Assignment 2 – Design Document

Team 6

Members
James Cassidy
Ciaran Gernon
Lewis Cooley
Nathan Donaghy

Introduction (5%)

This document aims to direct us in the right direction regarding where we take our solution and how it may look at the end of initial development. As with any piece of modern software, requirements can change during development. Our team aims to stick relatively close to what we set out in this document during development.

In the design document, we can document user stories, allowing us to put the team in the mind frame of potential users, allowing us to optimise specific areas of the software to cater to them. Alongside user stories, we can document specific frameworks and technologies that will be used in the proposed solution.

The software we are developing will be used to combat mental health problems for users undergoing treatment with their respective therapists. The software will be developed with the idea that it tracks users' emotions and feelings throughout the time between sessions with their therapist. The data collected will be heart rate via ECG sensors, cortisol levels via sensors created by researchers at UCLA [1], and the user's feelings during a time of stress or happiness. This data will be stored relational database and can be viewed by the user's therapist. This will allow the therapist to have a deeper understanding of their patient's needs and allow them to pinpoint the patterns in which they feel the most stressed.

Figure 1: Uses of Smart Wearable Medical Devices in Everyday Practice [2]

Extensive research was carried out by each member of the team to look for areas in everyday life that may be improved using IoT and digital transformation. In the end, we decided to look further into wearable health monitoring. Solutions for this already exist in various forms from Fitbit, Apple, and Google. Our solution takes this a step further. Various health data such as cortisol, stress, ECG, and sleep tracking are collected through our proposed sensor. This data combined will be able to paint a better picture for 3rd parties such as therapists and pharmacists. These types of users are given the supervisor role within our solution. With better access to their clients' statistics, they can better prescribe assistance for specific ailments a user may be experiencing.

User Stories/Scenarios (10%)

David has recently started a new job in a big law firm. However recently he has been feeling under pressure in his work environment and cannot manage his workload and time. Recently he has been attending therapy sessions in order to combat this increase in stress. His therapist recommended he keep track of when and where he experiences an increase in stress.

Matthew is a Ph.D. researcher. He has been experiencing stress, depression, and anxiety recently. His therapist asks him to download a new app on his smartwatch. The app will allow his therapist to monitor his mood and general health throughout each day and use this data to get a fuller insight into the patient. Matthew has been struggling to explain his symptoms and what his triggers are. So, his therapist suggested when he feels any spike in any mood (happiness, sadness, angriness, worried, nervousness, etc.) to select the respective emoji on his wearable. This will not only improve Matthew's self-awareness of his mental health but will also provide his therapist with talking points and data to discuss in their sessions.

Joe is an elderly war veteran who struggles with PTSD. He is unable to relax and partake in the leisure activities he used to love. His family is becoming increasingly concerned for his health. His therapist has requested he download the smartwatch app so they can pinpoint where and when (in what situations) he is experiencing the most trauma and stress so that these feelings could be mitigated. His therapist has booked Joe in for another appointment in 2 weeks, where they can discuss the results.

Cara has been struggling with body dysmorphia since she was in her early teens. She used to weigh her food and calories daily. Recently she has been seeing a doctor for bi-weekly appointments in order to keep track of her health. Her doctor recommended she use a smartwatch that connects to her smart fridge so she can keep track of items and groceries that she has or needs to purchase.

Scenarios (30%)

MVP Scenario

Matthew

Matthew has been working on a new project for his research. Over the past few months, he has been developing a solution to combat traffic congestion on the motorway. However, after the development of the product, he notices a fundamental flaw in the system that causes it to behave in an undesired manner. Matthew has a history of thinking he has been a failure despite being a Ph.D. researcher. The smartwatch that Matthew wears allows him to input his various emotions throughout the day. Matthew selects emojis specific to how he is feeling. He selects the emoji related to feeling down and depressed. He inputs a short summary of the reason for feeling down and depressed. The wearable device stores the information relating to Matthew's input such as the time, date, and location. The location of where he felt depressed is important so that Matthew's Therapist, Sebastian, can keep track of patterns and trends in Matthew's emotions. Matthew spends a couple of hours working on finding a solution to his problem. Luckily, he solves the error, and the system works as anticipated. Matthew records his feelings, by selecting the emoji relating to happiness and relief. The wearable records this data and stores it in the database.

During Matthew's therapy session, Sebastian requests access to Matthew's data via his version of the application. Sebastian can view this data and discuss it with Matthew during the session. Sebastian sees that Matthew was depressed due to his project a couple days prior; however, he also sees that Matthew was happy about his work a few hours after the previous event. Sebastian and Matthew discuss the events that caused a change in emotions that related to the same situation. Sebastian asks Matthew what he did and how he overcame the problem. Matthew describes that he sat down and focused completely on the problem.

Sebastian knows that Matthew struggles with feeling like a failure. Sebastian uses his version of the application to record some feedback to output to Matthew when he feels down about his work. He inputs into the system a reminder that Matthew has overcome problems like this before and that he can produce a good standard of work. The application stores this information in the system's database.

Some weeks later Matthew, starts a new project in which he feels out of his depth. Matthew inputs that he feels inadequate into his smart watch wearable. The software's machine learning algorithm recognises that Matthew is feeling emotions like what he was feeling a couple of weeks prior. The application then outputs Sebastian's feedback and reminds Matthew that he has been in this situation before and that he can produce quality work.

Aspirational Scenarios

David

David is about to go for lunch however, his head of department has asked him to call in for a meeting after his lunch break to discuss the current project David is working on. His supervisor gave him no other information on the topics that would be discussed in this meeting. Recently, David has been very concerned about the quality of his work in his new job so after hearing about an unscheduled meeting he is unprepared for, he becomes very worried. David's smartwatch sensors can detect this raise in stress levels via sensors that detect heart rate and cortisol levels through sweat. The smartwatch sends out a subtle notification of this rise in stress via a short but identifiable vibration in the smartwatch. The hidden nature of this notification is important so that other people (e.g., David's head of department) do not notice that David has become more stressed. The smartwatch application then allows David to input a short summary of what has happened in this situation to increase his stress. This input could be taken in via a voice recording or a text-based input. The wearable then stores this input along with the date, time, and location of this stressful situation. The wearable then stores this information in a database so that it can be used in David's next therapy session.

David attends a well-being session three days later with his therapist Andrew. Andrew has a slightly different version of the application. This application will be an "administrator" app. In this version of the application, Andrew must request permission from David to view the data that has been logged via David's wearable. David can deny or accept this request. If David accepts, Andrew can see that David had a significantly increased stress level, three days prior at his workplace. Andrew asks David about what happened in the situation that caused this. Andrew and David discuss this situation and work on a few solutions and techniques to incorporate, the next time something like this happens. After the session, Andrew makes a few short articles containing the solutions and information they discussed during the session. Andrew then inputs the solutions into his version of the application, and they are stored in the database.

A few weeks later David experiences a similar situation in work. His head of department asks to discuss with him a recent development in his current project. David's stress levels increase, and the wearable detects this. David inputs a short description of what has occurred. The wearables software enables machine learning and is able to take this input, location and, time and determine that this event is similar to the previous scenario. The wearable then outputs to David, the solutions that Andrew had stored in the database. David reads these solutions and is able to control his stress levels before his meeting.

Joe

Joe recently attended a celebration for his grandson's graduation. There were many people at the event which made Joe feel claustrophobic. Joe experiences these anxious feelings as it reminds him of traumatic times during the war. Joe inputs this into his smartwatch wearable and the data is sent to the database. Joe's grandson had purchased fireworks for the occasion. Unbeknownst to Joe, the fireworks were set off during the celebrations. This was deeply unsettling to Joe as it reminded him of gunshots and explosions during his conscription. This raised Joe's heart rate and cortisol levels which were then detected by the wearable.

Carl is Joe's therapist. Previous therapy sessions between Joe and Carl concluded with options on how Joe can deal with events that induce PTSD. Carl suggested that Joe excuse himself from situations and find a comfortable location in order to calm himself.

When the fireworks were ignited, Joe went to the bathroom and locked the door. However, he could still hear the fireworks during this. This triggered Joe to experience further increased stress levels and induced a panic attack. The wearable smartwatch detected this and notified Joe. However, because of Joe's panic attack, he is unable to input a response or reason for this. Due to the lack of response from Joe, the wearable enables the voice recorder. Using machine learning the wearable detects that Joe is having a panic attack by listening out for heavy breathing, audible upset such as crying, or repeating distressing vocabulary. The wearable can then send a message to a pre-defined emergency contact alerting them to the fact that Joe is having a panic attack.

Joe's emergency contact is his wife, Sharon. Sharon is also at her grandson's graduation and receives the alert of Joe's distress and comes to the bathroom to check on him. Sharon then calms Joe down and tells the others to stop the fireworks for the time being whilst Joe is there.

Cara

Cara has been struggling with anorexia since her early teens. Her GP, Daniel, is concerned with her levels of food intake however, he does not want her to be keeping track of her food intake as this is a very common symptom of anorexia.

It is important that people overcoming anorexia do not count the numbers on their food such as calories. Cara's smartwatch is connected to the smart fridge in her home. The smart fridge can monitor the nutritional information of the individual items it contains. This is important as it allows Cara to have no idea of the number of calories she is consuming whilst also making sure she is getting enough nutrients. Whenever Cara, is running low on items that she requires, the smart fridge alerts the smartwatch that then sends her a notification notifying her of this. The smartwatch can also recommend Cara's recipes relating to the items she has in her fridge.

Daniel can use his version of the application to send Cara nutritionally dense foods via the smartwatch. Cara will then be able to make recipes that are high-quality as well as appetizing.

The smart fridge sends daily alerts to the smartwatch application of the contents that have been consumed. This data will then be stored in the application database. When Daniel requests access to this information during their next meeting and Cara accepts the request, Daniel can see the quality of food and how much food has eaten since the last time they saw each other.

Logical View (15%)

Package diagrams are structural diagrams used to show the organization and arrangement of various model elements in the form of packages.

The N-tier architecture pattern is a well-established software application architecture that organizes applications into separate tiers. The 3-tier pattern consists of: the presentation tier, or user interface (UI); the business tier, where data is processed; and the data tier, where the application data is stored and managed [3].

The microservices architecture is another software application architecture that decomposes an application into small independent services that communicate over well-defined APIs. Since each service can be developed and maintained by autonomous teams, it is the most scalable method for software development. [4]

This proposed solution will be a combination of both 3-tier and microservices architecture. The implementation for our proposed solution is detailed as follows:

- The UI/Presentation is the top tier and front end. This is the smartwatch, smartphone, and webpage application, where the user views their health information. The content for the prototype can be developed using HTML and JavaScript.
- The middle tier houses the business logic used to process the users' inputs/actions, implemented with microservices. This tier queries the database in response to an action from the user.
- The Data layer is the bottom tier database server and the backend of the application. It will run on database management software, such as MySQL.

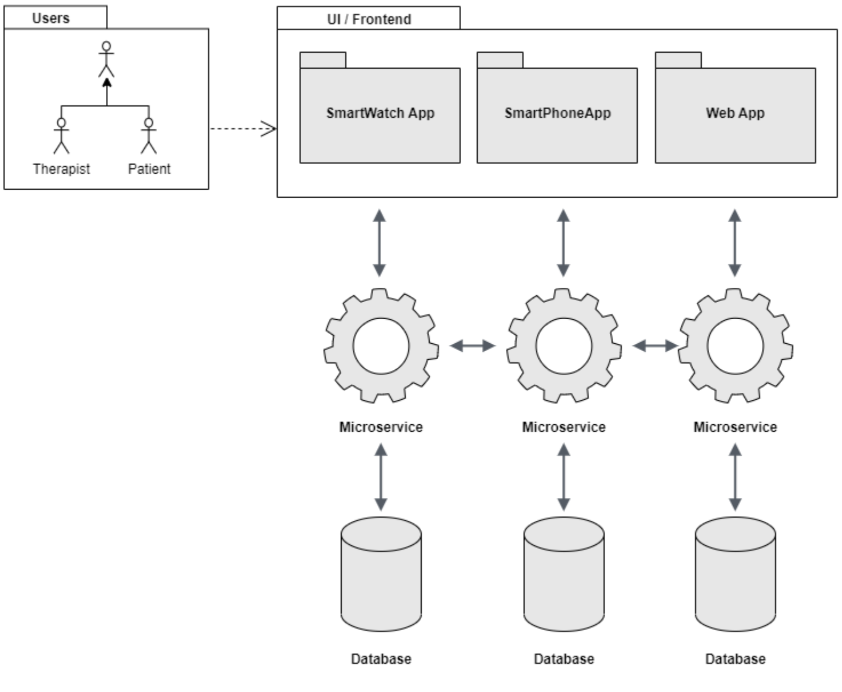

Figure 2: Logical View Package Diagram of Proposed Solution [5]

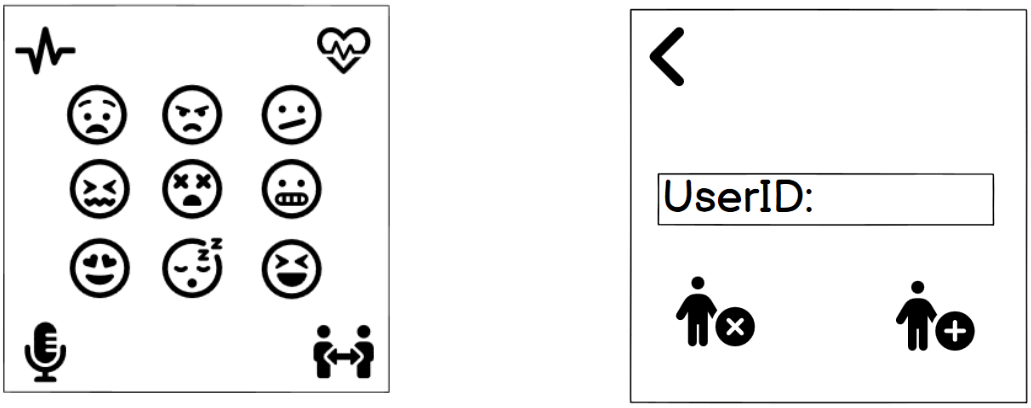

Figure 3: Mock-ups of Patient Smart Device View of Proposed Solution [6]

It is important that the user interface for the patient's application is simple as people suffering from mental illness tend to find it difficult to learn new skills and techniques. The ease of use and learning of the interface will allow the users to be more comfortable with the application. Being more comfortable with the application will encourage the user to use the application more often.

The User interface shows nine emoticons that the user will select depending on how they feel. The cortisol and heart rate detection can be viewed at the top of the interface. A microphone button will allow the user to make a quick speech input on how they are feeling at that current moment. The icon showing two people with an arrow connecting them allows the patient to add their therapist/ accept or deny permission for data viewing.

The second interface is for adding a user. Again, this is a simple design for ease of use. The user can type in their therapist ID, add them or block them. A similar user interface will be used for accepting or denying permission to access of the data.

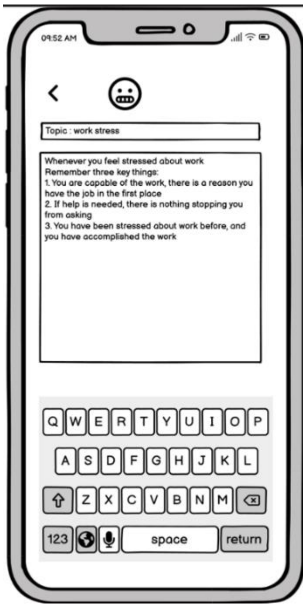

Figure 4 Mock-up Smartphone View of Proposed Solution

This is the user interface for the user to view the therapist's feedback on their phone. The smartwatch will be connected to the user's phone via Bluetooth. The feedback that the therapist provides can be displayed here if the article provided is too long to fit in the smartwatch display.

The therapist will be able to view all their patient details on a website. The response from each patient is brought up for the therapist to read, analyse, and suggest what the patient can do going forward. The patient's textual or voice response to how they're feeling will be displayed for the therapist, along with the indicative emoji(s) selected.

The therapist will be able to take notes on their patients and type up their recommendations for the patient, including supplying them with helpful online resources via links. The therapist will be able to view all the patient details and has quick actions such as: telephoning the patient, emailing the patient, and contacting their GP. The current status of their treatment is available, as well as the data analytics provided by their IoT wearable.

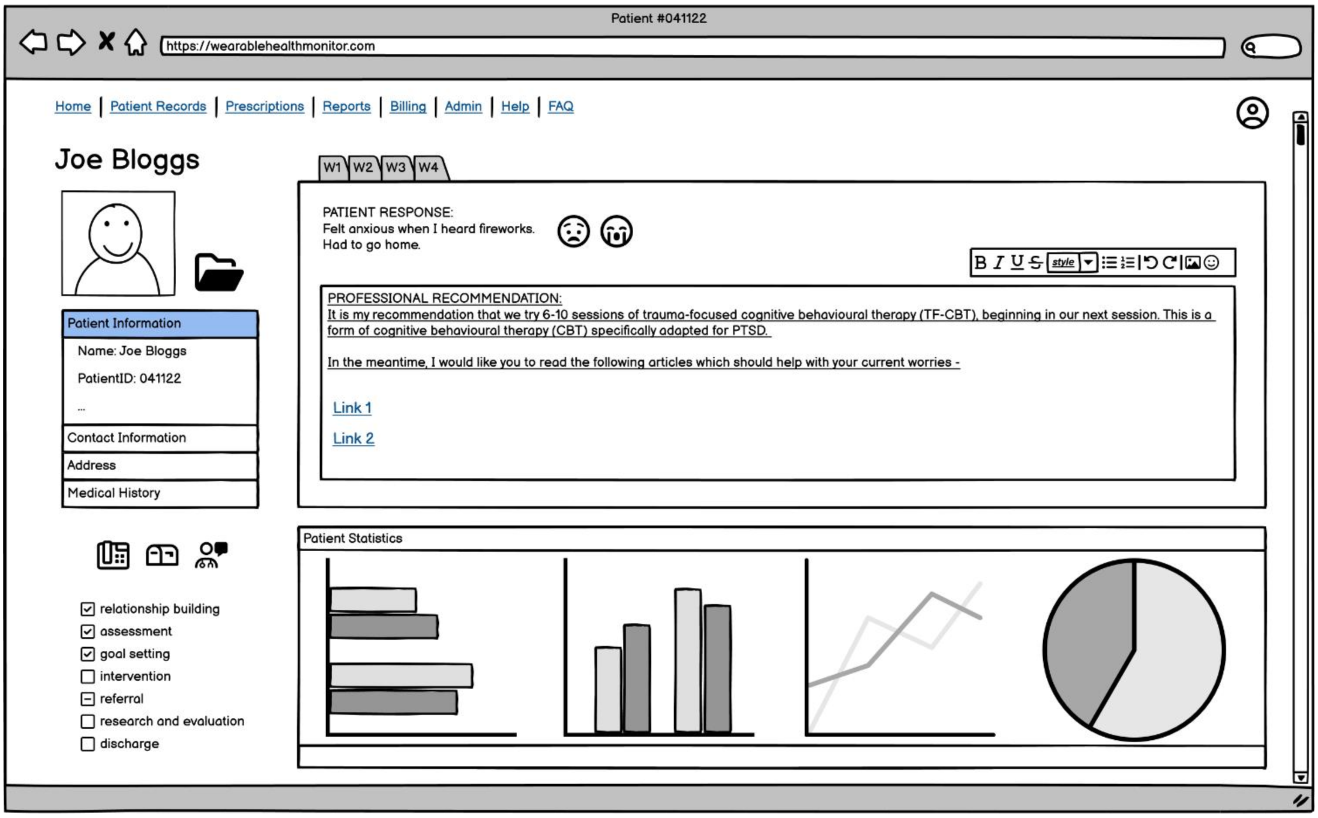

Figure 4: Mock-up Therapist Webpage View of Proposed Solution [6]

Physical View (5%)

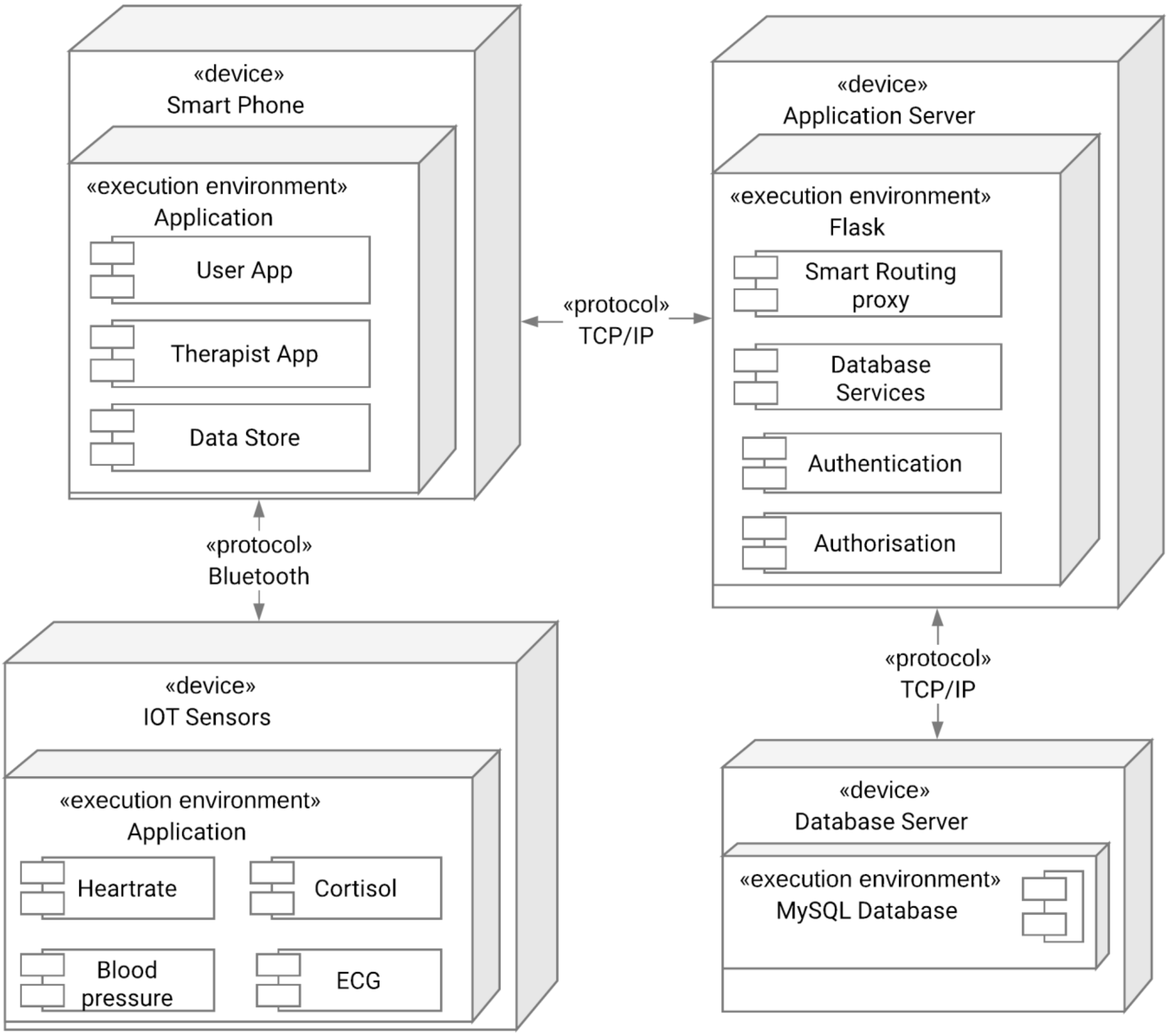

Figure 6: Physical View of Proposed Solution [7]

Physical sensors are located on the smartwatch worn by the user. These sensors will measure heart rate, cortisol levels, and blood pressure and can run an ECG to measure the heart's rhythm and electrical activity. This collected data will then be sent to the user's smartphone over a Bluetooth connection, and subsequently to the application server and database server over a TCP connection.

The smartphone will be home to an application that both the user and therapist can access, albeit with slight differences. The smartphone application will connect to the application server for authentication purposes when logging in. The application will then connect to the server for data authorisation and return only the relevant data for the current user's privilege level. The application will perform data analysis and visualisation for both the user and the therapist, as well as displaying therapist-based recommendations during an incident.

The application server will be a bridge between the smartphone application and the database server, performing all authentication and authorisation services as well performing database queries. The server will contain a smart proxy router to increase security and route traffic to the relevant service.

The database server will be home to a MySQL database which will contain information on all users and supervisors, as well as the IoT data from the sensors and therapist's notes and recommendations.

Process View (15%)

Performance

The system should have a high throughput of data from the IoT sensors for each user. The system must be able to load balance efficiently to concurrently handle the receival and retrieval of data from many users at the same time.
The system will also cache popularly visited pages on the app that change infrequently, such as NHS recommendations on reducing stress levels or anxiety.

Scalability

The system should be able to scale to meet current usage demands which could vary drastically throughout the day. A containerised microservices architecture with orchestration will be deployed to handle this.

Availability

The system must be available at all times of the day with the overall availability of 0.99. Our containerised microservice architecture with fault detection will be able to automatically redeploy containers that are not functional and allows for redundancy between different geographic regions to in order to meet this requirement.

Reliability

The system should be highly reliable and fault tolerant. The microservice architecture will ensure the application can recover from failure automatically without causing user disruption.

Maintainability

The system will be highly maintainable with automated deployment and CI/CD pipelines. Containerised microservices are perfect since each service is independent and can be updated without impacting the other services
.

Security

Authentication of users/applications - verification of identity using passwords, biometrics or digital certificates.

Authorisation of users/applications – Will have an Identity and Access management system in place which will follow the principle of least privilege. This will ensure that only an authenticated user has access rights to the relevant data or services.

Data integrity – The use of checksums to identify data transmission errors, or the use of cryptographic hash functions to ensure data integrity against malicious modifications.

Confidentiality – As we will be working with personal user data, it is essential that it is encrypted. An asymmetric encryption method will be used. This involves a singular cryptographic key to encrypt and decrypt any user data that is received. As only a single key is used for both operations, it makes the process relatively straight forward whilst also making it extremely fast and more efficient.

Login and Analytics Process from User Side

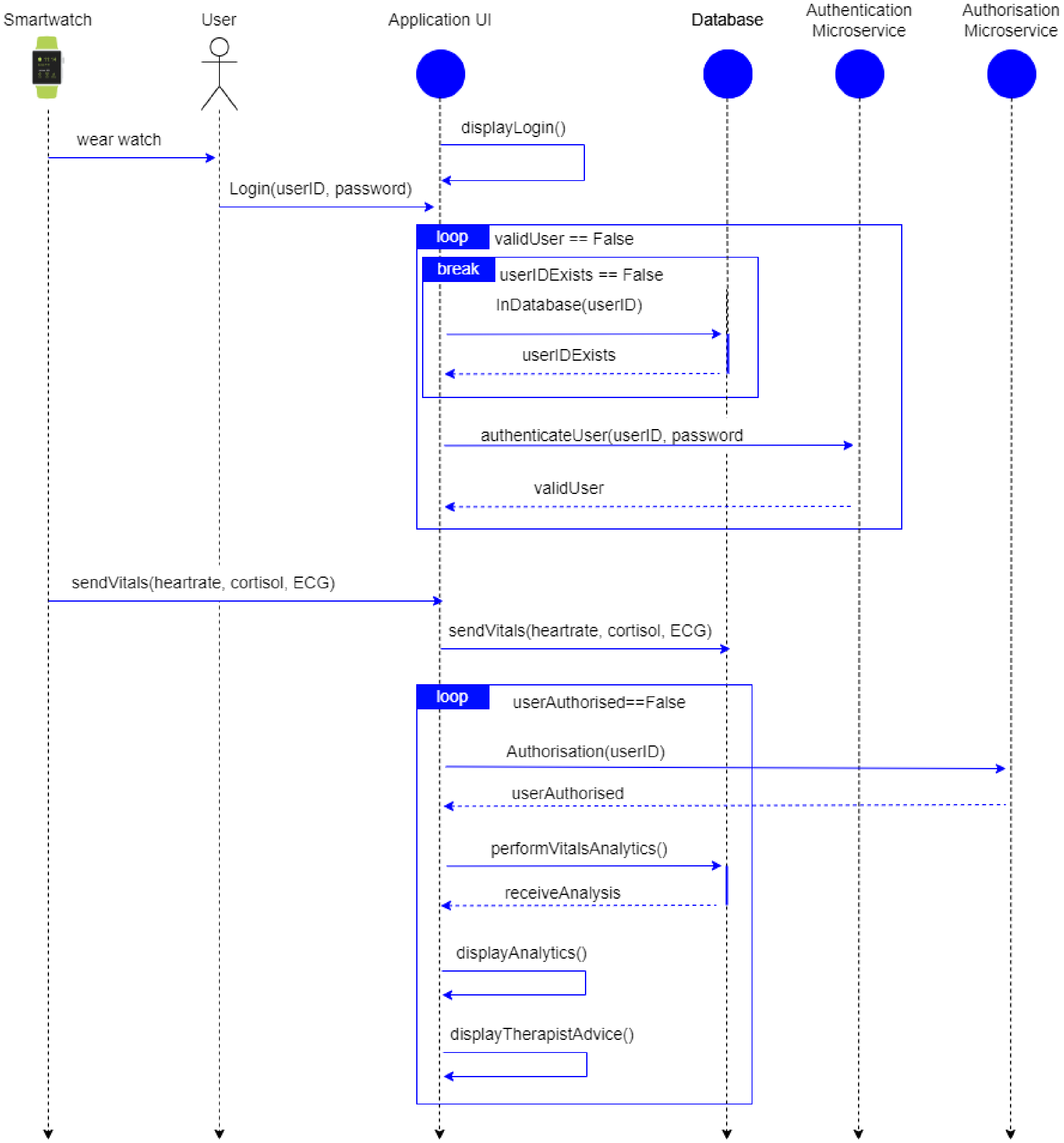

Figure 6: Sequence Diagram of Proposed Solution from Patient Perspective [8]

The user opens the application which displays a login screen. The user can then enter their user id and password which is validated using the Authentication microservice. If the user id and password is correct the user's smartwatch will begin logging vitals, such as heart rate, cortisol levels, blood pressure, and ECG readings. These vitals will then be sent to the smartphone and subsequently stored in the server database.

The smartphone application will also check the user's authorisation level, and if this pass can receive and display analytics on the user's vitals and display therapist's advice.

Login and Analytics Process from User Side

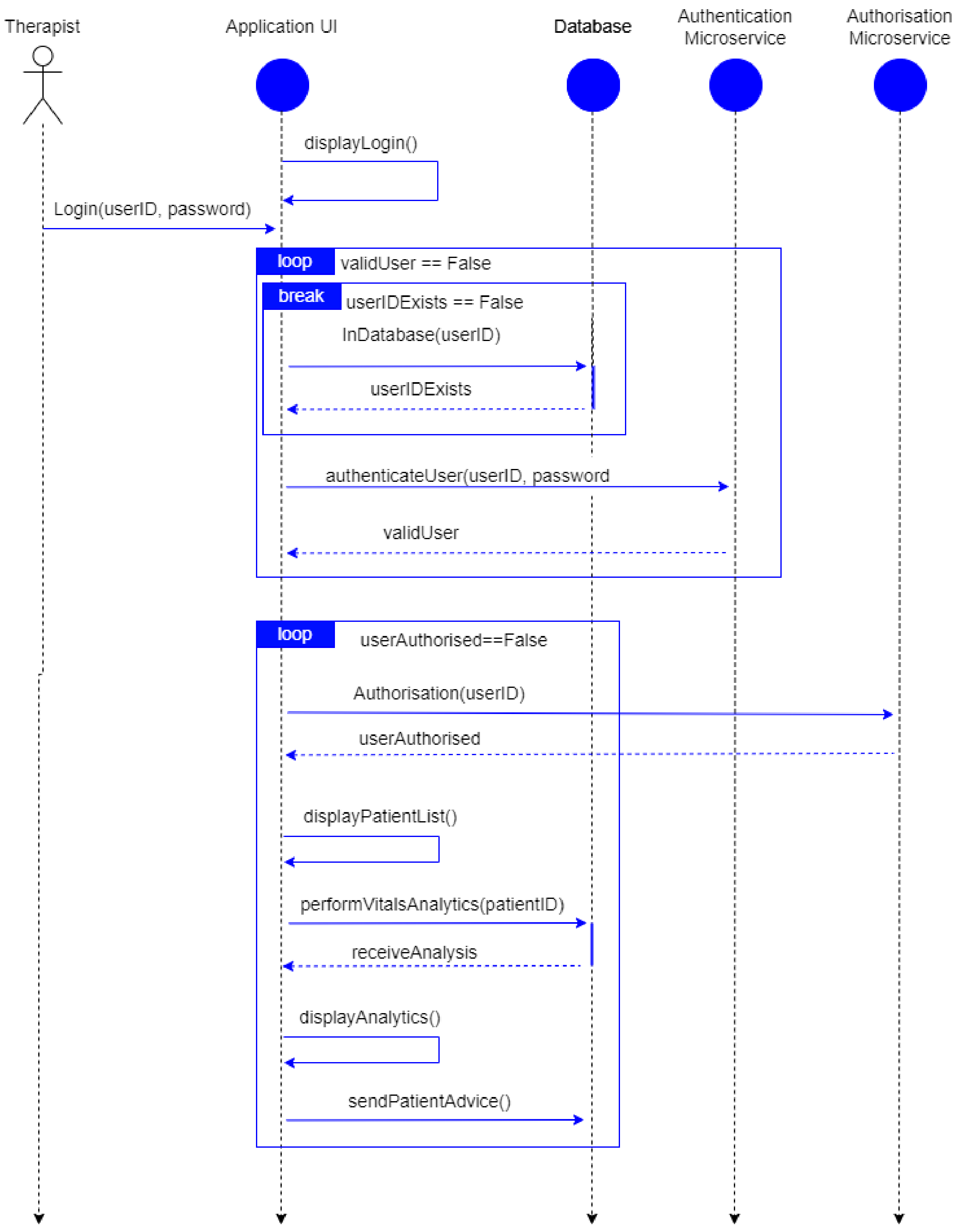

Figure 8: Sequence Diagram of Proposed Solution from Therapist Perspective [8]

The therapist opens the application which displays a login screen. The therapist can then enter their user id and password which is validated using the Authentication microservice. If the user id and password is correct the smartphone application will also check the user's authorisation level, and if this passes the user can view their list of patients and view analytics on the patient's wellbeing. From this, the therapist can then log advice for the patient to view.

Development View (10%)

<u>Architectural Style</u>

We shall implement a microservices architecture [3] as it offers a more dynamic and agile approach to developing and managing applications compared to a monolithic architecture. It can allow our development team to work in parallel on different independent components of the system and in any programming language, they desire, such as login services, authentication or retrieval, and saving of data. [9]

This architecture style also allows implementation of services using Docker containers with rapid scaling of services based on demand using container orchestration. It also increases fault tolerance as components are isolated therefore a problem in one service will not take down the entire system.

<u>Software Process</u>

We will use an agile approach with one-week sprints and daily stand-ups since we have a very short space of time to get this project completed. This approach prioritizes quick delivery, collaboration, and adapting to change. This way we can quickly identify and resolve any blockers and dynamically assign tasks to team members.

We will have an initial discussion to identify each member's strengths and weaknesses, such as front-end development, back-end, or cloud services. From this, we can more accurately designate tasks to the relevant individual. Each team member can then work to their strengths during development.

A project manager will also be assigned to keep track of outstanding tasks and set strict deadlines to increase productivity and ensure work is completed on schedule.

During each one-week sprint, team members will be able to bring up any specific issues they have experienced during development. The team then will be able to consolidate and plan how to tackle this in accordance with their abilities, this may result in one member of the team taking over from someone else and working together to solve a specific issue.

For large tasks, we will first split them into smaller goals which can make hitting the large milestone much easier.

A project backlog will initially be created to identify all project requirements. These requirements can then be partitioned and assigned to a category, such as front-end, back-end, cloud services or security. This will ensure that each requirement can be handled by the relevant person.

Project Management Tools

We will use Miro for project backlogging as well as collaborating with and assigning tasks to each member for each sprint. This way we can easily keep track of our progress and who is working on what to ensure we are on target to meet the project deadline. Miro can also be used to create collaborative mind maps to visualise how the different parts of the system will interact together.

We shall use GitLab to hold the project repository as well as managing version control, issue tracking and integrating CI/CD pipelines for compiling testing and validating software builds.

For team communication, we will utilize Microsoft Teams as this also supports miro workspace and GitLab integration so we can easily view everything within the Teams app itself.

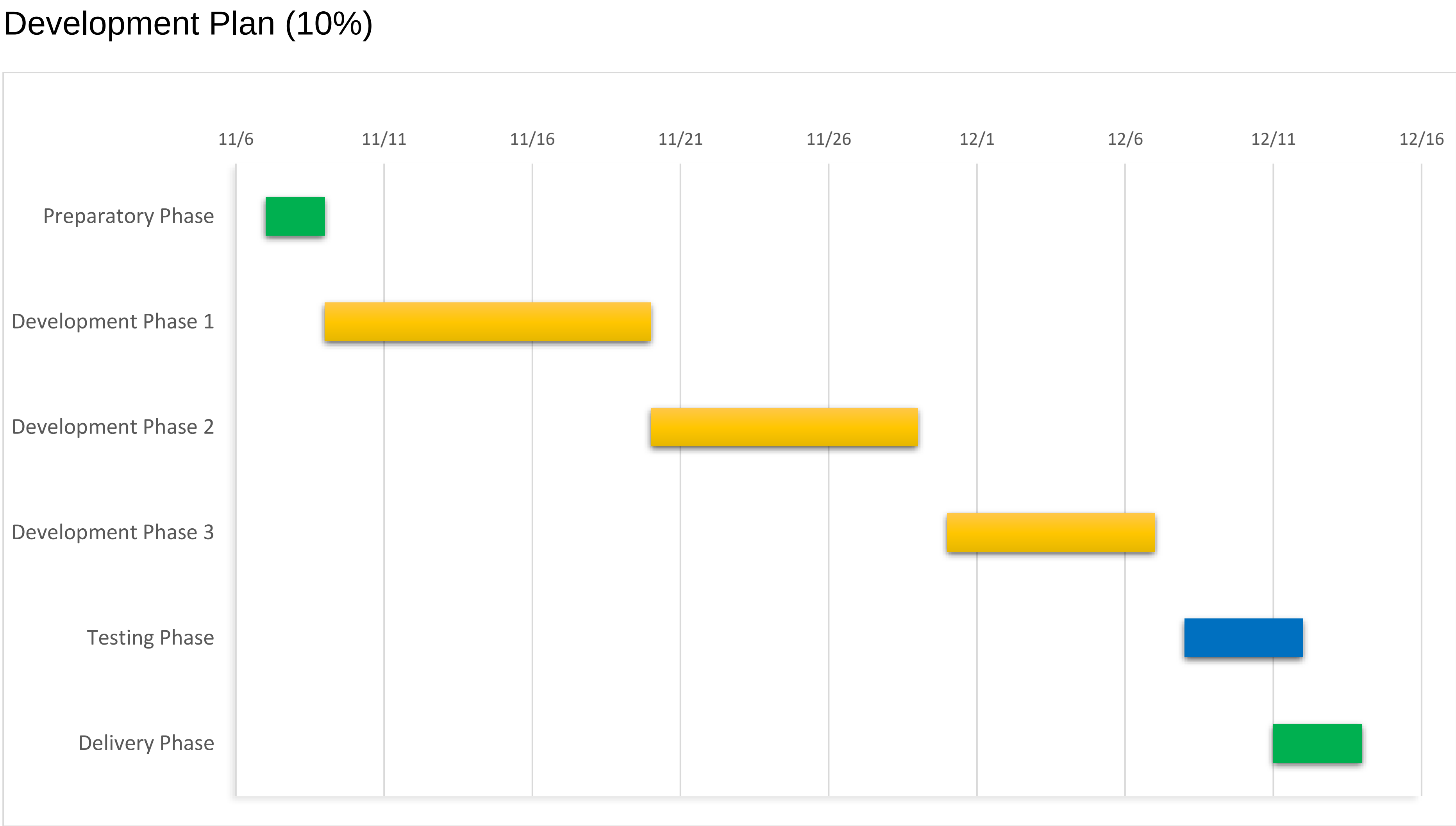

Figure 9: Gantt Chart for Development Plan

With just over 5 weeks to develop a solution, our time must be used effectively and efficiently. As with any piece of agile software, requirements can change during development.

When developing our solution, the Gantt chart used will be followed in order to develop an MVP in time and with most, if not all features implemented.

The preparatory phase will last only a couple of days. This phase of development will ensure each member of the group is aware of their in the development of this solution as well as having software environments set up and ready for development.

Three development phases have been allocated, with each development phase lasting a week or longer. Development phase 1 lasts the longest at 11 days. As the team gets to terms with developing a new system with technology some have never used before, adding more days to this development phase is the best way forward. By the end of phase 1, the team hopes to have a basic software solution developed, a piece of software that can be a foundation for the following two development phases. The application server will be planned to be completed in this phase alongside as well as a basic interface for the user and supervisor roles. The application server will be several containerised microservices built using Docker [10], that will be deployed on an orchestration platform, such as Rancher [11]. Rancher will be used to ensure adequate load balancing and dynamic scaling of these microservices based on demand.

With an initial skeleton solution developed, development phase 2 can begin. This will involve creating a database server that will store various user health data alongside. The database software used will be MySQL running on Amazon Web Services [12]. This will link with our initial solution that has been developed in phase 1. With IoT being the main source for retrieval of health data, AWS IoT Core [13] will be utilised for data ingestion from these devices.

With an initial user and supervisor interface, application server and database server created development phase 3 can begin. This phase of development can be seen as building on our first two development phases.

With a Minimum Viable Product developed, the testing phase of the project can commence. Manual testing will be carried out alongside unit tests. Manual tests will put us in the perspective of the user and supervisors that will utilise this solution daily while unit tests are low-level and close to the source of the solution. Both applications of testing will ensure bugs are kept to a minimum before an MVP is presented. Four days have been set aside for this testing phase.

With testing completed, the delivery phase can begin. Two to three days will be devoted to making sure the software solution is available to users and supervisors.

References

[1] B. Wang et al, “Wearable aptamer-field-effect transistor sensing system for noninvasive cortisol monitoring,” Science Advances, 2022.

[2] HBC Editors, “Healthcare Business Club,” 25 August 2022. [Online]. Available: https://healthcarebusinessclub.com/articles/consumer/8-uses-of-smart-wearable-medical-devices-in-everyday-practice/. [Accessed 01 November 2022].

[3] D. T. CSC4008. [Online]. Available: https://canvas.qub.ac.uk/courses/19773/files/3126721/download?wrap=1. [Accessed 31 October 2022].

[4] NetApp, “What are Microservices?,” NetApp, 2022. [Online]. Available: https://www.netapp.com/knowledge-center/what-are-microservices/. [Accessed 3 November 2022].

[5] “Lucidchart,” [Online]. Available: https://www.lucidchart.com/pages/uml-package-diagram. [Accessed 01 11 2022].

[6] Balsamiq, Balsamiq, [Online]. Available: https://balsamiq.com/. [Accessed 30 October 2022].

[7] Lucidchart. [Online]. Available: https://www.lucidchart.com/pages/. [Accessed 30 October 2022].

[8] “Create UML sequence diagrams in draw.io,” Draw.io, [Online]. Available: https://drawio-app.com/create-uml-sequence-diagrams-in-draw-io/#:~:text=Creating%20UML%20sequence%20diagrams%20in,notation%20used%20in%20sequence%20diagrams.. [Accessed 29 October 2022].

[9] Coursera, “What is Agile? And When to Use It,” Coursea, 15 June 2022. [Online]. Available: https://www.coursera.org/articles/what-is-agile-a-beginners-guide. [Accessed 3 Novemeber 2022].

[10] Docker. [Online]. Available: https://www.docker.com/. [Accessed 2 November 2022].

[11] Rancher. [Online]. Available: https://www.rancher.com/. [Accessed 02 November 2022].

[12] AWS, "Amazon RDS for MySQL," [Online]. Available: https://aws.amazon.com/rds/mysql/. [Accessed 03 November 2022].

[13] AWS, "AWS IoT Core," [Online]. Available: https://aws.amazon.com/iot-core/. [Accessed 03 November 2022].

Appendices

Appendix 1 – How Document Work was Split Up.

Introduction	James Cassidy
User Stories/Scenarios	Lewis Cooley, Ciaran Gernon
Scenarios	Lewis Cooley, Ciaran Gernon
Logical View	Lewis Cooley, Ciaran Gernon
Physical View	Nathan Donaghy
Process View	Nathan Donaghy, James Cassidy
Development View	Nathan Donaghy
Development Plan	James Cassidy