510 FP – Phase 2 Specs

Objective: To implement your phase 1 idea into an MVC format.

Goal: To use a MVC framework such as JAVAFX to accomplish your objective.

Instructions: Include packages for your phase 1 work to organize things.

Create packages and name them controllers, models and views which serves as your MVC format.

Move any files into your packages that make logical and organizational sense. Coding will be easier this way.

Example of an MVC format for a Bank.

**Models** (Bank domain) can be -Transactions, Customer, Users, Bank, Account

**Controllers** can be – LoginController, CustomerController, AdminController, CustomerController,

ViewAccountsController, AddBankController

**Views** can be – AddBankView, AdminView, CustomerView, LoginView, ViewAccountsView

As you can see, it is good to have a similar naming convention to “glue” files together into an MVC style.

Note you can rename your ‘existing’ classes, project, or packages in Eclipse anytime simply by right clicking on an item and choosing Refactor.

Code your program to include a database ( a minimum of 3 entities include user credential table) to hold pertinent data for the processing of data into information. Make use of functions to perform various CRUD operations.

Your classes should be organized into packages to perform the following deliverables:

Model classes should allow for queries to be implemented and for data storage.

View files (FXML) should serve the needs for user interfaces for basic IO needs.

Controller classes should help “glue” together model classes with View files.

Include the ability for your app to allow for multiple user roles. One role should be for a regular user and another will be for an admin user at minimum.

For credit, include the following snapshots:

1. Showing a successful login for a regular user.
2. Demonstrate functionality for the regular user. Ex. User purchases a product and can view their order.
3. Show a successful login for an admin.
4. Show admin functionality for at least an update of a record, and the deletion of a record.
5. Show a snapshot of your user table (i.e., the user login credentials)
6. Show a snapshot of at least 5 records added by some user in some table or columnar view.

Example: User creates 5 trouble tickets. Show result of tickets by some order.

Feel free to have any of your snapshots to be from popup dialog boxes or from labels, showing the needed results.

Completion of project for credit

1. Package your project files into a zip file. Include a jar file of your main application file.
2. Include a word doc file with the following:

An abstract description of your app.

Description of which file to execute at startup and what any login credentials are for regualr users and for the admin.

Snapshots of running app and db table.

1. A link or video file of your running app. A video can easily be made by downloading Jing at [techsmith.com](https://www.techsmith.com/jing-tool.html)

Teams must include again, two additional classes plus allow for the addition of a user role. Also a minimum of 4 database tables is necessary. Include a table view for all records added for each user role. Make a snapshot of your ERD’s (mandatory to include) as proof of all your tables.

Extra credit options (may include any or all of the following).

1. ERD diagram (+10)
2. Outstanding GUI (Charts, CSS style effects, menus, user interactions, functionality) (+15)
3. Use case diagram (+5)

\*Make sure to include snapshots for EC, labeled appropriately.

Use the following credentials for your DB connectivity

url =

"**jdbc:mysql://www.papademas.net:3306/fp510?autoReconnect=true&useSSL=false**";

username = "**fpuser**";

password = "**510**";

Grading rubric:

|  |  |
| --- | --- |
| Criteria | Points |
| Database Creativity/Deployment: | **50 points** |
| Programming Creativity, UI & Code Logic: | **50 points** |
| Documentation/error handling/comment/program desc./output: | **50 points** |
| Extra Credit: | **25 points** (max) |

Site references:

[Login](https://docs.oracle.com/javafx/2/get_started/form.htm)

[Login with FXML & Controller](https://docs.oracle.com/javafx/2/get_started/fxml_tutorial.htm#CHDICHAE)

[Swing & FX](https://docs.oracle.com/javafx/2/swing/swing-fx-interoperability.htm#CHDIEEJE)

[FX bar chart](https://stackoverflow.com/questions/18896908/open-new-barchart-window-javafx-from-inside-a-swing-gui)