InTouch!

GitHub Username: tal32123

InTouch!

Description

Have you ever found it hard to stay in contact with the people who really matter in your life? With InTouch you can! InTouch gives a notification with actions to call or send a quick pre-made text to the important people in your life at a pre-selected time interval. This works well for remembering to stay in touch with parents, siblings, grandparents, significant other, and more!

Intended User

This app can be used by anyone. However, the app will most likely be used by people aged 18 above. This is the age that many people end high school and move out in order to go to work or college thus resulting in getting out of touch with family and friends.
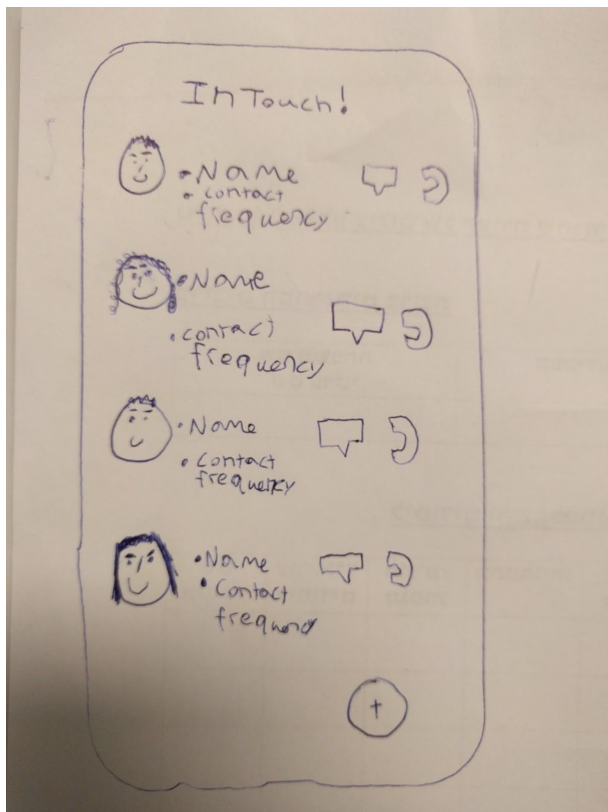
Features

- Saves contact and contact frequency information
- Sends notifications with call and text actions
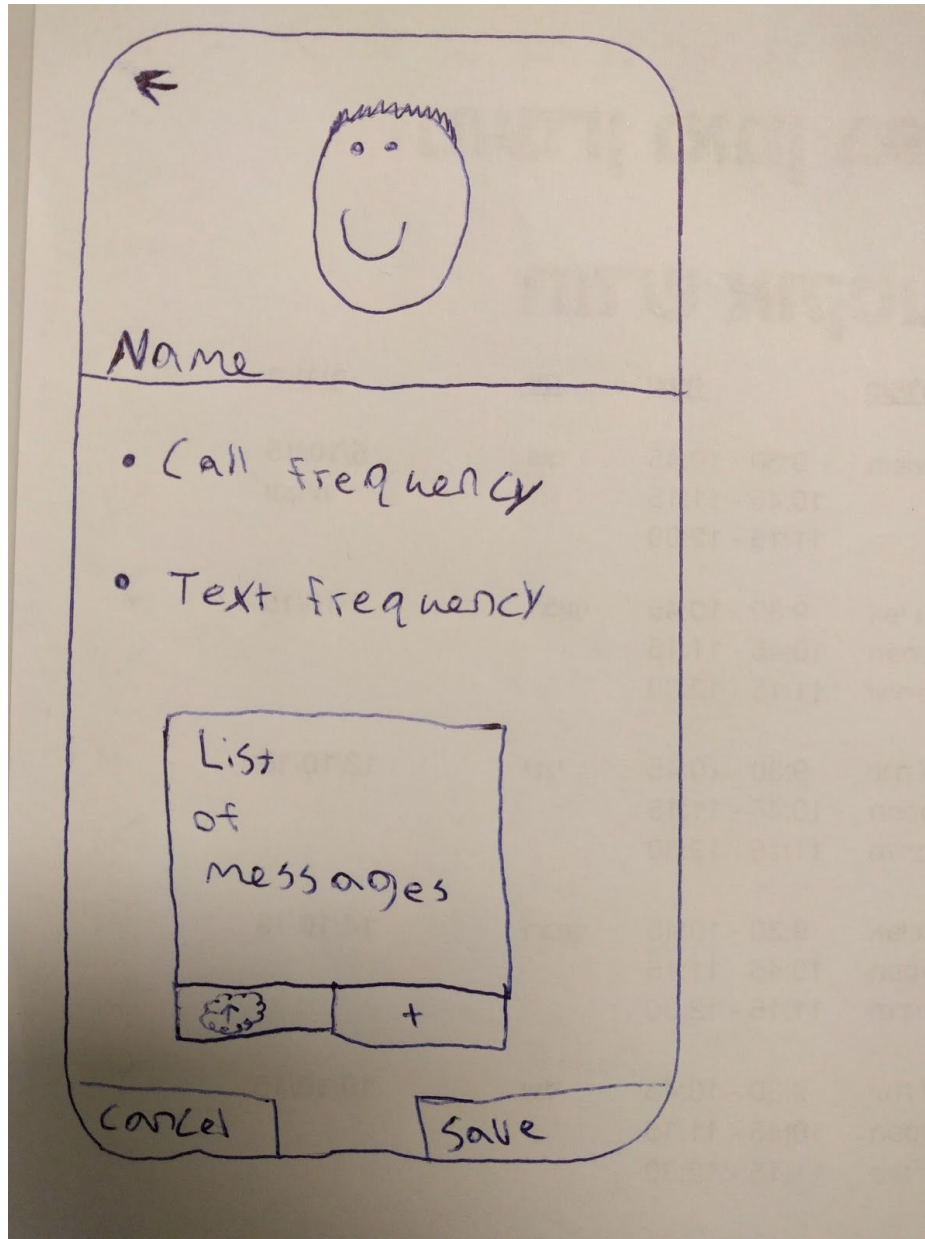- Can make custom messages

User Interface Mocks

Screen 1

This screen shows the list of everyone who is in the list with a button to send a quick call or text message. In the bottom there is a button to add someone to the list. The free version of this app (not pictured) will include advertisements in the bottom.
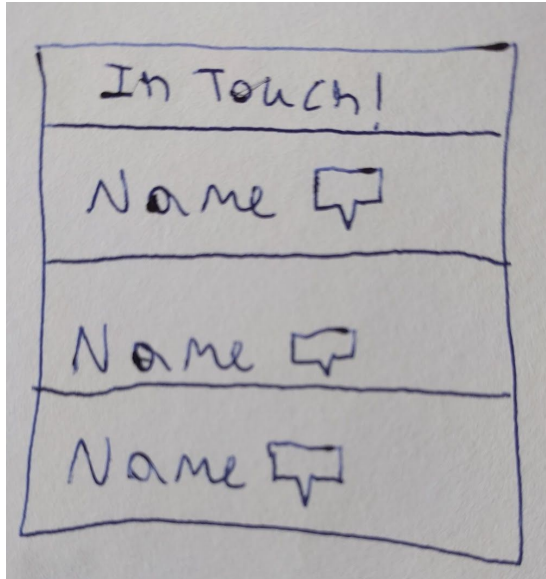

Screen 2

This screen shows an up button to go back to home screen. It also shows the name and photo of the contact if there is one. The app also shows options to change the Call and Text frequency as well as edit the list of messages that can be sent. Messages can also be uploaded to Google Drive for preservation if user sees fit.

Widget

This is the widget which allows quick access to contacts and text/call functionality without having to open up the app.

Key Considerations

How will InTouch! will handle data persistence:

This app will use the contacts content provider in order to easily get previously made contacts.

In addition the app will have its own database with content provider in order to save the app data for contacts, contact frequency, and messages for each contact.

Describe any corner cases in the UX.

The UX is very simple the first screen is the contacts list. The second screen is the contact details. To go back the user clicks the up or back button. In addition there will be a cancel and save button that will bring user to main screen.

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

- Support Design Library for the FAB
- Admob Library for ads
- Google Drive library for cloud saving messages feature

Describe how you will implement Google Play Services.
- I will use the Ads in the free version of the app.
- I will use Google Drive to backup the user created messages in Google Drive.

Next Steps: Required Tasks

Task 1: Project Setup
- Configure libraries

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:
- Build UI for MainActivity
- Build UI for contact info activity

Task 3: Set up Contact Detail Screen

- Set up contact detail screen
  - Add contact image
  - Add up button
  - Add contact name
  - Set up call frequency field
  - Set up text frequency field
  - Set up list of messages field
  - Set up Google Drive backup field
  - Set up database
  - Set up save and cancel buttons

Task 4: Set up Main Screen

- Set up FAB on main screen
  - Fab allows user to use a previously made contact or add new contact to use
- Set up list of contacts on main screen
  - List includes:
    - Swipe to delete
    - Quick call
    - Quick text

Task 5: Set up notifications
- Set up notifications for call and texts

Task 6: Create Free and Paid versions

- Limit free version to 3 contacts
- Add ads to limited version's main screen

Task 7: Set up Widget
- Set up buttons for quick call and texts functions

Task 8: Finalize project

- Generate signed apk
- Upload to Google Play