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**"

GitHub Username: kthomas1208

Food Wheel (working title)

Description

Food Wheel helps you decide what you want to eat.

Ever been stuck with friends or your significant other on deciding where to go? Food Wheel lets you spin a wheel to pick a local restaurant. You'll get reviews, directions, and it'll save your recent picks to keep track of where you're eating.

Intended User

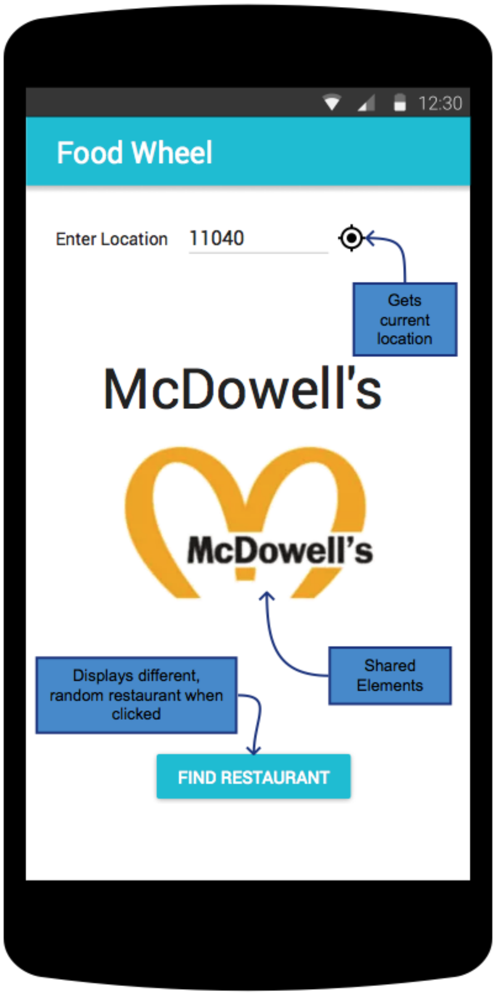
This app is for anyone looking for a place to get food. Can be especially good for families or couples deciding what to get for dinner. Could be fast food, or dine in restaurants. People looking for variety will find the app useful and they can discover new places in their area.

Features

- Chooses a random restaurant nearby
- Displays local restaurants
- Displays restaurant ratings and information
- Shows a map of the restaurant's location
- Saves favorites

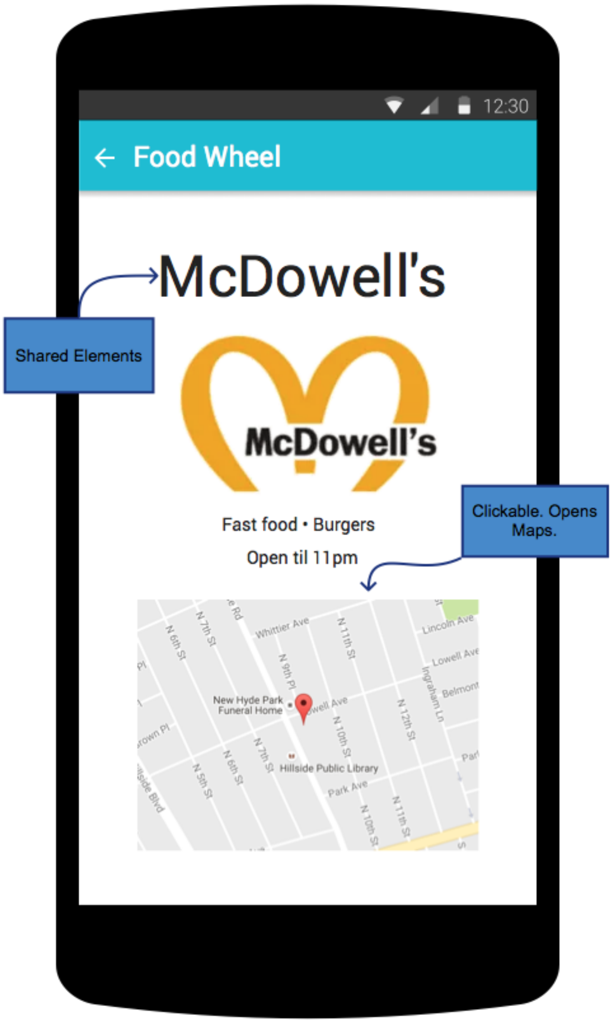
User Interface Mocks

Main Activity Screen



- Main activity screen will allow user to enter current location by typing it in or tapping the crosshairs to get current location.
- Below it, the screen will be blank or have a placeholder until the user clicks ‘Find Restaurant’
- Once user clicks ‘Find Restaurant’ a random restaurant will appear on the screen, consisting of the name of the restaurant, and a logo (if available--a placeholder image will be used otherwise).
- Tapping the logo will bring the user to the details page.
- Restaurant Title and Logo will have a shared element transition to the details page.

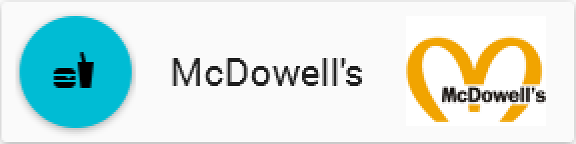
Detail Activity Screen



- Details screen will show more specific details of the restaurant selected from the main activity screen
- Restaurant title and logo will have a shared element transition from the main activity
- Type of restaurant will be displayed, along with the rating
- Will display how long the restaurant will be open until
- Actionbar should have a color based on primary colors in logo (if available)
- Logo will display, or a place holder if there is none
- A map displaying the location of the restaurant will be shown. Tapping the map will launch an external intent for Google Maps (or other map application)

]

Widget Layout



- Widget will perform the main functionality of the app without even opening it
- Tapping the button will display a random restaurant along with its logo (if available)
- Tapping the restaurant name/logo will open up the details activity page

Key Considerations

Handling Data Persistence

The app will save restaurant data from the api call to the DB. The app will then use a CP to load the data from the DB and into our views.
Users can save favorite restaurants and the app will store them into the DB.

Corner Cases In The UX

- Pressing back from the details screen will always bring you back to the main activity
- Tapping on the restaurant from the main activity will always bring you to the details screen
- If a user clicks on the map in the details screen of a particular restaurant, pressing back should being the user to the details activity
- Rotation shouldn't fetch data but keep what's already there
- If there's no data available, the user should be notified without crashing the app. There will be place-holders for empty data.

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

Picasso will be used to handle the loading and caching of restaurant images found through the API
A Content Provider generator library (need to find a good one out of the dozens that are out there)

Describe how you will implement Google Play Services.

Location will be used to determine the user's location and find nearby restaurants
Places will be used to get nearby restaurant data
Maps may be used to display a map of the restaurant's location

Next Steps: Required Tasks

Task 1: Project Setup

- Create new project and run the hello world default app
- Add to Git
- Create Github repo and push
- Configure libraries and run to make sure nothing broke

Task 2: Implement UI for Each Activity and Fragment

- Build UI for MainActivity
  - Use scrollable list for proof of concept
  - Decide whether to use animated spinning wheel or scrollable list
- Build UI for Detail Fragment

Task 3: Implement Location Service

- Test Location service and display current location as a string

Task 4: Implement Google Places

- Test Places service and make an API call to get nearby restaurants using the location. Ensure data is received in JSON format

Task 5: Implement Content Provider

- Create schema for DB (hopefully the CP generator will expedite this process)
- Test Content Provider by saving the JSON data to the DB

Task 6: Bind data to views

- Connect the adapters to the CP data
- Update the views accordingly using the data from the DB

Task 7: Implement favorites

- Add a favorites button to the details activity that will save to the DB

Task 8: Get Map working

- Add a map with the location of the restaurant on the Details activity

---

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**"