Project 4 Task 1 – Currency Converter

Jing Zhan

Description:

My application takes converts the currency by prompting user to input a number, choose the from and to currency and give back the converting result.

Here is how my application meets the task requirements

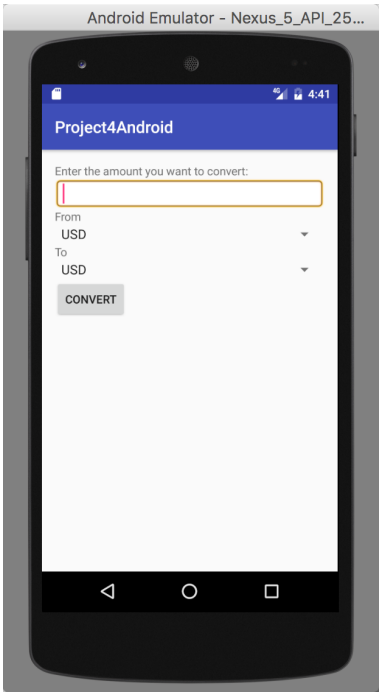
## 1. Implement a native Android application

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

### 1.1. Has at least two different kinds of views in your Layout (TextView, EditText, ImageView, etc.)

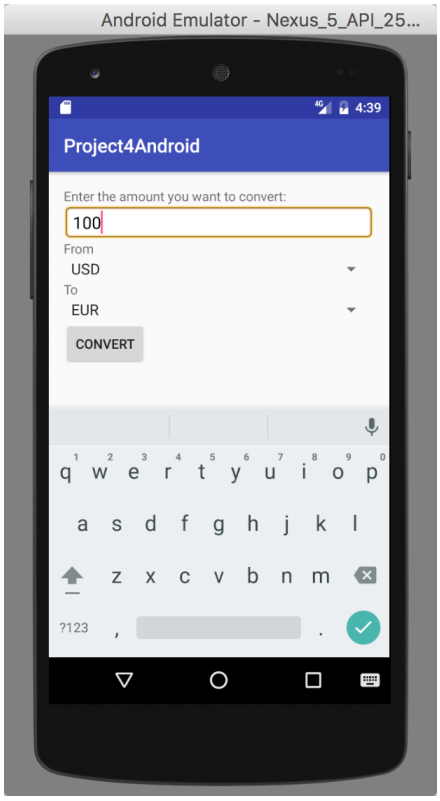
My application uses TextView, EditText, Button, and Spinner. See content_main.xml for details of how they are incorporated into the LinearLayout.

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




### 1.2. Requires input from the user

Here is a screenshot of the user converting USD to EUR of 100.




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

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

```
"https://shielded-cliffs-71886.herokuapp.com/MyAppServlet?searchAmount=" + searchAmount + "&searchFrom=" + searchFrom + "&searchTo=" + searchTo;
```

where searchAmount is the user input number; searchFrom is the from currency user choose; searchTo is the to currency the user choose.

The search method makes this request of my web application, parses the returned JSON to find the picture URL, fetches the convert result, and returns the convert number.

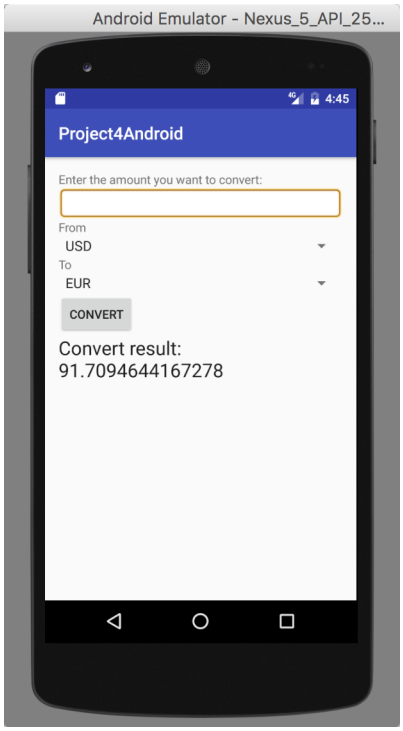
1.4. Receives and parses an XML or JSON formatted reply from the web service

An example of the JSON reply is:

{result: 31.02}

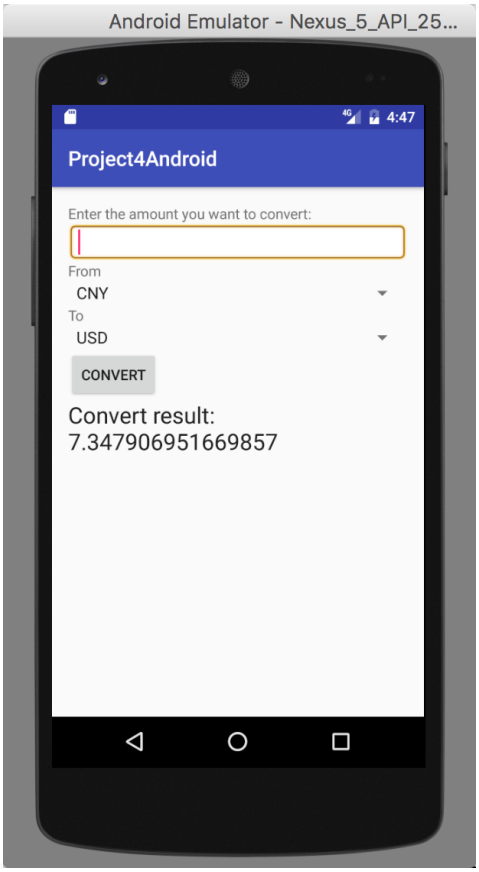
1.5. Displays new information to the user

Here is the screen shot after the result has been returned.




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

The user can type in another search term and hit Submit. Here is an example of having typed in 44 and from CNY to USD.




2. Implement a web application, deployed to Heroku

The URL of my web service deployed to Heroku is https://shielded-cliffs-71886.herokuapp.com/
The project directory name is Project4Task1.

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

In my web app project:
Model: ConverterModel.java
View: login.jsp, result.jsp
Controller: MyAppServlet.java

## 2.2. Receives an HTTP request from the native Android application

MyInterestingPictureServerServlet.java receives the HTTP GET request with the argument "searchAmount", "searchFrom", "searchTo". It passes these search strings on to the model.

## 2.3. Executes business logic appropriate to your application

InterestingPictureModelUsingWS.java makes an HTTP request to:
http://api.fixer.io/latest?symbols=from,to
It then parses the JSON response and extracts the parts it needs to respond to the Android application.

## 2.4. Replies to the Android application with an XML or JSON formatted response.

Response.jsp formats the response to the mobile application in a simple JSON format of my own design:
{result:50.4}