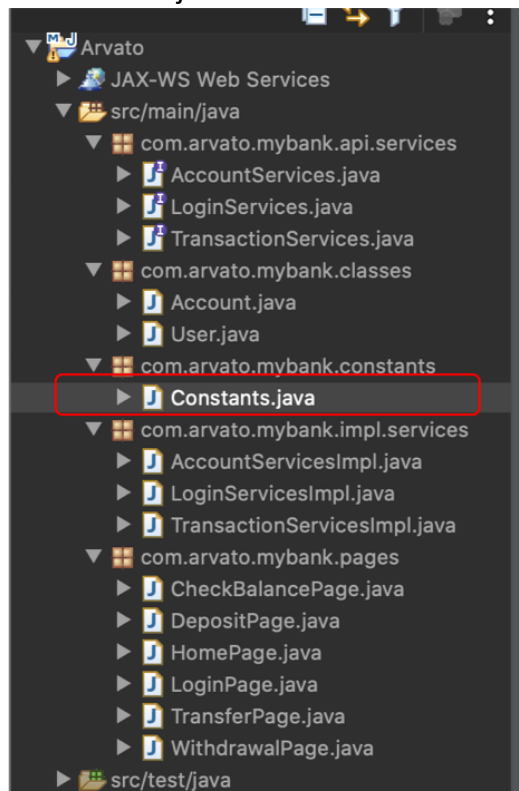


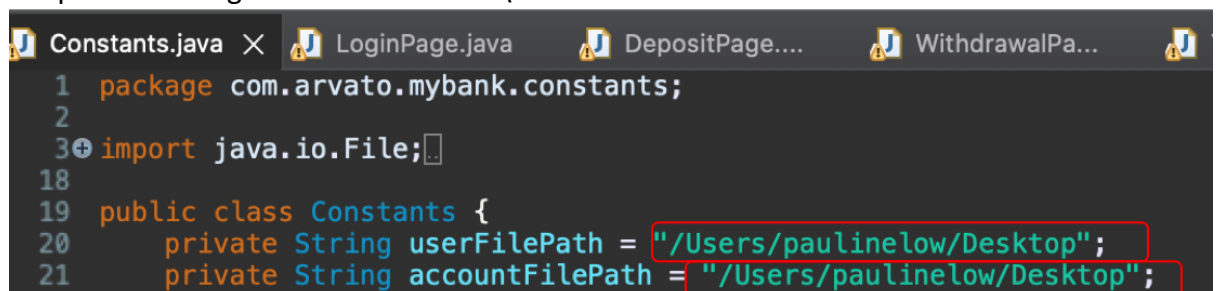
MyBank Application

Setup

1. Please retrieve the code source from this Github link:
<https://github.com/plpaulinelow/myBank>
2. Import this Javaproject as a dynamic web application into your IDE (preferably Eclipse).
3. The user.xlsx and account.xlsx are provided in the email. Please put them somewhere in your local computer and copy the file path location for these excel sheets.
4. Paste the file path location in the Constants.java file.
5. To locate Constants.java:



6. Once you are on Constants.java, kindly copy the location of the excel files in your local computer and replace them in the highlighted area shown below. Picture shown is the sample location for those using a Mac. For Windows, please locate the file path including the disk such as "C:\...."

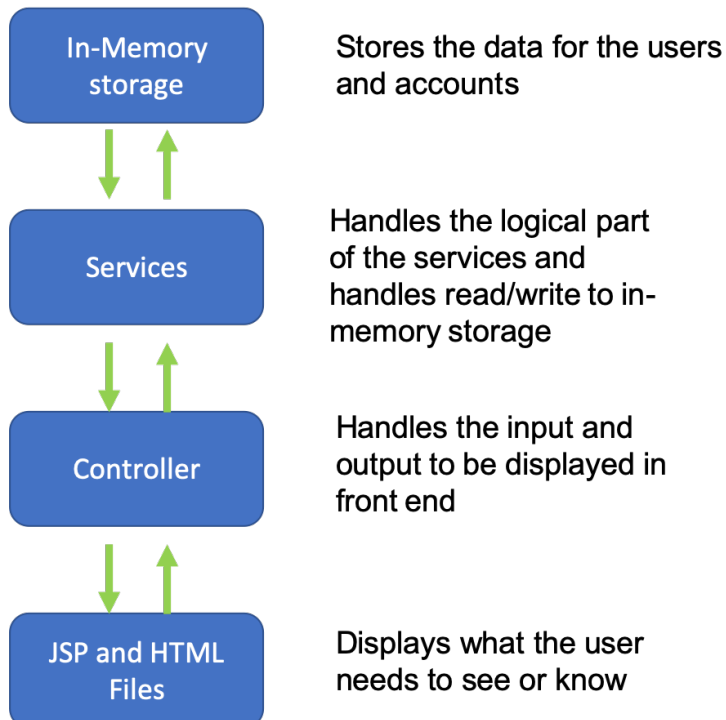


7. Save the changes made. Now we can proceed to build this project and run in on your server.

8. For development, Tomcat v9.0 Server is used to host this application locally via <http://localhost:8080/Arvato/>

Design

This application is a simple Java based Dynamic Web Application using the Servlet and Java Server Pages (JSP). There are several components built to make this app functional such as the services, classes, constants, pages and JSP pages.



1. The services allow developers to implement them across other platforms or customize the usage as required. There are already classes that implemented these services in MyBank which can be extended further as well for any required customization in the other application. The services include:
 - a. AccountServices
 - b. LoginServices
 - c. TransactionServices
2. The basic classes used to make this application includes:
 - a. Account
 - b. User
3. The constants allow the developers to read and write the values from the in-memory database which is stored in excel sheets. Reading and writing of data can only be achieved through the services. The excel sheets are:
 - a. user.xlsx
 - b. account.xlsx
4. The Pages are the controller for the JSP Pages in this application. The screen inputs and outputs are captured in the JSP Pages and processed further in the controller with the necessary logic. The controllers are:

- a. LoginPage.java
- b. HomePage.java
- c. CheckBalancePage.java
- d. DepositPage.java
- e. WithdrawalPage.java
- f. TransferPage.java

The JSP Pages include:

- g. LoginPage.jsp
- h. HomePage.jsp
- i. CheckBalancePage.jsp
- j. DepositPage.jsp
- k. WithdrawalPage.jsp
- l. TransferPage.jsp

- 5. There is also a system test included to test the functionality of the services which includes Deposit, Withdrawal and Transfer. These tests are included in ServicesTest.java.

Improvement (if given more time)

- 1. I would create a button to create new bank account in the Login page and user registration page.
- 2. I would try to include the excel sheet to be parked in the application folder, so that the users do not need to manual copy paste the file location of the excel sheets.
- 3. I would have added more security in storing the sessions and passwords.
- 4. I would have improved the UI layout to showcase the application.
- 5. I would work on the extra features mentioned in the specification.
- 6. I would handle the wrong inputs from the user more elegantly instead of always showing a popup only, example showing a red border around the input text box.