**To do:**

**Add DFD once done**

**Further documentation in other document not sat related**

**Mention git backups**

**Criterion**

**0- First 100 words**

My program will track and advise the CCC businesses composting processes. It will take inputs like soil values, date and location and record the data if they wish. After processing, the output of this program will then be the visualizations and a recommended optimal process for improving the soil. This will be used cross platform and be stored in the cloud (OneDrive). The GUI should be quick and easy to understand, and the program should be well documented incase changes are to be made. This program will improve the quality of the soil the business produces and the outcome for their customers.

**1- Gantt**

<https://www.teamgantt.com/>

**2- Data collection**

An interview is the most appropriate method of collecting primary source of data because an understanding of the goals of the owner is important in a complex project.

**In steps, how do you work out what to put on a farm?**

The amount of compost to put on a farm is calculated by taking in to account a lot of factors... what is the land manager trying to achieve...is it maximum profit… is it better soil?... what is the budget... what other fertiliser is being used...

For this program, it is trying to work out how heavy the compost can be applied before we get a nutrient overload.

We wish to experiment with adding the compost at higher rates, but don't want to overdo it.

**Who are the stakeholders of this software?**

Stakeholders are Camperdown compost, farmers adding compost and SESL doing soil tests

**How would you like the software to be designed?**

It is your job to design the software...we just need it to work

**How would you use a record of tests and jobs?**

We need to keep a record of all soil tests and all compost batches, so we can mix and match.

**How important is this software to be accessible on a phone?**

Phone access would be good but not essential

**Question analysis**

Reliability is the most important trait of the program required by the company

More time will be spent on errors and testing

Phone access is less important than previously thought and will not be included

Seeking feedback from the company will be important when designing the program

**New 100 words**

My program will advise the CCC businesses composting processes. It will take in soil and compost files, verify them and make a backup on OneDrive. The GUI output of this program will then be the visualizations and a recommendation of an optimal process for improving the soil. The GUI should be quick and easy to understand, and the program should be well documented and accessibility flexible incase changes are to be made. The most important feature as expressed by the company is the programs reliability. In making the program reliable testing and how the program handles errors is extremely important. This program will improve the quality of the soil the business produces and the outcome for their customers.

**3- Software Requirement Specification**

**Audience**

The SRS is used as documentation by programmers and users of the software

**Stakeholders**

Camperdown compost company is the main shareholder;

And those employed by it will be the exclusive users of the software.

**Environment**

Rich Client is the main platform used by shareholders and so is the most appropriate architecture

Furthermore the Windows operating system is the main software used and should be tested as such.

**Requirements**

The software needs to visualize and advise on soil nutrition

Functional requirements

Reliability

Non-Functional requirements

The program should be able to process corrupt files and well documented.

**Constraints**

File protection is the main constraint of the project. The files will use git version control and be stored locally and on GitHub and OneDrive

**New scope**

Phone access removed



Reliability is new priority

**4- Designs**

**Shared design elements:**

Sub menus:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Files |  | Settings |  | Help |  | | Save |  | Compound |  | Search... |  | | Open |  | Text size up |  |  |  | | Drop |  | Text size down |  |  |  | | Quit |  | Light/ Dark |  |  |  | |
| Most of these are self-explanatory |
| Save, open and compound will all open a file menu like so and compound files are xml files that have settings like ideal values and slider values other than compost. |
| Help will search documentation like this to find answers to problems.  A search might be, “file error”, which would show results below.   |  |  |  | | --- | --- | --- | | Help |  |  | | file error |  |  | |  | Fixing files (page 12) | | |  | All errors (Page 13) | | |  | File formats (page 3) | | |
| Drop file will open a window like so where files can be dragged onto to be uploaded.  Image result for drag and drop file  Sliders: |
|  |

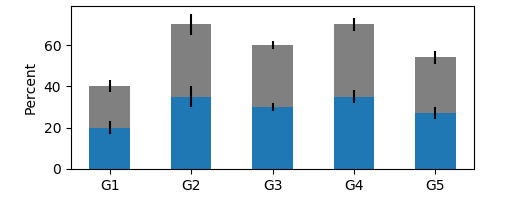
Moving these up and down change the amount of substance added to the soil which changes the nutrients in the soil and the graph.

Match:

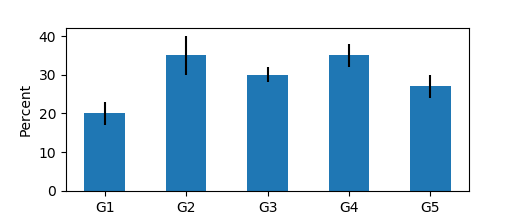
Grey is the ideal values for the soil and blue is the soil.

The function changes the sliders to get as close as possible to the ideal soil values.

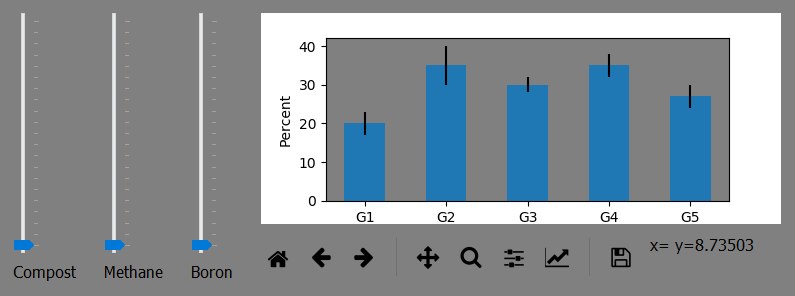
Before:



After match:



Light/Dark:



**Designs:**

Design 1 (Chosen):

Design one takes the elements from the weighted criteria matrix that score best

It’s a simpler design than the second that leaves more room for the graph, the main element.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Files | Settings | Help |  |  |  | - | ❐ | X |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Compost | Methane | Boron |  | Nitrogen | Phosphor | Potassium | Calcium | Sulphur |

Design 2:

\*Grey fill indicates buttons\*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Files | Settings | Help |  |  |  | - | ❐ | X |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  | Nitrogen | Phosphor | Potassium | Calcium | Sulphur |
|  |  |  |  |  |  |  |  |  |
| Compost | Methane | Boron |  | Match | Compound settings | Drop  Soil  files | Drop compost files |  |

**Weighted criteria matrix:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  | |  | |
|  | | |  | |  | |  | |  |  |  | **Stacked** | **File drop** |
| **Criteria** | | |  | | **Weight** | |  | | **Slide bars** | **Icons** | **Buttons** | **bar chart** | **out of menu** |
| Usability | | |  | | 5 | |  | | 3 | 1 | 1 | 1 | 1 |
| Performance | | | | | 3 | |  | | -3 | 0 | 0 | 0 | 0 |
| Aesthetics | | |  | | 2 | |  | | 0 | -3 | -4 | 0 | -3 |
| **Total** | | |  | |  | |  | | **6** | **-1** | **-3** | **5** | **-1** |

Notes:

Slider bars are practical, as they make changing values quick, even if they impact performance.

Icons help usability for initial users, but don’t look good aesthetically.

Buttons make the UI more upfront but take space away from the graph and look anaesthetics.

Having the file drop out of the menu takes space from the graph and is anaesthetics.

**Design analysis**

Our design focuses on practicality and usability of the end user