**.NET Application Programming**

**Project Status and Design Report**

|  |  |  |
| --- | --- | --- |
| **Topic:** | *Milestone 2: Scrum Planning* | |
| **Date:** | *8/15/2020* | |
| **Revision:** | *2.0* | |
| **Team:** | 1. *Joseph Cooper* | |
| 1. *Cameron Deao* | |
| 1. Vison Martin | |
|  | |
| **Weekly Team Status Summary:** | |  |  |  |  | | --- | --- | --- | --- | | **User Story** | **Team**  **Member** | **Hours**  **Worked** | **Hours Remaining** | | *Registration page/view* | *Vinson Martin* | *2* | *0* | | *Login page/view* | *Vinson Martin* | *2* | *0* | | *Login controller* | *Cameron Deao* | *3* | *0* | | *Registration controller* | *Cameron Deao* | *3* | *0* | | *SQL Database* | *Joseph Cooper* | *3* | *0* | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | | |
| **GIT URL:** | <https://github.com/cmdeao/CLC---CST-247/tree/Milestone2-Login-And-Registration-Submission> | |
| **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**

**Agile Scrum Product Backlog:**

<https://github.com/cmdeao/CST-247---CLC-Project/blob/master/Planning%20and%20Design/CST-247-RS-SprintProductLog%20-%20CLC.xlsx>

**Agile Scrum Sprint Backlog:**

<https://github.com/cmdeao/CST-247---CLC-Project/blob/master/Planning%20and%20Design/CST-247-RS-SprintBackLog%20-%20CLC.xls>

**Agile Scrum Burn Down Chart:**

<https://github.com/cmdeao/CST-247---CLC-Project/blob/master/Planning%20and%20Design/CST-247-RS-SprintBurnDown%20-%20CLC.xlsx>

**Agile Retrospective Results:**

*The following table should be completed after each Retrospective on Things That Went Well (Keep Doing). An alternative to the following table is to use a Mind Mapping tool such as Coggle. If you use a Mind Mapping tool you must include a URL or Image File.*

|  |
| --- |
| **What Went Well** |
| **Communication was much stronger and more frequent.** |
| **Code came together in a timely manner for peer review.** |
|  |

*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). An alternative to the following table is to use a Mind Mapping tool such as Coggle. If you use a Mind Mapping tool you must include a URL or Image File.*

|  |  |  |
| --- | --- | --- |
| **What Did Not Go Well** | **Action Plan** | **Due Date** |
| **Repository was missing a file.** | **Create new repository with appropriate .gitignore file included.** | **8/16/20** |
|  |  |  |
|  |  |  |

**Design Documentation**

**Install Instructions:**

*Download and unzip project folder.*

*To view in Visual Studio 2019: File -> Open -> Project/Solution -> Minesweeper Web Application (Minesweeper Web Application/Minesweeper Web Application.sln) -> Open.*

**General Technical Approach:**

*As a team we’ll be porting a previous version of Minesweeper, developed in CST-227, into an ASP.NET MVC format. We’ll be utilizing the code developed by Cameron Deao as a basis for gameplay, the web application will be built through a .NET N-Layer, and the entire project will be built within Visual Studio 2019.*

**Key Technical Design Decisions:**

*Within the project, views will be built utilizing the Razor Syntax engine, the SQL database will be established on each machine and initialized with SQL scripts, and the application framework will revolve leverage MVC 5. Alongside the project frameworks the team will utilize an established GitHub repository and Sourcetree for source control. Razor Syntax was chosen by the group in order to reduce required syntax for views, personal SQL database initialized with SQL scripts was chosen for convenience amongst the group members, and MVC 5 for the separation of concerns within the project. GitHub and Sourcetree were chosen so the team could perform their work on separate branches, maintain a working copy of the project in a ‘master’ branch, and perform pull requests for peer reviews of code before merging into the ‘master’ branch.*

**ER Diagram:**

**

**DDL Scripts:**

**CREATE TABLE [dbo].[users] (**

**USERID int IDENTITY(1,1) NOT NULL PRIMARY KEY,**

**FIRSTNAME varchar(25) NOT NULL,**

**LASTNAME varchar(25) NOT NULL,**

**AGE int NOT NULL,**

**GENDER varchar(1) NOT NULL,**

**STATE varchar(25) NOT NULL,**

**EMAILADDRESS varchar(50) NOT NULL,**

**USERNAME varchar(25) NOT NULL,**

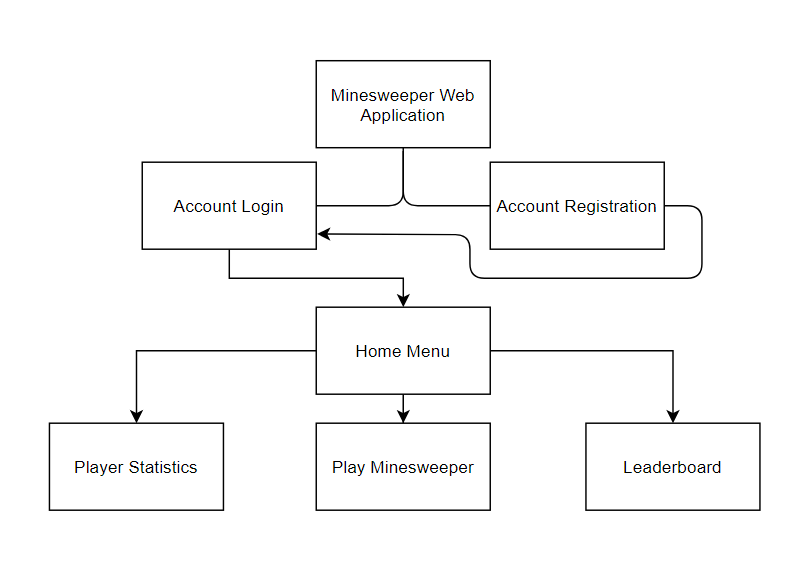
**PASSWORD varchar(25) NOT NULL,**

**)**

**INSERT INTO dbo.users(FIRSTNAME, LASTNAME, AGE, GENDER, STATE, EMAILADDRESS, USERNAME, PASSWORD)**

**VALUES ('Cameron', 'Deao', '26', 'M', 'Texas', 'testing@yahoo.com', 'cdeao', 'testing');**

**Sitemap Diagram:**

**

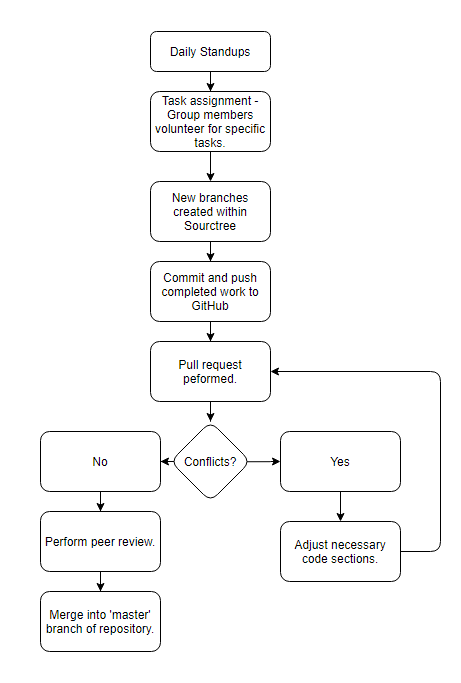
**Security Design:**

*The authentication and authorization will be broken down into a services class. Users attempting to login to the application will be passed through this class and compared against established users within the database. In the event that features are added to the project that require administrative roles, the database will be updated along with the services class to compare roles when allowing access into the application.*

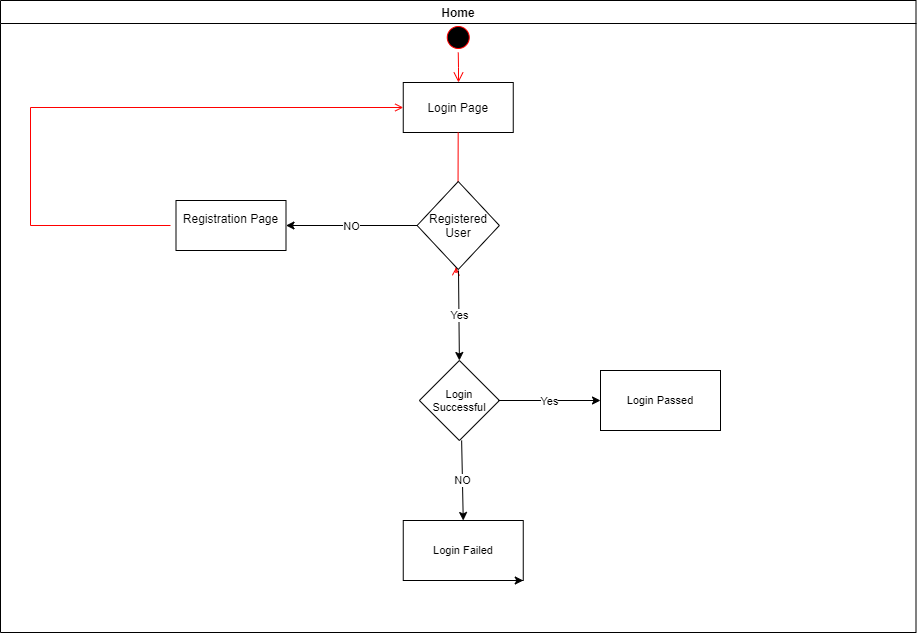
**Third Part Interface Design:**

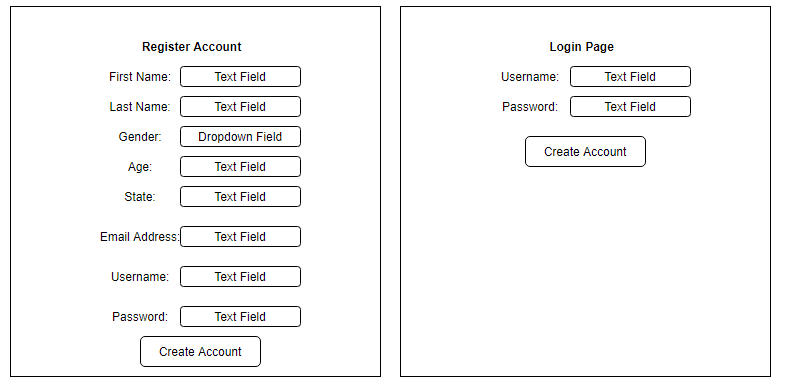
*N/A*

**Flow Charts:**

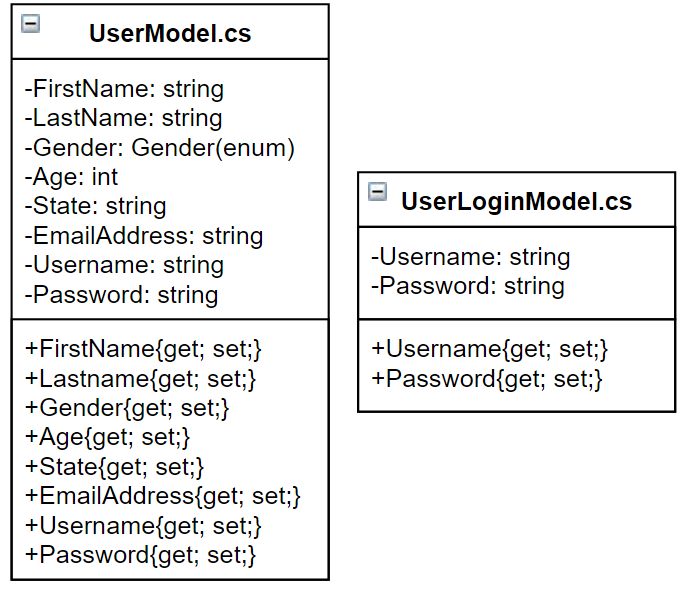
****

**User Interface Diagrams:**





**Class Diagrams:**



**Pseudo Code:**

*The Minesweeper portion will be ported based on the application built by Cameron Deao in CST-227.*

*https://bitbucket.org/cdeao/cst-227-milestone-6/src/master/*

**Other Documentation:**

*N/A – Currently no drawings, storyboards, or project schedules exist outside of the product log, back log, and burn down chart. This section will be updated as necessary as we move further into production and develop mockups of gameplay, statistics, and leaderboard views.*