GitHub Username: apodraz2

Gaogao

GaoGao is an app that coordinates the care of dogs among family members or friends. Finally you'll know if your dog was fed, walked, or cared for when you get home from work or school!

Intended User

Families or friends that own and care for dogs.

Features

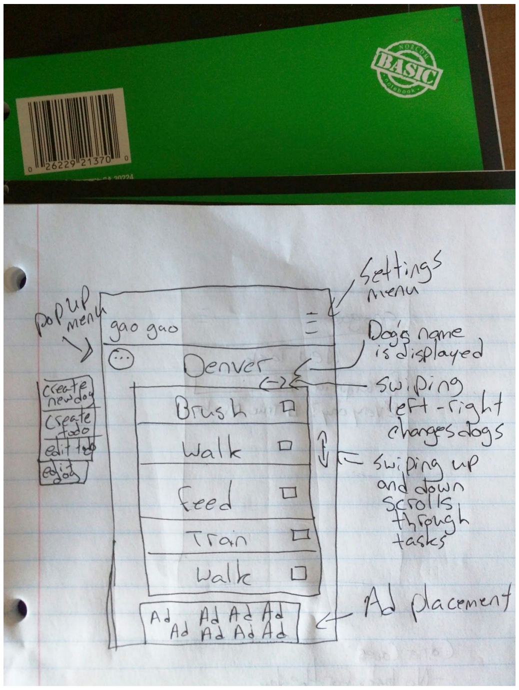
List the main features of your app. For example:

- Tracks todos for your dog
- Updates todos for every user who is responsible for that dog
- Allows for more than one dog can be tracked by more than one user

User Interface Mocks

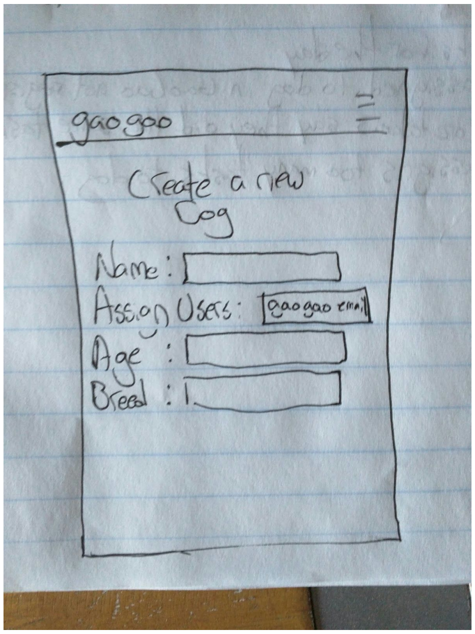
These can be created by hand (take a photo of your drawings and insert them in this flow), or using a program like Photoshop or Balsamiq.

Screen 1



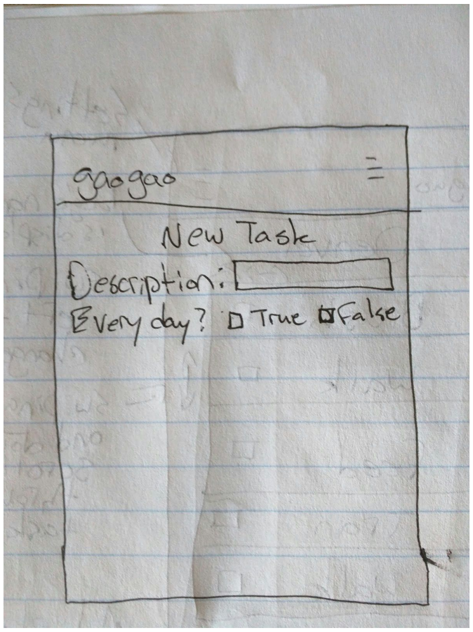
Main screen for GaoGao. Includes all the current dog’s todos. Swiping left-right changes between user’s dogs. Swiping up-down scrolls through todos.

Screen 2



This is the screen to create a new dog, pretty straightforward.

Screen 3



This is the screen to create a new todo, pretty straightforward.

Key Considerations

How will your app handle data persistence?

GaoGao will have a content provider that pulls updates from a central server (which I will build). The central server will receive data updates from user's apps.

Describe any corner cases in the UX.

The app will need to handle
-When there are no tasks assigned to a dog
-There are no dogs
-A user assigns the dog to another user whose email is not registered with GaoGao
-The event that two caretakers simultaneously update that they've completed the same task
-A user assigns too many tasks to a dog

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

Libraries:
-OkHttp for network calls
-Google Play Services (AdMob) for monetization
-Design support library for beautiful UX

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

- Create project from Android Studio template
- Configure libraries
- Implement user login functionality

Task 2: Implement UI for Each Activity and Fragment

List the subtasks. For example:

- Build UI for MainActivity
- Build UI for Create/Edit Dog
- Build UI for Create/Edit Todo
- Build UI for Settings Menu

Task 3: Build and Deploy Central Server Backend

List the subtasks. For example:

- Create all persistent models
- Create endpoints
- Secure endpoints
- Seed database with dummy data

Task 4: Implement Basic Data Updating

List the subtasks. For example:

- Implement basic data CRUD from app to endpoints
- Create update data button for now
- Ensure data is updating on endpoints

Task 5: Build Backend Data CRUD

Describe the next task. List the subtasks. For example:

- Build and Implement Content Provider
- Implement GCM alerts

Task 6: Testing

- Write JUnit tests
- Test for corner cases
- Distribute app to beta users

Task 7: Final Updates

- Correct all issues revealed during testing

- Ensure app conforms to Android design guidelines

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**”