# Program Algorithm.

This program will start at an activity where the program will ask which type of user is the one currently using the program. This program only recognizes two types of users: student and teacher. If the user identifies him/herself as a teacher, then the program will ask the user to input 4 digits passcode (default passcode is 2212). The program will output a message if the user entered a wrong passcode or empty passcode through toast. Once successfully login, the program will then prompted which administration task the user wants to do (Create a student, View Student, View Student Test Results, and Send Student Test Results).

On the other side, if the user identifies him/herself as a student, then the next stage is identifying the student's name. The user will be able to identify him/herself by clicking the spinner and picking the right name. After clicking the button, then the student will be transferred to the test page. To start the test, the student must click the start button located on the top right corner of the screen. Once the test started, the student cannot go back. In other words, they must click the finish button to stop the test. Once the test is finished, the student is transferred to a page where it shows the statistics of the test (score, time taken, date started, etc.)

Create a student page that will prompt the teacher to input basic information about the student such as Image, First Name, Last Name, Emails, and phone number. The student image can be set through three methods. The first method is by taking a picture through the phone camera, the second method is by importing an image from the phone files, and the last method is by importing an image from an external website (PixBay). The teacher can also prefill the student information by importing the data from the phone's contacts list.

View Student and View Student Test Results pages will show the list of students registered and the list of test results respectively. These data are stored in the internal database. The teacher can only remove and edit the registered student that is listed on the View Student page. To remove, the teacher simply presses the remove button. To edit, the teacher must press the "view more" button and then the teacher can simply make the necessary changes. The teacher cannot remove or edit any test results that are shown on the View Test results page.

# 3RD PARTY API

In this program, I decide to use two different APIs: Commons-io version 2.11.0 and commons-validator version 1.7. These two 3rd party APIs are from Apache Commons Library. Apache's Commons-IO library is a library of utilities to assist with developing IO functionality. I use the Commons-IO library in this program to handle test data from the node.js server. This is because the conventional java IO code to download the data is long and complex (requires a while loop). Using IOUtils.toString() method from the Commons-IO library reduces the complexity of the program (Only need one line of code).

On the other side, Apache's Common-Validator library is a library that verifies the integrity of the data from the user input. I use this library to verify the email the teacher inputted when registering a student or

adding an email to an existing student. I could implement my email validator method. However, it requires a complex pattern of RegEx in which if I miss a single character on the RegEx pattern, it would not validate the inputted email correctly which causes some invalid emails to be valid.

## References

1. Apache Commons. 2020. "Validator – Commons Validator"
   https://commons.apache.org/proper/commons-validator/
2. Apache Commons. 2021. "Commons IO – Commons IO Overview"
   https://commons.apache.org/proper/commons-io/