Project 4 Task 2 – Payment Calculation

By Chenjie Zhao

Andrew Id: Chenjie1

Description: my app takes in two parameters from user: Price and down payment percentage. Then it will display a fixed 30-year monthly payment, current loan rate and monthly insurance payment.

Here is how my application meets the task requirements

1. Implement a native Android application

The name of my native Android application project in Eclipse is:

Project4Task2Android

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

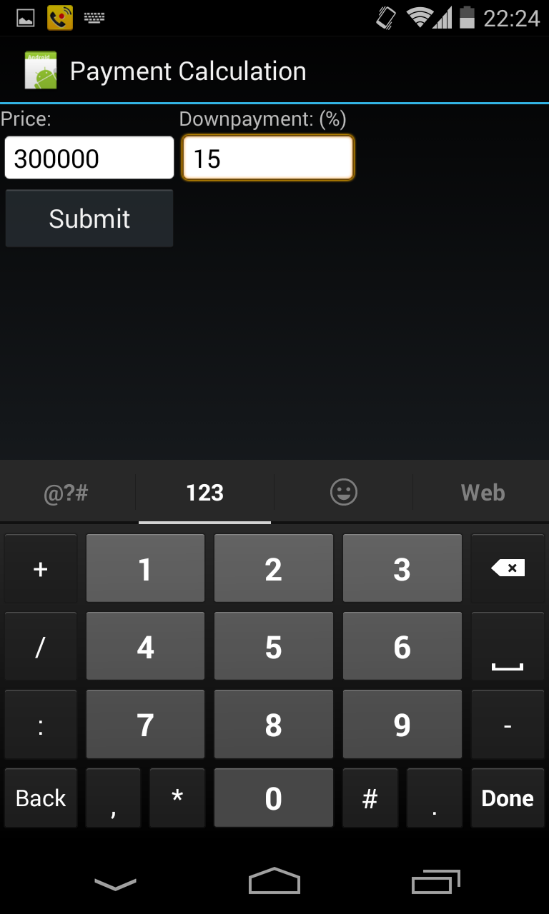
ImageView, etc.)

My application uses TextView, EditText, Button. See main.xml for details of how they are incorporated into the TableLayout.

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 calculating with a price: 300000 and down payment percentage: 15.



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

My application does the HTTP request in GetPayment.java. The HTTP request is:

“http://1-dot-jackproject4task2.appspot.com/project4task2gae?price=" + price + "&downpayment=" + down

where price and down stands for the user's input terms.

The search method makes this request of my web application, parses the returned

XML to find the result URL, and return it to my app.

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

An example of the XML reply is:

<payment loanType="thirtyYearFixed">

<rate>4.22</rate>

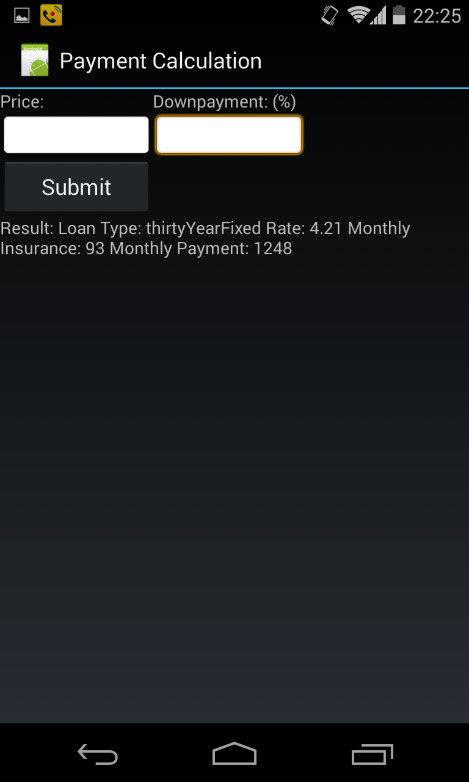
<monthlyPrincipalAndInterest>125</monthlyPrincipalAndInterest>

<monthlyMortgageInsurance>9</monthlyMortgageInsurance>

</payment>

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 price: 400000, down payment: 10.

2. Implement a web application, deployed to Google App Engine

The name of the Google App Engine project in Eclipse is:

Project4Task2GAE

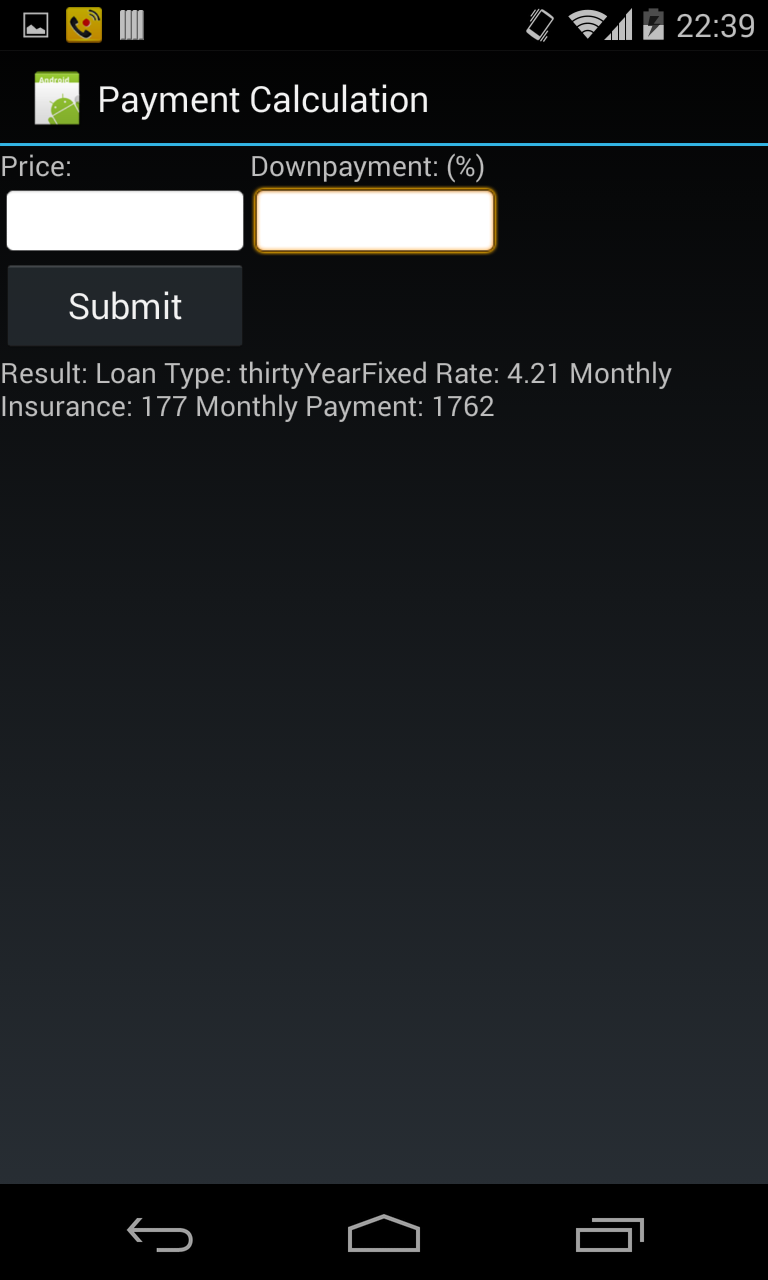
2.1. Uses the MVC design pattern (if using an HttpServlet) or uses JAX-RS/Jersey

In my web app project:

Model: PaymentModel.java

View: index.jsp

Controller: Project4TaskGAEServlet.java



2.2. Receives an HTTP request from the native Android application

Project4TaskGAEServlet.java receives the HTTP request with the argument "price" and “down”. It passes this search string on to the model.

2.3. Executes business logic appropriate to your application

PaymentModel.java does an HTTP request to:

“http:// www.zillow.com/webservice/GetMonthlyPayments.htm?zws-id=X1-ZWz1dsbizu9mob\_2vy2l&price=" + price + "&down=" + down + "&zip=98104"

It then parses the XML 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.

The response within Servlet formats the response to the mobile application in a simple XML format of my own design:

<?xml version="1.0" encoding="utf-8"?><MonthlyPayments:paymentsSummary xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:MonthlyPayments="http://www.zillow.com/static/xsd/MonthlyPayments.xsd" xsi:schemaLocation="http://www.zillow.com/static/xsd/MonthlyPayments.xsd http://www.zillowstatic.com/vstatic/LATEST/static/xsd/MonthlyPayments.xsd"><request><price>30000</price><down>15</down><zip>98104</zip></request><message><text>Request successfully processed</text><code>0</code></message><response><payment loanType="thirtyYearFixed"><rate>4.21</rate><monthlyPrincipalAndInterest>125</monthlyPrincipalAndInterest><monthlyMortgageInsurance>9</monthlyMortgageInsurance></payment><payment loanType="fifteenYearFixed"><rate>3.15</rate><monthlyPrincipalAndInterest>178</monthlyPrincipalAndInterest><monthlyMortgageInsurance>9</monthlyMortgageInsurance></payment><payment loanType="fiveOneARM"><rate>3.15</rate><monthlyPrincipalAndInterest>104</monthlyPrincipalAndInterest><monthlyMortgageInsurance>9</monthlyMortgageInsurance></payment><downPayment>4500</downPayment><monthlyPropertyTaxes>19</monthlyPropertyTaxes><monthlyHazardInsurance>48</monthlyHazardInsurance></response></MonthlyPayments:paymentsSummary>