GitHub Username: Ben Mohammad

ProTalk

Description

ProTalk is a messaging app available for Android. ProTalk uses your phone's Internet connection (4G/3G/2G/EDGE or Wi-Fi, as available) to let you message friends and family or work colleagues.

Includes contacts tab, chat tab and profile tab , functionality to share and a customisable UI in light mode or Dark mode.

Intended User

This application is for anyone with a large social network looking to communicate via sending messages in real-time rather than making costly calls.

Features

List the main features of your app. For example:

- Send messages between devices.
- Save info and share info.
- Change theme light or dark.
- Add and edit profile display name and status

User Interface Mocks

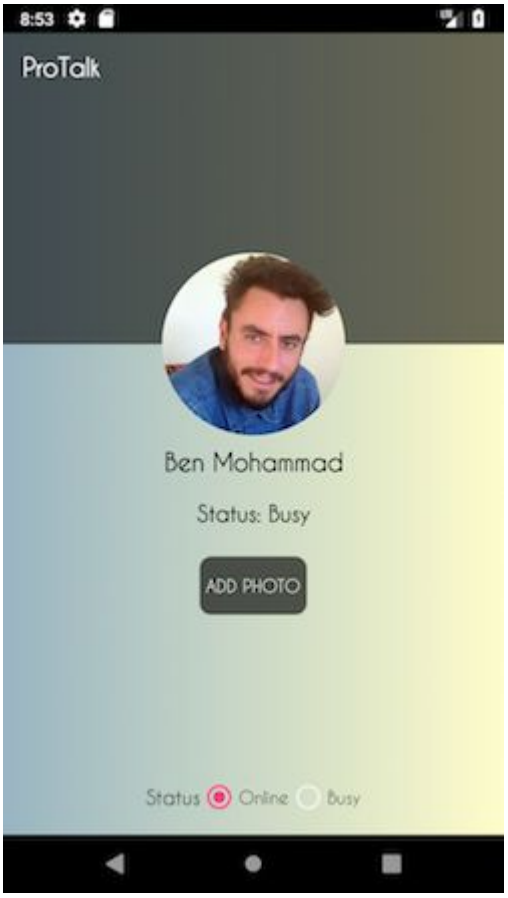
Screen 1

Login initial page that app is launched.
2 options Login if already registered or the option to register which will take you to another activity to create account via intent.

Screen 2

Register activity where data will be stored via Firebase database.

Screen 3



Profile activity where Status can be changed via a Radio button to show whether user is online or busy .

Also option to upload photo from gallery (permissions in manifest required..).

Screen 4

Chat activity where registered users can send messages to each other. This will be implemented via firebase messaging.

Input text interface at bottom of screen with send button icon to send the message.

Left Side linearlayout for reading received messages , right side for messages sent.

Screen 5

Settings activity which displays instructions text and switch to customise the UI to your prefered state (Dark or Light).

Key Considerations

How will your app handle data persistence?

App will handle Messaging Storage with Firebase and contacts list with a provider.

Describe any edge or corner cases in the UX.

User will have to login to use app and will have to be registered in order to Login.

In order to implement a logical orientation around the UI intents after login etc will be followed up by finish() so that flow of app is not disturbed.As sometimes dont want the user to go back.

It would be good to use Fragments via a viewpager to swipe between profile , chat, settings Both will be tried out to experiment to get the desired effect.

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

'com.android.support:design:27.1.1'
'Com.android.support:cardview-v7:27.1.1' cardview will be used for material design affect
'Com.android.support:recyclerview-v7:27.1.1' receyclerview will be used for contacts
'De.hdodenhof:circleimageview:1.2.1' circleImageView library will be used for profile photo.
Google play services for Admob implementation

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

Admob in app advertisement will be used.

Next Steps: Required Tasks

This is the section where you can take the main features of your app (declared above) and break them down into tangible technical tasks that you can complete one at a time until you have a finished app.

Task 1: Project Setup

Create app in Android Studio add dependencies build out the known dimens and Strings and colors in the res folder Add fonts .that will be used before tweaking UI at later stage.

Task 2: Implement UI for Each Activity and Fragment

- Build Login in Xml
- Build Registration in Xml
- Build ViewPager(experiment with fragments...) || build Activity (Profile) in Xml
- Build Chat Activity in Xml
- Build Settings in Xml

Task 3: Your Next Task

- Implement Admob in app advertising

Task 4: Your Next Task

- Build Logic in Java between activities/Fragments tweaking on the fly (improvisations).

Task 5: Your Next Task

- Test App