GOSH: App for child growth charts in R

Team 14

Rajan Hirani   **Saleh Khalil**   **Sander Da Mata Miranda**

Department of Computer Science

University College London

Abstract

When children are born, a lot of information is gathered about them as they develop such as weight, height etc. This data is usually visualized in a graph format and generally on paper. With over 130,000 births last year in London alone, a lot of data has to be collected and resources such as paper used to produce the growth charts for this data. This is where our problem lied. Our task was to produce an application for Great Ormond Street Hospital that can produce these growth charts and display useful information to the user about the growth of the child for example, what centile their weight/height lies in etc. Our client also wanted us to make this application available on as many platforms as possible especially mobile devices.

The solution we created was a web app coded in the statistical language R. We used R as it easily allowed us to read and manipulate data from the .csv files in which the information was stored. We used two main libraries from R for our app. Firstly, we used R-Shiny to make our app accessible from the web, allowing our application to be used on all platforms with web functionality. As well as giving us web functionality, R-Shiny provided the resources for us to create a UI for our application. The other library we used was 'plotly'. This library allowed us to display the growth charts in a visually stimulating format and contained a lot of built in functionality that provides a more robust experience for the user allowing them to zoom in/out, compare data at different points etc. We produced two growth charts for each type of growth and gender. One graph was a measurement/age graph with the child's data as well as the lines for some centiles so the user can see how child's growth compares to the rest of the population. The other graph is a z-score/age graph which allows the user to see more explicitly how the child's growth moves in centiles providing more information to the user.

Given the time restraints and lack of experience with the R programming language, the progress we've made with the project is quite remarkable as we met all the essential and

most of the optional requirements the client set for us. With this in mind, we can state with confidence that we have produced a complete product. However, the app isn't ready to be used in hospitals just yet, with more time we could add more functionality and provide a data store such that hospitals can upload their data.

Contents

1 Introduction

1.1 Client and Project Background

Our project lead client is Professor Tim Cole, a professor of medical statistics at UCL Institute of Child Health. His main area of research covers many aspects of child growth assessment, which includes growth chart construction and centile graphs.

Growth is an important aspect for a child's development, it can show any early signs of diseases or how a child responds to treatment. Generally, growth was recorded using paper charts and recently this process is moving to plotting these points digitally. This means that it is easier to determine which centiles a child lies in.

1.2 Problem and Project Goal

The current problem is that generating these growth charts are complex as there are other factors that can affect growth, such as sex (boys and girls), ethnicity etc. The project goal is to create an application to eliminate paper-based growth charts.

1.3 Development Team

1.3.1 Programming Experience

Rajan – VB.NET, C#, Python

Saleh – Python, HTML

Sander – Python, Java

Team members hadn't worked with R before the group project and so it was the first time working with the language.

1.3.2 Main Team Roles

Rajan: Programmer, Front End Developer, Client liaison, Report Editor

Saleh: UI Designer, Researcher, Programmer, Client liaison

Sander: Report Editor, Tester, Programmer, Client liaison

2 Requirements

2.1 Personas

Typical users who would use this application would be health centres so that they can monitor the growth of their patients when children come in for their routine check-up. The application could be extended in the future so that any user can use the application so parents can produce their own growth charts for their children.

2.2 MoSCoW Requirements

ID	Requirements	Priority
1	Plotting of serial measurements of an individual on a normal growth chart with centile detection	Must
2	Plot height and weight growth charts	Must
3	Have web app functionality	Must
4	Use GOSH's data to produce growth charts	Must
5	Show some sort of growth trajectory using the data that is inputted	Should
6	Integrated with SMART on FHIR so it can be compatible with any health centre that uses SMART on FHIR for their data storage	Should
7	Have some functionality to output data to files or in a pdf format	Could
8	Have some functionality for data security	Could
9	Have some functionality for the application to run in a mobile browser	Could

Table 2.1: MoSCoW Requirements

3 Research

3.1 Potential Solutions

There were three main practical options we could have chosen to complete our application project: Python, HTML or R. Each had its own advantages and disadvantages tailored towards our project and eventually R was the selected language.

3.1.1 Python

Python is a simple and easy to use programming language. It is also efficient at graph drawing and plotting data. It can be limited and slow at times compared to other languages. Additionally, weak mobile compatibility would make it hard to support the app on mobile devices if we choose to do so.

3.1.2 HTML

HTML is the standard mark-up language for creating web pages and web applications. It is widely used and therefore supported across all browsers. There are many available resources when coding in HTML. However, HTML isn't ideal for plotting data onto graphs which is one of the main features of our project.

3.1.3 R

R is a free language orientated around statistical and data analysis. All of the standard statistical tests, models and analysis are incorporated into R. This makes it ideal for dealing with and manipulating large amounts of data and plotting them onto graphs. Also, RShiny can be used with R to create a web app without having any code in HTML. However, a disadvantage of picking R is that it was a new language to everyone in the group.

4 Design and Implementation

4.1 Design

4.1.1 System architecture diagram and component descriptions
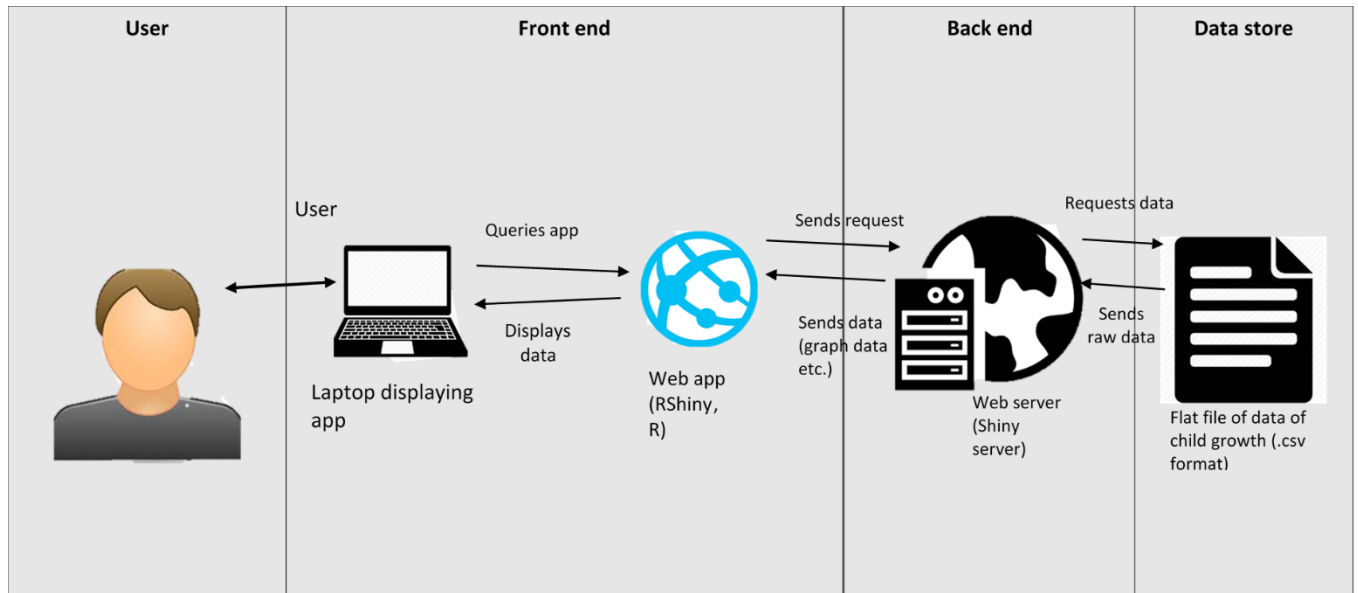
When interacting with the application, the user is presented with the front end of the system (the web app / laptop). From here, the user can request data such as the male height growth chart by clicking the relevant buttons and the data is also displayed here for the user to access. When data is requested by the user, a request is sent to the back end of the system (the shiny server hosting the application) which then retrieves the data from the .csv file uploaded to the server, the data is then sent up back to the web app so that it can be processed and outputted to the user accordingly.

The data store we used to store the data on child growth was a .csv file as there are built in functions in our front end, R that allowed us to manipulate .csv files to extract data with ease. We used "shiny server" as our back end as there is a module that can be added to R called 'RShiny' that easily allowed us to port our R code into a web app that can be accessed on all platforms without needing the source code to be present.

4.1.2 Site Map
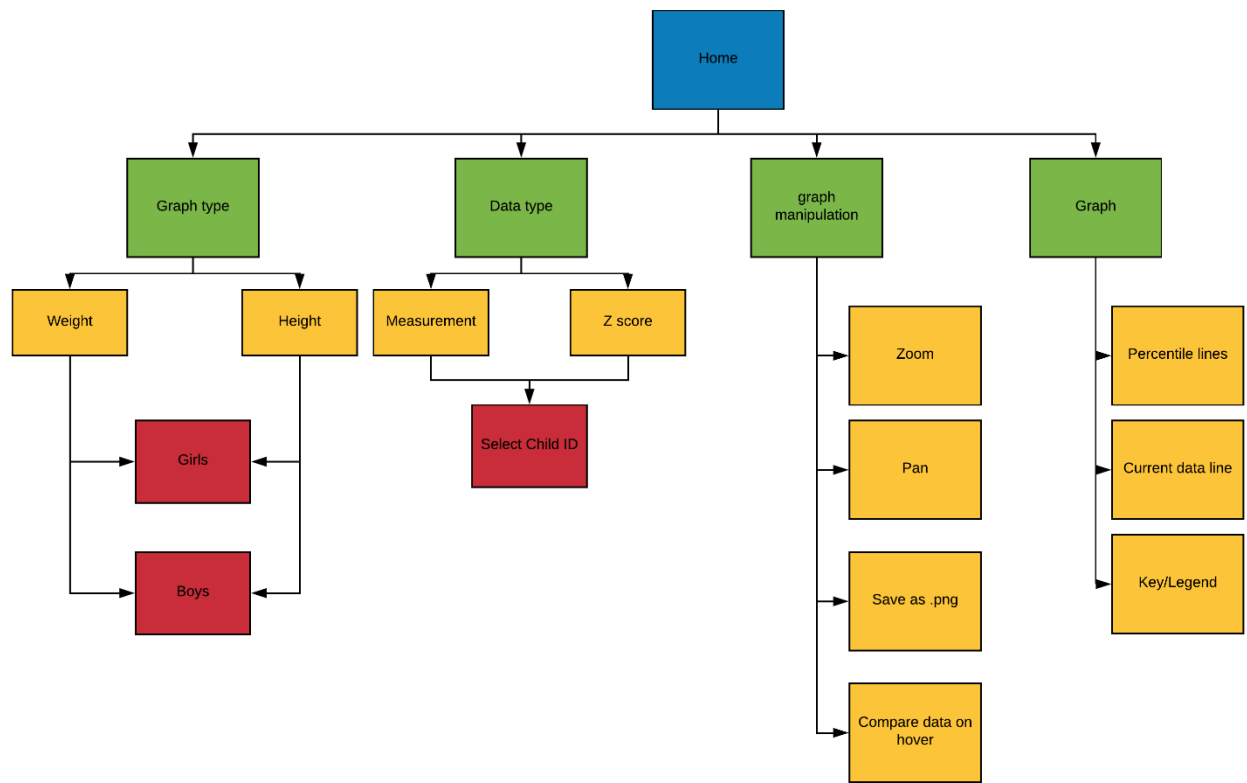
4.2 Implementation

4.2.1 Version control

While developing our project we used GitHub to share what we were working on with ease. Through use of this software we were able to simultaneously work on the code and commit it to a master branch when we want to use what we've created. This has also saved us from having to send constant emails regarding changes and having to trust one person with the code etc.

4.2.2 RStudio

We used RStudio as our IDE for coding the front end of our application as it is the industry standard for coding programs in R and our client, Tim Cole advised us to use R for the creation of our application as it has very easy to use modules and functions made for displaying data on graphs which is what the main purpose of our app is.

4.2.3 RShiny and shiny server

RShiny is a module we used for enabling our R code to uploaded to the net so that our app can be used as a web app that can be accessed across all platforms with web-based e universally accessed as well as acting as our back end, we used this server as it has direct compatibility with RShiny and is free to use.

4.2.4 Implementation of core features

**Showing weight/height graphs with centile curves:**

In order to implement this core feature, we used a library in R called plotly to create a graph that is aesthetically pleasing for the user. We used this library over others as it had simple and easy to use built in functions that allowed us to add graph features such as **a** legend, axis titles as well as features that allow the user to manipulate the graph for example zooming in and panning onto different sections of the graph and being able to see the exact value at a particular point on

the graph upon hovering. What we did to create the values for the graph was import all the LMS values from a .csv file into our R script and using an algorithm that our client, Tim Cole gave to us we were able to use the LMS values to calculate a measurement value that we
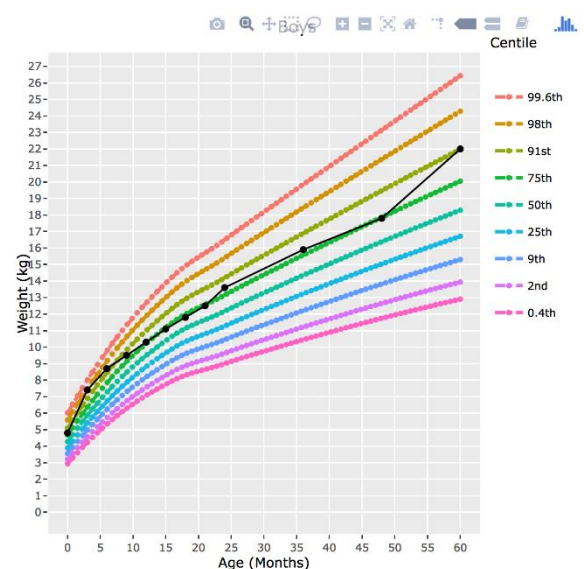
use for the graph. For the different centile lines, similarly to what we did for each child's data we had a set of values for each centile that we imported via a .csv and using the same algorithm we calculated the measurements for each desired centile and plotted each centile as a distinctly coloured line.
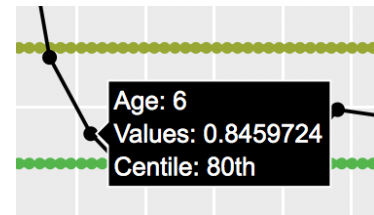
Transforming measurement graphs into z score graphs:

In addition to having a measurement time graph for the user to see, we also added another type of graph that makes it easier for the user to compare a child's growth to the national average more directly. The graph we chose to do that will achieve this is a z score time graph which shows how the child's growth changes in the centiles over time. We did this by importing the same set of LMS values for each child that we used for the measurement time graph but instead used a different formula that calculates the z score instead of the measurement. This formula was also provided to us by our client Tim Cole.
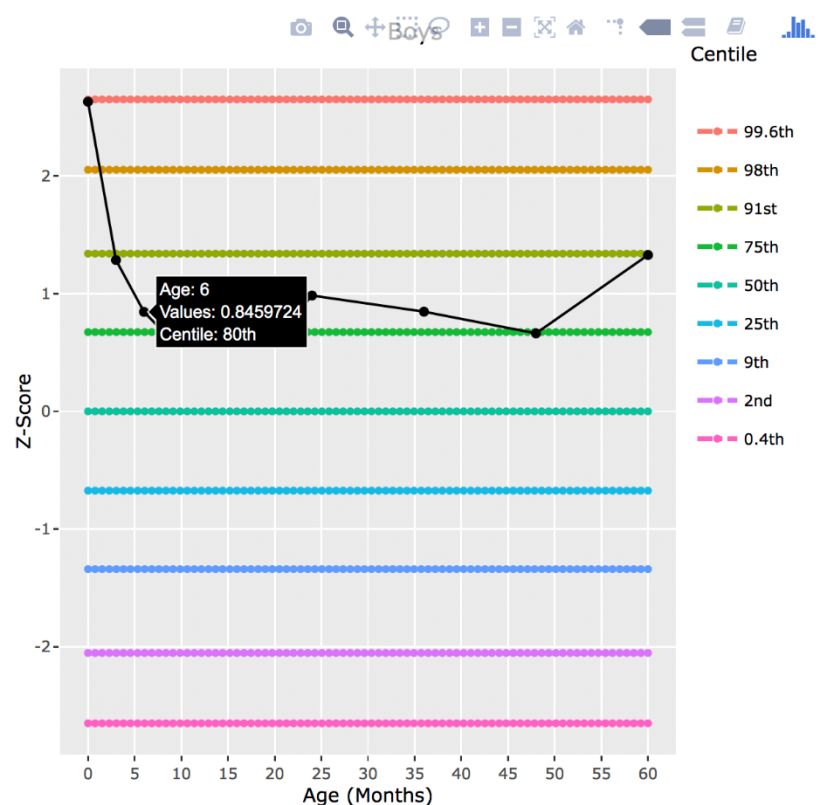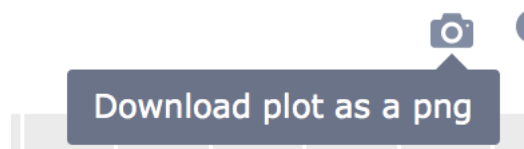
Viewing centile on hover:

When we first created the z score time graph, the centile
wasn't showing when the user hovers over a point instead only
showing the calculated z value at that point which doesn't
clearly indicate how that specific child's growth compares to
the rest of the sample. What we did to implement this feature was add a 'centile' column to
the data frame which contained the centile for each calculated z value. With this, we then
used a function in the 'plotly' library that includes the value of this column which comes up
when the user hovers over a point in the graph. Although this may seem like quite a small
feature, it is the feature we are most proud of as it shows the user a direct and clear
comparison between a particular child's growth and the rest of the general population
making our application more advatageous and resourceful than the current method of
viewing child growth charts (paper format).

Saving graph to an image file:

Through our use of the 'plotly' library in R, when we
created a graph, there was already a built in feature
within the library that allows the user to save the graph to a .png file. This is a useful feature
as it would allow doctors to send a particular childs growth chart to the patient for their
own personal use and without having to have our application.

5 Testing

5.1 Compatibility Testing

Browser	Version No.	Result
Chrome	65.0.3325.162	Fully functional
Microsoft Edge	41.16299.248.0	Fully functional
Mozilla Firefox	59.0	Fully functional
Safari	11.0.2	Fully functional
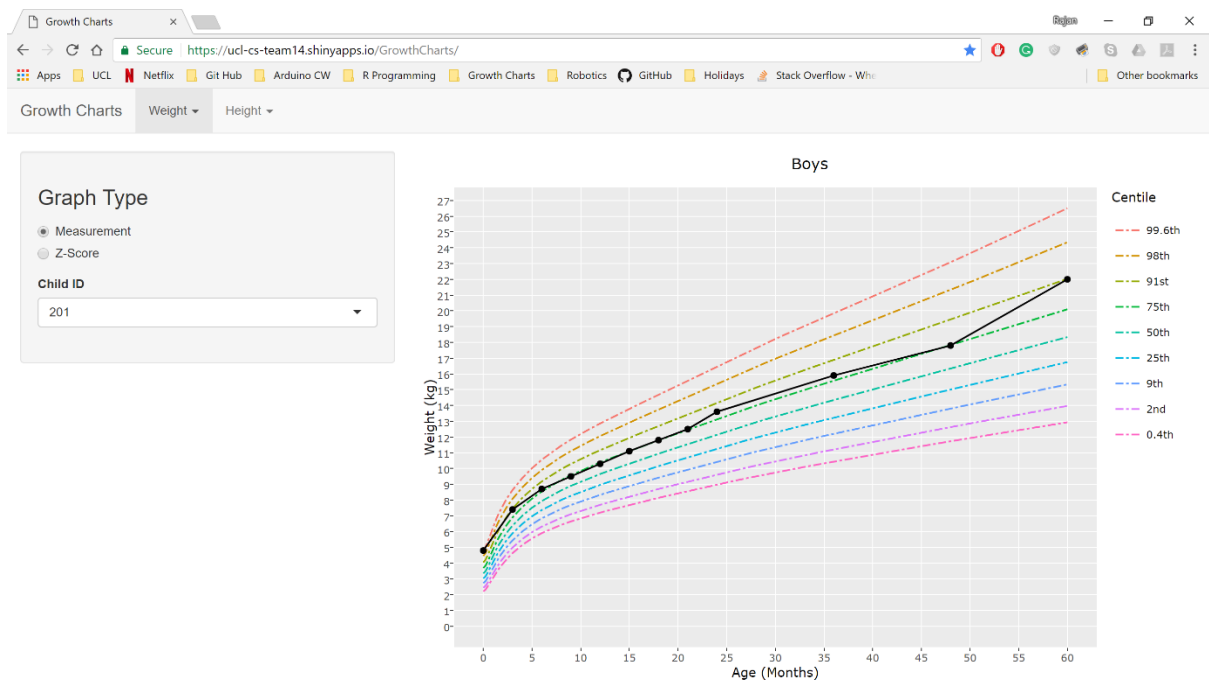
Figure 5.1: Google Chrome Screenshot
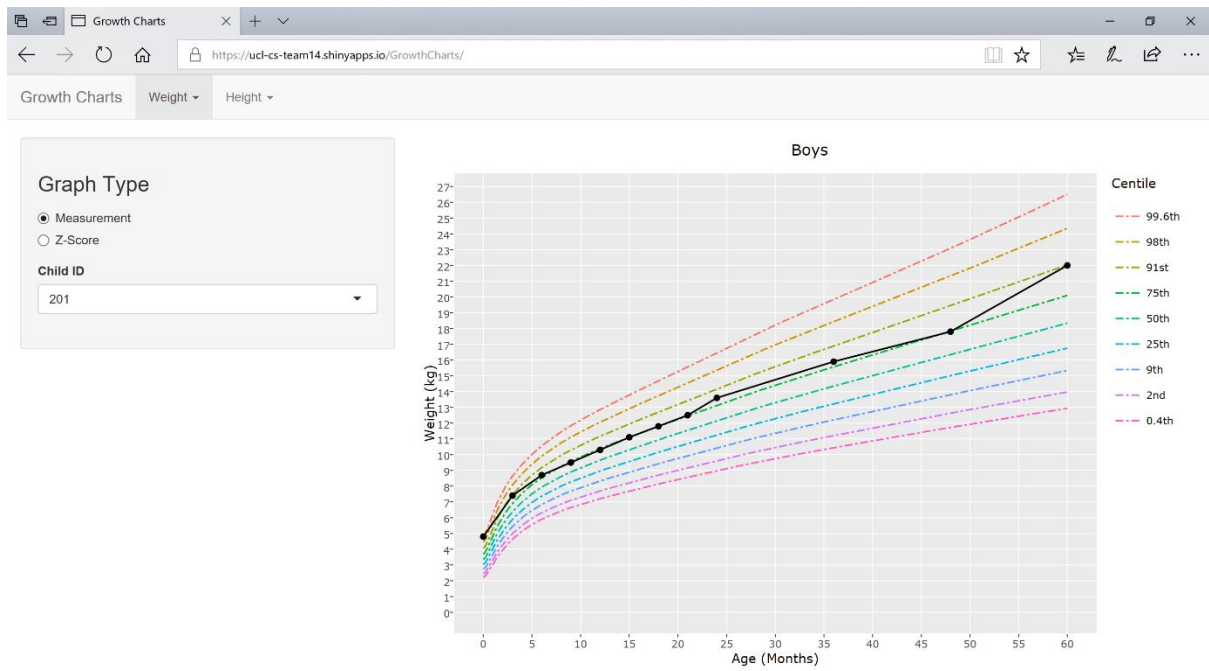
Figure 5.2: Microsoft Edge Screenshot
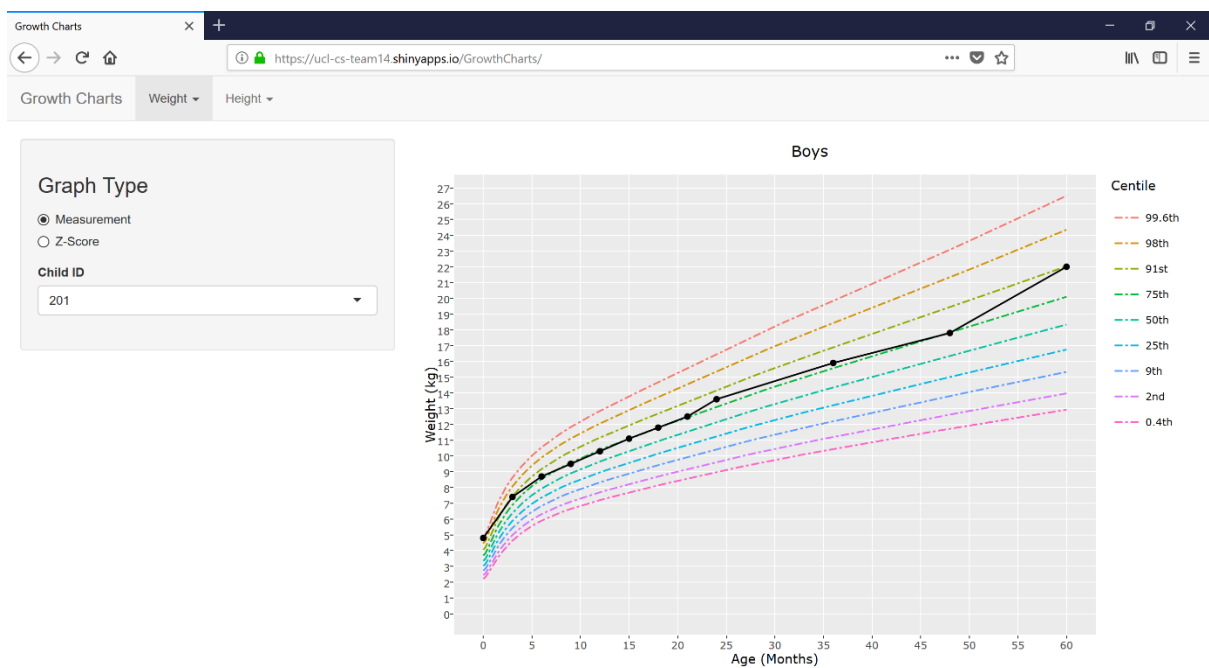
Figure 5.3: Mozilla Firefox Screenshot
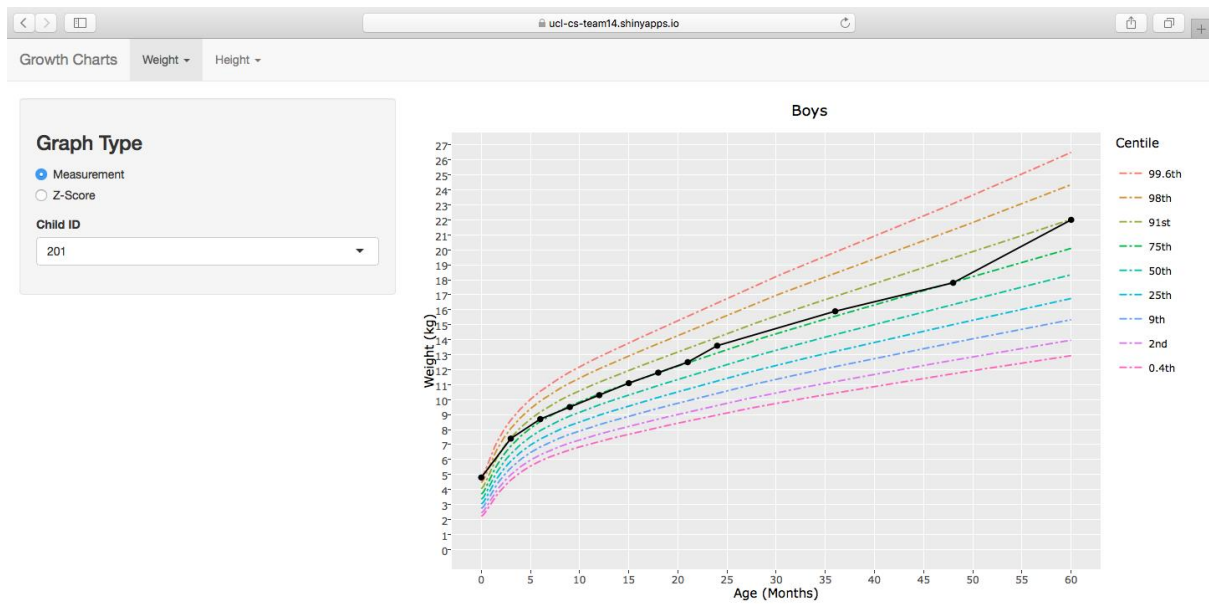
Figure 5.4: Safari Screenshot

5.2 Responsive Design Testing

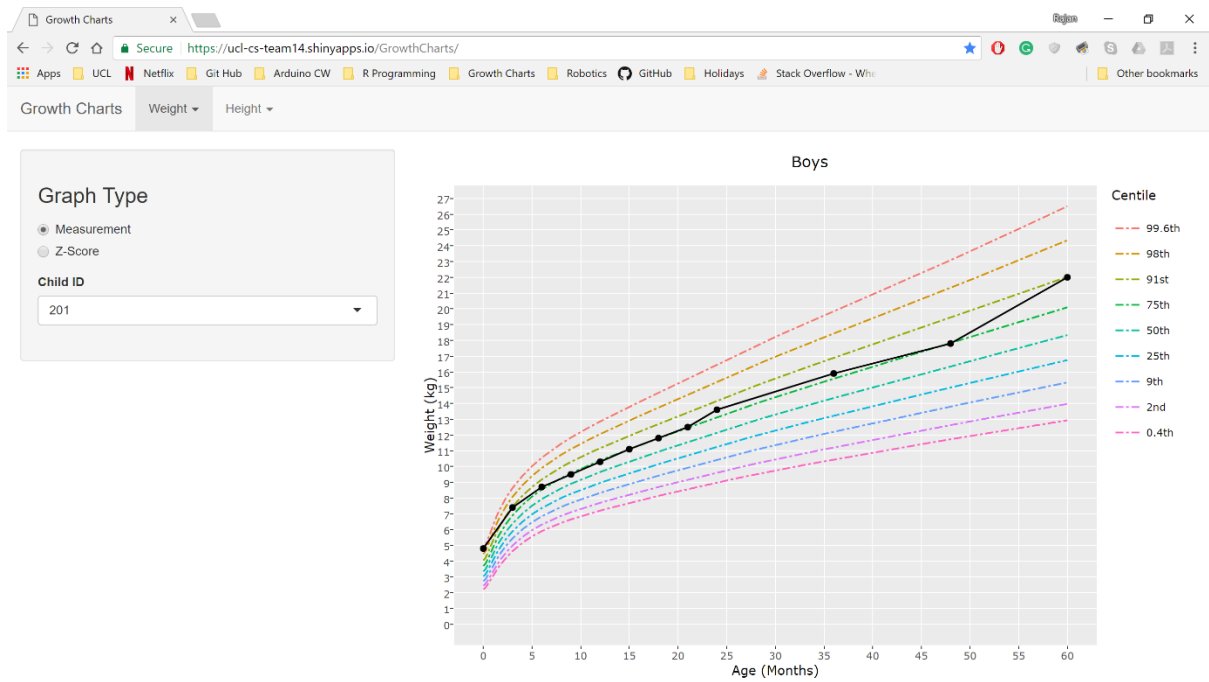| Screen	Result
Laptop (width 2736px)	Fully functional
iPhone 7	Fully functional
iPhone X	Fully functional
iPad	Fully functional

Figure 5.5: Laptop Screenshot
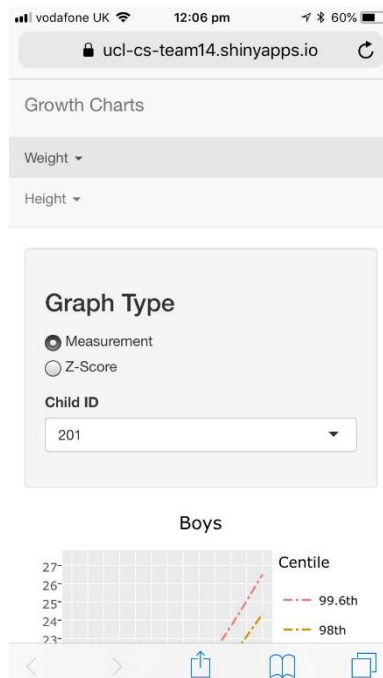
Figure 5.6: iPhone 7 Screenshot
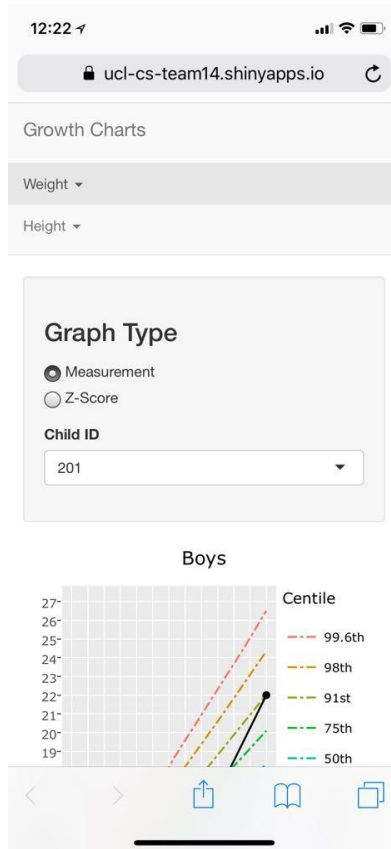
Figure 5.7: iPhone X Screenshot
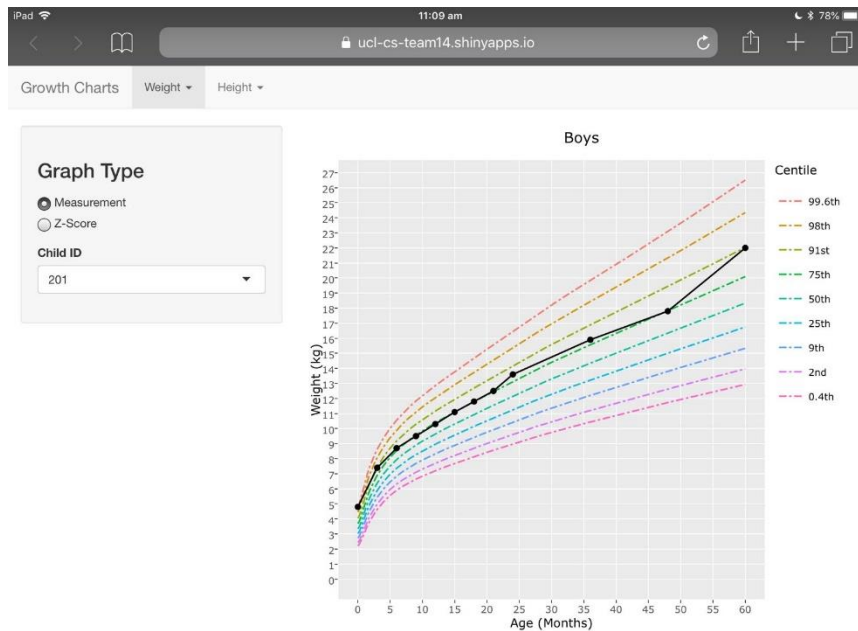
Figure 5.8: iPad Screenshot

5.3 User Acceptance Testing

A survey was carried out by different users (including our client) asking a variety of questions. These are the results from the survey:

Q1 Did the website ever freeze or crash when using it?
Multiple Choice

Choices	Totals
● Yes	0
● No	3

100%
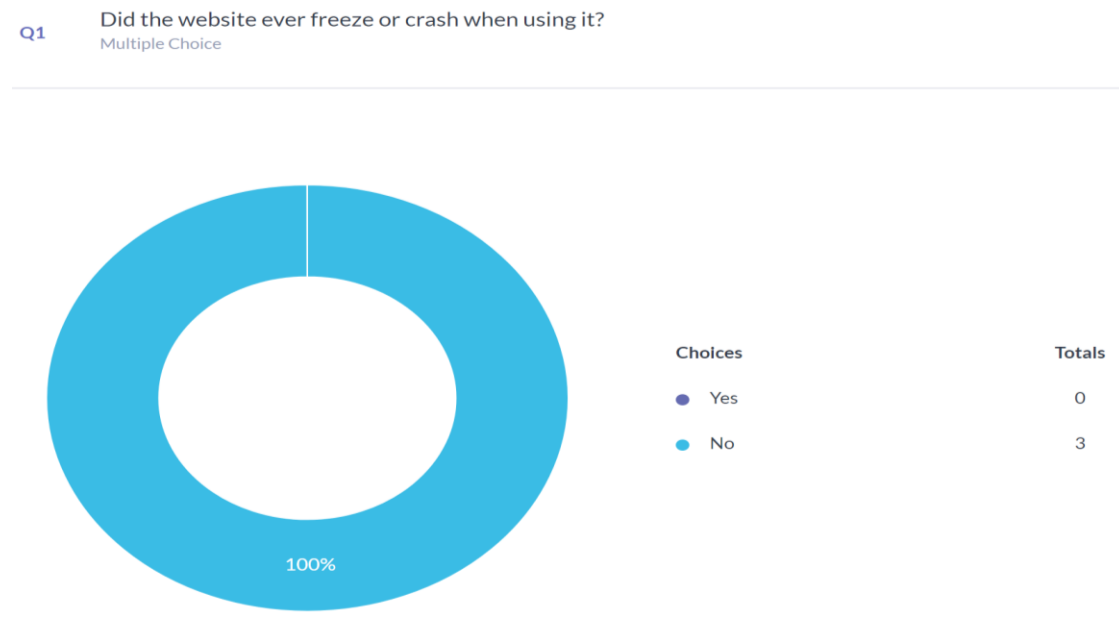
Figure 5.9: Question 1 Results

Q2 How would you rate the responsiveness of the web app?
Scale

4	6.5	5
Lowest	Average	Highest

Figure 5.10: Question 2 Results

Q3 What features do you like about the web app?
Essay

Latest Responses

You are able to see the z-score of the children

I like how you can tell whether the child is above or under average in relation to the national population.

The data visualisation is very effective, with clear, well-coloured and labelled points and lines.
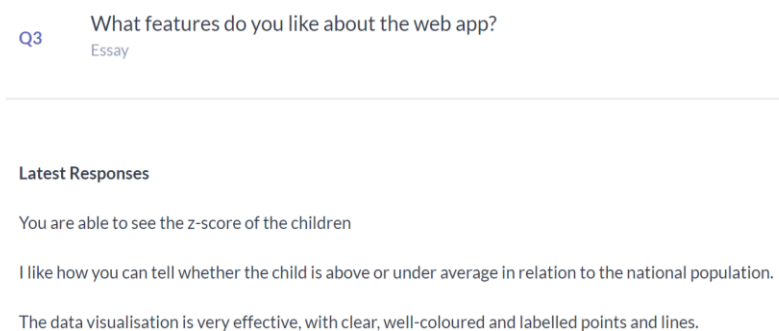
Figure 5.11: Question 3 Results

How user-friendly did you find the web app?
Multiple Choice

Choices	Totals
● Extremely user-friendly	2
● Very user-friendly	1
● Somewhat user-friendly	0
● Not very user-friendly	0
● Not at all user-friendly	0
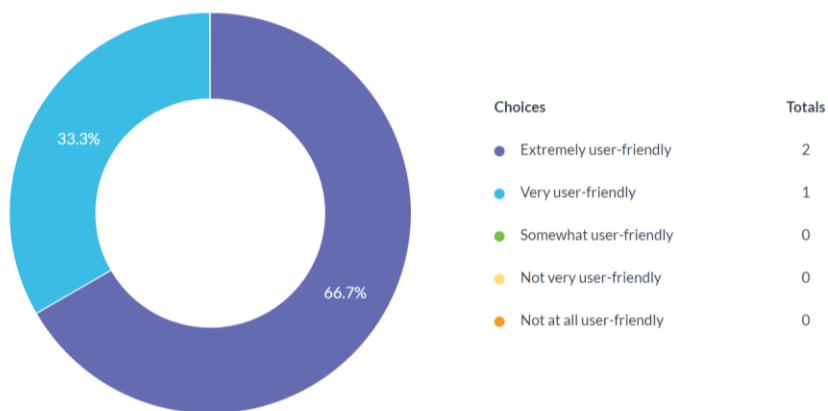
Figure 5.12: Question 4 Results

Q5 Have you encountered any bugs whilst using the web app? (If yes then please outline some of the bugs you encountered)
Multiple Choice

Choices	Totals
● No	2
● Yes	1
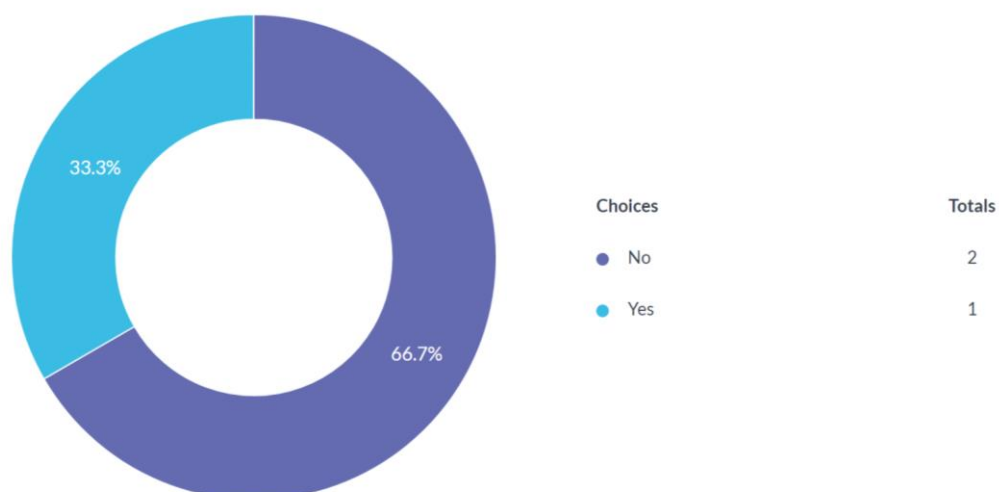
Figure 5.13: Question 5 Results

Q6 What do you think can be improved with the web app and are there any features that could be added into the web app?
Essay

Latest Responses

Ability to add data for a new child

NA

These were discussed with the team, but they ran out of time. Adding a BMI (body mass index) chart would be useful, and providing a palate to alter the colours, line types, line thicknesses and plot characters would make it more customisable.

Figure 5.14: Question 6 Results

Q7 What device are you using the web app on?
Multiple Choice

Choices	Totals
Laptop	0
Mobile Phone	1
Tablet	0
Desktop Computer	2

33.3%

66.7%

Figure 5.15: Question 7 Results

21

What operating system are using the web app on?
Multiple Choice

Figure 5.16: Question 8 Results

Q9 What internet browser are you using the web app on?
Multiple Choice
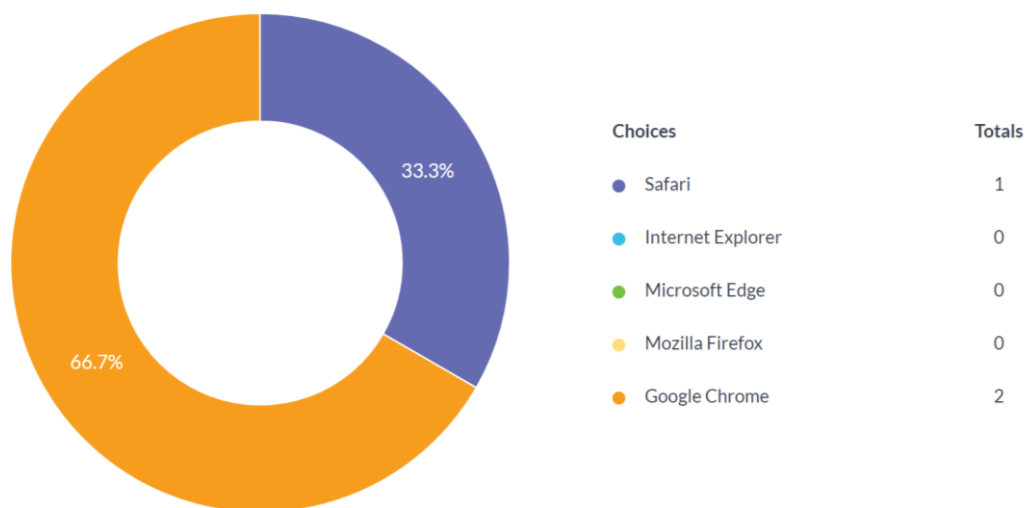
Figure 5.17: Question 9 Results

Q10   Is there anything else you would like the add about the web app?
       Essay

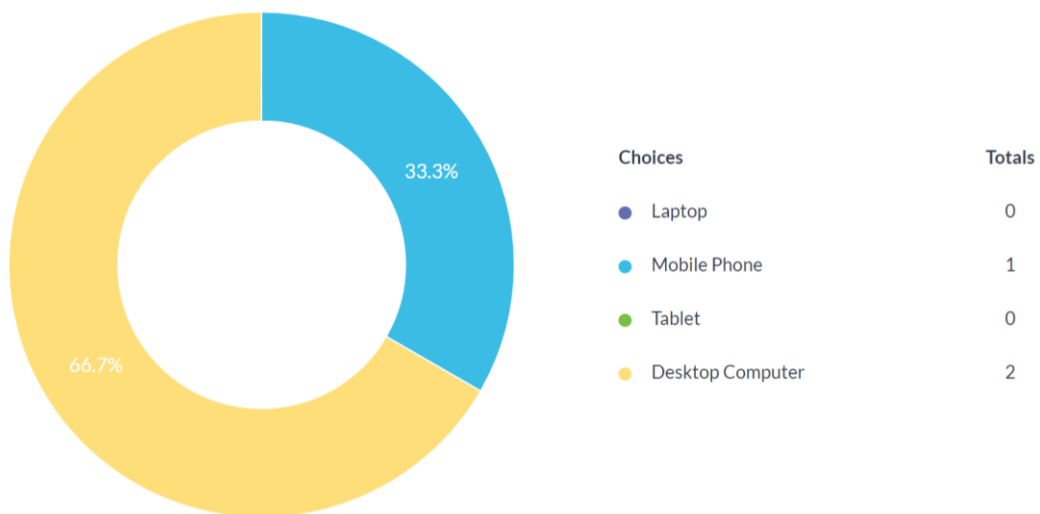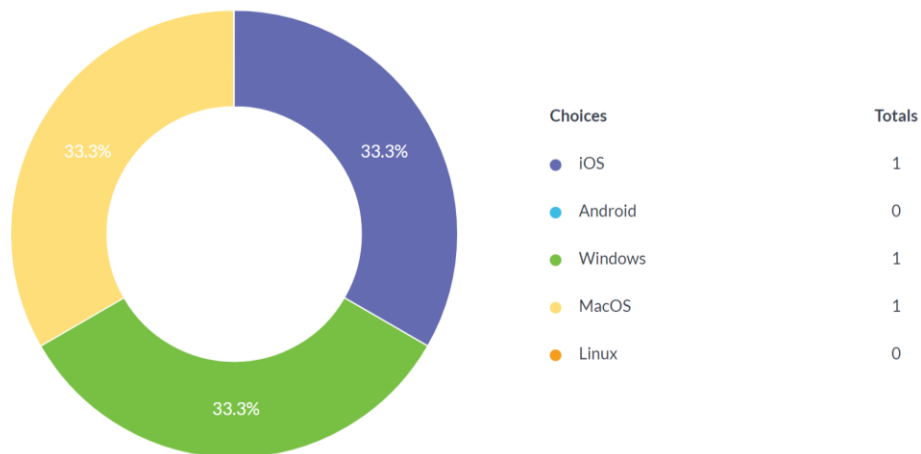**Latest Responses**

No

NA

It works well and does the job.

Figure 5.18: Question 10 Results

Overall, the survey gave positive feedback and only one bug was outlined (discussed in *6.1.3 Bug Table*) as well as some improvements that could be made (discussed in *6.3 Future Work*).

6 Conclusion and future work

6.1 Summary of Achievements

6.1.1 Achievement Table

ID	Requirements	Priority	State	Contributors
1	Plotting of serial measurements of an individual on a normal growth chart with centile detection	Must	✓	All
2	Plot height and weight growth charts	Must	✓	All
3	Have web app functionality	Must	✓	All
4	Use GOSH's data to produce growth charts	Must	✓	All
5	Show some sort of growth trajectory using the data that is inputted	Should	✓	Rajan
6	Integrated with SMART on FHIR so it can be compatible with any health centre that uses SMART on FHIR for their data storage	Should	X	All

7	Have some functionality to output data to files or in a pdf format	Could	X	All
8	Have some functionality for data security	Could	X	All
9	Have some functionality for the application to run in a mobile browser	Could	✓	All
Key Functionalities (must have and should have)		**83%** completed		
Optional Functionalities (could have)		**33%** completed		

6.1.2 Contribution Table

Work packages	Rajan	Saleh	Sander
Client liaison	33%	33%	33%
Requirement analysis	33%	33%	33%
Research	20%	40%	40%
UI Design	25%	50%	25%
Programming	40%	30%	30%
Testing	33%	33%	33%
Progress Report	33%	33%	33%
Technical Report	33%	33%	33%
Poster Design	0%	0%	100%
Video Editing	0%	100%	0%
Overall contribution	**40%**	**30%**	**30%**
Roles	**Report Editor, Programmer, Front End Developer**	**UI Designer, Researcher, Programmer**	**Report Editor, Programmer, Tester**

6.1.3 Bug Table

ID	Bug Description	Priority
1	Centile values in measurement graph were displaying incorrect values	High
2	The hover text for the centile values in the measurement graph display the centile value twice (from survey results)	Low

Bug 1 has already been fixed. We encountered this bug during a meeting with our client and we saw that the z-score values were not matching up to the centile values. The bug was caused as the function that was creating the centile curves was creating a line of best fit and this was not what we wanted so we changed the source code and now the z-score values and the centile values match up.

6.2 Critical Evaluation of the project

User Interface:

From the survey results, testers believe that the website has a good user interface as it is easy to navigate through the website using the tabs and drop-down menus. The website will always have a title on each page of the website so the user knows what part of the website they are on. The visual aspects of the website make it easy for the user to understand what information is being displayed.

Functionality:

The website is easy to navigate through and is organised into tabs and nested tabs. From the survey results, the website is very user-friendly and so users find it easy to use the website and understand what the website does after telling them the background of the website.

Compatibility:

The website is compatible with any browser and on any platform, whether it is a laptop, desktop computer, mobile phone or a tablet. The users from the survey tested the website on different platforms and browsers and the website was compatible with those platforms.

Maintainability:

The website can be easily maintained. Currently, the website is being hosted on a shiny server, which has a numerous amount of feature and it is also very easy to update and expand the website. The source code just needs to be changed and added to and then the website can be redeployed on the shiny server.

Project Management:

The project was managed quite well. We had regular meetings with our client to discuss any issues and any details about the project as well as showing our client the progress of our website. The tasks for the project were split equally between the team to match our strengths in programming and designing. Also, the met up regularly once a week to discuss any problems we had and to discuss how to implement certain features into the website.

6.3 Future Work

The project could be extended in many different ways if there was more time to complete project. The website could display BMI centile values as well as weight and height. This would mean users would be able to see the BMI values of children and see how they compare with the centile values. Also, we could have implemented the SMART on FHIR data storage system so that the website would be able to handle the data of any health centre in the UK. Due to this feature not being implemented, the website uses sample data for the centile values as well as sample children data. Another feature that could be implemented is a customisation tool for the graphs so users can control certain aspects of the graph e.g. colour, layout etc.

References

Web Page

- https://shiny.rstudio.com/gallery/widget-gallery.html

- https://shiny.rstudio.com/articles/shinyapps.html

- https://plot.ly/ggplot2/

Appendix

A User Manual

The web application can be found at this URL: https://ucl-cs-team14.shinyapps.io/GrowthCharts/

B Deployment Manual

After the client has the source code for the application, they can host it on any server they like. Currently, the web application is being hosted on a Shiny server on a basic subscription package. The client can upgrade the subscription package to have new features like authentication and adding authorised users.

C Code Citation

No.	Function Name	Code File Name	Source
1	z2cent	server.R	https://github.com/statist7/sitar/blob/master/R/LMS2z.R