Ctrl-F to find

**Components**

**1 – Need or opportunity, 10 marks**

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.

**2 – Project plan, 10 marks**

**2.1 + 2.2**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Duration** | **Resources** | **Due** |
| Planning and research | 1 week | Company staff | - |
| Design | 2 weeks | - | - |
| Programming | rest | - | September |

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

**3 - Analysis, 20 marks**

**3.2.1**

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

**3.2.2**

**Analysis**

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

**3.2.3**

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.

**Purpose and Audience**

The SRS is used as documentation for clarification 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.

Files will be backed up and shared on one network, but root will be on each computer.

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 -

Graph and store nutrient values for soil and compost.

Non-Functional requirements -

The program should be reliable, fast and well documented.

**Constraints**

There is little constraint on this project.

The files will use git version control and be stored locally, in GitHub and OneDrive

**New scope**



The program will graph and store nutrient values for soil and compost reliably.

**4 - Designs 10 marks**

**4.1**

Sub menus:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | Files |  | Settings | | |  | Help |  | | Save |  | Text size up | | |  | Search... |  | | Open |  | Text size down | | |  |  |  | | Drop |  | Light/ Dark | | |  |  |  | | Quit |  |  |  |  | | |
| Most of these are self-explanatory |
| Save, open and compound will all open a file menu like so and compound files are Excel 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.    Sliders: |
|  |

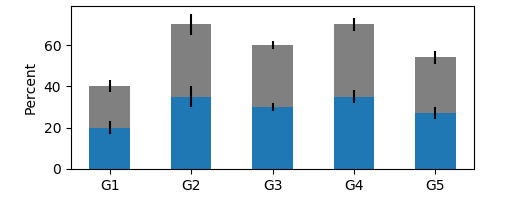
Moving these up and down change the amount of substance added to the soil which changes the nutrients in the soil and the graph.

Match:

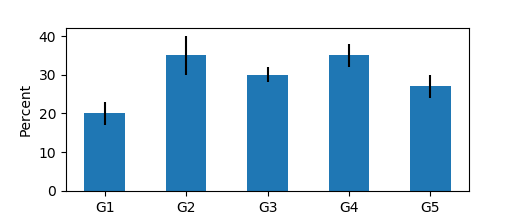
Grey is the ideal values for the soil and blue is the soil.

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

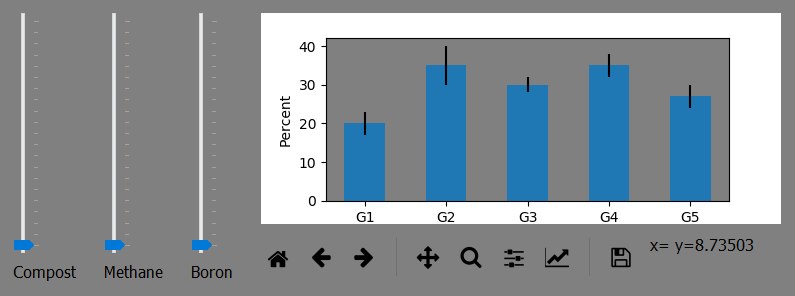
Before:



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 |  |

**4.2.1 + 4.2.2**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | |  | |  | |
|  | | |  | |  | |  | |  |  |  | **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** |

**4.3**

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 and take up space.

Buttons make the UI more upfront but take space away from the graph and look un-aesthetic.

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

Design analysis

The design focuses on practicality and usability of the end user

**4 – Development of the software, 30 marks**

**4.1**

Git and OneDrive were implemented as mentioned,

**4.2**

Excels are used for date storage.

**4.3 – 4.5**

This involved the UI, data and logic. A note on the Data validation, rather than testing for any errors the program simply reads 0 for the file if it can not extract data and has no range checking. This makes the program simpler and means it is more reliable.

**4.6**

Testing solution.

**4.7**

Testing usability:

Functional requirements like graphing and recommendations will obviously be tested, as well as non-functional requirements like performance and reliability.

Tests will be carried out on CCC office employees, those most likely to use the software.

Test metrics will include time to decide and how many mistakes they made in navigating the UI.

The following list of questions will also be used:

How would you rate this software’s usability?

Is there any way you could see it improved?

Does it complete its function?

Is there anything you would change to make it more functional?

Success will involve the user completing the function with minimal intervention.

**4.8**

User 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Item Tested** | **Instructions** | **Success** | **Comments** |
| Opening files | Select files from open menu | Yes | 5 sec |
| Deletion of sliders | Delete a slider | Yes | 15 sec, tried to press delete button instead of left click |
| Use of graph and functions | Find what mix of compost to use with the right-click functions. | Yes | 40 sec |
| Changing settings | Increase the text size and change to dark-mode. | Yes | 10 sec |
| Use the Drag-Drop open | Use the Drag-Drop open | Yes | 50 sec |
| Open the help menu | Open the help menu | Yes | 4 sec |
| Quit the program | Quit the program | Yes | 3 sec |

**How would you rate this software’s usability?**

The software is reasonably easy to use.

**Is there any way you could see it improved?**

I think it needs a manual, or at least a clear drop-down help menu. Perhaps it could become more intuitive by being able to easier load new soils and compost samples. Labelling the soil would also be good.

**Does it complete its function?**

The primary function is to ensure that we do not apply too much compost and have difficulty with excess nutrients. The program can quickly calculate which nutrient runs into excess first, and at what rate per hectare this occurs. It can compare different composts, so we can pick the most appropriate compost for different paddocks. So yes, it does perform its function

**Is there anything you would change to make it more functional?**

The program performs its function. To add more functionality, we would need to increase the scope of the program.

User 2

|  |  |  |  |
| --- | --- | --- | --- |
| **Item Tested** | **Instructions** | **Success** | **Comments** |
| Opening files | Select files from open menu | Yes | 15 sec, had to explain file system. |
| Deletion of sliders | Delete a slider | Yes | 5 sec, right clicked before left. |
| Use of graph and functions | Find what mix of compost to use with the right-click functions. | Yes | 120 sec |
| Changing settings | Increase the text size and change to dark-mode. | Yes | 15 sec |
| Use the Drag-Drop open | Use the Drag-Drop open | No | Had never used drag drop before. |
| Open the help menu | Open the help menu | Yes | 5 sec |
| Quit the program | Quit the program | yes | 3 sec |

**How would you rate this software’s usability?**

8/10

**Is there any way you could see it improved?**

No

**Does it complete its function?**

Yes

**Is there anything you would change to make it more functional?**

Being able to create new soil test after compost has been applied.

**4.9 + 4.10+4.1**

100-200-word report

It’s clear from the interview that the software the software is usable and functional. Usability could be improved by the ideas mentioned but being able to create new soil tests is outside the scope. Other ideas like labelling the soil and ideal graphs and more user-oriented documentation in the help menu will be implemented. Both participants who took the usability test were advocates for manuals and not familiar with designing software intuitively, so it is important to have accessible documentation if many employees are in this habit. At times in testing, a participant lack understanding of what the software was supposed to do was the biggest barrier to using it software and should be explained in documentation. On a slide note, we also discussed other variables that could impact how accurate the program was, like depth and nutrient runoff, but the consensus was that this program was a major improvement over the previous method

**4.12**

To evaluate the solution, previous actions by the company will be compared with what they decide with the program.

**4.13**

Report on solution requirements

**Functional requirements -**

**Graph and store nutrient values for soil and compost.**

**Non-Functional requirements -**

**The program should be reliable, fast and well documented.**

**5 – Project plan assessment, 10 marks**

**5.1**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Adjustments** | **Reason(s)** | **Impact on timeline** |
| 15th May 2019 | Extended design time | Low marks | 5 days |
| 6th July 2019 | Compound settings in merging into open | Simplifies the design, as previously these files were going to include may regions (e.g. high and low) that be hard to make | None |
| 18th July 2019 | Help will open documentation file rather than search in application | Qyqt5 doesn’t support search bars in that configuration | Minimal |
| 19th July 2019 | Toolbar introduced | Usability of the graph | Minimal |
| 10th August 2019 | Extended refinement | Bugs | 7 days |
| 19th August 2019 | Help will open files directly and Soil and Ideal files will be labelled | Useability testing | Minimal |

**5.2**

The project plan was a good outlier for what the project should achieve and allowed for adaption for changes with limiting factors such as libraries and performance.

The costly changes to the timeline were related to unpolished products when finishing a component. The design of the program wasn’t presented in a way that could be used in documentation and the program at the end of refinement still had many bugs. These both had to be revisited and revised

Other changes to the project were influenced by different libraries and their limitations, Matplotlib for example needed a toolbar to support zoom and Pyqt5 didn’t support search bars in menu bars. Although switching libraries was possible, in both cases there was too much invested.

Finally, Design oversights had to be overcome. In the initial plan for the design ideal files would be much more complicated, but after an understanding that these wouldn’t be used to this extent they could be simplified to work in the same format as soil files. Another oversight was the lack of labels for the soil values.