

## How to Use this Template

1. Make a copy [ File → Make a copy... ]
2. Rename this file: **“Capstone\_Stage1”**
3. Replace the text in green

## Submission Instructions

1. After you’ve completed all the sections, download this document as a PDF [ File → Download as PDF ]
  2. Create a new GitHub repo for the capstone. Name it **“Capstone Project”**
  3. Add this document to your repo. Make sure it’s named **“Capstone\_Stage1.pdf”**
- 

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[Screen 3](#)

[Screen 4](#)

[Key Considerations](#)

[How will your app handle data persistence?](#)

[Describe any corner cases in the UX.](#)

[Describe any libraries you’ll be using and share your reasoning for including them.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

[Task 2: Implement UI for Each Activity and Fragment](#)

[Task 3: Create Content Providers](#)

[Task 4: Add all strings, animations and styles](#)

[Task 5: Add ShareActionProvider](#)

[Task 6: Add Text to Speech api](#)

[Task 7: Add AdMob to display ads](#)

[Task 8: Build tablet UI for all activities](#)

**GitHub Username:** sanyuktaghai

# Healing Stories

## Description

Reiki groups in India currently use chat applications to share success stories and experiences. This is very disorganised as the content gets lost and difficult to search. There is a need for a better way of storing and accessing these amazing stories.

Healing Stories app will be a platform for users to save and access motivational stories, articles or experiences. They will be able to share this content within their social circles and also save the ones that they really connect to.

## Intended User

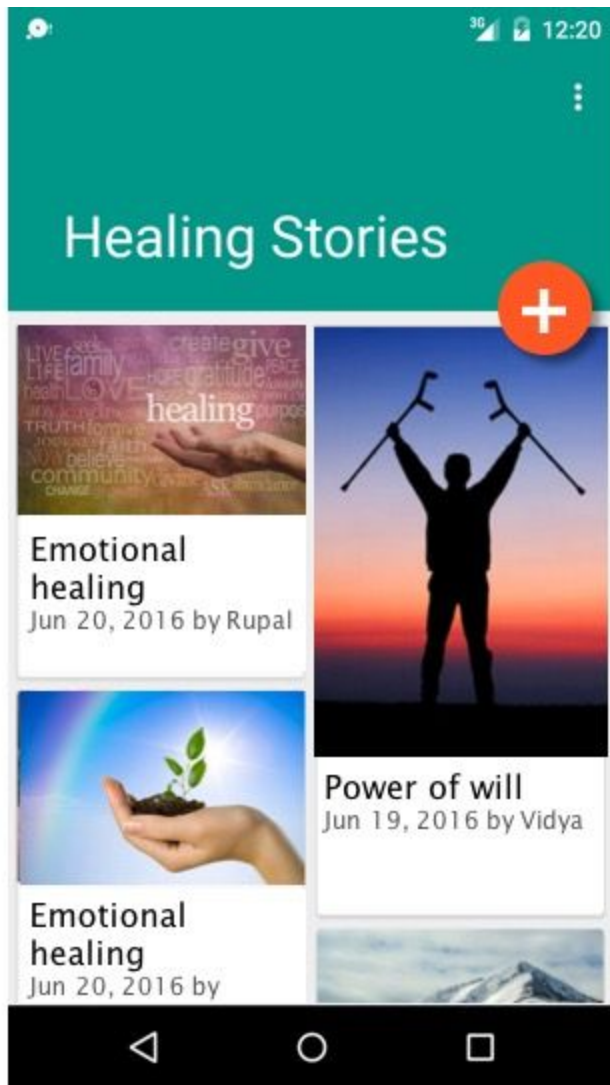
This is an app for followers of reiki who believe in sharing all the positive success stories they come across.

## Features

- Read and Write some inspiring short stories
- Save the ones you really like
- Comment on your friends stories
- Share the positivity within your social circles
- Easy to use simple interface

## User Interface Mocks

### Screen 1 - Main Activity



This screen will be the Main Activity. It will show all recent stories updated by people. Clicking on any of the card will take you to the detail activity.

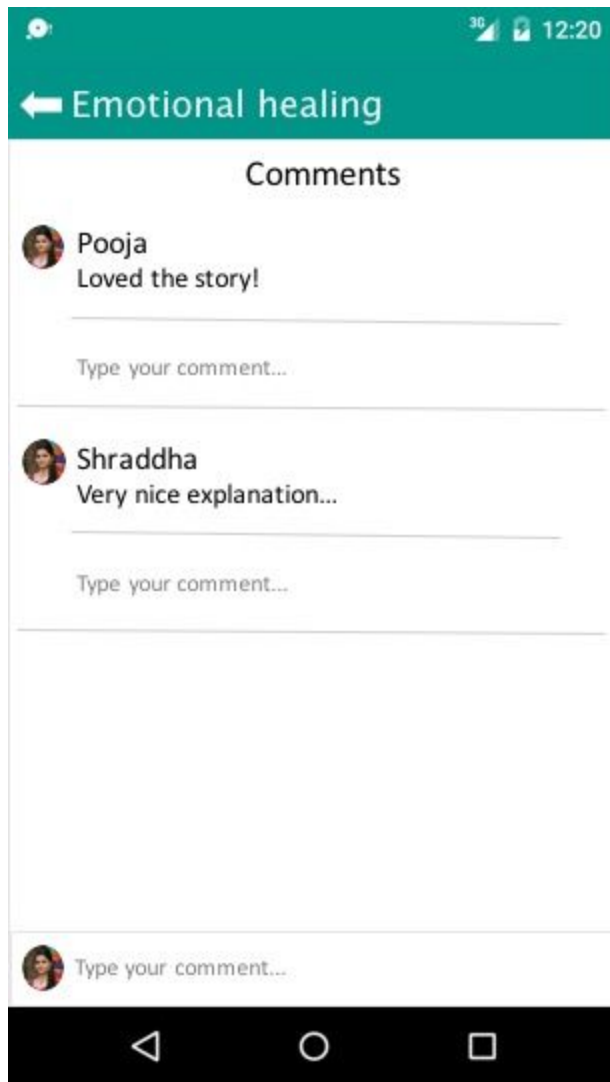
## Screen 2 - Detail Activity



This will be the detail activity displaying the entire story. You can save the stories you like in your personal collection by clicking on the FAB button. Options menu will have

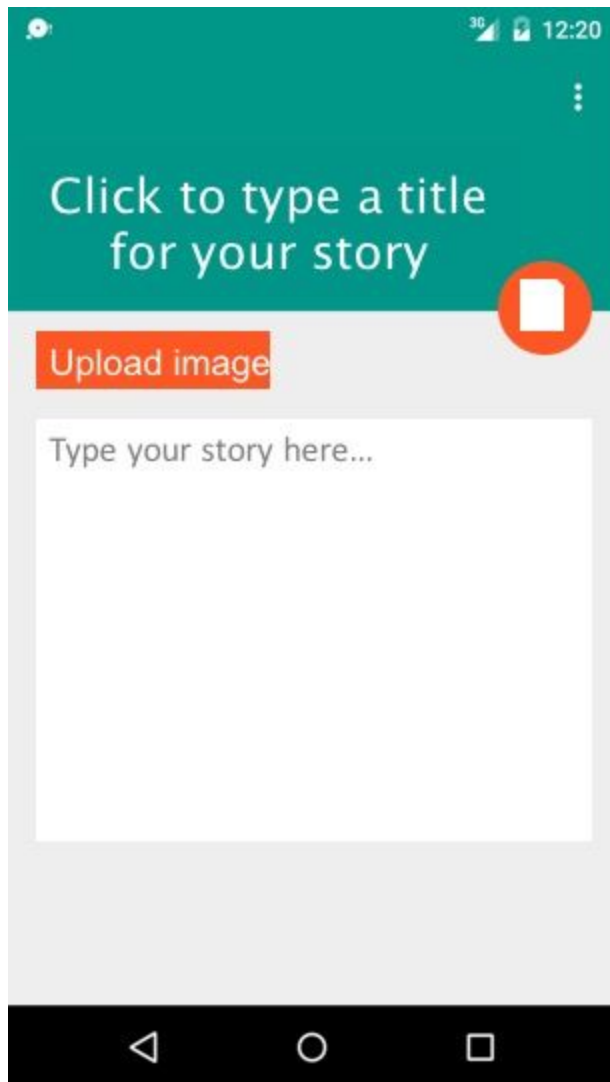
- the option to share the story on social accounts
- The facility to view the interesting comments and give your own comments

### Screen 3 - Detail Comment Activity



This will be the detail comment activity. It will show all the comments received. You can also give your own comments on the story and individual comments received.

## Screen 4 - Write Activity



An activity to write and share your own stories. The stories posted through this activity will be displayed in the Main Activity. People can then comment and share your stories.

## Key Considerations

How will your app handle data persistence?

Entire data will be hosted on Firebase. The favourite stories will be saved locally using SQLite and will be accessed using Content Providers.

**Describe any corner cases in the UX.**

1. Clicking on back in the Detail Comment Activity will take you to the Detail Activity.
2. Clicking on back in the Detail Activity will then take you to the Main Activity.

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

- Picasso to handle the loading and caching of images.
- Firebase to handle authentication, hosting and storage.

## Next Steps: Required Tasks

### Task 1: Project Setup

- Configure libraries
- Check all the android versions
- Setup the database on Firebase
- Create the necessary packages

### Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
- Build UI for DetailActivity
- Build UI for DetailCommentActivity
- Build UI for WriteActivity

### Task 3: Create Content Providers

- Create and test the content providers
- Hook them up with the loaders

**Task 4: Add all strings, animations, styles in respective resources files**

**Task 5: Add ShareActionProvider**

**Task 6: Add Text to Speech to read out the story (Optional)**

**Task 7: Add AdMob to display ads**

**Task 8: Build tablet UI for all activities**

---

### **Submission Instructions**

1. After you've completed all the sections, download this document as a PDF [ File → Download as PDF ]
2. Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
3. Add this document to your repo. Make sure it's named "**Capstone\_Stage1.pdf**"