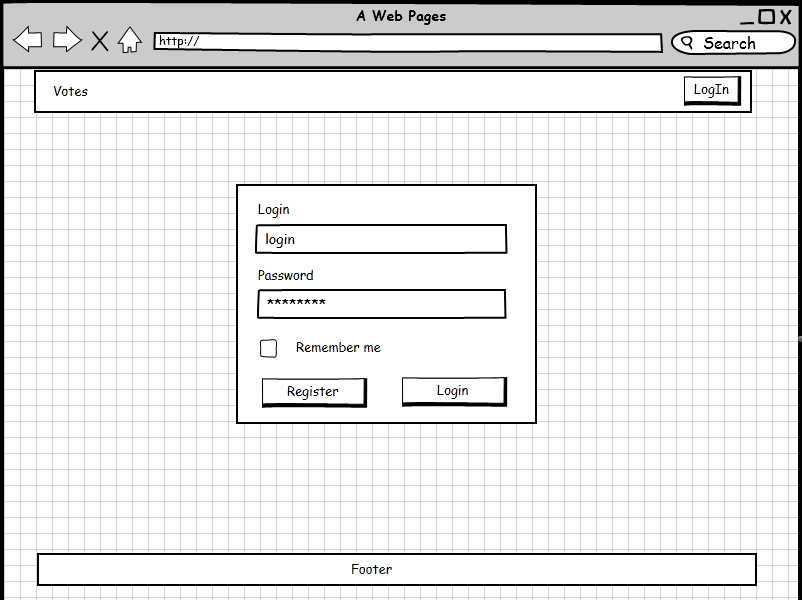
**CLC Project Assignment – Milestone 2**

**Register, Login and Display Products**

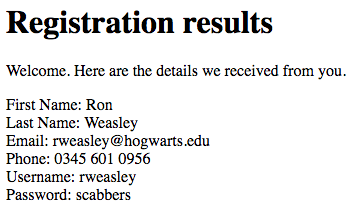
Requirements:

1. Review requirements in Project Overview.
2. Main Application Module:
   * This will be the main application page that is visible when the root URI of the application is accessed.
   * The main application should provide initial navigation concepts, such as a menu bar, to access the applications core functionality such as Login and Register.
   * The main application at this point should have well defined styles, fonts, colors, and overall application theme implemented that will be used for the remaining milestone deliverables. Include a wireframe drawing of each page of your app.

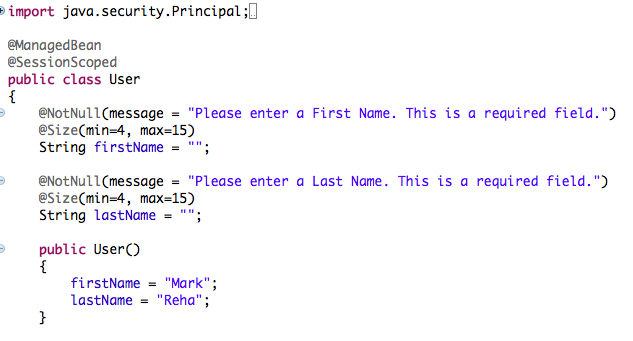


* + The application should leverage Facelets to support reusable page components (i.e. headers, footers, menu bars, etc.) and for Page Layouts.
  + The application should have a Title (and possible a Logo).

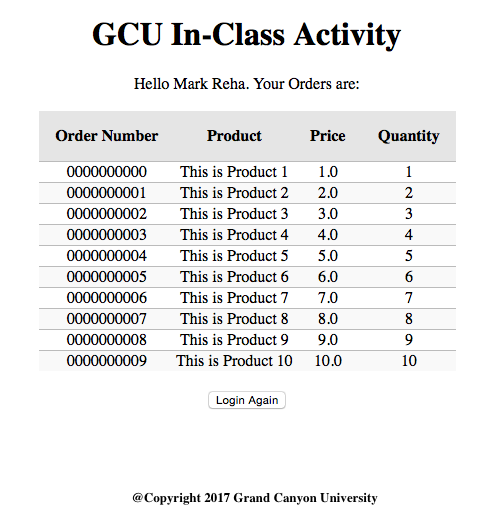
1. Registration Module:
   * A user should be able to register to access the application.
   * Registration details at a minimum should include First Name, Last Name, Email Address, Phone Number, and Login Credentials (Username and Password).
   * Initial persistence will be values simply echoed back to the user. The data will not be saved to the database until milestone #4.



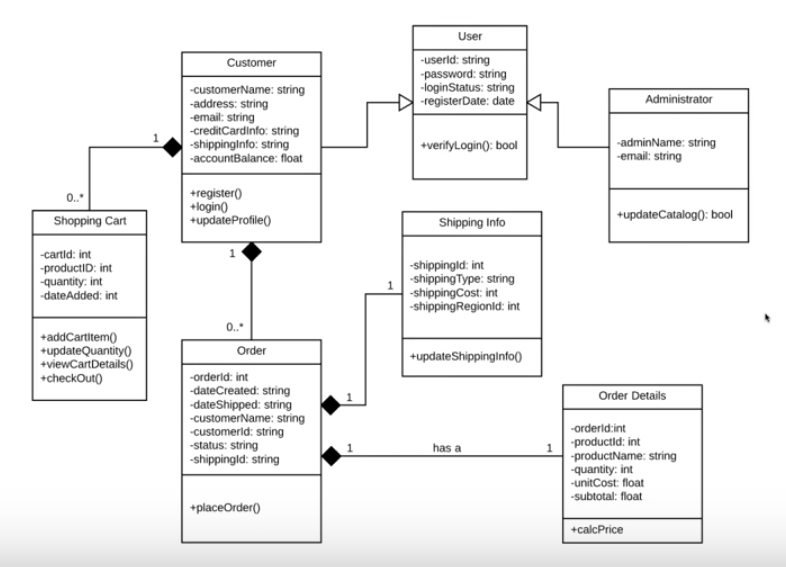
* + The actual (and static) login data will be stored in a class as shown in the in-class examples. Later, the registration will be refactored to use a database. The picture below shows that the registration data for “Mark Reha” has been statically saved into the User class.



* + The Registration and Login pages should leverage common Facelets. <h:submitButton> etc. The pages should utilize page template design. The following diagram shows the use of headers, main content and footers in a template design used in one of our examples.



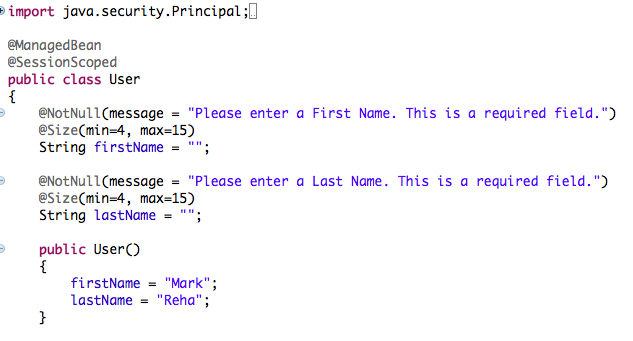
* + A User Object Model should be designed and implemented. UML diagrams are used to define the methods and properties of classes used in object oriented programming. Here is an example of a UML and a link to a tutorial for making a UML



Watch the following video for a refresher on UML design.

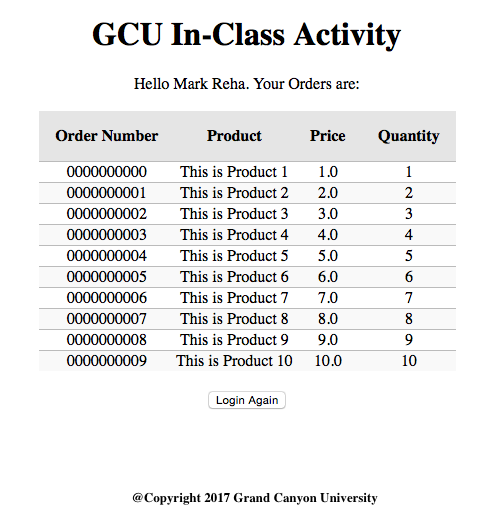
[**https://www.youtube.com/watch?v=UI6lqHOVHic**](https://www.youtube.com/watch?v=UI6lqHOVHic)

* + All data entered by the user must be validated and provide proper error messages enabling the user to easily correct the data.



* + It should be noted that this module will be refactored a number of times during the milestone releases. The JDBC API will be replaced by JPA in Milestone 4 and the Login Credentials fields will be removed in Milestone 7.

1. Login Module v1:
   * Initial Login Module will use the data saved in the Users class to authenticate the user. Later, the app will use JDBC API to authenticate the user with a relational database. Security will re-implemented as part of Milestone 6. It is not required to securely protect any pages in the application.
   * The Login pages should leverage common Facelets.
   * Once the user is logged in the main “product” page should be displayed and the menu system be dynamically updated to reflect the logged in state of the application.



1. Updated Design Report.
2. It is expected that the Team will perform peer code reviews on all code.
3. It is expected that the Team will fully document all code modules, classes, methods, and use inline comments for all code.
4. It is expected that the Team will meet with the instructor if there are project management issues.

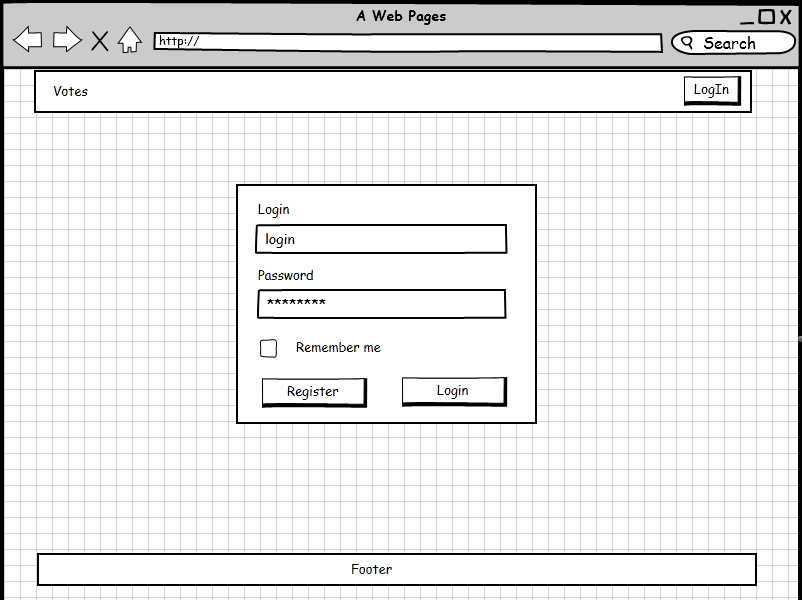
Deliverables and Submission:

1. Updated Design Report with the following sections completed:
   * Cover page with list of tasks completed.
   * Planning documentation: task list/schedule or Scrum artifacts.
   * Design documentation:
     + General Technical Approach
     + Key Technical Decisions
     + Install or Configuration Instructions
     + Known Issues
     + Risks
     + Sitemap Diagram
     + User Interface Diagrams
     + ER Diagram
     + DDL Scripts
     + Class Diagrams (for all Object Models)
2. Upload and manage all code in Bitbucket.
3. Zip up code.
4. Upload all code and design to LoudCloud.

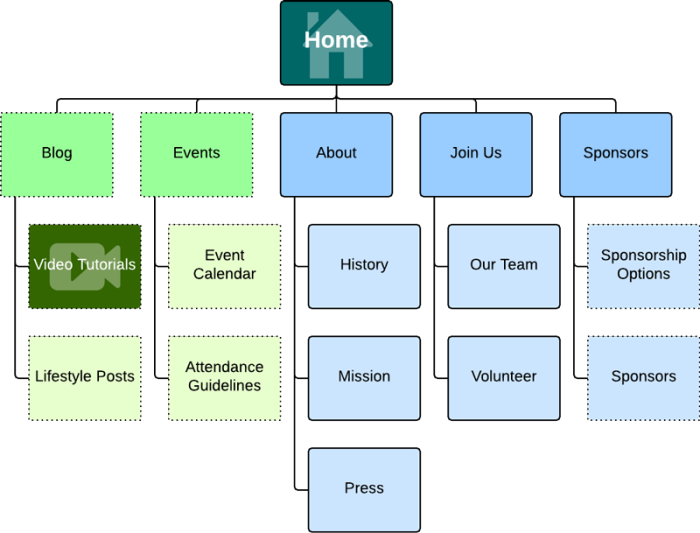
This assignment uses a rubric. Please refer to the rubric prior to beginning the assignment to become familiar with the expectations for successful completion.

# Examples of various design documents:

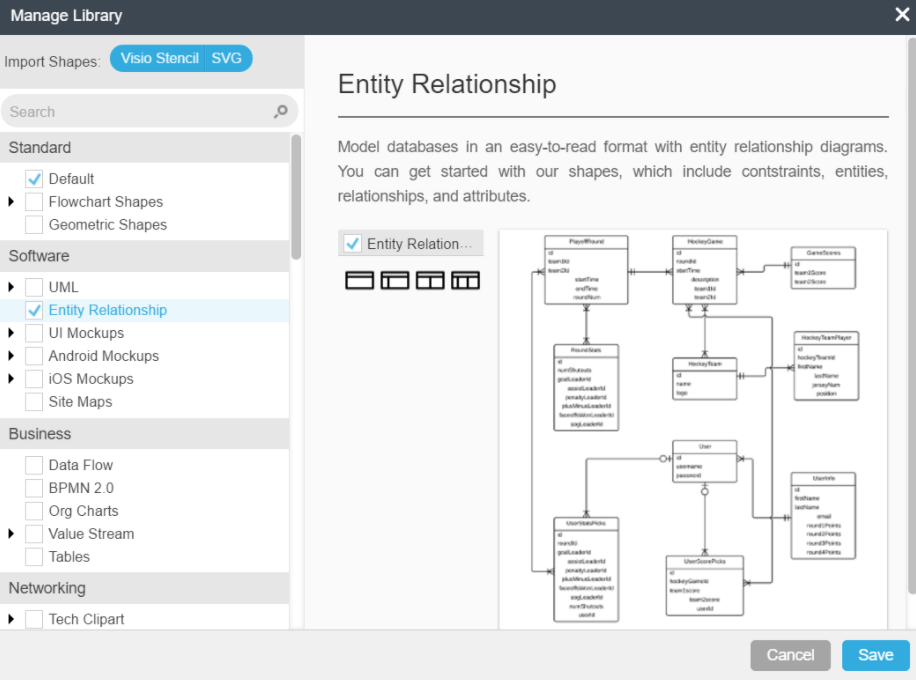
1. **Wireframe** design for the **User Interface Diagram** for the login page. This was designed using a tool from mockupbuilder.com <http://mockupbuilder.com/Gallery/Index/982>



1. **Site Map**. Here is an example of a site map built with a tool called LucidChart. <https://www.lucidchart.com/blog/how-to-make-a-sitemap-using-lucidchart>



1. ER Diagrams display the design of your data.

****

**View the following tutorials for a refresher on ER Diagrams**

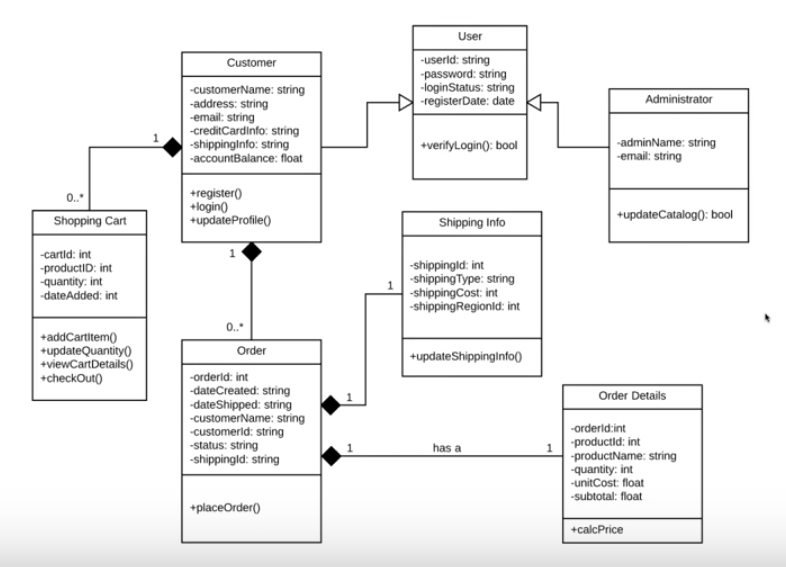
<https://www.youtube.com/watch?v=QpdhBUYk7Kk>

and

<https://www.youtube.com/watch?v=-CuY5ADwn24&t=250s>

<https://www.lucidchart.com/pages/er-diagrams>

1. UML diagrams are used to define the methods and properties of classes used in object oriented programming.



Watch the following video for a refresher on UML design.

[**https://www.youtube.com/watch?v=UI6lqHOVHic**](https://www.youtube.com/watch?v=UI6lqHOVHic)