Coursework write up

Each title(underlined) will be on a separate page.

Content page

Analysis

My program involves a dynamic GUI and multi-threading functions. I’ve built it with the user in mind and for it to be as simple as possible. The user was the computer science and physics department to teach logic gates to the younger years. At the moment no current program is used by the school but to have bespoke software designed for them will change that.

This is aimed at educational purposes and to teach younger years the basics and understand of how logic gates and logic circuits work. For the computer science department, they wanted a more sophisticated system that would allow more advance logic circuits like flip flop. They also required basic systems like save and open previous files and a help guide on how to use the program for the students.

This project required a good understanding of how to use WPF C# as this was my programming choice. I’ve had to came familiar with the XAML and how the classes in WPF worked. Because of the requirements I had to research into multithreading. This was convivence due to C# have a good support for it OOP and multithreading.

This project required a good understanding of how to use WPF C# as this was my programming choice. It’s brought me to make tough choices in the direction that the program should go to. I was required to learn new aspects of the C# language for the more advance features ad challenges I came across.

Interview

I’ve interviewed my computer science teacher (Mr Stephenson) on what functions the program should hold. I asked class mates on the design to the GUI and how it should interact. This allowed me to model and develop based directly on their requests.

General Objectives

Achieve a fully working logic gate circuit simulator that will work flawlessly and fulfil the required needs of the departments. As a minimum it should incorporate all of the gates on the physics and computing specifications in addition to allowing multiple outputs from each gate.

I aim to work closely with each group and get feedback on the way to improve and master what is required.

Data sources & Destination

All data for the project that will be stored will be based on the class called Gate\_class. The list of object will be saved so that the user can load it back up later if required. I believe this doesn’t oppose a security threat as the data is not autonomous and only storing information on the formation and data values of the logic gates and how they’re placed. Everything else will be reconstructed so that the file size is a small as possible. The file type will be JSON as they’re perfect for storing class and are easy to read again.

Programmed in C#

Although C# is a very powerful and useful language when creating graphical user interfaces the language struggle to call and pass variables into the event handlers of a WPF class. There are multiple ways of getting around and solving the problem but none of them are appealing or suitable for large projects. The method I choose in the end works well for what is required but is messy and complicated but I will talk about that more later.

Justification of chosen solution

pros cons of 2 other languages

It is a language I know well and suitable for the project due to it’s OOP capabilities. It also gives a lot of control to the programmer which was required and will be talked about more. Mr Stephenson also wanted file saves which is more compatible when it’s easy to convert classes to JSON. I programmed it inside of Visual Studio 2019 using windows presentation foundation.

Limitations

Due to how my UI is a canvas and not a box grid I could not implement a successful path finding algorithm for the wires that connects the gates. This was seemed reasonable by the computer science department as a task like that is implausible to add due to the infinite possibilities and hard to extrapolate data. I’ve worked around this by adding different colours to the wires so that it’s easier for the eye to track.

Method and sources

Identifying a third party

Further research

Prototyping and critical path