
GitHub Username: abHiShekAppDev

CallerU

Description

Problem:

1. If we are not in touch with someone for few days and when we want to contact them then we found that they change their number. So we are not able to contact them when we are in need.
2. When we need to update our contact number then we have to inform all our contacts about our new number by either calling or texting them one by one that's takes a lot effort as well as time that's why we lastly decide not to update our contact number.

So, I decided to make CallerU where we can update our contact and keep our phonebook updated all time without any effort.

Key Features:

1. Updater - Help you to update your contact number with a tap within 10 sec and inform your new number to all in your phonebook
2. Update Notify -It will notify you about new contact number of your contacts when anyone in your phonebook updated their number. This help you to keep your phonebook updated all time without any extra effort
3. Other –
    - Search - Search contacts by their name to view their profile and know more about them
    - Organize your address book / Phonebook - solve the duplicate contacts issues. No more phone book mess...
    - Unified “recent” feed – Track your recent communication log - call log
    - Smart dialer - App dialing made fast & easy with support of speed dialing

Intended User

Anyone can use CallerU who want to keep their phonebook updated or they want to update their contact number.

Features

List the main features:

- Help in updating mobile number
- Keep mobile phonebook updated
- Notification about the update in phonebook
- Dial pad with speed dialing
- Organize phonebook without any duplication
- Call log
- Free and paid version of app
- Home widget to show all recent notification

User Interface Mocks

Screen 1

- User can see here their call log
- User can dial any number to call on that number

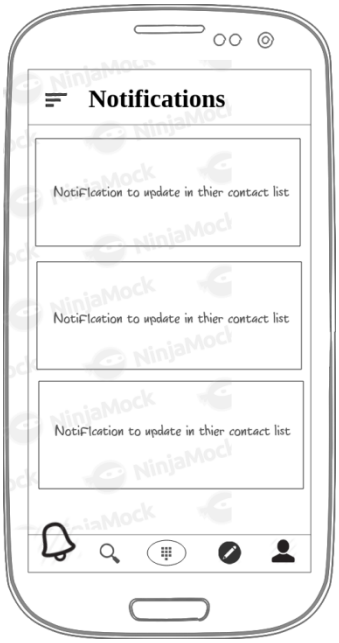
Screen 2

- Navigation Drawer to navigate between given option

Screen 3

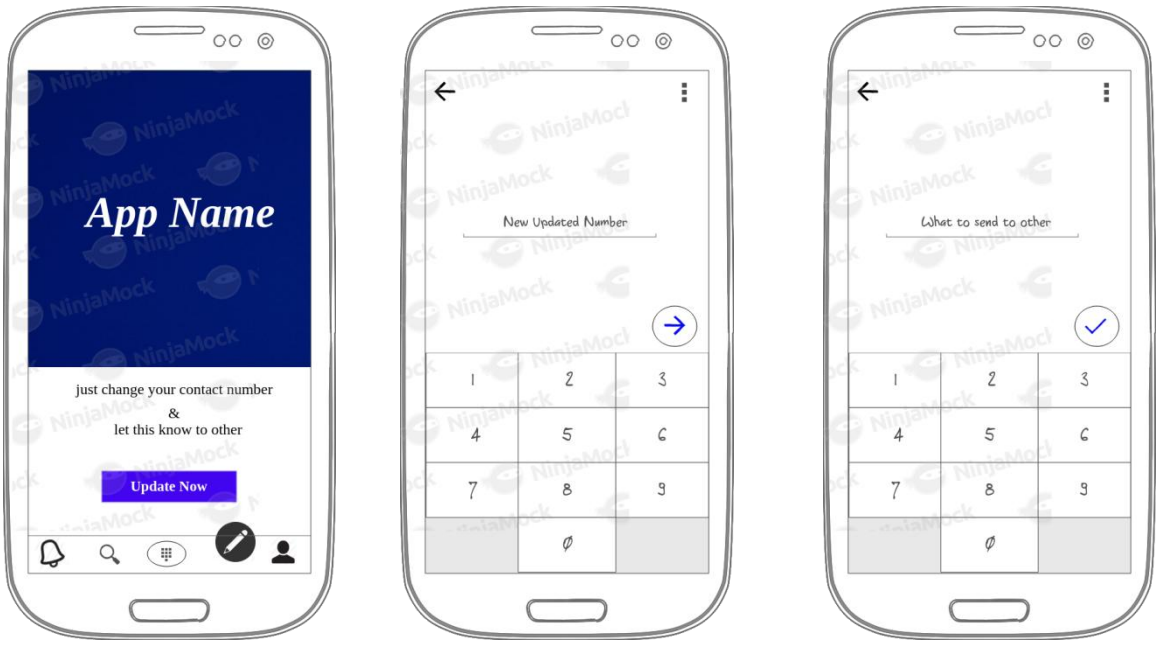
- User can search the contacts in their phonebook by name and can also see their previous searches

Screen 4



- When anyone will update their contact number, user will notify by notification and here all the older notification will stored. So that user can see their notification anytime

Screen 5



- These screen will help user to update their contact number and notify other about same

Screen 6

- User can see their all contacts save in their phonebook without any repetition

Screen 7

- User can see and edit their profile which will updated locally as well as on Firebase

Key Considerations

Basic consideration.

- Programming Language – JAVA
- Gradle version – 4.4
- Min sdk version – 19
- Targeted sdk version – 28
- Compiled sdk version – 28
- Android studio version – 3.1.4

How will your app handle data persistence?

I will use Firebase Realtime Database as online storage and in offline mode I will use SQLite database for saving some information from Firebase Database which need to show without internet connection.

Describe any edge or corner cases in the UX.

- When a notification will arrive then on clicking that user will redirect to Notification section and on pressing back there user shouldn't exit the app. They should send to Homepage.
- When there is no internet connection user can see some offline database without any problem and can also able to use all features which not need a connection.
- The null value and Internet connection should handle properly.

Describe any libraries you'll be using and share your reasoning for including them.

1. Picasso –
    - One of the best image loading library which are constantly solving their issues and updating library on Github.
    - Version: 2.71828
2. Retrofit –
    - Easy to use for Json Parsing because it reduces a lot of boilerplate code. Also it is faster than its competitor library like volley as I think.
    - Version: 2.4.0
3. Room with live data –
    - A new library with many awesome features which reduces a lot of boilerplate code.
    - Version : 2.0.0
4. Butterknife –
    - This library easily bind view and resources
    - Version: 9.0.0
5. OneSignal –
    - It is the library which help in sending notification using firebase in simpleway
    - Version: [3.9.1, 3.99.99]
6. Support Library
    - Version: 28.0.0

Describe how you will implement Google Play Services or other external services.

- Firebase Database - I will use Firebase Database to handle and Store all data that need to be online like Updated number of user and some other data. I will use Firebase Database by making a project on firebase console and following all instruction in doc of Firebase.
- Google ad - I will also use ad in my project and for testing purpose I will use some dummy Api key which shown on the site (will use only in free version)

Next Steps: Required Tasks

Task 1: Project Setup

1. Make an Android Studio project named CallerU
2. Create some resource folders like dimens and free, paid version folder which requires
3. Enable RTL support in Manifest
4. Configure all needed libraries
5. Connect with Firebase

Task 2: Implement UI for Main Activity and Fragment (Screen 1, 3, 4, 6)

1. Create a simple design which look like the given screen design above by taking simple view.

Task 3: Implement simpler features

1. Implement the recent call log features with dialer pad to call on any number
2. Implement the contacts screen by reading all contacts and displaying with help of RecylcerView
3. Implement the search feature on phonebook (use AsyncTask while searching)
4. Test the app on device or emulator

Task 4: Implement Authentication

1. Implement login and signup features by using their mobile number

Task 5: Implement contact update features

1. Make all screen which need while updating contact number(Screen 5)
2. When user will fill all the required detail and click update then the app should find all the number which are in contact list as well as using this app in background with showing a progress bar to user and when the task completed send notification to all found contacts number by using OneSignal and Firebase, hide progress bar.(use IntentService for background task)
3. Test the app that app perfectly sending notification

Task 6: Implement notification section

1. When user will receive notification and open the app, the received notification should save to local database for future reference
2. Implement the notification page by reading all notification from local database and show using RecyclerView
3. Implement widget which also show notification

Task 7: Implement navigation drawer

1. Implement Share, Rate, Exit and Home (Home will take to Dial pad screen)
2. Implement profile by creating a new activity and show all data which are shown in design by fetching from database and if user edit their profile update it in database
3. Test the app on device or emulator

Task 8: Some other work

1. Implement Google Ad in free version and remove it from paid version
2. Write some Espresso Test to test the app

Task 9: UI Implementation

1. Choose the color scheme for app

2. Implement UI by using the choosed color scheme and some good looking fonts

Task 10: Finishing touch

1. Removing all unused resources and imports from code
2. Make sure all Resources are in correct file and folder
3. Testing the app