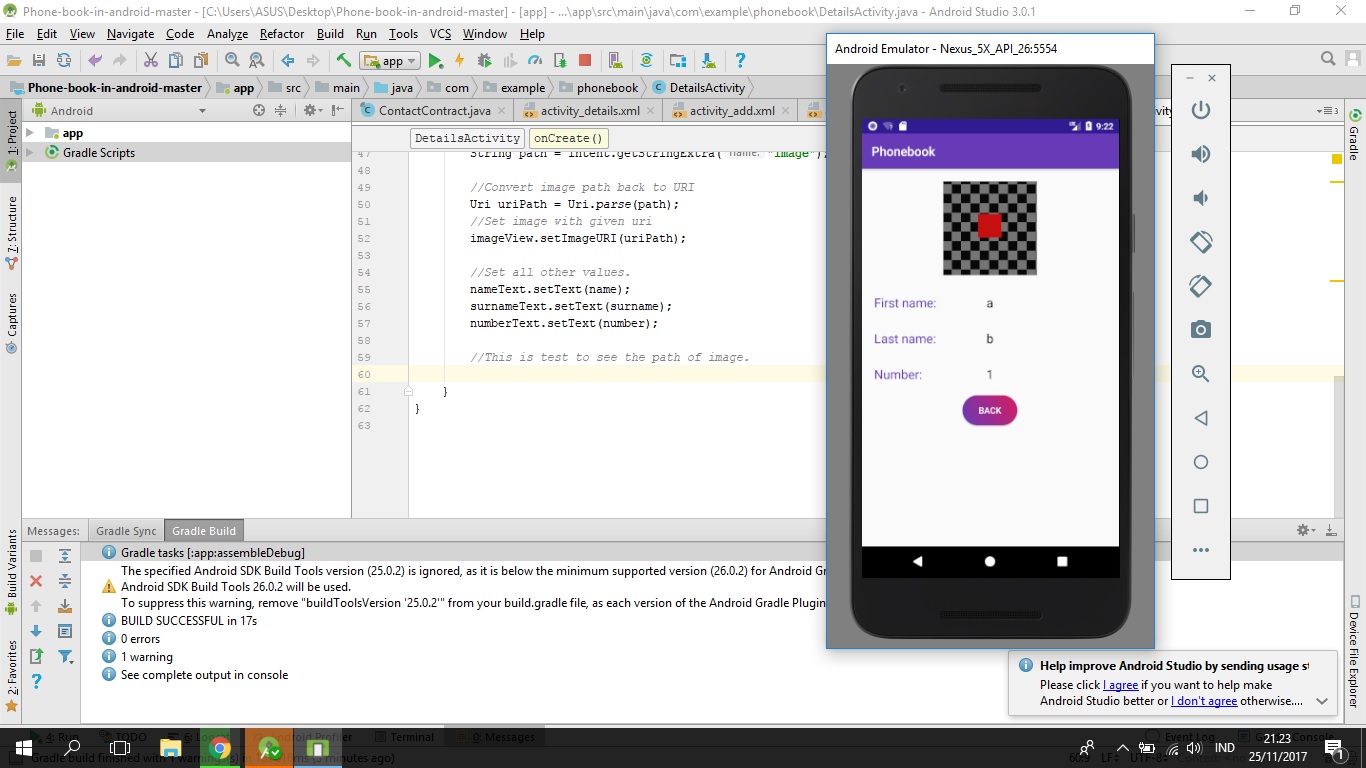
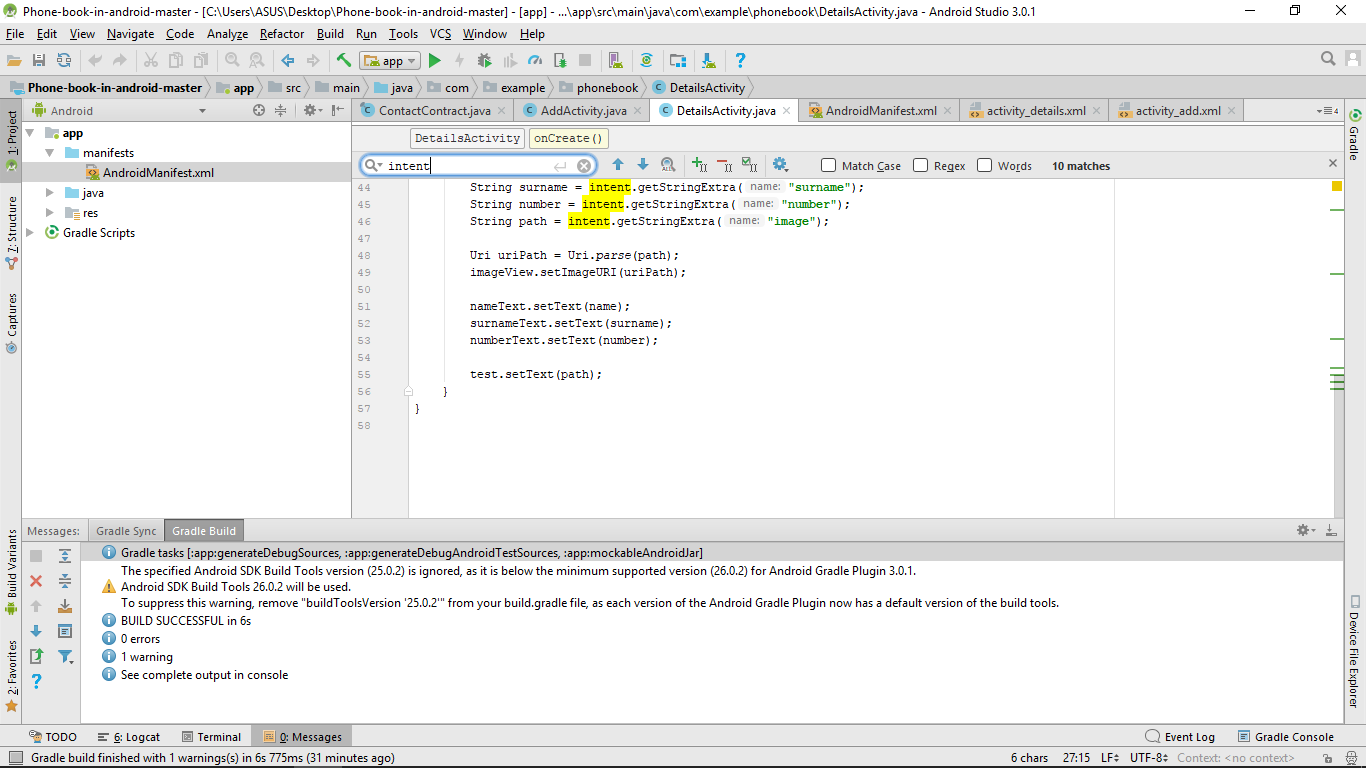
\**An Uniform Resource Identifier (URI) can be further classified as a locator, a name, or both [3].*

*\*Android Intent is the message that is passed between components such as activities, content providers, broadcast receivers and services [4].*

**The Presentation**

Our lecturer asked about one of the key components in our project during presentation. Sadly, we are unable to answer it because we never digress and side questing the project deeper. However, once we have enough time to dig deeper into the code, we are able to find the answer for the question.

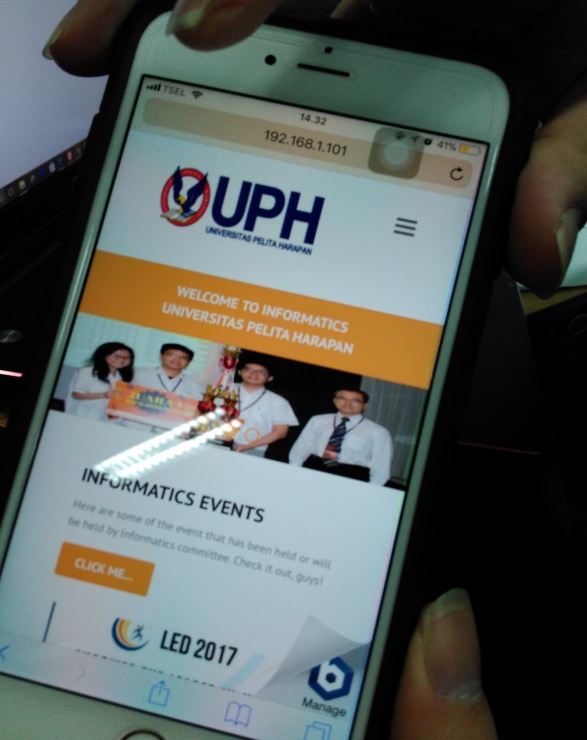
Our lecturer asks about the “**word**” below the purple Back button. This word is the place or storage where the picture is taken. As we examine the code, this “**word**” comes out from ‘**DetailsActivity.java**’. The lowest point of the code, there is “test.setText(path);” in line 55. This is the function that make the “word” appears, and by deleting the function, the “**word**” no longer comes out. This can be seen in the figure below.

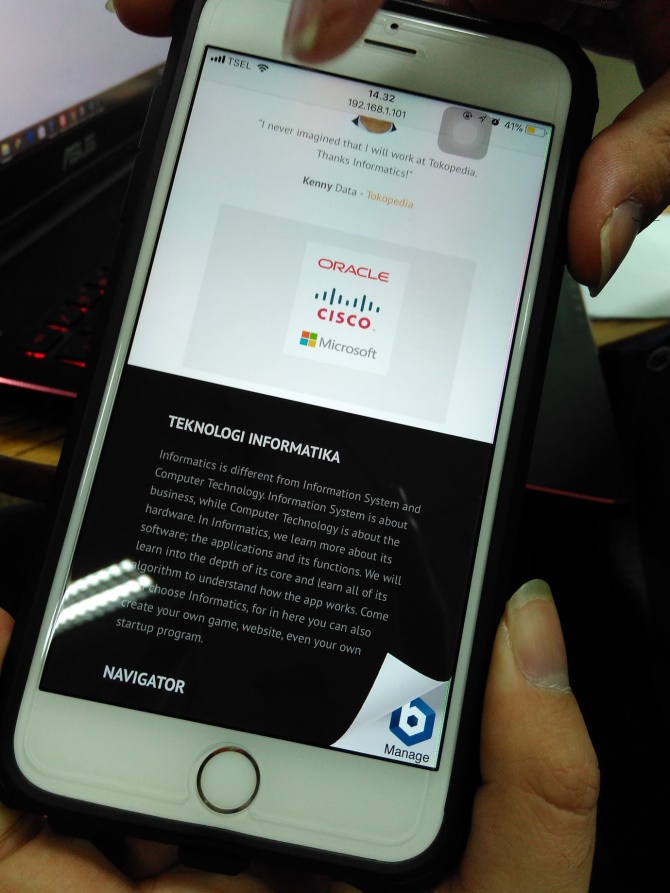


When the line is deleted, it become like this

**The Website**

In relation with our Web Project, we have created a way to open our website in the mobile device. Using Virtual Box (with Bitnami’s component), we are able to transfer the website to our mobile device [5].

****

****

**Limitation Consideration**

There are a lot of limitations that create the project unfinished or incomplete. Firstly, let’s take a look about the group. We both need to be honest, we both aren’t the best programmer and we are aware of our coding skill. We also busy creating other projects given by another lecturer from another programs, creating a messy time-schedule. With this problem, we are unable to create this project fully. With that said, our project, **Phonebook**, has a lot of flaws and not yet perfect. One of the flaws is the design which is not appealing and too simple. Another flaw is the format in activity\_details screen, which already explained in ‘**Procedure**’ segment.

**Reference**

[1] – <https://www.goodreads.com/quotes/183896-man-is-by-nature-a-social-animal-an-individual-who>

[2] – <https://www.gcflearnfree.org/computerbasics/mobile-devices/1/>

[3] – <https://tools.ietf.org/html/rfc3986>

[4] – <https://www.javatpoint.com/android-intent-tutorial>

[5] – <https://docs.bitnami.com/virtual-machine/get-started-virtualbox/>