

“Forget Me Not”

Final Report

Mobile App Engineering & User Experience Design
Fall 2019 | Janne Lindqvist

Group #1

Daniel Nguyen: dnn21@scarletmail.rutgers.edu

Felix Zheng: fdz6@scarletmail.rutgers.edu

Manish Kewalramani: mkewalra@scarletmail.rutgers.edu

Yuqing (Sunny) Feng: yf184@scarletmail.rutgers.edu

Motivation

Taking care of plants can be difficult, especially for people who are first-time gardeners or veteran gardeners who have a lot of plants. We want to make sure that users have the resources available to properly take care of their plants in a timely manner. By sending notifications on when a plant should be watered and when to move plants indoors due to hazardous weather, users can make sure that their plants will stay healthy and bloom. We want users to enjoy gardening by incentivizing good gardening habits with badges and points. With this application, we aim to create an interactive framework that users can use to manage their own personal gardens.

User Functions

Users will be able to:

1. Insert each of their garden plants into a database and see their plants on the Dashboard
2. Mark and see the time of the last watering on the Dashboard
3. See the history of plant waterings on the Home page
4. Upload and view images of their plants in the Gallery
5. Receive reminders regarding when it is time to care for their plants
6. Receive notifications about incoming harmful weather from the OpenWeather API
7. Enjoy the gardening experience with fun badges and points

Key Components/Features

Notifications (Weather & Plant Care)

The application sends notifications to the user to remind them that it is time to water their plants. The notification includes the names of the plants that need to be watered. The notification service runs in the background and checks every hour to see if any plants need to be watered. Then the user can go into the app and confirm that they successfully watered their plants within the Plant Dashboard. This will automatically give the user points (to be discussed later). It also sends a notification that is prompted by hazardous weather conditions that serve to remind the user that they need to bring their outdoor plants indoors. In the notification service, we connect to the OpenWeather API and get the weather conditions of the user's location. If the temperature is under 50 degrees Fahrenheit, the service sends a notification to the user warning of hazardous weather and asking them to bring outdoor plants inside if possible.

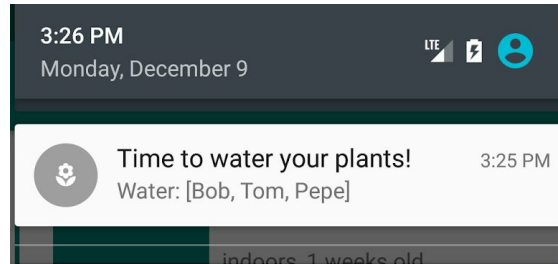


Figure 1: Notification

Plant Dashboard

The application allows users to see their collection of plants all in one place. Users can add their plants through the “Add Plant” feature, where they input the name, type, age, and location (indoors, outdoors) information for the plant. Users can scroll through their plants in a Recycler View and can see the corresponding information (name, type, location, age, and the date they were last watered). Long clicking the plant will indicate that the user has watered the plant and will update the last watered time accordingly.

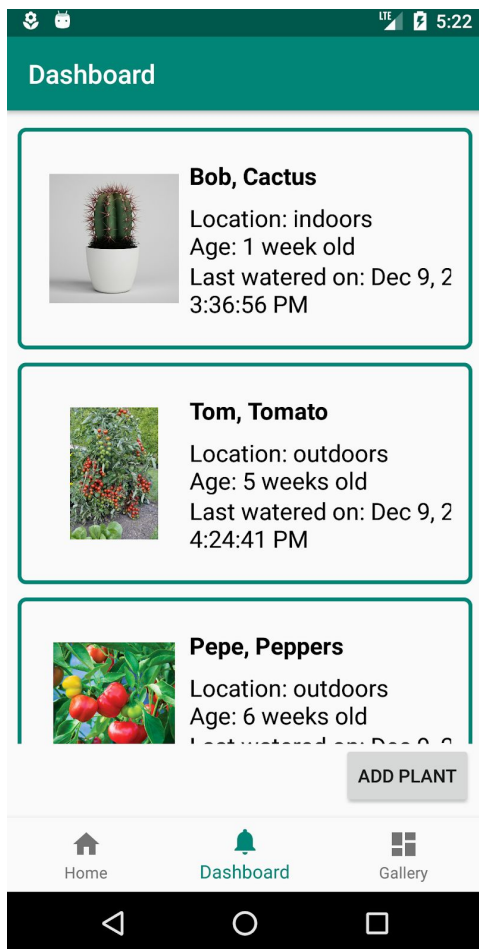


Figure 2: Plant Dashboard

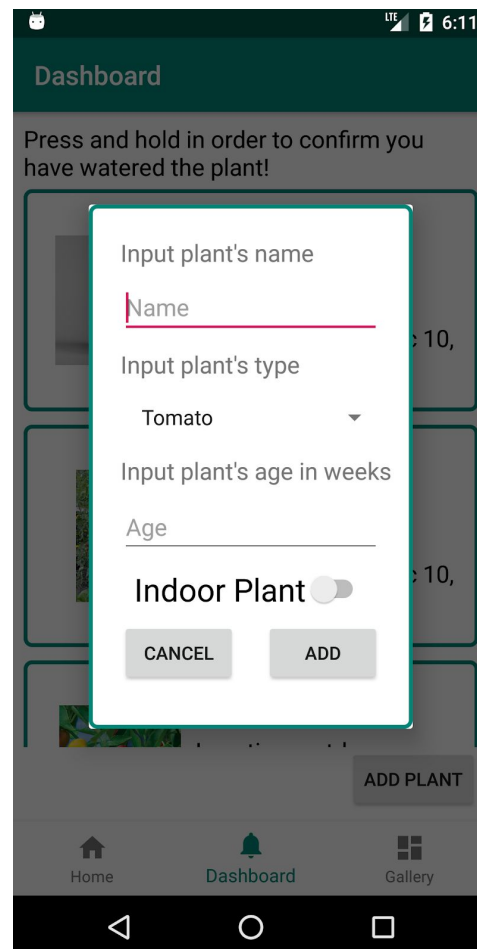


Figure 3: Add Plant

Plant Image Gallery

The application provides the user with a gallery function that allows the user to select one of their plants by name and display all of their images of that plant. The user can also use the gallery to link photos on their phone to their plants for easy access later. Enabling this gallery requires permission to read images from the user's device. We stored the images in SQLite and connected them to the plants through the `plant_id`.

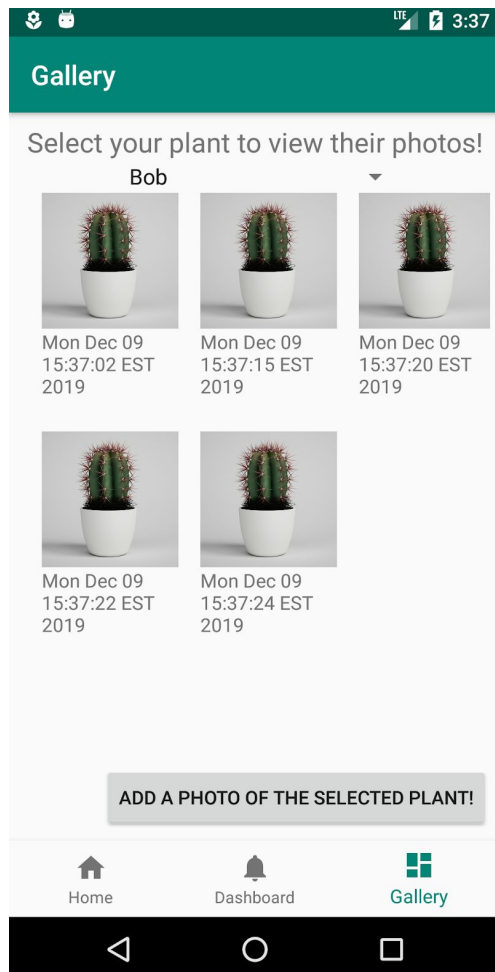


Figure 4: Plant Gallery

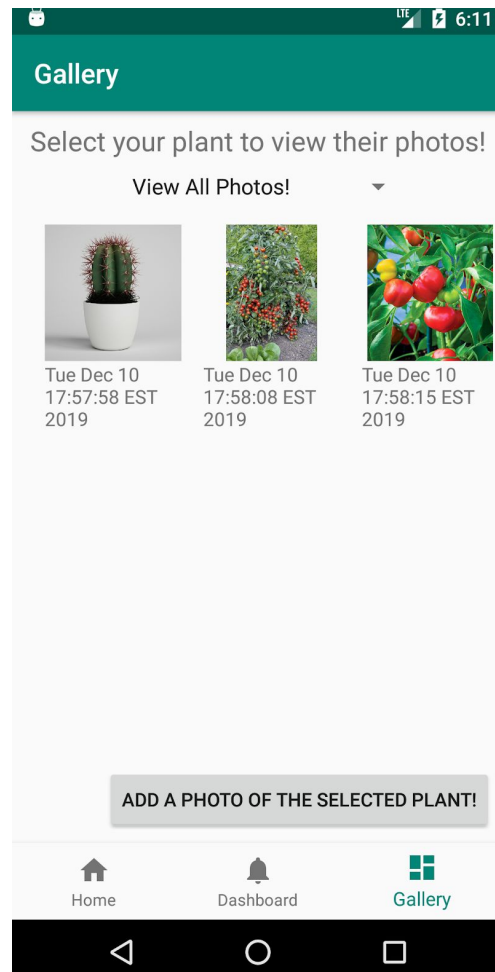


Figure 5: All Photos

Badges & Points

As learned in class, gamification is a great way to drive user engagement and draw new users towards your application. We reward the user for continuing to water the plant, for collecting more plants, and more. Our implementation focuses upon the innate human joy found in collecting shiny trinkets. These badges were custom made for the purposes of this application. The badges fall under these 5 categories: Consistency, Diversity, Photographer,

Green Thumb, & Badge of Badges. Earned badges can be viewed on the homepage of the application. Whenever a new badge is earned, a pop-up will appear alerting the user they have received a new badge! Pressing on the badge will make a pop-up appear showing the name and description of the badge. These badges were designed and custom made using an open-source vector image editing software called Inkscape.



Figure 6: Home Page Desc.

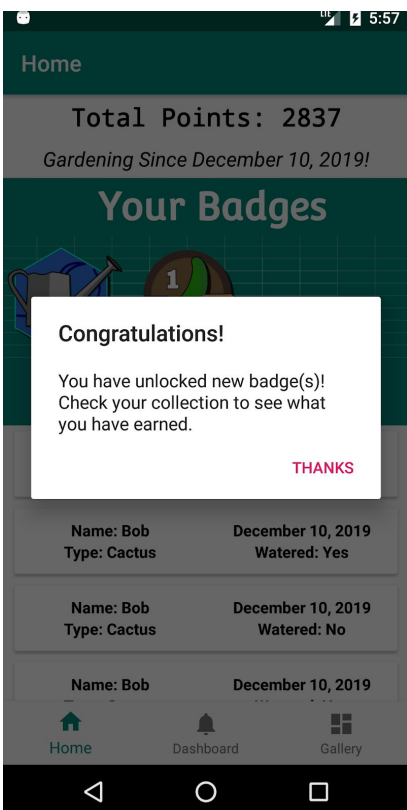


Figure 7: New Badge

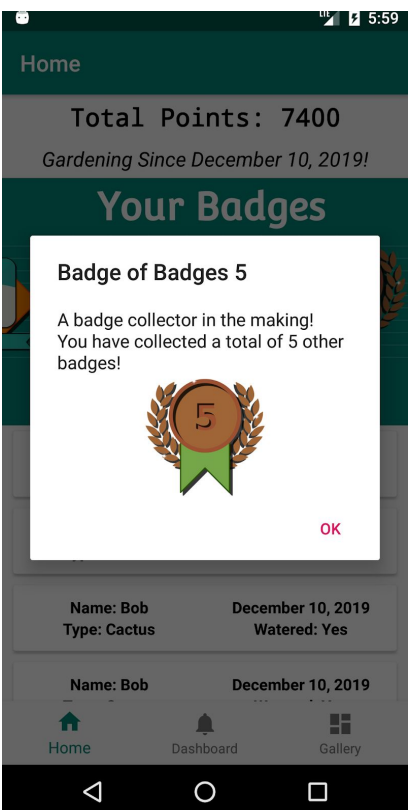


Figure 8: Badge

All Badges

Consistency is for the number of consecutive days that the user waters their plants.



Diversity is for the number of plant types that the user owns.



Photographer is for the number of photographs uploaded to the gallery.



Green Thumb is for the number of total cares (waterings).



Badge of Badges is for the number of total badges earned.



OpenWeather API

The API we decided to use is the OpenWeather API. We give the location and the API returns the weather information. We put this in the service and check the temperature so that if it is too low, we send a notification to the user to move their plants indoors.

SQLite Database

We had to create and then debug a lot of functionalities of the databases used in this project. The tables we made are:

1. **User** - stores badges, points, and location of the user
2. **Plant** - stores plant information (name, age, type, etc)
3. **Care** - stores the times of water to be displayed in the history section of home
4. **Image** - stores images from the gallery that is tied to the plant id in Plant table

We had to implement functionalities to read, insert, delete, and update entries in each of these four tables. Storing images required converting the image file to bitmap.

Organization

We organized our team by splitting ourselves based on the core features of the application (see appendix for breakdown). We also crossed over and helped each other when needed. We used many concepts learned from class in this project:

1. **LoFi prototyping** - We used the technique of LoFi prototyping to create the user interface on paper and then used “Think Aloud” testing to test the design.
2. **Recycler Views / List Views** - We used recycler views in our projects and in this project as well. We used it for the badge list view, the care history, the collection of plants, and for the gallery.
3. **Background Services** - From the homework assignment on background services, we learned how to create a background service for the notifications and for the weather check.
4. **Database** - We used the SQLite database concepts from our previous assignments to implement the complex database in this project. We created 4 tables connected through foreign keys to each other and implemented many functions (insert, update, delete, read) for the tables.
5. **Gamification** - Using the lessons learned from Foursquare, we wanted to gamify the experience to incentivize users to return to the app and continue gardening. We include badges to be collected and track points to show the user on the home page.

6. **Location Services** - We get the current location of the user using the skills we learned from the location-based assignments.
7. **Kotlin** - We all used Kotlin for the project and learned a lot during the course of this class. Kotlin is a huge improvement to Java: it's easy to use, intuitive, and clean for application development.

Appendix

Source Code located at <https://github.com/MobileAppEngg-Group1-Fall2019>

Main Authors:

Home Fragment & Related Files	Daniel Nguyen
Dashboard Fragment & Related Files	Yuqing (Sunny) Feng
Gallery Fragment & Related Files	Manish Kewalramani
SQLite Database & Related Files	Felix Zheng

We all helped each other out on different parts and contributed equally to the project.