Pre-Operative Assessment Clinic

Team 9

Jonathan Choi **Nishchal Sen** **Sheng-Wen Huang**

COMP0067 App Design

MSc Computer Science

April 23, 2019

Department of Computer Science

University College London

Abstract

The **Pre-Operative Assessment Clinic** (**POAC**) application is a digital platform to screen patients prior to their operation and identify those that are more complex and who may need a **POAC** appointment and/or Multidisciplinary Team (MDT) involvement.

It was developed by UCL Design Module's Team 9 - Jonathan Choi, Nishchal Sen, and Sheng-Wen Huang - in conjunction with the Taunton and Somerset NHS Foundation Trust (TST). Currently, pre-operative assessments are conducted in hospital and take around 45 minutes. This is an inefficient use of patient and staff time. The **POAC** application is designed to be taken at home or in hospital with or without supervision.

Team 9 built a web and mobile app. The web app was built using Angular and is designed for use by NHS staff to manage assessment questions and generate reports of assessment and feedback results. The mobile app was built using Ionic and is designed for use by patients to view their assessment history and submit feedback. Both apps use Node.js to interface with an SQL database built using MySQL.

The project achieved all of the "Must Have" and "Should Have" requirements. Three of the seven "Could Have" requirements were removed during development (see **Requirements** section for further details). During the project inception stage, we agreed that the project would not involve addition of Nurse/HCA-input details such as blood test results; would not have gender-specific questions; and would not create or record patient appointments themselves.

Despite fulfilling the necessary requirements, our project does not, as yet, interface with the NHS records system MAXIMS, therefore further work is necessary to deploy the project to a live environment.

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Project Introduction

1.1.1 The Client

Musgrove Park Hospital is part of Taunton and Somerset NHS Foundation Trust.

Musgrove is a successful, progressive and ambitious District General Hospital, the largest in Somerset. It serves a population of over 340,000 with around 4,000 employees and with an excellent reputation for providing a comprehensive range of medical, surgical and specialist services.

An NHS Foundation Trust since 2007, the hospital has received consistently good external ratings from the Healthcare Commission. In September 2016 the Trust was awarded a Global Digital Centre (GDE) status and as result the Programme has transitioned from Phase 2 of the EPR Programme to a GDE Programme, which will run until 2020.

1.1.2 The Problem Statement

The **Pre-Operative Assessment Clinic** (**POAC**) application is a digital platform to screen patients prior to their operation and identify those that are more complex and who may need a **POAC** appointment and/or Multidisciplinary Team (MDT) involvement.

Currently, pre-operative assessments are conducted in hospital and take around 45 minutes and involves completing a paper form (see Appendix XX) which is then used to make an assessment of the patient. Generally fit patients are screened out, and other patients are referred on for additional attention.

This is an inefficient use of patient and staff time.

1.1.3 The Project Goals

The project brief was to develop a mobile application that can be used by patients to complete the assessment. The application is to be issued to patients digitally, to be completed at home. Patients are to log-in via their NHS number and surname and complete the assessment which will be saved in a database.

A web-based application is to be used by NHS users to access the database to administer the question set, search patients, and view results.

1.2 Project Management

Our team comprises the following UCL MSc Computer Science students:

- Sheng-Wen (Alice) Huang is a graduate of the National Taiwan University with a Master's degree in Civil Engineering. Her interests include dancing, jogging and travelling.

  Her focus is on data and databases.

- Jonathan Choi is an Economics graduate from the London School of Economics who has most recently worked at Barclays Bank. In his spare time, he enjoys bouldering and kickboxing.

He is project lead and responsible for testing.

- Nishchal Sen is a Mathematics graduate from UCL. His hobbies include karate and playing the guitar.

  His primary responsibility is design and coding.

1.3 Project Scheduling

Our project schedule was based on the timetable outlined by the UCL project coordinator, Dr. Fu, and was visualised using a Gantt chart.

The Gantt chart shows activities down the vertical axis, and time along the horizontal axis.

In line with the agile methodology, we dynamically revised this chart as the project progressed. This allowed us to accommodate changes to the requirements, and unexpected changes to the schedule, in terms of proceeding both more and less quickly than initially planned.

Our final post-hoc Gantt chart reflecting our actual timelines is shown below:

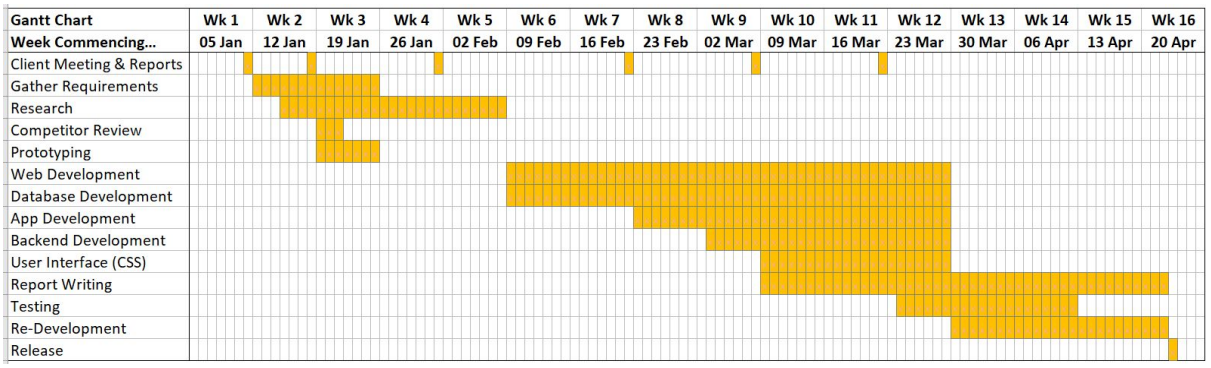

Figure 1.3.1 Post-Hoc Gantt Chart

2 Requirements

2.1 MoSCoW Requirements

We corresponded via calls and e-mail with the client to ascertain and categorise requirements.

The project requirements were defined under the Must, Should, Could, Won't (MoSCoW) framework [1] as below. The requirements were revised after week 1 following discussions between the UCL project coordinator, Dr. Fu, and the client. These revised requirements are in red below.

Project Goals:

- This project is to digitise the first page of the **Pre-Operative Assessment Clinic** (**POAC**) (items 1 to 4 and 6 in the paper form) only, which is wholly patient-facing **(Must Have)**
- If any "Yes" responses are received, the form will be passed to a nurse for additional processing **(Must Have)**
- Forms will be issued to patients digitally, to be completed at home **(Must Have)**
- Log-in will be via QR code or password and be linked to NHS number **(Should Have)**
- Website interface on which the NHS administrator (e.g. a doctor) can add / remove / amend questions **(Should Have)**
- Searchable database of results by patient name or NHS number **(Should Have)**

Additional features under consideration:

- Web version of assessment tool instead of / in addition to App **(Could Have)**
- Summary statistics of respondents and answers **(Could Have)**
- Return a summary page for each patient comprising all questions answered "Yes", additional comments, and regular medications **(Could Have)**
- User can edit their profile (name, password, e-mail address) via the App **(Could Have)**
- Interactive help on each question e.g. hover text, providing additional information **(Could Have)**
- Patients can input additional information via voice recording **(Could Have)**
- User can provide feedback on form **(Could Have)**

Features not under consideration:

- Completion and/or consolidation of Nurse/HCA-input details such as blood test results **(Won't Have)**
- Gender-specific questions **(confirmed with client – Won't Have)**
- Creation or recording of patient appointments **(Won't Have)**

2.2 Personas

Next, we developed personas. Personas are fictional characters used to represent user archetypes that would use the **POAC** application [2]. They are used to help understand user needs, experiences, behaviours and goals.

Figure 2.2.1 POAC Personas

2.3 Project Structure

Our project comprised three main elements - a mobile application, a web application, and a database.

Based on our brief and discussions with the client, we defined and agreed the following structure:

Part 1 - Mobile Application

For use by triage nurse and patient in hospital at first and by patients at home at a later stage

Patient logs in via NHS number and Surname:

1. **Assessment** - questions comprising Yes/No answers or free text fields.
2. **History** - patient can retrieve previous answers, if any.
3. **Profile** - patient can modify limited fields (first name, last name, contact details).
4. **Help** - frequently asked questions.
5. **Feedback** - form to be completed by patient. TST provided documentation and confirmed use of Likert Scale for feedback.
6. **Logout**

Part 2 - Web Application

For use by NHS staff / administrators

Login is via Staff ID

1. **View Patient** - user (e.g. a nurse) can search for patients and view survey responses.
2. **Assessment Management** - question wording and answer choices can be modified here. Questions can be created or deleted, re-ordered, edited for changes to wording, and toggled between Yes/No questions and Free Text.
3. **Assessment Dashboard** - summary results from the survey are shown here. Yes/No questions are presented as pie charts.
4. **Patient Feedback Dashboard** - similar to the Survey Dashboard, results from patient feedback are shown here, with bar charts representing feedback on a Likert scale.
5. **User Management** - the administrator can edit user details and deactivate users, e.g. if they have left the organisation.

Part 3 – Database

To store question and answer data and patient and user profile data

We used json/SQL for our own 'private' base, with no integration with MAXIMS during the creation and testing process.

· For the updated project scope, we developed wireframes and interactive prototypes. See *Section 4 Human-Computer Interaction (HCI) Design* for examples and details.

3 Research

3.1 Competitor Review

We reviewed several other popular apps in both mobile and desktop (where available) iterations, including Spotify, MyFitnessPal, Google Play Store, and iTunes. We examined the apps for usability and formatting features.

In particular, as our project would involve several tabs on both the mobile and web applications, we focused on how each application used tabs.

Tabs - size, text labelling, position and orientation, behaviour

Font size - readability and maximum text per page

Interaction - buttons, transitions, empty state screens (e.g. expected behaviour if the patient has no survey history)

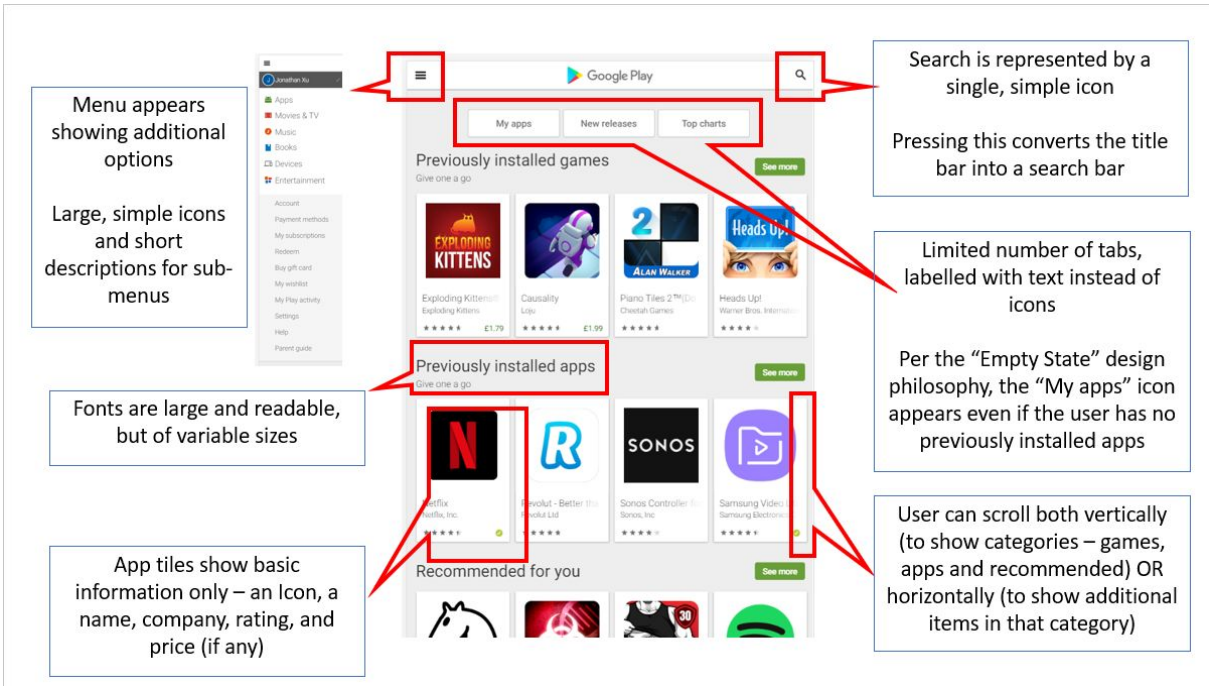

Figure 3.1.1 Google Play Store - "Competitor" Review

3.2 Technology Review

Several languages and frameworks were available for each of the components of our project.

Below, we describe the alternatives available for each component, and our considerations in making our selections.

Note we selected Visual Studio Code as our editor and the Angular framework for development. For project management, we used Git/GitHub.

- **Web App**
    - **PHP** - PHP is an ageing but widespread and relatively simple language that was specifically designed to work on the Web. It benefits from a broad code base with sufficient functionality for everything that our project needs to do. It also benefits from good portability and can be run on almost any server.
    - **Node.js** - Node.js employs a JavaScript single-threaded event looping model. Due to its design, servers can respond in a non-blocking way, and it can therefore be used to create I/O applications that can handle much larger numbers of requests than traditional servers. However, since Node.js is single-threaded, it is less efficient in CPU-intensive tasks, and is more suitable to be used on I/O-intensive applications.

PHP	Node.js
synchronous	asynchronous and event driven
multi-threaded	single-threaded
slower	faster
runs on the Apache web server (or IS web server on windows)	NMP does not need a web server
server-side language	
simpler to learn	requires more time to understand

Table 3.2.1 Comparison between PHP and Node.js

    - **Our Decision** – Due to its relative newness, support, and for the benefit of any future projects that might require Node.js support, we selected Node.js. Despite being more difficult to learn, we believe the investment in learning and using Node.js will be of benefit in our future work. Lastly, since we are building an online survey application which is more an I/O intensive application, Node.js is more appropriate.

- **Mobile App**
    - **Ionic** - Ionic is a framework that allows developers to create cross-platform applications using only JavaScript. It is open source and has very well-designed UI components.
    - **Swift** - Swift is a language created by Apple for iOS applications. As such, it is more powerful and supported but for Apple development only.
    - **Our Decision** – Ionic is expressly designed to be cross-platform and was therefore chosen so that we would not have to develop multiple platform-specific versions of our app.

- **Database**
  - **MySQL** - MySQL supports many data types and third-party tools make it very easy to use. It can handle a large amount of data yet still has good performance in terms of speed. Furthermore, MySQL has many security features.
  - **SQLite** - SQLite is file-based, which makes it highly portable. The library is simple and easy to learn.
  - **Our Decision** – We were relatively indifferent between the two implementations. However, given the personal and potentially sensitive information to be collected in our project, we chose MySQL for its security features.

4 Human-Computer Interaction (HCI) Design

Human-Computer Interaction (HCI) focuses on the design of computer technology [3]. In particular, and as the name suggests, it concerns the interaction between users and computers.

4.1 Process Overview

The design stage comprised the four steps before beginning the implementation of our final design in software.

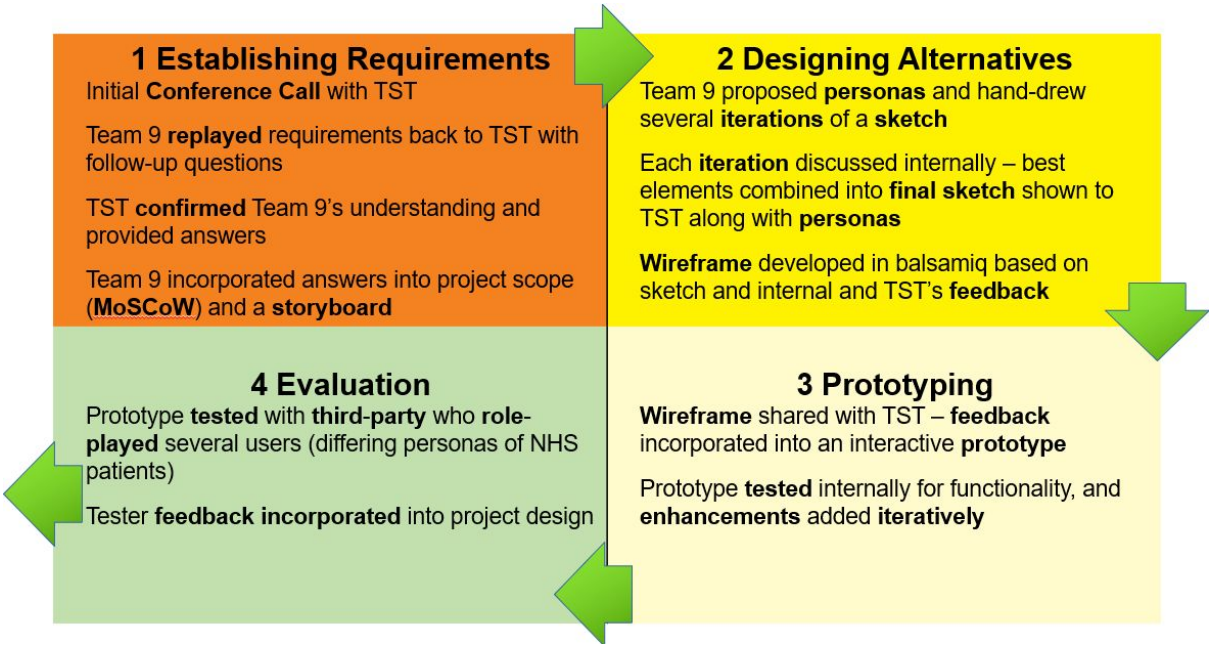

Figure 4.1.1 Process Overview Flow Diagram

4.2 Storyboarding

During the process of establishing the requirements and proposing personas, we produced a storyboard to illustrate a typical use and context over time. This "provides context ... [using] a sequence of images to tell a more complete story [including] where, ... personalities ... and ... details about ... actions" [4]

Figure 4.2.1 Narrative Storyboard

4.3 Wireframes

We used templates to sketch the mobile and web apps. These were used to assess the usability of our proposed app.

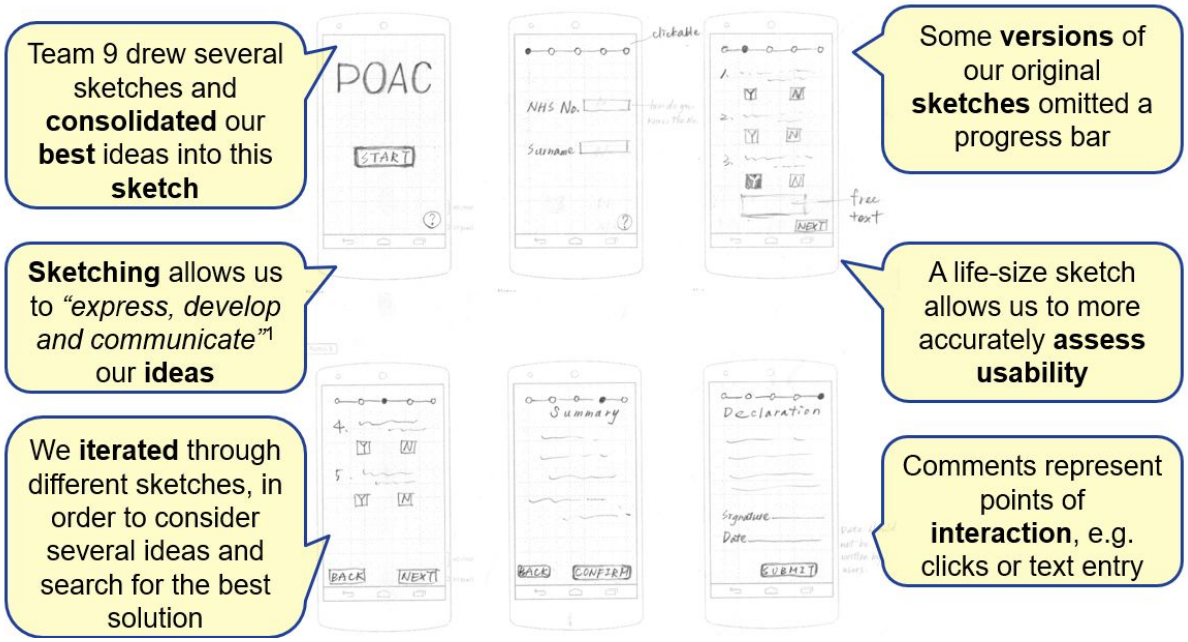

Figure 4.3.1 Hand-Drawn Wireframes

4.4 Prototypes

Next, we used prototyping software to build minimally interactive wireframes or "prototypes". These were again developed iteratively based on our sketches and client and tester feedback. Mark-up was

used to add commentary, and alternative screens were used to map pathing of user choices e.g. “Yes” and “No” responses to questions.

The wireframe serves as a type of site-map, allowing the designer to demonstrate progression through use of the app.

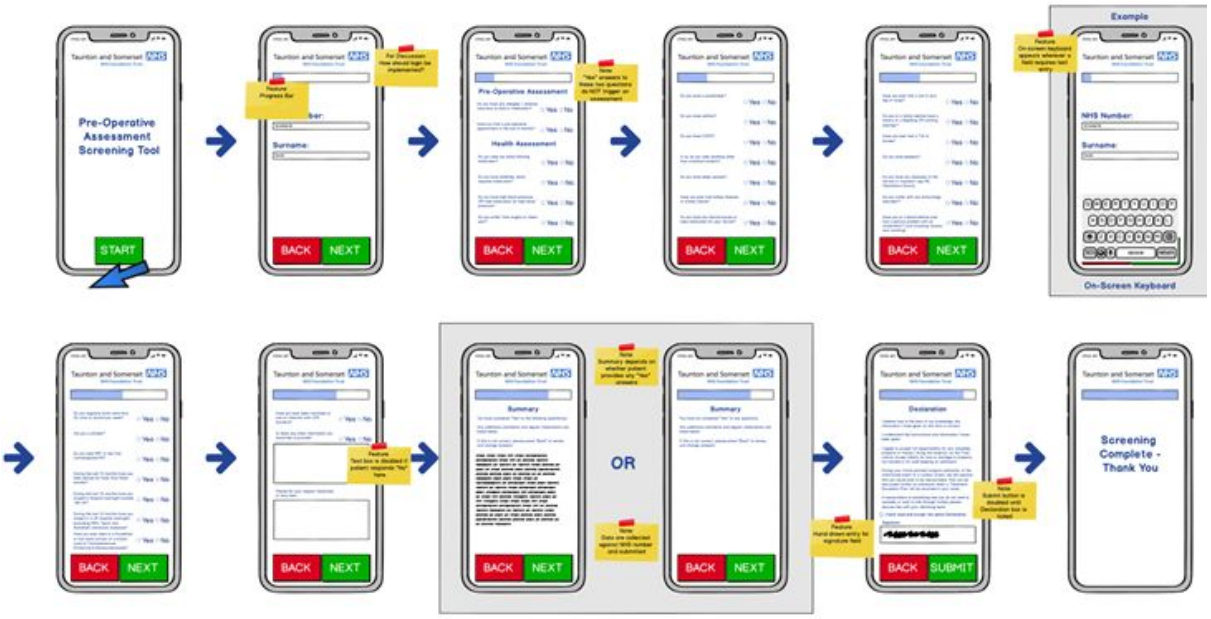

Figure 4.4.1 Interactive Prototypes

4.5 Design Testing

The last stage of our HCI Process was testing. A third-party user user was asked to test our prototype unsupervised, then again with our commentary and guidance while role-playing different personas.

We recorded user feedback and considered each comment. Feedback that was appropriate, within scope of the project, and actionable was included in subsequent iterations of the app’s design.

5 Design and Implementation

5.1 System Architecture

Our System Architecture Diagram shows the high level interaction between front- and back-ends, and the database.

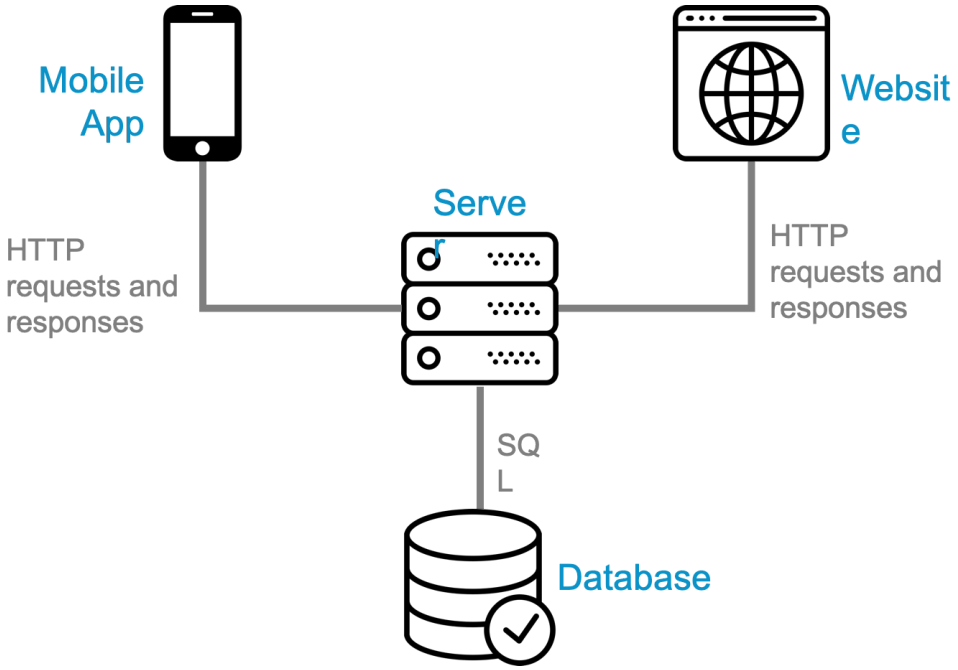

Figure 5.1.1 System Architecture

Element	Users	Application	Notes
Website	NHS Staff (Admin)	Angular	The website makes HTTP requests to the server to obtain, edit, and delete questions and users
App	NHS Patients	Ionic	Similarly, the mobile app makes HTTP requests to the server to obtain user history, personal information, and sends survey answers to store in the database
Server	N/A	Node.js	The server interacts with the website, app and database It also handles tokens and password management*
Database	N/A	MySQL	Data here is persistent, and can be stored and retrieved

Table 5.1.1 Description of System Architecture Components

The server uses the express.js framework and employs the RESTful API for Create, Read, Update and Delete (CRUD) Operations.

Tokens – valid web or mobile users generate a unique token, valid for one hour, which gets sent to the respective front-end, using the **jsonwebtoken** query.

This token is used to identify and authenticate the users' requests.

Passwords – When a new user is created, the password is hashed and salted instead of being stored unsecurely in the database as plaintext. For this, we use the **bcrypt** library, which is also used to verify users on log-in.

The backend is connected to the database via the **MySQL** library.

5.2 Entity Relationship Diagram

Our Entity Relationship Diagram demonstrates how our project database has been structured.

It includes the variables and types in each table, primary and foreign keys, character limits (if any), and how tables interact with each other.

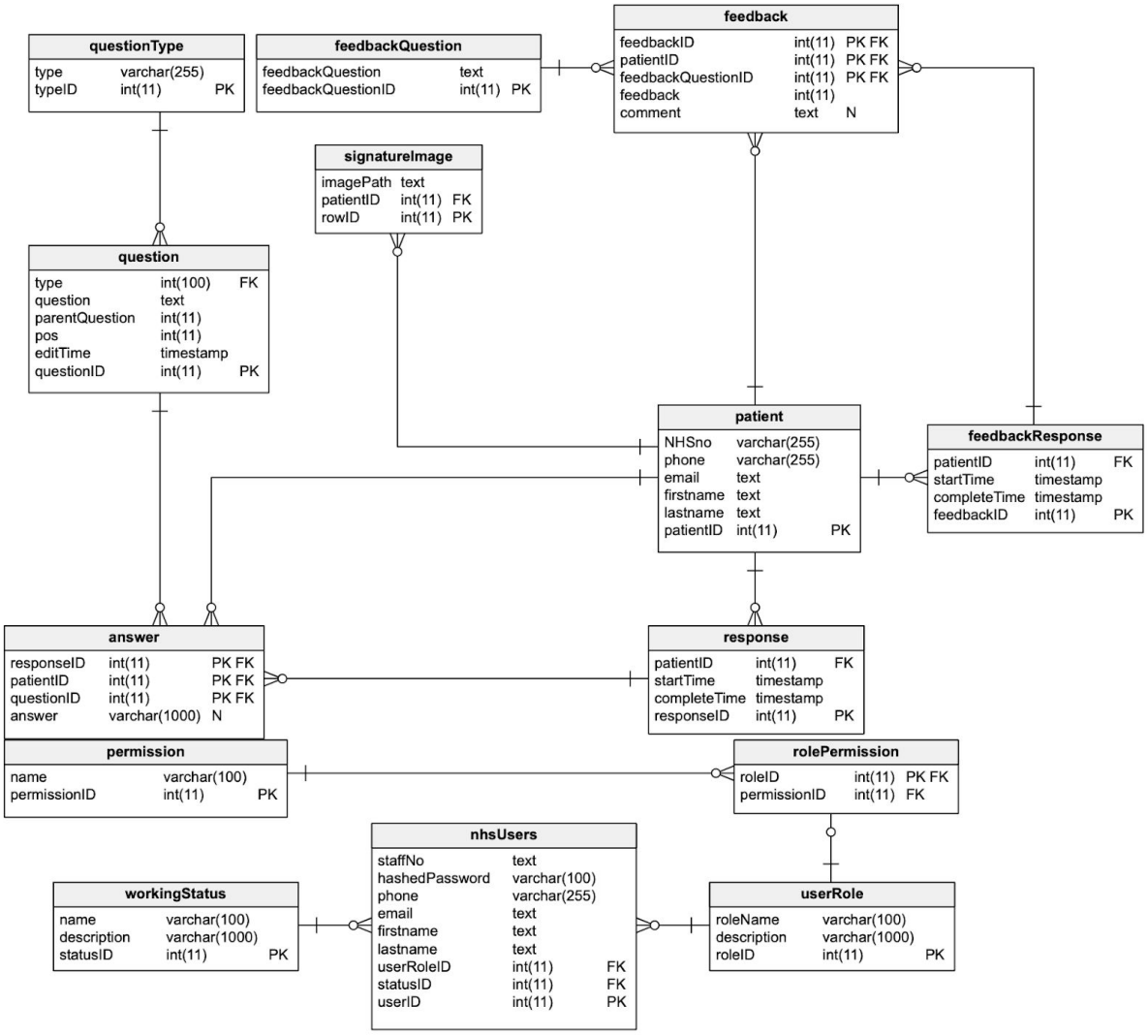

Figure 5.2.1 Entity Relationship (ER) Diagram

6 Testing

Testing is an important part of development, needed for the elimination or minimisation of bugs that can have useability or security implications.

Automatic testing tools are available but given the limited scope of our project, we tested our applications manually.

During our development, we hosted our web and mobile applications and our backend locally. This allowed us to compile continuously as we made changes, and performance basic checking and testing.

Towards the end of our development, we hosted our project on a virtual machine using Microsoft Azure and asked three associates to test each element of our project.

As we did not have a jailbroken Apple device, we were unable to publish to a device without paying for a developer license. To save costs, we instead tested the iOS version of our mobile application in browser simulators on Mac, PC and Linux instead. Android follows a more open philosophy which allowed us to sideload an Android version of our .apk of our mobile application to a Samsung Galaxy S7 device, as well as perform testing in simulated devices in browsers.

Our team of three was fortunate to be running one each of Linux, MacOS and Windows. As a result, we were able to test across operating systems. Accordingly, we performed testing of our web application on Chrome, Edge, Firefox and Safari browsers.

6.1 Use Case Testing

Use case testing was performed by actioning each use case in the respective web or mobile application.

During our tests, all 14 Use Cases across both applications were completed as expected.

ID	Actor	Description	Test Result
UC001	NHS User	Log-in using staff number and password	Success
UC002	NHS User	Search patients by NHS Number and view assessment history	Success
UC003	NHS User	Perform CRUD on Assessment questions	Success
UC004	NHS User	View summary of all patient assessments	Success
UC005	NHS User	View summary of all patient feedback	Success
UC006	NHS User	Perform CRUD on NHS Users (Admin only)	Success
UC007	NHS User	Logout	Success
UC008	Patient User	Log-in using NHS number and surname	Success
UC009	Patient User	Complete POAC Assessment	Success
UC010	Patient User	View previous assessment history, if any	Success
UC011	Patient User	Complete Feedback if desired	Success
UC012	Patient User	View Frequently Asked Questions (FAQs)	Success
UC013	Patient User	View own profile	Success
UC014	Patient User	Logout	Success

Table 6.1.1 Use Case Testing Status

We also tested with different devices, browsers and devices. For these tests, we iterated through the full workflow, with a particular focus on a single use case per device tested. Again, these tests all completed successfully.

Description	ID	Test Result	Comment
Mac - Safari Browser	UC001	Success	Tested on a Macbook Pro running MacOS 10.14.3 Mojave
PC - Firefox Browser	UC002	Success	Tested on a Dell 17R running Windows 10 Home
PC - Chrome Browser	UC003	Success	Tested on a Dell 17R running Windows 10 Home
PC - Edge Browser	UC004	Success	Tested on a Dell 17R running Windows 10 Home
Linux - Firefox Browser	UC005	Success	Tested on a Lenovo 320S running Ubuntu 16.04
Linux - Chrome Browser	UC006	Success	Tested on a Lenovo 320S running Ubuntu 16.04
Mobile - Chrome Browser	UC007	Success	Tested on a Samsung Galaxy S7 running Android 8.0 Oreo
Samsung Galaxy S7 - Android	UC008	Success	Tested on a Samsung Galaxy S7 running Android 8.0 Oreo
Samsung Galaxy S7 - Android	UC009	Success	Tested on a Samsung Galaxy S7 running Android 8.0 Oreo
Samsung Galaxy S7 - Android	UC010	Success	Tested on a Samsung Galaxy S7 running Android 8.0 Oreo
iPhone 5/SE (Emulated)	UC011	Success	Emulated in a Chrome browser window
iPhone 5/SE (Emulated)	UC012	Success	Emulated in a Chrome browser window
iPad (Emulated)	UC013	Success	Emulated in a Chrome browser window
iPad (Emulated)	UC014	Success	Emulated in a Chrome browser window

Table 6.1.2 Use Case Testing on Different Devices

6.2 User Testing

For User Testing, we explained the purpose of the application to three associates and showed them our fictional personas. We asked them to each imagine interacting with the web or mobile application as one of the respective personas.

Tester 1

Role: NHS Administrator

Device: PC running Windows 10 Home / Edge Browser (15 inch screen)

Comments:

- Have to double-click arrows before questions are re-ranked
- **Y/N - Text** toggle only updates after refreshing screen
- Would like to see more extensive test data in dashboard screens

Tester 2

Role: NHS User (non-Administrator)

Device: MacBook Pro running MacOS 10.14.3 Mojave / Safari Browser (15 inch screen)

Comments:

- Formatting looks ugly on Assessment Management screen when browser window is resized e.g. icon alignment
- Axis labels on graphs for feedback screen are disproportionate when browser window is resized
- Editing questions defaults to **Y/N** type instead of inheriting source question's type

- What happens if a question type is toggled after it has been used?

Tester 3

Role: Patient

Device: Samsung Galaxy S9 running Android 9.0 Pie / Chrome Browser (6.2 inch screen)

Comments:

- Easy to use
- A lot of scrolling on assessment screen
- Could not find log-out button until prompted
- Can submit unlimited number of assessments per day
- Can submit unlimited number of feedback per day
- Dependent questions need to be filled in before submitting is allowed

6.3 Conclusions

We developed a working product across three architectures (for each of the web application, mobile application, and backend) which was functional and well-received by testers. Apart from the bugs identified and listed below, we did not find any major issues.

7 Conclusions and Future Work

7.1 MoSCoW Assessment

Must and Should Have Requirements:

We achieved all of the Must and Should Have Functionalities.

Note that following additional discussions with the client, we determined that log-in would be via the combination of NHS Number and Surname rather than via Password or QR code.

Could and Won't Have Requirements:

- As we had developed a web application for the NHS users and a mobile application for patients, we decided not to develop a third web-based application for patients. However, this could be explored with additional time and resources.
- Initially, we considered allowing patients to edit their own profiles. However, once we established that patients would be logging in using their NHS Number and surname, we realised that this would present a risk to the safety of patient data. In order to implement this change, we would instead change the underlying login structure and process so that access did not depend on fields that can be modified by patients.
- Voice-recorded inputs were discussed briefly as part of the requirements gathering. However, ultimately this was not implemented.

Requirement	Priority (... Have)	Status	Contributor
Digitise first page of POAC	Must	Implemented	Team
Can issue digitally to patients	Must	Implemented	Team
Log-in via Password/QR	Must	Log-in via NHS Number / Surname	Nishchal
Website interface for NHS users	Should	Implemented	Nishchal/Team
Searchable database of patients	Should	Implemented	Sheng-Wen/Nishchal
Web version of assessment tool	Could	Not implemented	N/A
Summaries of responses	Could	Implemented	Jonathan
User can edit profile	Could	Not implemented	N/A
Voice recorded inputs	Could	Not implemented	N/A
User can provide feedback	Could	Implemented	Nishchal
Input test results e.g. blood test	Won't	Not implemented	N/A
Have gender-specific questions	Won't	Not implemented	N/A
Creation of patient appointments	Won't	Not implemented	N/A

Table 7.1.1 Summary of Achievements

7.2 Known Bugs

Description of Problem / Bug	Severity	Remediation
Users can submit assessments unlimited times	Low	Implement control of maximum one submission per day
Users can submit feedback unlimited times	Low	Implement control of maximum one feedback per assessment
Frequently Asked Questions page is missing	Low	Update Frequently Asked Questions once provided from client
Toggling question type only updates after page refresh	Medium	Code refactoring to refresh immediately
Loss of legibility on smaller screens	Medium	Code refactoring to manage fonts and icons on resizing
General stability and error checking	Medium	Add error and consistency checks

Table 7.2.1 Known Bugs

7.3 Contribution Table

The team was fortunate to benefit from a mix of skills and experience which allowed us to specialise in our own areas of expertise.

This is reflected in the splits below, e.g. Sheng-Wen's design skills are reflected in a high contribution to prototyping and the user interface; and her Database experience was useful in building our Entity Relationship diagrams.

Pleasingly, the final contribution metric split almost exactly between the three team members and this reflected our own feeling that project work was evenly distributed.

Activity	Jonathan	Nishchal	Sheng-Wen
Role	Project lead; testing	Design; coding	Data; databases
Client Meetings	60%	20%	20%
Gather Requirements	33%	33%	33%
Research	15%	15%	70%
Competitor Review	60%	20%	20%
Prototyping	25%	25%	50%
Web Development	30%	50%	20%
Database Development	10%	40%	50%
App Development	0%	70%	30%
Backend Development	10%	60%	30%
User Interface (CSS)	20%	20%	60%
Report Writing	60%	20%	20%
Testing	60%	20%	20%
Re-Development	10%	60%	30%
Release	70%	15%	15%
Overall Contribution	**33%**	**33%**	**33%**

Table 7.3.1 Contribution Table

7.4 Conclusions

What Went Well?

The first phase of the project was largely spent on learning principles of Human-Computer Interaction (HCI). While this was useful and interesting, the benefits of HCI might be more pronounced in a larger project, where the design implications are more impactful.

We made extensive use of available collaboration tools, including Git/GitHub, a group WhatsApp, a Group Slack Channel, and shared documents on Google Drive. While Git/GitHub entailed some up-front effort to learn, it proved to be an intuitive and powerful tool to share development work.

We found the regular fortnightly check-ins with Dr. Fu and the client to be useful in enforcing discipline on our timelines, and to reflect on and review the fortnight's progress.

What Were Problems?

Our initial project brief was to build a **POAC** assessment tool for patient use only, which would have required development in just a mobile environment. However, on further review and in consultation

with our client and the UCL project organiser, Dr. Fu, requirements were expanded to include a website based administrative tool for NHS users to manage the assessment.

This necessitated significantly more development work, not only in learning the relevant languages to develop the website, but also in building the database interfaces to modify the assessment and return assessment and feedback results. The change also entailed additional effort to coordinate workstreams and consolidate work.

As requirements and the choice of technologies and modules evolved, frequent refactoring was required. For example, the client asked for a change to the login process in which staff ID was replaced by username. We also experimented with Angular Chart and Google Chart plugins before settling on Chart.js. The amount of refactoring required may have been reduced with more up-front planning and communication.

Lastly, we found hosting to be a particular challenge. The team was unfamiliar with the process of provisioning and accessing a virtual machine. Subsequent to that, loading our database and application on to the virtual machine and making it publicly accessible took a substantial additional commitment of time and effort.

Final Thoughts?

Our project development would benefit from a set of stricter internal timelines in future. This would allow us additional time to review and test. However, we did find the introduction of a web application to our project scope - and the interdependencies that this introduced - imposed a significant time and effort cost on our development. For example, the work in moving our project between a live test state, and back into its offline development state was doubled by the increase in scope.

Overall our project was a qualified success, with all of the mandatory requirements (Must and Should Haves), and some of the optional requirements (Could Haves) met.

7.5 Future Work

Three **Could Have** requirements were not implemented, namely a web-application version of the patient application; voice-recorded responses; and patient modification of their own profiles. Given more time, we would like to explore each of these.

- A web-application version of the patient application would be built in Angular, similarly to the NHS User web application. Note that the NHS User web application is accessible and works on mobile devices.
- We would use publicly available APIs to implement voice-recording of responses and tag these to the relevant “Free Text” questions. An additional step would be using text-to-voice API’s so that visually-impaired patients could also hear questions. Voice recognition could also be used to convert spoken responses back into text.
- Patient modification of their own profiles would require an overhaul of the login structure and process to avoid users modifying their own login credentials.

There were several items in initial discussions which were categorised as **Won’t Have** e.g. allowing medical test results to be input into the application. This has ramifications for user privacy and data security so would have only be implemented in conjunction with larger changes to the overall

application structure (e.g. patients would have to register log-in details and passwords securely in order to have sensitive medical data recorded and retrievable).

Finally, our application was developed as a standalone application with dummy data and no integration with the NHS patient data system, MAXIMS. This was to ensure separation of our project team from sensitive real-world patient data. An additional step would be to integrate our database into MAXIMS.

References

[1] Agile Business Consortium, “The DSDM Agile Project Framework”, 2014. [Online]. Available: www.agilebusiness.org/content/moscow-prioritisation [Accessed: 22-Apr-2019]

[2] R. Dam, T. Siang, “Personas - A Simple Introduction”, 2019. [Online]. Available: www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-them [Accessed: 22-Apr-2019]

[3] Interaction Design Foundation, “Human-Computer Interaction (HCI)”, 2019 [Online]. Available:

www.interaction-design.org/literature/topics/human-computer-interaction [Accessed: 22-Apr-2019]

[4] S. Greenberg, et al, “Sketching User Experiences”, Morgan Kaufmann 2011

Appendices

Appendix A - System Manual

A **video demonstration** of our web and mobile applications can be viewed here:

https://mediacentral-upload.ucl.ac.uk/LibraryContent/DownloadMedia/17638

Our **web application** is hosted on a Microsoft Azure Virtual Machine with the following DNS:

poac.uksouth.cloudapp.azure.com

It is hosted on port 8080, so is accessible via: poac.uksouth.cloudapp.azure.com:8080

1. An NHS User with administrative rights: **Username:** djuniper0 **Password:** theone
2. An NHS User without administrative rights: Username: adehooch1 **Password:** theone

Our **mobile application** can be accessed by side-loading the application apk on an Android device.

- A patient with past assessment history: **NHS No.:** 1612010321699 **Surname:** Shields

The **database** can be accessed directly using phpmyadmin:

poac.uksouth.cloudapp.azure.com/phpmyadmin

… with **Username:** Team-9-POAC and **Password:** root

Appendix A.1 - Web Application

View Patient Screen

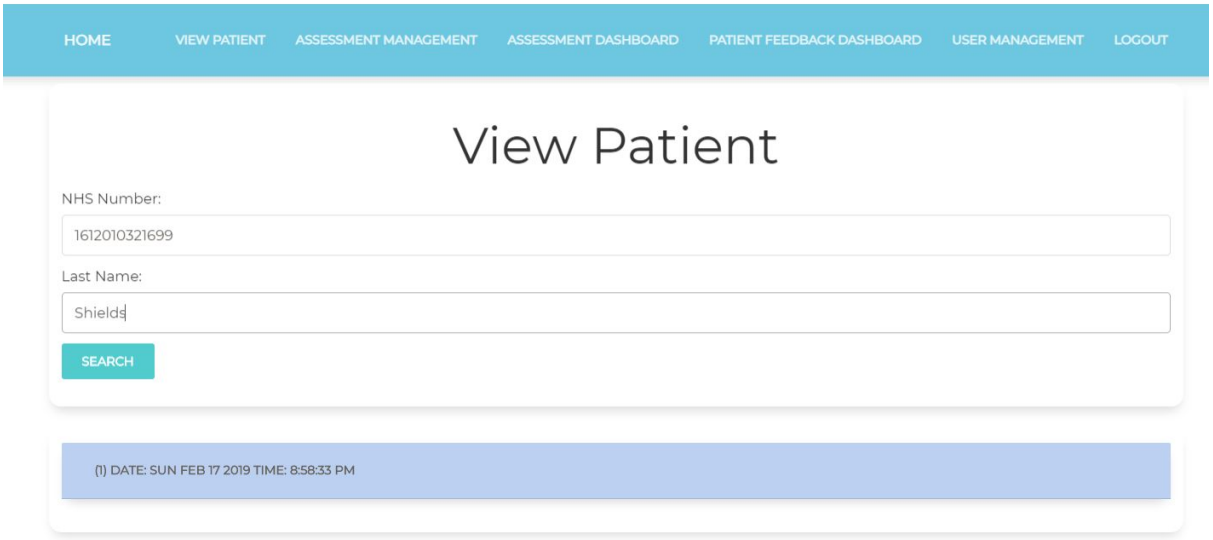

Figure A.1.1 View Patient Screen (Web)

1. Enter an NHS Number
2. Enter the corresponding Last Name
3. Press Search
4. A list of past assessments will appear - click on each in turn to view

Assessment Management Screen

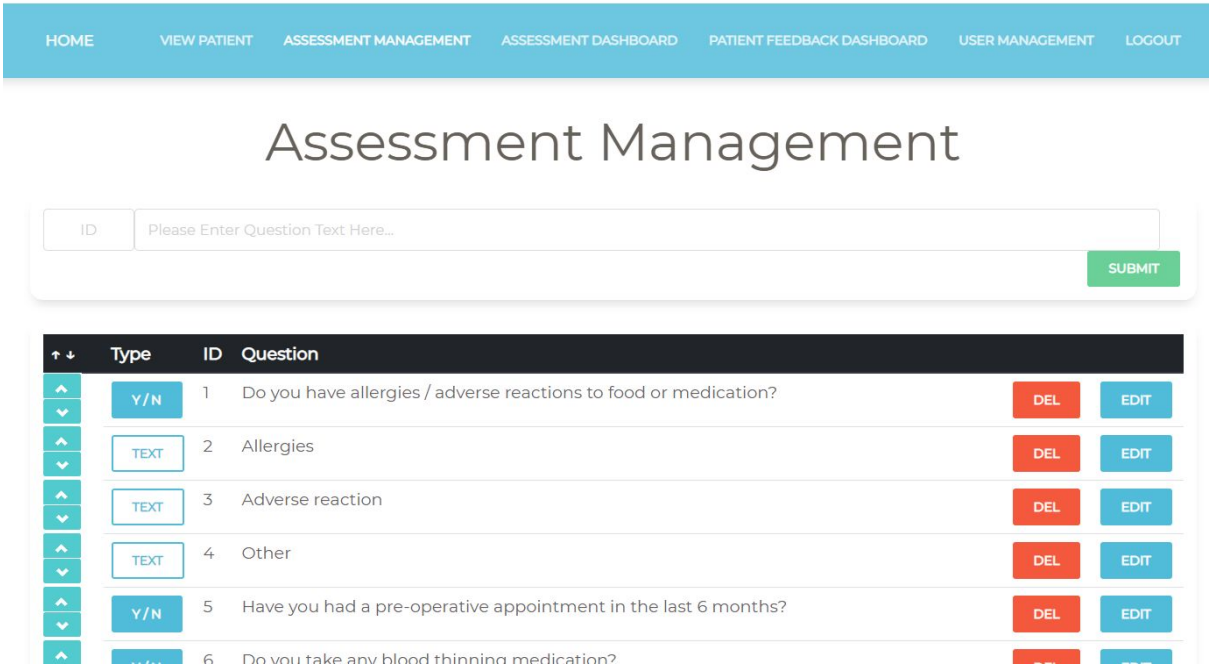

Figure A.1.2 Assess Management Screen (Web)

1. **Question management:**
    a. **New question / No ID** - Enter question text with no ID and press Submit - a new ID will be automatically generated
    b. **New question / New ID** - Enter question text with a new (high) ID and press Submit - a question with that ID will be generated
    c. **Modify question / Old ID** - Enter question text and the corresponding existing ID and press Submit - the existing question will be overridden with the new question text
    d. **Modify question via Edit** - click the Edit button next to any question. The question and ID will appear in the entry fields at the top. Modify text (and optionally, the ID) and press Submit. Note this is equivalent to (c).
2. **Manage existing questions:**
    a. **Change question order** - click the up and down buttons to move the question up and down the question order.
    b. **Toggle question types** - Click the Type button to change questions from Yes/No to Free Text questions.
    c. **Delete question** - Click the Delete button to remove questions from the list

Dashboard Screens

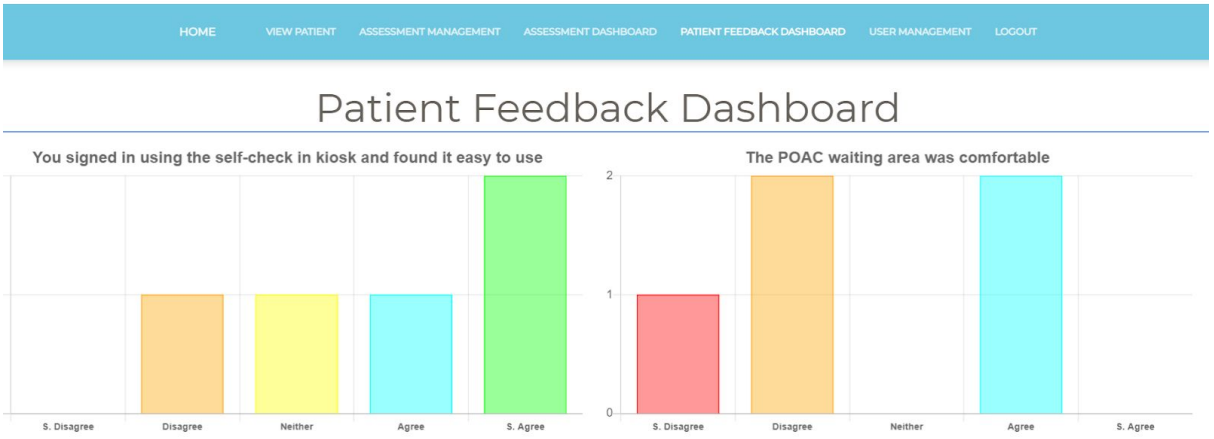

Figure A.1.3 Patient Feedback Dashboard Screen (Web)

1. Select the **Assessment** and **Patient Feedback Dashboards** to see summaries of patient inputs.

User Management Screen

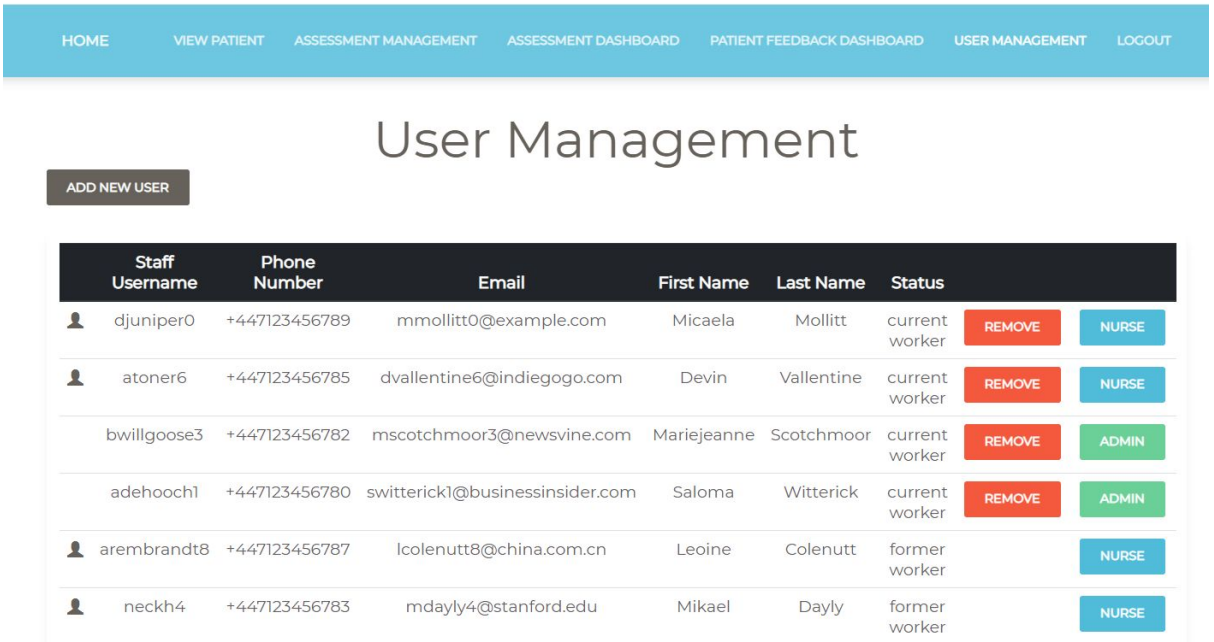

Figure A.1.4 User Management Screen (Web)

1. **Add User** - Press Add New User. An Add User screen will appear. Enter details and press Enter.
2. **Remove User** - press Remove button next to a user. Status will change from current work to former worker and that user will no longer be able to access the application.

3. **Change Access** - press the Admin toggle to grant/remove administrator rights.

Appendix A.2 - Mobile Application

Assessment Screen

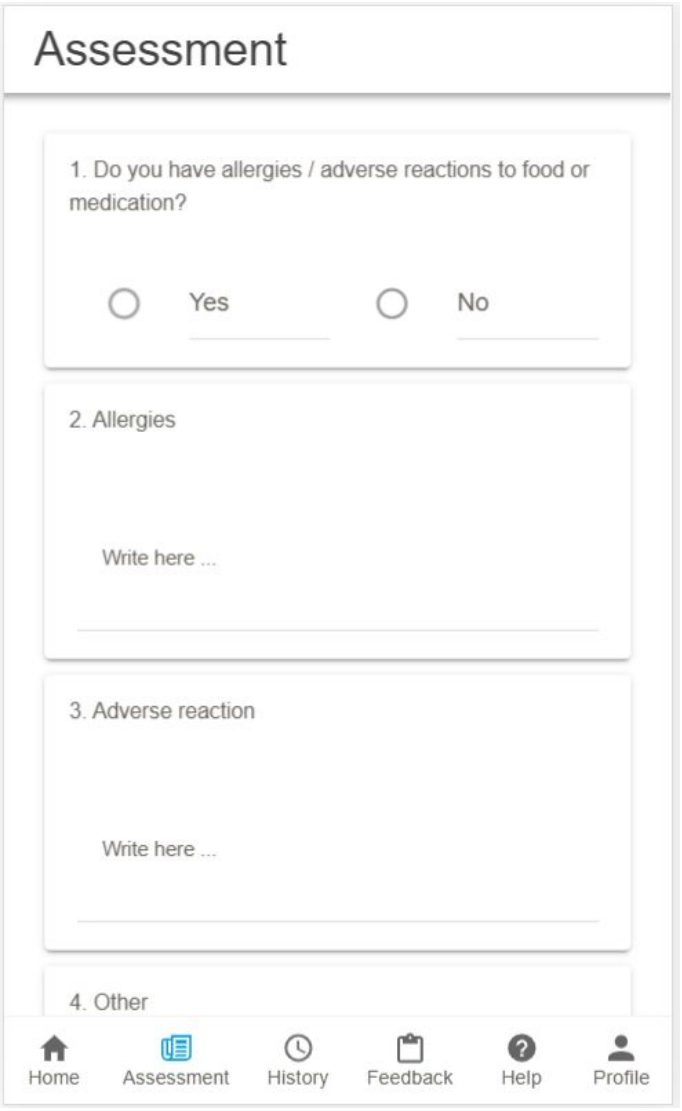

Figure A.2.1 Assessment Screen (Mobile)

1. Select Yes/No or enter Text where appropriate.
2. Press Submit to upload responses

History Screen

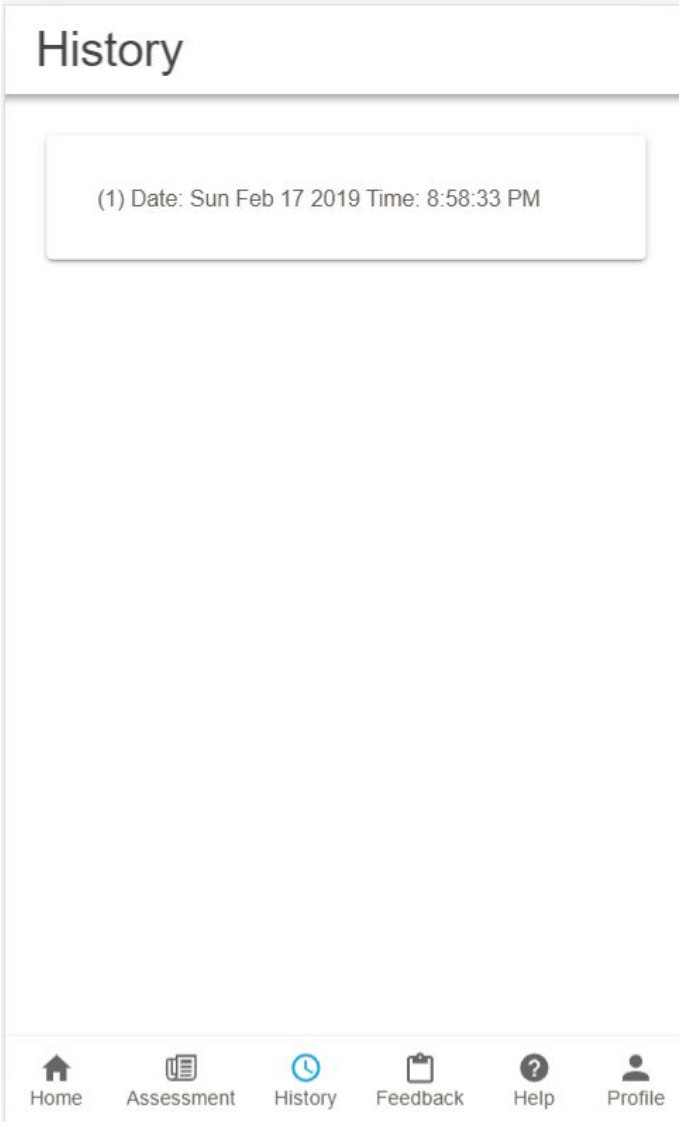

Figure A.2.2 History Screen (Mobile)

1. Select prior assessments (if any) to view previously submitted results.

Feedback Screen

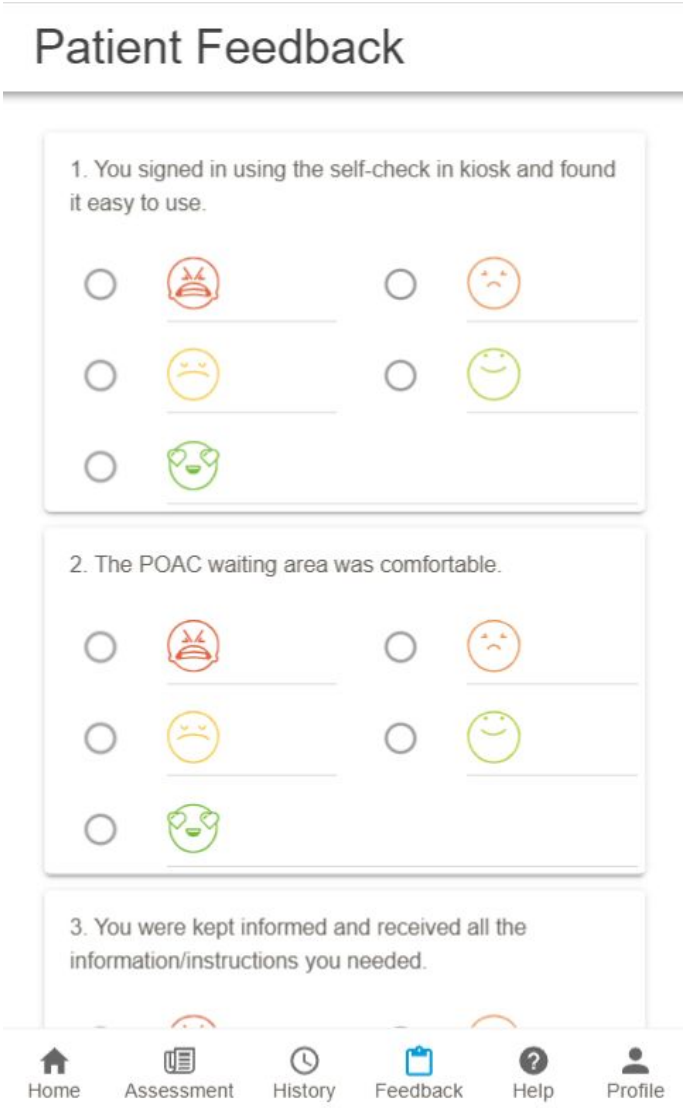

Figure A.2.3 Patient Feedback Screen (Mobile)

1. Select a face icon representing strongly disagree to strongly agree for each of the five feedback questions.
2. Press **Submit** to upload responses.

Help Screen

1. The **Help screen** is a read-only page showing Frequently Asked Questions and additional contact details.

Profile Screen

1. The **Profile screen** is a read-only page showing the user's details.

Appendix A.3 Deployment

- The application is hosted on a Microsoft Azure Virtual Machine. Microsoft provides an estimate of £13.09 per month in server costs.
- The DNS is poac.uksouth.cloudapp.azure.com. Access with user name **Team-9-POAC** and using the private key file.
- If hosted on this virtual machine or another, the **backend** depends on an XAMPP stack running. Run mysql and Apache services.
- Launch the back-end by navigating to the backend folder and running nodemon app.js.
- Serve the website by running entering **ng serve --host 0.0.0.0 --port 8080** in the terminal or command prompt.
- The **web application** is then publically accessible on that port on a remote machine, e.g. by navigating to poac.uksouth.cloudapp.azure.com:8080.
- If hosting on a different machine, the **api.service.ts** files in each of the web and mobile application, and the **auth.service.ts** in the mobile application's **src > app > services** folders have to be pointed to the new machine, e.g. replace:
    - private loginUrl: string = 'http://poac.uksouth.cloudapp.azure.com:3000/login';

... with ...

    - private loginUrl: string = 'http://[NEW_IP_ADDRESS]:3000/login'
- The **mobile application** must be side-loaded on an Android device.
- Copy or download the .apk file to the device. Select the .apk file. If **Install unknown apps** is not enabled, a prompt will appear to change this setting.

Appendix B - POAC Workflow

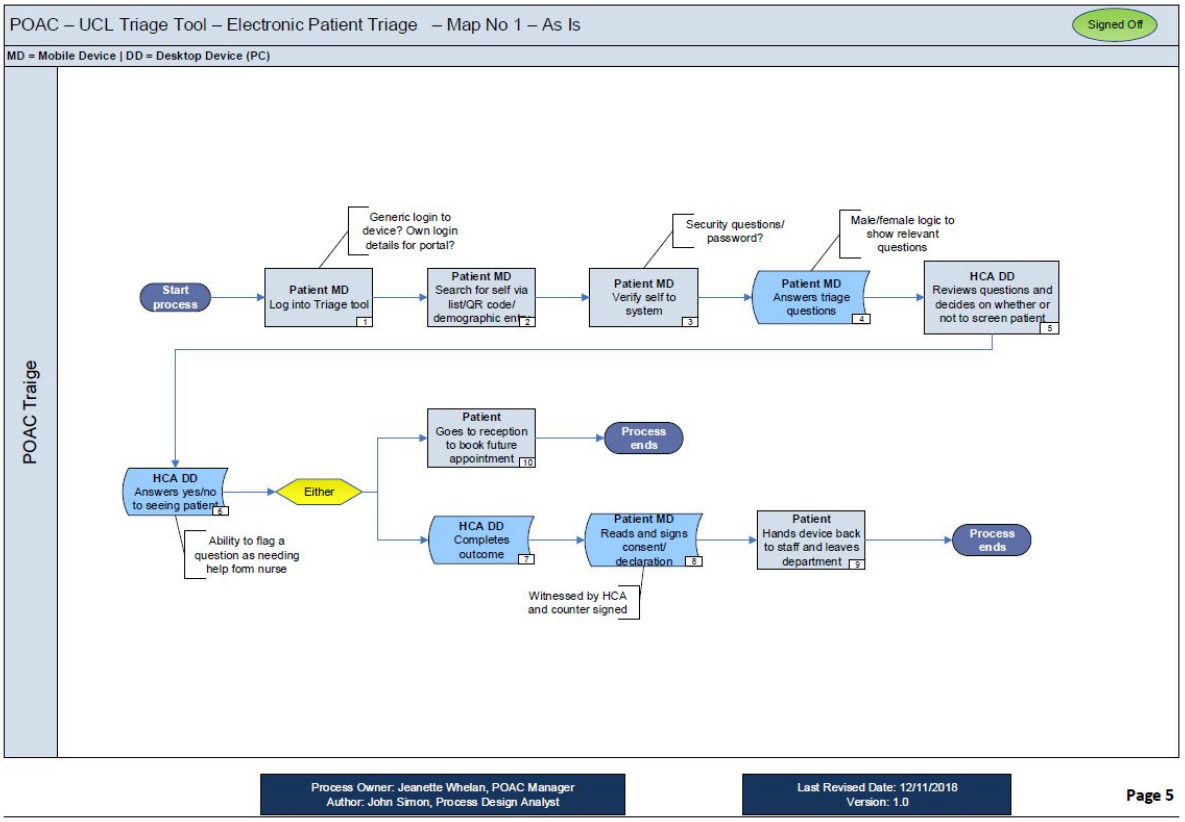

Figure B.1 Pre-Operative Assessment Clinic - Existing Workflow

Appendix C - POAC Paper Form

Allergies / Adverse reactions to food or medication?		
Have you had a pre-operative appointment in the last 6 months	**Yes**	**No**
Please answer the questions below. If you answer Yes to any questions you may require another appointment, which maybe on another day.		
Health Assessment	**Yes**	**No**
Do you take any blood thinning medication?		
Do you have diabetes, which requires medication?		
Do you have high blood pressure OR take medication for high blood pressure?		
Do you suffer from angina or chest pain?		
Do you have a pacemaker?		
Do you have asthma or COPD? (Please tick box and circle which)		
If so, do you take anything other than standard inhalers?		
Do you have sleep apnoea?		
Have you ever had kidney disease or kidney failure?		
Do you have any thyroid issues or take medication for your thyroid?		
Have you ever had a clot in your leg or lungs?		
Do you or a family relative have a history of a bleeding OR clotting disorder?		
Have you ever had a TIA or Stroke?		
Do you have epilepsy?		
Do you have any diseases of the nerves or muscles? (eg MS, Myasthenia Gravis)		
Do you suffer with any immunology disorders?		
Have you or a blood relative ever had a serious problem with an anaesthetic? (not including nausea and vomiting)		
Do you regularly drink more than 30 units of alcohol per week?		
Are you a **SMOKER?**		
Do you take HRT or the Oral Contraceptive Pill?		
Would you struggle to manage 2 flights of steps without stopping?		
During the last 12 months have you lived abroad for more than 3 months?		
During the last 12 months have you stayed in hospital overnight outside the uk?		
During the last 12 month have you stayed in a uk hospital overnight, excluding MPH, Yeovil and Somerset community hospitals?		
Have you ever been in a household or had ward contact of a known case of Carbapenemase Producing Enterobacteriaceae (CPE)?		
Have you ever been colonised or had an infection with CPE bacteria?		
Is there any other information you would like to provide?		

Please list your regular medication here:	
1	**6**
2	**7**
3	**8**
4	**9**
5	**10**

Figure C.1 Pre-Operative Assessment Current Paper Form (page 1 of 2)

Observations					
Wt kg	Ht cm	BMI kg/m2	BP	Pulse bpm	SaO_2 %

Patient Declaration

I believe that to the best of my knowledge, the information I have given on this form is correct. I understand the instructions and information I have been given.

I agree to accept full responsibility for any valuables, property or money I bring into hospital, as the Trust cannot accept liability for loss or damage to property not handed in for safe keeping on admission.

During your future planned surgical admission, in the unfortunate event of a cardiac arrest, we will assume that you would wish to be resuscitated. This can be discussed further on admission when a 'Treatment Escalation Plan' will be recorded in your notes.

If resuscitation is something that you do not wish to consider, or wish to talk through further, please discuss this with your admitting team.

Signed.....Date.....

Outcome		
ASA		
The surgery is Minor/Intermediate/Major		
Observations have been completed and are within normal limits		
The BMI is < 35kg/m2		
BP < 180/110		
The patient has answered "No" to all questions		
No Medications which need stopping / further intervention		
No Bloods taken		
The MRSA swabs have been completed if needed		

Fit to go maxims	On		Nurse Appointment Made			Discussed with registered nurse/anaesthetist		
Information given:			Leaflets given			Other advice		
Fasting								
Pain Control								

TESTS Required			Date of results:								
FBC	HbA1C		Hb	M 135-175 F 120-160		Na	136-145		HbA1c	<69	
U/E	ECG		MCV	82-104		K	3.5-5		Bil	5 -- 17	
G&S	MRSA		Plt	150-450		U	2.4-606		ALP	25-92	
LFT	MSSA		WCC	4--11		Cr	60-120		ALT	0-40	
TFT	CPE		Hct	0.4-0.5		TFT			Alb	35-50	
			Ferritin			LFT			G&S	for X Match	

Comments

Figure C.2 Pre-Operative Assessment Current Paper Form (page 2 of 2)

Appendix D - Use Cases

Use cases are a description of actions or steps defining the interaction between a role (an “Actor”) and a system to achieve a goal (the “Result”).

Our use cases are divided between those for NHS Users (including NHS Administrators) using the web application, and Patients using the mobile application.

ID	Actor	Description
UC001	NHS User	Log-in using staff number and password
UC002	NHS User	Search patients by NHS Number; view assessment history
UC003	NHS User	Perform CRUD on Assessment questions
UC004	NHS User	View summary of all patient assessments
UC005	NHS User	View summary of all patient feedback
UC006	NHS User	Perform CRUD on NHS Users (Admin only)
UC007	NHS User	Logout
UC008	Patient User	Log-in using NHS number and surname
UC009	Patient User	Complete POAC Assessment
UC010	Patient User	View previous assessment history, if any
UC011	Patient User	Complete Feedback if desired
UC012	Patient User	View Frequently Asked Questions (FAQs)
UC013	Patient User	View own profile
UC014	Patient User	Logout

Table D.1 Use Case Summary

ID	UC001
Actor	NHS User
Description	Log-in using staff number and password
Main Flow	*Enter staff number and password*
	If combination is not valid, display error message
	If combination is valid, log-in and display confirmation
	Show home page
Result	*Logged in and token created*

ID	UC002
Actor	NHS User
Description	Search patients by NHS Number; view assessment history
Main Flow	*Enter NHS number and surname*
	If combination is not valid, display error message
	If combination is valid, list previous patient assessments
	Expand assessmenet details on selection
Result	*Read-Only history of assessment details*

ID	UC003
Actor	NHS User
Description	Perform CRUD on Assessment questions
Main Flow	*View questions in order*
	Add questions with/without ID
	If ID is already in use, replace question
	Edit existing questions
	Toggle question types from Y/N to Free Text
	Change the order of questions in the assessment
	Delete questions
Result	*Create, Read, Update, Delete Assessment questions*

ID	UC004
Actor	NHS User
Description	View summary of all patient assessments
Main Flow	*View pie charts of Yes/No answers*
	Each Yes/No response has its own pie chart
Result	*Read-Only overview of patient assessments*

ID	UC005
Actor	NHS User
Description	View summary of all patient feedback
Main Flow	*View bar charts of patient feedback*
	Each feedback question has its own bar chart
Result	*Read-Only overview of patient assessments*

ID	UC006
Actor	NHS User
Description	Perform CRUD on NHS Users (Admin only)
Main Flow	*View list of NHS Users*
	Select NHS User and modify or delete details
Result	*Create, Read, Update, Delete NHS Users*

ID	UC007
Actor	NHS User
Description	Logout
Main Flow	*Log out from the web application*
Result	*Logged out and token cleared*

ID	UC008
Actor	Patient User
Description	Log-in using NHS number and surname
Main Flow	*Enter NHS number and surname*
	If combination is not valid, display error message
	If combination is valid, log-in and display confirmation
	Show home page
Result	*Logged in and token created*

ID	UC009
Actor	Patient User
Description	Complete POAC Assessment
Main Flow	*Enter responses to POAC questions*
	For Yes/No, select answer
	For Free Text, enter text
	If responses are missed, display error
	Submit responses
Result	*POAC responses uploaded to database*

ID	UC010
Actor	Patient User
Description	View previous assessment history, if any
Main Flow	*View list of previous assessments if any*
	Select previous assessment
	Show previous assessment responses
Result	*Read-Only view of previous responses*

ID	UC011
Actor	Patient User
Description	Complete Feedback if desired
Main Flow	*Select responses to feedback questions*
	Submit responses
Result	*Feedback responses uploaded to database*

ID	UC012
Actor	Patient User
Description	View Frequently Asked Questions (FAQs)
Main Flow	*View page of Frequently Asked Questions*
	View contact details for further information
Result	*Read-Only view of help information*

ID	UC013
Actor	Patient User
Description	View own profile
Main Flow	*View details of own profile*
Result	*Read-Only view of own profile*

ID	UC014
Actor	Patient User
Description	Logout
Main Flow	*Log out from the mobile application*
Result	*Logged out and token cleared*

Table D.2 Use Case Details