Project 4 Task 1 - Find Activity App

By Enliang (Leo) Wu

Description:

My application takes a username string from the user, and to give user recommend activity by clicking “Find Activity” Button. User is able to choose “Do it” or “ Try another activity” by clicking the corresponding button and also able to return to find activity just by clicking “Back to find activity” button.

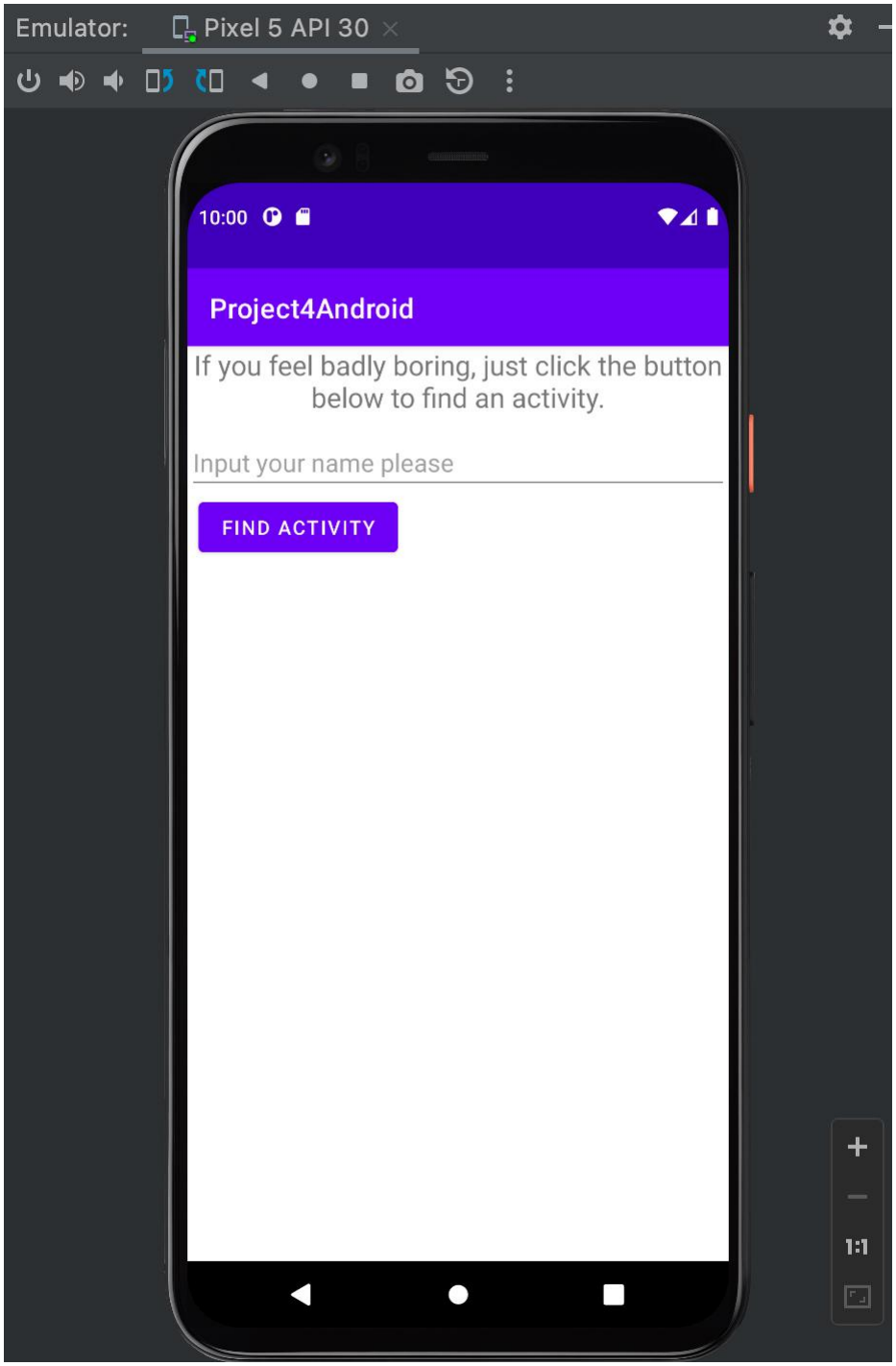
Here is how my application meets the task requirements:

1. Implement a native Android application

The name of my native Android application project in Android Studio is:
Project4Android

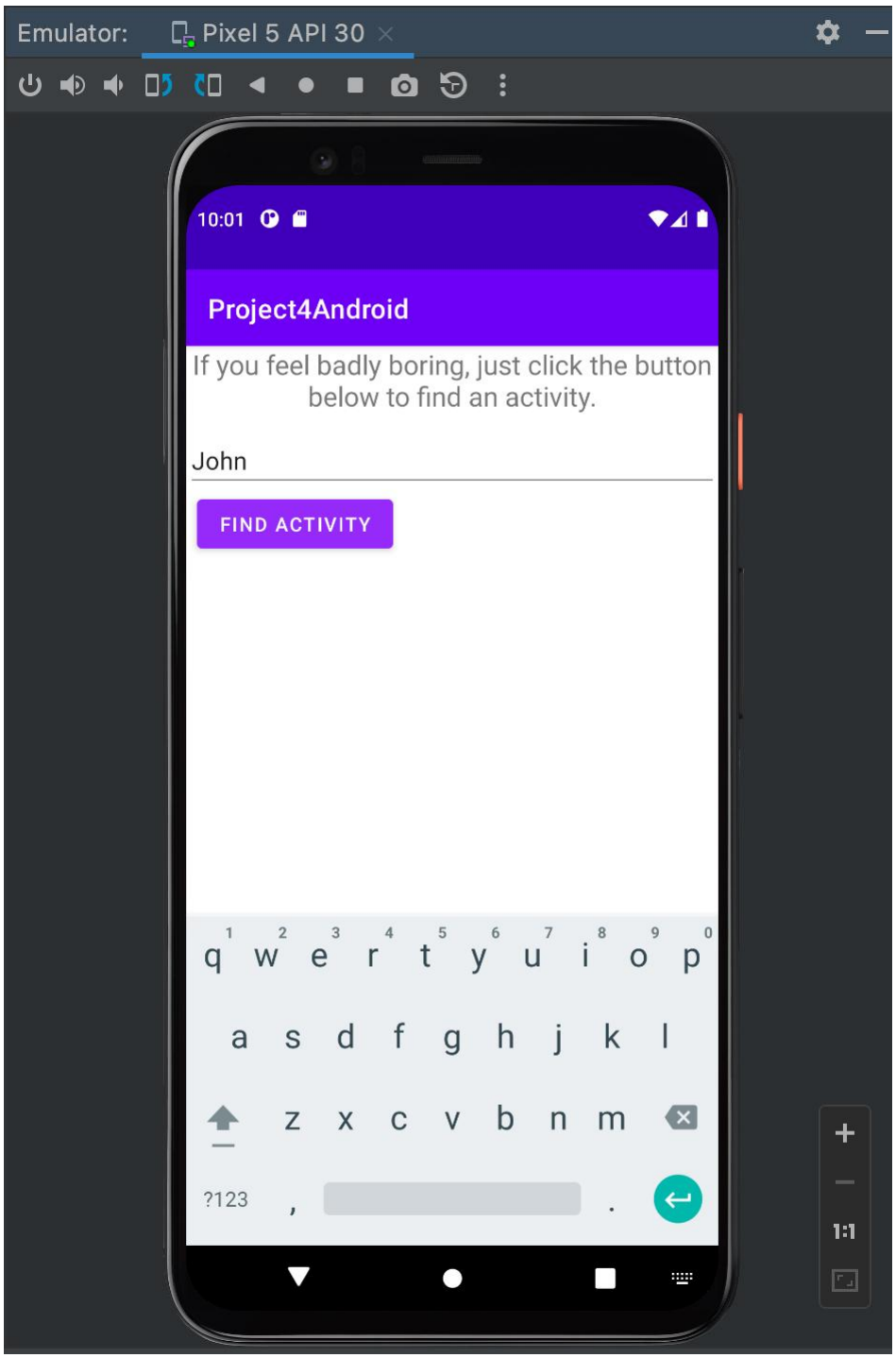
1.1 Has at least three different kinds of Views in your Layout

My Application uses TextView, EditText, Button views. See content_main.xml for details. Here is a screenshot of the layout before user has found any activity.



1.2 Requires input from the user

Here is a screenshot of user having inputted his/her name.



1.3 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://frozen-castle-25997.herokuapp.com/activity/user=” + userName
Where userName is the user inputted name.

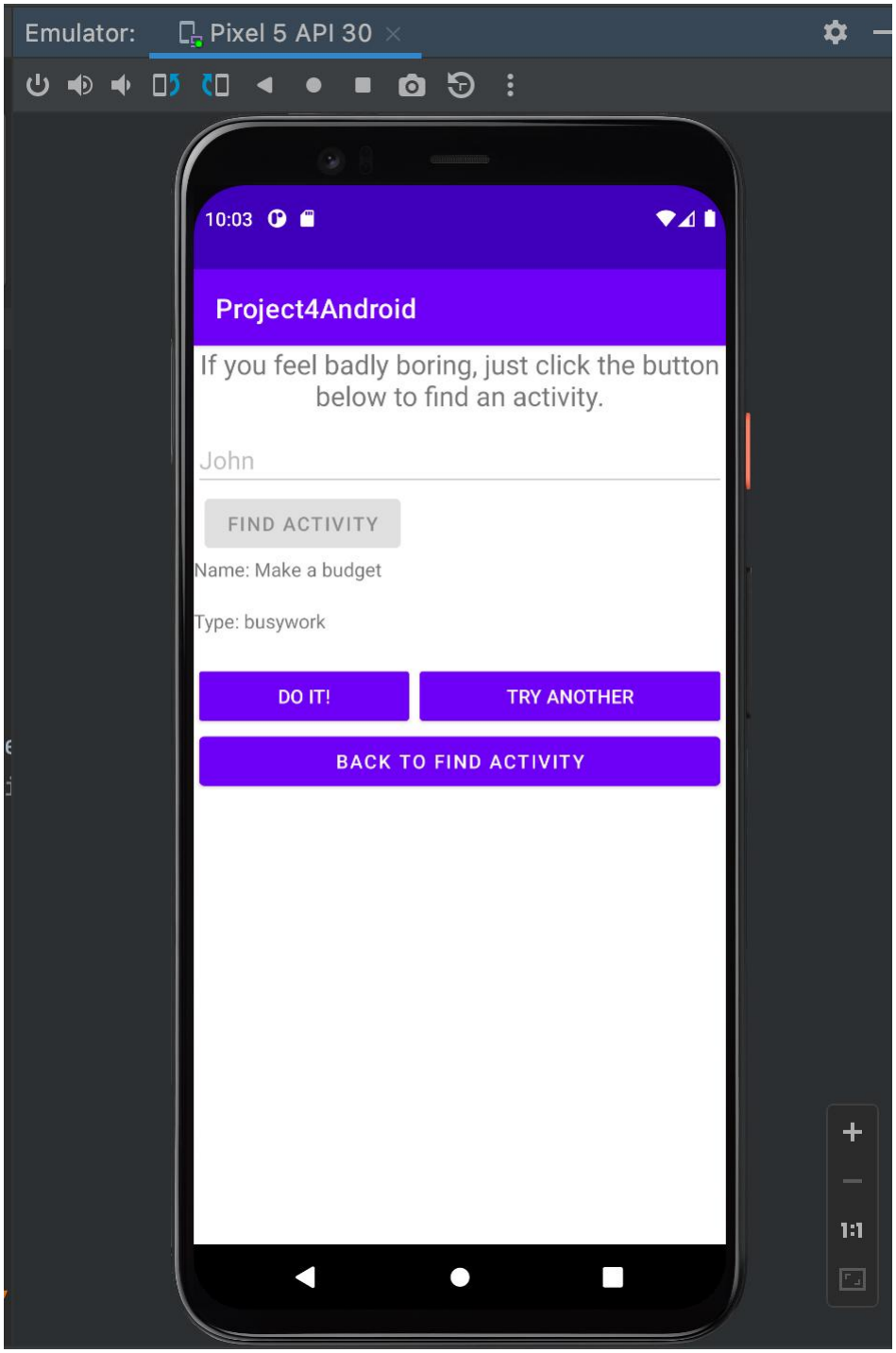
1.4 Receives and parses an XML or JSON formatted reply from your web service

An example of the JSON reply is:

```
{
    result: true,
    name: "Have a paper airplane contest with some friends",
    id: "8557562",
    type: "social",
    message: ""
}
```

1.5 Displays new information to the user

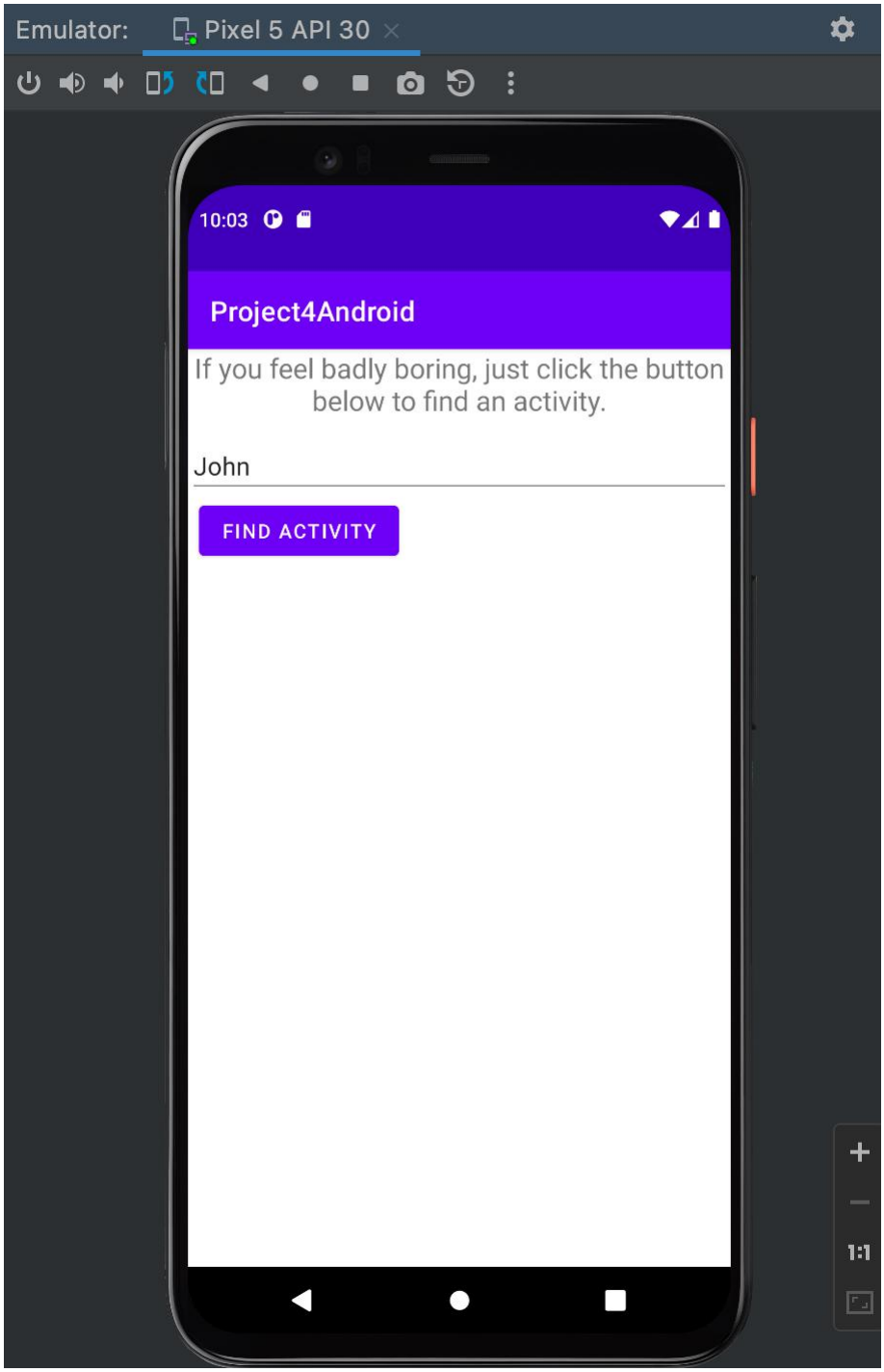
Here is a screenshot after user clicking “Find Activity” button.



The random activity type and name will show to the user. Then user can choose to do this activity by clicking “DO IT!” button or get another random activity by clicking “TRY ANOTHER” button.

1.6 Is repeatable (I.e. the user can repeatedly reuse the application without restarting it.)

User can just click the “BACK TO FIND ACTIVITY” button to go back to find activity. Here is a screenshot of my application restore to default view and “Find Activity” button is enabled again. User can input another name now.



2. Implement a web service, deployed to Heroku

The app name of my web service deployed to Heroku is:
frozen-castle-25997
URL is:
https://frozen-castle-25997.herokuapp.com/
The project directory name is:
Project4Task1

2.1 Implement a simple (can be a single path) API.

In my web app project:
Controller: leow/project4task1/ActivityProvider.java
Route:
/activity for getting a random activity
/activity/doit client action do this activity
/activity/dislike client action dislike this activity

2.2 Receives an HTTP request from the native Android application

/activity receives an HTTP GET request with parameter user and passes the user to the controller.
/activity/doit receives an HTTP GET request with 2 parameters: user and id, user is user name from android client, id is an id of an activity. They are both been passing to the controller.
/activity/dislike is the same as /activity/doit.

2.3 Executes business logic appropriate to your application. This includes fetching XML or JSON information from some 3rd party API and processing the response.

ActivityProvider.java makes an HTTP GET request to 3rd party boredapi which URL is “http://www.boredapi.com/api/activity/”.
It will reply a JSON formatted random activity information to my web application.
An example of the JSON:

```
{
    activity: "Plan a vacation you've always wanted to take",
    type: "relaxation",
    participants: 1,
    price: 0,
    link: "",
    key: "7265395",
    accessibility: 0.05
}
```

ActivityProvider.java will package the useful information to android client.

2.4 Replies to the Android application with an XML or JSON formatted response. The schema of the response can be of your own design.

makeJSONResponse method in ActivityProvider.java will make a JSON response which schema of my own design is:

```
{
    result: <a boolean value true or false>,
    name: <a string value>,
    id: <a string value>,
    type: <a string value>,
    message: <a string value>
}
```

In android client I use the boolean value result to check whether the request succeeded or not. And message is used to show error message while result is false.
Name is the name of an activity.
Id is the key code of an activity.
Type is the type of an activity.