**.NET Application Programming**

**Project Status and Design Report**

|  |  |  |
| --- | --- | --- |
| **Topic:** | *FR7 Logging, DI, Security Filter and general clean up* | |
| **Date:** | *10-29-2020* | |
| **Revision:** | *7.0* | |
| **Team:** | 1. *Matt Sievers* | |
| 1. Crystal Sciarrino | |
|  | |
|  | |
| **Weekly Team Status Summary:** | |  |  |  |  | | --- | --- | --- | --- | | **User Story** | **Team**  **Member** | **Hours**  **Worked** | **Hours Remaining** | | BUG - Right click on game board not reloading partial view | *Matt Sievers / Crystal Sciarrino* | *3* | *0* | | As a developer, I need to using unity framework to implement the dependency injection of the new logger service on the user login | *Matt Sievers* | *2* | *0* | | As a developer, I need to update the home view to return the login page instead of a string | *Crystal Sciarrino* |  | *0* | | As a developer I need to add a security filter to ensure that a user cannot pass the login page if not logged in | *Crystal Sciarrino* | *0* | *0* | | As a developer, I need to implement a logging service that can be used throughout the application in conjunction with DI | *Matt Sievers* | *2* | *0* | | As a team, we need to do weekly scrum paperwork | *Matt Sievers / Crystal Sciarrino* | *2* | *0* | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | |  |  |  |  | | |
| **GIT URL:** | https://github.com/Malleas/CLCMinesweeperApp2 | |
| **Peer Review:** | *Y* | We acknowledge that our team has reviewed this Report and we agree to the approach we are all taking. |

**Planning Documentation**

**Agile Scrum Board:**

[**https://www.meistertask.com/app/project/sSO4jU4u/cst-247**](https://www.meistertask.com/app/project/sSO4jU4u/cst-247)

**Agile Scrum Product Backlog:**

https://github.com/Malleas/CLCMinesweeperApp/tree/master/CLCMinesweeperApp/Planning%20and%20Design/Week%207

**Agile Scrum Sprint Backlog:**

*https://github.com/Malleas/CLCMinesweeperApp/tree/master/CLCMinesweeperApp/Planning%20and%20Design/Week%207*

**Agile Scrum Burn Down Chart:**

*https://github.com/Malleas/CLCMinesweeperApp/tree/master/CLCMinesweeperApp/Planning%20and%20Design/Week%207*

**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** |
| **Continued with great working session and peer programing.** |
| **Able to implement remaining items and fix the workflow so you can’t get to any page in the 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). 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** |
| Right click proved to still be challenging to fix. | Bug logged, will continue to work on it | **10/31** |
|  |  |  |
|  |  |  |

**Design Documentation**

**Install Instructions:**

*Connect and pull from master from repo. Refer to DDL scripts before running as a local DB will need to be created and table added. Click debug menu option and select Run without debug. Backup provided of DB for full install as needed.*

*Please note that two additional tables have been added since previous build so refer to the DDL scripts to build those tables in a local database.*

**General Technical Approach:**

*For this we used existing assignment work to help fill in the gaps needed to create the security filter, logging and dependency injection. Using the ILogger and MyLogger from Assignment 1 helped implement this feature rather quickly. Continued to troubleshoot the right click button to get the partial page to refresh on that click.*

**Key Technical Design Decisions:**

*Instead of trying to reinvent the wheel, we both chose to reuse code from the assignments to help speed up development processes, allowing for more time to fix the bug.*

**ER Diagram:**

*n/a for this requirement*

**DDL Scripts:**

*CREATE TABLE [dbo].[Player]*

*(*

*[USERID] INT NOT NULL PRIMARY KEY IDENTITY(1,1),*

*[FIRSTNAME] VARCHAR(50) NOT NULL,*

*[LASTNAME] VARCHAR(50) NOT NULL,*

*[GENDER] VARCHAR(50) NOT NULL,*

*[AGE] INT NOT NULL,*

*[STATE] VARCHAR(2) NOT NULL,*

*[EMAILADDRESS] VARCHAR(100) NOT NULL,*

*[USERNAME] VARCHAR(50) NOT NULL,*

*[PASSWORD] VARCHAR(50) NOT NULL*

*)*

*CREATE TABLE [dbo].[Stats] (*

*[Id]     INT IDENTITY (1, 1) NOT NULL,*

*[Time]   INT NOT NULL,*

*[Clicks] INT NOT NULL*

*);*

*CREATE TABLE [dbo].[Game] (*

*[Id]        INT           IDENTITY (1, 1) NOT NULL,*

*[Gameboard] VARCHAR (MAX) NOT NULL*

*);*

**Sitemap Diagram:**

*N/A not required for this FR*

**Security Design:**

*Added hashing to the password entry at new user creation so we’re not storing actual passwords into the db.*

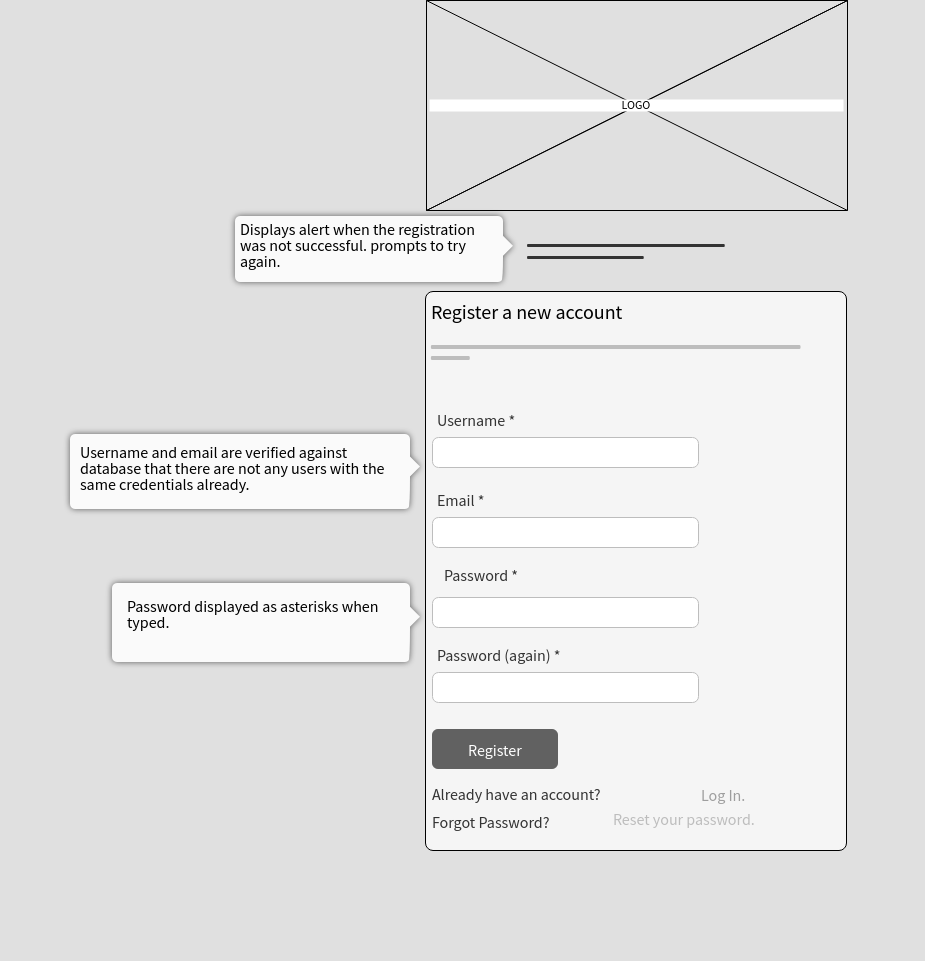
**Third Part Interface Design:**

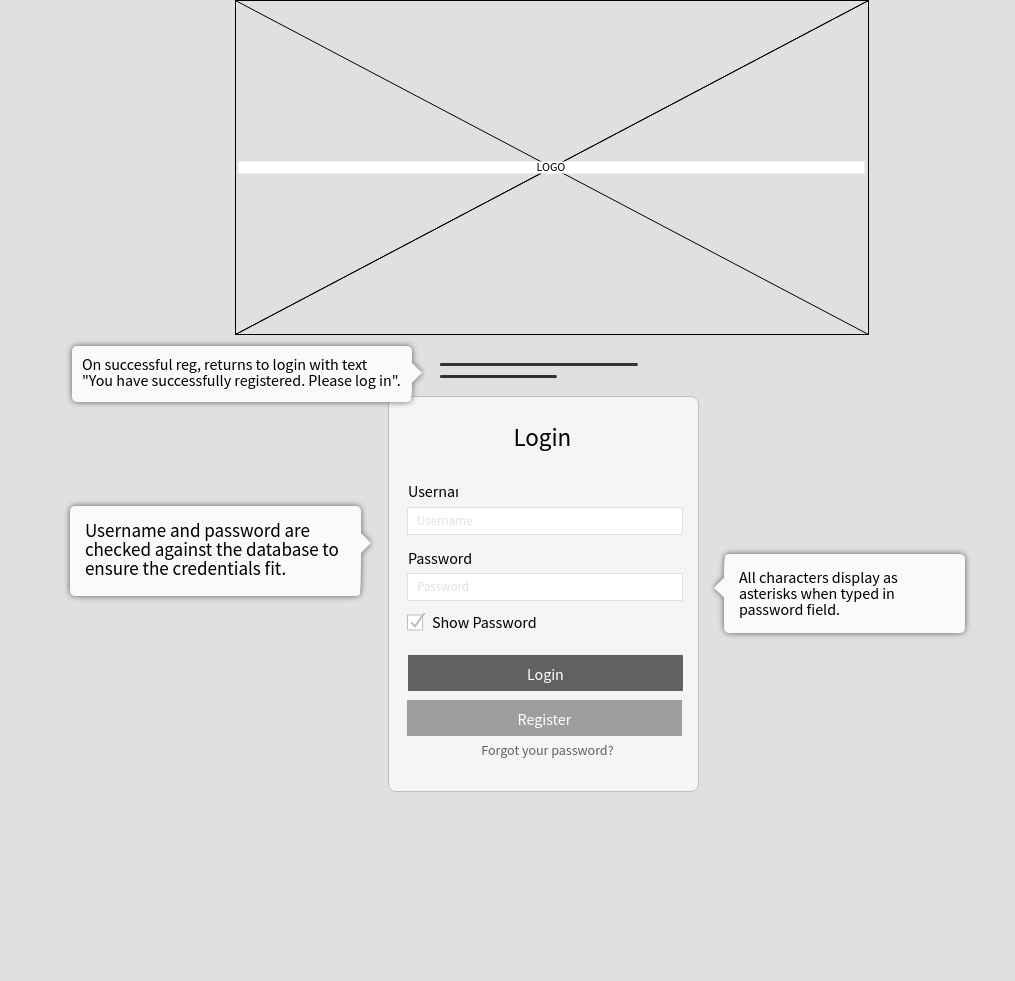
*NA*

**Flow Charts:**

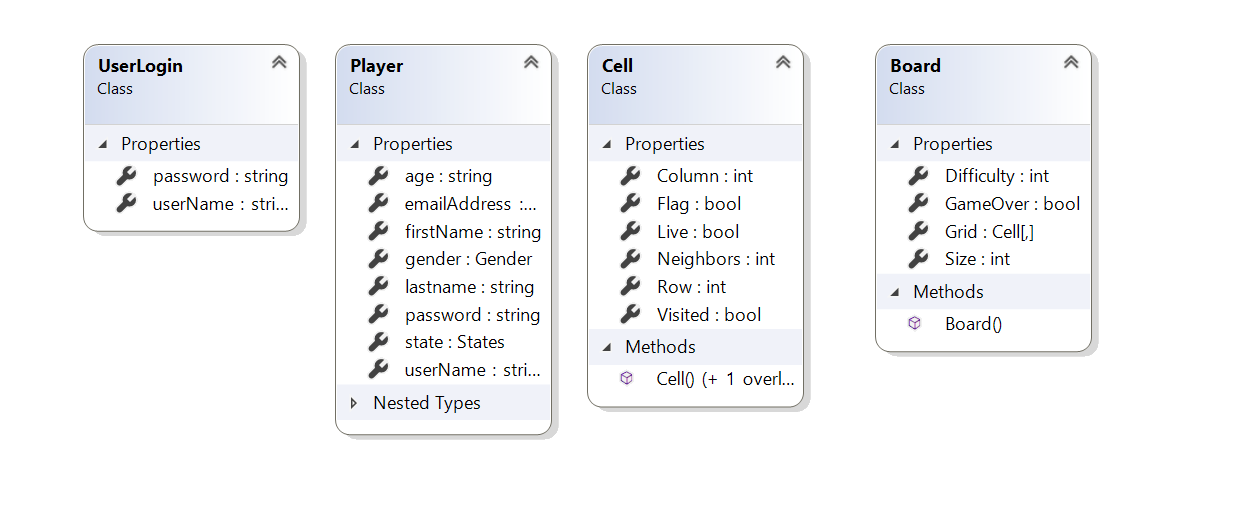
*NA*

**User Interface Diagrams:**





**Class Diagrams:**



**Pseudo Code:**

*N/A*

**Other Documentation:**

*N/A*