**CST-339 Programming in Java III**

**Project Status and Design Report**

|  |  |  |
| --- | --- | --- |
| **Topic:** | *Topic 2* | |
| **Date:** | *3/26/23* | |
| **Revision:** | *1.0* | |
| **Team:** | 1. *Kevin Bloomquist* | |
| 1. *Noah Funderburgh* | |
| 1. *Arleth Martinez* | |
| 1. *Michael Melichar* | |
| **Weekly Team Status Summary:** | |  |  |  |  | | --- | --- | --- | --- | | **User Story** | **Team**  **Member** | **Hours**  **Worked** | **Hours Remaining** | | *As a developer, I want to know what user stories to incorporate in my project so I can more easily work on the project* | *Michael* | *1* | *0* | | *As a user, I want to be able to view a nicely formatted web page so that I can easily navigate and find the information I need.* | *Kevin* |  |  | | *As a developer, I need to link a bootstrap CSS stylesheet to my Spring Boot project so that the web page can be styled properly.* | *Kevin* |  |  | | *As a developer, I need to set up an index.html page with a title and logo so that the main page is visually appealing.* | *Arleth, Noah* | *1* | *0* | | *As a developer, I need to use Spring MVC for dynamic page generation so that the main page can be updated easily.* | *Kevin* |  |  | | *As a user, I want to be able to register for the website so that I can access its features.* | *Noah* | *2* | *0* | | *As a developer, I need to use Spring MVC for page generation so that the registration page can be updated easily.* | *Noah* | *1* | *0* | | *As a developer, I need to create a User object so that user information can be passed from the controller to the database.* | *Noah* |  |  | | *As a developer, I need to design a database model for user registration, but implementation is not required until Milestone 4.* | *Arleth* |  |  | | *As a developer, I need to validate user input and give appropriate error messages so that users can register correctly.* | *Noah* | *2* | *0* | | *As a user, I want to be able to log in to the website so that I can access its features.* | *Kevin* |  |  | | *As a developer, I need to use Spring MVC for page generation so that the login page can be updated easily.* | *Kevin* |  |  | | *As a developer, I need to emulate authenticating the user since there won't be a database to validate against until Milestone 4.* | *Kevin* |  |  | | *As a developer, I need to design and implement a Login/principle object model so that user authentication can be handled properly.* | *Kevin* |  |  | | *As a developer, I need to validate user input and give appropriate error messages so that users can log in correctly.* | *Kevin* |  |  | | |
| **GIT URL:** | *https://github.com/ktbloomq/Milestone-CST-339* | |
| **Latest ZIP:** | [*milestone 2*](https://mygcuedu6961-my.sharepoint.com/:u:/g/personal/kbloomquis_my_gcu_edu/ETUGdkyGWuFJvW2MQJWbG9cBVrdYWhcWGdjBkUsqNboo5Q?e=Ru6epe) | |
| **Screencast URL:** | *https://youtu.be/bltl2EA5BFU* | |
| **Peer Review:** | *Y/N* | We acknowledge that our team has reviewed this Report and we agree to the approach we are all taking. |

**Planning Documentation**

**Initial Planning:**

*Tasks for this week included making user stories for the week and making sure they were completed within the allocated time frame. Team members took the assignments that they felt comfortable completing and messaged the group discussion as they completed their tasks, pushing their updates to a secondary branch before being accepted into the working branch.*

**Retrospective Results:**

*The following table should be completed after each Retrospective on things that went well (keep doing).*

|  |
| --- |
| **What Went Well** |
| App Envisioning – Effective communication amongst the team regarding planning the web app. |
|  |
|  |

*The following table should be completed after each Retrospective on things that didn’t go well (stop doing) and what would be done differently next time with an action plan to improve (try doing and continuous improvement).*

|  |  |  |
| --- | --- | --- |
| **What Did Not Go Well** | **Action Plan** | **Due Date** |
|  |  |  |
|  |  |  |
|  |  |  |

**Design Documentation**

**Install Instructions:**

*Navigate to the desired install directory and clone into* [*https://github.com/ktbloomq/Milestone-CST-339.git*](https://github.com/ktbloomq/Milestone-CST-339.git) *using git clone* [*https://github.com/ktbloomq/Milestone-CST-339.git*](https://github.com/ktbloomq/Milestone-CST-339.git) *.*

**General Technical Approach:**

*Our team is looking to develop a new e-commerce website to sell products online. The website will allow users to browse and purchase products, create new products, display details about a product, update an existing product and delete products. We want to create an easy-to-use, visually appealing website.*

**Key Technical Design Decisions:**

*Any final technical design decisions, such as framework decisions, should be documented here. This should list the technology/framework, its purpose in the design, and why it was chosen.*

**Known Issues:**

*Any anomalies or known issues in the code or functionality should be documented here.*

**Risks:**

*The scope of this project could become too ambitious for the given timeframe. User payment and transaction information could be insecure*

**ER Diagram:**

*Table

Description automatically generated with medium confidence*

**DDL Scripts:**

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

**Sitemap Diagram:**

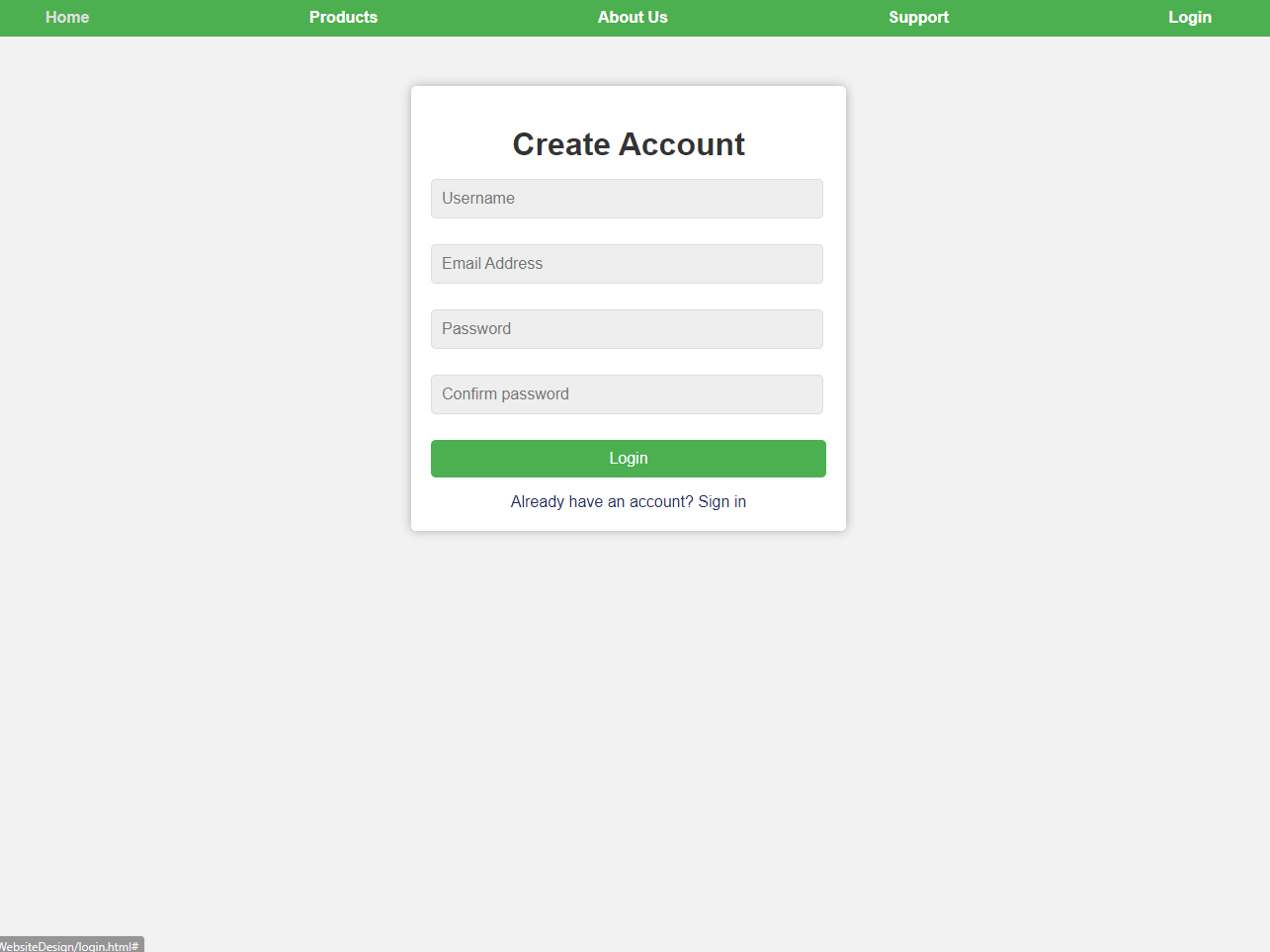
*Diagram

Description automatically generated*

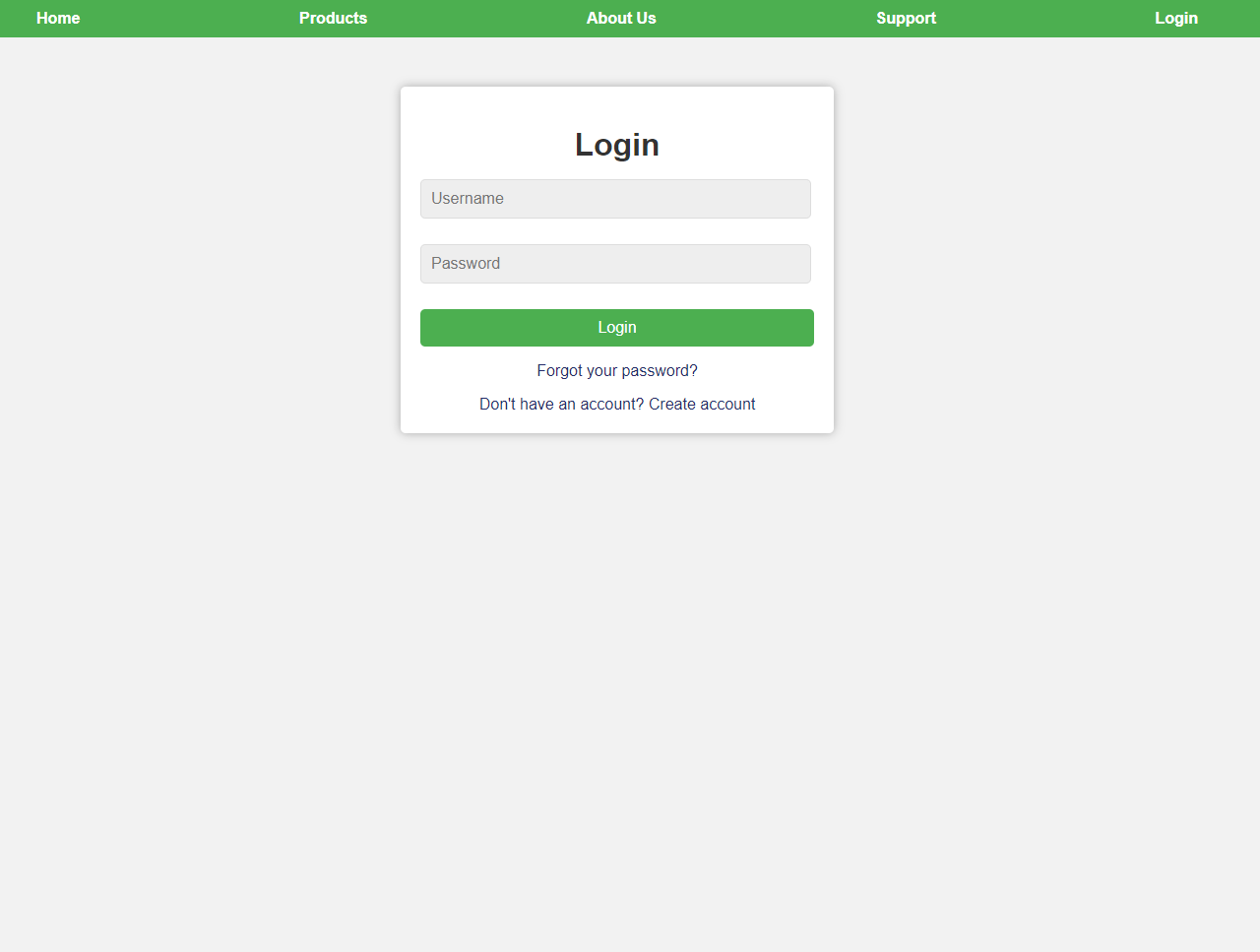
**User Interface Diagrams:**

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

*Registration Design Diagram*

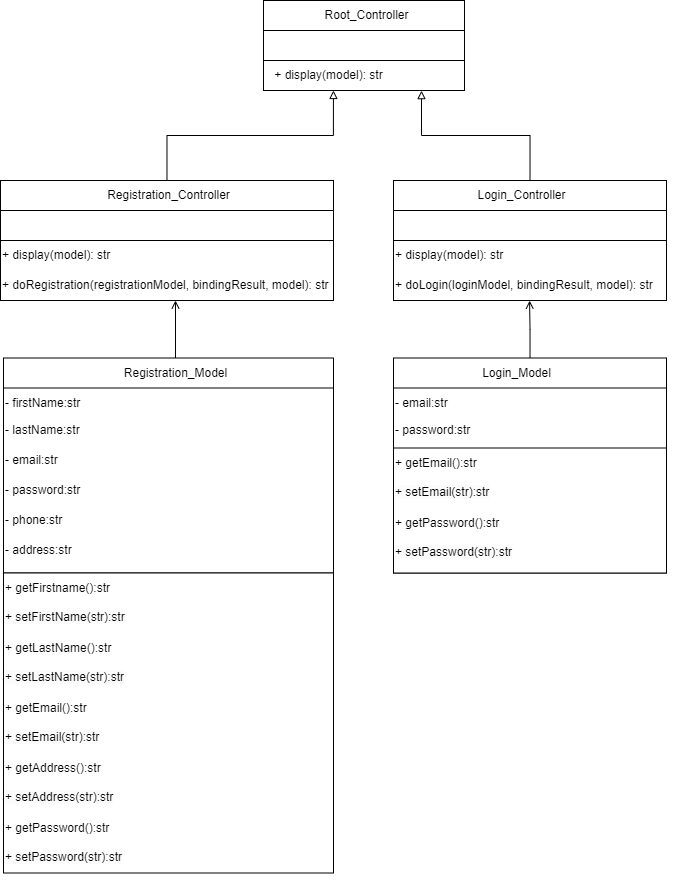


Login *Design Diagram*



**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 rationale for why you are able to leave this section as N/A.*



**Service API Design:**

*This section should fully document any service API’s (like REST API’s) that are being published, 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 third party developer to integrate with the service and API. The design can also be captured with tools such as Swagger.*

**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.*

**Other Documentation:**

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