**Year 12 Computer Science**

**Project Proposal**

**Introduction**

GCSE physics students in DGSB require knowledge of circuits as part of their course, but unfortunately the physical equipment to learn many of these circuits is expensive and limited in the school, is easily broken, and home learning is not possible as equipment cannot be taken outside of school. Many students simply learn about these circuits from an image from a textbook.

A practical solution would be a circuit simulator program that demonstrates simple electrical circuits to determine if they will work, and allows students to check if they function correctly. This would benefit for the physics department as the equipment their equipment is fairly old, and sometimes broken, and so a simple program would be a good way to show current split, resistance over distance, voltages, etc. Their equipment is also limited, so is often shared between classes, with not enough for each student.

This program would allow students to experiment with circuits at home as well, as usually students cannot take home equipment because there is a limited amount, and they are also quite delicate.

I will aim for around 10 of the most common components, but if I have time, I will be able to add many more with a simple polymorphic structure for the components.

**Stakeholders**

The physics department would be my stakeholder, as they need a program like this to demonstrate circuits for students. They would want to be able to demonstrate resistance over distance and wire gauge, as these are GCSE core practicals. It is safer to do it on a computer, as a resistance wire heats up considerably, and can burn people.

This means that my program would need to deal with resistance calculations and have variable battery voltages. The info tab would need to be able to select wires, and therefore have resistance/length sliders.

**Existing Solutions**

<https://www.partsim.com/simulator> - a website, but is more suited to the industrial market, many components which are completely irrelevant to most people.

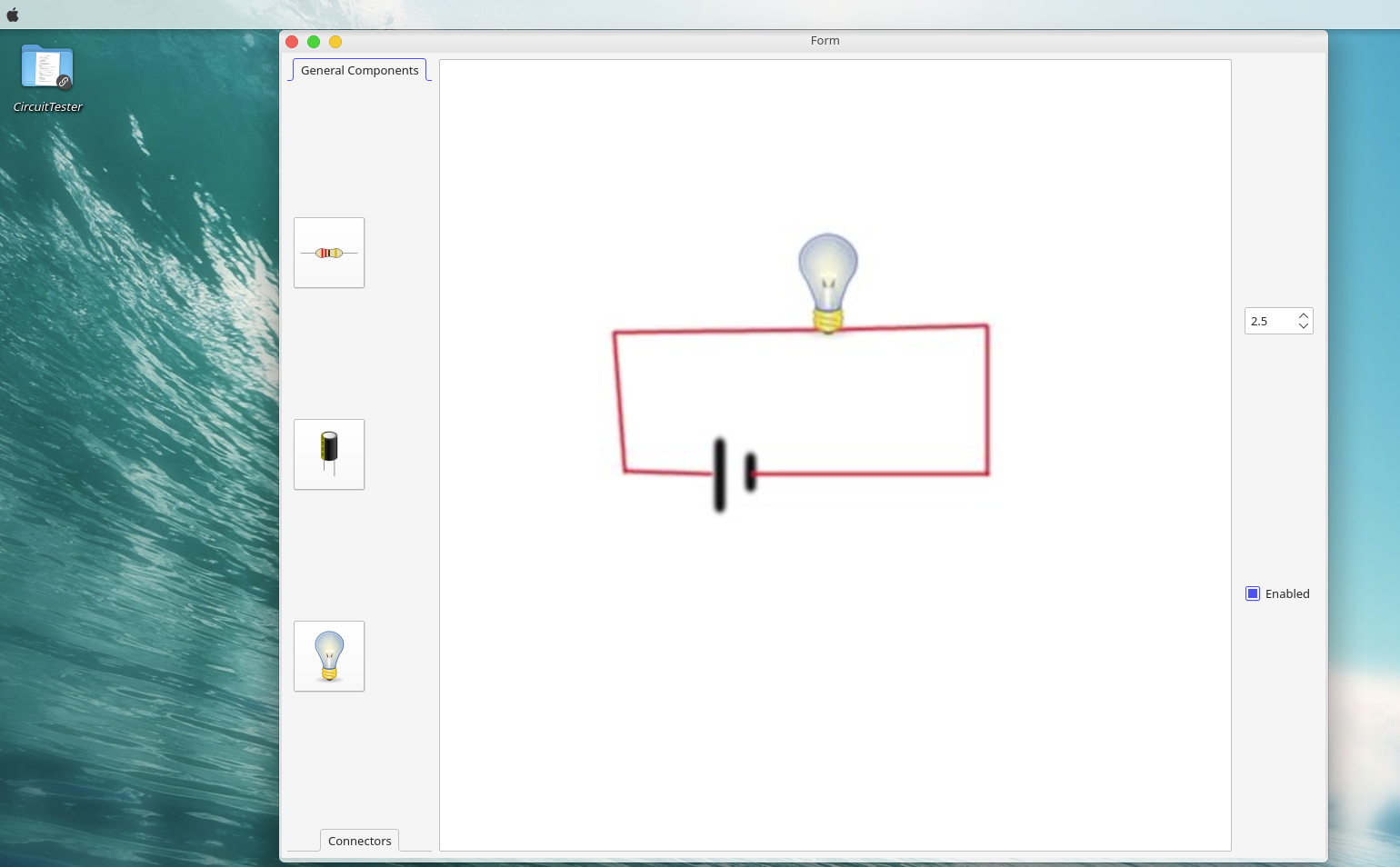
<https://sourceforge.net/projects/qucs/> - a GUI program, this is more like what I had imagined, although there are many components that are not relevant, and some that would be helpful, but are not included here.

Both of these are just too complicated for the average GCSE level student, and therefore people need a simpler program, that allows anyone with a basic understanding of computers and physics to build circuits.

**Classes and Methods**

|  |  |
| --- | --- |
| **Classes** | **Methods** |
| Component (parent, children inherit from this to create the various components) | getOutput(takes in input voltage, current, etc),  getName(returns the component name),  getImage(returns the icon that is used for the component),  getConnected(returns a list of components that are connected), |
| MainWindow – is the window where the elements are placed | init(initialises the main window, adds the toolbar, graphicsArea, infoMenu, etc), |
| GraphicsArea – the window element which is where images and connections are drawn, and where the user can interact with them | addComponent,  removeComponent |
| Toolbar – where the components are stored | addComponent(add item to the toolbar), deleteComponent,  collapse(collapses the toolbar to hide it),  expand(opposite of collapse) |
| InfoMenu – shows information/toggles for a specific, selected component, e.g. can turn on/of a switch or battery, or change battery voltage. | selectComponent(takes in a component reference)  collapse,  expand |

**Interface**



**Success Criteria**

* *Simple circuits can be saved, loaded and executed, e.g. as above with a switch, bulb, battery*
* *A few simple components such as switches, bulbs, resistors, capacitors, LEDs, diodes, buttons, batteries*
* *Have a simple interface that can be understood by a normal person*
* *Identify what has gone wrong with a circuit – for learning*
* *Be able to search for components*
* *Be able to run on a variety of hardware*
* *Have no noticeable lag on calculations, have everything run smoothly and quickly even on low-end hardware.*
* *Have code look clean, make it modular and easy to add new components by subclassing the “Component” class.*
* *Have a simple window that explains how to use the program*
* *Have a uniform icon set for my program, don’t have wildly different icons.*

**Skills Needed**

*I feel that I am capable of completing the project, but may need to further learn the specific toolkit that I will be using, e.g. using tutorials to understand fully how to use layouts, slots and signals.*

* I Would need to advance my knowledge of the QT toolkit, such as signals, slots and layouts.
* I would need to refresh my knowledge of electricity calculations for the algorithm.
* I would need to create a linked list to act as the path for the circuit, so each circuit can call the next one, to create a sequence of current flows.