

# Project 4

Author: Jingyi(Olivia) Wu<andrew id: jingyiw2>, Keqing Xing<andrew id: kxing>

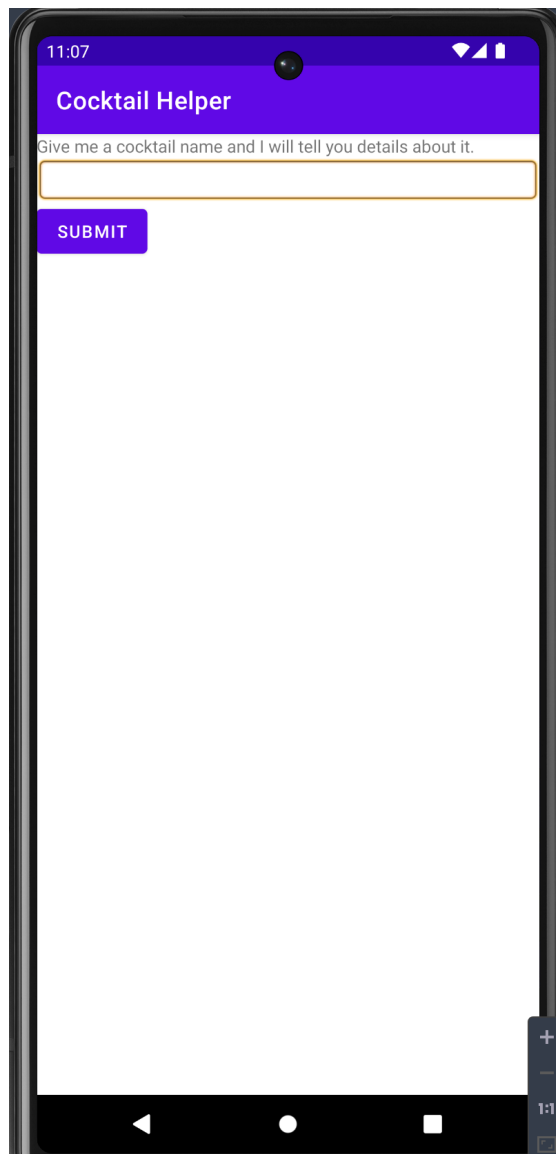
Description: Our application is a cocktail helper for those who are interested in all kinds of cocktails and want to try some homemade cocktail. Users input a cocktail name and our app fetches and displays detailed information about it.

Below is how our app meets all requirements.

1. **Implement a native Android application**

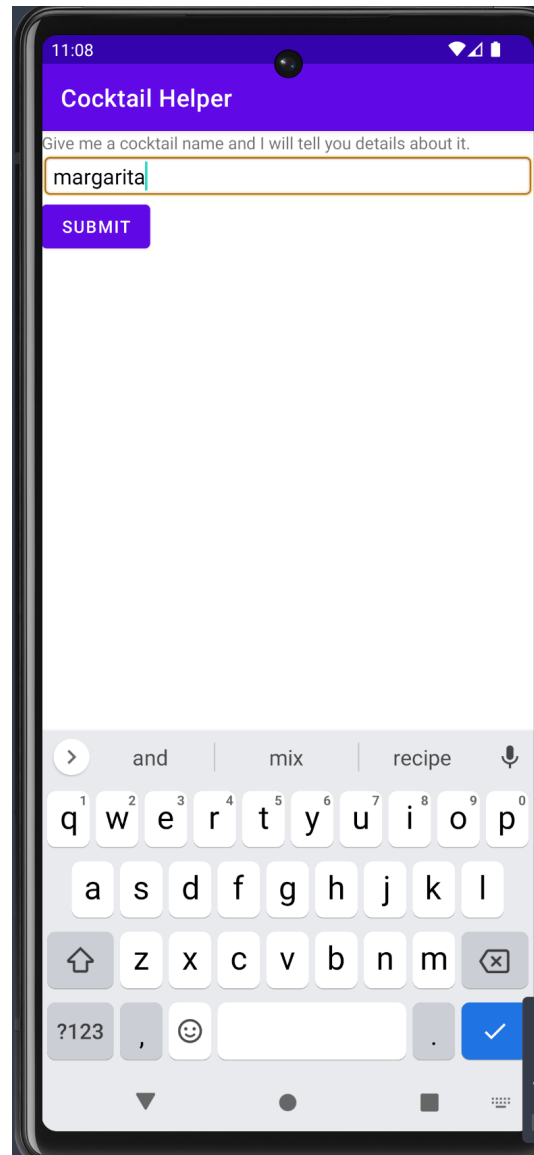
- a. **at least three different kinds of Views**

There are several text views, imgae view, edit text and button view in my app. See /res/layout/content\_main.xml for details.



**b. Requires input from the user**

Here is a screenshot of user search for margarita.



**c. Makes an HTTP request (using an appropriate HTTP method) to your web service**

Our app does a HTTP get request in `getData.java`. Request url is `"https://frozen-reef-96952.herokuapp.com/getCocktailInfo?searchWord="+searchTerm` where `searchTerm` is user's input cocktail name. This request will go to my web service deployed on heroku to fetch data, parsed the return html to get json-formatted data about specific cocktail. Then we parse the json data and render them on app.

**d. Receives and parses an XML or JSON formatted reply from your web service**

an example of json-formatted reply from web service is as below.

```
{
  "strInstructionsDE": "Reiben Sie den Rand des Glases mit der Limettenscheibe, damit das Salz daran haftet. Achten Sie darauf, das",
  "strImageSource": "https://commons.wikimedia.org/wiki/File:Klassiche_Margarita.jpg",
  "strIngredient10": null,
  "strDrink": "Margarita",
  "strIngredient12": null,
```

```

"strIngredient11": null,
"strIngredient14": null,
"strCategory": "Ordinary Drink",
"strAlcoholic": "Alcoholic",
"strIngredient13": null,
"strIngredient15": null,
"strCreativeCommonsConfirmed": "Yes",
"strIBA": "Contemporary Classics",
"strVideo": null,
"strTags": "IBA,ContemporaryClassic",
"strInstructions": "Rub the rim of the glass with the lime slice to make the salt stick to it. Take care to moisten only the outside of the glass with the lime juice. Add the tequila, triple sec, lime juice, salt, and ice. Stir well and serve. \r\nAvere cura di inumidire solo il bordo del bicchiere con la fetta di lime per far aderire il sale.\r\nAvere cura di inumidire solo il bordo del bicchiere con la fetta di lime per far aderire il sale.",
"strIngredient1": "Tequila",
"strIngredient3": "Lime juice",
"strIngredient2": "Triple sec",
"strIngredient5": null,
"strIngredient4": "Salt",
"strIngredient7": null,
"strIngredient6": null,
"strIngredient9": null,
"strInstructionsFR": null,
"strIngredient8": null,
"idDrink": "11007",
"strInstructionsES": null,
"strInstructionsIT": "Strofina il bordo del bicchiere con la fetta di lime per far aderire il sale.\r\nAvere cura di inumidire solo il bordo del bicchiere con la fetta di lime per far aderire il sale.",
"strGlass": "Cocktail glass",
"strMeasure12": null,
"strMeasure13": null,
"strMeasure10": null,
"strMeasure11": null,
"strImageAttribution": "Cocktailmarler",
"dateModified": "2015-08-18 14:42:59",
"strDrinkAlternate": null,
"strDrinkThumb": "https://www.thecocktaildb.com/images/media/drink/5noda61589575158.jpg",
"strInstructionsZH-HANT": null,
"strMeasure9": null,
"strMeasure7": null,
"strMeasure8": null,
"strMeasure5": null,
"strMeasure6": null,
"strMeasure3": "1 oz ",
"strMeasure4": null,
"strMeasure1": "1 1/2 oz ",
"strMeasure2": "1/2 oz ",
"strInstructionsZH-HANS": null,
"strMeasure14": null,
"strMeasure15": null
}

```

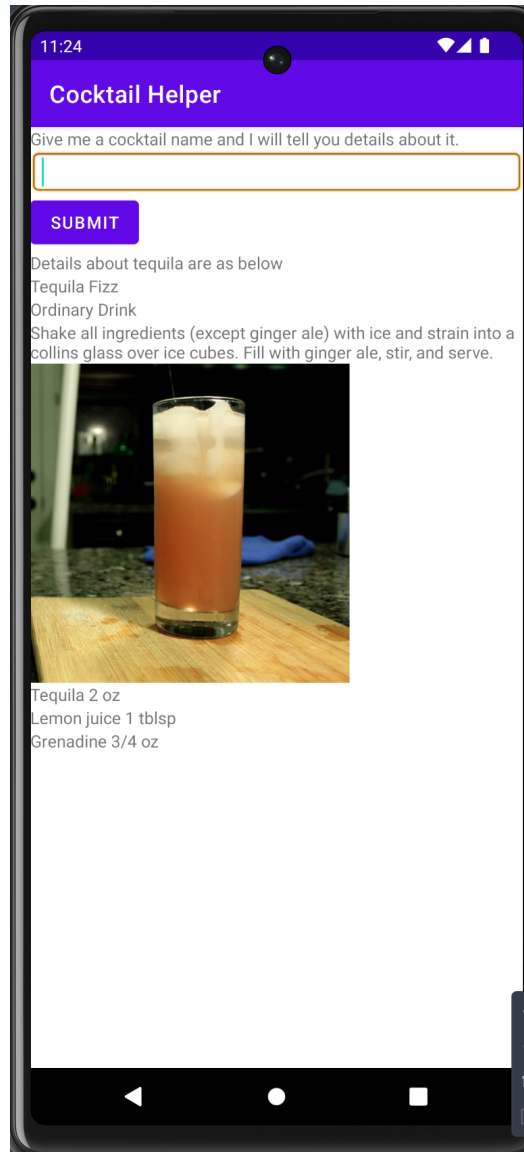
e. **Displays new information to the user**

Here is a screenshot of returned information about margarita.



f. **Is repeatable**

User can type in another cocktail name and click on submit button to get another cocktail's details. Here is a screenshot of "tequila".



2. **Implement a web service, deployed to Heroku** (url of web service deployed on heroku is <https://frozen-reef-96952.herokuapp.com/getCocktailInfo?searchWord=> project directory is task1.
  - a. Implement a simple (can be a single path) API.  
 In my web service project,  
 model: CocktailModel.java; DashboardModel.java  
 view: index.jsp and result.jsp;  
 controller: CocktailServlet
  - b. Receives an HTTP request from the native Android application  
 CocktailServlet.java receives the HTTP GET request with the argument "searchWord". It passes this search string on to the CocktailModel, and send request to API to fetch json-formatted data.
  - c. Executes business logic appropriate to your application  
 CocktailModel.java makes an HTTP request to: <https://www.thecocktaildb.com/api/json/v1/1/search.php?s=>+searchWord.  
 It then parses the JSON response and extracts the parts it needs to respond to the Android application.

- d. Replies to the Android application with an XML or JSON formatted response  
Result.jsp formats the response to the mobile application in json, as is shown above.

3. Handle error conditions (no need for writeup in this section)
4. Log useful information
5. Store the log information in a database

fields	description
search word	keywords searched by users
request time	time when android app request the web service
response time	time when web service get response from 3rd party API
latency	the gap between request time and response time in ms
user agent	full user agent information from request header
device model	users' phone model information parsed from user agent ( blank when the device is not mobile)
response status	if search succeed or not ()
request from app	request content from Android application
request to 3rd party API	request content to cocktailDB
response from 3rd party API	response content from cocktailDB

Atlas connection string: "mongodb+srv://keqingxing:kq123456@cluster0.od4bwm4.mongodb.net/?retryWrites=true&w=majority"

6. Display operations analytics and full logs on a web-based dashboard
  - a. A unique URL addresses a web interface dashboard for the web service.  
<https://frozen-reef-96952.herokuapp.com/>
  - b. The dashboard displays at least 3 interesting operations analytics.
    - i. top 10 high frequency device models
    - ii. top 10 high frequency popular searched cocktails

### Top 10 Popular Search Words

search word	times
tequila	6
margarita	6
brandy	3
cat	3
rum	2
vodka	1
gin	1

**Average Latency: 252 ms**

### Top 10 Popular Device Models

device type	times
	13
sdk_gphone64_arm64	9

- iii. The dashboard displays formatted full logs.

[illegible]

## 7. Deploy the web service to Heroku

<https://frozen-reef-96952.herokuapp.com/getCocktailInfo>

<https://frozen-reef-96952.herokuapp.com/>

<https://frozen-reef-96952.herokuapp.com/getCocktailInfo?searchWord=margarita>