A Pocket-Sized Clinic

Cardiff University School of Computer Science & Informatics

Mariza Celliers

CARDIFF UNIVERSITY

PRIFYSGOL CAERDYDD

Module Title: One Semester Individual Project, 40 credits
Project Supervisor: Dr Alia Abdelmoty
Project Moderator: Dr Matthew J.W. Morgan

Abstract

Long term health conditions need to be monitored for the health and safety of the patient. This can be an ongoing task, and without a suitable method can be frustrating.

The project aimed to create a web application designed to monitor patients' health and lifestyle variables, and has the added functionality to share this data with practitioners of their choice. The application has been built to facilitate integration into the NHS as a future improvement, following standards and guidelines to make this process easier.

The project uses a user centred design to create the application with the emphasis on usability and customisability. This is to ensure that all users of the application are able to create an account which can be personalised towards them.

The application is evaluated to ensure it reaches its goals, and future work from this is identified.

Acknowledgements

I would like to thank my friends and family for acting as users to test my application, with special thanks to Corli for giving me advice from the perspective of a clinician. Thank you for all the help and support given to me.

I would also like to give my thanks to my supervisor Alia for all the knowledge and guidance throughout this project.

Contents

1 Introduction

Long term medical conditions often need to be monitored to ensure health statistics stay within a healthy range. For example, a patient with diabetes needs to monitor their blood glucose levels to ensure they stay within a specified range. If a patient with a long-term condition doesn't monitor these levels, it can potentially become fatal.

One way of monitoring this is through an electronic application, to track and record different variables to understand the correlations between health levels and other variables which may affect these levels. This can give the patient a better understanding of their condition to effectively monitor their variables.

This report goes into details about the process surrounding creating an application to monitor a variety of health statistics. I focus on users with hypertension during development, however the application is developed to be useful for all long-term health conditions. This is achieved by developing a user's profile in a very customisable way, allowing them to monitor any variable. I have followed the code of practice for developing health and wellness apps as set out by the British Standards Institution. According with these guidelines, I have fulfilled all the application lifecycle phases before deployment using a user centred design approach. (Health and wellness apps – Quality criteria across the life cycle – Code of practice, 2015)

I have placed a high level of emphasis on the usability of the application, and using standards outlined by Hl7 to enable the integration of the application with the NHS. (Hl7.org.E, n.d.)

2 Background

Throughout this section I will be conducting background research for the development of the application. This will give me a solid understanding of the project scope, and find the resources needed to continue with the specifications of the application.

I will focus on researching into long-term health conditions, how they are currently monitored, what users want from an application and I will understand how the system can fit in with the NHS. From this I will decide what methodology will be used during the development of the application and create the requirements that the application will need to fulfil.

2.1 Project Background

A chronic medical condition is a persisting condition which is usually described as lasting over 3 months. These conditions often need to be tracked to give the patients knowledge about their symptoms. (Bernell and Howard, 2016)

Levels which would be considered as normal will change from person to person depending on different variables, therefore understanding about their condition is a necessity in management. This enables patients to understand when their data deviates from their own normal levels instead of someone else's.

By tracking their own data, it solves this problem. It gives them the ability to see their trends to find their own normal levels, and what other factors may affect these levels by comparing their datasets. This enables them to self-manage their condition, and find their own patterns and trends to give a better quality of life. There is also evidence that self-management of long term conditions can reduce hospitalisation and accident and emergency attendances, and that people with these conditions may not be able to get more involved in self-management as the services available aren't flexible enough. (Corben and Rosen, 2005)

By tracking their own conditions, it enables them to make informed decisions relating to their healthcare, the NHS are implementing shared decision making where both patient and doctor give input into the patient's healthcare. If the patient is aware of their condition and the variables surrounding it, they will be able to partake in SDM. Enabling them to reach a consensus with the practitioner instead of the practitioner making decisions for them. SDM has a high effectiveness rated amongst the practitioners and patients. (NHS, 2018)

The transfer of knowledge between patient and doctor can become easier as well. The doctor's knowledge is not limited to the tests they can run at the appointment, but also readings which have been taken over the previous few weeks. This leads to more effective treatment of the patient, as the doctor will understand the specific condition, and how treatments are affecting the patient in more depth.

2.2 Project Scope:

A generic condition tracker application would be able to track many different symptoms of different conditions to understand how they can affect each other. This will enable the self-management of any condition for any patient, allowing them to have a flexible service which enables them to monitor any conditions. However, the creation of a generic application will be too large a project to complete within the limited time frame. By focussing on one condition it will allow me to create a product within the allocated time period for this project.

The primary objective is to create an application for users to track anything that the patient thinks is affecting their conditions, and view these values in a useful format. The goal is to enable the user to self-manage their condition, by seeing their vital statistics change as different variables change. An application will be created with one specific condition in focus, however it will extensible and customizable to any condition. The system will be able to log any data both numeric and text, and visualise this data.

The requirements will be finalised and decided by using user centred design to ensure the high usability of the application. This project will be completed within the 12-week period, the tasks will be split up as specified in the Gantt chart in appendix 2.

2.3 Choice of Condition

Throughout this project, the system scope will be focussed on patients with high blood pressure. The system will allow users to track and monitor their statistics relating to this condition, however will be expandable so that patients with other conditions may find it useful as well.

I will focus specifically on a condition where the patient is able to monitor the variables relating to this condition at home easily using an electronic device. This limits my choices to diabetes, asthma and hypertension. With diabetes, patient's blood glucose levels are measured at home regularly, patients with hypertension are required to monitor diastolic pressure, systolic pressure and pulse, asthma patients can measure their peak flow. These conditions have their pros and cons.

Diabetes as a condition is very complicated which would require a very specific application. To monitor all the different factors which can affect this condition, the application will become very detailed and particular. The factors that need to be monitored aren't widely measured for other conditions, and as such aren't reusable. Therefore, diabetes will not be a suitable choice to centre the application around. (Diabetes UK, n.d)

Between asthma and hypertension, either could be used. However, hypertension is a condition which affects over 3 times the amount of people than asthma, which means the application has the potential to be more widely used. Hypertension needs to be monitored whilst finding medications, or employing lifestyle changes, which would be made easier by using an application to compare multiple datasets. (Asthma UK, 2018) (Ageuk.org, 2015)

When a patient is diagnosed with hypertension, a blood pressure monitor is often given by a medical professional to monitor their systolic and diastolic pressure along with their pulse. This gives three values to monitor in comparison to asthma which would only have peak flow, and inhaler usage. High blood pressure also has more factors affecting their readings which can be reusable for other conditions as well such as caffeine intake, sleep levels and stress levels.

2.4 High Blood Pressure

High blood pressure or hypertension is a long-term condition where the pressure within the blood vessels is higher than what it should be. This makes it more difficult to pump the blood around the body, which can lead to complications, and therefore needs to be managed. These complications include heart diseases, heart attacks, strokes, etc. The people with an increased risk of blood pressure are over the age of 65, overweight, have a high salt intake and have a sedentary lifestyle. (nhs.uk, 2018)

Blood pressure is measured by diastolic and systolic pressure in mmHg (millimetres of mercury). Systolic is the force of which the patient's heart pumps blood around their body, diastolic is the resistance within the vessels. (nhs.uk, 2018)

If a patient has a reading of over 140/90 mmHg, a home blood pressure monitor will be given to record measurements over a period of time. A patient will also be encouraged to buy their own to continue this after the time has elapsed. Once high blood pressure has been confirmed, if severe a patient will immediately be given antihypertensive drug treatments, else the patient will be advised on lifestyle interventions. (Pathways.nice.org.uk, 2018)

Antihypertensive drug treatments will be prescribed to patients under 80 years of age which reach criteria such as target organ damage or established cardiovascular disease. (Pathways.nice.org.uk, 2018)

2.5 Review of Existing Solutions

Throughout this review of existing solutions, I will be looking at the usability of the systems listed below using Jakob Nielsens usability heuristics. This may help me to identify possible problems which I can avoid whilst implementing my solution. I will also be evaluating the functionality of the system, to understand where the project can be extended or modified. (Nielsen, 1994)

2.5.1 Advanced Blood Pressure Monitor - £39.99

'The Kinetik Advanced Blood Pressure Monitor is designed to make it simple to regularly check your blood pressure'

The blood pressure monitor would be a potential solution to the problem, as it enables the user to track their blood pressure to better understand their condition. However, It's specific to measuring blood pressure and therefore too narrow for the full scope of this project.

2.5.1.1 Functionality:

- Check the users Systolic and Diastolic pressure, and their pulse.
- Storage of 90 sets of measurements for the patient to see previous readings.

- Traffic light system changes the backlight to red, yellow or green depending on the measurement.

2.5.1.2 Review:

The functionality of this system is effective in terms of providing a solution to the problem, however it's very limited in comparison to the project scope. The user is able to read their own statistics by using the system, thus it is effective as it makes it convenient for the patient. The patient does not have to transfer the data from one reading system to another recording system, however most patients decide to.

The data can only be viewed as values, not as graphs, this can make it more challenging to interpret the collected data. The users therefore will find it more difficult to understand how their data has changed over time, and be unable to easily identify patterns.

Nielsen states the importance of **consistency and standards**, allowing users to easily understand the system. This solution is consistent and logical throughout, each of the recorded measurements have the same layout. Therefore, when a user scans through it's easy to compare the measurements on different screens. The annotations on the side of the display that explain the readings ensure the consistency and ease of interpretation as they are not digital.

Another heuristic is **minimalist design**, the traffic light system is effective as it's very simple to understand what the values mean without having to understand, or having other text or values on the display. The large numbers with annotations on the side also allows for easy reading of the measurements simply, without extra clutter.

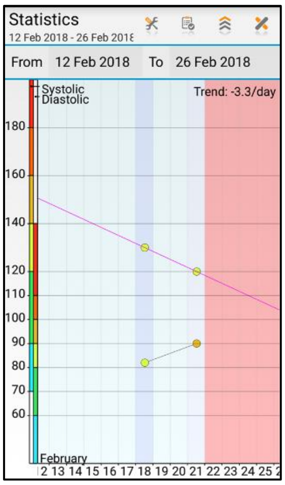
Help and documentation is effective as although the system is easy to use it may not be as intuitive. The buttons can be confusing as some users may find the abbreviations like 'SET' and 'MEM' not be readily descriptive. The user guide that is sold with the product however, is well laid out with good diagrams to aid the users understanding. This enables a new user to access the added functionality without resorting to trial and error when initially using the system.

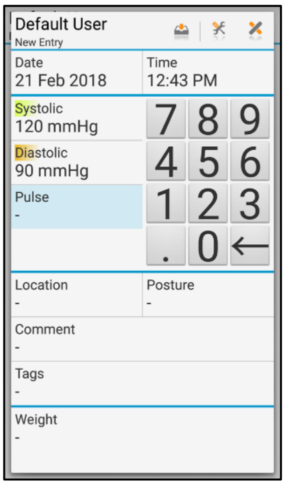
Although, **Recognition rather than recall** is a limitation in this system. The readings that are saved are difficult to compare as they cannot be seen next to each other, only on separate screens. This can make it difficult to evaluate how the users blood pressure has varied over time, as the user will continuously have to scroll through their readings to compare.

Overall the system is a simple solution which works well, however there's not much functionality.

2.5.2 Blood Pressure Log – MyDiary

A product like this one would most likely be used in conjunction with a blood pressure monitor like the previous technology discussed. A monitor will be used to read the patients statistics, these will be entered into an application to be able to keep track and monitor the changes.




Default User
New Entry

Date 21 Feb 2018

Time 12:43 PM

Systolic 120 mmHg

Diastolic 90 mmHg

Pulse -

Location -

Posture -

Comment -

Tags -

Weight -

'Log the blood pressure, view trends, and send the results to your doctor.'

Blood Pressure Log is the highest rated blood pressure tracker in the Google Play store at the time of writing. It allows users to enter data about their blood pressure readings, and view this data over time on graphs. This doesn't meet the full scope of the project as it's specific only to blood pressure, and other variables can't be monitored.

2.5.2.1 Functionality:

- Ability to store Diastolic and Systolic pressure, pulse, weight, blood glucose and oxygen saturation.
- Logging of qualitative data such as extra text, mood, cuff location and seating position.
- Viewing of data on linear graphs, bar graphs and category diagrams.
- Ability to compare different datasets on the same graph.
- Highlighting values in red, green and yellow depending on the measurement

2.5.2.2 Review:

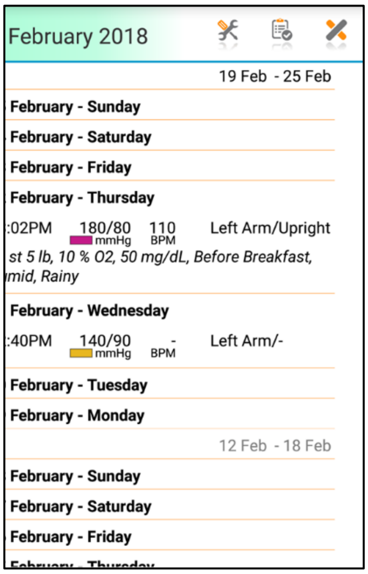
The application has a traffic light system, similar to the previous technology, which enables the user to quickly and easily interpret the status of their data without understanding the data values. This can help save the users time as they don't have to look up what their readings mean. The application also allows the users to store extra information which can help them understand their condition better, extra text fields allow the user to annotate information about their symptoms, stress levels, etc. This can be looked over on a different date to find patterns. By allowing the user to overlay datasets more trends can be spotted, the user can evaluate different statistics to understand the correlation between them.

However, the graphs can get complicated when adding datasets. The application does not give a key to be able to decipher what the plotted values mean, which can be confusing to inexperienced users of the application.

Consistency and standards are not strong points of the application, all pages look very different. For example, creating a new entry initiates a pop-up to enter the data into with a new set of unlabelled, not standard buttons in the toolbar. In comparison to view a graph takes you to a new page of the application, these differences change the layout of the page which makes it unnecessarily complicated for new users.

The match between system and the real world is emphasized with the unfamiliar buttons. The buttons on the toolbars throughout the application change with each screen. They are unlabelled and are not standard. This makes it difficult to understand how to use the application and utilize the full range of functionality of the system.

However, **User control and freedom** within the system is good. There is a 'X' in the corner of all pages and pop-ups to enable the user to return to the homepage or previous page. If data has been entered, a warning sign appears signifying that data will be lost if exited. This enables a user to exit unwanted states with ease, saving them time when performing actions, whilst still warning them that data may be lost.



February 2018

19 Feb - 25 Feb

February - Sunday

February - Saturday

February - Friday

February - Thursday

:02PM 180/80 mmHg 110 BPM Left Arm/Upright

st 5 lb, 10 % O2, 50 mg/dL, Before Breakfast,
ımid, Rainy

February - Wednesday

:40PM 140/90 mmHg - BPM Left Arm/-

February - Tuesday

February - Monday

12 Feb - 18 Feb

February - Sunday

February - Saturday

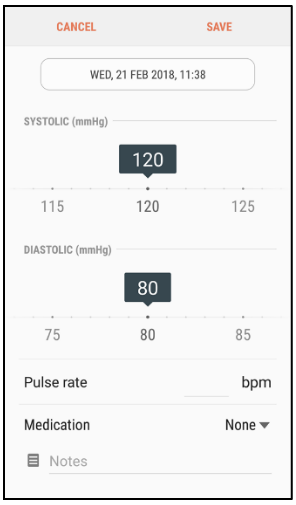
February - Friday

The **Aesthetics and minimalist design** of the system is not always achieved. The dashboard is well laid out with 6 options for the user, the buttons are spaced out and in decreasing order of use. However, beyond this page the screens become cluttered and difficult to use. For example, the history page is given as a list focussed in on today's date, the user can scroll up and down to future and past dates. The information given for a specific date doesn't fit in the screens width, needing the user to scroll horizontally to read the rightmost pieces of data. The list also gives all aspects of the information on that date, meaning the user doesn't have to click on the day to see what they need, however this also means the page becomes messy and cluttered.

This system is a good solution in terms of a blood pressure tracker, allowing a wide range on flexibility, even though it's lacking an easy to use interface in comparison to the other solutions evaluated.

2.5.3 Samsung Health

A broader technology such as this one, is able to log more than just a patient's blood pressure. This enables a patient to be able to track many variables they may be interested in, and be able to spot patterns over these.



WED, 21 FEB 2018, 11:38

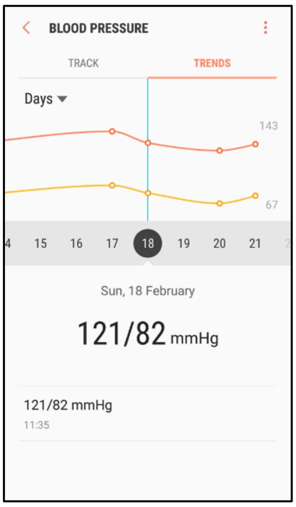
SYSTOLIC (mmHg) 120

DIASTOLIC (mmHg) 80

Pulse rate bpm

Medication None

Notes



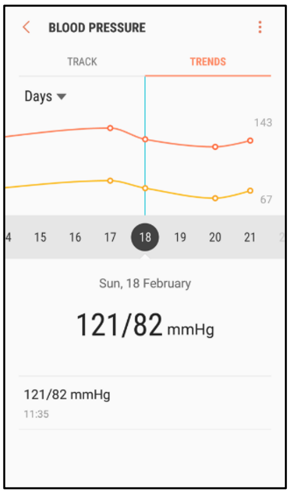
Sun, 18 February

121/82 mmHg

121/82 mmHg
11:35

'Samsung Health analyses your exercise and activity history, and helps you achieve weight loss and lead a healthy lifestyle.'

Samsung health is an application to be able to track multiple different aspects of your lifestyle in a central location. The system allows you to track blood pressure and other measurements which may affect these readings such as caffeine intake, sleep, stress, etc. in separate modules within the app.



Sun, 18 February

121/82 mmHg

121/82 mmHg
11:35

2.5.3.1 Functionality:

- Set goals for the statistics the user tracks.
- Compare data with friends.
- A dashboard for a quick overview of the user's daily readings.
- Ability to view data on graphs separately.
- Integrates with other systems enabling data to be recorded automatically.
- Ability to track multiple health statistics such as: Blood pressure, Sleep, Stress, etc.

2.5.3.2 Review:

The system allows the user to view their statistics on a graph rather than raw data in a table, this helps the user to spot trends more easily. The wide range of functionality allows flexibility to the user, allowing them to record data within the pre-set range, this enables the user to take control of managing their own conditions.

However, the system doesn't allow the user to compare these datasets, they are only able to see one graph at a time. This makes it more difficult for the user to see how different values can affect each other. For example, how sleep levels may affect their blood pressure.

This system has a strength with **visibility of system status**. The user is always aware of the page they are visiting, the navigation bar works as the main navigation for the application allowing the user to switch between the pages 'ME', 'TOGETHER' and 'DISCOVER'. By highlighting the current page, it's obvious where the user is on the application. When the user is exploring their statistics on the ME page, the statistics name they are viewing is displayed as a large title. This makes it easy for users to navigate throughout the application and to discover the whole range of functionality of the system.

The **match between the system and the real world** is shown in multiple aspects of the system, such as the caffeine levels are recorded in cups rather than mg. It will be easier for the user to be able to measure the caffeine levels by how many caffeinated drinks they consume in one day in comparison to calculating the amount of mg of caffeine in each drink. However as different drinks have different amounts of caffeine, this can become unreliable as 2 cups of an energy drink will contain significantly more than 2 cups of green tea. The system currently doesn't have a method of changing units, so it may not work as well for users who take this tracking seriously.

Consistency and standards of the application is somewhat effective. Within the application, when choosing a statistic to track, a similar navigation bar as before is shown, with the options 'TODAY', 'TRENDS' and 'REWARDS'. This enables the user to intuitively know how to navigate through the 'ME' portion of the application. However, throughout the other two pages, 'TOGETHER' and 'DISCOVER', the page layout changes significantly which may cause confusion amongst new users.

The applications **minimalist design** allows a screen with a lot of functionality to still look neat. It's modules on the 'ME' page shows the main point of information that the user will want without having to click on the module, e.g. daily step count or last recorded heart rate. This enables the user to see the information that they most likely want, without having to click on the module and without cluttering the screen.

This system is a wider solution to be able to track health statistics associated with many chronic conditions. The interface is easy to use and minimalistic, but still falls short in terms of the comparison of data sets. This can be a problem for a user wanting to try and understand correlations such as the reasons why their stress levels or blood pressure have increased.

2.5.4 Conclusions

None of the solutions evaluated are full solutions of the project scope, however they have good functionality and interfaces which can be incorporated.

By reviewing their interfaces, the importance of an easy to use system is apparent. Whilst using Blood Pressure Log, there was quick frustration trying to learn how to use the system and having too many items on a screen at once. A minimalistic style will therefore be important such as the Samsung Health interface. Along with this, making the system as easy as possible for the users will be a priority. By following a user centred design by user testing my interface before development, and usability testing afterwards it will enable me to reach a high level of usability.

The Advanced Blood Pressure monitor has emphasised the need for graphs and how difficult it can be to interpret a user's raw data. The user is able to see their blood pressure and how it's changing, but it's time consuming to spot trends and patterns within this data. The graphs on the other applications have shown how to design easy to read systems, and how overlaying datasets can work. Functionality from these will be included such as focusing on a minimalist design, and not showing extra data when it is not necessary.

Blood Pressure Log has helped me decide to create the system on a web platform instead of a mobile application. This is due to how much information needs to be fit on one screen per graph, by using a mobile platform to view multiple graphs overlapped it's difficult to position all data on one screen without it becoming cluttered. To focus on the usability aspect of this project, focussing on a web platform will be more suitable.

One problem that all three applications had was a limit on their flexibility, all of them had a specific range of variables that they could track. Samsung health being the most flexible, and the Advanced Blood Pressure Monitor being the least. By limiting the users on what they are able to monitor, it limits how useful the application can be as either a blood pressure monitor, or a general condition tracker. By enabling the user to choose these variables and how they want to view this data, it helps the user self-manage their own condition the way they want to. Because of this, my solution to the problem will be as flexible to the user as possible, allowing them to customise data entry and the graphs they view.

2.6 NHS Systems

2.6.1 Current Technology

The NHS use an application called SPINE which stores all clinical information in a database which is essential to patient's treatment and care. SPINE has a lot of functionality such as the secure transmission of electronic health records between GP practices and electronic prescriptions. Up until 2014, this was run on an Oracle SQL database, and has now been transferred to Riak, a NoSQL database.

A NoSQL databases purpose is for unstructured data and can be stored in many ways, therefore being more flexible for the user, having many different forms of data and changing data structures. This would be a benefit for the use of the NHS. It makes for an easier way to create and implement new features within SPINE as the data can be added in any format, within reason. However, using a non-relational database the ACID properties do not firmly apply. Atomicity, consistency, isolation and durability are not fully supported which can cause problems with the reading and writing of data to the database.

Within SPINE and other applications in use within GP practices, the current method of sharing patient data between professionals is that all clinicians with the correct level of access can access all patient's data. Different levels of this access determines how much patient data they can view. This level is determined by a SmartCard, this ensures data can only be retrieved by people who have a valid reason. (NHS Digital, 2018)

2.6.2 Hl7 Standards

The NHS have adopted Hl7, an interoperability standard "dedicated to providing a comprehensive framework and related standards for the exchange, integration, sharing, and retrieval of electronic health information that supports clinical practice and the management, delivery and evaluation of health services." This enables applications to be able to access information held in the SPINE database and other services easily and in a controlled format. (Hl7.org.E, n.d.)

The specifications on the structure and the standard of data is FHIR. It creates resources which are easy to implement and are consistent throughout healthcare applications to reach Hl7's goals. These resources are well defined, extensible and reusable to enable the development of all applications upon these. These guidelines have been well adopted within the healthcare community. Therefore, this project will have to align with these standards to ensure the application is maintainable and can integrate with other applications, or the NHS in the future.

2.6.3 Conclusions

To ensure interoperability with the NHS in the future, these FHIR standards will have to be adopted and used. A non-relational database will be more appropriate for use with these standards, along with making the integration with the NHS easier in the future. The use of Riak will not be possible as they do not support windows operating systems, which I will be using for the duration of the project. Other options for a non-relational database will be evaluated to use the most appropriate option.

2.7 User Centred Design

Because of the emphasis on usability of the system, a user centred design will be undertaken throughout the development of the system. This methodology enables the application to meet the user's expectations and requirements and tailors the application to specific tasks. This will result in the project having minimal user error and a high usability. (The Interaction Design Foundation, n.d.)

There are four stages of the user centred design process: (Usability.gov, 2018)

Specify the context use: This relates to finding the users of the application, and understanding what they will need to achieve by using it.

Specify Requirements: Take the information gathered in step one to create requirements for the application. This allows for designing with the users in mind, specific to their problems.

Create design solutions: Based on the goals and requirements, create a design which meets these specifications. This will be based on what the user will need, and how best to solve their problems in the easiest manner.

Evaluate designs: Getting feedback on the product allows it to be improved to create the best solution for the users. Focussing on how they'll interact with the product over how the product will work.

2.7.1 Conclusions:

Within the digital assessment questions, it outlines that an application must be created using a user centred design to be accepted by the NHS. There are many questions related to this process such as 'Were the user needs and user stories the basis upon which the apps were developed? '. All of these questions will need to be taken into account during this process. (Health Developer Network B, n.d.)

As the usability of the application is such an important aspect of the application, it is logical to use a user centred design approach to this project. If the application doesn't fulfil what a user wants from a condition tracker, the application would be considered a failure. By using this approach, it enables me to put the users and their interactions as a priority, ensuring a usable application.

Firstly to start the design process, I will create personas with clear goals to be able to find the users of the application, how they will use it, and what they need from a system. A generic 'User' isn't helpful as they are too vague to create an application around, and can end with too many unuseful features. By directing the design around one or a few personas, it enables me to focus on their specific goals, behaviours and motivations. (Blomkvist, 2018)

After creating these personas, I will create the requirements from these to ensure the application reaches the users goals. By creating use cases it will also enable me to understand how the specified personas will use the program to help with the creation of the designs in the next stage.

Designs will be created as wire frames to give a representation of how the application will look and be used. The personas, use cases and requirements will all be taken into consideration during this step.

The designs created will be subject to a heuristic evaluation and user testing to ensure the usability of the application. Ideally there would be many users within the target audience testing the application. However due to time constraints, only a few users from a wider audience will be able to be tested. The conclusions from these evaluations will be used to update the designs, this will enable me to create final designs before progressing to development.

Once developed, the application will also go through usability testing to find any more usability problems which may not have been identified previously. This is to ensure an end product which solves all of the goals of the personas, with a high level of usability.

This methodology will ensure that I can fulfil the appropriate questions on the digital assessment questions relating to the design process.

2.8 Personas

Creating personas is the first step in the user centred design process, it enables me to focus on the users tasks and goals to create a more effective design. This means creating the system based on the following users, their problems, and what they need from a system. This is instead of being distracted the wider audience's problems. The Digital Assessment Questions also put emphasis on the importance of having user needs to base the application upon, and will allow me to be able to answer 'yes' to the relevant questions. (Health Developer Network B, n.d.)

The requirements will directly link back to the goals of these personas, and the designs will reflect the personas goals.

2.8.1 Persona 1:

Name: Alice
Age: 52
Occupation: Paramedic
Location: Nottingham

2.8.1.1 Background:

Alice has a high stress job, which requires her to work night shifts and under pressure constantly. She found completing her bachelor's degree in paramedic science also very stressful, so is used to managing and coping through different techniques. However, because of the long hours of her job and sporadic shift patterns, she finds it difficult to find time for herself.

After work, Alice will usually opt for a microwave meal over home cooking for the convenience, finds herself running errands whilst she has the time, watching TV with her partner, and browsing the internet on her tablet. As hobbies, she enjoys seeing her friends, and gardening.

Previously, Alice never had any existing long-term medical conditions. A month ago, her sister was diagnosed with high blood pressure so decided to check herself out. Her blood pressure at the GP was 156/91, she's been advised to lead a healthier lifestyle. This includes, cutting down her salt intake and more regular exercise.

Having a background in medicine, Alice knows her lifestyle isn't healthy, and does need to change. She wants to be able to track her changes easily and see if it's having an effect on her blood pressure. Keeping a track on her readings to ensure she can bring it down without medication ideally. Her partner is supportive and is also making the lifestyle changes.

Currently Alice tracks her blood pressure after her shifts, every day or every two days, whilst watching TV with her partner. She does this on paper in a diary however this is tedious. She finds herself losing motivation and wants to remind herself how much better she's feeling after changing her lifestyle, she feels like this would encourage her to keep going.

When visiting the GP she finds it frustrating that she has to share her readings with him, and they have to take time reviewing it during her short appointments. She would like them to be able to view this data before she arrives, to keep the appointment concise and short.

2.8.1.2 Alice's Goals:

- Recording lifestyle changes easily to stay motivated.
- Keep all of her data organised, and easy to reference and review.
- Be able to control how and when she records her information.
- Communicate her data effectively with her GP

2.8.2 Persona 2:

Name: Erik
Age: 65
Occupation: Dentist
Location: Reading

2.8.2.1 Background:

Erik has a Masters in special care dentistry. He's worked for the previous 40 years, as a NHS and private dentist. Having a successful career, he's winding down from work now and getting ready for retirement. He enjoys spending time with his grandchildren, working 3 days a week and babysitting for the other 2. He tries to keep an active lifestyle to keep healthy to stay alive to watch his grandkids grow up. As a single dad, he's enjoyed cooking for himself and his kids since his kids were small.

Working in his profession, Erik has a high computer literacy having to input patient data into computers frequently. At home he uses a computer most evenings to find recipes online, and read the news, whether this is in his study, or using a laptop in his lounge.

He's been diagnosed with high blood pressure for the past 7 months. Since then, Erik has stopped drinking as much, and has cut down on his caffeine intake to try and lower his readings but they have stayed high. He's currently trying out one medication to try and manage his readings, so far it's been successful. However, the side effects are too bad to manage and wants to try a different medication to manage it instead.

Currently Erik is using Samsung health on his phone to monitor his blood pressure, twice a day every day. He enjoys being able to track many aspects of his lifestyle changes to keep on top of them. Although, he is unable to track his medications on the graph and finds it difficult to remember when he started the medication to compare before and after.

2.8.2.2 Erik's Goals:

- Need an easy way to review how medication changes affects his condition.
- Simple way to record and view if his lifestyle changes have affected his condition
- Keep all of his data in one place, to not switch between applications.

2.9 Requirements

The requirements are the outlines everything that the application should fulfil. These relate back to the personas created to satisfy their goals, the Digital Assessment questions and also the Hl7 guidelines mentioned throughout this report.

2.9.1 Essential Objectives:

2.9.1.1 Functional:

FR1 - Requirement:
The system should enable the user to view the information they've entered for their profile.
Justification:
A user may need to see information such as their birthday for reference, and see what data the application holds about them.
Acceptance Criteria:
The application has a profile which shows the user all of their entered personal information.

FR2 - Requirement:
The user should be able to change the profile information shown.
Justification:
A user may change their information such as names, and therefore needs update the information shown. This relates to FR1.
Acceptance Criteria:
The user should be able to easily change their profile information from the profile page. The information which is shown in the user's profile, when updated must change.

FR3 - Requirement:
The user should be able to enter the information relating to their high blood pressure into the system, for example their systolic pressure.
Justification:
By allowing the user to record data whenever, and not requiring all data fields to be filled before submission will fulfil Alice's goal of being in control of when she records data.

It is important for the correct information to be shown on the graphs, which means the users will have to enter the data themselves.
Acceptance Criteria:
The user interface must have specific areas that enable the user to input their data with the correct unit.

FR4 - Requirement:
The system should support the user wanting to record more than the specified fields given in FR3.
Justification:
One of Erik's user goals was to be able to see how his lifestyle choices have affected his condition. By giving him the ability to record anything, and then be able to view these variables on the graph, he would be able to track all of his lifestyle changes, and not just the pre-set range.

This also allows the application to be more generic for users who don't have hypertension but other conditions.
Acceptance Criteria:

A user should be able to create new fields which include a measurement name, a unit and measurement type. Once a user has tracked a field once, they should not have to create the field again.

FR5 - Requirement:
The system should enable the user to view their condition information easily via graphs.
Justification:
A goal of Alice's was to be able to stay motivated, this could be achieved through graphs. If she could see how changing one lifestyle variable affects her health statistics, she'd be able to see the positive impact of her efforts. This would make it easier to stay motivated.

The easiest way of finding correlations in data is by using graphs. By displaying the user's information in graphs, it enables them to analyse their data easier and faster than displaying a table of their raw data.
Acceptance Criteria:
The system should show be able to graphs relating the specific user's data.

FR6 - Requirement:
The system should support the user tracking qualitative data such as how they are feeling.
Justification:
By tracking their emotions and other qualitative data a user can understand their condition better than just numeric values and how it could be making them feel.
Acceptance Criteria:
When creating a new field, a user should be able to choose the fields type to be text. Once data has been entered, this field should not be plottable on a graph, but when a data point has associated text data this should be shown when rolled over.

FR7 - Requirement:
The system should enable a patient to share their data with a healthcare professionals profile.
Justification:
The problem that Alice had of not being able to easily share her information with practitioner would be solved by her GP being able to have their own account to view the data. This would allow them to be able to make more justified decisions on medications and treatments based on the blood pressure data submitted over time by Alice or a patient, in comparison to the one blood pressure measurement they can make at the appointment.

As clinicians in GP's have such short time to see patients and their meetings are rushed, even for long-term patients, this functionality will enable a professional to see the data before their appointment, saving time. However, as they have such little time, extra functionality such as being able to contact the practitioner through the application will not be required.
Acceptance Criteria:
A user should be able to share their data with specific medical practitioners using their email address.

FR8 - Requirement:
A user should be able to compare different datasets in graph formats by overlaying datasets. For example, a user choosing being able to compare the weight graph and blood pressure graph.
Justification:
By being able to relate two distinct measures, a user will be able to understand their condition better, and how different measures may affect each other.
Acceptance Criteria:

When creating a new graph, a user should be able to add multiple data sets. These datasets should show on the same graph with the axis changing if needed.

FR9 - Requirement:
A medical professional will be able to have their own profile to view all the data shared with them by their patients.
Justification:
As in FR7, if they have their own account they will be able to review the data before an appointment to be able to save time during the appointment.
Acceptance Criteria:
A medical practitioner should have their own type of account, and be able to view the data relating only to the patients who have shared their data with them.

FR10- Requirement:
The user should be able to see how changes in medication has changed their blood pressure and other symptoms.
Justification:
This will solve Erik's goal of needing to view his medication changes. He needs to see his medication changes to keep track of the side effects of the medications, and how it effects his health statistics over time. This will enable him to make justified decisions on whether the medication is working, and how well it's working.
Acceptance Criteria:
The application will be designed to be able to view the medications on the graphs, and within the raw data.

2.9.1.2 Non-Functional:

NF1 - Requirement:
A user should be able to use the system easily and intuitively.
Justification:
By making the system easy to use, a user will enjoy the application more. If the system is difficult to use, users will find other ways of recording their measurements.
Acceptance Criteria:
Once the application has been developed, usability testing will be carried out to ensure the ease of use of the system. A user should be able to use the application without any prior training, with minimal confusion and few errors.

NF2 - Requirement:
The application will be developed on an appropriate platform, this platform should take into account the ease of use of the system.
Justification:
A platform must be easy for the users to use, must be available for the target audience and have the functionality needed for the application.
Acceptance Criteria:
The platform must be the most appropriate choice for the project.

NF3 - Requirement:
The data should be stored securely on the database and the application should be secure.
Justification:
The data that will be stored is personal private information which should be secure from people who don't need access to the data.

Acceptance Criteria:
The storage of data will be up to security standards currently in place.

NF4 - Requirement:
The storage of data should be up to current standards.
Justification:
Healthcare data needs to be stored in a specific format to enable the exchange of data between other applications in the future. By having the storage up to the standard as specified online, it will enable the application to be more extensible and interoperable in the future when exchanging information with other applications, for example the NHS systems.
Acceptance Criteria:
The application will be designed to standards in FHIR and Hl7.

2.9.2 Desirable Objectives:

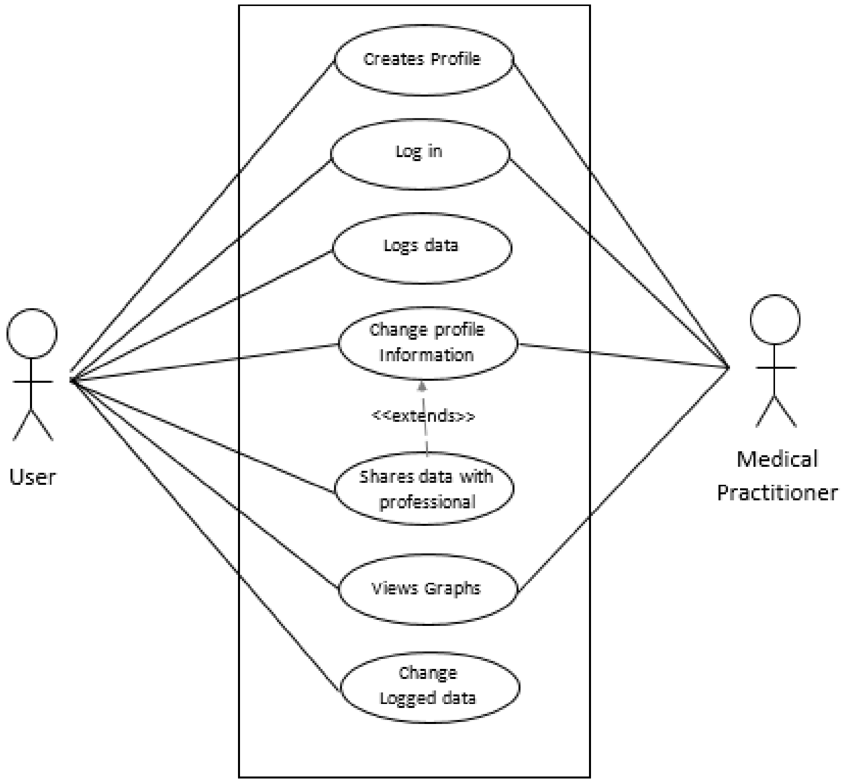
Functional:

FR10 - Requirement:
A user should be able to use the system to find out the normal levels of their data.
Justification:
A user will have their blood pressure at normal levels for them depending on their age, weight, height, medication etc. By the system undertaking data mining techniques it will find the levels that are normal to the specific patient.

If the user is aware of what is normal for them, they will understand their condition better and be able to understand when their blood pressure deviates from this and the underlying causes easier than currently.
Acceptance Criteria:
When the user inputs data, the user will be able to see when this has deviated from the normal level. This data will also be available on the graphs provided.

2.10 Use Cases

Use cases have been used to define the way the personas will interact with the application. These helps understand the extent of the project, and help with the design ensuring the personas have been at the centre of this. The main use cases of how the user will interact with the application can be found below, the remainder of the interactions will be held in appendix 1.



2.10.1 ID: UC3

Title: User adds data to profile
Description:
The user adds data about their condition to the profile.
Primary Actor: User
Preconditions: UC1, UC2

Main Flow:
1. The user selects 'log a reading'
2. The application proceeds to the logging page
   Data Entry
3. The user enters what data they would like into the relevant fields
   a. Blood Pressure, Salt intake, sleep etc.
4. User clicks save

Alternate Flow:

A1. The user wants to record data which isn't specified.
3. The user clicks the '+' button
4. The System adds 3 fields, labelled 'Name', 'Data' and 'Unit'

The Use case continues at **<u>Data Entry</u>** in the main flow.

Post Conditions:

The new data is saved into the database, and the relevant graphs are updated.

2.10.2 ID: UC5

Title: The user views their graphs

Description:

The user views their graphs containing their data.

Primary Actor: User

Preconditions: UC1, UC2, UC3

Main Flow:

1. The user is directed onto the dashboard after logging in.
2. The user can scroll down to look at the specific graphs they want to.

Post Conditions:

The user can view their data, and scroll over the graph to see other information related to the data points.

2.10.3 ID: UC6

Title: The user compares two datasets on a graph.

Description:

The user wants to overlay two data sets, to produce a new graph comparing two measures. For example, blood pressure and caffeine intake

Primary Actor: User

Preconditions: UC1, UC2, UC3

Main Flow:

1. The user is directed to the graphs page once logged in.
2. The user can choose the plus button to create their own graph.
3. The user chooses from the list 'Diastolic Pressure'
4. The user can choose from the list 'Caffeine intake'
5. The graph will change to view both levels
6. The user clicks save.

Post Conditions:

The created graph will then be saved to the user's dashboard, so the user doesn't have to continue creating the same graph every time they want to view the data.

2.10.4 ID: UC8

Title: A user shares their data with their medical practitioner

Description:

The user identifies which professionals they want to have access to their data.

Primary Actor: User

Preconditions: UC1, UC2

Main Flow:

1. The user clicks the profile button.
2. The application proceeds to the profile page.
3. The user clicks the plus next to ‘Sharing data with:’
4. The system creates a text input with a send button.
5. The user enters the email address of the new practitioner, and clicks send.
6. The system prompts the user to ensure they want to add this practitioner.
7. The user clicks accept.

Post Conditions:

The practitioner gets an email alerting them that someone has allowed them to view their data, when the practitioner logs into the system or creates a new account they will be able to view this data as in use case UC9.

2.10.5 ID: UC9

Title: Medical professional viewing the data
Description: The Medical Practitioner logs onto the application to view the data.
Primary Actor: Medical Professional
Preconditions: UC8

Main Flow:

1. The medical professional is taken to the list of patients that has given them access to their data
2. The professional can choose the patient they want to view the data of.
3. The professional is led to the patient dashboard.

Post Conditions:

The professional is able to see the specific data relating to the patient before an appointment.

3 Design

In this section I will define what technologies will be the most appropriate to use for the development of the application. Along with this the design choices about the architecture of the system will be discussed and decided. To achieve the final interface designs, the user interface wire frames are created, evaluated, tested and improved to focus on the user centred design methodology.

3.1 Data Sharing and FHIR

As described earlier, FHIR is a specification for interoperability across different healthcare applications. It defines many things pertaining to healthcare data, such as how information will be exchanged, the format of the information, the infrastructure of the application, etc. (Hl7.org.E, 2018)

If time and resources weren't limited during this project, I would attempt to integrat the application within the NHS. This would enable a professional to have access within the application to the data of all the patients registered to their practice. This would streamline the process for users so they wouldn't have to specifically determine who has access to their data, and also for the practitioner as they wouldn't have to log into a separate application. This won't currently be possible due to inability to access a SmartCard or a reader, and also the inability to achieve all tasks on the Digital Assessment Questions. Many of the questions will require more time to fully complete, but I will aim to complete as many as possible throughout the remainder of the development process. (Health Developer Network A, 2018) (Health Developer Network B, n.d.)

For this to be developed in the future, the application will have to conform to the FHIR standards. By starting the practice now, the applications underlying architecture won't have to be rewritten at a later date.

Currently to be able to control who has access to each patient's data, it would be better for a user to be able to identify which practice they belong to, therefore identifying which practitioners have access to their data. However, FHIR do not have unique identifier for a practice currently represented within their resources. To be able to accurately identify who has access to the data, without changing FHIR data formats unnecessarily, the user will identify specific professionals that they share their data with to simplify this process.

A user must always be aware when they are sharing their data with professionals and must consent to this, as outlined in the NHS Code of Practice. Therefore, before the user adds any practitioner to share their data, they will be prompted to ensure they understand what they are doing, and always have the ability to remove a practitioner from their profile. By removing a professional from their profile, it will ensure that person will no longer be able to view their data. (DH/IPU/Patient Confidentiality, 2018)

3.2 Database

The choice of a database revolves around the storage of FHIR documents and the NHS's choice of Riak, therefore a non-relational database will be used. FHIR supports data in the form of XML or JSON, therefore patient data will have to be stored in one of these types. Throughout the descriptions of the data on the FHIR website, JSON is more widely used and therefore it is easier to learn how to manipulate the data. JSON is also a more human readable format and therefore easier to understand how the data is formatted, and easier to learn. The documents will also have a varying structure depending on what the user is logging, and what graphs have been created. Thus, a non-relational database which is schema less would fit the project better, over a rigid relational database with a schema.

Non-relational databases fit the CAP theorem, only being able to support two of the three attributes: Consistency, availability and partition tolerance. The application being developed will benefit from partition tolerance in the future as the application grows, however the other two don't need to be strongly supported. (Sakr and Gaber, n.d.)

Data for a practitioner does not always have to be completely up to date to view, eventual consistency would be good enough, as the patient in the appointment can inform the doctor if there's been a large change in the past few minutes they should be aware of. Along with this only one person should be able to edit the data at once, the patient who's document it is, therefore availability of the data won't be a problem.

The two main choices for a non-relational database would be MongoDB and CouchDB, these both support JSON documents and have partition tolerance within the database. I would potentially be able to use both databases as I am not familiar with either of them so either will have to be learnt from the beginning. However, as MongoDB queries are more similar to SQL, which I have knowledge of, I will be using MongoDB for this project. (Couchdb.apache.org, 2018) (MongoDB, 2018)

3.2.1 FHIR Representations

The system will use the FHIR representation of a patient to be stored within the database. (Hl7.org.A, 2018) This representation includes name, gender, birthdate, etc. However not all elements need to be included, therefore only the needed attributes have been included in the databases specified below.

Other elements have been added via an extension as specified by FHIR. The description of the extension must be registered online, using a specific format, and then referenced in the document under the attribute 'Extension'. The advised way to register extensions has not yet been implemented by Hl7, therefore 'URL' will contain the name of the extension as a string, and will be described below under 'Extensions'. (Hl7.org.C, 2018) (Hl7.org.D, 2018)

The representation of professionals will use the FHIR practitioner representation, again with altering some attributes to make the database more specific to the application, also described below. (Hl7.org.B, 2018)

FHIR doesn't have a specific format for how to store readings about conditions, therefore a new collection will be used to store this information in an appropriate format detailed in the diagram.

Like with readings, FHIR doesn't have an appropriate way to store login information, to be able to access hashed password efficiently, they will be stored in their own collection.

3.2.1.1 FHIR Patient Representation

The FHIR representation of a patient without any alterations contains many attributes which are not relevant within this application. Elements which have been left out contain: 'deceasedBoolean', 'animalBreed' and 'multipleBirthInteger'. Attributes like these have been left out because they are not relevant to tracking a long-term condition of a patient.

According to the Data Protection Act ‘Personal data shall be adequate, relevant and not excessive in relation to the purpose or purposes for which they are processed.’ These attributes would be classed as excessive for the purpose of the application, and therefore will not be stored. (Ico.org.uk, n.d.)

The basic FHIR representation containing the elements I will be tracking: (Hl7.org.A, 2018)

```
{
        "identifier" : [{ Identifier }], // An identifier for this patient
        "name" : [{ HumanName }], // A name associated with the patient
        "gender" : "<code>", // male | female | other | unknown
        "birthDate" : "<date>", // The date of birth for the individual
}
```

Within MongoDB, the identifier for a document is labelled “_id” and cannot be changed. Therefore, “identifier” will be labelled “_id” instead.

3.2.1.2 Clinician Representation

The basic FHIR representation of a clinician containing the elements I will be tracking: (Hl7.org.B, 2018)

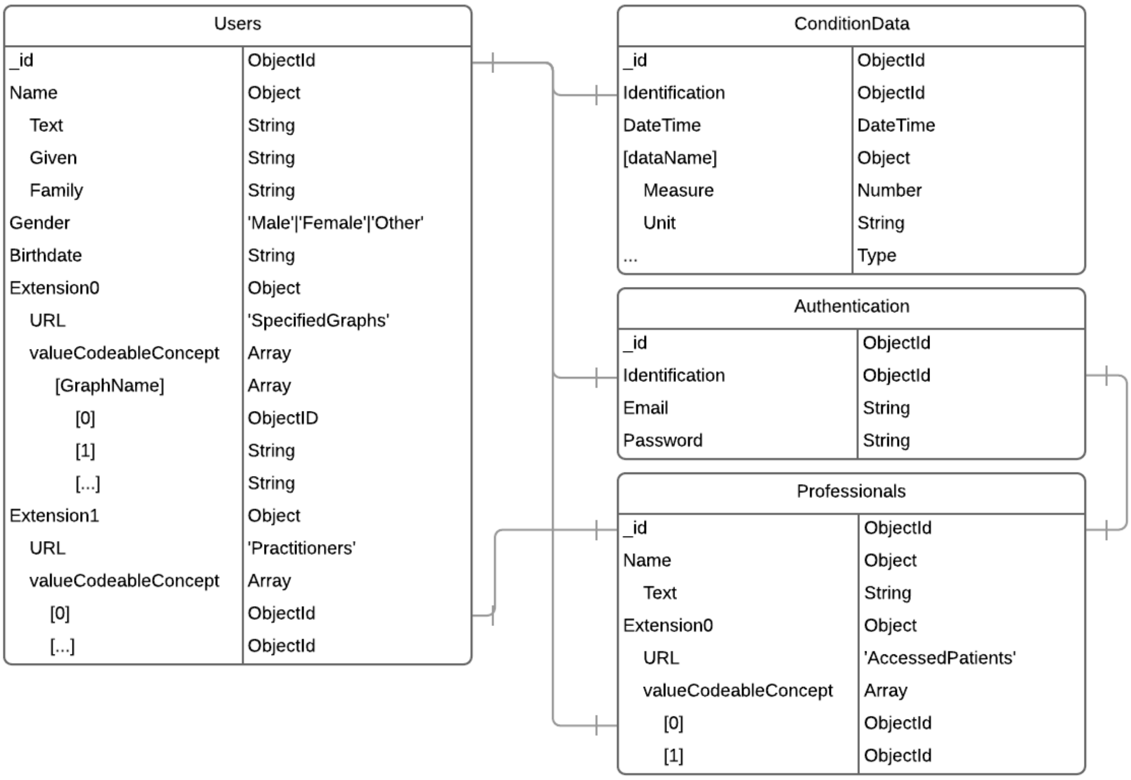
```
{

        "identifier" : [{ Identifier }], // A identifier for the person as this agent
        "name" : [{ HumanName }], // The name(s) associated with the practitioner

}
```

3.2.1.3 FHIR Extensions:

```
StructureDefinition {

        name: SpecifiedGraphs
        status: ‘active’
        description:
                ‘an object containing 0 or more arrays which specify the graphs made by the users of
                the system.’
        type: CodeableConcept

}

StructureDefinition {

        name: Practitioners
        status: ‘active’
        description:
                ‘An array of all the practitioners that have access to a user’s data.’
        type: CodeableConcept

}
```

```
StructureDefinition {

        name: AccessedPatients
        status: ‘active’
        description:
                ‘An array of all the patients that the practitioner has access to.’
        type: CodeableConcept

}
```

3.2.1.4 Database Diagram



Users	
_id	ObjectId
Name	Object
Text	String
Given	String
Family	String
Gender	'Male'|'Female'|'Other'
Birthdate	String
Extension0	Object
URL	'SpecifiedGraphs'
valueCodeableConcept	Array
[GraphName]	Array
[0]	ObjectID
[1]	String
[...]	String
Extension1	Object
URL	'Practitioners'
valueCodeableConcept	Array
[0]	ObjectId
[...]	ObjectId

ConditionData	
_id	ObjectId
Identification	ObjectId
DateTime	DateTime
[dataName]	Object
Measure	Number
Unit	String
...	Type

Authentication	
_id	ObjectId
Identification	ObjectId
Email	String
Password	String

Professionals	
_id	ObjectId
Name	Object
Text	String
Extension0	Object
URL	'AccessedPatients'
valueCodeableConcept	Array
[0]	ObjectId
[1]	ObjectId

3.3 Backend

Java and python would be two options for a backend, these two technologies would be advantageous in applications which would need a lot of background processing and complex algorithms. However, as the application will just need to retrieve and upload data to and from the database, PHP will also be a good option. As I have more experience with PHP, it has helped me come to the conclusion to use this as the backend technology. (Php.net, 2018)

Having a familiarity with PHP will enable me to set up the application, and also allow for development to be faster. As I have a limited development time, and also need to learn MongoDB in this period, this will be favourable. It has a large community, with a lot of help concerning MongoDB as issues arise during development. The flexibility of PHP will also allow me to use it to create the full functionality of the application and the low barrier to entry will allow me to develop this skill more easily.

3.4 Frontend

JavaScript will be used for server-side scripting purposes, this will enable the web application to be interactive and responsive to the user. A framework won't currently be used to prioritise the readability and simplicity of the code base.

3.5 Graphs

The graphs which will be created will be line graphs, with the variables on the y axis and time on the x axis. This choice is for users to be able to view their data easily over time, which what the personas were trying to do. In the future, users of the application can be questioned to understand what other graphs they may need or want, and further development can continue, however for the 12-week period only line graphs will be created.

The main choices for JavaScript graphing technologies are D3.js and HighCharts.js. D3 has a steep learning curve, but will allow the graphs to be fully customisable, has a very large community and lots of examples to help the development of my own graphs. HighCharts also has a large community, and enough customisation to create most of what I have designed, however there aren't many examples to help with the creation of these graphs. Interactivity was the main deciding factor between the two choices, HighCharts being more difficult to be able to create the level of interactivity I needed with the created graphs, this led me to the choice of D3.js.

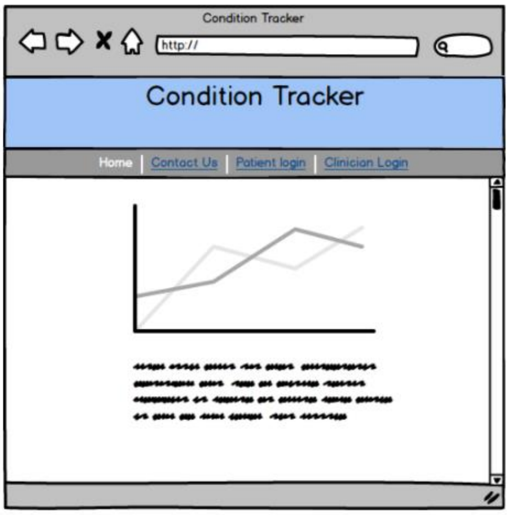
D3 uses SVG as a language to be able to append different graphic elements to a background. This is similar to JavaScript and HTML, so will be easy to learn and get started with the development.

3.6 User Interface

The wire frames below are an outline of the interfaces that will be created during this project. To ensure the usability of the system I will be undertaking user testing. These tests will enable me to find out whether the system is laid out appropriately and it is intuitive before development.

All screens used in the use cases have been implemented in wire frames to show how the main functionality of the application is designed to work:

Use Case	**Wire Frames Involved**
Use Case 1: Creating a profile	Home, Create User
Use Case 2: User logs in to profile	Login
Use Case 3: User adds data to profile	Graphs, Log a Reading, Create New Field
Use Case 4: Change profile Information	Graphs, Profile
Use Case 5: User views their graphs	Graphs
Use Case 6: User compares 2 datasets	Graphs, Create a Graph
Use Case 7: Change saved Data	Data
Use Case 8: Sharing Data with practitioner	Profile
Use Case 9: Practitioner Views Data	Patients Graph

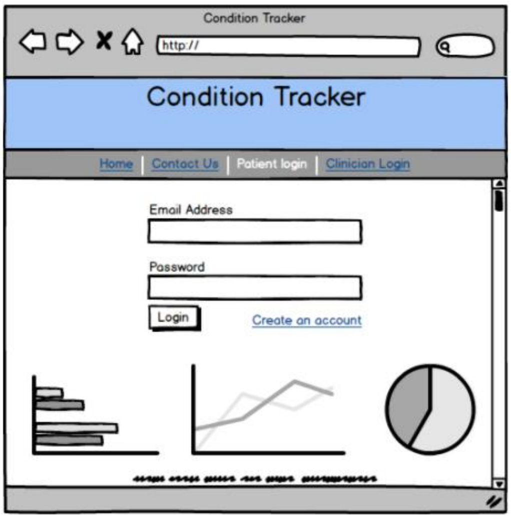


Home page

The landing web page of the system.

Here the user will be able to read about what the system is, how to use it, and sign up to the system.

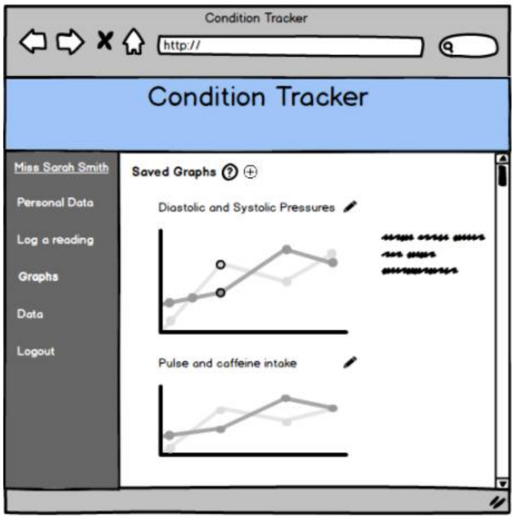
The user can click user log in to access the application.



User Login page

How the user logs into the system, or creates an account.

This page will also show the user some of the graphs and functionality of the web application under the log in, to showcase the application.



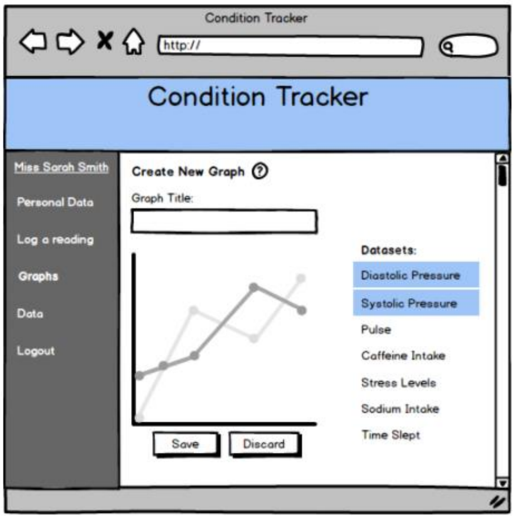
Graphs Page

Once the user has logged in, they will be taken to their graphs page.

These graphs will be the ones they have created themselves, they will be able to edit them using the pencil button on the right. To make this intuitive for users who don't spend a lot of time online, when the button is rolled over text will display saying 'edit graph'.

With these graphs, the user is able to scroll over data points to see information relating to it, that is not displayed on the graph to the right. The data will not be shown automatically so as not to clutter the page, and be distracting to the users.

Each page within the application will have a help button next to the page title, this will initiate a help popup showing the user the different functionality of the specific page.



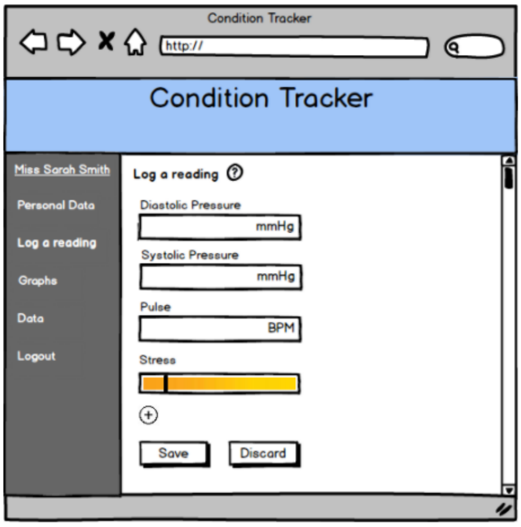
Create a Graph Page

When creating a new graph, a user is able to label their own graph, and add many different datasets to it.

As users add datasets, the graph will update with the user's specified data, and the selected datasets will become highlighted in the list. This will make the graph interactive to give the user confirmation of their actions.

Once the user is happy with the created graph, they can save it, and it will be added to their graphs screen.

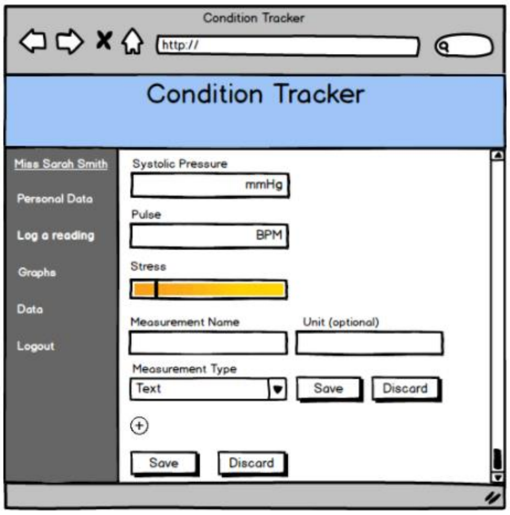
If a user clicks discard, they will be prompted to ensure that this is the right decision. If it is, the user will be redirected to the graphs page which will not be updated.



Log a Reading Page

To log a reading, the user clicks 'log a reading' in the navigation bar. The user will be presented with all the fields they are currently tracking to create a new reading.

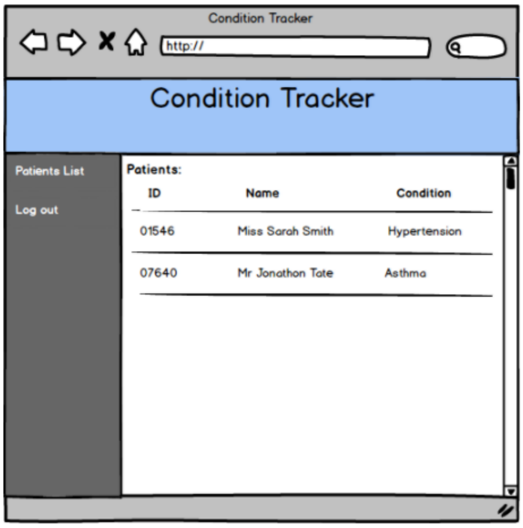
They will be able to create a new field to record by clicking the plus button at the bottom of the screen, and save the reading by clicking the save button. When clicking discard, the user will be prompted if they are sure about their decision, to make sure they wanted to discard their reading.



How to create a new field

When the plus is chosen to create a new field, the user will be prompted to enter the new measurements name, the optional unit, and how the measurement will be recorded. The user has the choice of two measurement types, Number or Text.
The user can save the new measurement, and it will then be saved on the log a reading screen for the users to fill each time they enter a new reading.

If the user clicks on the plus, or the bottom save or discard, they will be prompted that the unsaved data will be lost to ensure this is the action they want to take.



Patients:

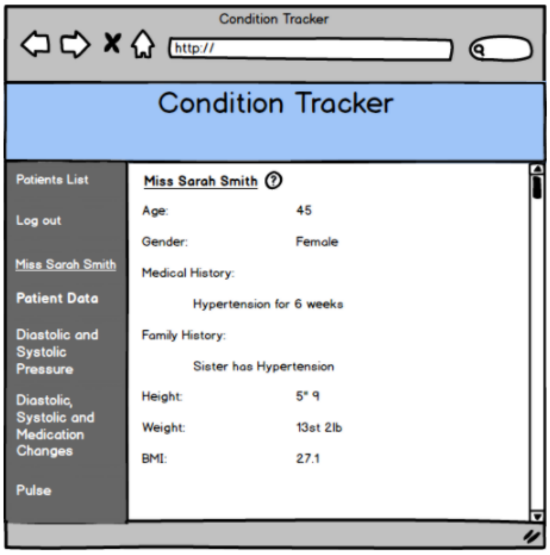
ID	Name	Condition
01546	Miss Sarah Smith	Hypertension
07640	Mr Jonathon Tate	Asthma

Patients Page

The medical professional will see a different profile to the patients.

The clinicians home page will be the patients screen. It will enable a medical professional to see all the patients which have shared their data with the clinician.

The clinician can choose the patient they want to view the data of.

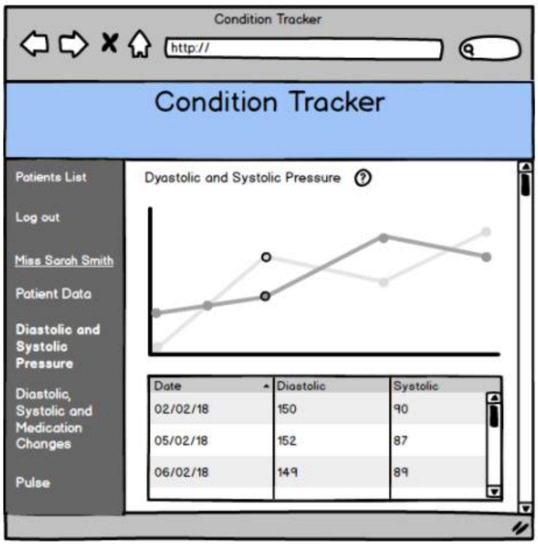


Miss Sarah Smith

Age:	45
Gender:	Female
Medical History:	
Hypertension for 6 weeks	
Family History:	
Sister has Hypertension	
Height:	5" 9
Weight:	13st 2lb
BMI:	27.1

Patient Profile

Once a clinician has chosen the patient, they are taken to the patient data screen containing all their information. From here they will be able to navigate to the other graphs.

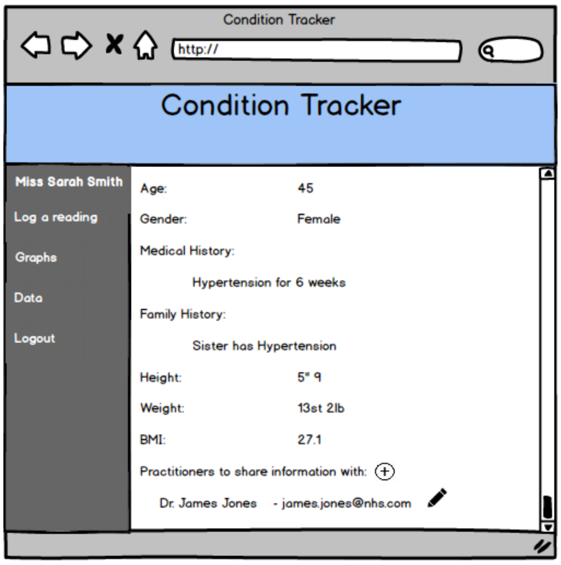

Condition Tracker
http://
Condition Tracker
Patients List
Log out
Miss Sarah Smith
Patient Data
Diastolic and Systolic Pressure
Diastolic, Systolic and Medication Changes
Pulse
Dyastolic and Systolic Pressure

Date	Diastolic	Systolic
02/02/18	150	90
05/02/18	152	87
06/02/18	149	89

Patient Graphs Page

The graphs that the clinician can see are listed in the navigation bar.

The page will show the users data in a graph, and a table showing the raw data. The professional will be able to roll over the data points, this will highlight the relevant rows in the data table.



Age: 45

Gender: Female

Medical History:

Hypertension for 6 weeks

Family History:

Sister has Hypertension

Height: 5" 9

Weight: 13st 2lb

BMI: 27.1

Practitioners to share information with: ⊕

Dr. James Jones - james.jones@nhs.com

Users Profile Page

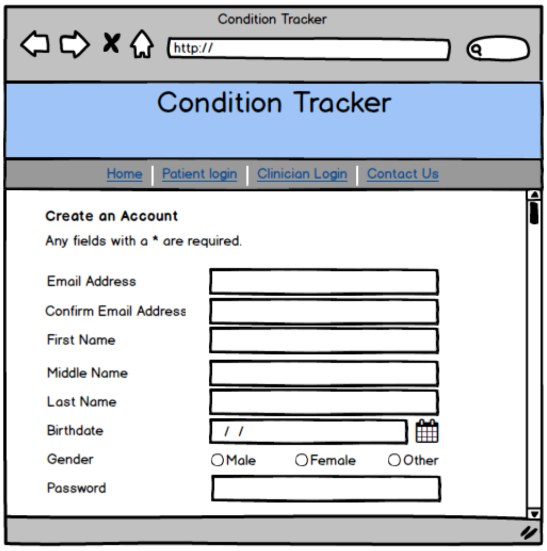
The user can see all of the entered information about themselves, they are also able to see which practitioners have access to their data.

At the top of this page, next to the title there is a pencil button to allow the user to edit any data stored on their profile. Once this data has been changed, it will be updated within the database.

When clicking the plus button, the user is able to add practitioners who are able to view their data. When the plus button is clicked, the user is warned that the clinician will be able to view all data entered on the account including profile information. And ensure they want to complete this action.

They are then prompted to enter the clinicians email address, and this user will show up on the clinician's patients page. If the user changes their mind, they can click the edit button, and either edit the email they have given, or delete the clinician from the list.

The user can have as many clinicians as they need listed here.



Create an Account

Any fields with a * are required.

Email Address

Confirm Email Address

First Name

Middle Name

Last Name

Birthdate / /

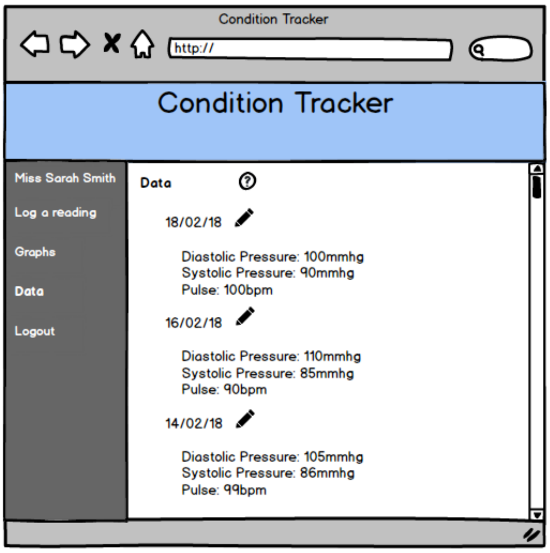
Gender ○Male ○Female ○Other

Password

Create Account Page

For a user to create an account, they will have to enter all their personal details.

The users email and password will have to be entered twice to ensure they are correct. Any details which are entered here can be changed again after the account has been made.



18/02/18

Diastolic Pressure: 100mmhg
Systolic Pressure: 90mmhg
Pulse: 100bpm

16/02/18

Diastolic Pressure: 110mmhg
Systolic Pressure: 85mmhg
Pulse: 90bpm

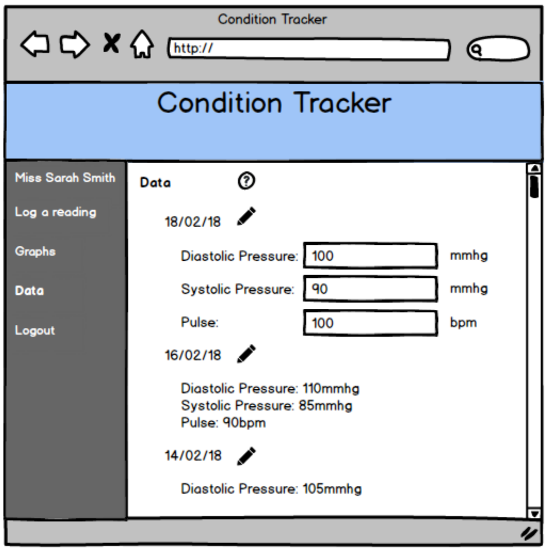
14/02/18

Diastolic Pressure: 105mmhg
Systolic Pressure: 86mmhg
Pulse: 99bpm

User Data Page

All data the user has entered will be available to view on this page.

This will allow the users to see the raw data that the graphs are based on, and edit any mistakes they may have made.



18/02/18

Diastolic Pressure: 100 mmhg
Systolic Pressure: 90 mmhg
Pulse: 100 bpm

16/02/18

Diastolic Pressure: 110mmhg
Systolic Pressure: 85mmhg
Pulse: 90bpm

14/02/18

Diastolic Pressure: 105mmhg

Edit Data Page

To edit the data, a user can click the pencil button to edit any particular entries.

The user will be given text boxes to edit the data. Any data fields which the user has not already filled in, will appear below the already entered data for the user to add, if they need.

3.7 Heuristic Evaluation of Design

As before, Jakob Nielsen's 10 usability heuristics will be used to evaluate the proposed system. This will be used to ensure there aren't flaws in the design which will can cause issues for new and regular users. (Nielsen, 1994)

Visibility of System Status

The heuristic of visibility of system status is to ensure the user knows how the system is working, and that there's appropriate feedback to the user. The designed system will always allow the user to know where they are within the navigation of the application. With all pages on the main website, user's and clinician's application, there is a clear navigation bar with the current page underlined. If users are confused about where on the application they are, they will always be able to view the navigation bar, and find which part of the application they are using.

Along with this, when making a change to the system, for example changing their profile information, or creating a new graph, the user will be notified that the changes have been saved. This allows a confirmation to the user, that they have been able to complete the action, not leaving them in doubt if the process has been completed or not.

Match between system and real world

The system should be understandable to the user, using their own language and concepts. Information should be in a logical order for the users. Because of this, the system is customizable through addition of graphs and data fields to allow the user to choose their own units and layouts. This will enable the system to be familiar to them as they will be able to customize it however they want.

User Control and Freedom

This heuristic is about giving the user control to exit processes easily. The user is always given the navigation bar on the left of the page to be able to exit any processes and go back to another page. The user is also given the option to logout at all times, this enables a user to exit the system completely with minimal effort. If the user exits in the middle of a process, they will be given a dialog box ensuring they want to quit the process, and that this isn't a mistake.

Consistency and Standards

The system should follow previous well-known platform designs to make it easy to understand how to use. When designing the application, Apple human interface guidelines and Microsoft's design basics were taken into account, for example ensuring there is no horizontal scrolling to read text. The designed system follows current conventions of a web application, such as the navigation bar down the left-hand side, the active page on this will be underlined to show where the user is currently. Along with this, when the other options are rolled over, they will be highlighted to show they are hyperlinks. Other conventions such as the header and footer, will be found at the top and bottom of the page. (Developer.apple.com, n.d.) (Docs.microsoft.com, n.d.)

On a few of the pages, there are four buttons which have been reused throughout the application, the question mark, the pencil, the plus, and the minus. By consistently using these over all pages, the user will not have to continuously guess the meaning of the images on the pages. These are also well-known buttons to make it as intuitive to use as possible.

Error Prevention

As mentioned previously, the user is able to quickly exit any processes. Before the user quits by pressing the 'back' button or another navigation button, a popup will ask the user if they're sure they want to quit. This will ensure the data they have entered will not be lost if they weren't supposed to quit.

Recognition rather than Recall

By allowing for recognition rather than recall, it allows the user will be able to navigate and use the web application faster. This would fit into Alice's persona, allowing her to complete her tasks quickly and efficiently on a day to day basis.

With the professional’s area of the website, by having the patient's name always shown on the left-hand side in the navigation so the practitioner will always be able to read whose data they are viewing no matter what page they are on. With practitioners and patients, you are always able to see what page they are on as the active page on the navigation will always be highlighted, and a title will always be shown at the top of the page.

Along with this by using buttons which users will recognize, such as the help button being a question mark, this enables the user to intuitively use the site rather than relying on the provided help to understand the site.

Flexibility and Efficiency of Use

Flexibility in the system is shown by the user being able to customize the existing graphs and data fields, and also create new ones to change the application to work for them and their condition. Emphasis has been placed on the flexibility of the site to allow the user to customise their application as much as they need.

Aesthetics and Minimalist Design

By having a minimal design of the application, it allows the users to navigate with more ease, and help them understand how to complete tasks more effectively as they aren’t overloaded. This would help to fulfil one of Erik’s goals who wanted an easy way to view if his lifestyle changes have affected his blood pressure. By having a minimal screen, it helps Erik to simply see his graphs without extra clutter and information.

A simplistic design helps a user to be able to complete their tasks, the site is easier to understand and navigate. Any text is minimalistic, and does not include extra information. The designed application only shows the minimal amount of information to as not distract the user from more important information. For example, having data about the graphs available only when rolling over the graphs, ensures extra space isn’t taken up by non-relevant data.

The elements on the page will be spaced out to not appear cluttered, and allow the user to explore the application more easily.

Help Users Recognize, and Recover From Errors

Error messages will be shown across the application when things go wrong with the user. For example, before a user submits the login form it will be validated to ensure their email and password are valid. This saves them time, instead of submitting and waiting for the application to reject their credentials, they’re able to amend them before submission.

Another way to help users recognize an error, is when a user is entering new data, if not all the fields are filled in, the system will prompt the user on submit if they meant to leave some fields blank.

Along with this, if a user clicks out of a form they will be prompted to ensure that is what they were meant to do. This stops the user from making accidental errors and losing data they have entered and not submitted.

Help and Documentation

On any page there will be a question mark help button next to the title, this button will give help pertaining to the current page, and how to use the page. The user will have to actively click on the button to show the help, to not hinder the users who know how to use the application. For the target audience of people above 65, they may not be as computer literate as the younger generation, and therefore extra help navigating the functionality of the application may be needed.

3.8 User Testing

This user testing will be to see how a user reacts to the system and how they are expecting to be able to use it. I will be testing the interface on three users. A computer literate adult of the age of 23, a 53 year old with less computer literacy, and a 64 year old with a mid-range computer literacy to see how different people react to the system. Two of the three users are within the target audience age range, so will give me a representation of what the personas would think of the application. This will allow me to answer 'yes' to questions surrounding user testing within the Digital Assessment Questions.

Users 2 and 3 are also similar to the personas created, this will be to understand if I have fulfilled the goals of these users. Ideally this would have been completed with many users similar to the personas who have hypertension, however the time limit has constricted this and will only be completed with three.

During the user testing, I will firstly explain the system and what it will be used for, and the functionality of the application. I will then show them the wire frames, and ask them to navigate and use the system whilst thinking out loud. By not offering them responses to any questions they may have when using the site for the first time, it will allow me to see how users are expecting to use the interface, and where the system does not match this expectation.

After the user test, there will be a conversation to understand where they would improve the system, why they made mistakes with the navigation, and how to increase the usability of the web application.

Once all user tests have been carried out, I will change the wire frames of the system accordingly to be able to use in the implementation of the system.

3.8.1 User 1:

User 1 is a 23 year old female software engineer, she has a high level of computer literacy to review the ease of use, and design of the site.

She thinks that the layout of the site is clean, and easy to understand. She found it easy to navigate, and simple to access and use the whole range of functionality on both the patient application and the professional application.

One thing she found that could work better was when adding another field in the 'log a reading' page. The page has two sets of save and discard buttons, one to save or discard the new measurement, and the others are to save and discard the entire reading. She thought that this could be misleading, and without being able to click on the help button to have it explained, can be difficult to use for the first time.

Another thought was that the button for the user to log out is in a different place than for the medical professional. This may be a problem with the consistency of the site.

An additional problem with consistency is the two placements of the plus arrow on the 'log a reading' page and on the 'graphs' page. On one the plus is at the bottom and on the other it is at the top.

She also picked up on the fact that a long-term user may have created a lot of fields that they want to track, and created a lot of graphs that they have saved. The site wouldn't be optimised for these types of users. Having to scroll through all the graphs they've created would be annoying for a regular user as there is no a way to order or filter the graphs. When creating a new graph, the user has to scroll through all their datasets and that can become time consuming.

3.8.1.1 Improvements:

One area which will be improved upon using User 1's feedback will be adding a new field to the 'log a reading' page. A way to improve upon this may be to split this process into two, having the 'data entry' screen, and the 'create a new field' screen. When a user clicks the plus button, the system will grey out the previous entry fields so the user can still see the values they've entered. The system will have the same layout of the new field entry, but only one set of buttons which say 'continue' and 'back'. Once a user clicks continue, the entry fields to create a new dataset will be replaced with the newly created entry, and the previous entries will be usable again. The 'save' and 'discard' buttons at the bottom will then refer to saving or discarding the entire entry.

Another improvement for the long-term users of the site would be to implement some time saving features. On the graph page, being able to filter the created graphs to find the specific ones the user is looking for easily will speed up the searching process. This filter will work on the datasets used in the graphs.

Also in the 'create a graph' page, a search box at the top of the datasets options would enable users to find datasets they're looking for quickly and easily, without having to scroll through them all. These improvements will save time for long term, regular users.

Both of the plus buttons can be placed at the top of the page, to solve the consistency issue. This can be reviewed at a later date to see if this helps with the usability of the application.

User 1 pointed out the consistency issue of the placement of the 'log out' button, however this was purposeful. On the user's application, it's placed at the bottom of all their options which is where you would expect to find the 'log out' button. However, on the professional's application, the navigation bar is split into two. The top section is their own options, and the bottom section refers to the patient, this is split by the underlined name of the patient. Because a user and a professional will generally not see both applications this won't be a usability problem, and therefore can be left for the time being.

3.8.2 User 2:

User 2 is a 53 year old receptionist with a lower level of computer literacy, to be able to test the interface on a user in the target audience range.

Overall, the user found the wire frames easy and clean to use. She appreciated the titles on each page to help with her navigation, and understanding where on the application she was. She liked the colour choice of a pastel blue and greys, that the contrast was good to see without the help of her glasses.

However, she didn’t understand how to add a new graph or reading whilst using the application, and felt that the functionality could be more obvious to new users. She overlooked the plus buttons at the top of the screen, not even noticing them. When asked what she thought they meant, she didn’t know.

3.8.2.1 Improvements:

One improvement which was pointed out, was how the functionality of adding new graphs and readings can be more obvious. Adding rollover text to the buttons will help with this, if a user rolls over the plus button it will show the text ‘Create a new reading’ or ‘Create a new graph’. This will make it more obvious for a new user to understand what these buttons do. In the help functionality on the two pages, it will also make obvious what these buttons are designed for.

3.8.3 User 3:

User 3 is a 64 year old, he spends a lot of his spare time on computers so is computer literate and is also in the target audience range.

With the use of the navigation bar in the website before logging into the application, he found that the ‘contact us’ button is most commonly found at the end of the navigation bar. It may be unintuitive to have the button in the second position.

He also found that it was confusing to have the profile button under the name, without the name being a hyperlink. He expected the name to lead to the profile page, and didn’t understand that the name was just a header and why he would need to be reminded of his own name.

Having the current page bold and the name underlined also confused him, initial reaction was that he was on the user data page as it was underlined and the bold of the current page wasn’t obvious.

Another comment was that if he wanted to enter all of his family and medical history into his profile, that he would have to scroll through all the data to be able to read values underneath this. This could become annoying if he needed to access this data fairly regularly.

With the professional side of the application, he pointed out the clinician doesn’t have a search bar to find specific patients and would need to scroll through all of them when they had an appointment. This will become an issue if a practitioner has a lot of patients using the application.

3.8.3.1 Improvements

Firstly moving ‘contact us’ to the end of the navigation bar on the website, to improve the ease of use for the users. Placing things where users will expect them to be located.

Adding a search bar into the professional’s patient’s page will allow the practitioner to easily and quickly find patient data they want to view.

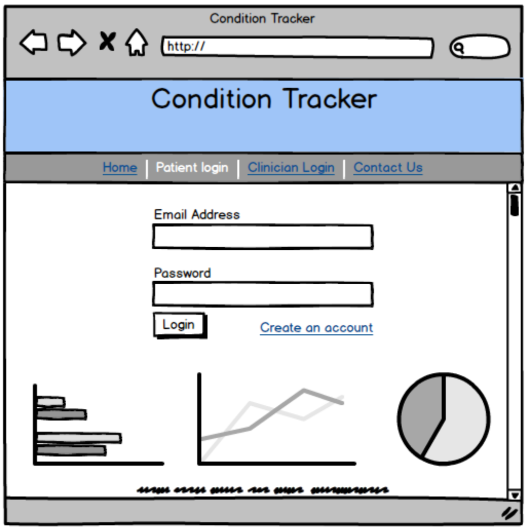
To remedy the profile and name situation, by removing the user data button and making the users name the link to their profile will be more intuitive. Keeping the name the same format as the other buttons on the navigation bar. By restyling the links so the active page is underlined and rolling over the other links will underline them, will solve the problem of the confusing formatting of the navigation bar, as mentioned previously.

The problem with the profile page being too lengthy can be solved by having these headers as drop downs. A small arrow next to the header, indicating whether the data is being shown or is just the title, can be shown and will compact the page if the user doesn't want to see that data.

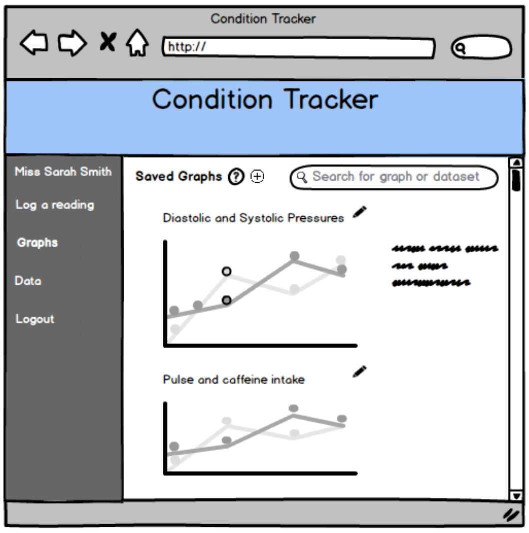
3.9 Final Design

3.9.1 User Application

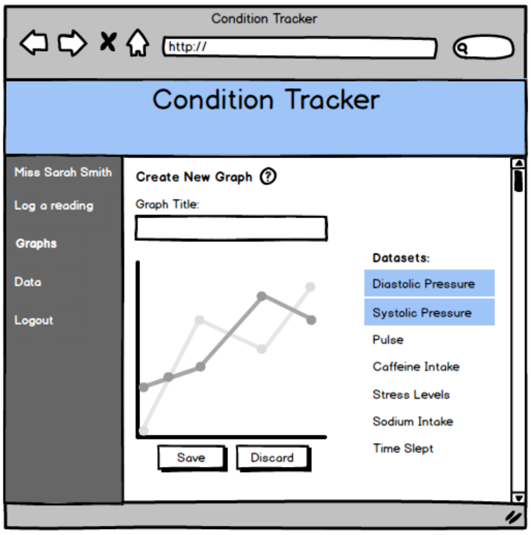
User Login



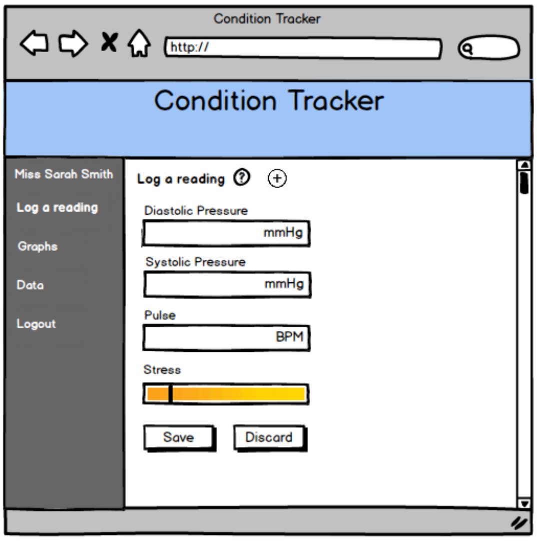
User Graphs



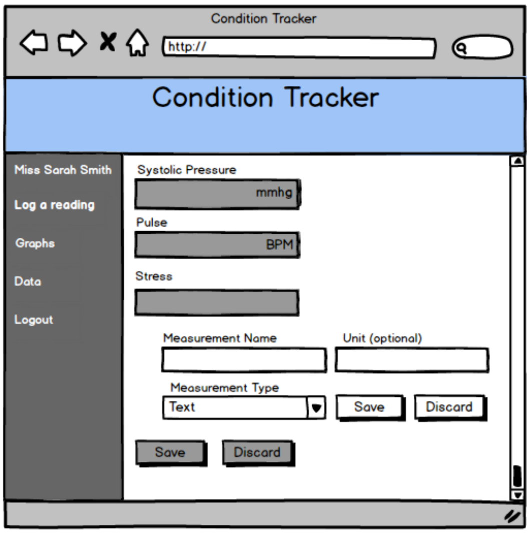
Create New Graph



Log a Reading



Create a New Field



Systolic Pressure mmhg

Pulse BPM

Stress

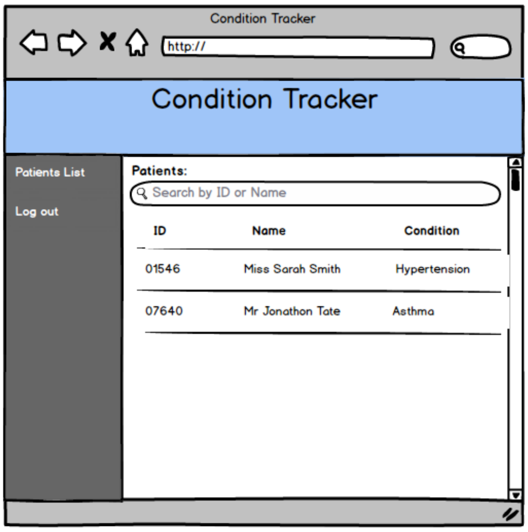
Measurement Name

Unit (optional)

Measurement Type Text

3.9.2 Professional Application

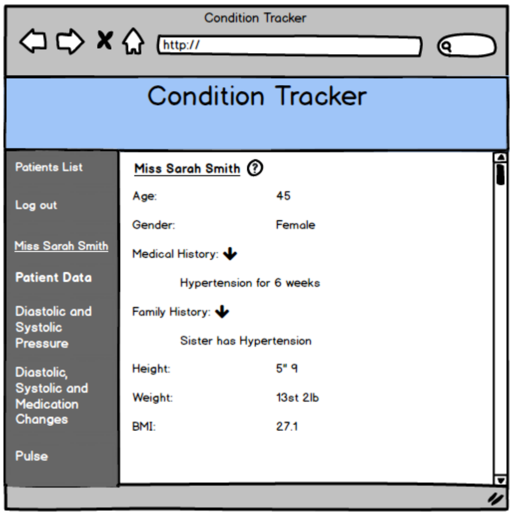
Patients



Patients:

ID	Name	Condition
01546	Miss Sarah Smith	Hypertension
07640	Mr Jonathon Tate	Asthma

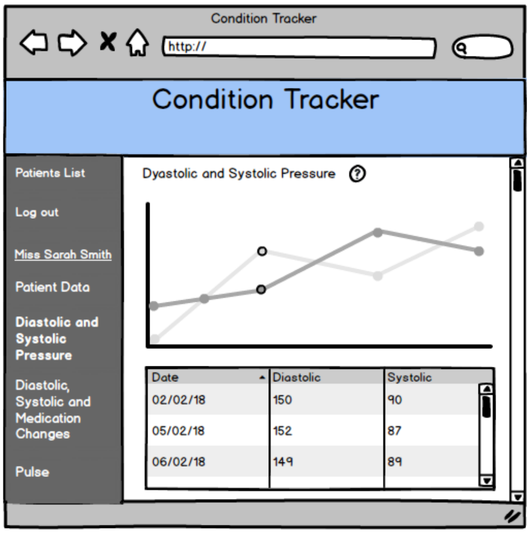
Patient Data



Miss Sarah Smith

Age:	45
Gender:	Female
Medical History:	
	Hypertension for 6 weeks
Family History:	
	Sister has Hypertension
Height:	5" 9
Weight:	13st 2lb
BMI:	27.1

Patient graphs



Condition Tracker
http://
Condition Tracker
Patients List
Log out
Miss Sarah Smith
Patient Data
Diastolic and Systolic Pressure
Diastolic, Systolic and Medication Changes
Pulse
Dyastolic and Systolic Pressure

Date	Diastolic	Systolic
02/02/18	150	90
05/02/18	152	87
06/02/18	149	89

4 Implementation

This section focusses on how the application has been developed. Explaining how to navigate the file structure of the codebase, and how the main functionality of the application is achieved using the technologies previously chosen.

4.1 High Level View

Throughout the codebase, the documentation is provided to ensure the code is maintainable and readable. Along with this, a consistent naming convention of camelCase is used throughout the front end, back end and database. This choice has been made as it's the same design choice that FHIR has chosen, therefore it's logical to carry throughout the remainder of the project.

As described above, the database has 4 collections: authentication, conditionData, professionals and users. Documents can be added and modified through the application using the php scripts created. The php gives a standard way to upload and change the documents, however allowing a lot of flexibility to the users. The documents are designed to store as many recorded fields as the user wants to, and allows them to record this with any unit. This is the same for the graphs, it enables them to overlay as many datasets, in as many graphs as they would need to.

For the backend, php has been used. This controls the way the database is modified. I've created specific functions to be able to retrieve and change data in specific ways, this is to control the interactions with the database. The design has been kept simple to try and keep the code readable and maintainable.

The frontend is made up of HTML, JavaScript and CSS. These files don't interact with the database, but pass data down from the user to the PHP. The user interface has been made interactive to enable a user to customize the application as much as they want. One large piece of functionality in the front end would be the forms, these have visible and hidden elements to pass data through the app. The frontend also gives the user the customisability to personalise the application to their standards.

The visualisations have been made using D3.js, as mentioned previously. This functionality is only included in a few specific files. One JavaScript file contains all the functionality to create and edit graphs, and these functions are called throughout the web pages to their specifications and needs. These functions are made to be reusable for all the graphs in the application, allowing as many datasets to be added, and any title to be given.

D3.js was a difficult technology to use with a steep learning curve. To make a graph, an svg is appended to the page, and then the x axis, y axis, title, labels, and the data have to be appended to the this separately. This requires a lot of lines of code for little functionality, this took me time to make these functions extensible and reusable.

4.2 File Structure

The application is held within the file structure:

/resources

Contains the PHP server-side scripts. These scripts connect to the database, add, delete and edit documents contained within the collections. Contains how the client side can interact with the database to ensure this is done in the correct way.

Also contains the library for into.js used for the help on the pages.

/Static

/images

Contains all the images for the application in one centre location so they can be easily replaced and maintained in future.

/Views

The client-side code, contains majority of the functionality of the website and application.

Contains the web sites HTML.

/App

Contains the web applications HTML.

/css

Contains all CSS for the application and website.

/JS

All the JavaScript for all the dynamic webpages.

Contains the scripts for creating, editing and configuring graphs.

4.3 GitHub

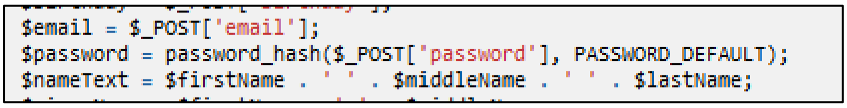
A source control is needed for a project, so that if the computer mainly used for development becomes unusable, there is a backup of the program. A source control will also keep previous versions of the codebase, if a problem occurs it can be reverted to a former version, or can be used to find which version of the code the problem is originating from.

Git has been used specifically over other options such as SVN mainly due to having previous experience. It would have taken time to learn a new source control, for not many benefits. Git would also allow me to develop from multiple computers easily.

https://github.com/reeza1/FYP

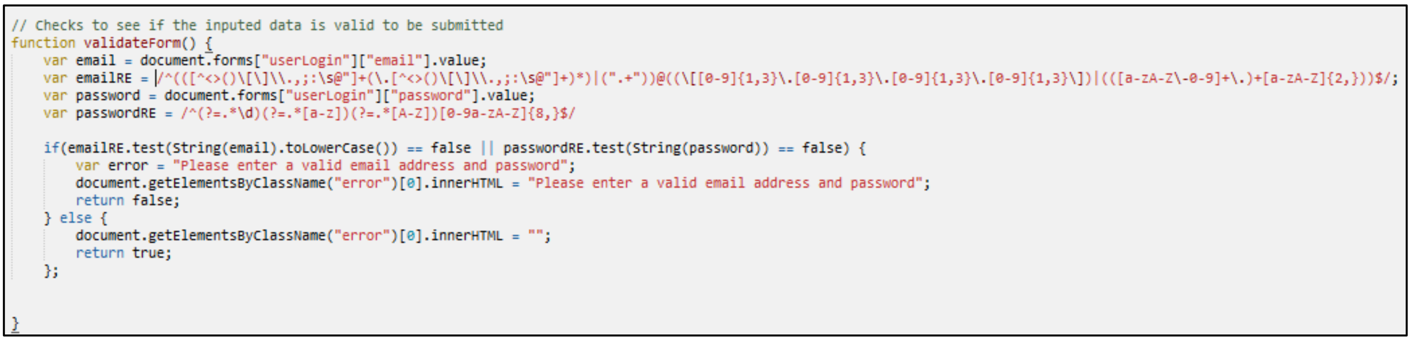
4.4 Login

When creating a new account, the user must provide the necessary details to the application including birthday, gender, etc. They must enter their email address and password twice to confirm that these haven't been misspelt. This data is then sent to the 'Users' and 'Authentication' collections, the passwords are hashed and salted using PHP's Password Hashing API. This uses the BlowFish hashing algorithm as it's more computationally expensive than MD5. It would take longer to break this algorithm whilst still being scalable if an attacker wanted to get the passwords stored. (Php.net, 2018)



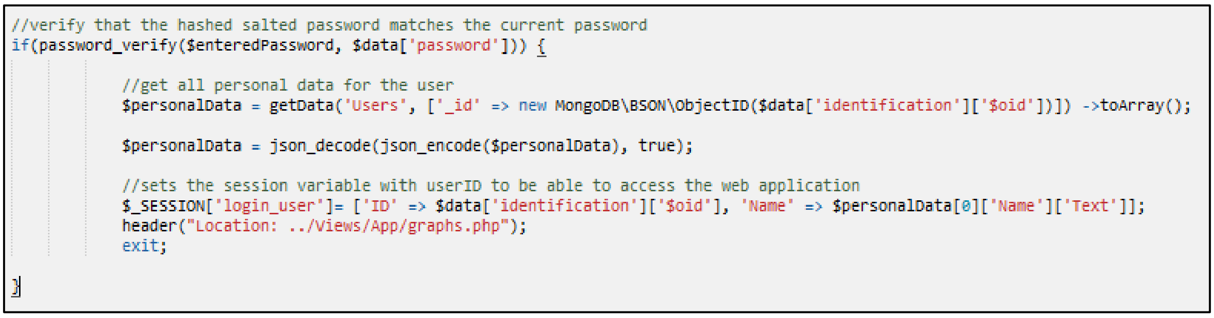
```
$email = $_POST['email'];
$password = password_hash($_POST['password'], PASSWORD_DEFAULT);
$nameText = $firstName . ' ' . $middleName . ' ' . $lastName;
```

During login, the application will use JavaScript to ensure the user has entered details which are in the correct format before submitting to the server. This form validation uses regular expressions to determine whether it's the correct format.



```
// Checks to see if the inputed data is valid to be submitted
function validateForm() {
    var email = document.forms["userLogin"]["email"].value;
    var emailRE = /^(([^<>()\[\]\\.,;:\s@"]+(\.[^<>()\[\]\\.,;:\s@"]+)*)|(".+"))@((\[[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\.[0-9]{1,3}\])|(([a-zA-Z\-0-9]+\.)+[a-zA-Z]{2,}))$/;
    var password = document.forms["userLogin"]["password"].value;
    var passwordRE = /^(?=.*\d)(?=.*[a-z])(?=.*[A-Z])[0-9a-zA-Z]{8,}$/

    if(emailRE.test(String(email).toLowerCase()) == false || passwordRE.test(String(password)) == false) {
        var error = "Please enter a valid email address and password";
        document.getElementsByClassName("error")[0].innerHTML = "Please enter a valid email address and password";
        return false;
    } else {
        document.getElementsByClassName("error")[0].innerHTML = "";
        return true;
    };

}
```

Once submitted, the PHP script will search the Authentication collection to find documents with the attributes 'Email' and 'Password' which match the email and the hashed password submitted. If there is one returned document, a PHP session will be started storing their user ID which is an automatically created ID from MongoDB. A user cannot access the web application without a user ID in their PHP session, they will be redirected back to the login page.



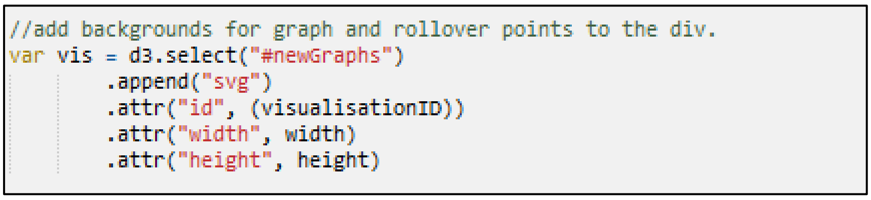
```
//verify that the hashed salted password matches the current password
if(password_verify($enteredPassword, $data['password'])) {

            //get all personal data for the user
            $personalData = getData('Users', ['_id' => new MongoDB\BSON\ObjectID($data['identification']['$oid'])]) ->toArray();

            $personalData = json_decode(json_encode($personalData), true);

            //sets the session variable with userID to be able to access the web application
            $_SESSION['login_user']= ['ID' => $data['identification']['$oid'], 'Name' => $personalData[0]['Name']['Text']];
            header("Location: ../Views/App/graphs.php");
            exit;

}
```

To logout, the PHP session is cleared, and the user will have to log in again.

4.5 Graphs

D3's versatility means that the graph needs to be created in sections, and that nothing is ready made to be used. Because the web application only uses line graphs, and the x axis always show time, one function has been written to be reused for all graphs.

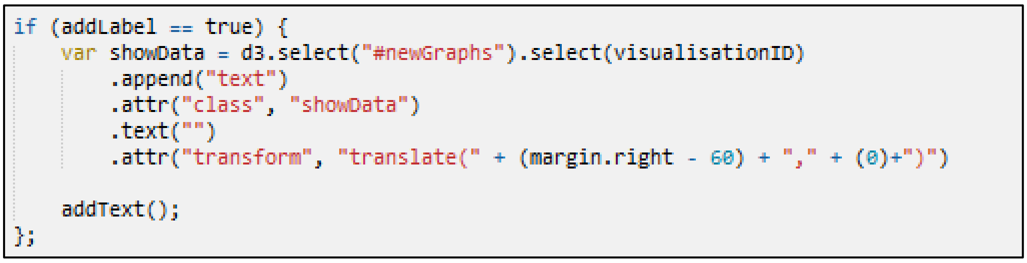
Firstly, an svg is created to append all elements to as the background of the graph.



```
//add backgrounds for graph and rollover points to the div.
var vis = d3.select("#newGraphs")
        .append("svg")
        .attr("id", (visualisationID))
        .attr("width", width)
        .attr("height", height)
```

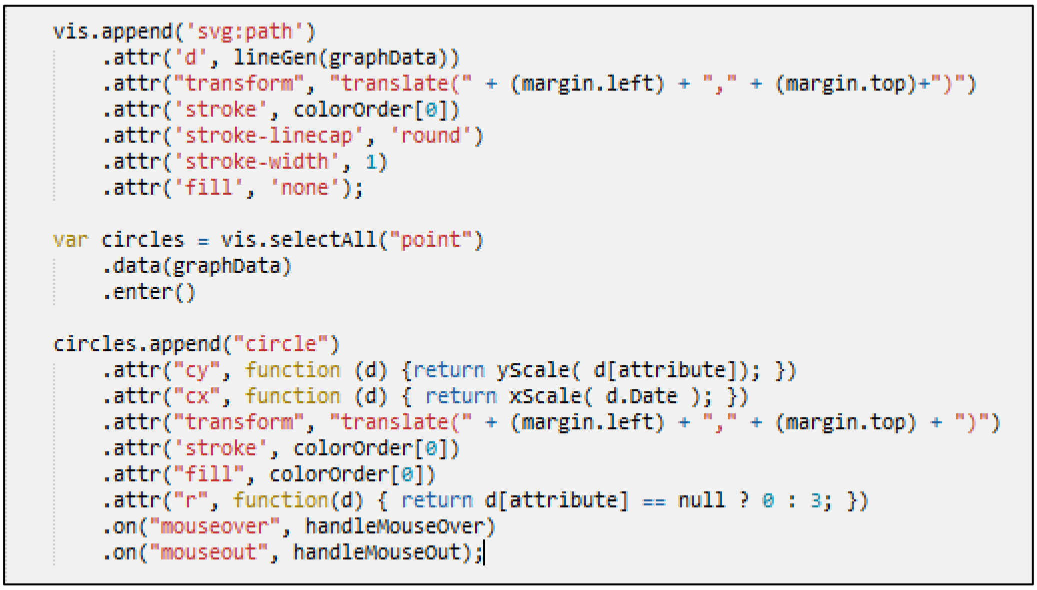
The function takes the data, sorts it in date order, and finds the highest and lowest values to create the y axis. Then takes the first and last value to find the minimum and maximum dates to create the x axis.

If the parameter 'addLabel' is set to true, the function creates a text area on the right for the rollover of data points. This is for the graphs on the main 'graphs' page, when creating a new graph, or editing, this can be set to false.



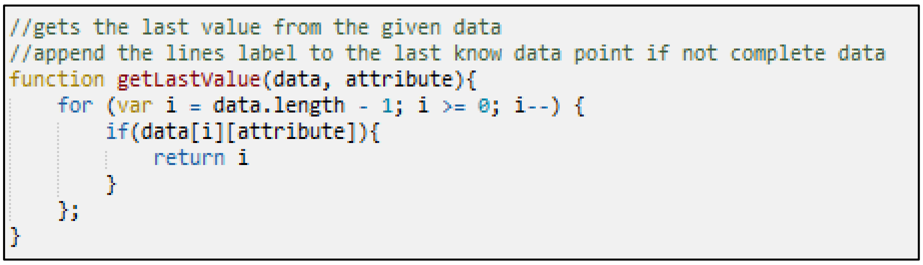
```
if (addLabel == true) {
    var showData = d3.select("#newGraphs").select(visualisationID)
        .append("text")
        .attr("class", "showData")
        .text("")
        .attr("transform", "translate(" + (margin.right - 60) + "," + (0)+")")

    addText();
};
```

It then iterates through each variable, and adds the line to the graph, adds the circles to the data points which have roll over functionality, and adds a label to the end of the line.



```
    vis.append('svg:path')
        .attr('d', lineGen(graphData))
        .attr("transform", "translate(" + (margin.left) + "," + (margin.top)+")")
        .attr('stroke', colorOrder[0])
        .attr('stroke-linecap', 'round')
        .attr('stroke-width', 1)
        .attr('fill', 'none');

    var circles = vis.selectAll("point")
        .data(graphData)
        .enter()

    circles.append("circle")
        .attr("cy", function (d) {return yScale( d[attribute]); })
        .attr("cx", function (d) { return xScale( d.Date ); })
        .attr("transform", "translate(" + (margin.left) + "," + (margin.top) + ")")
        .attr('stroke', colorOrder[0])
        .attr("fill", colorOrder[0])
        .attr("r", function(d) { return d[attribute] == null ? 0 : 3; })
        .on("mouseover", handleMouseOver)
        .on("mouseout", handleMouseOut);
```

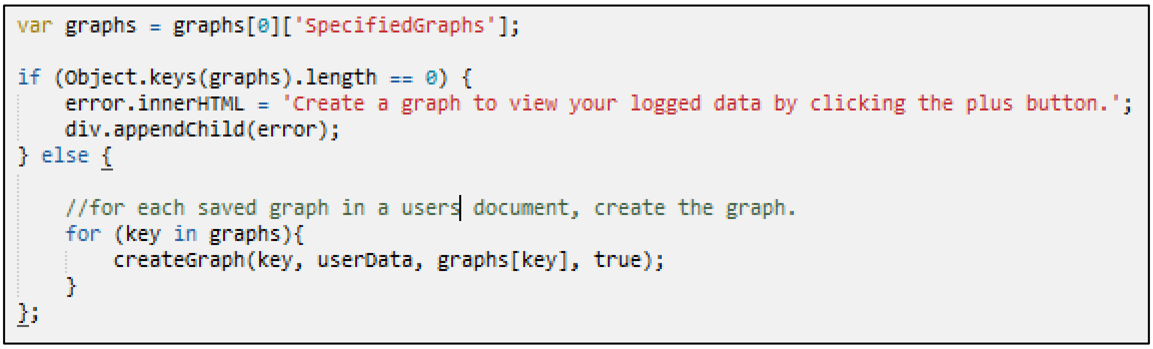
Not all readings will have all attributes filled out, therefore to append the label at the correct height at the end of the graph, the last known value in the data needs to be found. A function called 'getLastValue' iterates through the sorted list backwards to find the last element with the attribute, and takes the value from this. This value is then altered to fit the scale, and set to the y position at the end of the graph.



```
//gets the last value from the given data
//append the lines label to the last know data point if not complete data
function getLastValue(data, attribute){
    for (var i = data.length - 1; i >= 0; i--) {
        if(data[i][attribute]){
            return i
        }
    };
}
```

Then all of the previous elements are appended to the svg.

4.5.1 View Graphs



```
var graphs = graphs[0]['SpecifiedGraphs'];

if (Object.keys(graphs).length == 0) {
    error.innerHTML = 'Create a graph to view your logged data by clicking the plus button.';
    div.appendChild(error);
} else {

    //for each saved graph in a users document, create the graph.
    for (key in graphs){
        createGraph(key, userData, graphs[key], true);
    }
};
```

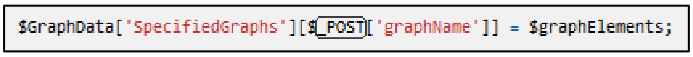
For a user’s ‘graph’ page, they’ll have a list of created graphs within their document in the database. The JavaScript on this page iterates over each entry in their document and creates a graph passing through the variables to the createGraphs function. This enables the user to have as many graphs as they would like. By having the ‘addLabel’ set to true, the roll over functionality of the data points is enabled.

4.5.2 Create Graphs

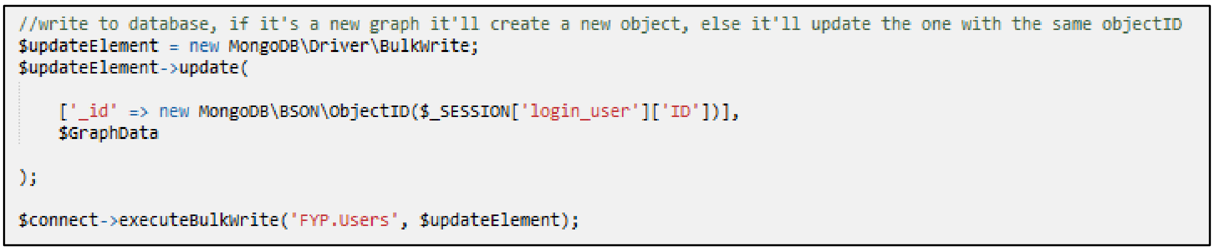
When creating a new graph, an array attributeList holds the values of the graph with is set to empty. If the value is held in the array, the background of the attribute in the list on the interface is set to blue so the user has confirmation that the element is active. When clicking an element, it gets added or removed from the attributeList array, and the displayed graph is updated with a new visual which is made with the updated list.

A new graph is made instead of updating the old one, as everything including the old lines need to be updated if the x and y scale change.

The graph name is in a visible form element, and the attributeList is in a hidden form which gets updated when selecting elements. On submit, this is passed to a php script which adds this to the correct place in the users document.



```
$GraphData['SpecifiedGraphs'][$_POST['graphName']] = $graphElements;
```



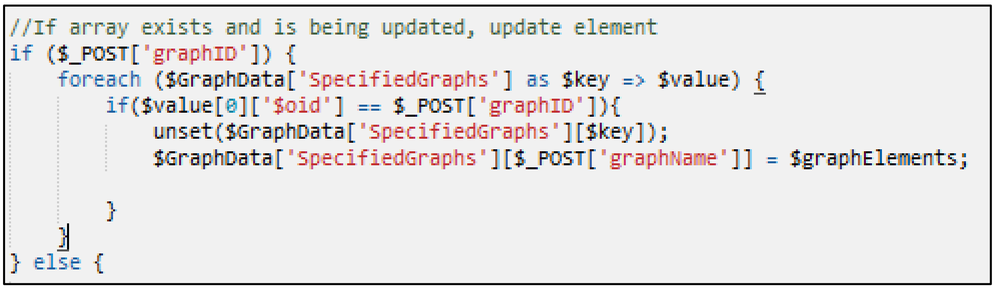
```
//write to database, if it's a new graph it'll create a new object, else it'll update the one with the same objectID
$updateElement = new MongoDB\Driver\BulkWrite;
$updateElement->update(

    ['_id' => new MongoDB\BSON\ObjectID($_SESSION['login_user']['ID'])],
    $GraphData

);

$connect->executeBulkWrite('FYP.Users', $updateElement);
```

This enables the new graph to be viewed at the bottom of the 'graphs' page for the user to view.

4.5.3 Edit Graphs

If a graph is being edited, it will send the graphs ID to the 'Create Graph' page via the HTTP method POST, and the page will be populated with the data from that graph to be edited by the user.

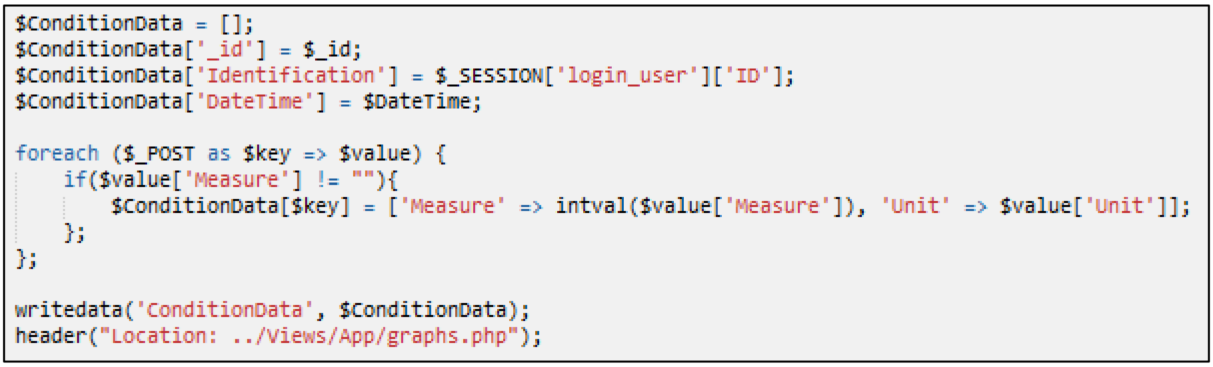
The ID of the graph is stored in another hidden element, which is also sent via POST to the PHP script. The PHP script, sets this as the ID of the graph, if there is an element already with this ID it deletes the element, and creates a new one. Effectively updating the previous entry.



```
//If array exists and is being updated, update element
if ($_POST['graphID']) {
    foreach ($GraphData['SpecifiedGraphs'] as $key => $value) {
        if($value[0]['$oid'] == $_POST['graphID']){
            unset($GraphData['SpecifiedGraphs'][$key]);
            $GraphData['SpecifiedGraphs'][$_POST['graphName']] = $graphElements;

        }
    }
} else {
```

4.6 Log a Reading

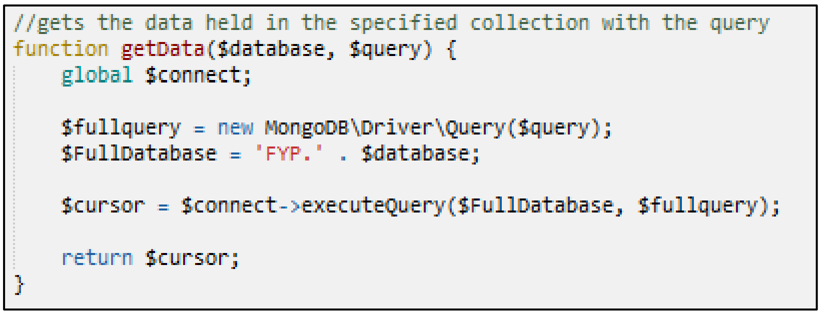
Readings are stored in the ConditionData collection of the database; each new reading is created in a new document with the UserID. The form is filled out with the data, and sent to a PHP script. The data doesn't need all fields filled out, and the fields are dynamic, therefore the script iterates over each field, and writes it to the document. This enables the fields to change with each user.



```
$ConditionData = [];
$ConditionData['_id'] = $_id;
$ConditionData['Identification'] = $_SESSION['login_user']['ID'];
$ConditionData['DateTime'] = $DateTime;

foreach ($_POST as $key => $value) {
    if($value['Measure'] != ""){
        $ConditionData[$key] = ['Measure' => intval($value['Measure']), 'Unit' => $value['Unit']];
    };
};

writedata('ConditionData', $ConditionData);
header("Location: ../Views/App/graphs.php");
```

4.7 Retrieving and displaying data

Data is retrieved and displayed at different points in the application, for example showing a profile of a user. This data is retrieved by a function in the config file getData, the parameters given to this function are the collection being queried and the query to find the data. Once it's been queried, the data is returned in the JSON format, and converted to an array for the JavaScript manipulation. This array is either iterated over, or specific elements are found.



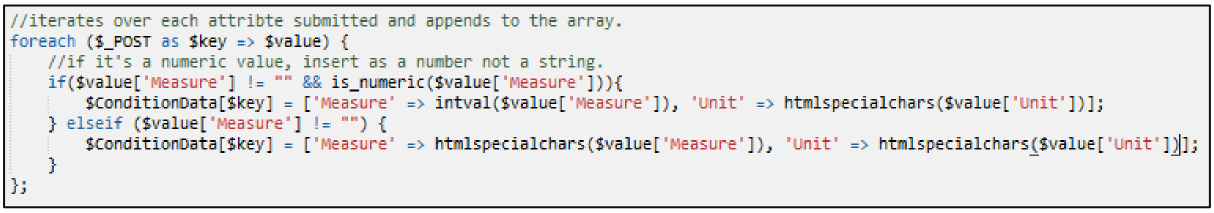
```
//gets the data held in the specified collection with the query
function getData($database, $query) {
    global $connect;

    $fullquery = new MongoDB\Driver\Query($query);
    $FullDatabase = 'FYP.' . $database;

    $cursor = $connect->executeQuery($FullDatabase, $fullquery);

    return $cursor;
}
```

When user data needs to be displayed, the database will be 'ConditionData' and query will be ['Identification' => $_SESSION['login_user']['ID']]. This queries the collection ConditionData to get all of the documents which have the attribute 'Identification' which matches the one stored in the session variable.

4.8 Forms

Having a large amount of forms on the application and website, form validation was needed. For all forms throughout, before the data is sent to the PHP script which will handle the data, it is run through form validation written in JavaScript. This form validation checks for innocent errors, such as length of passwords or regular expressions of emails, and also for HTML injection.

HTML injection is avoided with encoding the '<' and '>' symbols to their HTML equivalent, using php's htmlspecialchars function.

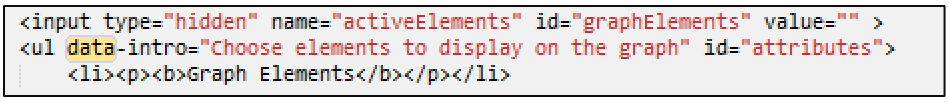


```
//iterates over each attribte submitted and appends to the array.
foreach ($_POST as $key => $value) {
    //if it's a numeric value, insert as a number not a string.
    if($value['Measure'] != "" && is_numeric($value['Measure'])){
        $ConditionData[$key] = ['Measure' => intval($value['Measure']), 'Unit' => htmlspecialchars($value['Unit'])];
    } elseif ($value['Measure'] != "") {
        $ConditionData[$key] = ['Measure' => htmlspecialchars($value['Measure']), 'Unit' => htmlspecialchars($value['Unit'])];
    }
};
```

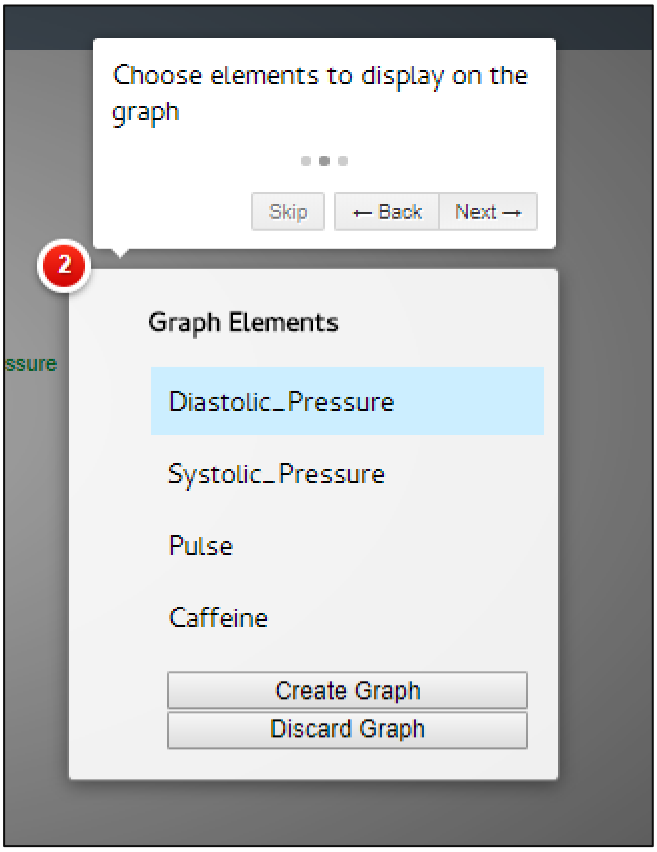
4.9 Help

Throughout the application, the library intro.js has been used to show the user around each webpage when clicking on the help button. This is for users who are confused, to help understand what each page is used for, and how it can be used. (Mehrabani, 2018)

When clicking the question mark button, it calls the start function for intro.js. This will then highlight in sequence all elements which have the 'data-intro' attributes and give text describing these elements.



```
<input type="hidden" name="activeElements" id="graphElements" value="" >
<ul data-intro="Choose elements to display on the graph" id="attributes">
    <li><p><b>Graph Elements</b></p></li>
```



This library creates an easy to use, and simplistic help functionality for the application. It creates a creates a grey background over all elements except the element being described, this draws the user's attention to the element. It also creates a minimalistic text box to give the information specified, and allows the users to go forward and backwards through each of the steps.

Intro.js was used as an out of the box solution instead of creating my own functionality to save time during the development of the application. This will also be maintainable in the future as usabli.ca, the company who created the library, will update the code with relevant bug fixes, and other issues.

5 Evaluation

The developed application has been evaluated to determine it's usefulness, this section will describe how this has been carried out. It contains the usability testing to ensure the ease of use of the interface, and requirements evaluation to certify that all goals that were set out at the beginning have been reached.

5.1 Usability testing:

Usability has been a large requirement of the project, so to ensure the application has fulfilled this requirement, I will be carrying out testing on three users. Out of these three users, only one falls into the target audience age range, the other two will be used for general feedback about the interface. Three users would not be considered a representative sample as the digital assessment questions outlines, ideally this would be carried out with more users of the correct age to get an accurate representation, however due to time constraints only these three will be used. This will give me an idea of how usable the system is, and some improvements that can be made. (Health Developer Network B, n.d.)

Each user will be asked to perform range of functions on the application in a specific order to understand how easy the system is to navigate, how easily tasks can be completed, and where the pain points of the application may be found. The tasks given to the users will simulate 'normal' use of the application, to be able to attempt to replicate a real user in a normal environment in a concentrated amount of time.

Whilst completing the tasks, the user will be observed, and notes about any hesitation, confusion or errors will be taken. If a user takes more than the allocated amount of time, has more than 2 errors per task, or cannot complete the task it will be considered a fail.

Once the user has completed the tasks, an open discussion on how they perceived the application, where the problems may lie, and any improvements to the system will be undertaken.

The allocated time has been decided by timing myself doing the tasks, taking time to read the text and type slowly. Trying to replicate how a new user, making no mistakes will use the application.

The tasks the user will be required to complete the following tasks:

- Create a new profile.
    - This will allow me to evaluate the navigation for the website, and the usability of the create user form.
    - Allocated time: 230 seconds
- Log into new profile.
    - Evaluate how the login process works, if it's intuitive to be taken to the graphs page as a 'landing' page.
    - Allocated time: 50 seconds
- Create 2 new readings, given dummy data.
    - If there's no data present, the application gives instructions on how to start. Evaluating the navigation of the application, how intuitive and easy to follow the instructions are.
    - Allocated time: 120 seconds

- Create a new graph given specifications.
    - Evaluate how easy the create a graph process is for a new user.
    - Allocated time: 120 seconds
- Log out
    - Evaluate the navigation of the application.
    - Allocated time: 40 seconds

The user will be logged into an existing account, filled with dummy data.

- Read a specific value from one of the existing graphs.
    - Evaluate how intuitive and obvious the rollover points on the graphs are.
    - Allocated time: 50 seconds
- Find birthdate from profile information.
    - Evaluate the navigation, and understand where the profile information would be held on the application. Understand whether having the name of the user as the profile button is intuitive.
    - During user testing this was an improvement which needs to be tested.
    - Allocated time: 50 seconds.

For the application to pass user testing, less than 10% of all tests should fail. This will ensure that the application has a high level of usability.

5.1.1 User Results:

5.1.1.1 Create new profile:

User 1: Pass

This user had no problems creating a new profile. Intuitively knew to click on the 'user login' and then 'Create Account'. She would have appreciated more descriptions about why her password kept being rejected, rather than just telling her that it didn't meet the requirements.

Completed in 75 seconds, so was under the allocated time.

User 2: Pass

This user thought the website was very clean and simple, had no trouble finding the 'create profile' button.

Completed in 217 seconds.

User 3: Pass

There were no problems when creating a new profile. He mentioned that being taken back to the login page without then being asked to login was a bit confusing. He didn't know whether the profile had been created or not.

Completed in 180 seconds

5.1.1.2 Log into new profile

User 1: Pass

Found the process easy and uncomplicated.

Completed in 20 seconds.

User 2: Pass

Completed in 17 Seconds.

User3: Pass

Apart from being unsure about whether the profile had been created, was able to login easily.

Completed in 15 Seconds.

5.1.1.3 Create 2 readings

User 1: Pass

It took her a few seconds to locate the appropriate navigation bar element, and understand that she needed to click the plus button to add a new field.

Initially she found it time consuming to create the fields, but once she understood that they only have to be created once, she found it very easy and nice to use.

Completed in 98 seconds

User 2: Pass

Initially clicked submit straight away without reading the text, which redirected her back to the graphs page which confused her. When navigating back to the log a reading page she then read the instruction text, and found it obvious what to do. She then completed the task quickly and proficiently.

Completed in 82 seconds.

User 3: Pass

This user clicked the plus button multiple times to try and add multiple fields at once. He then found it obvious and easy to enter the details after he understood only one form appears at the time. He thought that he would mainly be entering numbers, therefore in the drop-down list of ‘type of entry’, number should be the default option instead of text.

Completed in 115 Seconds.

5.1.1.4 Create a graph

User 1: Pass

Again, took her a few seconds to understand that she had to press the plus button to create a new graph. Enjoyed being redirected to the correct page to view the data once it had been entered.

Found it unintuitive that you had to press on the text, and not the highlighted box to add something to the graph. This will be a simple improvement.

Completed in 103 seconds.

User 2: Pass

After using the plus button to create the new fields, she felt it was easy to know how to create a new graph without reading the text. She found it confusing that the pointer didn't turn into a cursor over the buttons to add new datasets, but was still able to complete the task quickly and easily.

Completed in 98 seconds

User 3: Pass

Found it easy and intuitive to create a new graph, liked the interactivity of the page. Had no other comments.

Completed in 52 Seconds

5.1.1.5 Log Out

User 1: Pass

Intuitive and easy to complete.

Completed in 6 seconds.

User 2:

Found it very easy to log out.

Completed in 4 seconds.

User3:

Intuitive and easy to find.

Completed in 4 seconds.

5.1.1.6 Read a specific value from a graph

User 1: Fail

She found the value, by finding the data point and reading from the graph, she didn't know that you could roll over the points to get the values. Once the text on the right was pointed out she thought the feature was great and easy to use, however needs to be more obvious.

Completed in 32 Seconds.

User 2: Pass

She found it difficult to read the smaller texts, but she understood that if you rolled the data points it would give the raw data of the readings. She found the specific graph easily and quickly, read the information on the right, and rolled over the data point. She didn't try and read the data from the graph at all.

She made the comment that the 'Roll Over Graph Data' area and the edit button seemed to be connected, and confused her.

Completed in 17 seconds.

User3: Pass

Had no problems understanding that rolling over the graph made the values appear on the right-hand side.

Completed in 25 seconds.

5.1.1.7 Find birthdate

User 1: Fail

She forgot about the navigation bar, and kept trying to just search on the graphs page. Once prompted that it may be on another page, she found it immediately. Knowing that the name would lead to the profile page. When asked why she didn't think about looking on other pages, she said she got too focussed on one page, as the previous questions related to the graphs page.

Completed in 136 seconds.

User 2: Pass

The user understood that by clicking the name, it led to the profile, and knew that would be where this information would be held.

Completed in 17 seconds.

User 3: Pass

Found the birthday easily, however mentioned that it could be confusing with the option of 'Data' in the navigation bar. The little outline of a man that Facebook currently use, or something similar, placed next to the user's name could help erase that confusion.

Completed in 25 seconds.

5.1.2 Conclusions

Out of the 21 tests, there were only 2 fails. This is a good measure of how usable the application is, because there were only minor failings - I would conclude that the application has a high level of usability.

Logging into the application was perceived as easy, and 2 out of 3 users had no issues with creating and logging into their profile. The third user wanted more feedback after he created his account as a confirmation of his actions. This could be done by a small piece of text at the top of the page easily.

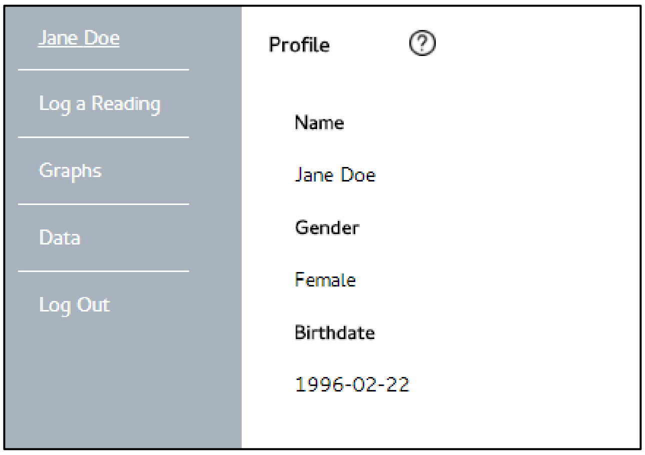
With the log a reading, all users found it very simple to find the page, yet all three found a small bit of confusion which was easily remedied by trial and error. If the user read the information given to them on the page, most of the confusion could have been avoided. Because all three users were able to understand how the system works with trial and error or reading, this isn't a large usability problem. However, to make this an easier process, the submit and discard buttons could be removed to minimise the possibility of error. By also making the help text bold it may help to draw their eyes to the plus button first.

Creating a graph was found to be a very easy simple process, a few improvements can be made to make it a bit more intuitive. Making the entire blue area a button, instead of just the text, and ensuring the cursor is made into a pointer will make the process of adding elements to the graph simpler. They all found that the interactivity of the page, and having the confirmation of the new graph being updated for them to see worked very well.

The area which needs most improvement is the rolling over values on the graph. The functionality works well, and all the users enjoyed using it once they understood how it could be used. However, the functionality needs to be more obvious. This is a mix of the guidance text being too small, and being out of the user's main view. By enlarging the and having it black instead of grey, it'll be more obvious for the user and potentially be easier to use. A user also felt that the edit button was part of the roll over text functionality, and found it confusing. Putting more space between these elements will make screen seem less cluttered, and a bit easier to understand.

An area which was raised previously in the user testing was the replacement of the profile button with the user's name for conciseness and clarity. This was tested in the usability testing and turned out to be a positive change. Two out of the three users understood that it was the profile page, the first user when trying to find the users birthday assumed it would be on the graph page, therefore didn't try looking elsewhere. I don't think this will be a common mistake, and will be treated as an anomaly.

5.2 Requirements Evaluation



Name

Jane Doe

Gender

Female

Birthdate

1996-02-22

FR1 - Requirement:
The system should enable the user to view the information they've entered for their profile.
Acceptance Criteria:
The application has a profile which shows the user all of their entered personal information.
Accepted: Yes
The user once logged in can click on their name from the navigation, this leads them to their profile page. This collects their data from correct document in the collection, and displays all the information they have entered. If a value is not complete it will not be displayed.

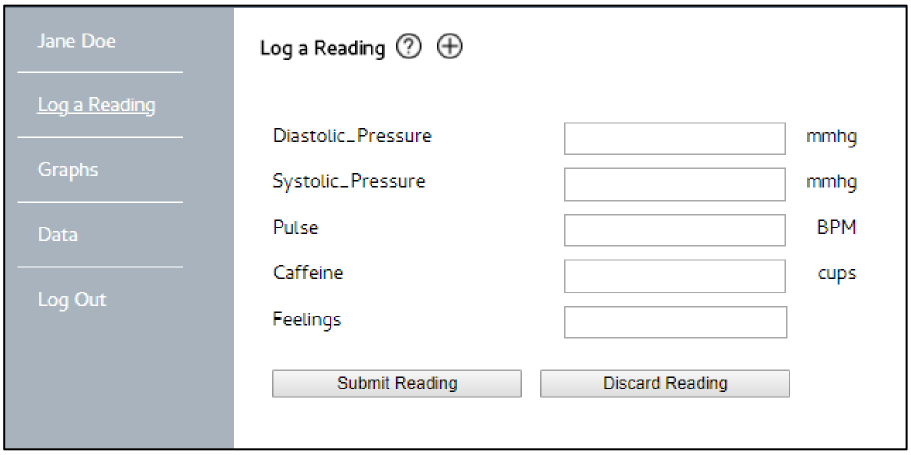
FR2 - Requirement:
The user should be able to change the profile information shown.
Acceptance Criteria:
The user should be able to easily change their profile information from the profile page. The information which is shown in the user's profile, when updated must change.
Accepted:
The design of the database and the application would allow the application to easily update the document held in the database with the new personal details entered from the user interface.

FR3 - Requirement:
The user should be able to enter the information relating to their high blood pressure into the system, for example their systolic pressure.
Acceptance Criteria:

The user interface must have specific areas that enable the user to input their data with the correct unit.

Accepted: Yes

If a user clicks ‘log a reading’, the user is then able to enter the information they want to log into fields created by the user. When the user clicks submit, it automatically saves the data with the current date. The data is saved into a new document with the user’s ID, into the correct database collection.



FR4 - Requirement:

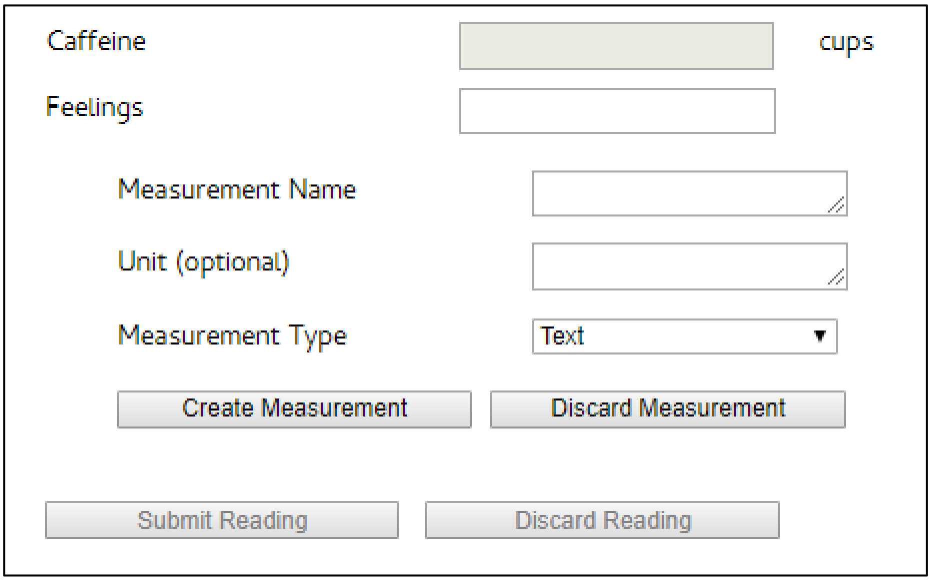
The system should support the user wanting to record more than the specified fields given in FR3.

Acceptance Criteria:

A user should be able to create new fields which include a measurement name, a unit and measurement type. Once a user has tracked a field once, they should not have to create the field again.

Accepted: Yes

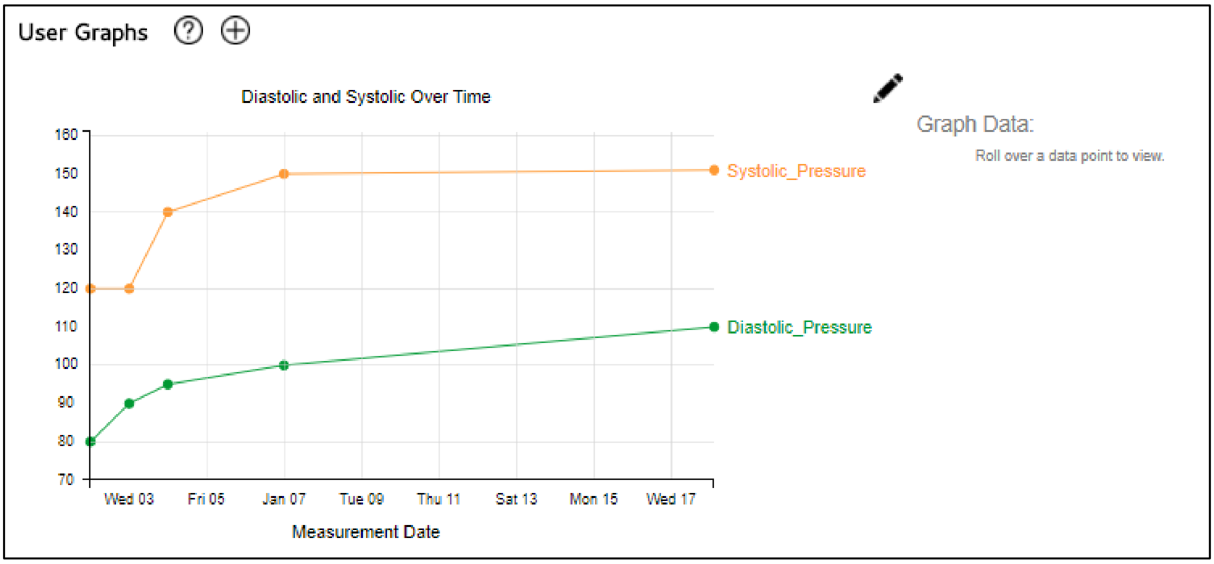
The user is able to click the plus arrow at the top of the page to create a new field. A new form will be created for the user to input the new fields name, unit and type of measurement. Once the user has saved the new field, it will be included in the input form, and can be filled out.


Measurement Type: Text

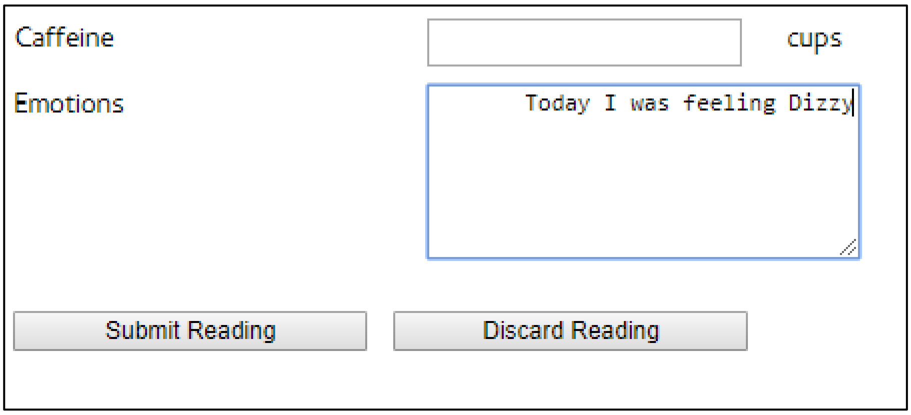
FR5 - Requirement:

The system should enable the user to view their condition information easily via graphs.

Acceptance Criteria:
The system should show be able to graphs relating the specific user's data.
Accepted: Yes
The User is able to view their data on graphs. The user will be automatically directed to the graphs page where the data is collected from the database, and displayed.



FR6 - Requirement:
The system should support the user tracking qualitative data such as how they are feeling.
Acceptance Criteria:
When creating a new field, a user should be able to choose the fields type to be text. Once data has been entered, this field should not be plottable on a graph, but when a data point has associated text data this should be shown when rolled over.
Accepted: Yes
By creating a field with the type text, the user is able to create a field to record how they are feeling, or anything else that can be described by text and not by a numerical value. These values will not be displayed on the graph, but when a point is rolled over it can be viewed with the data from that reading. Once one value is saved, it will always appear on their log a reading page to be entered again.



Caffeine		cups
Emotions	Today I was feeling Dizzy	

FR7 - Requirement:
The system should enable a patient to share their data with a healthcare professionals profile.
Acceptance Criteria:

A user should be able to share their data with specific medical practitioners using their email address.

Accepted:

Both the user and professional collections have been created in a way to make it feasible for users to add specific professional to view their data.

FR8 - Requirement:

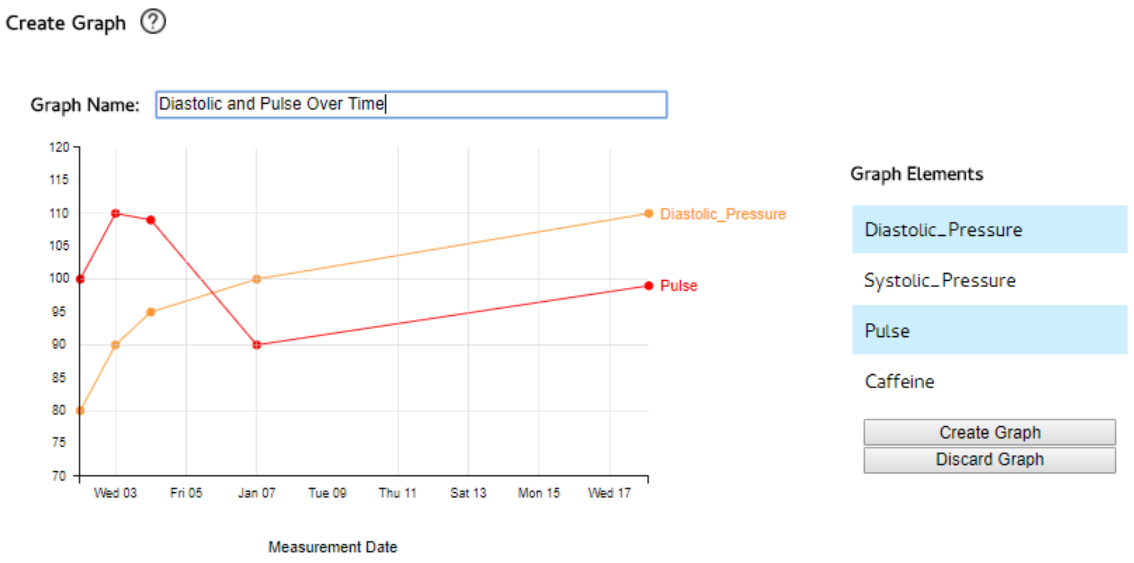
A user should be able to compare different datasets in graph formats by overlaying datasets. For example, a user choosing being able to compare the weight graph and blood pressure graph.

Acceptance Criteria:

When creating a new graph, a user should be able to add multiple data sets. These datasets should show on the same graph with the axis changing if needed.

Accepted: Yes

When a user crates a graph, can choose multiple different datasets which have numeric values. They can lay these over each other and the graphs scaling, colours and lines are responsive. If no data is chosen, the graph cannot be saved, and the user is prompted to choose a dataset.



Create Graph

Graph Name: Diastolic and Pulse Over Time

FR9 - Requirement:

A medical professional will be able to have their own profile to view all the data shared with them by their patients.

Acceptance Criteria:

A medical practitioner should have their own type of account, and be able to view the data relating only to the patients who have shared their data with them.

Accepted:

The way a professionals document is laid out allows for the addition of different user ID's, this professional will then be able to view the data from the specific users documents.

FR10- Requirement:

The user should be able to see how changes in medication has changed their blood pressure and other symptoms.

Acceptance Criteria:

The application will be designed to be able to view the medications on the graphs, and within the raw data.

Accepted:

The way the application and database has been built, it would be easy to add medication changes to the user graphs, and view this data visually.

NF1 - Requirement:

A user should be able to use the system easily and intuitively.

Acceptance Criteria:

Once the application has been developed, usability testing will be carried out to ensure the ease of use of the system. A user should be able to use the application without any prior training, with minimal confusion and few errors.

Accepted: Yes

Through my limited user testing, I have been able to conclude that the application has a high level of usability. Out of 21 tests only 2 failed, which is less than 10%. Therefore I can conclude that this has passed this acceptance criteria.

NF2 - Requirement:

The application will be developed on an appropriate platform, this platform should take into account the ease of use of the system.

Acceptance Criteria:

The platform must be the most appropriate choice for the project.

Accepted: Yes

The application has been developed as a web application, as described earlier this is the most suitable.

NF3 - Requirement:

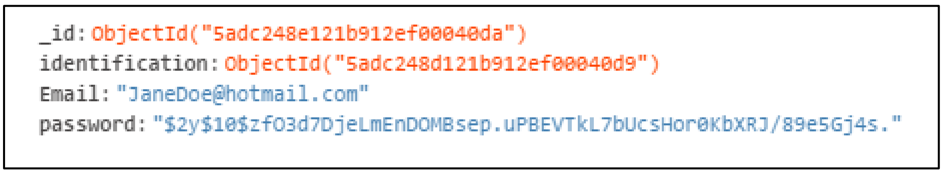
The data should be stored securely on the database and the application should be secure.

Acceptance Criteria:

The storage of data will be up to security standards currently in place.

Accepted:

Passwords are hashed and salted during storage.



```
_id: ObjectId("5adc248e121b912ef00040da")
identification: ObjectId("5adc248d121b912ef00040d9")
Email: "JaneDoe@hotmail.com"
password: "$2y$10$zfO3d7DjeLmEnDOMBsep.uPBEVTkL7bUcsHor0KbXRJ/89e5Gj4s."
```

NF4 - Requirement:

The storage of data should be up to current standards.

Acceptance Criteria:

The application will be designed to standards in FHIR and Hl7.

Accepted: Yes

All data is stored and accessed in the ways that FHIR has specified. The specific use of MongoDB is used to store the documents in a JSON format. This can be extended and developed further at a later point.

6 Future Work

The created application works by itself, but there are many improvements which can still be made, and functionality which can be added.

As mentioned previously if I had more time, more extensive usability testing would have to be carried out. These users would ideally have hypertension, and be within the target audience range. This would give a range of improvements which would have to be worked on first, before more functionality was added. This has the potential to create a more well-rounded application with a higher level of usability.

To make a solid foundation to the application, more research into secure storage of the data should be carried out. For example, the data saved in the database is held in plaintext, as this is personal information this should be encrypted before being saved. The data could also be encrypted on the client side before being sent to the server to be stored in the collection. As this is personal information more research should be done into this area. This would be an easy addition, which would add a lot of benefits before being deployed.

A large improvement that could be made, would be to make it available on other platforms such as mobile. Currently the application doesn't render well on mobile devices or tablets. To make the application accessible to all users who may not have laptops or desktops, a mobile site would be required. This would also make it easier for users to upload, and share their data when they are away from their computers. This would again increase the usability of the site.

An improvement mentioned earlier would be creating more types of graphs, currently only one type of line graph is available to the users. By questioning potential future users of the application, different graphs which may also be useful can be decided and implemented. This would allow more flexibility to the site, and allow the application to be more customisable.

A desirable requirement which didn't get implemented was to allow a user to understand their own 'normal' data levels using machine learning techniques and display this to the user. This would be a large development which would be of a significant benefit to the user. They would be able to easily understand their values without research, this would help with the self-management of their condition.

The main development of the application in the future would be linking it up to the NHS. The application has been created with this in mind, using a non-relational database and storing the data in the FHIR standard. The next step in this process would be to set up the Hl7 security and link to a FHIR server. This would be a large development and would be very time consuming, but would be a valuable addition to the system.

For the application to be considered by the NHS it would have to pass the Digital Assessment Questions and follow their guidance. I have not been able to meet some of these due to time constraints, however these can be completed in the future such as, enabling users to block certain devices from accessing personal data. (Health Developer Network B, n.d.)

7 Conclusions

When starting the project, the aim was to create an application to track long-term health conditions, and the variables surrounding them. A user would be able to use the application to monitor how different changes in their life style would change their health statistics.

The personas created aims were:

- Recording lifestyle changes easily to stay motivated.
- Keep all of their data organised, and easy to reference and review.
- Be able to control how and when she records her information.
- Communicate her data effectively with her GP
- Need an easy way to review how medication changes affects his condition.
- Simple way to record and view if his lifestyle changes have affected his condition
- Keep all of his data in one place, to not switch between applications.

The application created allows the users to record any data they may want to, in a single location, where they can easily view it in any way they need. The application has also been designed to allow a user to share their data with any practitioner with an account. Therefore, all of these goals have been fulfilled.

Along with fulfilling these goals, the application has a high level of usability which was a priority of this project, however it doesn’t always meet expectations and more improvements still need to be made. If more time was available, these improvements could have been implemented, and more user testing could be carried out with extra users to create a well-rounded application as described previously.

Overall the application is a successful application, however it still needs a lot of work to reach its full potential.

8 Reflection on learning

Throughout this project, there have been many learning experiences, and overall I feel like I've been proficient in the execution, but not effective. I have gained many learning experiences, developing and creating skills, and it was a new experience, however it could have been carried out in a better manner.

Firstly, the extent of my project management skills coming into this project were minimal, bounded by managing my own time and tasks for smaller projects such as coursework at university. Nothing to the extent of a 12-week project. Throughout my placement year, the level of responsibility was more similar to this project, but I had continual supervision and guidance if needed. This left me unsure on how to conduct a project such as this one. My Gantt chart created at the start was useful, and breaking down all the tasks of the project from design to evaluation into smaller chunks enabled me to have an idea of how long different aspects would take. However, the estimations of these tasks were not accurate, which meant this plan wasn't effective and couldn't be followed for long. Along with this, I stayed on certain tasks for longer than what I should have.

In the future, with more experience I'd be able to understand more fully when to move on from something that isn't going anywhere. For example, the research into FHIR took much longer than what was planned and was less successful than expected. Looking back, I should have continued on with the next task, and moved back at a later point, so as to keep on progressing.

Another skill which has been greatly developed is my programming skills. Having never developed a project from the start, setting up an environment, choosing a range of technologies and starting a project from scratch was something I had never done. By doing this once, I now know how to approach a situation similar to this in the future, and what mistakes to avoid again. These are skills which I'm sure will be useful in my career.

Skills in other technologies such as databases have been developed, having always used relational databases, using a non-relational database such as MongoDB was a steep learning curve. Research into how they are structured, how to set them up easily, and their optimisation was carried out. This enabled me to use the database efficiently for this project. In future projects when choosing technologies, this will enable me to weigh up pros and cons of relational and non-relational databases to understand which would be better to use.

Another aspect of this project was learning about the user centred design methodology, and its relevant advantages and disadvantages. The application developed has a high level of usability which couldn't have been achieved without, and a set of improvements which would benefit the system greatly. This has helped me understand how getting the user involved can improve the design and functionality of the system, in comparison to omitting personas, user testing and usability testing.

Throughout this project there have been many learning experiences which I had not been exposed to before. The hard skills mainly encompassing learning how to manage and be the only developer on a project, and the soft skills entailing independence, responsibility and time management. These soft and hard skills have been a valuable personal development, which I will be able to take with me and use in any potential projects and tasks in my post-university life.

9 Appendix

9.1 Appendix 1:

The remainder of the use cases:

9.1.1 ID: UC1

Title: A patient sets up a profile
Description:
A user has decided they want to create an account, and sign up for the service. They must enter their email address and other details to create the new account.
Primary Actor: User
Preconditions: None

Main Flow:
1. The user access the website and clicks 'create patient account'.
2. The user enters their email address, password and confirms password.
3. The user will then be sent an email confirming their account, and asking to activate it before use.
4. The user will activate their account though a link on their email.
5. The user will then be directed back to the website.
4. The user will be prompted to fill out additional profile information
   a. Name, Birthday, Family History, Current and previous medications.
5. The user clicks Next
6. The user is taken onto the web application

Post Conditions:
The user has set up their own account, and is now able to add and view their data.

9.1.2 ID: UC2

Title: User logs into their account
Description:
A user wants to access their data so needs to log into their account.
Primary Actor: User
Precondition: UC1

Main Flow:
1. The user accesses the website.
2. The user enters their email and password
3. The user is directed to the web application showing their data.

Alternate Flow:
A1. The user can't remember their password and needs to reset it.
2. The user clicks forgotten password.
3. The system notifies them an email has been sent to their email address.
4. The system sends an email with a temporary password which can be reset.
5. The user enters their email and temporary password
6. The user is directed to the web application showing their data.

Post Conditions:

The user is logged into their account and is able to view and add data.

9.1.3 ID: UC4

Title: Change Profile Information
Description:
A user will decide they want to change their family history information. They will be able to change this on the profile screen.
Primary Actor: User
Preconditions: UC1

Main Flow:
1. The user clicks the profile button.
2. The application proceeds to the profile page.
3. The user clicks the edit button.
4. The application proceeds to the edit page.
5. The User can then edit their family history.
6. The user clicks save.

Post Conditions:
The new information is saved in the database, and is available to view on the website.

9.1.4 ID: UC7

Title: Change Saved Data
Description:
The user logs onto the application, and wants to view their graphs containing their data.
Primary Actor: User
Preconditions: UC1, UC2, UC3

Main Flow:
1. The User can view the raw data on the main page.
2. The user can select the data they want to change, type a new value and press save.

Post Conditions:
The Data is then updated in the database, and the graphs are updated.

9.2 Appendix 2:

Task	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Easter (3 weeks)	Week 9	Week 10	Week 11	Week 12
Initial Report Write Up	■												
Chronic Condition Research		■											
Choose Chronic Condition		■											
Researching chosen Condition		■											
Choose Relevant Data Fields		■											
Requirements			■										
Requirements Gathering			■										
Requirements Finalising			■										
Research into tools			■	■									
Choosing Database Technologies			■										
Choosing Backend Technologies				■									
Choosing Frontend Technologies				■									
Security Research				■									
UI Design				■	■								
Creating Use Cases				■									
Creating UML Diagram					■								
Wire frames of UI					■								
Implementation						■	■	■	■	■			
Basic Profile						■	■						
Ability to add data							■						
Ability to add new data fields							■	■					
Data Alerts								■					
User Graphs								■	■				
Medical Professional Area									■				
Medical Professional Graphs									■				
Security Implementation									■				
Extra implementation and improvements									■	■			
Testing										■			
Usability testing										■			
System Evaluation										■	■		
Final Report write up											■	■	■

10 References

Bernell, S. and Howard, S. (2016). Use Your Words Carefully: What Is a Chronic Disease?. *Frontiers in Public Health*, [online] 4. Available at: https://www.frontiersin.org/articles/10.3389/fpubh.2016.00159/full.

Php.net. (2018). *PHP: Password Hashing - Manual*. [online] Available at: http://php.net/manual/en/faq.passwords.php [Accessed 19 Apr. 2018].

Hl7.org.A (n.d.). *Resource Patient*. [online] Available at: https://www.hl7.org/fhir/patient.html [Accessed 24 Apr. 2018].

Hl7.org.B (n.d.). *Resource Practitioner – FHIR v3.0.1.* [online] Available at: https://www.hl7.org/fhir/practitioner.html [Accessed 24 Apr. 2018].

Hl7.org.C (n.d.). *Extensibility - FHIR v3.0.1*. [online] Available at: https://www.hl7.org/fhir/extensibility.html [Accessed 24 Apr. 2018].

Hl7.orgD (n.d.). *Defining Extensions*. - FHIR v3.0.1. [online] Available at: https://www.hl7.org/fhir/defining-extensions.html [Accessed 31 Apr. 2018].

NHS Digital. (2018). *Registration Authorities and smartcards - NHS Digital*. [online] Available at: https://digital.nhs.uk/services/registration-authorities-and-smartcards [Accessed 24 Apr.2018].

Health Developer Network A. (2018). *Get Connected - Health Developer Network*. [online] Available at: https://developer.nhs.uk/library/systems/e-rs/start/get-connected/ [Accessed 24 Apr. 2018].

Asthma UK. (2018). *Asthma facts and statistics | Asthma UK*. [online] Available at: https://www.asthma.org.uk/about/media/facts-and-statistics/ [Accessed 24 Apr. 2018].

Ageuk.org.uk. (2015). *16 million people in the UK have high blood pressure*. [online] Available at: https://www.ageuk.org.uk/latest-news/archive/16-million-people-in-the-uk-have-high-blood-pressure/ [Accessed 24 Apr. 2018].

Diabetes UK. (n.d.). *Testing*. [online] Available at: https://www.diabetes.org.uk/Guide-to-diabetes/Managing-your-diabetes/Testing [Accessed 24 Apr. 2018].

DH/IPU/Patient Confidentiality (2018). *NHS Confidentiality Code of Practice*. [online] pp.7-9. Available at: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/200146/Confidentiality_-_NHS_Code_of_Practice.pdf [Accessed 24 Apr. 2018].

Nielsen, J. (1994). Enhancing the explanatory power of usability heuristics. *Proceedings of the SIGCHI conference on Human factors in computing systems celebrating interdependence - CHI '94*.

NHS (2018). *Measuring Shared Decision Making A review of research evidence*. [online] NHS. Available at: https://www.england.nhs.uk/wp-content/uploads/2013/08/sdm-evidence.pdf [Accessed 25 Apr. 2018].

Corben, S. and Rosen, R. (2005). *Self-management for Long-term Conditions*. King's Fund.

Developer.apple.com. (n.d.). *Themes - Overview - iOS Human Interface Guidelines*. [online] Available at: https://developer.apple.com/ios/human-interface-guidelines/overview/themes/ [Accessed 26 Apr. 2018].

Docs.microsoft.com. (n.d.). *Introduction to Universal Windows Platform (UWP) app design*. [online] Available at: https://docs.microsoft.com/en-us/windows/uwp/design/basics/design-and-ui-intro [Accessed 26 Apr. 2018].

Blomkvist, S. (2018). Persona – an overview. *The User as a personality. Using Personas as a tool for design.*

Usability.gov. (2018). *User-Centered Design Basics | Usability.gov*. [online] Available at: https://www.usability.gov/what-and-why/user-centered-design.html [Accessed 26 Apr. 2018].

The Interaction Design Foundation. (n.d.). *What is User Centered Design?*. [online] Available at: https://www.interaction-design.org/literature/topics/user-centered-design [Accessed 26 Apr. 2018].

nhs.uk. (2018). *High blood pressure (hypertension)*. [online] Available at: https://www.nhs.uk/conditions/high-blood-pressure-hypertension/ [Accessed 27 Apr. 2018].

Pathways.nice.org.uk. (2018). *Hypertension - NICE Pathways*. [online] Available at: https://pathways.nice.org.uk/pathways/hypertension [Accessed 27 Apr. 2018].

Sakr, S. and Gaber, M. (n.d.). *Large scale and big data: Processing and Management*. CRC Press, p.328.

Couchdb.apache.org. (2018). *Apache CouchDB*. [online] Available at: http://couchdb.apache.org/ [Accessed 29 Apr. 2018].

MongoDB. (2018). *MongoDB for GIANT Ideas*. [online] Available at: https://www.mongodb.com/ [Accessed 29 Apr. 2018].

Php.net. (2018). *PHP: MongoDB - Manual*. [online] Available at: http://php.net/manual/en/class.mongodb.php [Accessed 29 Apr. 2018].

Hl7.org.E (n.d.). *Health Level Seven International - Homepage*. [online] Available at: http://www.hl7.org/ [Accessed 30 Apr. 2018].

Mehrabani, A. (2018). *Intro.js - Step-by-step guide and feature introduction for your website*. [online] Introjs.com. Available at: https://introjs.com/ [Accessed 3 May 2018].

Health and wellness apps – Quality criteria across the life cycle – Code of practice. (2015). BSI Standards Limited 2015.

Health Developer Network B. (n.d.). *Digital Assessment Questions - Beta - Health Developer Network*. [online] Available at: https://developer.nhs.uk/digital-tools/daq/ [Accessed 4 May 2018].

Ico.org.uk. (n.d.). *The amount of personal data you may hold (Principle 3)*. [online] Available at: https://ico.org.uk/for-organisations/guide-to-data-protection/principle-3-adequacy/ [Accessed 6 May 2018].