**CST-256 Design Report Template**

**Video Link:** <https://www.loom.com/share/1514a7523e45474c8e304e2f7806ab0c>

|  |  |  |
| --- | --- | --- |
| **Topic:** | Topic 5: Integration Layer | |
| **Date:** | 3.14.21 | |
| **Revision:** | 5.0 | |
| **Team:** | 1. Thomas Biegel | |
| 1. Brian Basinger | |
|  | |
|  | |
| **Weekly Team Status Summary:** | |  |  |  |  | | --- | --- | --- | --- | | **User Story** | **Team**  **Member** | **Hours**  **Worked** | **Hours Remaining** | | Create design template | Thomas | 2 | 0 | | Update diagrams/tables/other documentation | Thomas + Brian | 1 | 0 | | Upgrade from list search results to table | Thomas | 1 | 0 | | Application module | Brian | 5 | 0 | | Upgraded Security | Brian | 3.5 | 0 | | Host To Azure | Thomas | 1.5 | 0 | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | | |
| **GIT URL:** | https://github.com/TBYT/MilestonePHP3 | |
| **Hosting URL:** | <https://teambladeclc.azurewebsites.net> | |
| **Peer Review:** | Y | We acknowledge that our team has reviewed this report and we agree to the approach we are all taking. |

**Design Documentation**

**Install Instructions:**

Install project from github. Run DDL scripts in mySQL database. Update .env file, DataAccess.php, and the BusinessService.php info for database connectivity if necessary. Update builder file and targets if needed. Run MAMP. Open localhost in web browser.

**General Technical Approach:**

This project uses the model view controller methodology. The model includes all the services, logic, and classes on the back end of the project. This involves database access and business service classes. The views are the pages that the clients and users get to see. They are contained in the views folder, and involve html, php, and blade content. The controller manages website navigation and involves the routes and controller classes. The routes control URL navigation and can point to the controller, which can access the business and data access layer, as well as perform application logic.

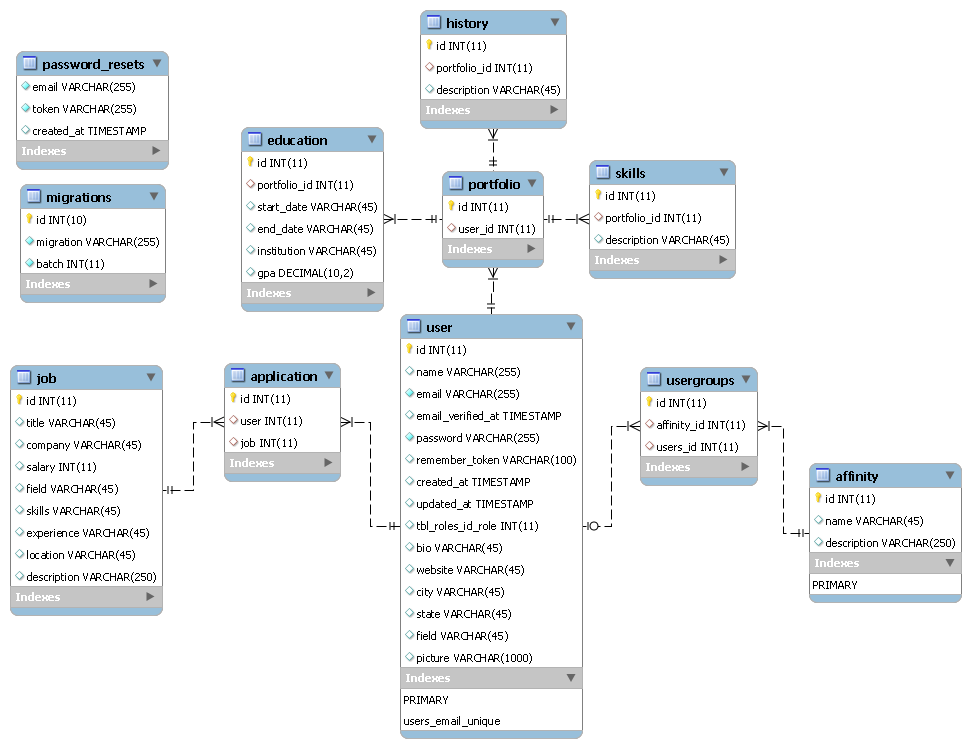
**Key Technical Design Decisions:**

We use the Laravel Framework in this project because it is the default framework used for this class. The Laravel framework controls many of the default functions such as site navigation with the routes folder and provides the blade framework to control much of the site logic in the html pages. The framework can be built on with controller classes and database and business rules access classes.

**Risks:**

The business rules for entering data may not be well defined, and at some points may be overly restrictive or not prevent invalid data.

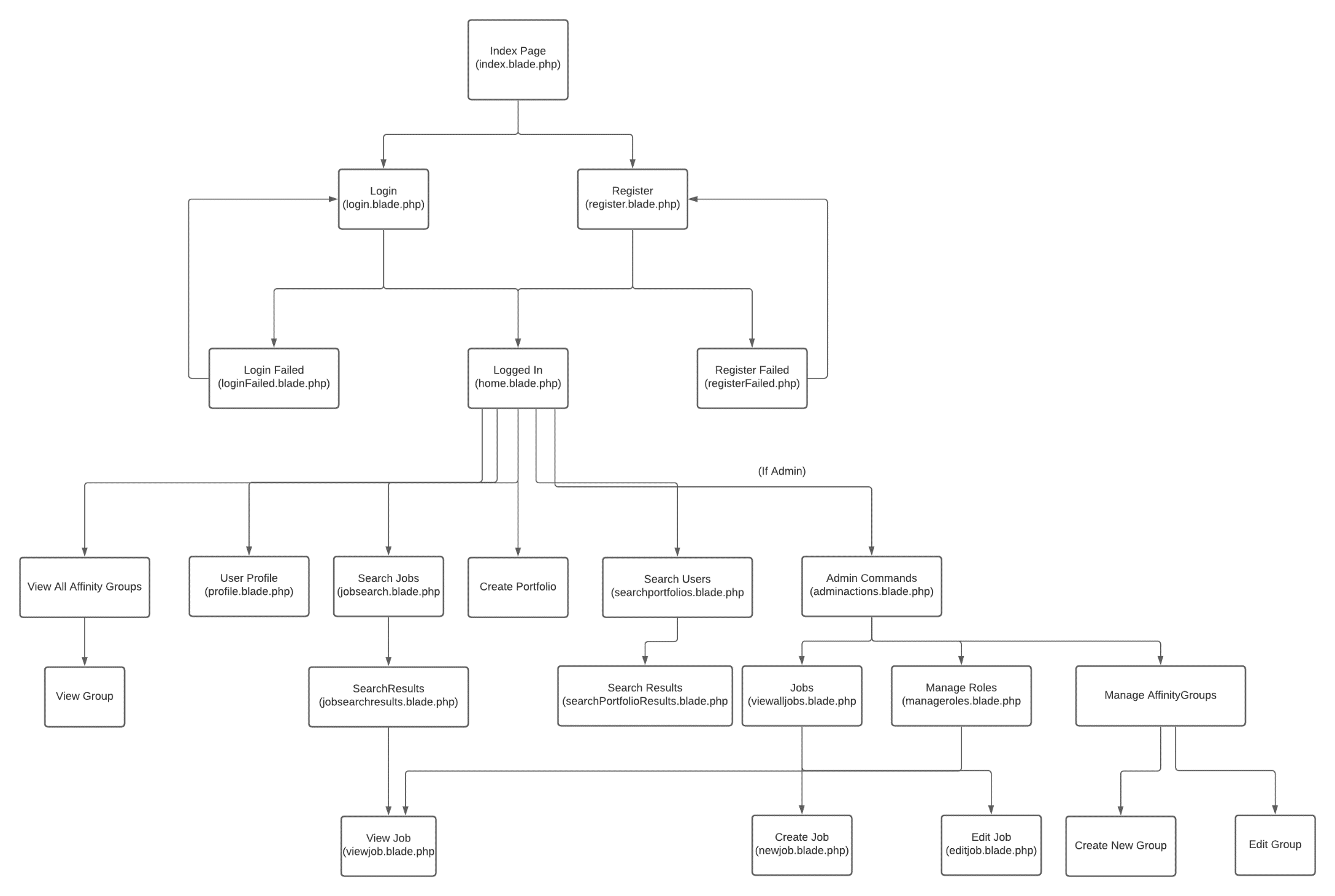
**ER Diagram:**



**DDL Scripts:**

https://github.com/TBYT/MilestonePHP3/blob/master/ddlscript.sql

**Sitemap Diagram:**



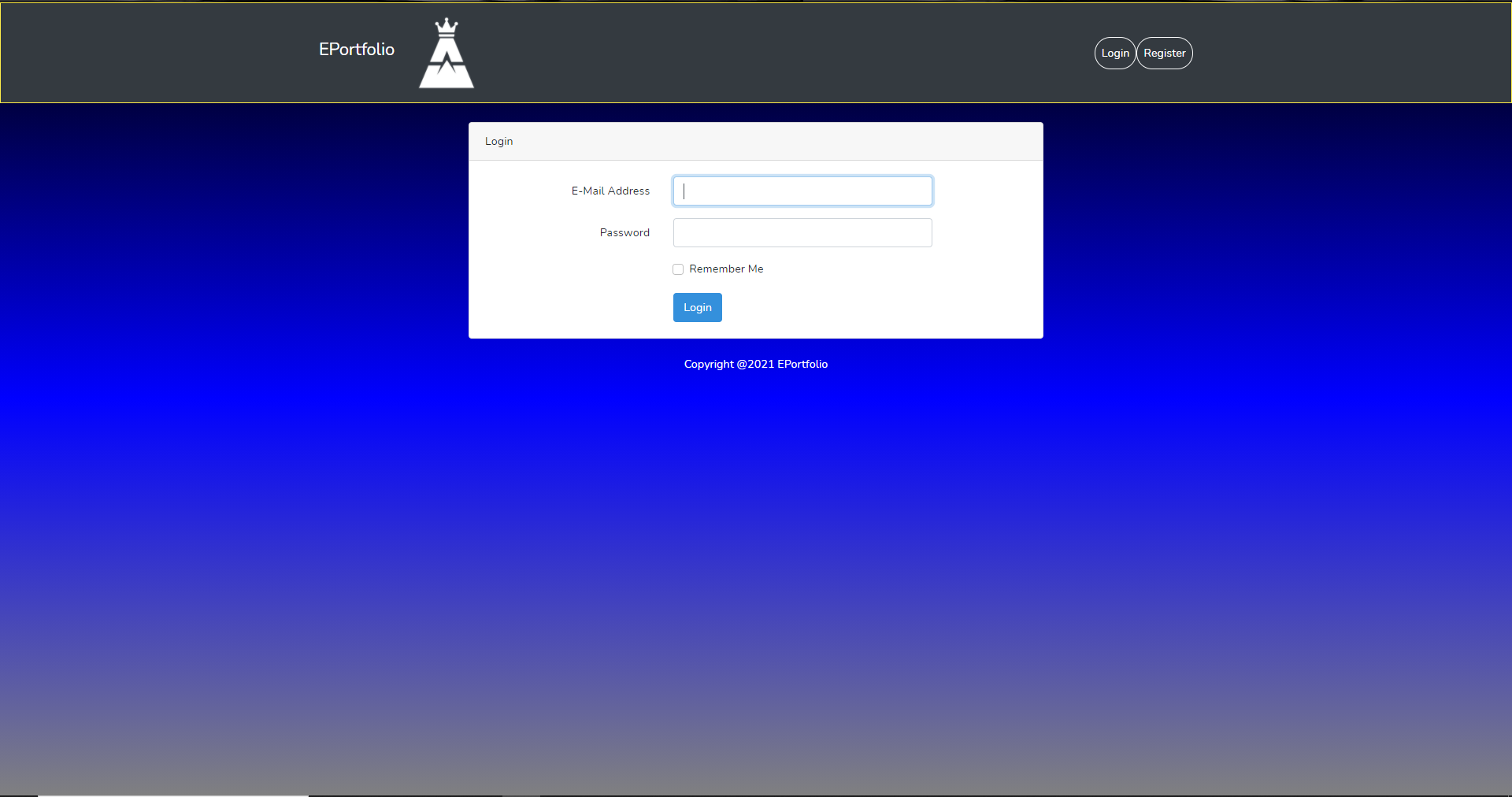
**Security Design:**

The two roles are users and admin. Users that are not logged in can only access the contact, login and register pages. A Security Check Middleware is implemented to keep users from accessing pages without the proper authorization. Encryption is utilized to keep passwords secure.

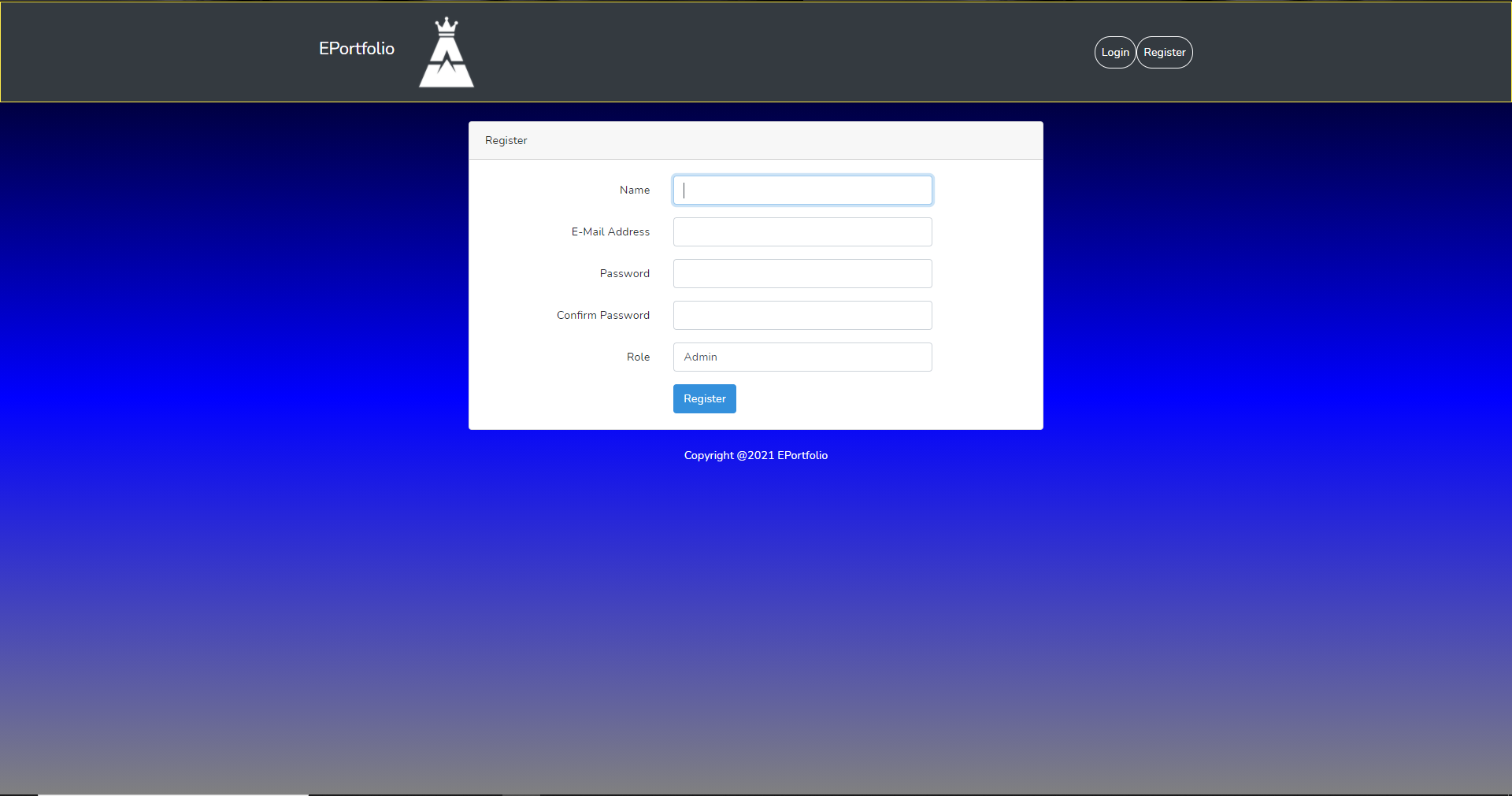
**User Interface Diagrams:**



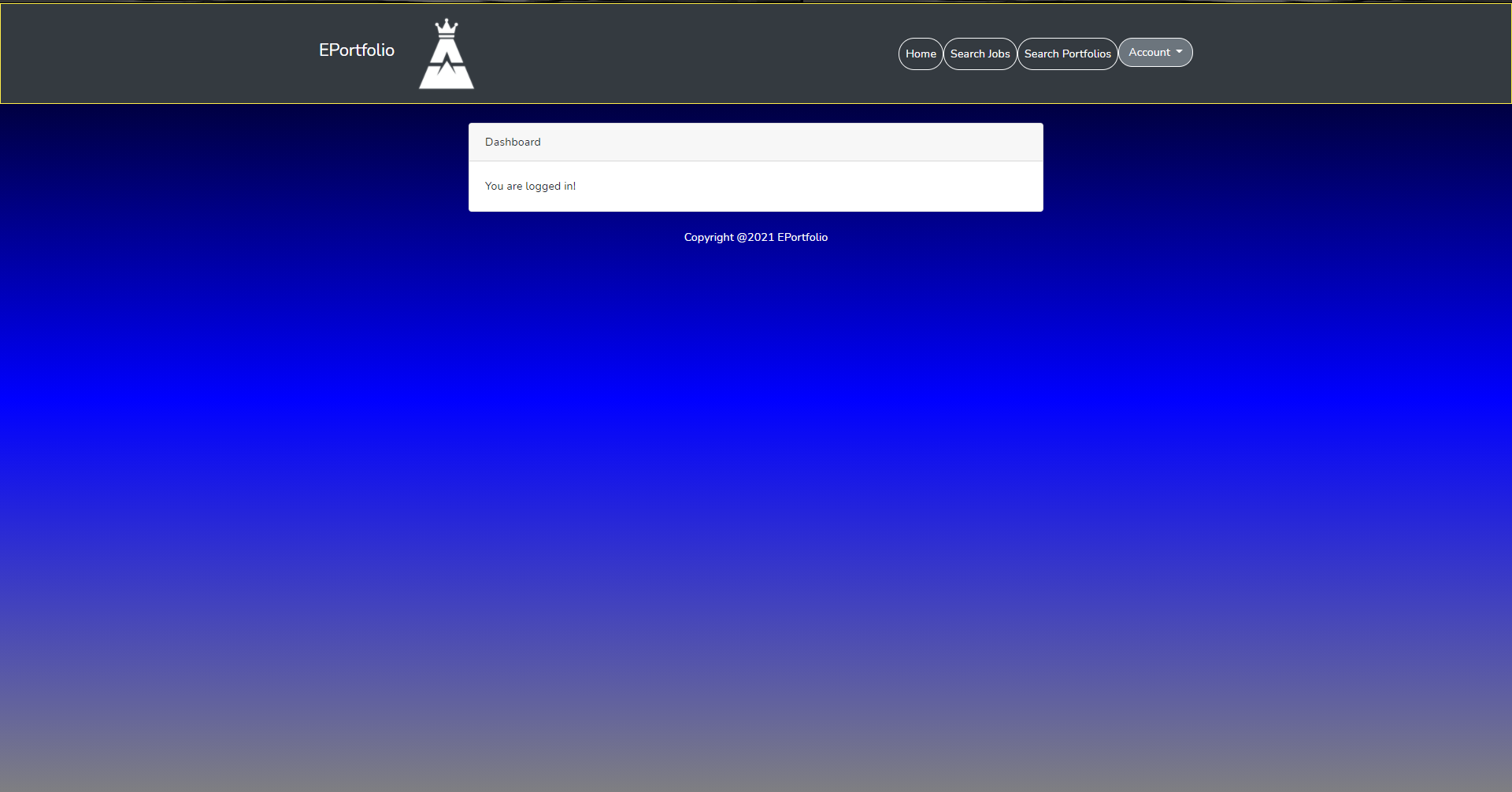
Login:



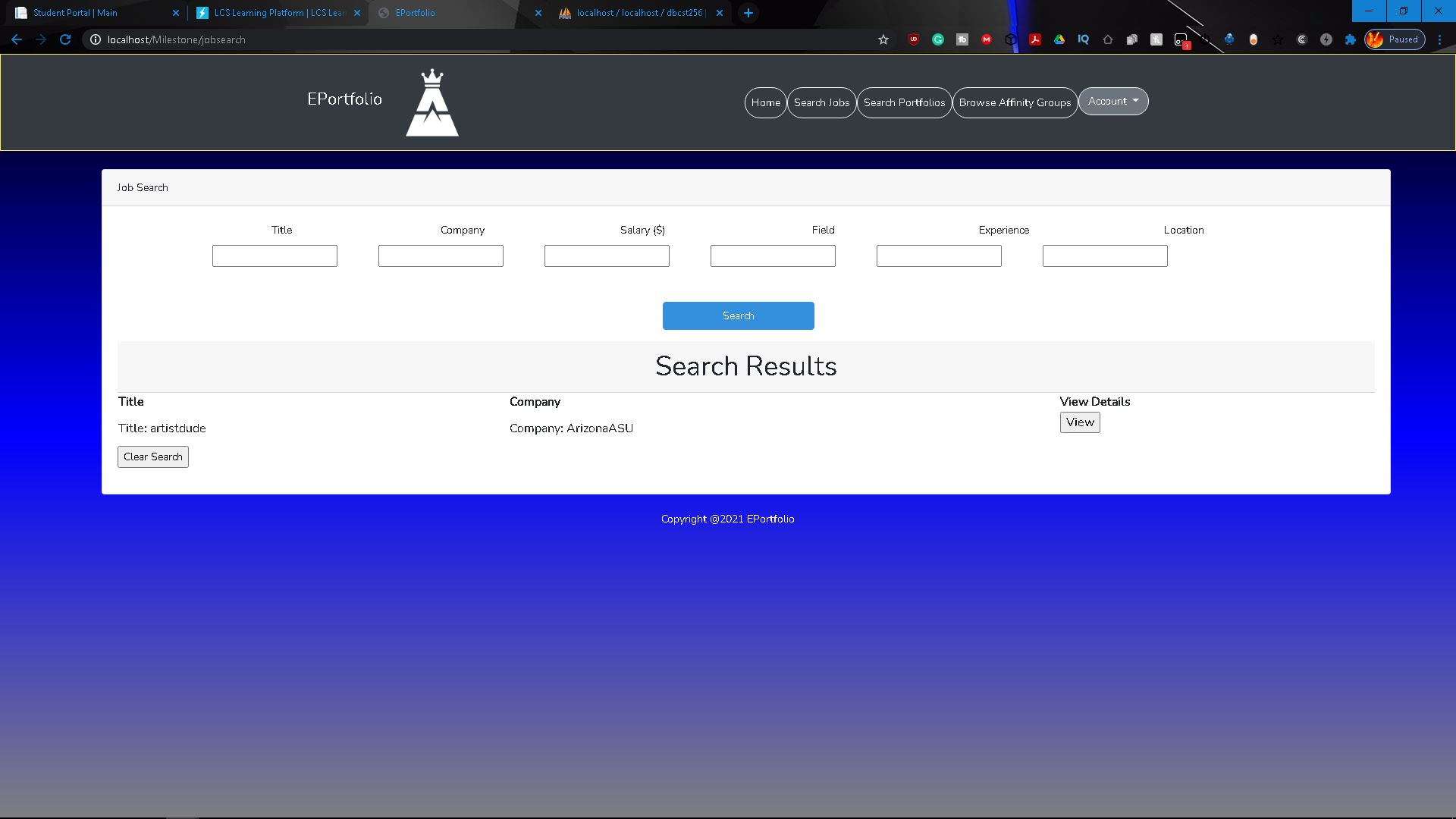
Register:



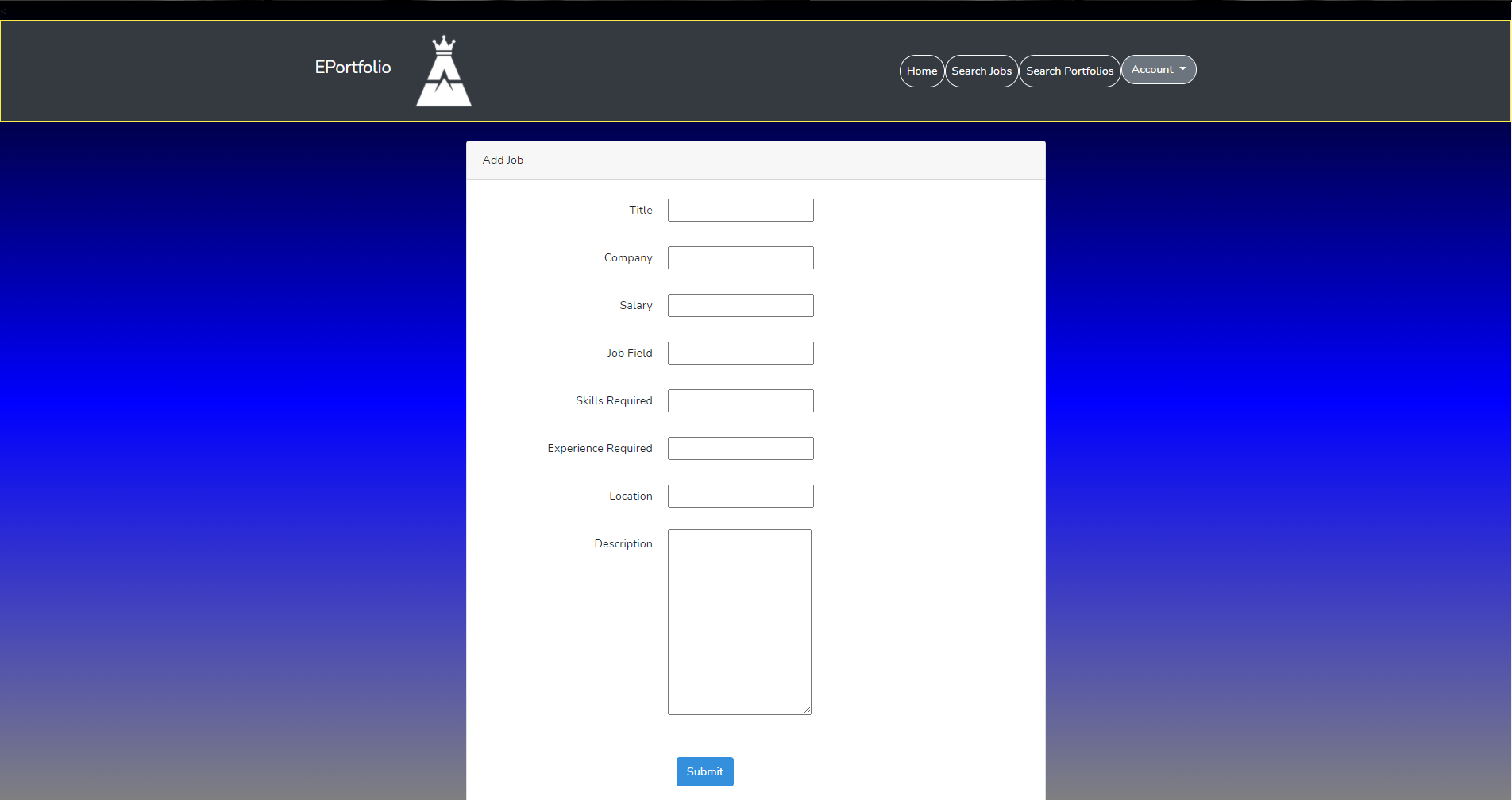
Home Page:



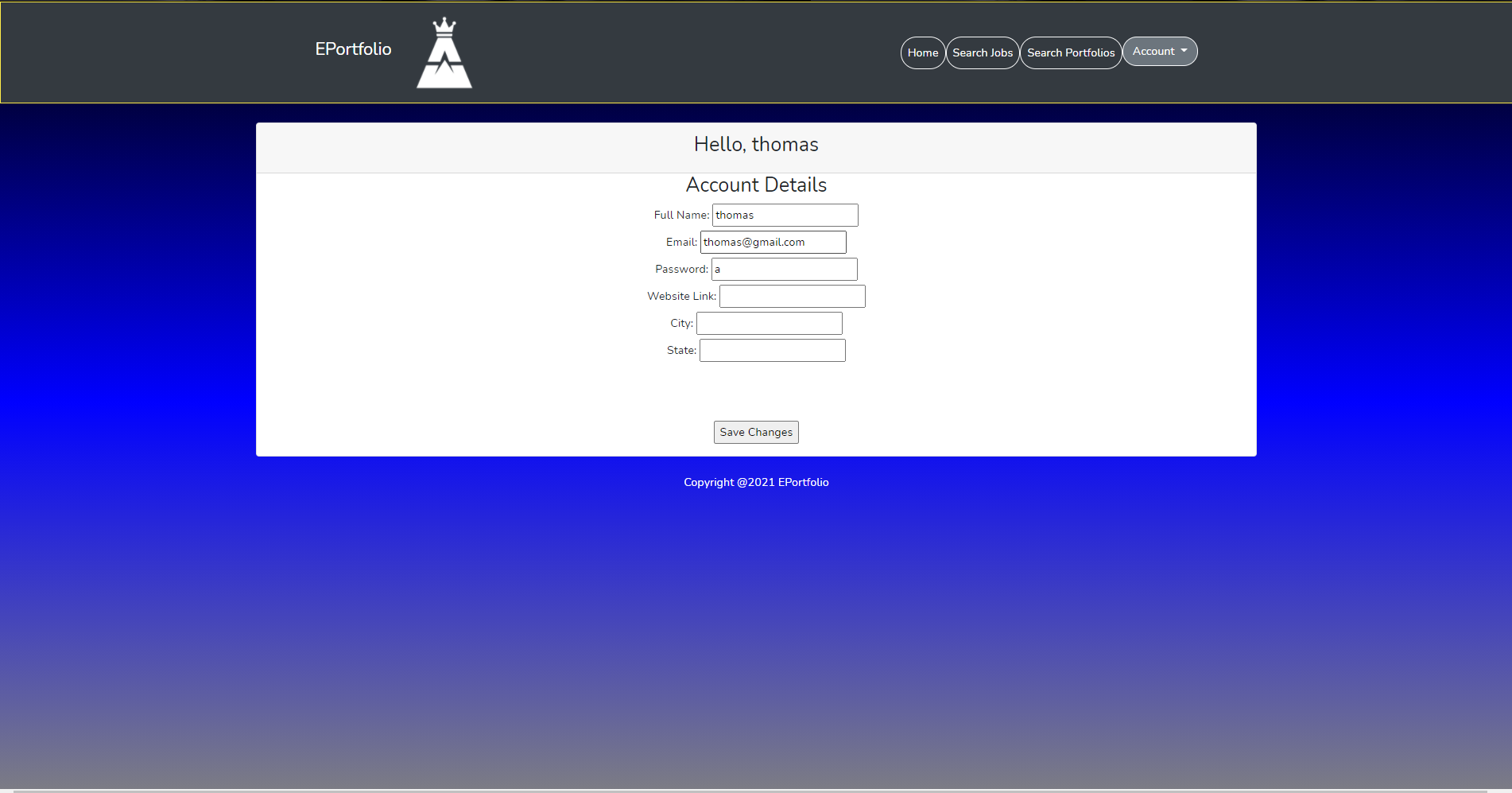
Job Search and search results:



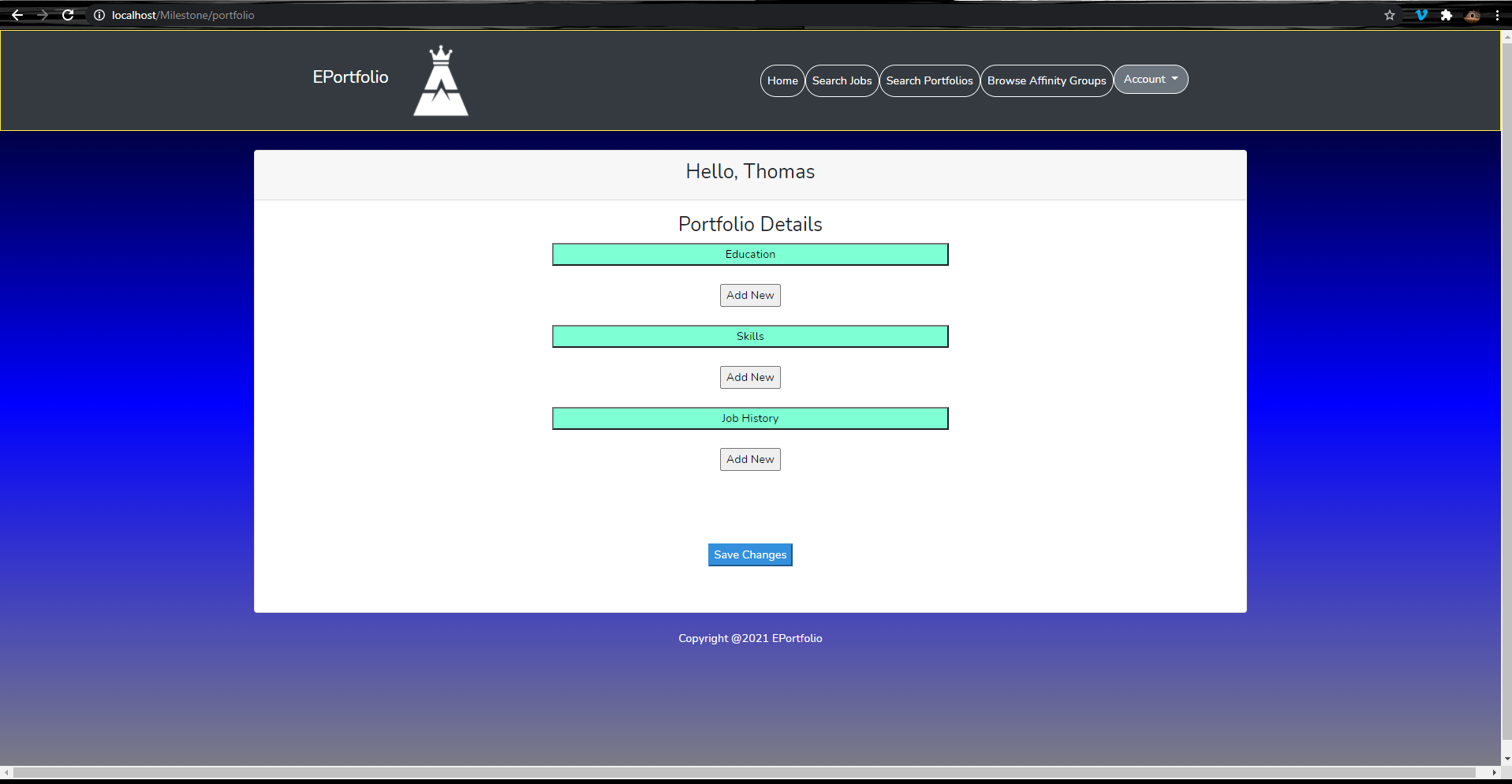
Add Job:



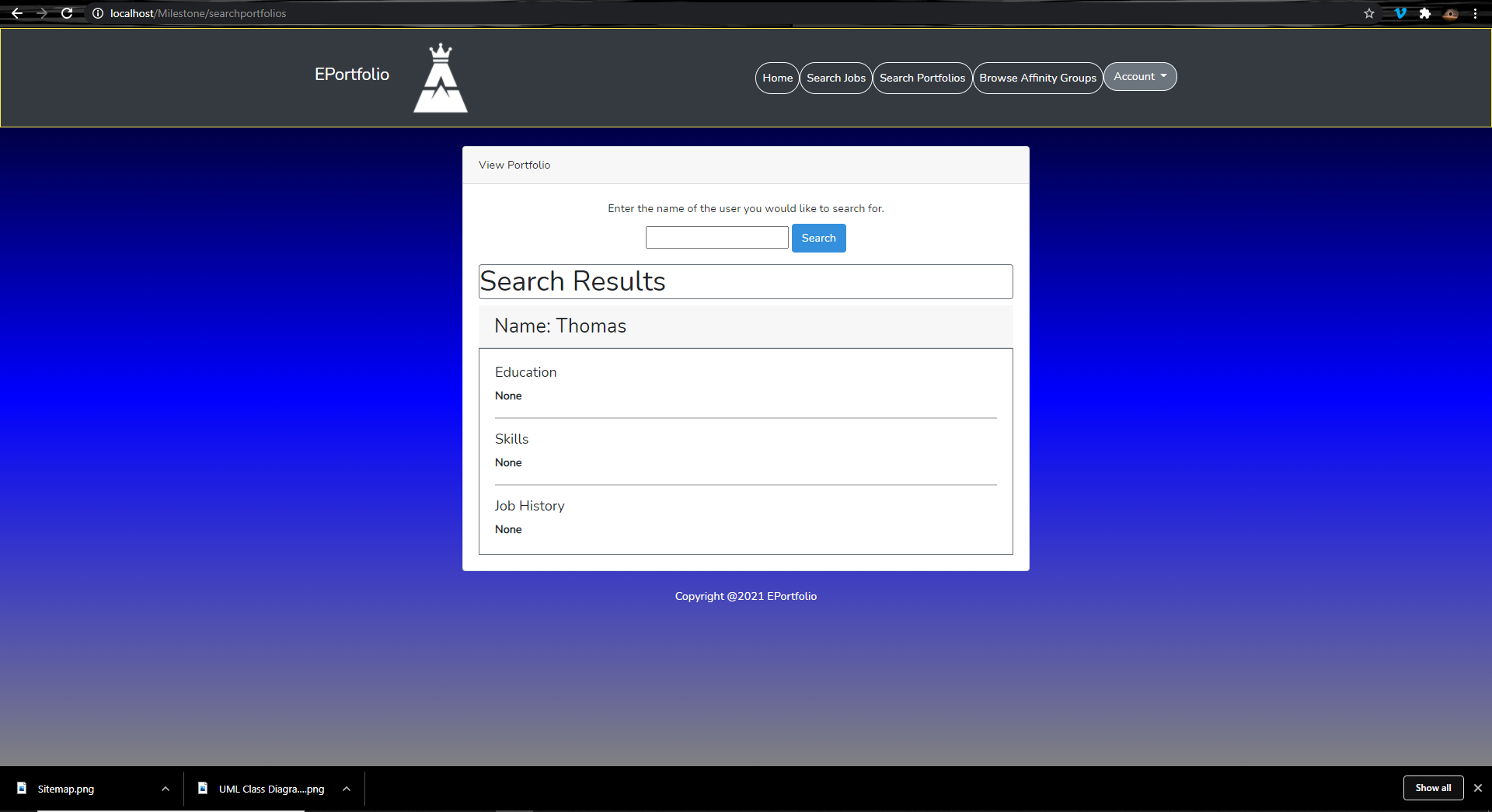
Account Page:



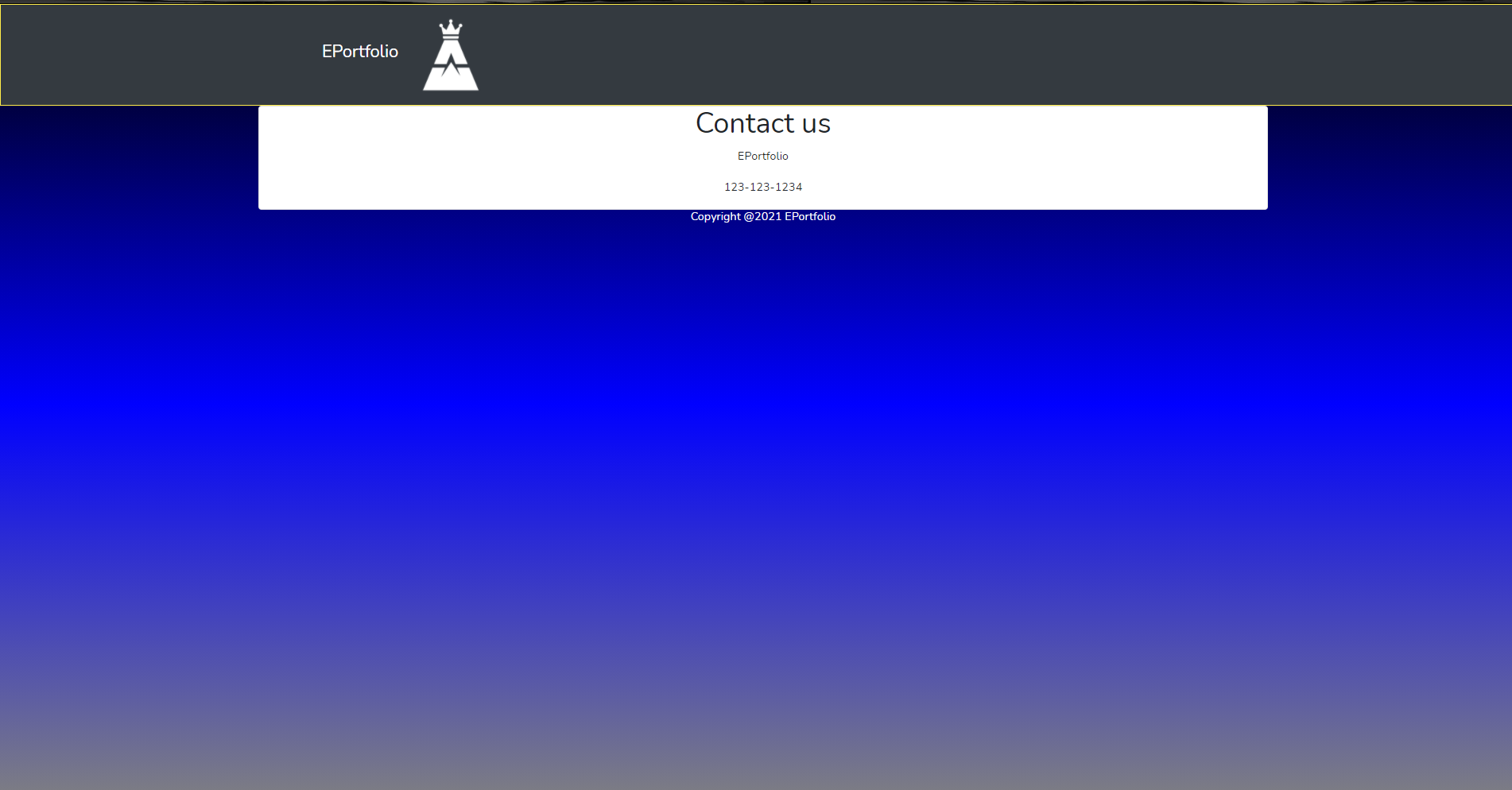
Create/Edit Portfolio:



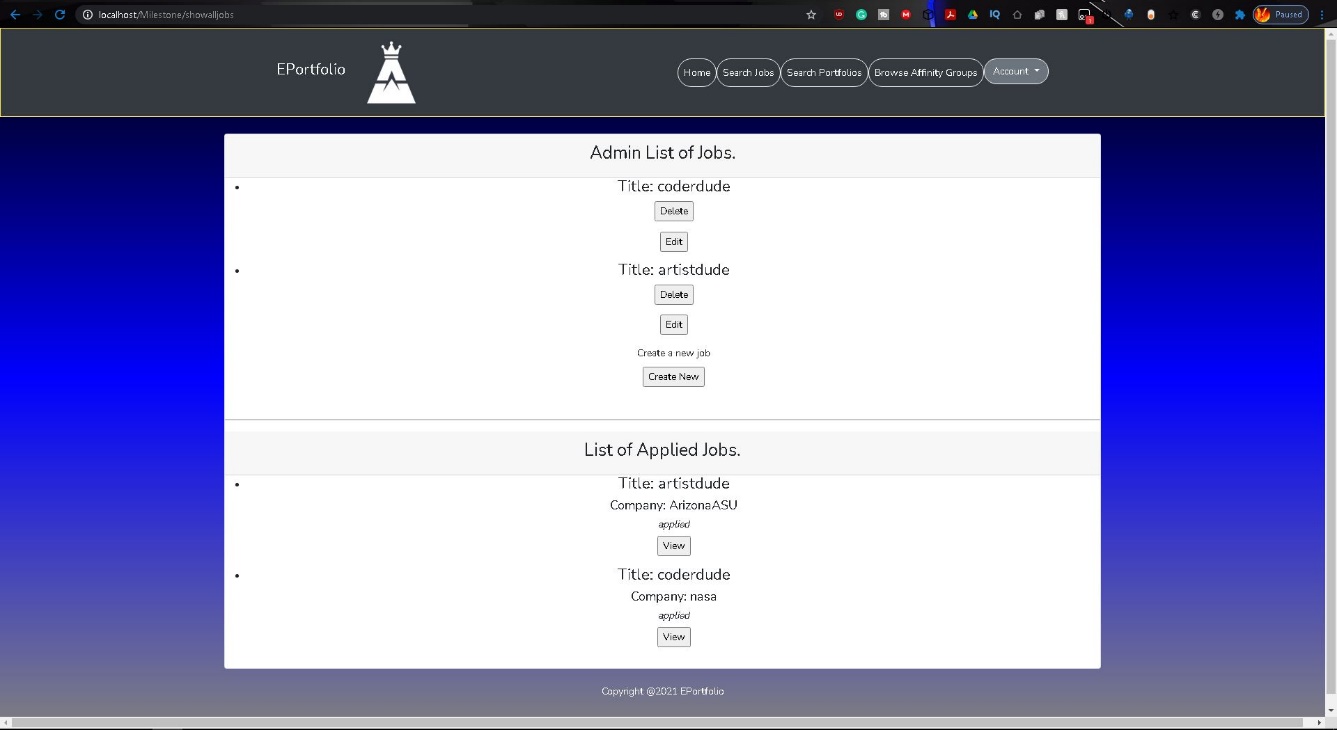
Portfolio Search Results:



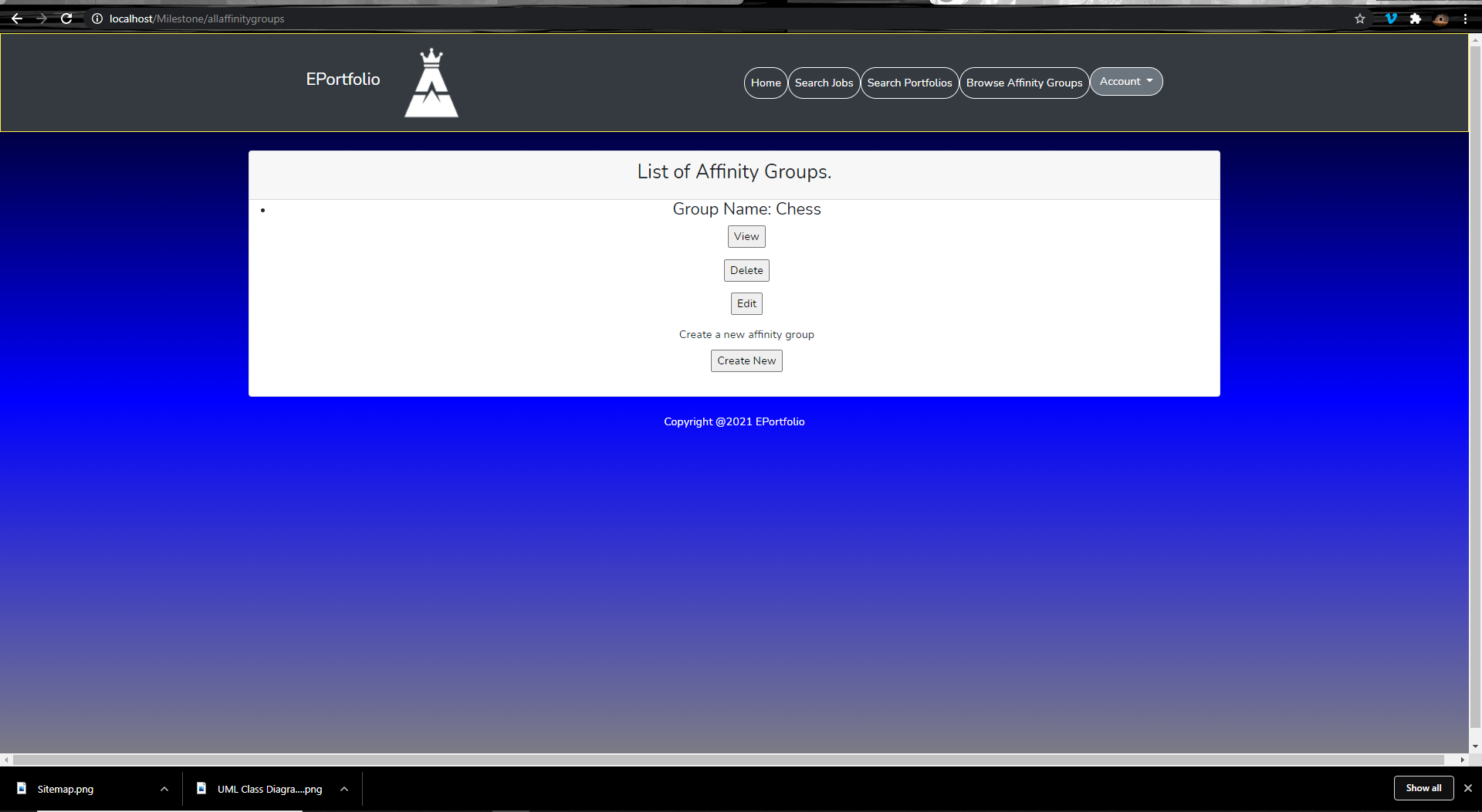
Contact Us:



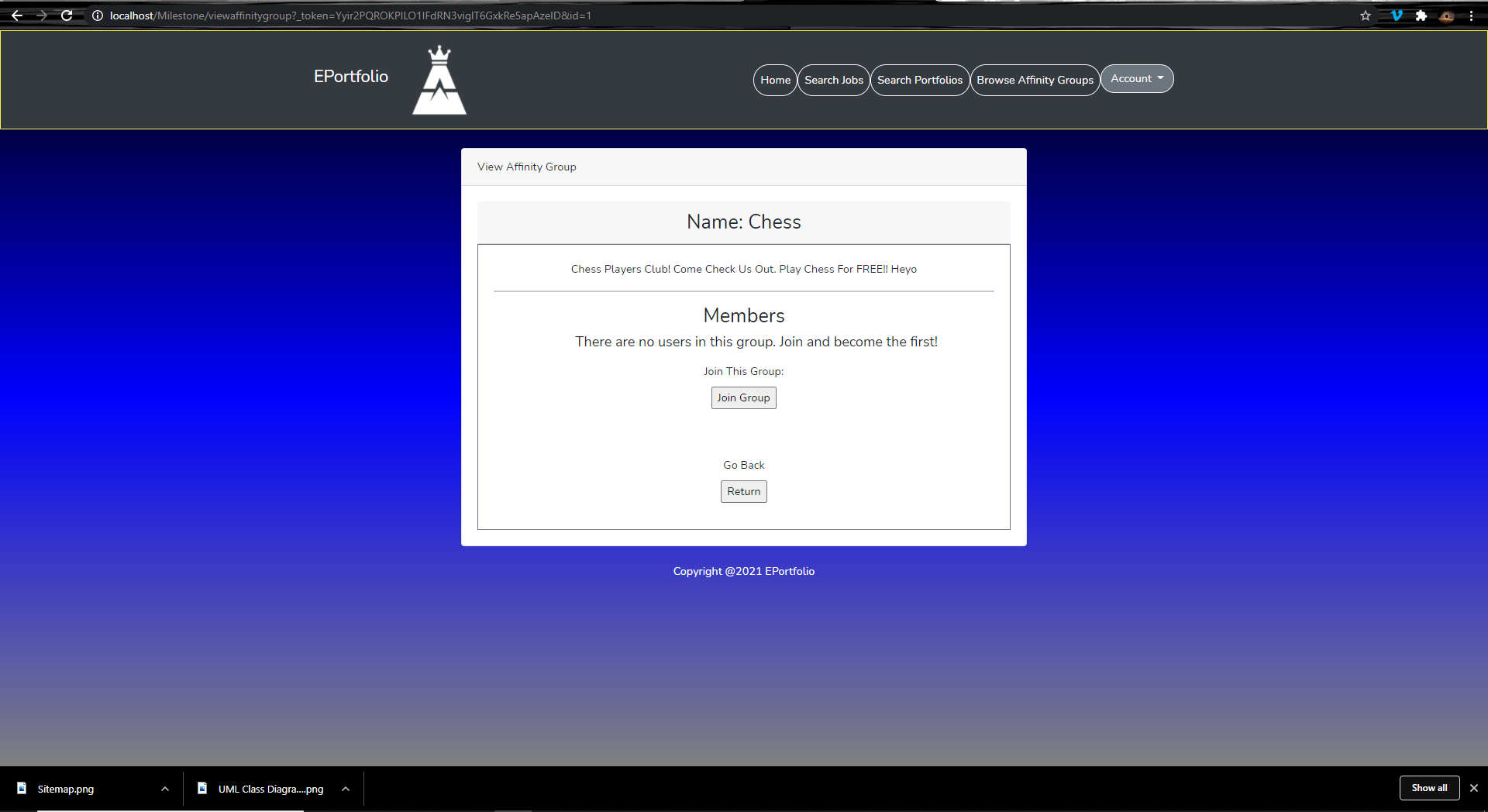
User Roles:



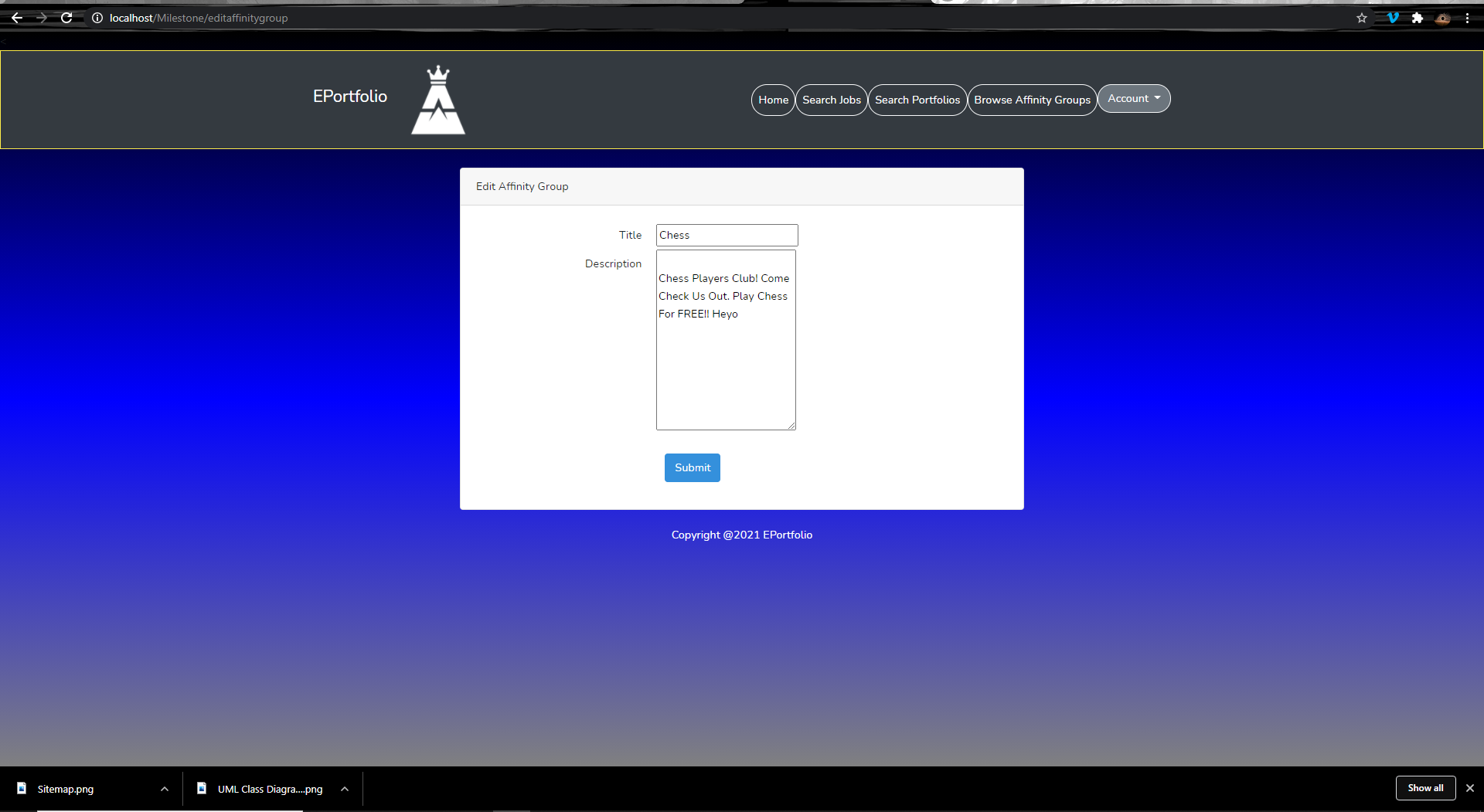
Affinity Groups Page:



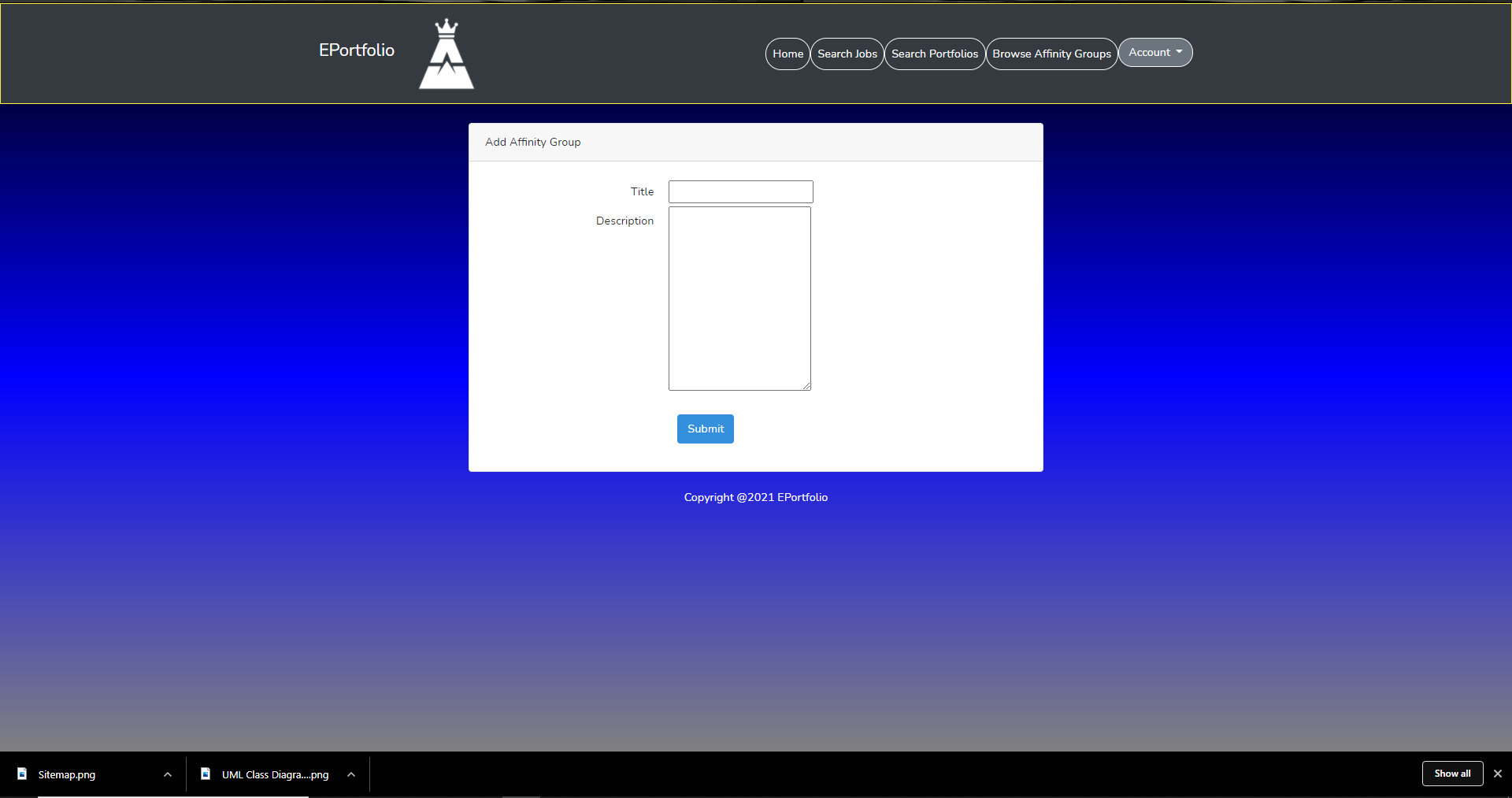
View Affinity Group Details Page:



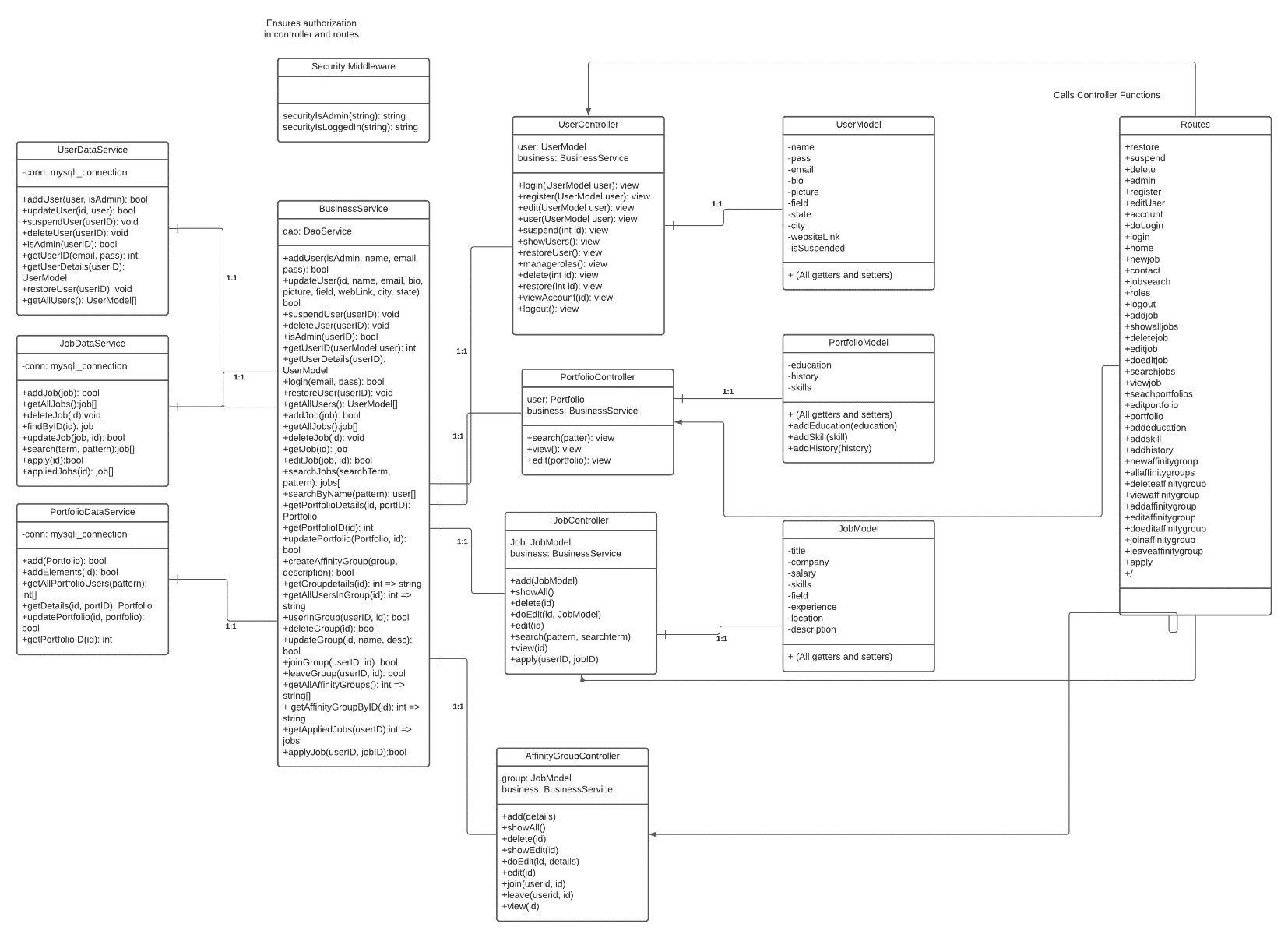
Edit Group Page:



Create New Group Page:



**Class Diagrams:**



**Pseudo Code:**

Code is documented and outlined in comments and the UML

**Other Documentation:**

Code is documented and outlined in comments and the UML