



# Apache Fineract CN mobile - Supporting Branchless banking and promoting financial inclusion in the fragile states



Rajan Maurya  
Android Developer @iDT Labs

# **Rajan Maurya - @therajanmaurya**



**Android Engineer @iDT Labs**

**Open Source Developer at  
@Apache Fineract and @Mifos Initiative**

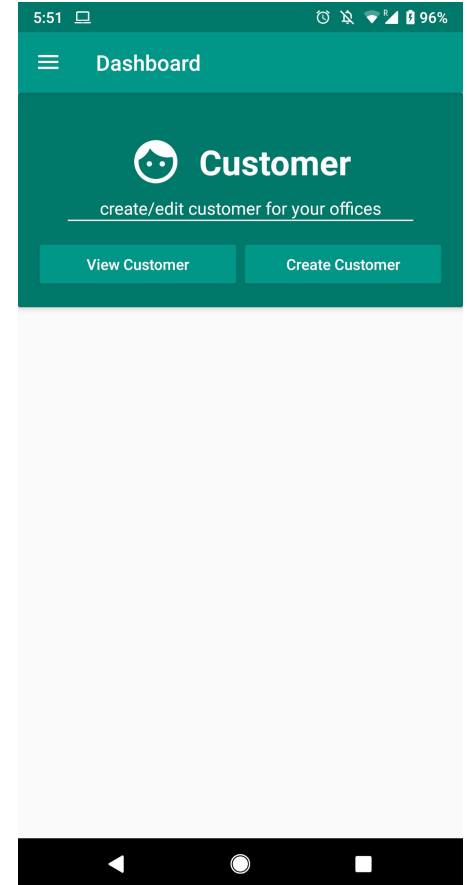
**Google Summer of Code Mentor  
Google Code In Mentor**

# What is Fineract-CN-Mobile?

Github: <https://github.com/apache/fineract-cn-mobile>

Initial Development Blog:

<https://medium.com/@therajanmaurya/apache-gsoc-2017-blog-post-f2791dc136b5>





Search or jump to...

7

[Pull requests](#) [Issues](#) [Marketplace](#) [Explore](#)



apache / **fineract-cn-mobile**

[Unwatch](#) 8 [Star](#) 3 [Fork](#) 54

[Code](#) [Pull requests](#) 6 [Insights](#)

Mobile field agent app <https://fineract.apache.org/>

218 commits 1 branch 0 releases 7 contributors Apache-2.0

Branch: **development** [New pull request](#) [Create new file](#) [Upload files](#) [Find file](#) [Clone or download](#)

	therajanmaurya Chore: Update Build tools and support library version with other libr...	Latest commit 95ab8a8 4 days ago
	.github fix: move files to .github	a month ago
	app Chore: Update Build tools and support library version with other libr...	4 days ago
	config/quality refactor: Mifos name and string with Fineract	a year ago
	gradle/wrapper Chore: Update Build tools and support library version with other libr...	4 days ago
	DS_Store refactor: Convert java models to kotlin data models	4 months ago
	.gitignore Initial commit	2 years ago
	.travis.yml chore: update build tools and gradle version	10 months ago
	AndroidStyle.xml style: Android Code Style Implementation	a year ago
	LICENSE.md Add license	a year ago
	README.md docs: Update Readme	4 months ago
	build.gradle Chore: Update Build tools and support library version with other libr...	4 days ago
	gradle.properties feat: Sync feature for customer and customer details	2 months ago
	gradlew Initial commit	2 years ago
	gradlew.bat Initial commit	2 years ago
	settings.gradle Initial commit	2 years ago

**README.md**

## Status

build **passing**

## Android Client of Apache Fineract CN

Android Client is built on top of the Apache Fineract CN platform.

## Building the Project

1. Install Java Development Kit (JDK)
2. Download, install, and run the latest version of [Android Studio](#).
3. Clone this repository and import it to Android Studio.
4. Once the gradle build is finished, click on the green play button to build and run the app.

## Demo Credentials

# What We have currently



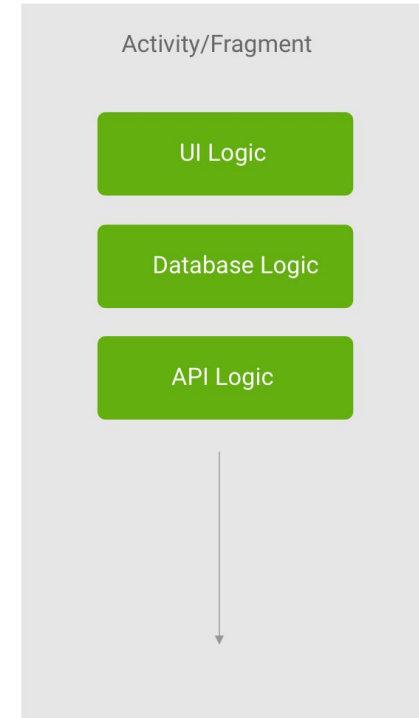
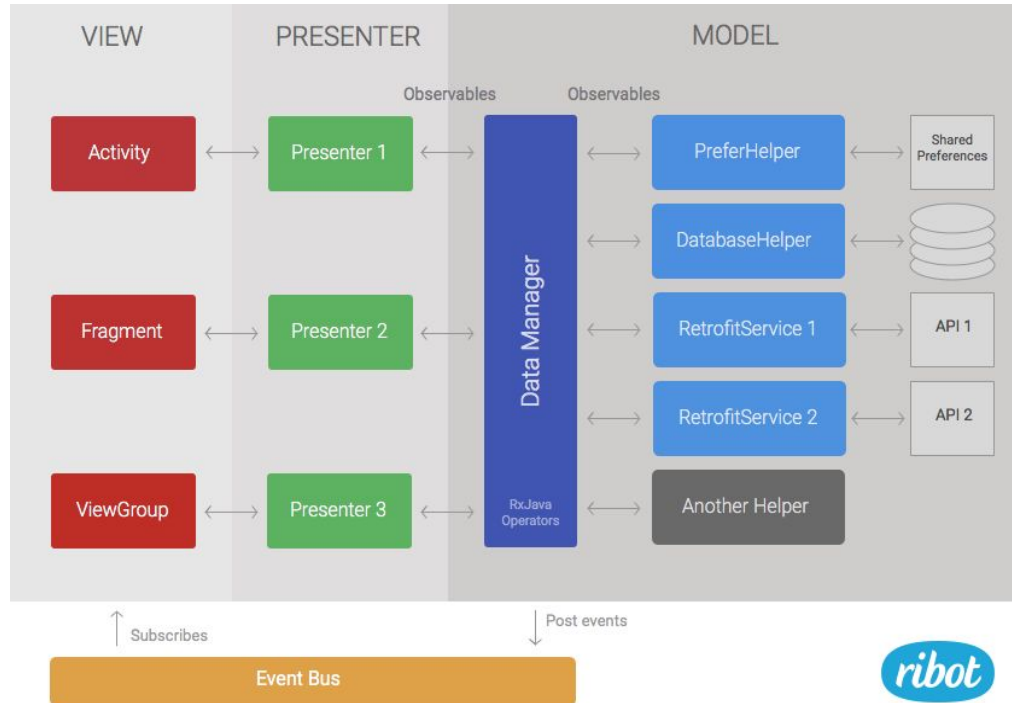
- ❖ **Manage Customers**
- ❖ **View Customer Details and create new customer and Edit customer details**
- ❖ **Manage loan accounts, view and apply for new loan account**
- ❖ **Manage deposit accounts and view and create new deposit account.**
- ❖ **New Identification card and Upload identification scan document.**
- ❖ **Update Customer status and customer profile image.**

# Things we solved that we will discuss



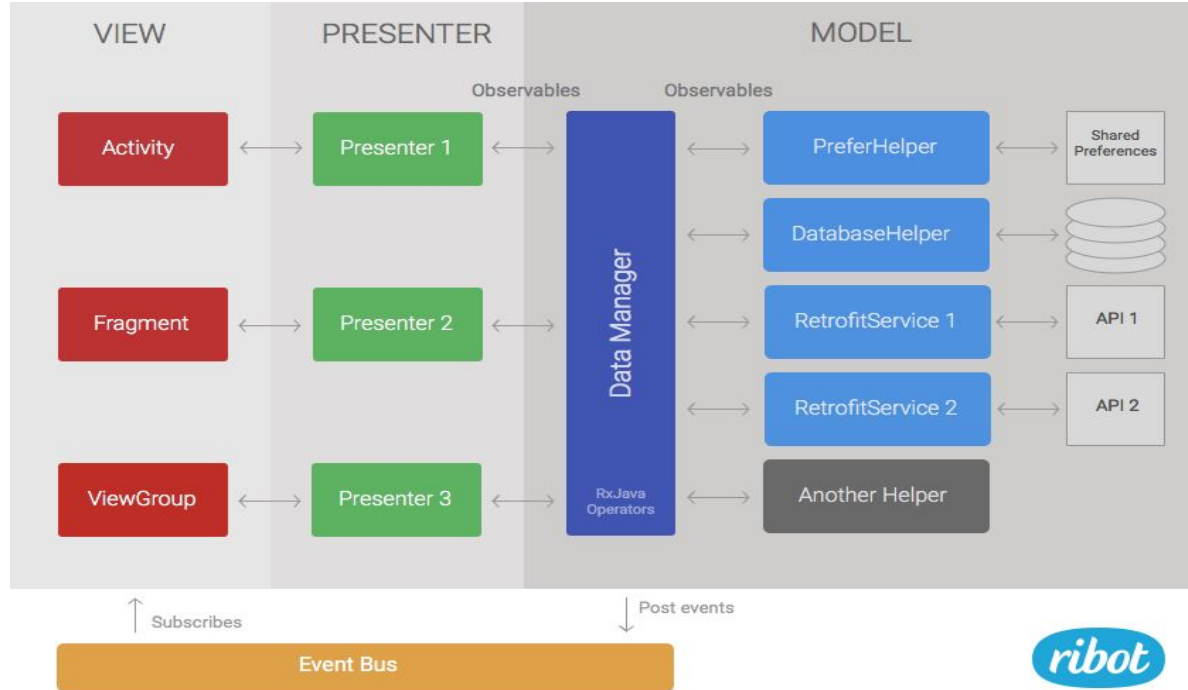
- ❖ **Maintainability and scalability with MVP architecture.**
- ❖ **Advanced level architecture that enables to upgrade advance libraries and give the ability to introduce any high-level codebase layer.**
- ❖ **Advance level security to access the app does not matter user have the internet connection or not.**
- ❖ **Simple and easy learnable UI & UX.**
- ❖ **Work with poor connectivity.**
- ❖ **Offline usage of the app (Many countries like Africa, Indonesia etc have very slow internet connectivity).**

# Maintainability and scalability with MVP architecture.



From: <https://github.com/ribot/android-boilerplate>

**Advanced level architecture that enables to upgrade advance libraries and give the ability to introduce any high-level codebase layer.**



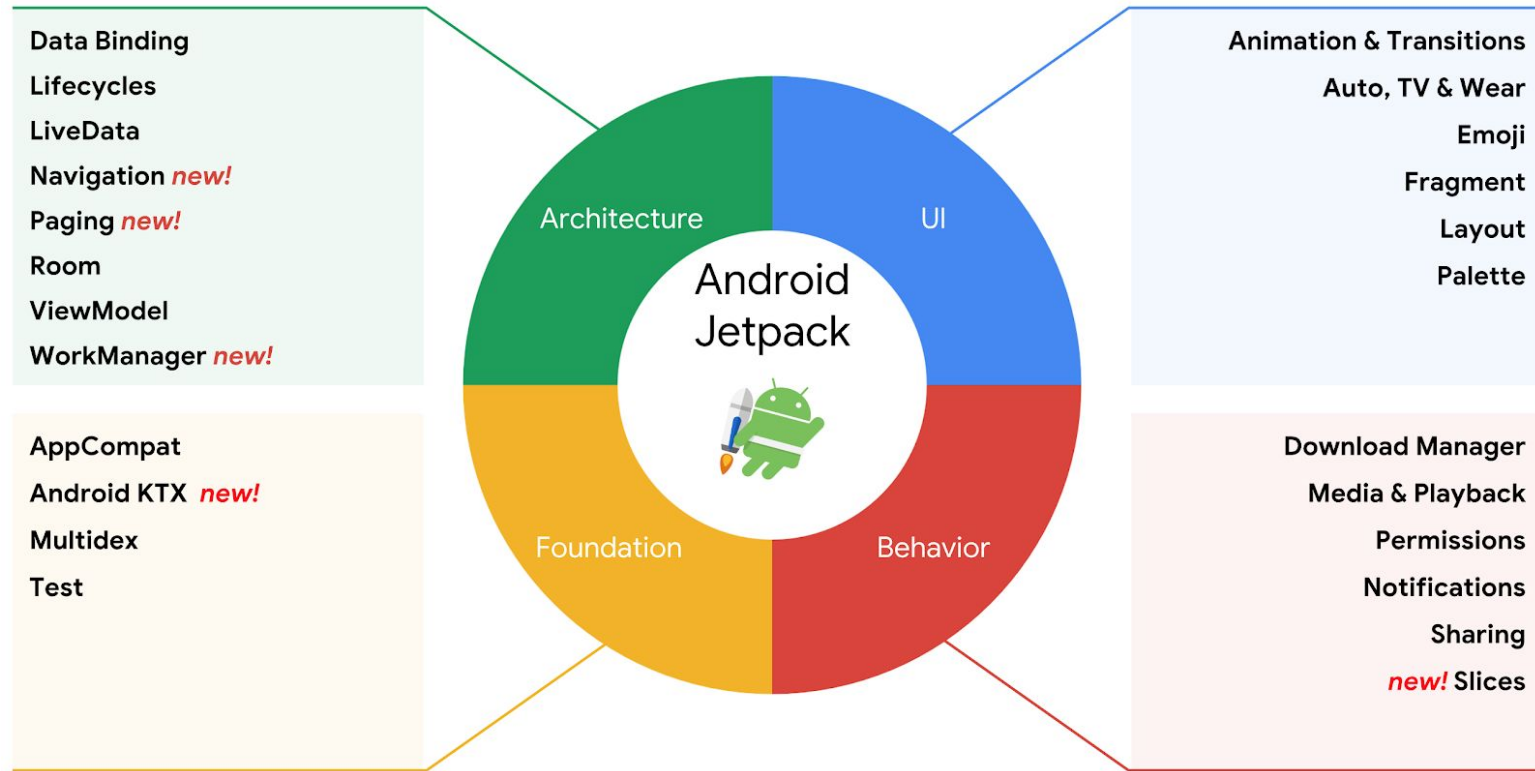
From: <https://github.com/ribot/android-boilerplate>





# No Java, Only Kotlin

# Moving to Jetpack

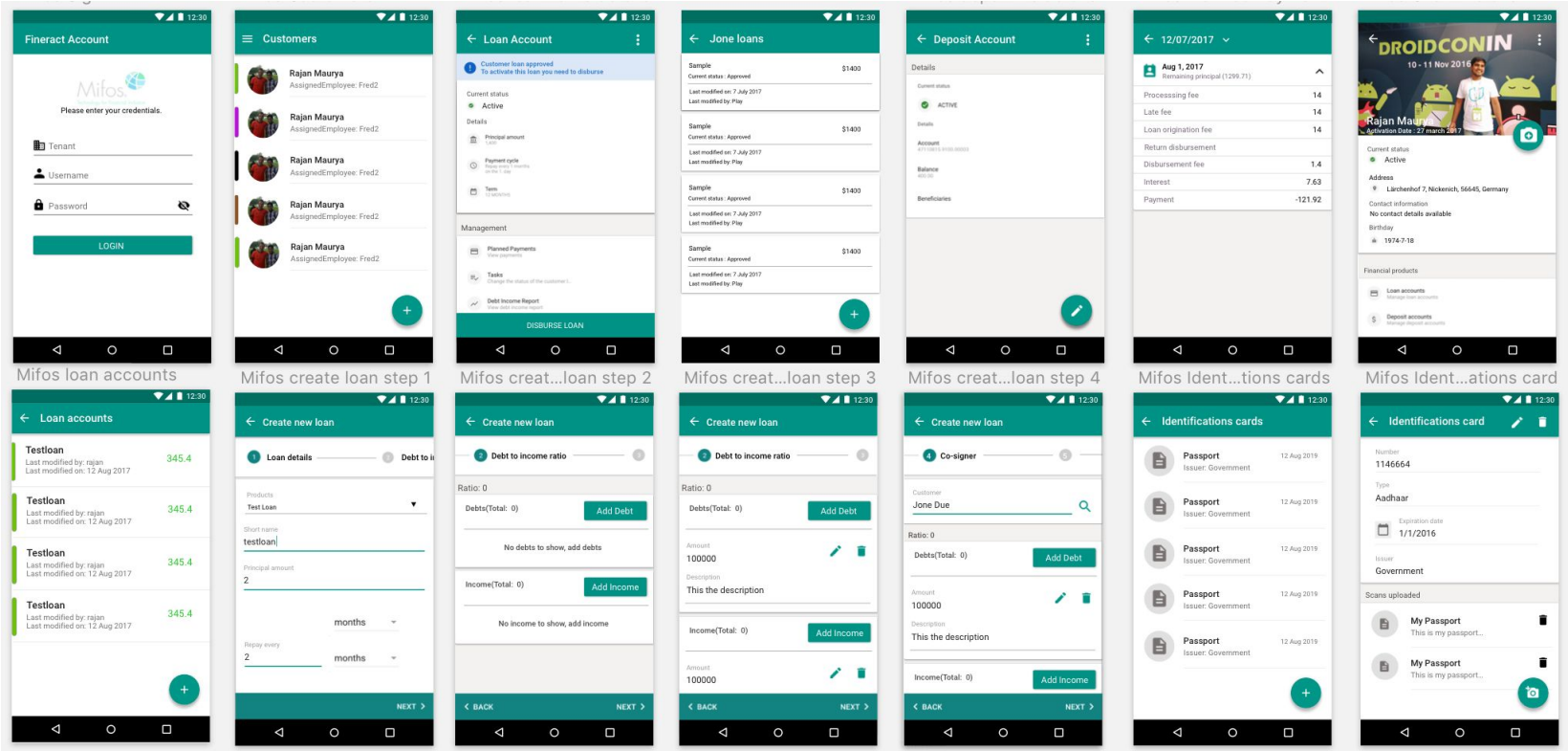




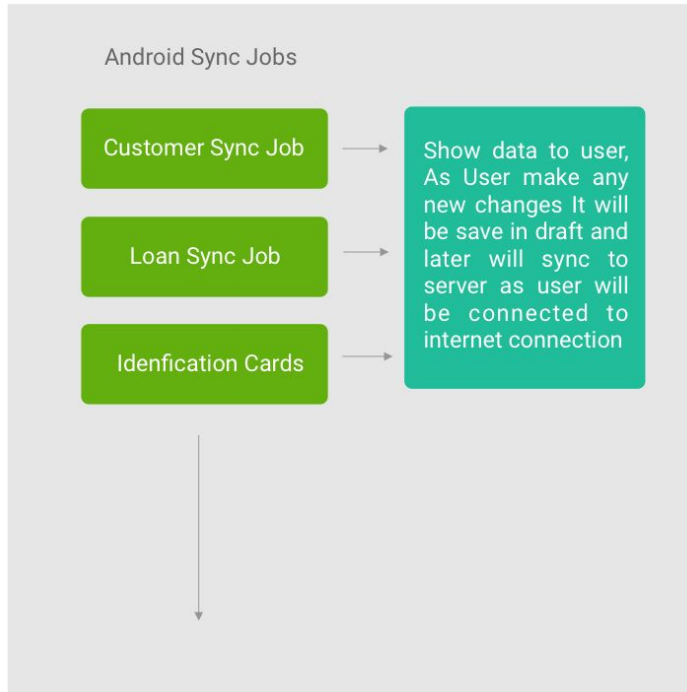
**Advance level security to access the app does not matter user have the internet connection or not. Using the SQLCipher.**

**Database is encrypted using SQLCipher**

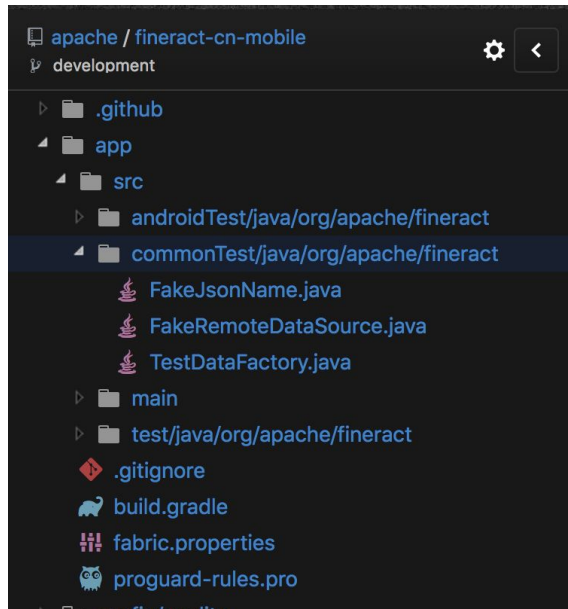
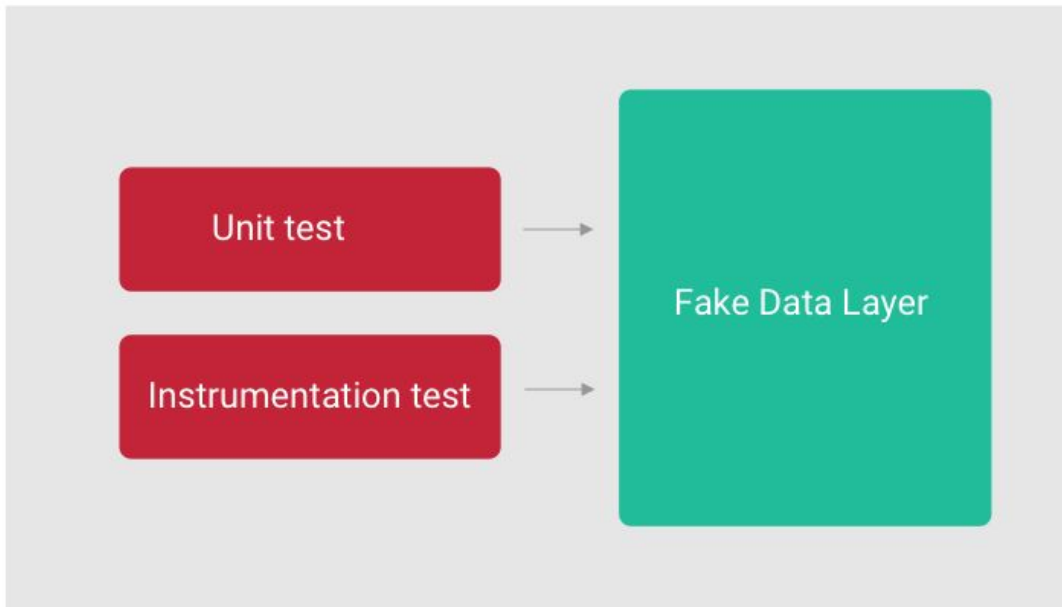
# Simple and easy learnable UI & UX (<https://github.com/openMF/Mifos-Mockups>)



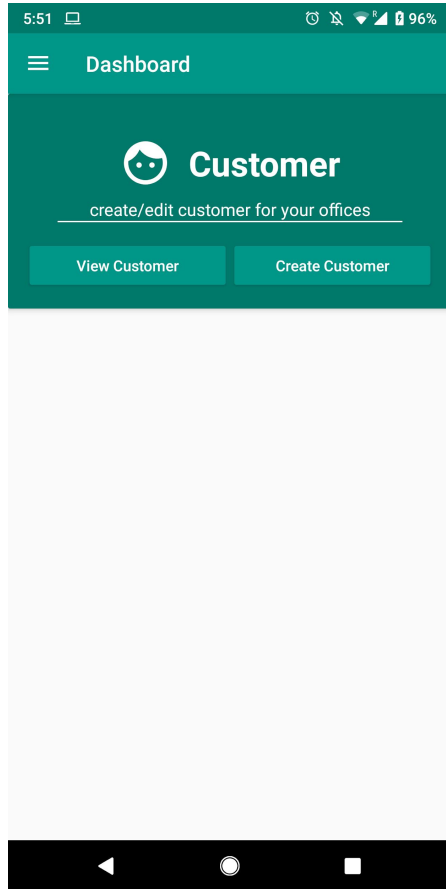
# Offline Support



# Fake Data layer to Run Unit and Instrumentation test



# Demo



# Pick Up codebase and start developing.



## Building the Project

---

1. Install [Java Development Kit \(JDK\)](#)
2. Download, install, and run the latest version of [Android Studio](#).
3. Clone this repository and import it to Android Studio.
4. Once the gradle build is finished, click on the green play button to build and run the app.

## Demo Credentials

You can use these credentials to log in.

```
Tenant: playground  
Username: apache.demo  
Password: ven3t1@n
```





**Thanks, Anything you wanna ask.**