Mifos Initiative

Android Field Operations App Version 6

Project Idea:

Field Officer Application is an Android application developed for the Bank staff field officer to keep track of Clients, Centers, Groups, Loan Account, Savings Account etc. Several versions of the application were released for field officers to go out in the field to process transactions, create clients, groups etc. Currently Mifos Android Client is in its Version 5. The app is still under development and this project aims at the release of Version 6 of this app.

Current Status of the project:

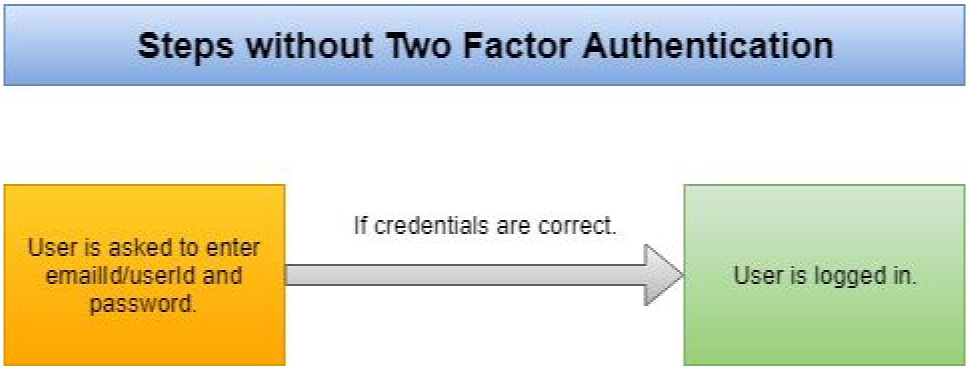
Mifos Android Client is currently in its **Version 5**. Recently many new significant functionalities and enhancements have been added. These include enhancement to various **Create 'Entity'** fragments, **Document Uploading Dialog** fragment and **Identifier Dialog**. The **Individual Collection Sheet Fragment** has been enhanced to three fragment multi-stage layout. Other features which are added in the previous version include **Automatic background sync** of client, center, group, Loan Repayment and Saving Repayment Payload as well as Support for various **Report types** like Client, Loan, Savings and Fund accounting.

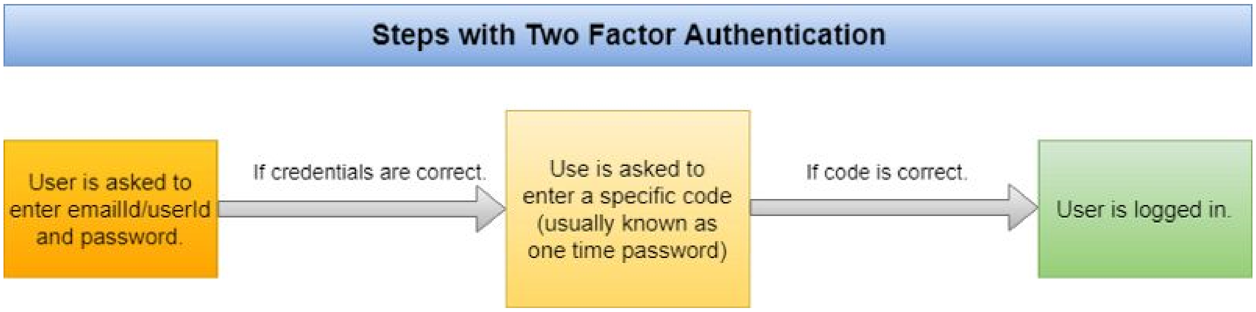
Goals:

- Integrating SMS communications
- Deeper Integration of the notifications framework
- Implementation of T-OTP based Two factor Authentication using Google Authenticator
- Redesign of UI and workflows
- Continued refactoring and performance enhancements
- Enabling Views for task list and reports
- Improving the User Interface for capture of surveys
- Unit and Integration Testing

Implementation:

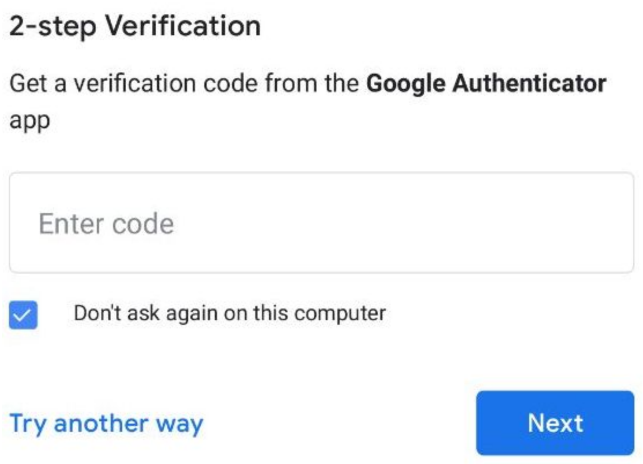
- **T-OTP based 2 factor Authentication:** Two factor authentication provides an extra layer of security for a user’s account.


Steps without Two Factor Authentication
User is asked to enter emailId/userId and password.
If credentials are correct.
User is logged in.


Steps with Two Factor Authentication
User is asked to enter emailId/userId and password.
If credentials are correct.
Use is asked to enter a specific code (usually known as one time password)
If code is correct.
User is logged in.

By using T-OTP based method, the One time password is creating on the user’s side rather than the server side through a smartphone application (Google Authenticator in this case).

In the Mifos Android Client app, if a user enables 2-factor authentication then only for the first time on a particular device, after login he will be prompted to a screen like this:



2-step Verification

Get a verification code from the **Google Authenticator** app

☑ Don't ask again on this computer

The user then has to enter the OTP generated by the Google authenticator app for that account.

To implement T-OTP based 2 factor authentication, when the user enables it then a secret key needs to be generated.

Pseudocode when user request to enable 2-factor authentication:

```
secretKey = generateSecretKey(20);
saveUserSecretKey(userId, secretKey);
qrCode = convertToQrCode(secretKey);
response(qrCode);
```

Pseudocode when user enters the code displayed in the application:

```
secretKey = getSecretKeyOfUser(userId);
if (codeTypedByUser == getHOTP(secretKey, currentUnixTime / 30)) {
  enableTwoFactorAuthentication(userId);
}
```

If that OTP is the same as one typed by the user then 2-factor authentication will be enabled successfully. Both these code snippets run on the server side. For the implementation, I'll be using this library .

Now after login operation, it will be asking for the OTP displayed on the Google Authenticator application. If the typed OTP is correct only then is the user authenticated.

Pseudocode when user enters the code displayed in the application to login:

```
// Fetch secret key from database.
secretKey = getSecretKeyOfUser(userId);
if (codeTypedByUser == getHOTP(secretKey, currentUnixTime))
 {
   signIn(userId);
}
```

- **Unit testing and Integration testing:** Unit and Integration testing should be performed fluently besides coding as Test Driven Development is quite effective. Currently there are some tests written already for testing presenters, util classes as well as Instrumentation tests which requires Android environment for testing. I will be continuing with this approach and for writing unit tests I will be using JUnit and Mockito. Integration testing can be implemented using AndroidJUnitRunner.

- **Deeper integration of Notification Framework:** From Android 8.0 the concept of Notification Channels is introduced in the notifications framework which are nothing but different Notification categories. This provides flexibility to the user to enable/disable notifications from different categories.

  Currently in PathTrackingService in the project for creating notifications in NotifcationCompat.Builder() method Channel Ids are not used.

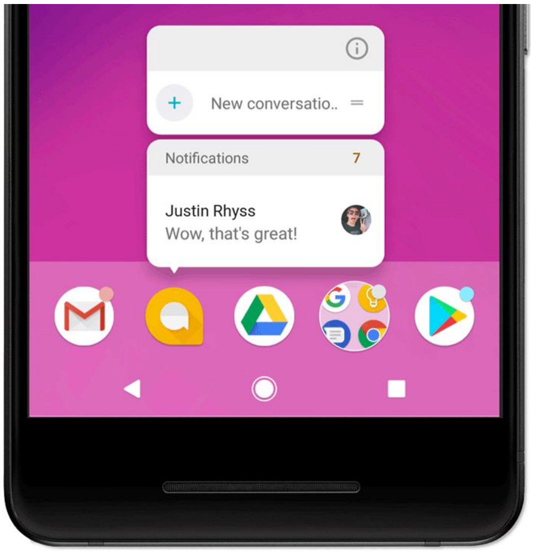
```
public void startNotification() {
notificationManager = (NotificationManager)
getSystemService(Context.NOTIFICATION_SERVICE);
```

```
notification = new NotificationCompat.Builder(this)
.setContentTitle(getString(R.string.mifos_path_tracker))
.setAutoCancel(false)
.setOngoing(true)
.setContentText(getString(R.string.description_location_tracking))
.setSmallIcon(R.drawable.ic_launcher);
```

This can be improved by creating an instance of NotificationChannel class and passing the channel id as an argument to NotificationCompat.Builder() method.

```
private void createNotificationChannel(){
if(Build.VERSION.SDK_INT >= Build.VERSION_CODES.O){
NotificationChannel notificationChannel = new
NotificationChannel(getString(R.string.NEWS_CHANNEL_ID),getString(R.s
tring.CHANNEL_NEWS), NotificationManager.IMPORTANCE_DEFAULT );
notificationChannel.setDescription(getString(R.string.CHANNEL_DESCRIP
TION));
notificationChannel.setShowBadge(true);
NotificationManager notificationManager =
getSystemService(NotificationManager.class);
notificationManager.createNotificationChannel(notificationChannel);}
}
```

App icon badge property can also be added which show a “notification dot” on the app icon indicating a new notification and user can long press on that app to see the notification.



Notifications 7

Justin Rhyss
Wow, that's great!

Further I am planning to add notifications whenever a new ‘Entity’ (Client, Group or Client) is created. The steps include:

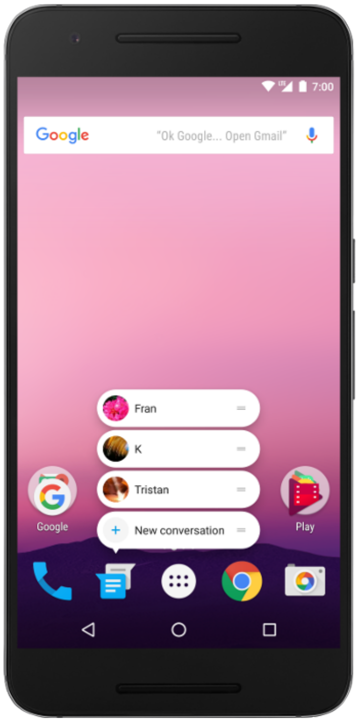
- Registering the Notification Channel (Explained above)
- Notification Construction (Using NotificationManager and NotificationCompat.Builder() to create and define various attributes of the notification)
- Notification Navigation (Using NotificationManager and PendingIntent to control the action when notification is clicked.)

- **Integration of SMS communications:** This includes sending in-app push notifications from the Mifos Android-Client Field Officer app to the client users. Notifications will be sent to the clients in 3 cases:
  - When the user applies for a loan and loan gets approved
  - When the Field officer wants to send a notification the client will receive it at the scheduled date and time
  - When a loan payment is overdue or a Happy Birthday notification to the user on his birthday

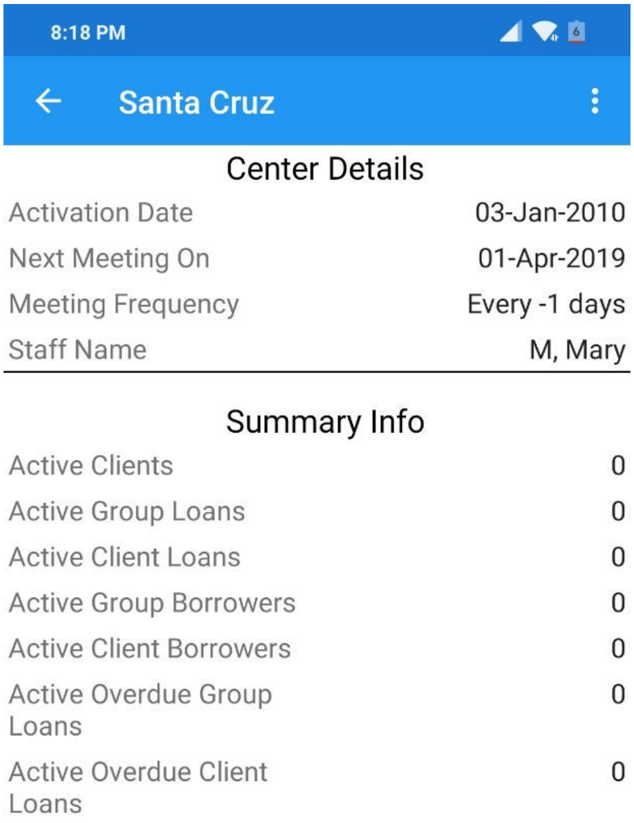
  All these functionalities can be implemented free of cost using Firebase Cloud Messaging (FCM) which can be used to send push notifications to the client app. The steps for integrating FCM to an android app can be found here.

- **UI Redesign and Improvements:** Several UI components of the app can be improved. For example, instead of simple EditText widget in the Login Page we can use TextInputLayout which can add more features to the EditText like hints of the EditText can be turned into Floating Labels.

  Another UI improvement that I can suggest is to add App shortcuts which can give a quick access to the key features of the app. App shortcuts were introduced in API 2. Each shortcut references one or more intents, each of which launches a specific action in your app when users select the shortcut. Most important features can be added as App shortcuts after discussing with the mentors. More information can be found here.



  The UI of details screen of clients, centers and groups can be improved by adding proper paddings/margins making the information shown in them more readable.



Santa Cruz

Center Details

Activation Date	03-Jan-2010
Next Meeting On	01-Apr-2019
Meeting Frequency	Every -1 days
Staff Name	M, Mary

Summary Info

Active Clients	0
Active Group Loans	0
Active Client Loans	0
Active Group Borrowers	0
Active Client Borrowers	0
Active Overdue Group Loans	0
Active Overdue Client Loans	0

Timeline:

May 6 - May 27 (Community Bonding Period):

During the first week of this period I will be busy with my University final exams.
I will be utilising the rest of the time discussing my implementation goals with the mentors. This period would give me opportunity to get to know the mentor I would work with during the summer and the application deeply focusing on parts where my goals can be implementing. I will also use this period getting more familiar with the codebase as well as Mifos Platform APIs.

May 27 - June 28 (Phase 1 coding):

Week 4 -5 (May 28 - June 11)

In these two weeks, I will be working upon redesigning the UI and workflow of the overall app. It will include giving more material look to the app, implementation of App shortcuts etc. I will also improve the user interface of Capture of Surveys during this period. These UI improvement tasks will help me in gaining more confidence and I will be able to work upon rest of my goals more confidently.

If these tasks are completed before this duration, I will be using the rest of the time reading about the implementation of T-OTP based 2-factor authentication in detail.

Week 6 - 7 (June 12 - June 26)

For this duration I will be focussing my attention towards the implementation of T-OTP based 2-factor authentication using Google Authenticator. I will be going through several online guides to get a clear view of how to implement this. I will first create a sample app and then try to implement this in that sample app using this library . Once done I will implement the same in Mifos Android Client app.

May 27 - June 28 (Phase 2 coding):

Week 8 - 9 (June 28 - July 13)

In the first two week of Phase 2 coding, I will be working on deeper integration of Notifications framework.

This will include:

- Creation of Notification channels
- Replacing the deprecated NotificationCompat.builder() by adding Channel ID as second argument
- Addition of badge property so that user can see important notifications by long pressing the app icon
- Triggering of Notification in case of events like ‘Entity’ creation etc.

Week 10 - 11 (June 14 - July 28)

After the successful integration of notifications framework, my goal for this duration will be to work on Integrating the SMS communications. For implementing this I will be using Firebase Cloud Messaging service.

This task will be bit challenging for me as it will require good knowledge of Fineract backend for knowing when a loan gets approved, birthday details of clients and things like that and send notifications to the client app from Field Operation App. But I will try my best to accomplish this and if I will get stuck somewhere I won’t hesitate asking mentors for help.

July 28 - August 19 (Phase 3 Coding)

Week 12 - 13 (July 29 - August 12)

For this time I will work on enabling views for task list and reports.

During this time I will also be writing unit and integration tests for the apps ensuring fully functional and bug free application. I will also try to fix any left over bugs and any performance improvements by checking for any memory leaks and other insignificant resource utilisations in the app.

Week 14 (August 13 - August 20)

I will devote this time for writing documentations of my work as well as improving the previous documentations of the project. Writing proper and well defined documentations will help other developers in the community to have a clear understanding of my work. This way I will be able to explain my work more clearly and it will help future developers who will be interested in this project to proceed further.

Why are you right person for this project ?

The Android Client for Mifos X platform is completely written in Java. Moreover this project requires a good understanding of Android Architectural patterns. This project uses MVP which allows separating the presentation layer from the logic.

I am fluent in Java and basic Android components like Activities, Fragments, Content Providers, Services, Broadcast Receivers etc. and good with Android Architectural patterns. I have developed this app Scrawl for making notes for day's work which is based on MVVM (Model-View-ViewModel) architecture. I also have a decent knowledge of MVP architecture which this project uses. In addition to these, I have also worked with networking libraries like Retrofit and ButterKnife which is a data binding library. These libraries are used within this project. I am also getting familiar with Mifos X platform.

I have worked extensively with Firebase API in my past projects, which might be helpful in this projectin implementing Push Notifications feature.

During contributing to Mifos Android Client project, I have also helped some fellow developers/contributors in the Gitter channel. Initially I was having trouble in setting up the project in my Android Studio. But after trying again and again and with the help of community members in the Gitter channel, I was able to set up the project successfully. I also helped other newcomers who are having difficulty in setting up the project and are facing the same problems that I had faced. I also helped some new community members in resolving errors due to which their Travis CI build is failing.

I have gained a considerable amount of familiarity with the codebase by submitting pull requests to the project related to UI improvements, bug fixes, code clean up, adding language support etc. I have made very informative commit messages which is very important in efficient maintaining of an Open Source project as guided by the mentor himself.

If in a college, Current area of Study

I am a third-year B.Tech undergraduate enrolled in, Computer Science and Engineering course at Shri Mata Vaishno Devi University, Katra (India).

Contact Information and preferred method of Contact

Email id: mokshmahajan008@gmail.com

Gitter Nickname: moksh-mahajan

Phone Number: +91 9086031210

Time-Zone: UTC +5:30

I can be contacted through Email, Gitter and Hangouts.

Career Goals:

I am very much passionate about Mobile Application Development and Android in particular. I am also in initial stage of learning frameworks like Flutter, a mobile SDK by Google which lets you to create Android and iOS apps from the same codebase. I am also interested in BlockChain technology. I am looking forward to become a successful Mobile Application Developer and a long term open source contributor.

Please share any links to source-code you have written or websites you have built

I have developed many Android apps while learning. These apps are built in either Java or Kotlin. My recent work also include some apps developed in Flutter which uses Dart language. Almost all of these apps are available on Github. You can browse them through my Github account: https://github.com/moksh-mahajan

Some of my projects include:

- Scrawl: A simple note taking app for days work based on MVVM architecture and using some important Android Architecture Components like LiveData, ViewModel and Room Persistence library.
- ToothTalk: A single screen Bluetooth chat app developed in Kotlin
- PokeDex: An app developed using flutter framework to turn your smartphone into a Pokedex. It uses HTTP package by Dart to make REST API calls and receive http response from server.

LinkedIn profile: https://www.linkedin.com/in/moksh-mahajan-b25498159/

If you have visited our Gitter chat room, what nickname do you use?

Yes, I have visited the Gitter chat room. My Gitter NickName: moksh-mahajan

Have you Contributed to other open source projects? If so? Which?

I have started my Open Source journey last year. I have been a contributor to Mifos Android-Client project. Other than that, I have also contributed to other open source projects as well.
These include:

Systers - PowerUp Android: https://github.com/systers/powerup-android

Commons Android App: https://github.com/commons-app/apps-android-commons

Open Data Kit - skunkworks-crow: https://github.com/opendatakit/skunkworks-crow

Rocket Chat Android: https://github.com/RocketChat/Rocket.Chat.Android

All contributions are available on my GitHub profile: https://github.com/moksh-mahajan

Have you any previous experience with AngularJS, Android, Java, Spring, Hibernate, MySQL?

Yes, I have experience in Android and Java. I have worked with Android and Java and developed many projects. I also worked with Firebase API and used it for implementing a Real-time Database, adding Authentication and sending Push notifications for a project which I had developed along with my friend for our University Techfest “Titiksha”. I also have a good knowledge of MySQL as well as SQLite database which I recently used in a Note-making Android app for saving and retrieving data locally from and to the device using CRUD commands. I have not worked with AngularJS, Spring and Hibernate though I am familiar with JavaScript. If they are required, I will try to learn them at a fast pace.

What Other commitments do you have this summer? (Working on this project is a full-time job and must be your primary commitment)

I have no other commitments during May to August. Working on this project will be my primary objective. Also, I won’t be having any college due to summer vacations. So, I will be available full-time and can give all my time to Mifos Android-Client project.

Have you submitted any patches or source code to Mifos X yet? Please provides links to the commit in GitHub or the JIRA ticket (students looking to be seriously considered should make at least one submission)

I have submitted my patches for the Mifos Android-Client project. Here are the links to my contributions:

Merged PRs:

https://github.com/openMF/android-client/pulls?utf8=%E2%9C%93&q=is%3Apr+author%3Amoksh-mahajan+is%3Amerged

Open PRs:

https://github.com/openMF/android-client/pulls/moksh-mahajan

Have you previously participated in the Google Summer of Code ?

No, this is my first time participating in Google Summer of Code.

Are you applying to Multiple organisation this year ?

No, I am not applying for any other organisation except Mifos Initiative.

If you answered "yes" to the last question, which is your first choice ?

Not Applicable