Project 4 Task 2 – Activity Recommendation App

By Hsiu-Yuan Yang (Andrew ID: hsiuyuay)

Description:

My mobile app will recommend the users activities depending on which choice they pick.

There will be 3 options, including just finding a random activity, finding a random activity with a preferred type, and finding a random activity with a certain number of participants. For the 2nd and 3rd option, the user will also have to enter their preferred type or number of participants.

With the input, the system will then ping Bored API for the corresponding activity recommendation and will display the activity name, the activity type, the number of participants, and the predicted price.

The following pages demonstrate how my application meets the task requirements.

1. Implement a native Android application

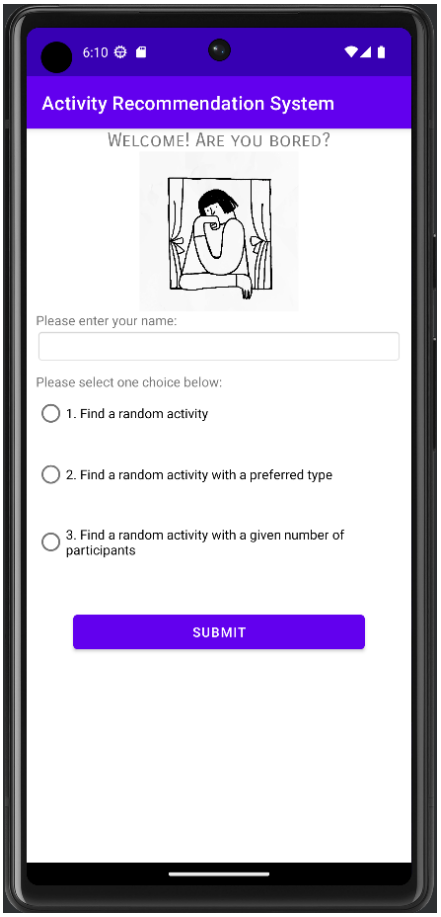
The name of my native Android application project in Android Studio is: Project4Task2.

a. Has at least three different kinds of views in your Layout (TextView, EditText, ImageView, etc.)

My application uses 4 TextViews, 1 ImageView, 1 EditText, 1 RadioGroup (containing 3 RadioButtons, 1 Spinner, 1 Slider), and 1 Button.

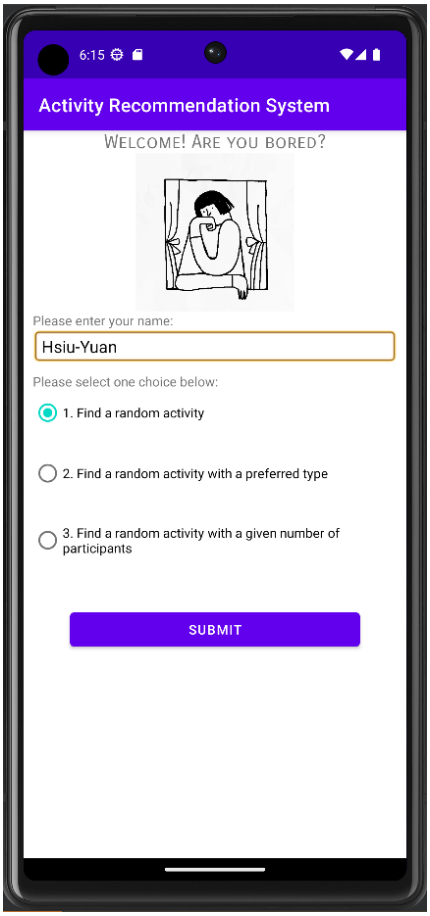
See content_main.xml for details of how they are incorporated into the LInearLayout.

Here is a screenshot of the layout before the picture has been fetched.



b. Requires input from the user

Here is a screenshot of the user "Hsiu-Yuan" searching for a random activity (choosing 1st option).



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

My application does an HTTP GET request in GetActivity.java. The HTTP request is:

"https://hsiuyuay-fuzzy-rotary-phone-4xg6pq44g5wcjw97-8080.preview.app.github.dev/getActivity?" + searchTermEncoded, where searchTermEncoded is the user's search term encoded.

The searchTermEncoded is composed of the user's name and his / her radio button selection (including parameters where applicable). For example, the searchTermEncoded for user "Hsiu-Yuan" picking the 1st option is "username=Hsiu-Yuan&searchapi=activity".

The backgroundSearch method makes this request of my web application, parses the returned JSON string to create the display response on the app.

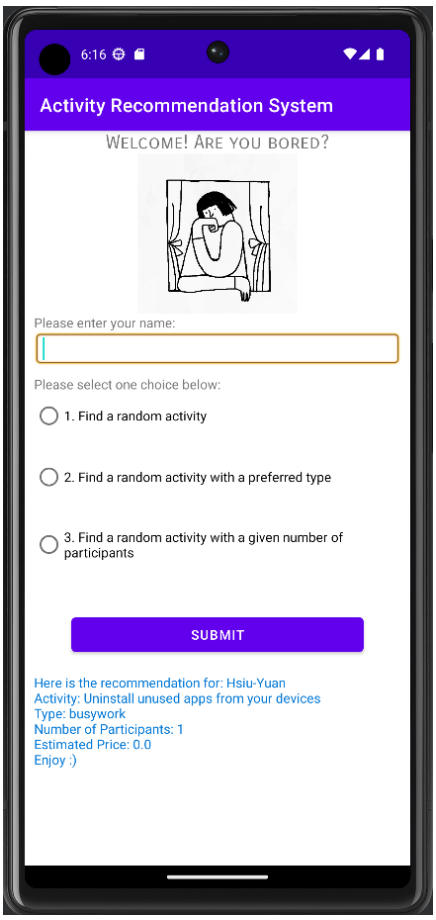
d. Receives and parses an XML or JSON formatted reply from the web service

An example of the JSON reply from the web service is:

{"username":"Hsiu-Yuan","activity":"Uninstall unused apps from your devices","type":"busywork","participants":1,"price":0.0}

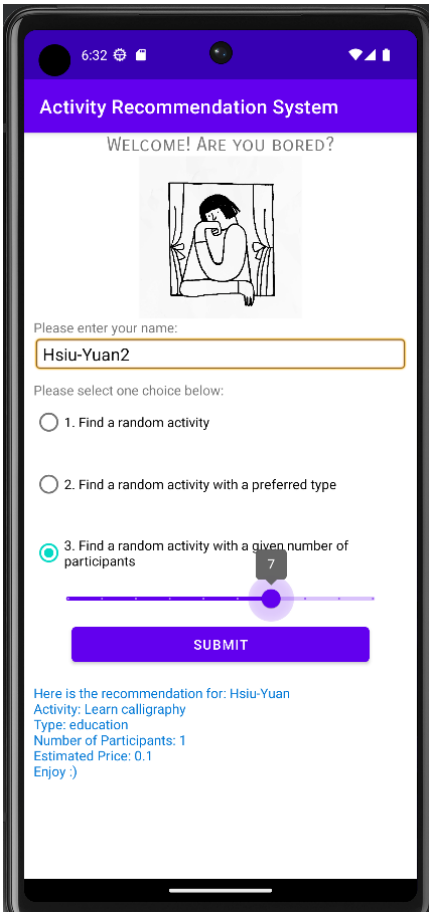
e. Displays new information to the user

Here is the screen shot after the activity recommendation has been returned.



f. Is repeatable (i.e., the user can repeatedly reuse the application without restarting it)

The user can input other search parameters and hit submit to get another recommendation without having to restart the app. Here is an example of user "Hsiu-Yuan2" searching for another random activity with a given number of participants "7" (choosing 3rd option) after she received the recommendation.



2. Implement a web application, deployed to codespace

The URL of my web service deployed to codespace is:
“https://hsiuyuay-fuzzy-rotary-phone-4xg6pq44g5wcjw97-8080.preview.app.github.dev/”

a. Using an HttpServlet to implement a simple (can be a single path) API

In my web app project:
Model: ActivityRecommendationModel.java
View: Dashboard.jsp
Controller: ActivityRecommendationServlet.java

b. Receives an HTTP request from the native Android application

ActivityRecommendationServlet.java receives the HTTP GET request with the argument “username” and “searchapi”. It then passes these terms to ActivityRecommendationModel.java for processing.

c. Executes business logic appropriate to your application

ActivityRecommendationModel.java makes an HTTP request to:
"https://www.boredapi.com/api/" + searchTerm, where searchTerm is the searchapi received from the app. It then parses the JSON string received 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

Leverage Gson and RecommendationToAPP.java to create a JSON formatted string with the needed information of my own design:

{"username":"Hsiu-Yuan","activity":"Uninstall unused apps from your devices","type":"busywork","participants":1,"price":0.0}

See RecommendationToApp.java and ActivityRecommendationModel.java for more details.

3. Handle error conditions – does not need to be documented

4. Log useful information

Below is the list of what I am logging onto MongoDB:

- Username: the user who submitted the app request
- Timestamp: the timestamp when the user request was made
- Response Generated Status: whether a recommendation has been provided from the API
- Response to App (only logged if the recommendation has been provided from the API): the recommendation from the API, including the following four pieces of information:
    - Activity: activity name recommended
    - Type: activity type recommended
    - Number of participants: number of participants recommended
    - Estimated price: estimated price recommended
- Request processing time (ms): the processing time spent to handle the user request

The reason why I chose the above information is:

i. For username: to gain a sense of user churn rate. If the user continues to utilize this app, it could mean that recommendation provided are very useful in the user's real life. From this, we could further conduct user sentiment analysis by collecting more data and improve the app where needed.
ii. For timestamp: to discover underlying patterns in the time requested. It will then help us monitor the app performance and make sure it does not crash during peak times.
iii. For response generated status: to examine whether BoredAPI is able to support us to process most user requests.
iv. For response to app: to document what recommendations have been provided to the users. Depending on the responses, it also gives us a sense of what might be the most popular searching parameters.
v. For request processing time: to check if the user request can be completed within a short timeframe.

5. Store the log information in a database

My Atlas connection string is:

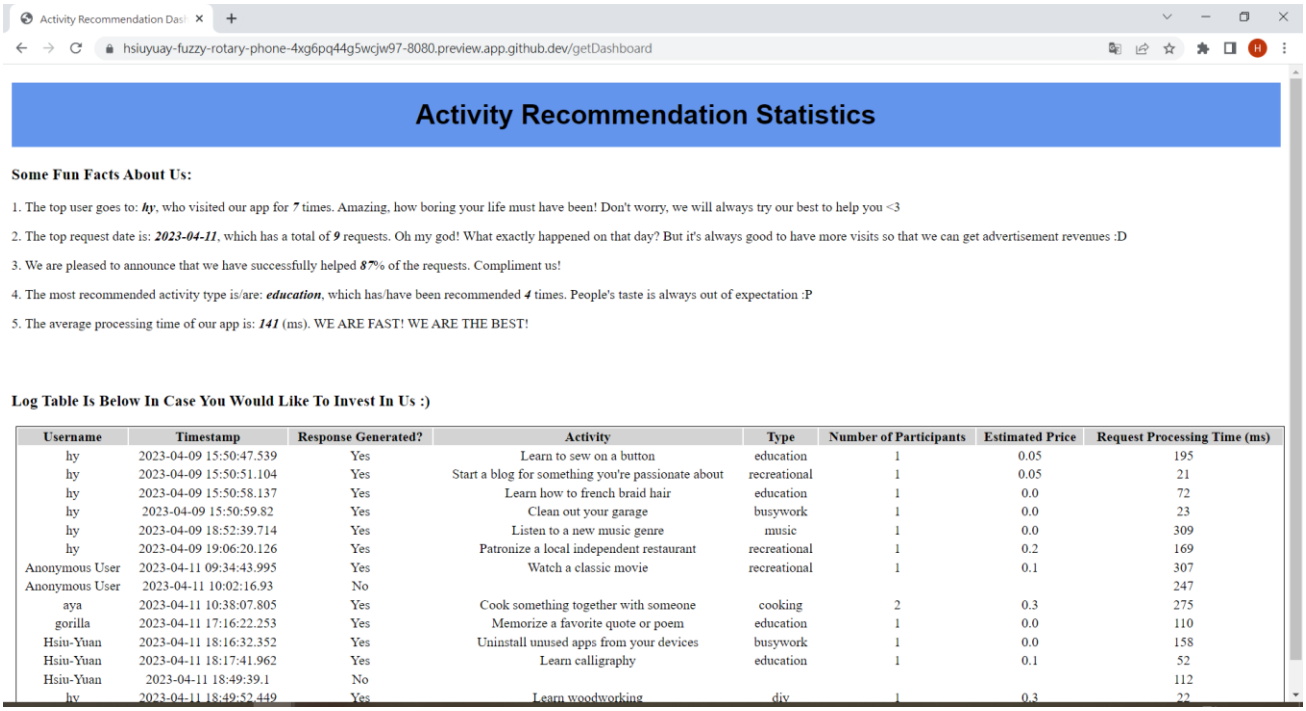
"mongodb://hsiuyuay:DSProject4@ac-cwk6wyo-shard-00-01.xstvb87.mongodb.net:27017,ac-cwk6wyo-shard-00-00.xstvb87.mongodb.net:27017,ac-cwk6wyo-shard-00-02.xstvb87.mongodb.net:27017/test?w=majority&retryWrites=true&tls=true&authMechanism=SCRAM-SHA-1";

6. Display operations analytics and full logs on a web-based dashboard

URL:

"https://hsiuyuay-fuzzy-rotary-phone-4xg6pq44g5wcjw97-8080.preview.app.github.dev/getDashboard"

Screenshot:



Activity Recommendation Statistics

Some Fun Facts About Us:

1. The top user goes to: ***hy***, who visited our app for ***7*** times. Amazing, how boring your life must have been! Don't worry, we will always try our best to help you <3
2. The top request date is: ***2023-04-11***, which has a total of ***9*** requests. Oh my god! What exactly happened on that day? But it's always good to have more visits so that we can get advertisement revenues :D
3. We are pleased to announce that we have successfully helped ***87***% of the requests. Compliment us!
4. The most recommended activity type is/are: ***education***, which has/have been recommended ***4*** times. People's taste is always out of expectation :P
5. The average processing time of our app is: ***141*** (ms). WE ARE FAST! WE ARE THE BEST!

Log Table Is Below In Case You Would Like To Invest In Us :)

Username	Timestamp	Response Generated?	Activity	Type	Number of Participants	Estimated Price	Request Processing Time (ms)
hy	2023-04-09 15:50:47.539	Yes	Learn to sew on a button	education	1	0.05	195
hy	2023-04-09 15:50:51.104	Yes	Start a blog for something you're passionate about	recreational	1	0.05	21
hy	2023-04-09 15:50:58.137	Yes	Learn how to french braid hair	education	1	0.0	72
hy	2023-04-09 15:50:59.82	Yes	Clean out your garage	busywork	1	0.0	23
hy	2023-04-09 18:52:39.714	Yes	Listen to a new music genre	music	1	0.0	309
hy	2023-04-09 19:06:20.126	Yes	Patronize a local independent restaurant	recreational	1	0.2	169
Anonymous User	2023-04-11 09:34:43.995	Yes	Watch a classic movie	recreational	1	0.1	307
Anonymous User	2023-04-11 10:02:16.93	No					247
aya	2023-04-11 10:38:07.805	Yes	Cook something together with someone	cooking	2	0.3	275
gorilla	2023-04-11 17:16:22.253	Yes	Memorize a favorite quote or poem	education	1	0.0	110
Hsiu-Yuan	2023-04-11 18:16:32.352	Yes	Uninstall unused apps from your devices	busywork	1	0.0	158
Hsiu-Yuan	2023-04-11 18:17:41.962	Yes	Learn calligraphy	education	1	0.1	52
Hsiu-Yuan	2023-04-11 18:49:39.1	No					112
hy	2023-04-11 18:49:52.449	Yes	Learn woodworking	diy	1	0.3	22