To provide a small background: Team is currently using software provided by Rockwell Automation to create and design HMI screens. This software is quite dated and can be difficult to use therefore Rockwell is releasing a newer and (hopefully) better software for this. I will be using an alpha release of this software as it is not released to the general public yet. I hope none of this is classified info??

The project will be mostly (if not all, I’m not entirely sure yet) C#.

**FYP Outline**

First section is to develop the screens, ensuring working functionality and uniform styling.

1. Input validation for all input fields. E.g.: Incorrect inputs will display dynamic error messages.

FluentValidation will be used to perform checks on the text within the input fields.

1. Button pattern => buttons will have multiple states. E.g.: Button greyed out until conditions are met (correct input, etc…).
2. Responsive design => Essentially, resizing the window will not break the design (think along the lines of Bootstrap).
3. Vector graphics => Will try to include SVG optics. E.g.: Box with 4 sides, 1 side will have different colour. (Will have more information on this point later)
4. Stylesheets to be used for the screens.
5. A functional 2D barcode scanner to be included.
6. Reusable graphics. E.g.: I’ll create 1 graphic (some sort of box or input field) and use multiple instances of that graphic throughout.

Next section is related to auto-generation of code. These screens are all just yaml files behind the scenes, allowing them to be auto-generated to a certain extent. I’ll most likely be using String Template 4 for this but this is not finalised yet. Once I have more concrete info on this section I’ll relay it to you.

The last section is related to automated testing of the HMI screens. The software allows the user to load a localhost webpage instance of the screens. This will make creating tests for the screens very accessible with the use of the developer tools window. The software to be used for this is Selenium and Specflow and the language will also be C#.