**Sprint review**

Contents

[Sprint review #1: 06/03/19 2](#_Toc8764440)

[Sprint review #2 19/03/19 3](#_Toc8764441)

[Code review: 3](#_Toc8764442)

[Sprint review #3 1/04/2019 5](#_Toc8764443)

[Code review: 5](#_Toc8764444)

[Sprint review #4 15/04/2019 6](#_Toc8764445)

[Code Review 7](#_Toc8764446)

[Sprint review #5 29/04/2019 8](#_Toc8764447)

[Code Review 9](#_Toc8764448)

[Sprint review #6 13/05/2019 10](#_Toc8764449)

[Code Review 10](#_Toc8764450)

## Sprint review #1: 06/03/19

After the first two-week sprint, a review was carried out to go over what the group had done within that time. During this review, each member of the group said what they were able and not able to do, which then made it possible to continue and for the group to be able to plan the following sprint.

During the first two-week sprint, the group were able to complete the diagrams, wireframes, a couple HTML pages as well as paper prototypes. As well as this, the group have generated the user requirements, functional and non-functional, as well as user stories and normalization and acceptance criteria.

* Wireframes – (Doctor): - Callum designed the wireframes for the look of the doctor side of the application using draw.io.
* Diagrams – Each member of the group were allocated diagrams to do based on the scenarios of the doctors / patients. The diagrams were equally distributed among the group
* Normalization – Laura: This has been stated that it is subject to change throughout the development of the project and will probably be reviewed and edited come the end of the development process.
* HTML pages – Dan. (Subject to change depending on the requirements from the client.)

During a meeting with the group’s client, the requirements have slightly changed from the original ones stated, therefore the documented requirements list has been altered to meet the current requirments.

The group were unable to create the database due to the alteration in sever space and database use, which has changed from an SQL to Microsoft SQL database, therefore it is being created through visual studio rather than SQL itself. – This is something which is now sorted, and the group are still on track for the current sprint.

Due to initially not having a database or server space, the group were unable to be able to connect the database to back end. This is something that the group will aim to do in the following sprint.

During the first sprint, the group also had a meeting with the client discussing the MVP and therefore have generated a requirements list to finalize and confirm the MVP – the MVP will be updated during each sprint, to be able to gauge what the group are able to complete during each sprint, as well as being able to see how the project is advancing throughout the development stage.

The group have decided that by the end of the next sprint, the group would like to advance more on the database, have all the HTML pages working and created (design does not have to be good at the minute). As well as having some of the C# back-end working. Due to all our requirements being finalised with the client, the sprint goal for the next two weeks is hopefully achievable.

The group as well as the client, have agreed on the fact that the functionality of the project comes before the design and how it looks – due to time, the group will focus on the HCI concepts after the functionality has been implemented; despite this, the group have considered user interface this whilst creating HTML pages and wireframes.

The group are primarily focusing on the doctor application for the time being, and once functionality is working, the group are going to move on to create the Patient application.

## Sprint review #2 19/03/19

After the second two-week sprint, the group are pleased with the outcomes of this sprint. The group have been able to create a good section of the database, along with several other HTML pages for the doctor application. However, there has been no connection to the database via C#, or back-end to make the HTML pages operate as they should.

**Tasks complete:**

* Generating HTML pages
* Start of login feature

The group are hoping to further the project by working at a quicker pace within the next sprint, to get the project to an outcome where we can show the client what we have got so far, and be able to gain more useful feedback for how we can develop further (in the following sprint).

The group have decided that by the end of the next sprint, we would ideally like to have the HTML pages linked to all other html pages, and are going to work on the back-end, so the C# can make the HTML pages work functionally as they should. The group are also going to look at the C# connection to the database and try to resolve this issue and have the database reading in and out.

The group have established some tasks for the next sprint, which will further the development of the project at a quicker pace.

***Tasks for next sprint can be found in the project backlog, or the sprint plan document.***

### Code review:

***Code reviews were taken place by all three group members – each member reviewed which they had not written.***

Log in - The log in page was an important aspect of the project, in this sprint the Login feature was partially implemented, being that it only works for doctors and not patients.

Code wise the log-in feature was based off a simple if-else statement, if the text in the input boxes did not match two values in the doctor table then access would not be granted to the site and the user is notified of this with an authentication error message, else if the two input box values match then the user is granted access and sessions based off the user info from the database are created, the session data can be called across the pages for use in multiple functions. The first of which is the page authentication, if the user does not possess session info created via a login then they are automatically redirected to the login page.

Code review Feedback: There are a few issues that need to be addressed, namely that incorrect information within the login page breaks the application as opposed to simply notifying the user of the incorrect information. There is also the lack of a login feature for the patients meaning they cannot access their side of the application.

When reviewing the code for the login, the quality of it could be better – the group have worked to improve the quality of the code and be able to fix some bugs which the group came across. The group also added some comments within the code, so it was easier to read and understand for someone else.

Doctor home page and database: The Doctor homepage was an important part of the Doctor’s part of the application. Daniel was able to create the html page for this to visualize how the final page will look. This included important functionality, such as the ability to search for patient conditions, and viewing those results on a table. These functions currently don’t work, but show how it could function in the final product.

Feedback: The database design was received well by the team, as the relationships between tables worked logically. Upon testing, the group could also insert data into these tables, including related tables, thereby showing that the database works as intended, and is fit for use within the application.

During this point in the sprint, the HTML pages were developed for the doctor application and – which included Dashboard and the log in page.

After these were created, the HTML code was reviewed by the group. This included looking at the quality of the code and making sure that it was working and readable. The html pages created gave a good foundation for the group to continue working on, as well as being able to have a good user interface for the end-user.

## Sprint review #3 1/04/2019

During this sprint, the group’s goals were to be able to make the Doctor Web application more functional.

This sprint has been successful in the fact that the group were able to make the database connect to the back end of the project, and the group have begun to start development on more of the functionality of the project. To do this, we have continued to work on our individual tasks which we assigned at the start of the sprint.

**Tasks complete within sprint 3:**

* Conditions were added to the Doctor Web application, enabling the doctor to be able to search certain parameters in a search bar. – Dan
* The development of the patient application was commenced by creating several wireframes, as well as HTML pages. – Laura
* Adding on click links to the HTML pages to have more of an idea what the application will be like once it is fully functional – Laura
* The HTML for the log-in page, the goal was too match it as closely as possible to the pre-established wireframe - Callum

The group were also able to make the HTML pages link, so there is more of an idea of what the application will be like when it’s working.

Due to the sprint being successful, and the group achieving what was planned for sprint 3, the group are going to continue to develop at this pace for sprint 4.

**The tasks allocated for sprint four, can be found in the project backlog on GitHub or the sprint plan document.**

### Code review:

***Code reviews were taken place by all three group members – each member reviewed which they had not written. As well as stating what the group member did, the group primarily aim to optimize the quality of the code for the overall application.***

Laura – working on the patient application and the html pages - The group was satisfied with the code quality and was at and was confident that the html pages would provide a good foundation for the rest of the project – in terms of functionality.

Callum - Login: The error in which false information permits access but breaks the application was removed and in its place there is now a fully working error message, the password text box now hides the information being inputted as to prevent people seeing other user’s passwords. Patients can now access the site with their information in exactly the same manner as doctors via inputting their email and password, however there is now also another method of access, that being the login code.

Feedback: the login feature allows all the users to access the site in every way provided, it would be beneficial security wise however in the information (the username and passwords) were encrypted (potentially using a simple hashing algorithm MD5), it would also benefit if salting was implemented as well.

Doctor Homepage – Daniel created the main homepage of the Doctor application, having based it off of the wireframes created during the design stage. This was to ensure that the interface for the page was user appropriate. Daniel added a search functionality to search for patient’s dependent on their conditions. This search feature worked successfully, and the user could search from a variety of different criteria’s: Patient ID, First Name, Surname, and Condition.

Feedback: The search feature successfully allows the user to successfully search through patient conditions. Feedback from this was that it was incredibly useful for finding specific requirements the doctor wanted to look for, as the search functionality made use of a variety of different parameters. A complaint made however, was that the view did not reset if the search bar contained no characters. This is a change Daniel will attempt to implement in a future sprint.

## Sprint review #4 15/04/2019

This sprint has also been somewhat successful – as in the group have achieved mostly what was specified to do.

The group have continued to gain functionality in the Doctor Web application, and the Patient HTML pages have been fully made – so the functionality can begin on this web application.

**Tasks compete within the forth sprint:**

* All of the HTML pages have been made for the patient application, so the functionality can begin. – Laura
* The log in for the doctor and patient has been created, and is working.
* Questions - Daniel
* Dashboard - Daniel
* Messaging - Callum
* Questions
* Notifications
* Account

Due to their being limited time left on the development process, the group are going to develop at a faster pace during the next sprint. To do this, the group have assigned specific tasks for each member to get on with during the **fifth sprint, and if those tasks are complete, then we develop the project further.**

### Code Review

***Code reviews were taken place by all three group members – each member reviewed which they had not written. As well as stating what the group member did, the group primarily aim to optimize the quality of the code for the overall application.***

Callum:

An issue with implementing the messaging service was that there was not a way of discerning which doctors had which patients, to overcome this Callum added the DoctorPatientMatches table and populated it with the foreign keys from the doctor table and the patient table, this implemented the one to many relationship between doctors and patients, a good use of database devlopment. Using the doctor patient matches table Callum created a table on the messaging page that displays information only concerning the current user, this is done by checking the session info and comparing it to the database information, if the Doctor/Patient ID matches an ID within the database (which it will as an ID session is required for page access and is a fundamental part of a users account) then the corresponding information is pulled from the doctor patient matches table and inputted into the table on the messaging page, this would mean that when it comes to adding a messaging service the user’s can only choose who to message based off of the users they actually have a relationship with within the table, this is a good bit of code as it prevents patients messaging the wrong doctor etc.

Feedback: message page works as intended, SQL connections between the doctor and patient using the doctor patient matches table is an effective middle ground and also serves to show the patients and doctors associated with one another, meaning that users cannot see information that doesn’t concern them. Members of the group have commented that the code produced needs working on – therefore is not ready to be reviewed for improvement.

Daniel: Daniel has made a decent start in the survey creation part of the application. At the moment, Daniel has added functionality for creating the survey itself. The doctor can pick what patient he/she wants to create the survey for, they can define how many questions they would like to ask, and how frequently they want the survey to be sent out (Daily, Hourly, Once, Every certain amount of days). For the questions creation page, Daniel has been able to add means to define the question name, as well as selecting a question type from a dropdown list (Numeric answers, Video, Audio, Free Text, Multiple Choice and Condition). Amount of Questions created has also been recorded by saving them into a session. For multiple choice style questions, the doctor can add 4 answers for the patient to select. Once they have done that, they are returned to the create questions page. Finally, Daniel has implemented a means to upload videos to the SQL database. The data that is stored on there include the name of the video, and its file path in the project directory. The video is also moved to the video folder on the directory upon upload. At this current time, there is no means to review the questions and answers created for the survey, but it is something Daniel intends to add.

Feedback: The doctor can successfully create surveys and questions, as this data can be successfully viewed by using select SQL queries on the database. Improvements could be made in terms of readability for certain attribute titles in both the survey and question creation page, as the scaffolding technique used to create these pages crudely ported over the column names of the tables. These need to be modified to be more readable. For multiple choice answers, it would also be good to be able to view how many answers have been currently created for the question.

Laura – Implementing the My Profile page for the patient application. During this, Laura has created the cshtml page as well as creating the controller for this page – this page is yet to be able to read information in and out of the database.

Feedback: Code not quite yet ready to be reviewed as it is not working at all at this stage.

## Sprint review #5 29/04/2019

The group met to discuss the outcome of this sprint, and what tasks have been achieved and what has not.

* Dan: Has been working on the questions, and the surveys - nearly finished this task, will be finished within the next couple of days.
* Callum: Been working on the messaging page between the patient and the doctor – finished, adding last touches which will also be finished within the next couple of days.
* Laura: Been working on the patient ‘my profile’ page and allowing the patient to be able to edit their profile and then save the changes: Will also be finished within the next couple of days.
* Jay: Was assigned the alerts page, however there has been no contact to if Jay has done this, or any contact at all with any members of the group.

To achieve the outcome which the group desire, Dan and Callum are continuing with their tasks and finishing them up immanently. Laura is finishing the profile page, and then taking on Jay’s task of the Alerts page for the doctor, and then creating the histogram.

The group are hoping to finish these tasks as soon as possible, however the outcome of this might not be possible due to time constraints as well as all members of the group not contributing.

### Code Review

Callum: In this sprint Callum added the messaging service, it works by taking the user’s first name, piping it into a session called “name” and then calling the name of the page, in this way the two differing sessions DoctorName and PatientName are converged into one, which was necessary for the messaging service as it belongs to both aspects of the application. It was implemented using signalR which does a majority of the heavy lifting for the service. A clients message is sent to the server with their name and the message itself and the information is returned to the corresponding clients.

Feedback on messaging service: Whilst it is good that messages can be sent and received by multiple user’s, in the context of a Doctor/Patient application, it is not suitable that the message service broadcasts globally to all users, meaning that confidentiality is out of the window.

Daniel: In this sprint, Daniel managed to finalise the create surveys side of the application. Numeric answer style questions could now be created for a patient’s survey, and the doctor user can now review the survey details and the questions of that survey. This was achieved by using a foreach loop to go through the survey data from the data model, and also allowed access to its collection of questions. Specific survey details were able to be imported in by accessing its specific session. Rule parameters for numeric answer style questions can now be created as well, helping to define whether the answer submitted by the patient is in a healthy or unhealthy boundary. Daniel was not able to get a means for the answers to those questions displayed, however, as the means to display the collection answers was slightly different, as the page itself relied on the survey data model, which only contained a collection of questions, not question answers.

Feedback: Feedback for the review survey page was positive. Being able to review survey details and questions was seen as good for the user before submitting. However, not being able to see the answers created for those questions was seen as a detriment to that page, and something that should be added. Foreach loop for answers needed to be added. Feedback for rule parameters was seen as positive, as the means to add these rules were simple, as you only had to type in a number to give a defined parameter.

Laura: My Profile page - As mentioned above – nearly finished.

Feedback: Good quality of code produced. No issues.

## Sprint review #6 13/05/2019

The group have met up to discuss the outcome of this spring and if all tasks have been achieved.

Callum:

* During the final sprint Callum was working on changing the SignalR messaging service. This code was fixed before the deadline, when the page is loaded, the SignalR start connection method is run, within this method the session “message match” was taken and parsed to the server where the variable becomes the name of the SignalR hosting group, essentially meaning that the private message function between single users (doctors and patients) is a group chat with only two members.
* This code be described as a bit botchy however to the user it is indiscernible from a normal private messaging service, it also meets its purpose of preventing other users from reading messages not meant for them.

Dan:

* : In the final sprint Daniel added the patient functionality for the surveys.
* The patient can now login and access the surveys they have yet to complete.
* They can also select that survey, and those questions can now be displayed, as well as a means to answer them (In this circumstance, a textbox).
* Unfortunately, Daniel was not able to get these answers saved to the database, so he instead created a submit successful page to redirect to in order to simulate a successful completion of the survey.

Laura:

* Laura has finished the My profile page on the web application
* Laura has created a html alerts page it has been inserted into the application by another member
* Laura has created a cshtml responses page as well as controller which has also been put into the application

### Code Review

Callum- Feedback on quality of code produced: Testing the messaging service amongst the group, having logged in using different users’ the group tested whether or not users could see messages that were not intended which we could not, we also tested the sending and receiving of messages for the intended users which worked as intended. The code was at good quality and the message part of the service was fully working.

Daniel – Feedback on the code produced: To test that patients could view the surveys they needed to complete, the group created a test survey on the doctor’s side of the application for 2 different users. This was to test that users could not view each other’s surveys. Once surveys were created for both users, we would then log in as both users in separate instances, to see if that we could view their respected surveys in the survey preview page. This worked as intended, with the patients only being able to view their respective surveys. Next, we tested if the survey would only render its specific questions by selecting the Start survey button on the survey preview page. This worked as intended, as it would use the unique id of that survey to generate its relevant questions. Finally, to test that the submit redirect to the submit successful page worked as intended, we typed in answers to the questions and then clicked submit. The redirect occurred as expected, and we were able to view the submit successful page.

Laura – Feedback on the code produced. My profile page was at good quality and functional as it should be. The alerts and responses page are fine in terms of html code – but they do not actually work.