
User Manual

- Note: This Application is made up of three graphical window user interface.
- There are no additional resources required to run this application.
- The application is developed with Visual studio.
- Use The latest version of the Visual studio for optimal performance.
- Running the application is simple, open the .sln file

Name	Date modified	Type	Size
packages	2021/10/23 13:50	File folder	
PROG6212_Task2	2021/10/24 14:46	File folder	
PROG6212_Task2.sln	2021/10/23 13:42	Visual Studio ...	2 KB

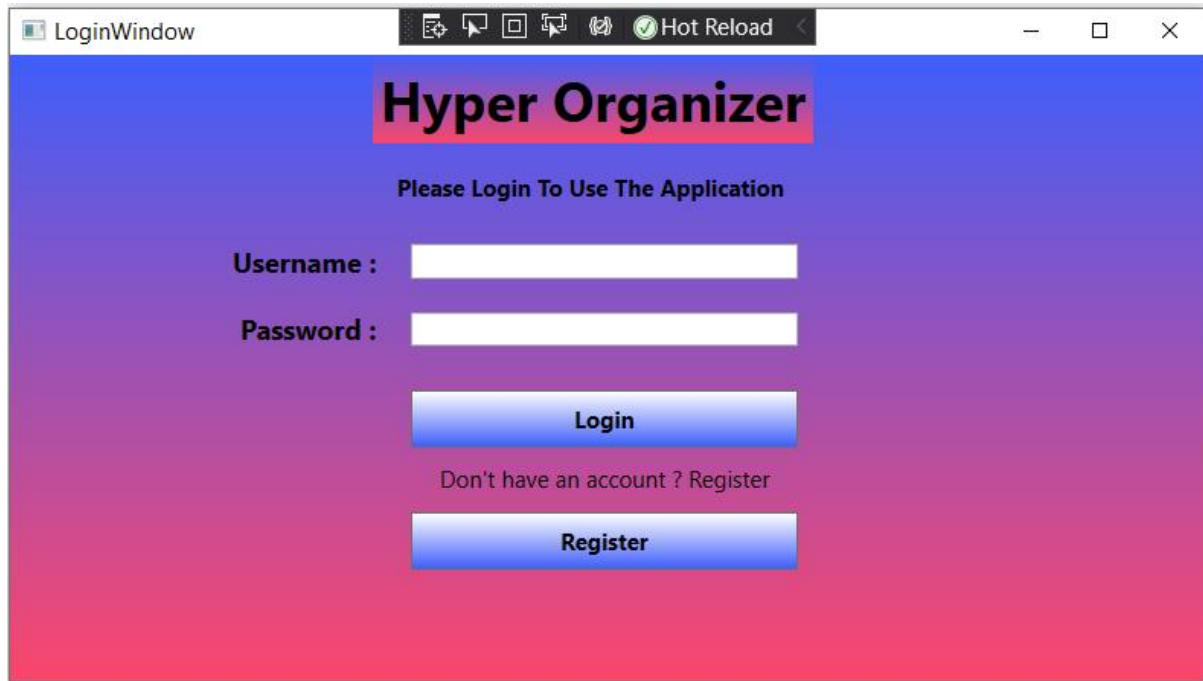
OR

- Run the exe file in the bin folder.

Name	Date modified	Type	Size
ClassLibrary.dll	2021/10/24 00:31	Application ex...	5 KB
ClassLibrary.pdb	2021/10/24 00:31	Program Debu...	22 KB
EntityFramework.dll	2017/10/23 13:15	Application ex...	5 103 KB
EntityFramework.SqlServer.dll	2017/10/23 13:15	Application ex...	608 KB
EntityFramework.SqlServer	2017/10/23 13:15	XML Document	155 KB
EntityFramework	2017/10/23 13:15	XML Document	3 594 KB
PROG6212_Task2	2021/10/24 14:46	Application	55 KB
PROG6212_Task2.exe	2021/10/23 13:50	XML Configur...	2 KB
PROG6212_Task2.pdb	2021/10/24 14:46	Program Debu...	118 KB

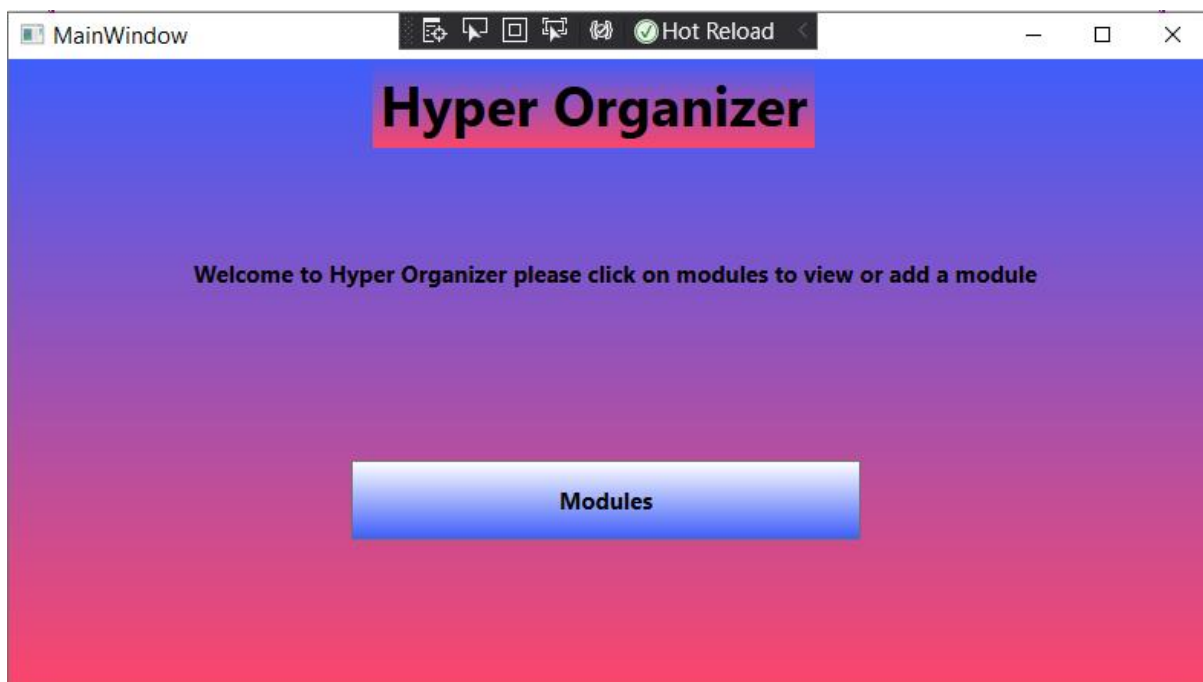
Login Window

- This is the login window and user is direct here before using any functionalities.
- The user can login or register an account here.
- Enter a username and password and click on register to create an account then proceed with login.



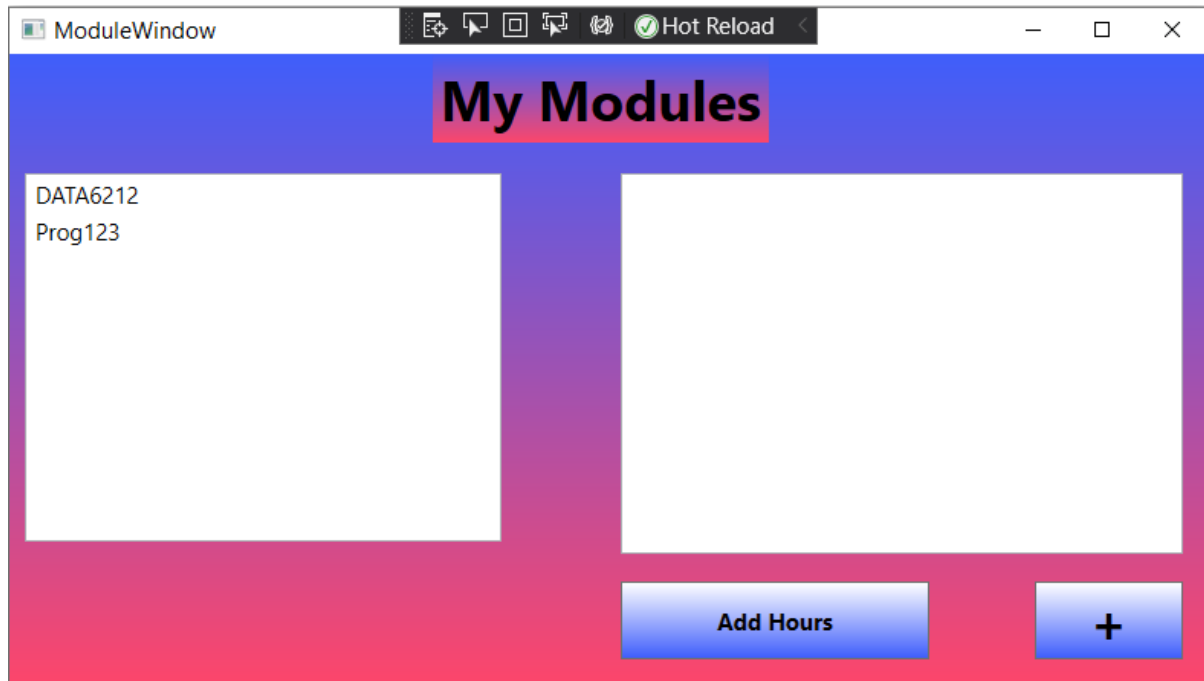
Main Window

- The main window serves as a menu and directs you to how to use the app.



Module Window

- The Module window contains two display areas and a list box that displays the stored modules and a textbox that displays module details.
- The user can view the modules that have been stored in the database.
- Add Module hours by selecting on the Add hours button.
- This window also contains the key button represented by **+** which takes the user to the add modules window.
- To view the details of a module, select the module.



Add Module window

- This window is where a user can add modules to a list that stores it in the database and can be accessed throughout the application.
- All Textboxes need to be inserted with correct character values before successfully storing a module.

The screenshot shows a web application window titled "AddModulesWindow". The interface has a blue header with a "View Modules" button on the left and a large "Add Modules" title in the center. Below the header is a form with two columns of input fields. The left column contains "Module Code", "Module Name", and "Module Credits". The right column contains "Class hours per week", "Semester Weeks", and "Module Start Date" (which includes a date picker set to the 15th). A "Save" button is located at the bottom center of the form area.

Add Module Hours window

- This window is where a user can add daily module hours for a specific module.
- This data is stored in the database and can be accessed throughout the application.

The screenshot shows a web application window titled "AddHoursWindow". The interface has a blue header with a "<< Back" button on the left and a large "Add Hours" title in the center. Below the header is a form titled "Add self-study hours for the selected module". The form contains three input fields: "Module :" (a dropdown menu), "Date :" (a date picker set to the 15th), and "Number of hours :". A "Submit" button is located at the bottom center of the form area.

Results

- Once the user selects a module from the list box, all the details about the module is displayed to the user.
- All functionalities become available once the user selects a module.
- The user can view the updated Remaining hours after it is added.

