

[Description](#)

[Intended User](#)

[Features](#)

[User Interface Mocks](#)

[Screen 1](#)

[Screen 2](#)

[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.](#)

[Describe how you will implement Google Play Services.](#)

[Next Steps: Required Tasks](#)

[Task 1: Project Setup](#)

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

[Task 3: Your Next Task](#)

[Task 4: Your Next Task](#)

[Task 5: Your Next Task](#)

**GitHub Username:** brunodea

## Never Too Late

### Description

Sometimes during our daily lives, we are just “not feeling it” and we need a few words to motivate ourselves in order to keep on going, right? *Never Too Late* is the application that tries to solve this problem. It can show you meaningful insights whenever you need them: when waking up, when feeling down or even when you simply want some help to stop procrastinating.

*Never Too Late* will help *you* make your life better.

### Intended User

This app is for people who want to improve their lives. That can be *anyone*.

### Features

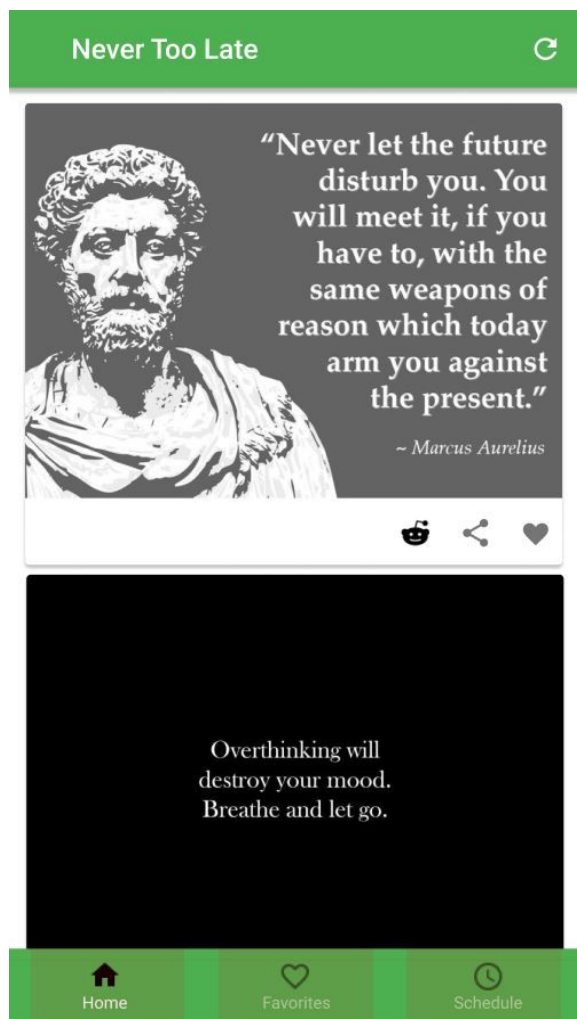
- Get motivational images from the subreddit “GetMotivated”;
- Download and save favorite images;

- Show notifications with motivational images at specific times, set by the user;
- Load motivational images by demand;
- Provide ways to check the reddit users discussion about a given image;

## User Interface Mocks

These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Google Drawings, [www.ninjamock.com](http://www.ninjamock.com), Paper by 53, Photoshop or Balsamiq.

### Screen 1



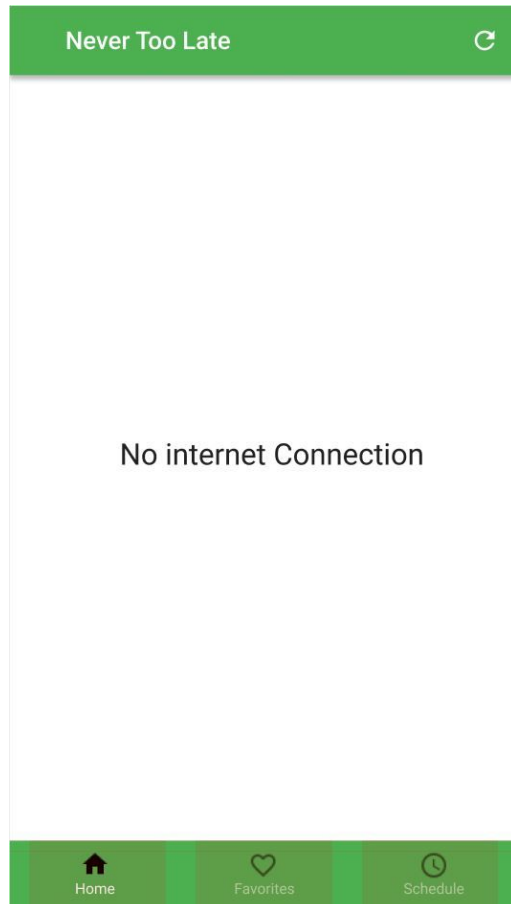
This should be the main screen for the application. It will display a list of insightful images grabbed from the “Hot” category on reddit, it will load the 10 hottest posts from [/r/GetMotivated](https://www.reddit.com/r/GetMotivated) with the tag [image]. The requests to Reddit will happen through an AsyncTask when the user opens the app or clicks the Refresh button.

In this screen, the user can:

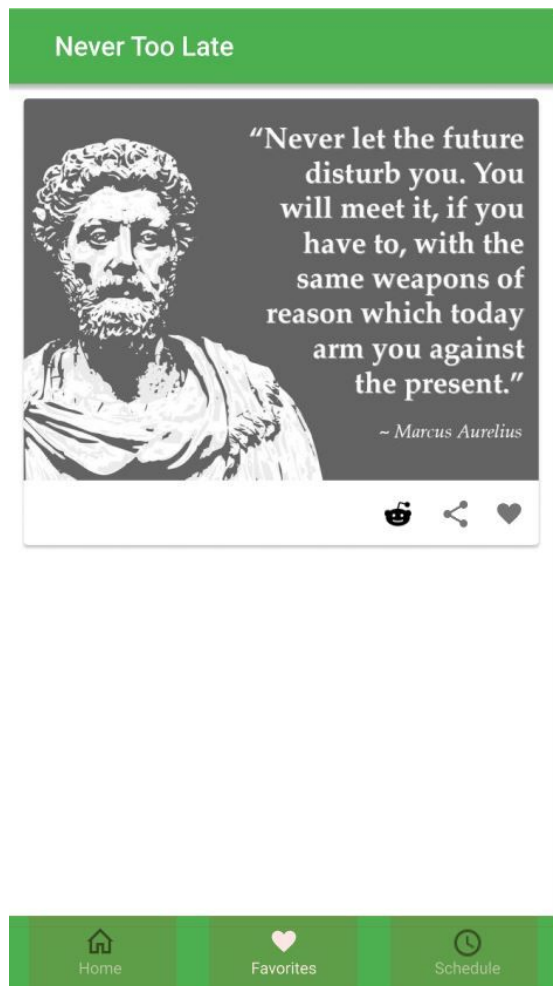
- “Refresh” so the images are re-loaded from the posts;
- Check the reddit discussion by clicking on the reddit icon (it will make the user go to another app);

- Share the image;
- Favorite the image;
- Go to either the list of favorites or schedules.

If there is no internet connection, the “home” screen will simply display a message informing the user:



## Screen 2



The second screen is the list of saved images by the user. They should be loaded from the database using a CursorLoader.

Here, the user can go to the reddit discussion, share and unfavorite the image.

If the user doesn't have any favorites yet, a message is displayed:

Never Too Late

You currently don't have any  
favorite insights



Home

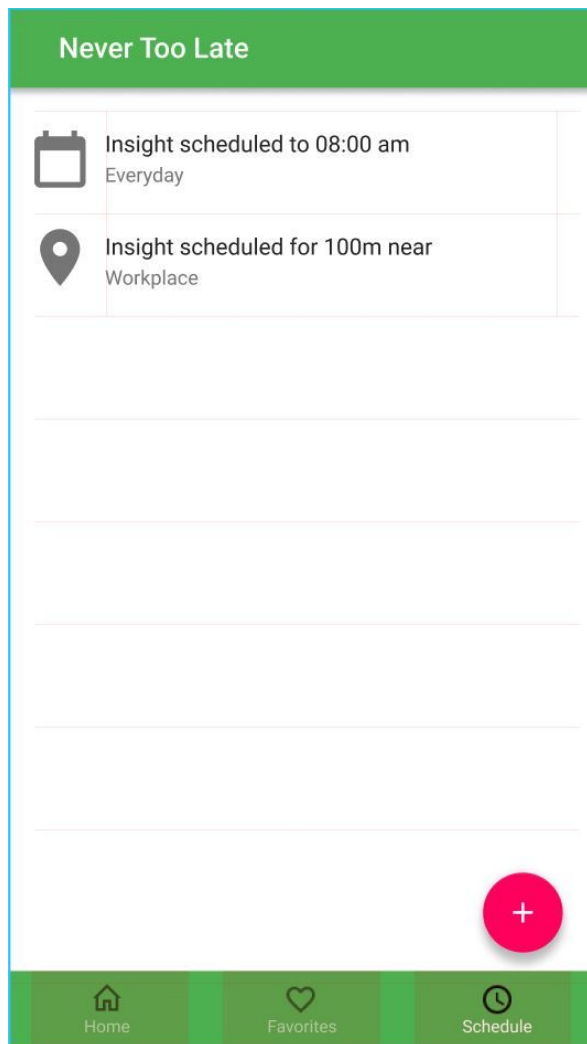


Favorites



Schedule

## Screen 2

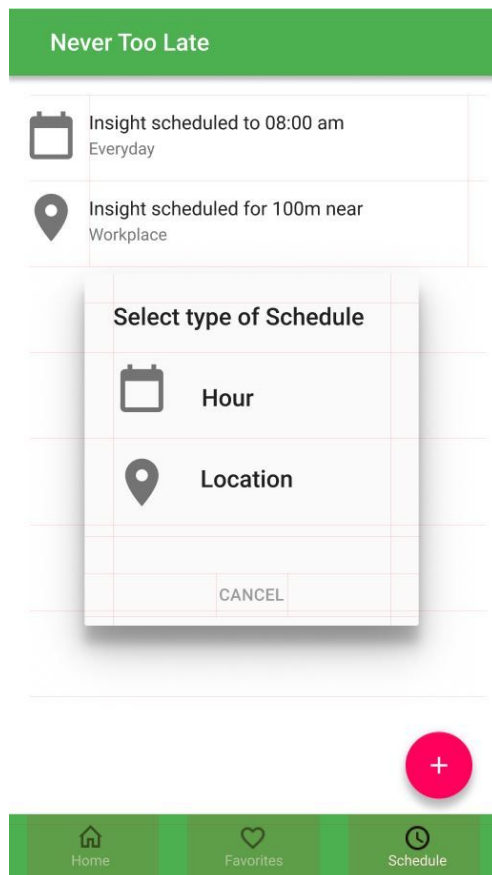


In the schedule screen, the user will see a list of “scheduled insights”, that is, the user will receive notifications with some insightful image according to some time/day or location, such as 100m radius of work.

The images sent to the user via notification is random. It should be one of the top 10 in the “Hot” category in /r/GetMotivated.

The user can remove schedules by swiping them. When doing so, a popup asking for confirmation should be displayed.

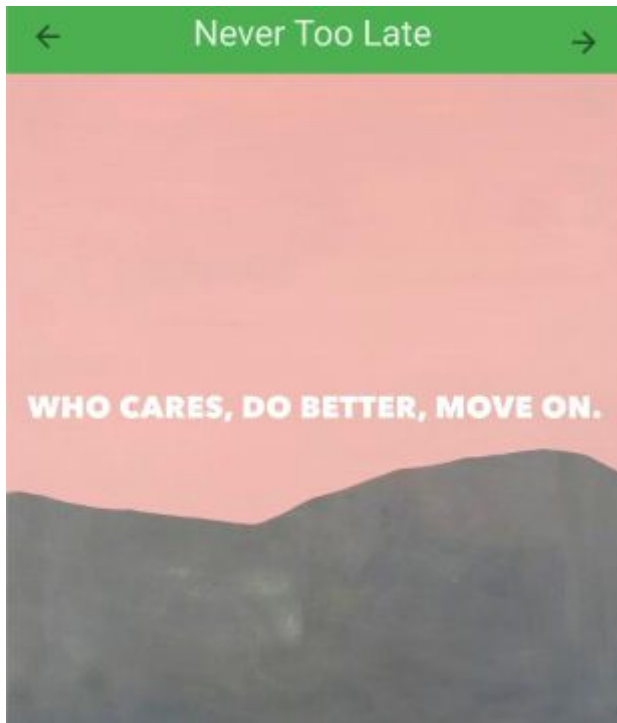
By clicking the FAB, the user will be able to add a new schedule:



When selection “Hour” a default dialog for choosing a time will be displayed and when selection Location, the user will choose a location and radius for when he wants to receive the motivational image.

## Widget

The widget will display the favorite images. The user will be able to see the “next” and “previous” favorite and go to the app by clicking the widget top bar.



## Key Considerations

How will your app handle data persistence?

The data that needs to be persisted for this application are the favorite images and the user schedules. A Content Provider can be used for that.

Describe any edge or corner cases in the UX.

If the user tries to refresh the home screen and there is no internet connection, the display should keep the images already loaded and a popup should be displayed for the user informing him it wasn't possible to update the images.

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

I will use Picasso for downloading and caching images, since this is an image-based application, it is important to have such a library to handle images.

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

I will use google maps for giving the user the ability to schedule notifications based on location.



Another service will be the notifications themselves and perhaps I will add Ads.

## 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

- Configure libraries for google maps and Picasso;
- Create a single activity with an action bar and a bottom navigation bar;
- Create the classes for dealing with the data from reddit;
- Create 3 fragments, one for each of home, favorites and schedule.

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

- Make the UI for the fragments and link them to the source from reddit;
- Implement the database for storing the necessary information (favorites/schedules).

### Task 3: Scheduling notifications

- Implement the flow for scheduling notifications with both the “by hour” and “by location” types.

### Task 4: Widget

- Implement a widget that display the favorite images;
- Handle the case when there are no favorites yet.

### Task 5: Tablet

- Make a different layout for tablets: instead of a list of cards in the main and favorite screens, display a grid of cards.

### Task 6: Ads

- Create an ads-enabled build variant in gradle.

Add as many tasks as you need to complete your app.

---

### Submission Instructions

- After you've completed all the sections, download this document as a PDF [ File → Download as PDF ]
  - Make sure the PDF is named "**Capstone\_Stage1.pdf**"
- Submit the PDF as a zip or in a GitHub project repo using the project submission portal

If using GitHub:

- Create a new GitHub repo for the capstone. Name it "**Capstone Project**"
- Add this document to your repo. Make sure it's named "**Capstone\_Stage1.pdf**"