

Akash Ramjyothi

Chennai, India

akash.ramjyothi@gmail.com

<https://www.linkedin.com/in/akash-ramjyothi/>

<https://github.com/Akash-Ramjyothi>

PH: +91 8939928002

Resume: <https://drive.google.com/file/d/1B-R3d94YU0qWgR7o3t7uRWKA0JWvnkzc/view?usp=sharing>

OpenMRS

Android Client 2.9.x Project

Google Summer of Code 2020

About Me:

I am currently in the 1st year of my Bachelors in Technology Degree from SRM University in Computer Science Engineering. I have been tinkering with code since school days. I love working on Android Apps, as well as the Android OS (AOSP), and have spent many of my nights up hacking on android projects. I also regularly participate in a lot of hackathons and have won a few of them in the Chennai circuit.

Overview:

I wish to contribute to the development of Android Client 2.9.x Project. There are a number of existing problems to be resolved, as well as multiple planned features that I plan to work on.

Work Experience:

- **Android Apps built -**

- **Airports Authority of India QR Scanner:**

This app is built on Kotlin . It uses the system's camera and ZXing library to scan a QR code and access an information which is stored in an online server

Link: <https://github.com/Akash-Ramjyothi/Airport-Authority-of-India-QR-Scanner>

- **LibreHealth Android App:**

This is an android app which uses ROOM database to store and also to validate the details stored in it. It captures the ESSENTIAL ELEMENTS of NEWBORNS and stores it .

Link: <https://github.com/Akash-Ramjyothi/Libre-Health-Android-App-Project.git>

- **Simple Quiz App:**

This app is built on flutter for better UI/UX. This app creates a new database on MongoDB for every new user. It retrieves simple quiz question and answers from online and displays in better interface.

Link: <https://github.com/Akash-Ramjyothi/Simple-Quiz-Android-App>

- **DSC SRM To do List:**

This app is built on swift specifically for iOS and it is just a simple to do list app which is integrated with google sign in.

Link: <https://github.com/Akash-Ramjyothi/DSC-SRM-ToDo-List-App>

Certifications/Achievements:

1) Google Code-In 2019

I worked for the organization “OpenMRS” to make their Android Client 2.8 more faster and accessible.

2) Developing Applications with Google Cloud Platform

Credentials:

<https://www.coursera.org/account/accomplishments/specialization/certificate/SC4E25MX266E>

3) Amazon Web Services Fundamentals

Credentials:

https://www.coursera.org/account/accomplishments/certificate/4L3MCYV9RG9H?utm_medium=certificate&utm_source=link&utm_campaign=copybutton_certificate

4) Google Cloud Platform Big Data and Machine Learning

Credentials:

https://www.coursera.org/account/accomplishments/certificate/H43GQ2ZR3UXP?utm_medium=certificate&utm_source=link&utm_campaign=copybutton_certificate

Goals

1. **Material UI design:** Making a synergetic UI for the whole android app on Adobe Illustrator which includes all transitions and animations from the login credentials to the final storage of the dataset.
2. **Integrate more components of the webapp to the client :** This can be done by using the WebView Tool in Android Studio or to import all assets from the WebApp into the Android and Editing it as per needed.
3. **Password reset via email :** This feature can be done by using the Google Sign In API which would allow only certain users to access specified features and also maintain the records easily. The reason for using this particular feature is that it makes password reset by email easier and faster which generates a random OTP when requested.

4. **Offline Database :** In case the device is not connected to a network, SQLite will be used to store the data temporarily on the device.
5. **Prevention of SQL Injection :** Classifying and using framework by importing DBHelper tool to prevent SQL Injection on the offline dataset stored in the SQLite database.
6. **Replace Java code with Kotlin:** This is done by using the suitable predefined library provided by Android Studio IDE with respect to the JAVA references used to convert into KOTLIN code without any NullPointerException errors.
7. **Automated Testing:** Automated testing the android app using Google Services on Android Studio to find all bugs which can be fixed.

Idea Description::

The goal of the Android client is to provide an alternative to access a hospital's OpenMRS instance by just using the provider's Android devices. Considering the services offered by a hospital, an Android application can help doctors, patients and other staff a lot with its mobility and ease of use, without them having to start the OpenMRS web app on a desktop computer. This will improve the productivity and efficiency of the hospital workflow. Also, offline first solution needs to be developed, where a provider cannot connect to the internet, and needs to work offline temporarily. Data should be synced again when back online. The current user interface is slow to respond to user inputs and does not look pleasing to the eye, compared to the newer and modern Android apps in the google play store. E.g. 'Undo' buttons (that takes the user back to the previous state if he/she errors) should be available for any operation. So we should enhance the feel of the app so that the user knows that he/she is in control every time. Finally, seeing the number of components in the webapp, the Android app should integrate more of them so that doctors and staff can expect to see the same functionalities in the webapp implemented in the client.

Availability:

I would be able to devote **approx 40 hours** every week to GSOC. During 15th to 20th May I will have my end-sessional examinations, and time devoted would be around 2 hour per day for that time.

I have no obligations after May, and would be devoting all my time to GSOC.

I will be reporting my work over daily scrum mails on the mailing list. ***I would strive to be regular, and sincere with my scrum and daily updates.***

Implementation Plan:

➤ **Development of the Android Application:**

The android app will consist of the front end made on XML and the functionality on KOTLIN/JAVA which will include all the screens, layouts and fragments with separate modules for specific tasks.

➤ **Creating endpoints for the project:**

The offline endpoint to store data temporarily which will transfer data into the cloud when a network is connected will be SQLite, this is achieved by using the class DBHelper to link the app components to the specified field of the endpoint

➤ **Developing Communication Medium:**

A communication between the android app and the endpoint whether it is in a network or offline storage will be done by the use of REST API which will be implemented in the app by the use of RETROFIT library

Timeline:

➤ **Bonding Time**

Discuss with mentors and org admins about roadmap, and finalise the plan of action and features to be implemented

➤ **Week 1**

I would use this period to create functionalities of the android app such as

- A. Implementation of the Overlay development of Android Client.
- B. Document the generation of credentials process.

➤ **Week 2**

- A. Improve themeability support on Android App.
- B. Developing the Material UI on the SCREENS/ACTIVITIES which have been integrated the WEBAPP components.
- C. Improve UI on the Android App, adhere strictly to Material Design guidelines.

➤ **Week 3**

- A. Adding all the discussed features on the app.
- B. Remove UI glitches like white text on white page, black line under status bar etc.

➤ **Week 4**

- A. Adding Notification features whenever there is an update on the datasets
- B. Developing the working Prototype of the app for demonstration at the end of Week 4

➤ **Week 5**

This week will be used on a larger set of work - that is to port the android app to a better, modern, responsive and fast framework. We can weigh the following options -

- A. Decreasing StartUp time by running few background tasks and reducing bitmap sizes.
- B. Parsing the data to process data the user is not going to see on the screen.

➤ **Week 6**

- A. Network Requests by keeping endpoints simple and fast
- B. Usage of Signed Tokens for user authentication

➤ **Week 7**

- A. Configuring app to dynamically allocate memory instead of static allocation to preserve and occupy less space.

➤ **Week 8**

Development and implementation of POST calls with references to create and access information on the database using predefined functions with objects of the linked database.

➤ **Week 9**

Development of REST API in case the developed POST call with references could not accomplish certain tasks like customized retrievals.

➤ **Week 10**

- A. Integrating the whole android app with a database or cloud and testing the android app with both customized and random inputs with generation of valid credentials and verification of required Fields from the database.
- B. Using POST calls with references or REST API to accomplish easier and faster access of data to the database
- C. Demonstration of the fully functional app

➤ **Week 11**

- A. Finding all visible bugs
- B. Bug fixing

➤ **Week 12**

- A. Automated testing of the app
- B. Quality Checking after fixing all bugs

Contributions to OpenMRS Community:

- 1) Worked on Minor Bug Fixes on the Open MRS Android Client 2.8 as a part of my GSOC 2019
- 2) Contributed on Content Creation as a part of Google Code-In - 2019 for OpenMRS

Why me ?

Other than being an Android developer for over 3 years, and a passionate open source contributor, I believe that I am well suited for this project, because I have already developed android apps for my college, and I already know very well about the project vision.

Also, previously with a collegemate, I had developed a mobile application for college HACKATHON

I have developed a similar app as a part of Google - Code In - 2019 for the organization "OpenMRS". I am a technical member at Microsoft Student Partner's Club at my University. It has been my dream to contribute to open source organizations as I use their softwares frequently. I also wanted to improve my current technical expertise. I am proficient in Kotlin, Java, Swift, Flutter. I have also developed a similar app for my college to easily input and retrieve student's marks whenever needed using MongoDB.

I have also worked on a personal project called "University and Shuttle Bus tracking system". It is an android app which gets Latitudes and Longitudes as input and uses a REST API to transfer those data into MongoDB and an android app which uses GOOGLE MAPS API to continuously track the bus.