**PHP II Application Programming**

**Project Status and Design Report**

|  |  |  |
| --- | --- | --- |
| **Topic:** | *Product Catalog* | |
| **Date:** | *Thu Sept 21, 2017* | |
| **Revision:** | *1.0* | |
| **Team:** | 1. *Zach Kujath* | |
| 1. Nathaniel Lee | |
|  | |
|  | |
|  | | |
| **Weekly Team Status Summary:** | |  |  |  |  | | --- | --- | --- | --- | | **User Story** | **Team**  **Member** | **Hours**  **Worked** | **Hours Remaining** | | *Updated classes* | *Nathaniel* |  |  | | *Updated html pages* | *Zach* |  |  | | *Updated database* | *Nathaniel* |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | | |
| **GIT URL:** | *https://github.com/nlee1131/cst236/tree/master/CLC\_JobFinder* | |
| **Hosting URL:** | *http://nlee.b.hostable.com/CLC\_JobFinder/index.php* | |
| **Peer Review:** | *Y* | We acknowledge that our team has reviewed this Design and we agree to the approach we are all taking. |

**Supporting Design Documentation**

**Install Instructions:**

*Step by step instructions for setting up your database, configuring, and deploying/installing your application. This section should also include detailed instructions for what configuration files are required by your application, what configuration settings need to be adjusted for various runtime (development or production) environments, and where the files need to be deployed to. This section should also contain detailed instructions for how to clone your application source code from BitBucket and deploy the application to an externally hosted site.*

*Go into phpMyAdmin, use the DDL scripts to create the database.*

*Go into hostable, go to cPanel, go to file manager and open up Public\_html, then upload files from GitHub, follow URL path to website.*

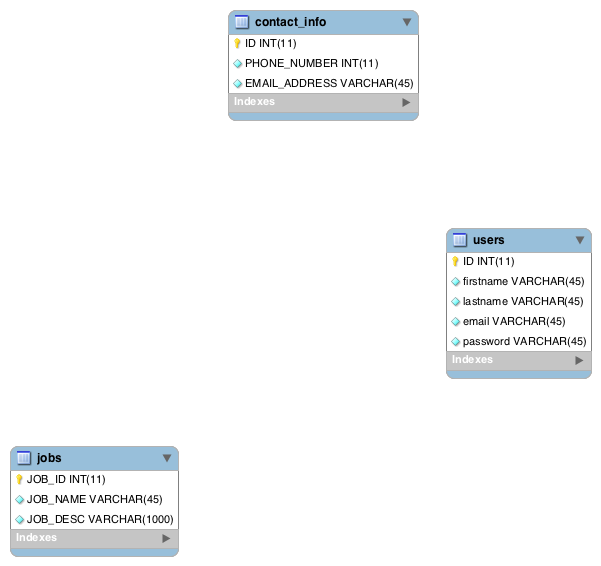
**Key Technical Design Decisions:**

*Registration requires first and last name, username and password and email. Admin key is optional, allows for account with higher privileges than the average user.*

*There will be a job class that has generic set of fields, that can be inherited by sub classes.*

**ER Diagram:**

*Image file of your ER database diagram.*

**

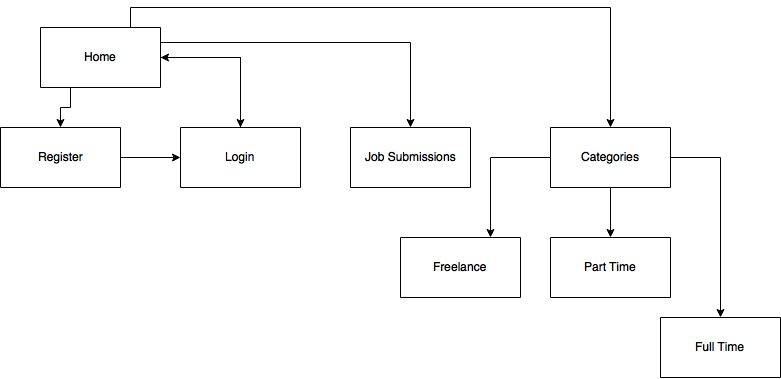
**DDL Scripts:**

*This should contain a link to Bitbucket where the DDL script can be downloaded from.*

*https://github.com/nlee1131/cst236/blob/master/usersTableDDL.txt*

**Sitemap Diagram:**

*Image file of your Sitemap diagram.*

**

**Security Design:**

*This section should outline the design for how authentication and authorization was supported. This section should also contain all of the roles and privileges that are supported by the design.*

*Password must be a minimum of 8 characters, and no more than 50.*

*Users will be allowed 5 login attempts, using their email & password.*

**Product Service Interface Design:**

*This section should fully document your Product Service Interface API, how to access the service, what parameters are required by the API, and the detailed JSON data format specification that could be used by a 3rd party developer to integrate with the service and API.*

**Card Transaction Design:**

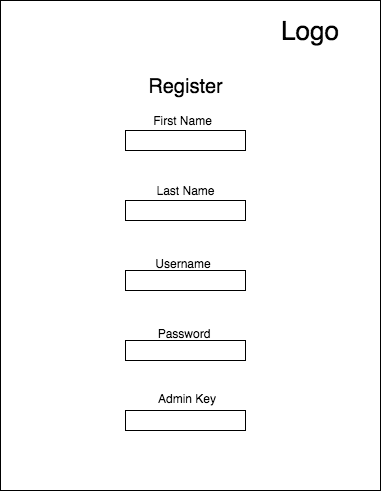
*This section should fully document your Card Transaction Service design (i.e. UML drawing), what methods and capabilities are supported, and how this was integrated into your eCommerce shopping card design.*

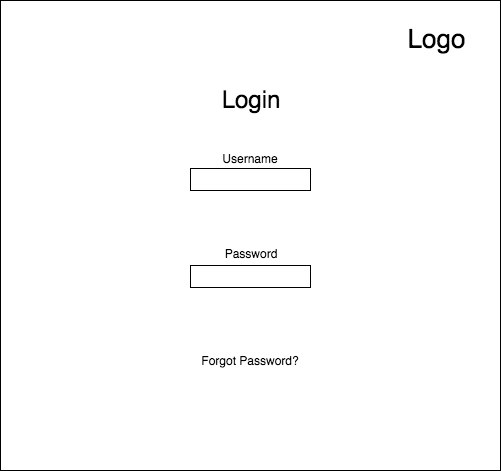
**Flow Charts:**

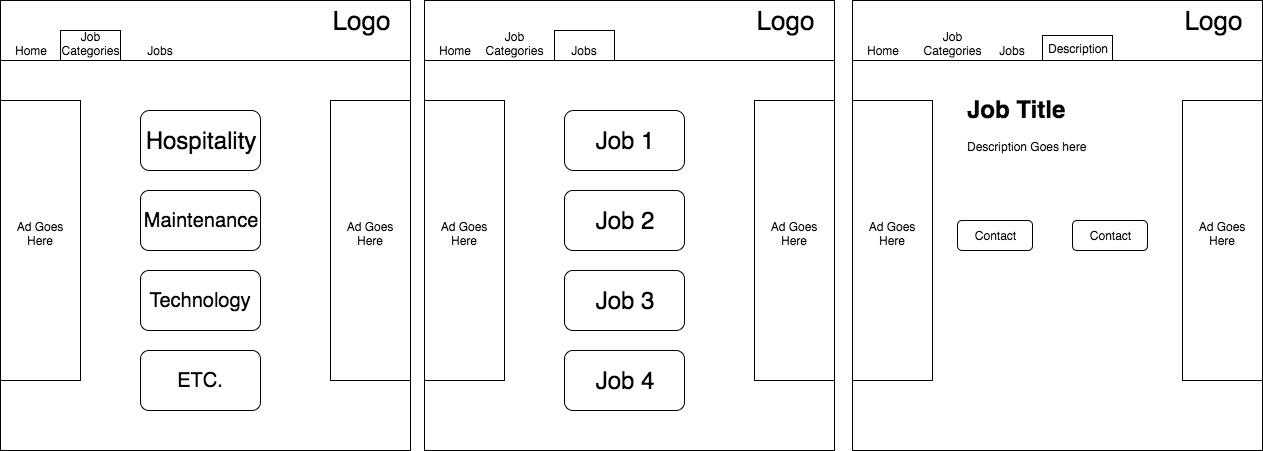
*You should insert any flow charts here. Flow charts should document algorithms or workflow (i.e. shopping logic, etc.) that will be implemented in your program. At a minimum this should contain a flow chart of your shopping experience.*

**User Interface Diagrams:**

*You should insert any wireframe drawings or white board concepts that were developed to support your application. If you have no supporting documentation please explain the rational why you are able to leave this section as N/A.*

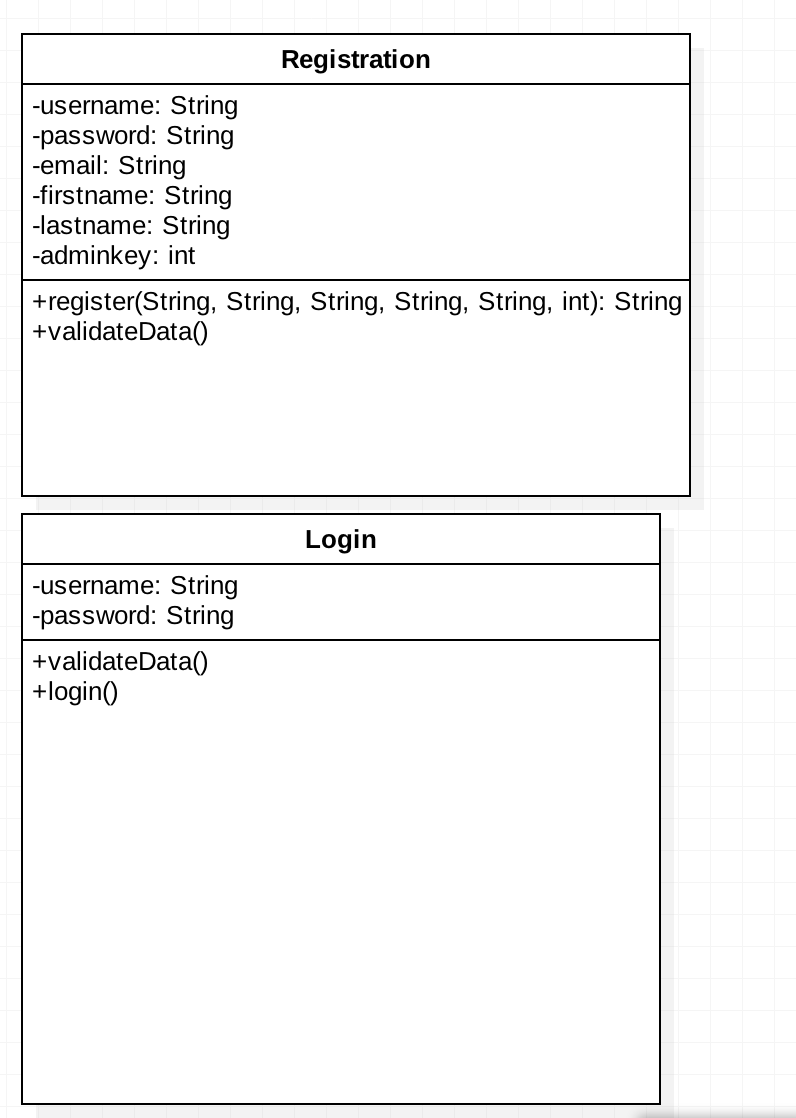






**Class Diagrams:**

*You should insert any class diagrams here. Your class diagrams should be drawn correctly with the 3 appropriate class compartments, + and – minus to indicate accessibility, and also the data types for the state/properties as well as method arguments and return types. If you have no supporting documentation please explain the rational why you are able to leave this section as N/A.*



**Pseudo Code:**

*You should provide Bitbucket URL references to any code stubs & pseudo code. If you have no supporting documentation please explain the rational why you are able to leave this section as N/A.*

**Other Documentation:**

*You should insert any additional drawings, storyboards, white board pictures, project schedules, tasks lists, etc. that support your approach, design, and project. If you have no supporting documentation please explain the rational why you are able to leave this section as N/A.*

*Trouble connecting the jobListPage to the database class to direct to the jobDescription.*

*RISK: Unknown how to make an admin user.*