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
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: mingrutar

Naturalist Hike

Description

So you love hike and you love nature. But have you ever walked into a field of beautiful wildflowers and wished you can call their names? Have you ever left your binocular home for a bird watch trip? Have you missed a trip because you could not find the meeting place? Well, if you have an Android smartphone, with this app, preparation of a naturalist hike will be a breeze.

This app will get you to hike on time, bring all items you need for the hike and prepare plant list for your pre-trip study. Once you are on the trail, the app let you mark the observed plants.

Intended User

For the initial version, the intended users are Washington Mountaineer's Introduction to Nature World students. However the app could be used by wider users when the server site is better facilitated. I have eyes on Firebase and finished first two (out of 4) Udacity Firebase lectures.

Features

The main features of the app:

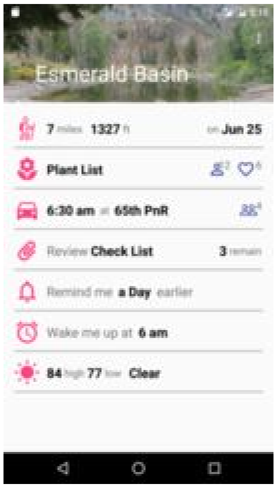
1. Set up a wake up alarm for the hiking day.
2. Compiling a multi-source checklists for items you needed for the trip. In addition to the club and leader's checklists, you can build your own checklist. The app can remember your choice if want.
3. Send you a trip reminder prior hike. If non-optional items found unchecked in checklists, you will be able to check them.
4. Provides weather information of hiking day at hiking location.
5. Showing you the carpool meeting location and time for the trip.
6. Provides a list of plant at the hiking location. The plants of leader's choice will be marked and you can set the plants you like to observe during the hike.

User Interface Mocks

The next trip will shown when starts in normal situation. The one time setup is described in next section.

The main screen of the app. Click 1st item (trip info) => the full trip description at Mountaineer.org.

Main Screen



Esmerald Basin

7 miles 1327 ft on Jun 25

Plant List

6:30 am at 65th PnR

Review Check List 3 remain

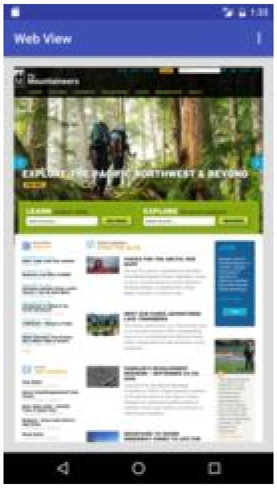
Remind me a Day earlier

Wake me up at 6 am

84 high 77 low Clear

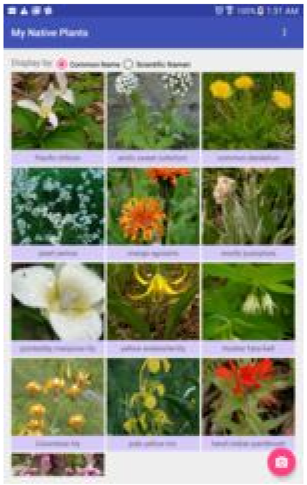
click item1

Trip Detail



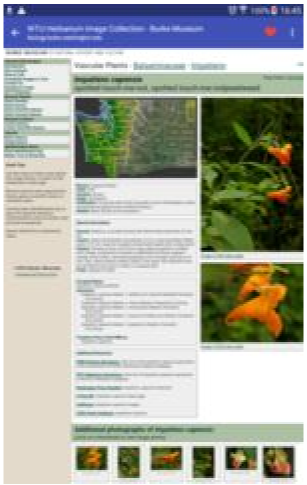
From Main click 2nd item => Plant Detail; Click the ‘Heart’ icon at top => added as Favorite. The plan list screenshot was from my project of Android Development Jam earlier this year. It is just a placeholder.

Plant List



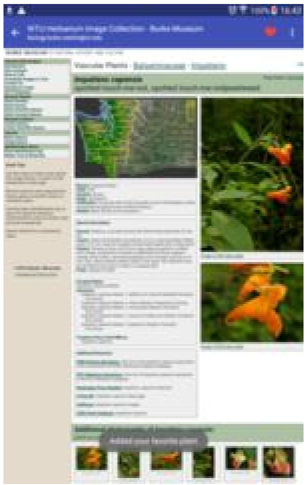
click an item

Plant Detail



click ‘Heart’

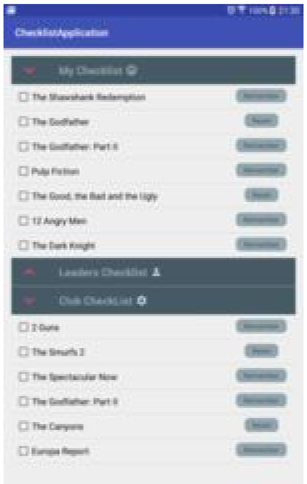
Adding to Favorite



Click the 3rd item (meeting info), an activity show Google map with capability of change meeting place and time. No screenshot yet.

Click the 4th item shows checklist

Checklists



CheckListApplication

My Checklist

□ The Shawshank Redemption

□ The Godfather

□ The Godfather: Part II

□ Pulp Fiction

□ The Good, the Bad and the Ugly

□ 12 Angry Men

□ The Dark Knight

Leaders Checklist

Club CheckList

□ 2 Guns

□ The Smurfs 2

□ The Spectacular Now

□ The Godfather: Part II

□ The Canyons

□ Europa Report

Click the 5th item (Reminder), show a dialog (no screenshot yet). The user can change reminding time.

Click the 6th item (Wakeup Alarm), show a dialog (no screenshot yet). The user can change the alarm time.

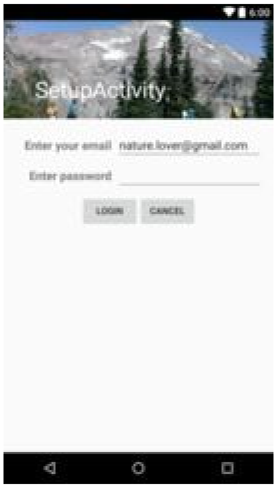
Click the 7th item (Weather Info), show weather info (no screenshot yet).

Additional Screenshots for Setting and Sync

The rules:

- The user accounts are stored at Firebase Server (need to learn more about it).
- The Email is the username.
- A sign-up user has password.
- A user can only get the trips he/she participates.
- A user must sign-in to pull his/her trip list (or Firebase push, how authentication works in this scenario? As I mentioned, I need to learn more about Firebase)

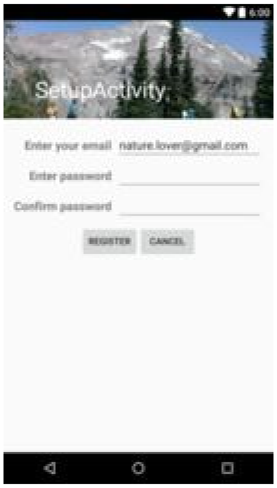
After the app is installed, It prompts for email. If the user already signed up, it prompt for password for sign in:



Enter your email nature.lover@gmail.com

Enter password

If the user is not signed up, show sign up screen

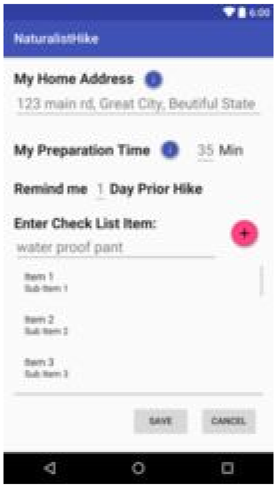


Enter your email nature.lover@gmail.com

Enter password

Confirm password

When the user is signed in or cancel, it ask for personal info (one time only), such home address, preferred morning preparation duration and checklist. The user can change such items later via Settings menu.



My Home Address
123 main rd, Great City, Beutiful State

My Preparation Time 35 Min

Remind me 1 Day Prior Hike

Enter Check List Item:
water proof pant

Item 1
Sub Item 1

Item 2
Sub Item 2

Item 3
Sub Item 3

Key Considerations

How will your app handle data persistence?

The user defined simple data, a name-value pair, such as home address will be stored in SharedPreference. The plant and trip data are stored in SQLite3. I build a content provider for plants and a content provider for trip respectively.

Describe any corner cases in the UX.

The plant detail screen will back to plant list screen. The rest, I think, will back to Main screen.

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

The app will use Glide to handle the loading and caching of images, Chrome Tab to show plant detail and jSoup to build JSON plant list from a web site. This likely is a temporary solution and the app will credit the website.

Describe how you will implement Google Play Services.

The app will use Firebase for trip and plant lists, and Google Map for meeting place. Also I consider Firebase for user accounts.

Next Steps: Required Tasks

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

Task 1: Project Setup

Build with gradle. Needed keys:

- An api key for weather http://api.openweathermap.org/data/2.5/forecast/
- Google-services.json for Firebase
- Does Google Map require separated key?

Task 2: Design UI for Each Activity and Fragment

Screens:

- MainActivity
  - Sign-in and Sign up fragment
  - Personal info Setup fragment
  - Main fragment
- Plant list
- Meeting Info
- Checklist activity
- Weather fragment
- Several dialogs

Task 3: Prepare Data Sources

- Data Extractor that extracts plant info from a web site and add to JSON.
- Create Trip JSON file
- Setup Firebase project

Task 4: Design Database

Design SQLite schema

- Consider various scenario to understand the tables and relations
- Exam various scenario to check the feasibility
- Finalize the scheme with expectation of fine tuning.

Task 5: Build Trip Content Provider

- Exam various scenario to understand the usage
- Design Contract and Content Uris.
- Define MatchCode for content uris.
- Define SQL Ops for MatchCodes.
- Write the content provider
- Write helper functions for data retrieving and manipulation.
- Write unit tests

Task 6 Build Trip Content Provider

- Exam various scenario to understand the usage
- Design Contract and Content Uris.
- Define MatchCode for content uris.

- Define SQL Ops for MatchCodes.
- Write the content provider
- Write helper functions for data retrieving and manipulation.
- Write unit tests

Task 7 Implement Plant List and Detail View

- Implement Plant List Activity
- Plant detail view uses Chrome Tab. The Chrome Tab is a RemoteView similar to Wedget. Need research on data exchange, then implement accordingly

Task 8 Implement Sign-up/Sign-in using Firebase Account Management

- Learn how to use Firebase for user account management (Udacity Firebase course)
- Implement it within the app

Task 9 Implement Meeting Info using Google Map

- Research how to place Google Map in a fragment.
- Research bilateral communication between Activity and Google Map
- Implement the Meeting Info activity.

Task 10 Implement Wake Up Alarm

- Research how to communicate with device Alarm app.
- Design the UI accordingly.
- Implement the the Wake Up Alarm activity.

Task 11 Implement Other UI

- Implement Checklist
- Implement weather
- Implement Reminder
- Implement Trip Info

Task 12 Style UI

- Final UI style
- Add animation if fits

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

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**”