A Data Science tool even your grandma could use

A project designed for the University of London BSc CM 2020, Agile Software Projects.

Authors:
T S Sheager – 200375858 – tss40
B Krengel - 200494300
J D Gomez - 200391342
K L Mwanza - 200154390 - klm13
C King – 200492683 – ck170

Foreword

This proposed project is a Data Science tool that makes generating graphics easy for even the most basic computer user. The project draws comparisons from real people, interviewed to determine what the best method of making this product as user friendly and useful as possible. The project is not in production yet and all development will occur after this stage of the project timeline. All information on the project is stored on GitHub (https://github.com/benekreng/agile)

Contents page

1.0 - Introduction

1. **Project overview.** In today's world, we have the ability to search for information about any subject or matter within seconds. We are often overwhelmed by the sheer number of search results we are provided and this in itself poses an interesting challenge. Information and data are often skewed for the purpose of spreading false news and misinformation[1] which makes it increasingly difficult for individuals to verify the data to which they may be exposed. Additionally, in many places of work and education, we are given datasheets of figures and statistics which can sometimes prove difficult to spot trends or to represent the given data in an easy to grasp graphical format.

The core concept underpinning this project is to provide basic computer users with a simple to use, streamlined, method of visualising a given dataset. This will allow them to formulate their own viewpoint by seeing patterns, trends and other correlations without needing knowledge of complex Data Science software.

Historically, when an individual is given a dataset that they would like to analyse visually, they would open it up in a spreadsheet application like Microsoft Excel, sort the data and fix any anomalies with the data import, select data ranges then finally, use the charting tool to visualise the data. While this may be relatively easy for many proficient spreadsheet users, others may find the task arduous and may stumble at one or more of the steps along the way. The tool that is being proposed will be targeted towards users that do not have a sufficient level of skill to use programs like Excel and will process and display the information with minimal input from the user.

2. **Market Research.** Currently there exists no Data Science tool that has the proposed 'plug and play' functionality with a data set. The market competitors in the Data science field all require some degree of knowledge to utilise the system properly. These are listed below

   a. **Microsoft Excel.** A spreadsheet and data management tool created by Microsoft[2]

   b. **Google Sheets.** A spreadsheet and data management tool created by Google[3]

   c. **Matplotlib.** A Data science tool that requires Python code to be entered to create graphics[4]

   d. **Seaborn.** A Data science tool built on the foundations of Matplotlib that requires Python code to be entered to create graphics.[5]

   e. **Jupyter Notebooks.** A software GUI that users multiple coding languages to process and display data[6].

Additional reading was conducted surrounding the project. All sources are listed within this reports bibliography.

[1] https://www.bbc.co.uk/news/topics/cjxv13v27dyt
[2] https://www.microsoft.com/en-us/microsoft-365/excel
[3] https://www.google.co.uk/sheets/about/
[4] https://matplotlib.org/
[5] https://seaborn.pydata.org/
[6] https://jupyter.org/

3. **Stakeholders.** The stakeholders for this project can be classified relating to the level of computer competence they have. The stakeholders are outlined below:

   a. **Basic computer users.** A basic computer user would be classified as someone who has little to no experience utilising IT systems. Examples of basic users include the elderly and children in elementary phases of education. Due to the ethical implications of working with children the project is biased towards adult basic computer users as the main stakeholder

   b. **Moderate computer users.** Moderate computer users can be considered to be competent in most users systems and would typically be considered office workers or newer generations who have grown up surrounding by technology. These users will be able to manipulate computer systems when presented as a GUI but would likely struggle if asked to use a command line interface

   c. **Advanced computer users.** Advanced computer users have an above average knowledge of computer systems and are able to access functionality that only developers or software engineers would understand. Computer science students, Network administrators and IT professionals would all be considerd as advanced computer users.

The proposed data science tool may be utilised by all stakeholders but it is primarily targeted towards basic computers like the elderly or children. Children are out with the bounds of this project as there would need to be careful ethical safeguarding to work with children. If this project had more time and resources working with children would be something that would be considered as an additional requirement above the minimum viable product (MVP).

It is also out with the bounds of this project but if given more time the system could be implemented to utilise an API to interface programmatically with other pieces of software. This would cater to the advanced computer users and potentially some moderate computer users. Allowing modifications on the software itself would also cater for the moderate computer users as it would give them more functionality to manipulate the graphics they have on the screen. Both would be considered additional requirements above the MVP.

4. **Scope of report.** This report will look at all of the user requirements, project design elements, and specifications needed to enable us to start the building and delivery of a successful MVP of the outlined project. When discussing the user requirements, key stakeholders (Basic computer users) and any additional stakeholders of our tool will be consulted and relevant feedback will be assessed and included in the project delivery. The project management tools and processes that have been used, and are expected to be utilised, will be reported on - focusing on why certain systems have been chosen over other tools and services. The model design will be discussed and all tiers of prototypes that explore the design and functionality of the software will be included. Functional specifications and the technical architecture will highlight all of the core functionality and determine the MVP. Finally, The ethics and legalities of the project will be examined.

5. **Limitations.** As mentioned previously this project will only focus on basic computer users. It will be hosted on a live web link but this will be hosted externally and the team will not be responsible for production or maintenance of the web service. In the event of a network failure the system will be able to be ran as a standalone file directory. The system will use JavaScript CSS and HTML to provide the service to the user. As such the user will need to have a modern web browser that is able

to run JavaScript installed on their computer. Installation of required software and navigating the host operating system are not features of this report and are expected as pre-requisite items for utilising this software. This project was created under a fixed and rapidly closing timeline. As such certain elements would not be given the same attention if they were produced commercially. This project will utilise any remaining time to add features that go beyond the MVP. The MVP is the baseline for the project's success.

2.0 - Project management

6. **Team structure.** Early in the project the team began to distribute roles to ensure the team's skillset was accurately distributed. Each role comes with specific responsibilities. These roles and responsibilities are outlined below.

    a. **Connor.** Connor was given the Team Leader (TL) position and would be overall responsible for the project's timeline and ensuring all key markers are met with the project's development. This role was important as the team needs one collective focus point to ensure the project is moving in the same direction across all areas.

    b. **Kashka.** Kashka was given the Chief Technology Officer (CTO) role and would be responsible for ensuring all technology utilised in the project is implemented correctly. This was important as the system is using multiple libraries for its functionality. Without such a role it would be easy to diverge from the intent of using such technologies.

    c. **David.** David was given the Chief back-end developer (CBD) role and would be responsible for ensuring all the libraries and back-end code interfaces correctly. This role was important as there needs to be a single point of contact for all of the "smart" features of the project.

    d. **Benedikt.** Ben was given the Chief front-end developer (CFD) role and would be responsible for ensuring all front-end development meets the specified designs. This role is important as there needs to be a single point of contact for the design so it does not diverge.

    e. **Thavelarn.** Thavelarn would be the Chief Requirements Officer (CRO) and would be responsible for ensuring the project meets the specified user and accessibility requirements of the project. All requirements are important and without a CRO it is very easy to introduce "market creep" where the requirements grow and grow. Commonly referred to as the "good ideas club" the CRO ensures the project stays within its scope

7. **Timeline Management.** The timeline was managed through a continuously updated Microsoft Excel spreadsheet acting as a "Gannt chart" style of timeline. A screenshot of this timeline is listed at Figure 3.0. This was important as all team members need a clear outline of the milestones that stand ahead in the project.

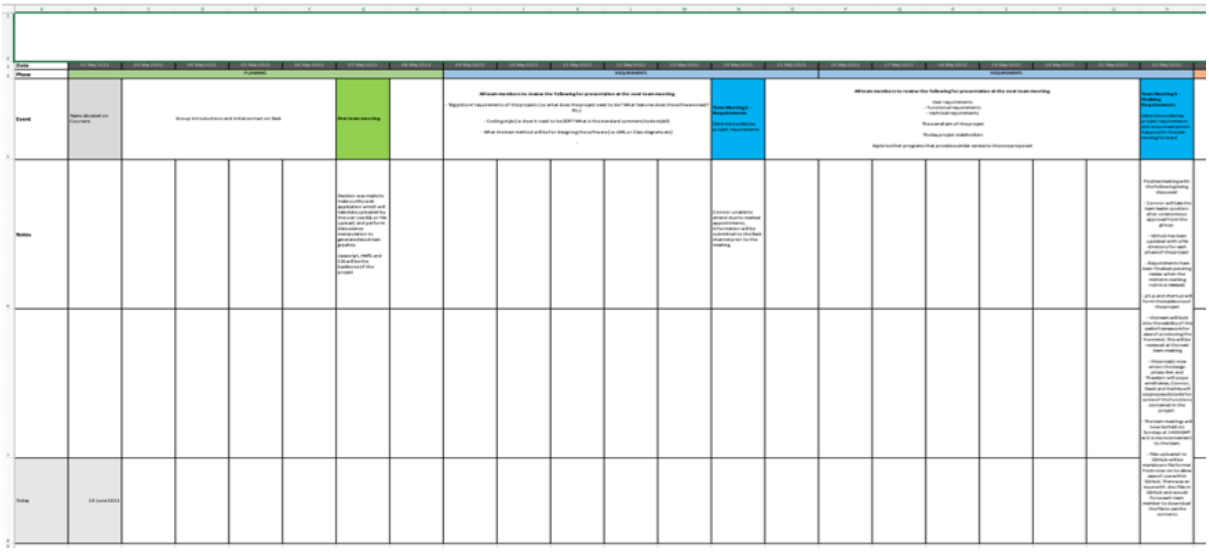


Figure 2.0 – Project timeline snapshot

The timeline has been broken down into phases, each phase is outlined below:

a. **Planning.** During the planning phase the team bounded ideas and determined the overall project aim. During this phase it emerged that there exists no entry level tool for Data Science. Comparisons were drawn with market competitors (Excel and Matplotlib) and both require a basic level of understanding to utilise the software. This phase was important as it gave the initial aim and scope of the project

b. **Requirements.** During the requirements phase the team conducted several information gathering exercises. Informal interviews and seminars with friends and family were used to determine the user requirements of the project. This requirements phase was important as it narrowed the scope of the project

c. **Design.** During the design phase the team created multiple iterations of prototypes, starting from low fidelity. After creating the basic design of the system these designs were again passed to friends and families in informal interviews and their thoughts were described. Utilising the feedback from the interviews medium and high fidelity prototypes were created. This phase is important as it will provide the framework to use in the development stage

d. **Proposal Draft.** The proposal draft is current stage of the project.

e. **Proposal Review.** The proposal review is where all sections of the proposal are finalised and combined.

f. **Midterm submission & Midterm proposal deadline.** This is where the group will submit the project proposal individually.

g. **Sprints 1-7.** This part of the project is the bulk of the development. Agile management has been selected as it offers the most flexibility with development. 7 sprints have been created and each sprint will have a clear goal at the end of it. This goal may be altered mid sprint if something of higher importance comes up mid sprint. Through all sprints each team member will be given clear tasks that suit their skillset best. As well as developing the software the program will be tested periodically with the users it is being designed for. Alpha prototypes will be distributed under limited release until the program is sufficient to be

upgraded to a Beta version. At this point it will be classed as the Minimum Viable Product (MVP) and will have all the basic functionality required. If this is completed with time to spare additional features will be added to the software in the last sprint phases.

h. **Testing & Evaluation (T&E).** Once the development phase is complete the program will be distributed to users for feedback. The users will conduct a series of black box testing as this is the target market for the program. Users who have knowledge of the workings of the system are beyond the scope of this project. Once testing is complete the user's feedback will be analysed. The T&E period spans 2 weeks and can be augmented to implement any user feedback into the main programs execution. In effect this becomes a contingency for the development and can be utilised as an 8th Agile sprint.

i. **Final Analysis.** Once all T&E is complete the project will be critically analysed by the team. Any errors will be highlighted and the system will be presented in such a manner that another software development team would be able to utilise the program and continue it's momentum if required. Additional steps as well as all previous steps covered will be clearly annotated.

j. **Project submission + Project submission deadline.** This is where all information on the project will be submitted for assessment by the University of London staff.

8. **Team communications.** The team had regular communication with each other through Slack and Google Meet conferencing. As the team is geographically split across a large portion of the planet it was essential that we deconflicted a time that suited us all. Initially the team tried meeting on Saturday 7 May 22 at ~1600GMT. This proved to be difficult for some team members due to the vast time difference. It was decided that Sunday at 1400GMT was an appropriate time for all team members. This was then utilised every week as the standard meeting time. During these video conferences the team discussed the work they had completed throughout the week and appropriate decisions were made through team voting. Tasks were also distributed throughout the team during these meetings. The team would also discuss what to expect in the coming weeks and would plan contingency periods for each phase of development.

Slack was used to great effect when the team were not chatting in the Google Meet video conferences. Messages would be left on the Slack channel for any member of the team to answer. This proved to be effective as all team members could read the message thread and gain situational awareness of all areas of project development. This was also a good medium to post links and other information discussed during the main meeting for reference later. One of the main reasons this worked so well was because this allowed all team members to access the material whenever it was convenient in their own time zone.

The team meetings were not always convenient for each team member and as such there needed to be a system to allow team members to catch up on what was missed. The team decided to create meeting logs that outlined broadly what the content of the meeting was. The best method of doing this was by creating a file that would be stored on the GitHub repo. Initially this was done in a .docx file format but this proved to be troublesome for the team to view the content as it would require a full download. The team then decided to use .md for all meeting logs and files on the GitHub. This would allow native viewing within the web browser and would increase the team's efficiency. All meeting logs have been stored in the GitHub repo.

9. **Team collaboration.** The team decided a common, multi-developer, platform was needed for creating communal documents like this report. As such Google Docs was used as it allows multiple creators to work on the same document without compromising another team members work.

The team decided early on that roles needed to be created to ensure the project stayed on timeline and each task was allocated to the appropriate individual. This was in part down to a strong first team meeting where all team members were friendly towards each other and all came well prepared for the problem at hand. This was quickly integrated into a 'battle plan' for how to move ahed with the project. As all team members were given clear bounds and timelines in which to achieve goals it resulted in little to no friction between team members. Any issues were resolved quickly within the Slack channel and all team members provide a strong input to the team. This has resulted in a good cohesive team.

One things that has been a particular highlight is the team utilising the strengths of the other team members. Early in the requirements phase the team were discussing ways to host the service on a live web link. David showed the team a previous project of his where he hosted his work on a live web link. By demonstrating this to the team it A) showed it was possible within the timeframe but also B) at little to no cost to the team. This would have not been possible, or at minimum time consuming if David had not done this. Other instances like this occurred with other team members and it significantly sped up elements of the project's timeline.

10. **Version control systems.** Early on in the project it was decided that GitHub would be used to control all the documents of the project. This ranged from meeting logs through to the source code for the main project. GitHub was selected as it acts as a central storage location for all team members to use. Ben was tasked with creating the Repo and then gave all team members access. Similar to the Slack channel this worked very well as it did not rely on time zone for stable usage. Often team members would wake up to work that their teammates had produced the night before. The GitHub is still used for all the core files.

Some team members were not as confident with the system as others so there were initially some teething issues making sure everyone's files were correctly uploaded for key markers of the project (designs being uploaded etc). Once corrected this was of little to no significance as the project progressed.

3.0 - Model design

11. **Initial concept.** During one of the first meetings the team got together and discussed individual brainstorm ideas. The team came to the conclusion that we should make an application that can be used to visualise datasets in an easy and straightforward manner. We chose to make this application a browser application, to ensure all devices would be able to use the software regardless of operating system. Upon conducting market research into the Data Science field, the team discovered that no tool like the one proposed exists for a basic computer user. The only products that bear the closest resemblance to the proposed project were a few complex APIs and long standing, steep learning curve, commercial software like Google Sheets. As there appeared to be a genuine hole in the market please the team decided this would be an appropriate project to pursue.

12. **Further concept analysis.** After the team had established the core concept behind the project it was narrowed down and a target user of this application was determined. The team first looked at what the particular purpose the application might serve best. The best use case was found to be that the software could be used quickly to visualise a dataset for a research application or a general quick glance of the data. This has utility for all the project stakeholders identified but would be targeted more towards basic computer users. Bespoke products for the other stakeholders are beyond the scope of this project. The team decided to focus on its ease of use, low requirements of computer knowledge and low technical requirements as, again, this where was the hole in the market appeared during the market research.

13. **Data gathering.** To prove the initial design theory the team set up a trial session that would be used to confirm the project was viable for the targeted stakeholder. The data gathering process that was undertaken involved a range of exercises and interviews with potential users of the application. In order to gain a better understanding of what users would like the application to do, understanding the end user requirement was crucial. Equally important was developing a successful benchmark to compare the MVP against current market competitors. To judge the final product metrics must be compared.

The two main ways in which meaningful user requirements and metrics were collected were by using focused interviews and by observations of users performing a given task.

a. **Focused interviews.** Focused interviews are one of the ways in which qualitative research can be carried out. The reason for this approach was to gain meaningful information from participants about their individual requirements in order to see where some areas of concern are and how to tailor the application to fit their given need.

b. **Observing user's performance.** Giving participants a monitored and timed exercise adds an element of quantitative research and allows for measurable goals to be set before developing the application. This ensures that the most time staking and tricky elements of data presentation are addressed correctly.

14. **Participants.** Six participants took part in the interviews and timed exercise, all of whom would be potential target users of the application. All stakeholders were covered by the participant audience. All participants agreed for their responses and information related to the exercises they carried out to be shared, although names and personal information will not be part of the analysis or report. All data was complaint with the UK GDPR legislation. The time spent with each participant

ranged from 15 minutes to 30 minutes. 4 sessions were conducted online using Zoom, and 2 sessions were in person.

15. **Interview and activity.** The sessions were split into 3 sections with each section lasting 5 to 10 minutes each. Firstly, there was an introduction where participants were told about the task they would be performing and about their technical literacy. This process involved open-ended questions about how well they thought they would do and their history of completing similar activities. Secondly, each participant was asked to complete the following activity:

Task	Notes
1. Download a small CSV file that was sent via email	The CSV file contained data about fruits sold on three different days of the week
2. Open the CSV file in their chosen spreadsheet tool.	Microsoft Excel and Google Sheets were the two used
3. Create two charts from the given data, a bar chart and a line chart	

Figure 3.0 – Data gathering exercise user tasks

Once the participant had completed the task (to the best of their abilities) there was another discussion about the task and about the elements and actions that they would like to see simplified.

16. **Session results.** The first part of the session that was conducted with the participants gave meaningful insight into the overall capabilities of each person. Responses ranged from students studying fashion who had little exposure to analytics, to departmental managers who were very familiar with spreadsheets and charting tools. One of the recurring themes when speaking with some of the participants, was the lack of charting tools when using mobile devices, especially mobile phones. When the exercise was discussed, 5 of the 6 participants were confident they would be able to complete the taste and were generally happy with their technical literacy.

The data related to the timed activity had to be truncated and a limit of 450 seconds was implemented, this was in the interest of time and to allow for further discussions in the third section.

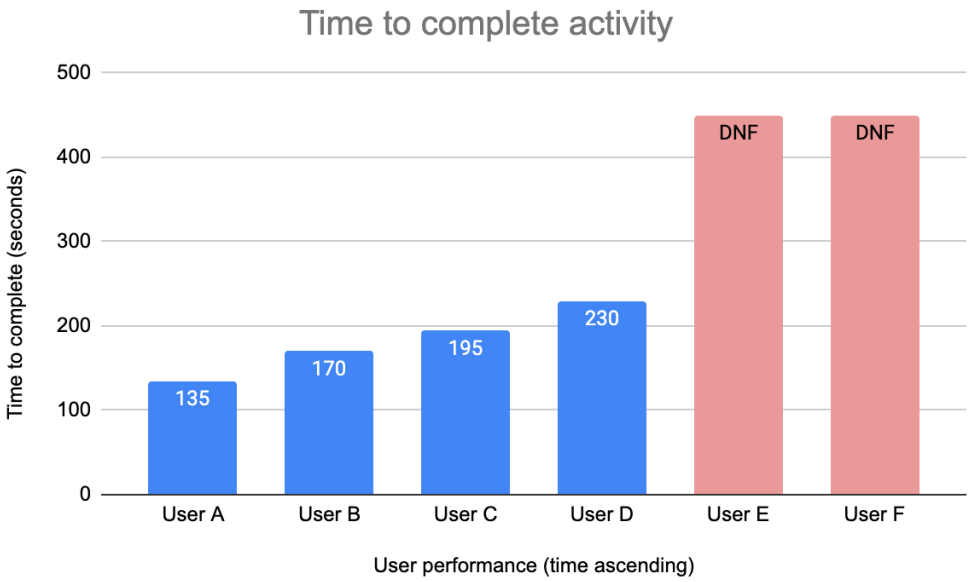


Figure 3.1 – User performance in data gathering exercise

The graph above shows the times each participant (user) took to complete the activity in ascending order. Times were rounded to the nearest 5 second interval and the two participants who did not complete the task within the 450 second marker are highlighted as DNF. Figure 3.1 shows that most participants are able to complete the activities in 230 seconds or less, additionally, the average time taken to complete the activity (excluding DNFs) was just 182.5 seconds (or just over 3 minutes). With this data the team gathered clear metrics which would be referred to throughout the future development phases. Although some users did not manage to finish the tasks within the given timeframe, this is also a metric that is valuable, as the development will be geared to ensure that the vast majority (if not all) participants will be able to generate charts from a given data set.

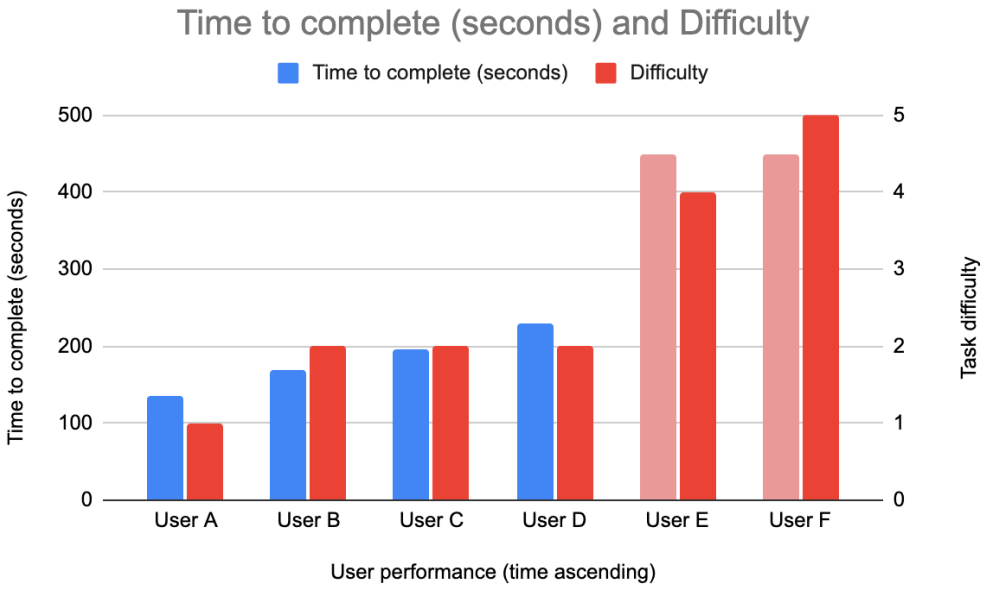


Figure 3.2 – User performance with difficulty in data gathering exercise

Finally there was discussion about how difficult the participant thought the activity was. Figure 3.2 displays this relation between task difficulty and time taken to complete the activity. Task difficulty ranged from 1 (very easy), to 5 (very difficult). Users who did complete the task gave a difficulty rating of either 1 or 2, whilst the participants who did not complete the task either gave a difficulty rating of 4 or 5. Again, this gives a further weight behind the teams' decision to target basic computer users. Simplifying an activity for individuals who would normally find it difficult would be important and a metric to measure of success later in the projects lifecycle.

Overall the post activity discussion was beneficial as it highlighted the importance of not only speeding up the data representation process for competent users of a spreadsheet system or tool (users A – D in Figure 3.2), but more so, for users who are not as literate with the tools offered to represent data in a meaningful way. A recurring theme during discussions was the importance of being able to export the generated charts quickly and easily so they could be printed or pasted into other applications, all respondents believed that any new application or tool should be able to perform this task.

17. **Product design analysis.** Coupled with the previous data gathering activity the team sought to further analyse and refine their user requirements. This was achieved by each member of the group conducting several interviews with a friends and family.

The questions posed in these informal interviews were:

Question	Answer	Example
"Would you use an online data visualisation application?"	Who? What Profession?	Student, Scientist
"What would be your specific use case?"	General use case in their field.	Student Essay, Research Project
"What device would you use the application on?"	Type of device	Smartphone, Tablet, Laptop
"What should be the basic features"	Features of application	Export, Type of graph, range of values
"Would you treat functionality for ease of use?"	Yes/No? Reasoning?	Yes otherwise I would use a different application No

Figure 3.3 – Interview questions for data gathering

After collecting the results from all interviews the team discussed and collated data.
The team first of all set the technical requirements of the user. For this it was important to consider the main stakeholders of the project again. The data showed that, Students, Colleges, Journalists, small researchers or scientists would be using the application. This made it very clear that all users will benefit from being able to access the application on a large range of devices – I.E. phones, tablets and computers. This agrees with the team's initial hypothesis of ensuring ease of use which initially led the project to be a web application.

According to the data gathered the requirement for an easy user interface in our application was a split decision. Approximately half of the people in the surveys would trade functionality for ease of use. After some discussion the team decided that, although there is correlation between less functionality and ease of use, it doesn't mean that we can't implement the best of both worlds. As such the team decided to try and balance a good collection of features with options with options for usability.

Considering our main stakeholder, the application should guide you through the process of using it - minimising the users confusion and maximising confidence for overall better productivity whilst still providing utility for all stakeholders.

Next the team looked at what our user's needs for functionality were. The majority of suggestions were including drag and drop items, to import the data, and an feature to export any graphics generated. Another suggestion was an option to be able to display a specific range of rows or columns of the dataset.

In conclusion the system was designed with these main aims in mind: Usability / Ease of Use and Low User Requirements.

18. **User interaction.** After our requirements investigation the team investigated what the usage of this application would look like. A student might research the suitability of usage of drones for delivery companies. For this the student might find a dataset about unidentified unmanned aircraft. He uses a laptop for his research from where he downloads the dataset from his browser in his download folder. He then finds that the dataset is hard to read, so he looks up something like "visualise dataset online free". Upon his great surprise he finds our website where it tells him to drag and drop or select the dataset file he wants to visualise. He drags his file from the file browser onto the marked area on the website. The website guides him to step two from one which gives him confidence as he sees that the next step is the export step. He then proceeds to set his preferred parameters for the visualisation. He decides to use a line graph which he finds under the options of 'type of charts'. He then sees his chart of which the timeline goes all the way from 1980 to 2022 although he wants to focus the data by shortening the timeline. After a quick glance he sees that he can select a particular range of rows or columns. He is pleased with what the graph looks like now and will, after clicking the 'next' button to move on to the last third step, now export the graph as either a pdf or jpeg by clicking either of those buttons. There is no confusion as it is made clear by symbols that either of these buttons should be pressed to export the graph. The download will begin and he will be able to paste the jpeg into his essay. This process is described graphically in Figure 3.4

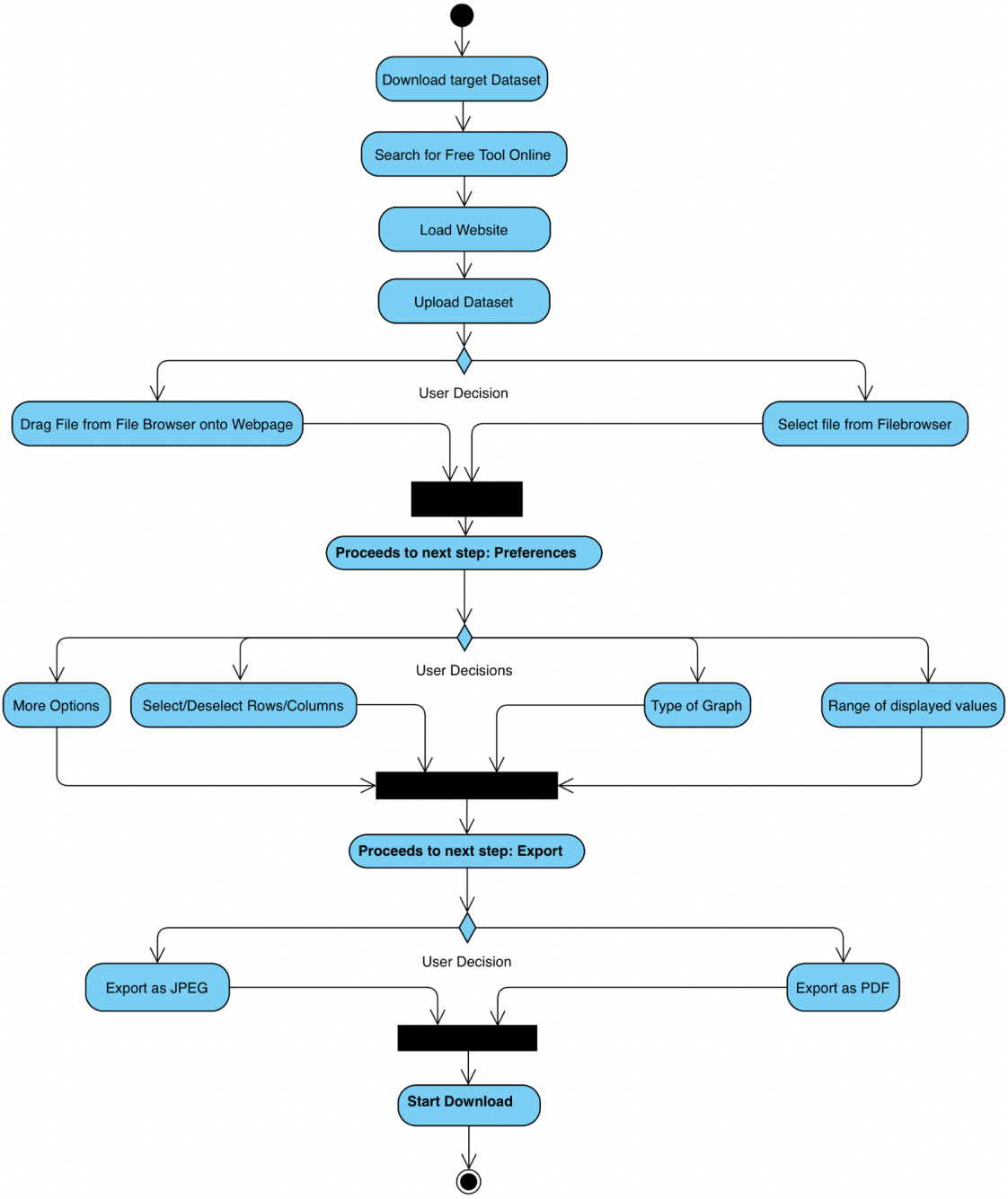


Figure 3.4 – Sample user interaction flow

The specific use case described is only one of multiple. Other applications can be research for a journalist for example who might not want to use the graph in an article but refer to trends he might see on a dataset he imported.

4.0 - Functional Specification

19. **Initial functionality.** Utilising the data gathered previously the team decided several features that will be pursued in production. The project will develop a browser-based piece of software that will allow users to import their own data or utilise pre-built data if needed, the software will handle any data issues without causing the main application to fail. Then it will provide Data Science visualisations and statistical analysis for the data supplied all while giving the users options to modify the data, type of graph, and change the range of values to be used.

Build upon the project model design, the core functional specification elements are defined and described as follows:

a. **Upload of data.** The user will be prompted to upload their own data in a .CSV format or, time permitting, a Google big Query in a simple non-error-prone interface. If the user does not load their own data, there will be a series of pre-loaded datasets for them to understand and get to know the capabilities of the application.

b. **Error management.** The application will handle omissions, erroneous data types, and other errors that would normally cause the software to crash inside the user's submitted data file. The programme will notify the user of any mistakes and advise the user of the consequence of these difficulties - for example, three lines of data were corrupted and were omitted.

c. **Graphic Options Choice.** Upon dataset upload and cleaning the user will have the options to change aspects of the dataset all while having the option to see the graph on real-time. Some of these options include select/deselect row and columns; change the type of graph, change the desired range. Mathematical functions will also be available for datasets in which they can be used I.E. mean, average etc.

d. **Graphic & Data Export.** When users determine that the graph generated by the application meets their needs, they will be able to download both the graph as a .PNG/.JPEG/.JPG file and the dataset that was used to construct it.

5.0 - Technical Architecture

20. **Technology.** The team now faced the situation would need to be built with custom or ready built technology. The team discovered that there are multiple well tested and powerful libraries to aid in generating charts. These are outlined below

    a. **Charts.js.** Charts.js is a JavaScript library used to generate crisp, user interactable charts.[7]

    b. **P5.js**. P5.js is a multifunction library used for multiple purposes. It is a library that all the team's developers have worked with previously.[8]

    c. **Native HTML, CSS, JS.**[9]

21. **Technology considerations.**

    a. **Reactive webpages.** As the data gathering exercise outlined that all users would prefer to use the system on multiple devices the team decided that the web pages need to be reactive

    b. **Live website.** To ensure all users have the same version of the product the team decided that it will need to be hosted on a live web URL. This means any time the system is used over the internet it is the latest production edition available. Where internet connections are unreliable a standalone file may be provided.

    c. **Built in functionality.** The team decided that all built in functionality within charts.js should be utilised. This was for several reasons but speed of production and knowing the code has been robustly scrutinised by the community that uses the charts.js library adds a layer of pre-emptive stability to the code base of the application.

    d. **Initial file types.** To ensure the MVP is attained throughout the program certain functionality will be released in later versions of the software. For example the system will initially only load external data as a .csv file and Google Big Query data will be implemented later. Likewise exported graphs will initially only be a single file type like .png until all other MVP items have been included.

22. **Cyber security.** As the user is importing data to the program there runs a risk of malicious code execution. As the system will only be downloaded to the computer and ran locally it should not be possible to be used a an attack vector. If the program were to be developed into an API it would need hardened connections to ensure the usage of the program isn't exploited.

23. **Technology choice.** The team decided to use charts.js as the main library and then review P5.js in due course if functionality from P5.js is required. P5.js does not offer the same degree of simplistic user interface that charts.js does. In addition to this utilising charts.js over native HTML coding is preferred as it will speed up development time as well as increasing user satisfaction with the application. Charts.js was also used in trail programs by multiple members of the team and they preferd working with charts.js over p5.js.

[7] https://www.chartjs.org/
[8] https://p5js.org/
[9] https://www.javascript.com/

6.0 – Prototypes

24. **Low fidelity prototypes.** After deciding the main features and technology to be used the team were tasked with designing low fidelity prototypes of application. These prototypes are outlined in Figures 6.0 - 6.6.

The main idea behind Figure 6.0 – 6.2 is to have a progress-based input menu for the user where the user selects how and what the data representation is going to be one after another. Thinking about the main stakeholder involved in this project these designs offer a simple but effective user interface that guides the user through the process of using the application

Figures 6.3 – 6.5 experiment with what the application could offer in terms of functionality and layout like the side menu, hot keys, and legends. They also explore the idea of the web application supporting mobile use and responsive layout in general.

Figure 6.6 is simply suggesting that the data representation is emphasised in the layout as much as possible. In other words, make it as big as possible. It also considers the idea of only having a collapsible bottom menu.

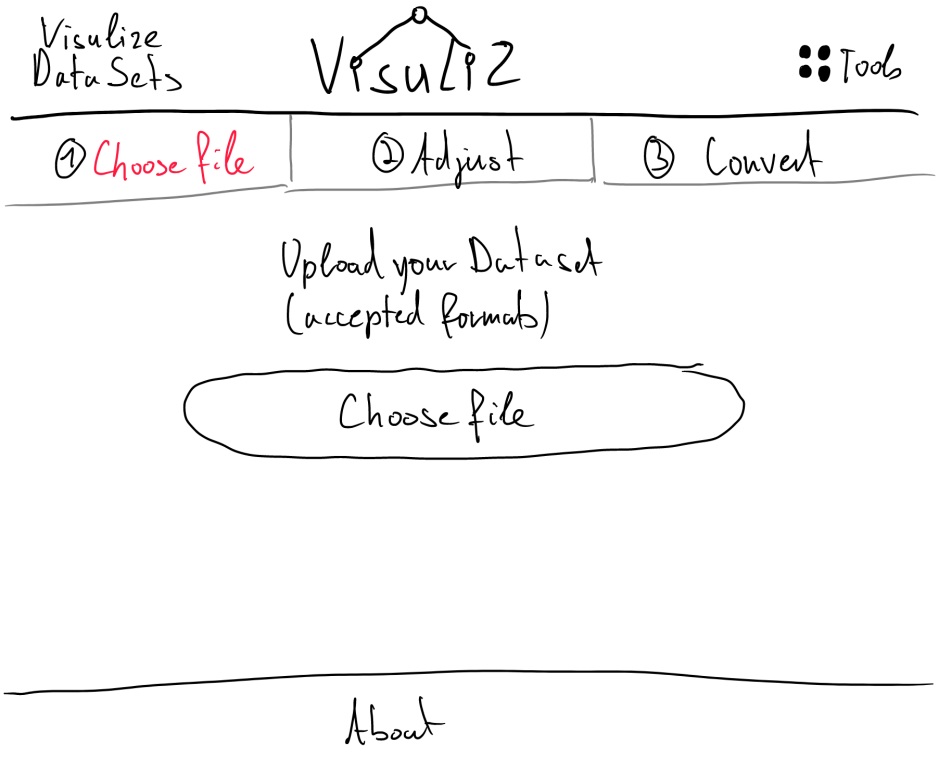


Figure 6.0 – Low fidelity prototype

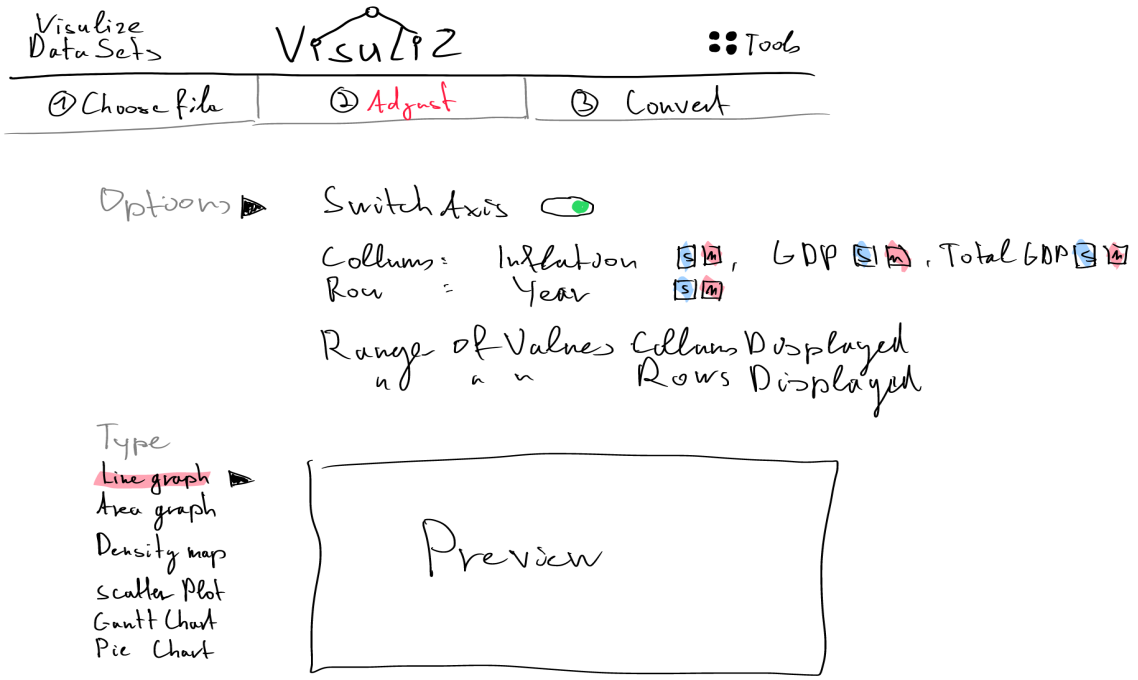


Figure 6.1 – Low fidelity prototype

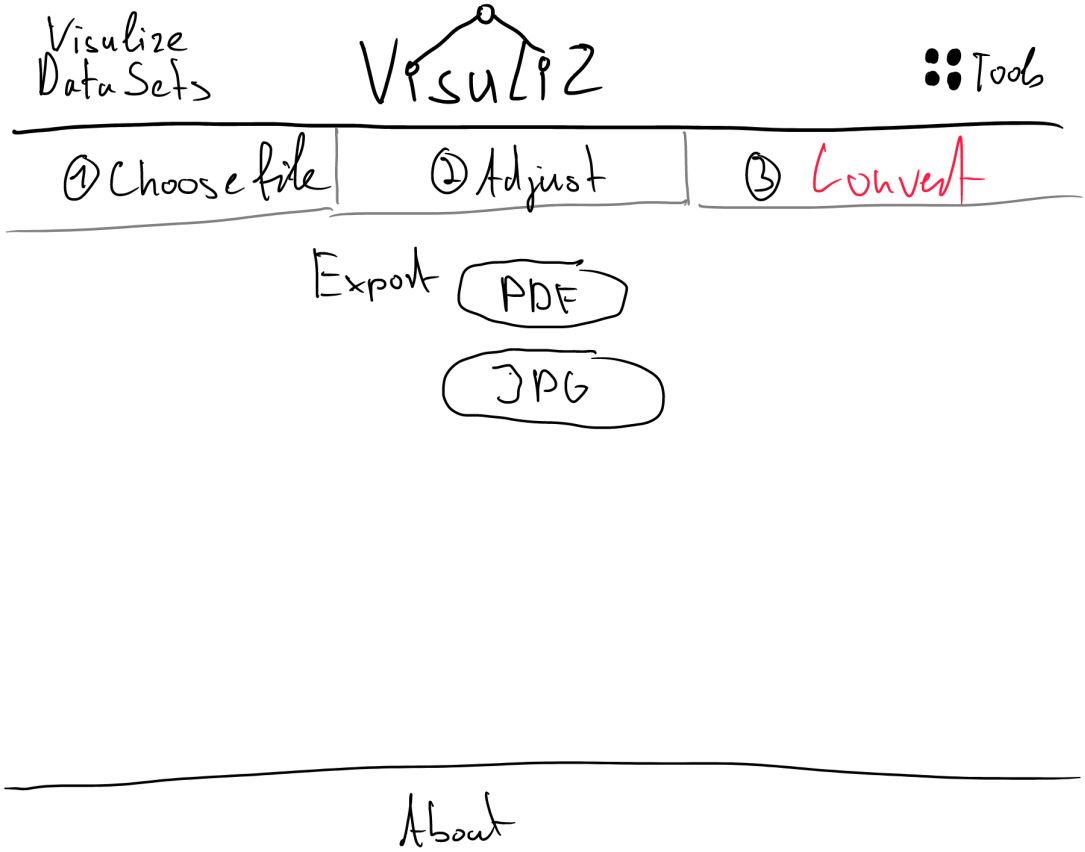


Figure 6.2 – Low fidelity prototype

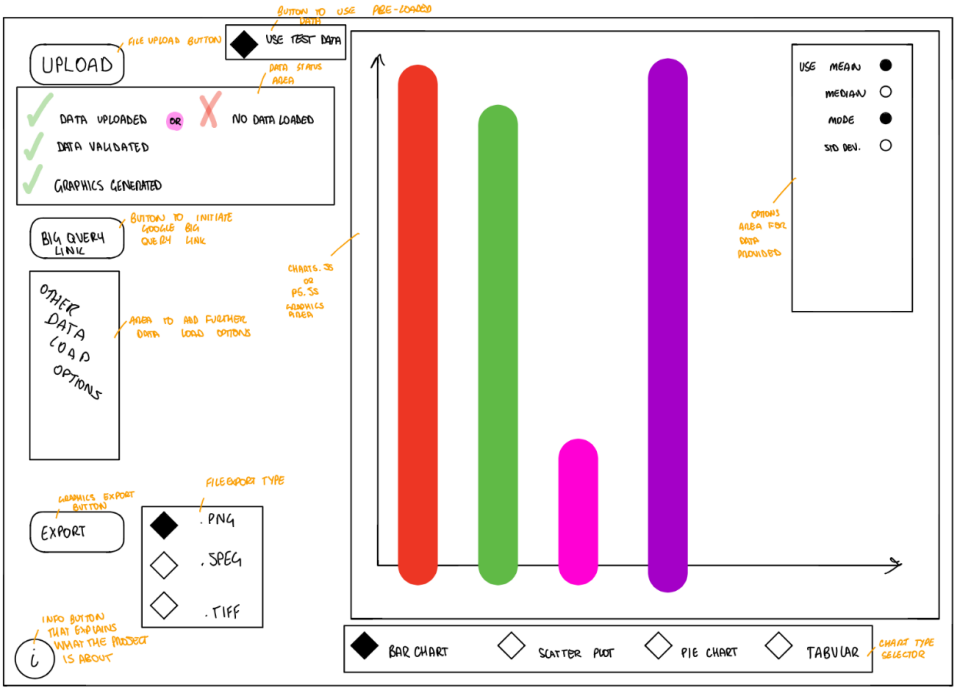


Figure 6.3 – Low fidelity prototype

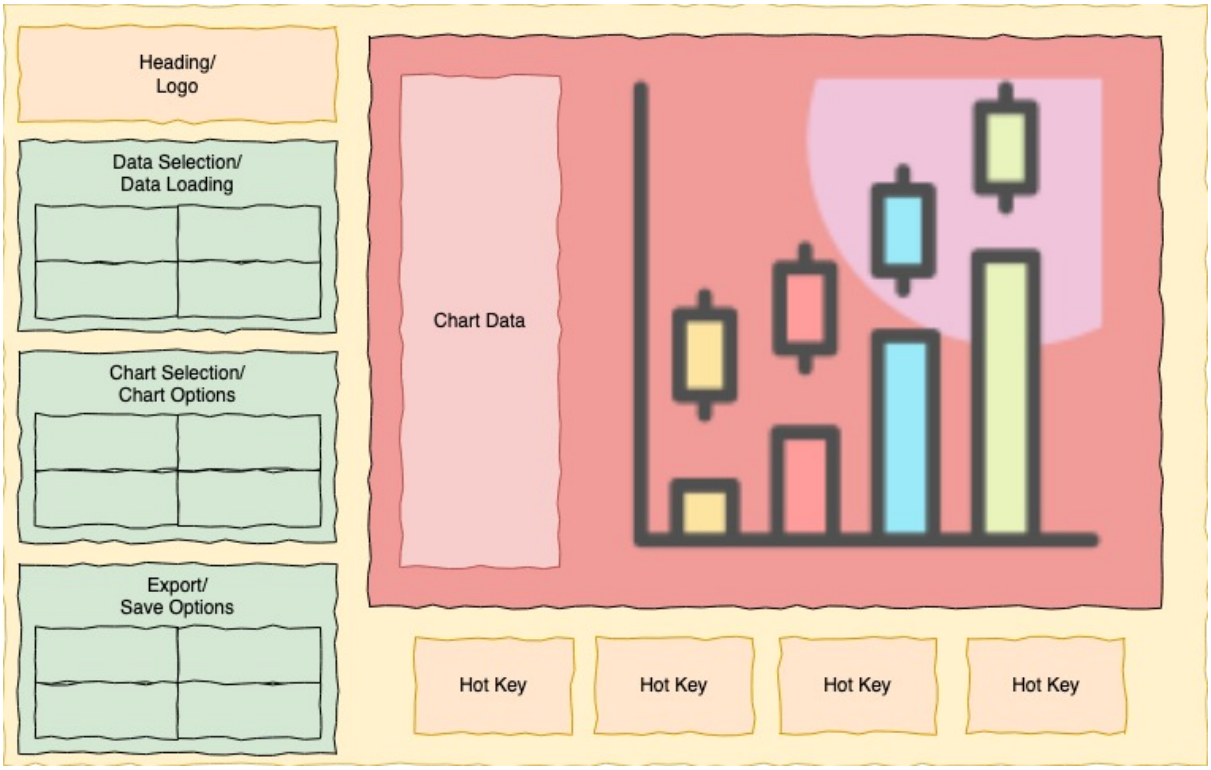


Figure 6.4 – Low fidelity prototype

Figure 6.5 – Low fidelity prototype

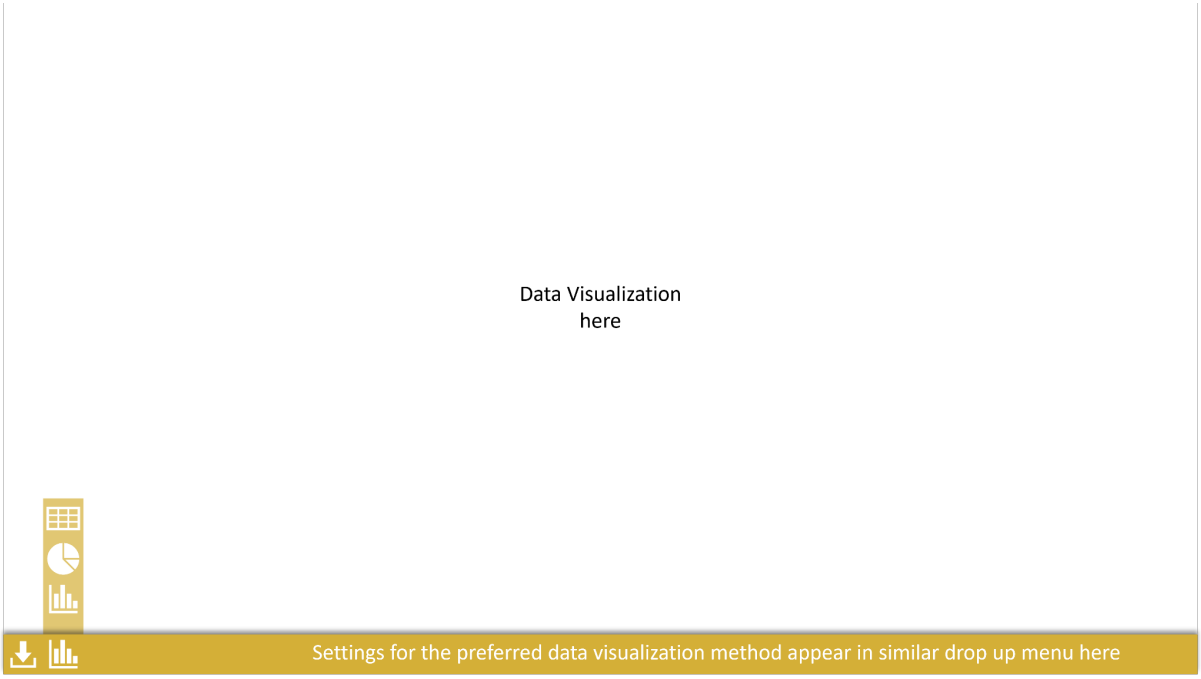


Figure 6.6 – Low fidelity prototype

25. **Prototype feedback.** The main feedback was to combine the best part of each wireframe to create the best the web application can offer. This includes progress-based input menu, simple menu features, collapsible menu and a large data representation space. All future prototypes use this princepal.

26. **Medium fidelity prototypes.** A medium fidelity prototype was created to steer the team into a design that would be viable for production. This prototype was assessed during one of the final Design team meetings and was selected to be progressed into a high fidelity prototype.

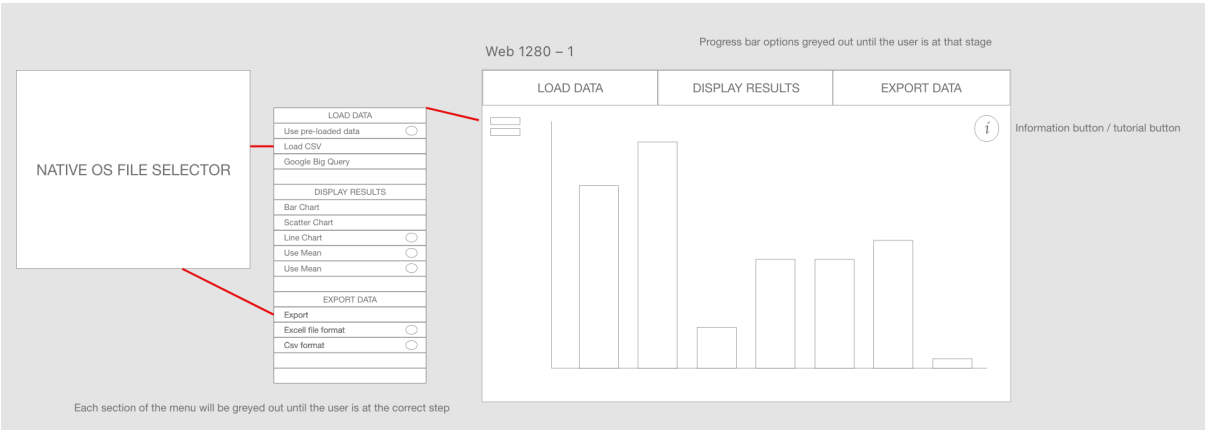
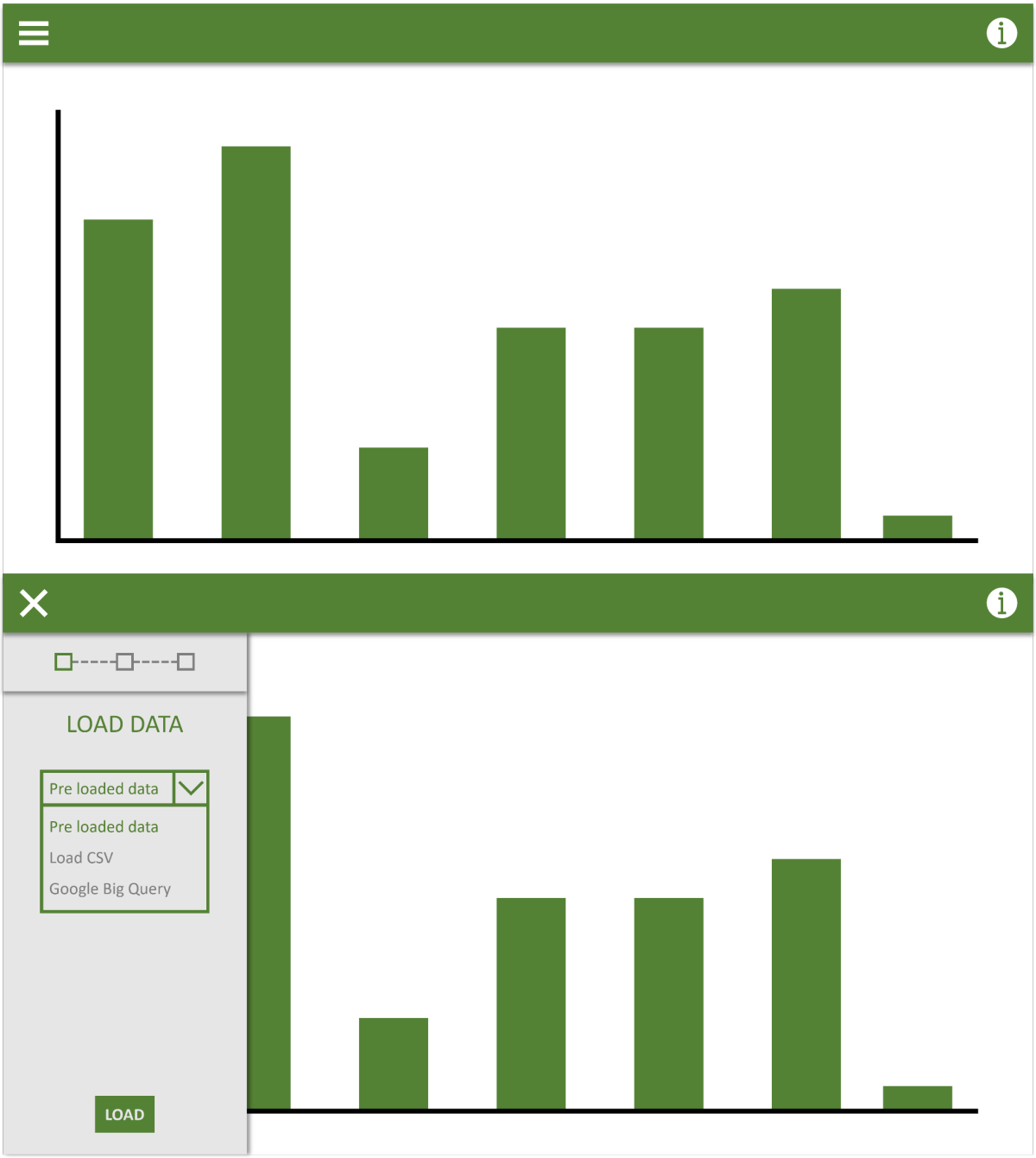
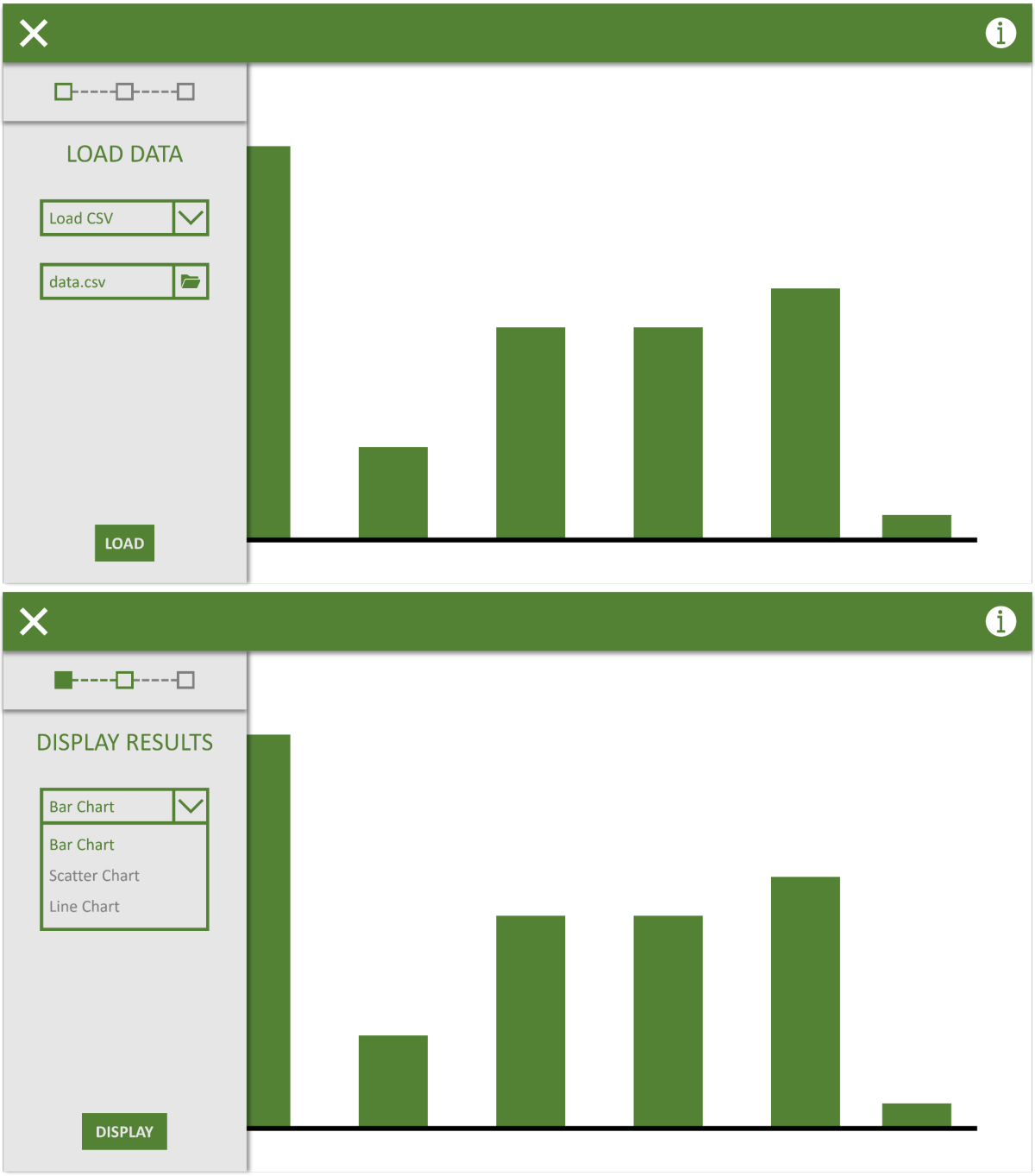


Figure 6.7 – Medium fidelity prototype

27. **High fidelity prototype.** Figure 6.8 is the final prototype that was created. This design was again put to friends and family to determine if it was fit for purpose. Several users repeated the need for it to be used on mobile applications. All other feedback was positive and the simple design and interface was praised by the basic computer users who participated in the feedback. Indicating the design is appropriate for production.


LOAD DATA
Pre loaded data
Pre loaded data
Load CSV
Google Big Query
LOAD


LOAD DATA
Load CSV
data.csv
LOAD
DISPLAY RESULTS
Bar Chart
Bar Chart
Scatter Chart
Line Chart
DISPLAY

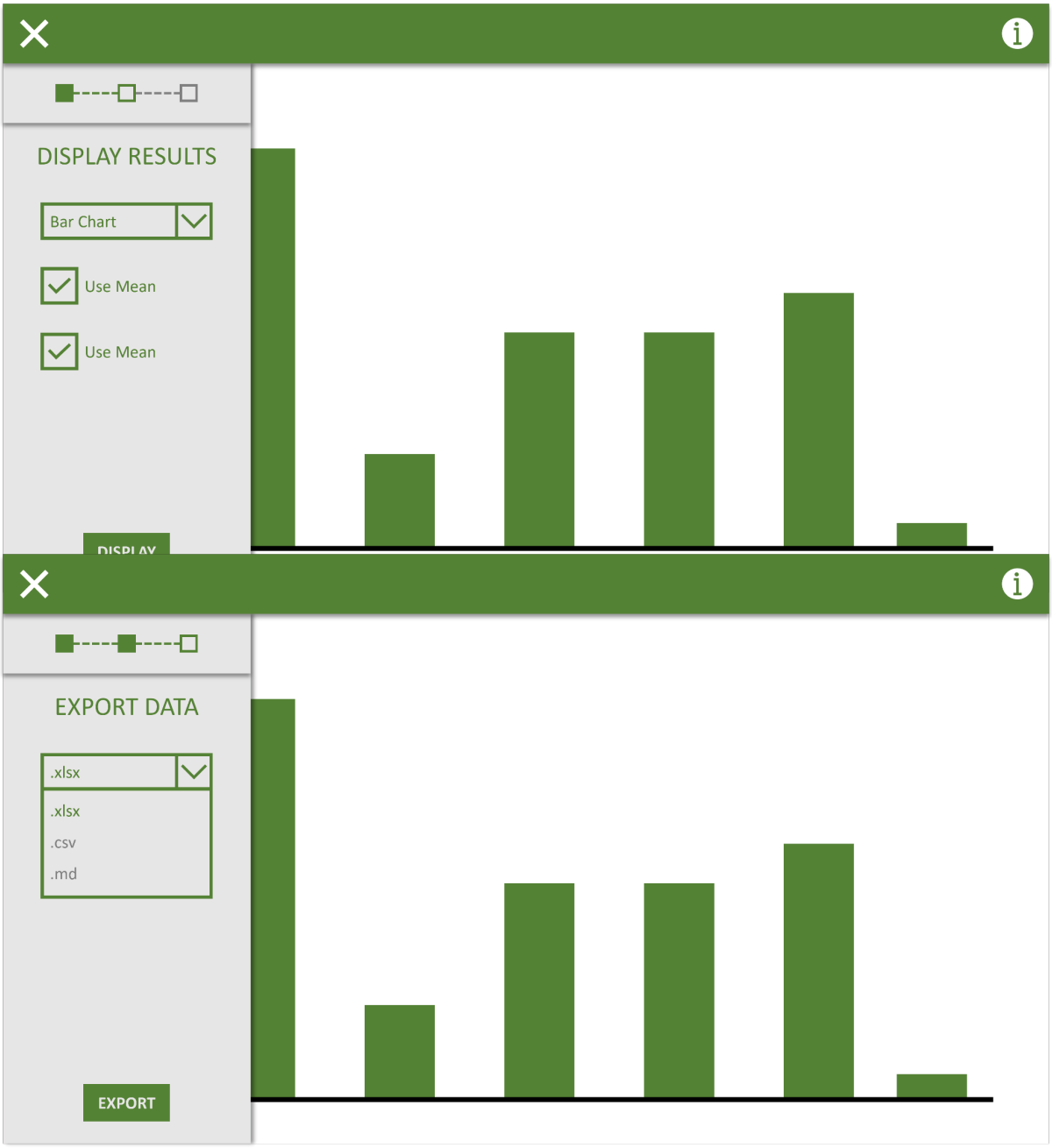


Figure 6.8 – High fidelity prototype

7.0 - Evaluation

28. The project has proved to be effective thus far. All interactions with friends and family point towards a positive attitude towards the proposed product. Market analysis also indicates that the project would be well received as there currently does not exist a competitor in the market for a simple Data Science tool. Any issues in designing the project thus far have been minimal and unlikely to have any impact on the products delivery and/or development. Usability has been a key consideration of this project and as such it has been essential to include the users we wish to target with the system. The system focuses on elementary computer users which could range from the elderly through to primary schools. This product would likely be better received by adults as children would almost certainly find Data Science too boring to pay any continued attention to. For that reason the product will be targeted towards adult basic computer users. This raises an interesting question about the product and that is “How would we make the product interesting to school children studying computer science?”. Such a question is beyond the scope of this project but it would be something that would take multiple iterations of requirements gathering to determine the answer.

With the feedback, requirements and designs thus far the project appears to be in a positive direction as we enter the development phase. The overall evaluation is strong and the team are motivated to complete it. This project is limited in scope due to the amount of time the team has to complete it. With more time There is the potential to add many new features to the system. The team are hopeful that some of these additional features will be added to the system if time permits.

8.0 - Conclusion

29. **Summary of findings.** Throughout this project it has been interesting to get the perspective of the users we are targeting. Something that was evident was the need to use the product on all devices. This is interesting as it is something that has become normal in modern times. Historically applications weren't able to be ran on mobile devices and now people expect software to run on every device. Another interesting finding was when the initial data trawl was conducted. Several users felt they had stronger computing abilities than they actually did. This may have been nerves from the product interview but it did indicate that not all people find basic tasks on a computer easy. This was why the team decided to bias towards these individuals. By making the application user friendly for these types of users it means anyone with more comprehension would also find benefit in its use.

If the project had more time there are multiple areas that would like to be explored. As the program targets basic computer users children start to fall into that set of individuals. No children were involved with this project due to the ethical implications. This is something that could be explored, under the correct circumstances, to find a solution for the program being friendly to all users at all ages. Another element that would require much more time is by making the product suit more advanced users as well. If the system had an API or the ability to tweak features more precisely on the application it would mean there would be more appeal from moderate and advanced computer users. Cyber security implications would also need to have a more comprehensive review to ensure there are no exploits that could be staged from the software.

9.0 - Ethics

30. **Privacy.** The team take privacy very seriously to ensure the user feels safe when using the application without being worried that their data gets stolen. To assure the user that we don't collect any data we will denote a disclaimer on the application informing the user of our zero collect data policy. In this message we will explain that the application runs solely in their browser, on the client side, and that all data therefore does not get saved anywhere except their browser cache. This resets when closing the tab or browser. Any interactions are purely with the program and nothing is stored on how the user manipulates the software. The reason this is so important to the tream is because the user may wish to use personal or confidential datasets. To strengthen the security of the system it will only work on HTTS connections to ensure the user’s privacy is not compromised. Any data exports are purely ran client side and will not be transmitted to the internet.

31. **Target Audience.** Building and researching the application, or any work towards the overall project, does not include any form of work from minors or vulnerable adults. Furthermore, the team are not doing the project for monetary gain or anything other personal gain. As the program is in development there might be minor risks at play. The system could make programming mistakes which could show the data incorrectly and thus spread wrong information. However we will make sure to test our application thoroughly and ensure all data remains on the local system. It may show sensitive data but it will never transmit it to a third party. The potential for malicious intents, like the spread of false information, is very low.

People in the usability study are free to use the application however they want. They can use any device they want, use any datasets and apply any seconds. The only 2 things we expect is that they use it at least 3 times and after that answer our questionnaire. We will make sure to tell them that these surveys are completely anonymous and are only shown to our development team to improve the product. The software will be released under the open source and all users who wish to utilise the development version of the software will be free to do so.

10.0 - Bibliography

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. Qualitative Research in Psychology, 3, 77–101.

Victoria Clarke & Virginia Braun (2017) Thematic analysis, The Journal of Positive Psychology, 12:3, 297-298

Sharp, Helen, et al. Interaction Design : Beyond Human-Computer Interaction, John Wiley & Sons, Incorporated, 2019. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/londonww/detail.action?docID=5746446

Free, H., 2022. Best PDF Converter: Create, Convert PDF Files Online (FREE). [online] Freepdfconvert.com. Available at: <https://www.freepdfconvert.com/> [Accessed 25 June 2022].

UserTesting. 2022. What is a good user experience according to Google?. [online] Available at: <https://www.usertesting.com/blog/good-user-experience-according-to-google> [Accessed 25 June 2022].

Business News Daily. 2022. How and Why Businesses Collect Your Personal Data - businessnewsdaily.com. [online] Available at: <https://www.businessnewsdaily.com/10625-businesses-collecting-data.html> [Accessed 25 June 2022].

Usability.gov. 2022. *User Interface Design Basics | Usability.gov*. [online] Available at: <https://www.usability.gov/what-and-why/user-interface-design.html#:~:text=The%20best%20interfaces%20are%20almost,get%20things%20done%20more%20quickly.> [Accessed 25 June 2022].

Legislation.gov.uk. 2022. *Data Protection Act 2018*. [online] Available at: <https://www.legislation.gov.uk/ukpga/2018/12/contents/enacted> [Accessed 25 June 2022].

GOV.UK. 2022. *Understanding accessibility requirements for public sector bodies*. [online] Available at: <https://www.gov.uk/guidance/accessibility-requirements-for-public-sector-websites-and-apps> [Accessed 25 June 2022].

11.0 - Appendices

Figure 2.0 – Project timeline snapshot - Screenshot of the timeline used to track progress of the project as well as key milestones

Figure 3.0 – Data gathering exercise user tasks – tasks given to users for initial data gathering
Figure 3.1 – User performance in data gathering exercise – Initial results from the data gathering exercise
Figure 3.2 – User performance with difficulty in data gathering exercise – user results from the data gathering exercise with perceived difficulty added
Figure 3.3 – Interview questions for data gathering – questions asked to potential users for further data analysis
Figure 3.4 – Sample user interaction flow – Flowchart denoting the steps a user would take when using the program

Figure 6.0 – Low fidelity prototype – initial design idea
Figure 6.1 – Low fidelity prototype – initial design idea
Figure 6.2 – Low fidelity prototype – initial design idea
Figure 6.3 – Low fidelity prototype – initial design idea
Figure 6.4 – Low fidelity prototype – initial design idea
Figure 6.5 – Low fidelity prototype – initial design idea
Figure 6.6 – Low fidelity prototype – initial design idea
Figure 6.7 – Medium fidelity prototype – development on finalised design idea
Figure 6.8 – High fidelity prototype – final design idea to be used as design reference.