Final Report

Integrated Biomedical Engineering & Health Sciences:
Department of Computing and Software

April 7, 2023

Repo: https://github.com/Zack-Ren/Hypnos

Zackary Ren, renx11, 400192420
Aditya Sharma, shara24, 400189221
Ethan Dhanraj, dhanraje, 400185166
Christian Hsu, hsuc13, 400210638

Table of Contents

Background Research

Sleep is one of the most important factors for maintaining a healthy lifestyle. Individuals are more vigilant and alert with proper sleep, and display higher cognitive ability, attention span, and clear thinking compared to those who are deprived [1]. Proper sleep also plays an essential role in emotional regulation; sleep deprivation has been shown to increase individuals' reports of subjective stress in relatively low stress conditions [1]. Ideally to achieve maximal cognitive function research has stated approximately seven to seven and half hours is required; however, it is a common occurrence to not get enough sleep [1]. For some, a lack of sleep may be due to disruption, which may be caused by stress, pain, jet lag, or sleep apnea [1]. Daily sleep disruption is harmful as these deficits can accumulate over time, resulting in 'sleep debt' [1]. Therefore, it is imperative to diagnose the cause of disruption and solve it quickly.

Sleep apnea is classified as ten seconds between breaths and hypopnea is a decrease in airflow for more than ten seconds resulting in a decrease in arterial oxyhemoglobin saturation and electroencephalographic signal [2]. Obstructive sleep apnea typically occurs when the upper airway is mechanically obstructed [3]. Usually this occurs when the muscles of the nasopharynx and surrounding tissues block the airway [3]. Oxygen levels are subsequently decreased while carbon dioxide levels increase, thus causing the subject to experience disrupted sleep [3].

The current standard for diagnosing OSA is polysomnography (PSG), which involves a collection of different recording tools that measure different signals associated with sleep. Typical instrumentation for PSG includes electroencephalography (EEG), electrooculography (EOG), electromyography (EMG), electrocardiography (ECG), airflow sensors, respiratory belts, pulse oximetry, body position sensors and video monitoring [4]. The results of each of these diagnostic tools contribute to the apnea-hypopnea index (AHI) which details the severity of sleep apnea [4]. More recently, many clinicians have suggested performing home sleep apnea tests (HSAT) as an alternative to PSG.

The American Academy of Sleep Medicine (AASM) has analyzed the use of home sleep apnea tests as an alternative medical test for the diagnosis of OSA. HSAT devices are classified by the FDA in the range of Class IV to Class II medical devices depending on the number of sensors [5]. These devices differ from PSG as they exclude EEG, EOG, and EMG and can therefore only estimate the severity of respiratory issues [5]. Consequently, HSAT devices usually underestimate cases of OSA and cannot detect hypopnea [5]. Despite the disadvantages of HSAT devices, the AASM believes they can be useful tools in diagnosing OSA. There are specific guidelines that must be followed to use HSAT devices: only a medical provider can diagnose OSA, the need for an HSAT must be ordered by a medical provider after evaluation of medical history, an HSAT should not be used by members of the public if they are asymptomatic or used as a screening tool, raw HSAT data should be evaluated and interpreted by certified professionals in sleep medicine, and diagnosis should not be based only on the results of an HSAT device [5].

Problem Analysis

Problem

Polysomnography, while currently the gold standard for sleep apnea diagnosis, has several drawbacks that need to be addressed. First, patients must undergo the examination in a specialized

facility or lab which can inconvenience a patient's daily routine. Additionally, sleep disorders may not occur daily, and their severity may vary [4]. The "first-night" effect is when sleep duration and quality is reduced when sleeping in an unfamiliar environment, which also negatively impacts sleep testing [4]. Second, PSG requires trained medical staff and expensive equipment, resulting in costs of approximately $1200 USD for the patient [6,7]. The steep cost in conjunction with the likely need for multiple tests leave ~85% of people with sleep apnea undiagnosed [3].

By understanding the limitations of polysomnography, the following problem statement was obtained: Polysomnography is an expensive, inaccessible, resource-intensive and is also an error prone test. The gold standard demands a high investment of time and money from patients, driving away individuals who require the treatment, resulting in many cases going undiagnosed.

Design Criteria

To address the problem statement, a more convenient and accessible method for sleep apnea classification must be developed. We leveraged traditional elicitation techniques such as background readings, interviews, questionnaires to better understand the domain and scope of the problem. We used existing research to learn about challenges associated with diagnosing sleep apnea and the experience that patients and clinicians undergo during the procedure. Open interviews were conducted with potential stakeholders such as professors, clinicians, and civilians. We were able to determine what features are necessary in designing a system for sleep apnea diagnoses. Questionnaires were issued to students (individuals not diagnosed with sleep apnea, but with some domain knowledge) to understand the perspective of potential patients.

The first criterion is that the device shall obtain positional data when the patient is sleeping for the duration of the sleep. This is required to determine sleep position and breathing rate when assessing sleep apnea severity. The second criterion is that the device shall not disrupt sleep nor exacerbate sleeping disorders. This is to minimize the variance of sleep disorders and the "first-night" effect that current patient's experience. The third criterion is that the device should be light-weight and flexible. This empowers users to be able to leverage our solution without much physical unease. The fourth criterion is that the device shall be affordable and easily accessible to the public. This removed the financial barrier presented by PSG. The fifth criterion is that the device shall be usable at any geo-graphic location with an internet connection. This criterion helps provide patients with the flexibility to undergo testing at any location they are most comfortable with, minimize the "first-night" effect. The sixth criterion is that the device shall be capable of providing a preliminary assessment of the acquired data to the physician. Automated assessments can expedite the lengthy PSG process for all parties. The seventh criterion is that the data being acquired shall remain secure and confidential. This will empower patients to leverage our solution and will build confidence in our ability to protect sensitive data.

Final Design

Hypnos is an innovative software-based solution to address the issues associated with traditional polysomnography methods. Designed to provide a seamless diagnosis experience for sleep specialists and their patients, Hypnos combines a mobile and web system to offers features to aid in diagnosis and provides convenience to all. There are four major components of the system: a cross-platform smartphone application for patients to collect data, a web application dashboard for clinicians, a Mongo database, and a C# server (Figure 1). These components communicate with one another via REST APIs.

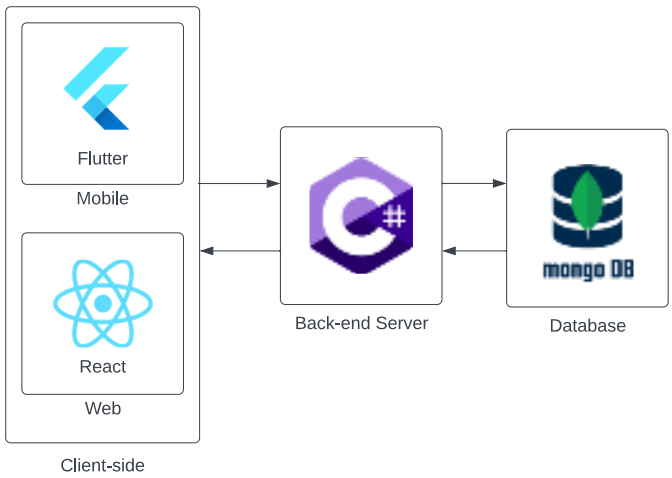


Figure 1: System architecture diagram.

Sleep specialists register patients by creating an account for them. Once registered, patients are provided a chest mounting device and given the login credentials for the smartphone app they download on their personal devices (Appendix A, Figure 3, 10). The patient is then instructed on proper usage and advised on any risks associated with the process. To record sleep data, patients start a new recording in the mobile app and secure the smartphone to their chest with the mounting device before sleeping. The smartphone's accelerometer and gyroscope sensors are used to record chest movements to extract sleep position and breathing rate (Appendix A, Figure 11). The data is stored as a batch once the patient stops the recording in the morning. From the web dashboard, physicians are presented with a list of their patients. In each patient's profile, all recorded data is visualized alongside additional metrics from the analysis algorithms, namely sleep position and breathing rate. The patient and doctor user flow and interactions with the system are described in a sequence diagram and use case diagram in Appendix A, Figures 1, 2, 4.

Mobile Application

The cross-platform mobile app is built with version 3.7 of Flutter and is compatible with any smartphone running a version of Android 19 to 30 or iOS 11 to 15. The app presents the user with a login page prompting them to enter their username and password. Upon logging in, users can visit the following pages via the bottom navigation bar: record, statistics, help, and profile. On the record page, users can collect their sleep data by pressing the center button to create a new recording. A pop-up is shown to confirm if the recording is intentional. Once the recording is confirmed and the patient secures the smartphone in the chest harness, relevant sleep data is collected from the phone's accelerometer and gyroscope sensors. The statistics page allows users to view all their previously recorded data in the form of acceleration and angular acceleration over time graphs (Figure 2). The help page provides clear instructions on how to use the app. The profile page lists the patient's personal information and their doctor's information (Appendix A, Figure 3).

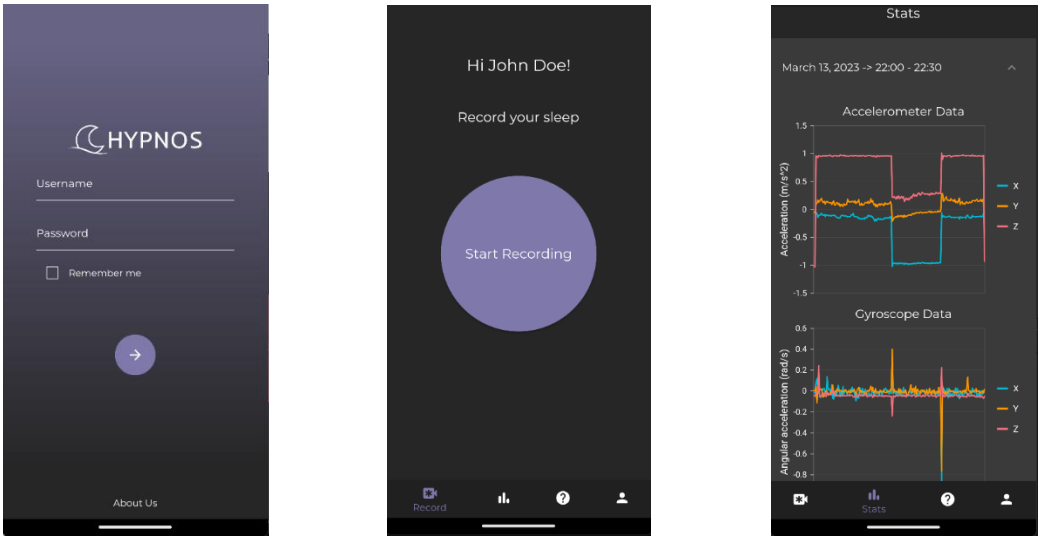


Figure 2: Smartphone app screens - login, record data, statistics (from left to right).

To address the design criteria of obtaining positional data, the smartphone's accelerometer and gyroscope sensors are leveraged to capture a patient's chest movements. This is the fundamental component in obtaining the required positional data for further sleep apnea analysis. Within the mobile app, patients can start a new recording and view previous recordings regardless of time and location if their device has an internet connection. To ensure the app is easy to use, intuitive icons and a help page were implemented to guide the patient around the app. The login page aims to improve data security by preventing unauthorized access to patients' private information. Flutter was selected to develop the cross-platform to address accessibility, as uses can access it regardless of the type of device they own.

Web Application:

The web application developed using React and written in Typescript, CSS and HTML, provides a comprehensive dashboard for physicians to manage and analyze their patients' sleep conditions. The platform contains three main pages – Login, Home, and Patient Profile (Appendix A, Figure 5, 6, 7, 8). The login page is the first point of entry for physicians, where they enter their credentials to access the dashboard. Upon successful authentication, they are redirected to the home page, which displays a list of all their active patients. Selecting a specific patient's profile displays the patient's personal information as well as a list of appointments (Figure 3). Each appointment has a section for the clinician to record notes and contains a list of diagnostic recordings performed by the patient. These recordings visualize the acceleration, angular acceleration, sleep position and breathing rate, helping physicians make a more accurate assessment of sleep apnea.

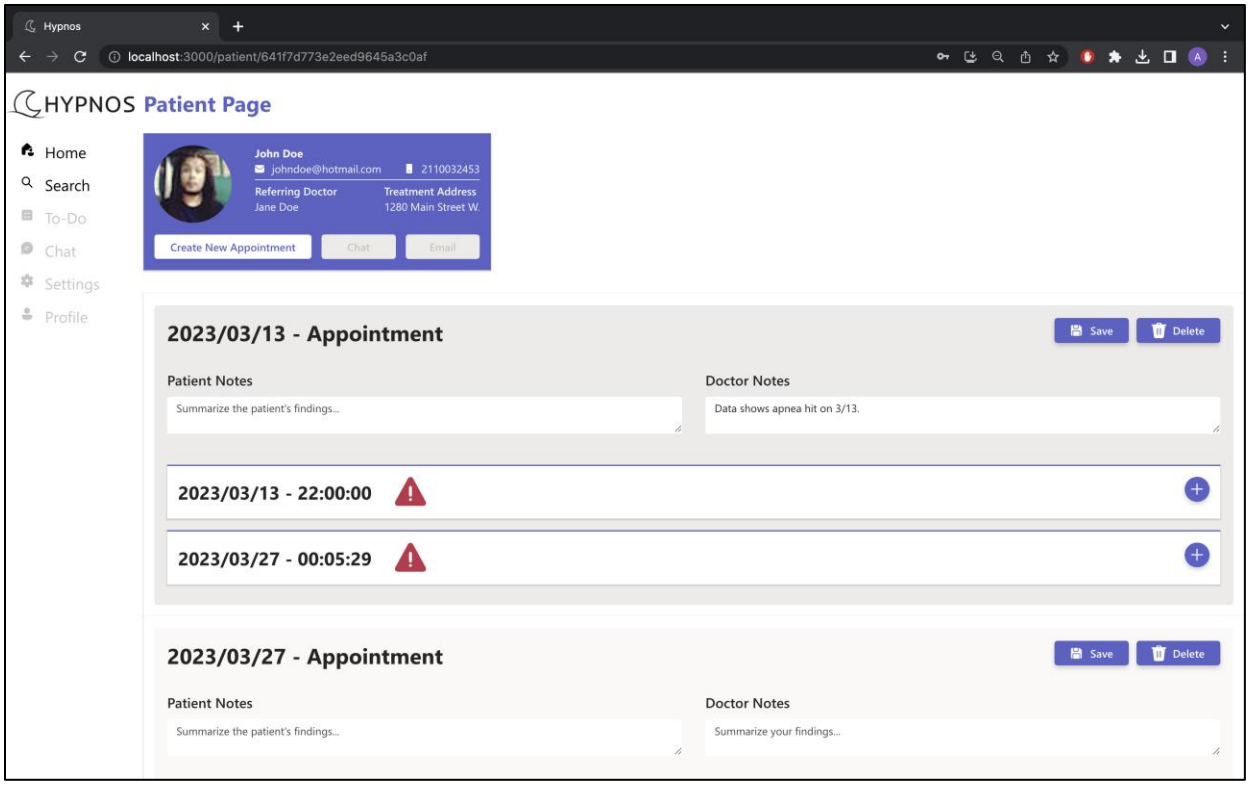


Figure 3: Web application Patient Profile Screen.

The dashboard displays the recorded data for the clinician to make their own assessment for sleep apnea. Sleep position and breathing rate are presented alongside the raw data to aid in the assessment. Having the dashboard protected by authenticated login protects patients' sensitive information. The web application also addresses usability concerns of PSG. Clinicians can easily find the patients they require with all their data in one location instead of needing to consolidate raw data. When the application is being hosted live on Amazon Web Services (AWS) for one year it will cost approximately $1600 CAD. The cost to host this platform for a year is equivalent to one PSG test, making it much more affordable for patients.

Backend

The backend application was developed using C# and .NET. The web and mobile component rely on this to acquire the information they need. Information is provided to each of the components in a stateful REST format where all information relevant to each of the resources (Analysis, Diagnostic, Doctor, Event, Patient, Login...) is provided. The Analysis resource contains the analyzed raw data, including sleep position and breathing rate. The Diagnostic resource contains the raw sensor data recorded by a patient. Doctor and Patient resources contain the information regarding both parties and include direct look-up links to each other. The Event resource contains information pertaining to an appointment that the patient set and contains a list of direct look-up links to the diagnostic data while also containing patient notes and doctor notes about the appointment (Appendix A, Figure FIX). The introduction of the backend service reduces the number of requests and the complexity of requests that would need to be issued to the database.

This is because the backend service exposes commonly used requests that the frontend components consume most frequently.

When patients are removed, all data associated with that patient should be deleted as well. This includes all records of their appointments and recordings. This also involves modifying their corresponding doctor and removing that patient from that doctor's patient list. To avoid introducing this computational complexity on client-side code, the back-end server was developed.

Data Analysis

The data analysis module is implemented within the system architecture as an API endpoint in the backend. The analysis module takes the raw accelerometer and gyroscope data, collected from the mobile application, and stored in the database, as input and returns metrics that aid in the assessment of sleep disorders. The two primary outputs of the analysis in the final design are sleep position and breathing rate. Methods for classification of sleep apnea and hypopnea using accelerometer and gyroscope data have been explored, however, they have not been implemented in the final design (Appendix A, Figure 9). The analysis is executed from the web application such that its outputs can be viewed by the clinician in addition to administrative information and diagnostic notes. As such, data analysis is executed asynchronously from data collection.

The analysis module classifies datapoints into four different sleeping positions (back/supine, stomach/front/prone, left-side, right-side) and upright (exception case). The analysis returns a sequence of sleep positions with the same length as the recorded data. This is used to indicate when and for how long one a patient is in a particular sleeping position. The analysis also returns a sequence of estimated breathing rates averaged across overlapping 60-second time intervals, in units of breaths per minute.

Database:

The selection of a MongoDB NoSQL database was a strategic decision based on the specific needs of our system. The primary advantage of a NoSQL database is the flexibility in supporting unstructured data. This NoSQL Databases such as MongoDB are designed to handle unstructured datasets and this nature makes them particularly useful for handling real-time applications that interact with a variety of data such as images, videos, audio, text and numerical data. This ability met the need of our application which contributed to why it was selected. Currently, we are using images, text and numerical data; but the potential to leverage audio and video will prove to be helpful when working with a NoSQL database. Furthermore, we selected this because of its scalability potential; it can scale horizontally across multiple servers, and this makes them ideal for handling high traffic and large datasets. Lastly, the cost-effective nature of NoSQL databases played a role in the long-term vision we had for our project. They are more affordable because they rely on commodity hardware and therefore require fewer resources than SQL databases. MongoDB was specifically selected due to the familiarity that we had with it and its APIs.

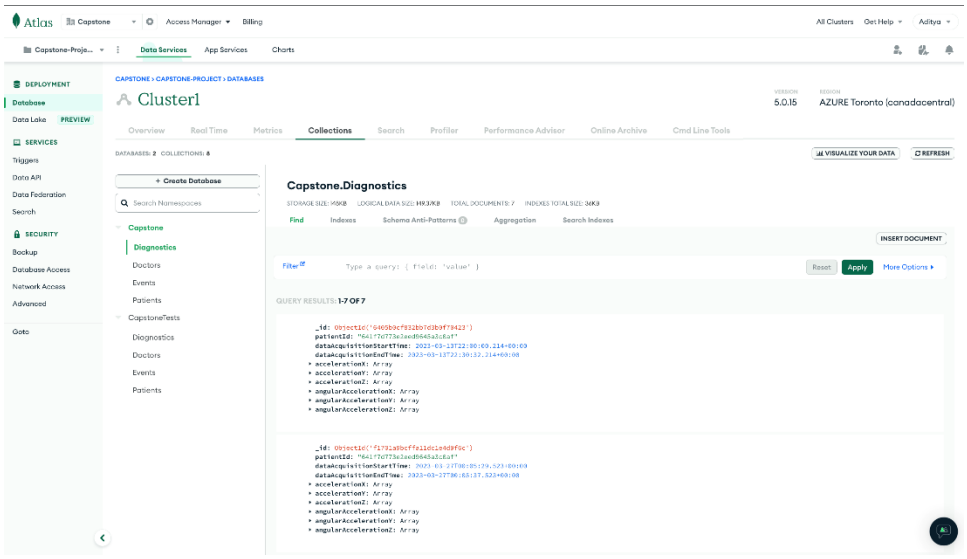


Figure 4: MongoDB NoSQL Database

Analysis

The analysis of the mobile and web application primarily pertains to design principles of human-computer interaction (HCI). This analysis will combine principles acquired from departmental knowledge as well as supplementary research about HCI. One principle that is followed to assess is safety and stability. The interface should be designed so that mistakes are reduced but if they occur the ability to recover is present [8]. A second principle is consistency. Modes of interaction should have similar appearance and layout [9]. The display of information should also be similar as both factors aim to reduce learning time and mistakes (which also makes the system safer and more stable). The next principle is the principle of habit which is that user's intuition and history must be considered [8]. This can apply to icons, naming conventions and layout of the interface itself [8]. Another principle that will be used to assess the solution is the use of shortcuts. This pertains to making navigation more efficient when filtering through copious amounts of information [9]. A fifth principle is feedback as the system should be able to respond to a user's actions to provide information [9]. This feedback can be displayed in a multitude of methods if the user is able to easily respond to the feedback [9]. An extension of that principle is providing effective help to users. A good interface should provide potential solutions or actions to users to solve problems or guide better usage [9]. Lastly, visually an interface should be easily identifiable which is built on the usage of color and sizing [9]. These principles along with will be addressed in the analysis of the web and mobile applications. The use of Nielsen's Heuristics was also used for analysis, some of the heuristics are repeated from the principles above. This informal heuristic evaluation was also used throughout the development process to guide adjustments.

Mobile Application

HCI design principles were applied to ensure a smooth user experience. If the user makes an error in entering their login credentials, they are provided appropriate feedback and are prompted to re-enter their information (Appendix A, Figure 3). All the screens are kept simple, with a consistent theme. Text colour maintains high contrast with the background for sufficient visibility while

intuitive icons guide users around the app. The record button affords a press to interact with. The help screen is in place to provide help by guiding users through the expected user flow.

Web Application

The first focus was the visibility of different elements. Colour selection was important to ensure that elements did not blend into each other. As seen in the screenshots of the prototype the background of elements is white or grey while foreground elements would have darker colour (Appendix A, Figure 8). Sizing of text and different elements that could be interacted with were made larger and as a result pages have less elements on them which also aids in making navigation simpler. Therefore, the web application satisfies the design principle of elements being easily identifiable. Another focus was to ensure elements that needed to be interacted with were clear. The use of icons and buttons as signifiers for affordances is very apparent within the web application. Examples include the use of a house icon for navigation to the Home page or a plus/minus icon for an expandable section. The use of these icons is consistent with what users will typically interact with other systems satisfying the consistency and habitual design principle. Another indicator that was used for interaction was that certain elements were responsive when hovered. Cursor changes on buttons or text inputs were implemented. To increase navigation a hierarchical design was applied. The use of spacing and grouping were utilized to ensure that associations were clear. An example is found within the patient information page where it is clear which recordings are associated with each appointment. A weakness found within the design is that the web applications are missing “breadcrumbs” which are indicators that a user has navigated to a certain page before. This was discovered during usability testing and indicates that safety of the interface is lower since recovery from mistakes is difficult. Another weakness is that currently the web application does not have any shortcuts for clinicians, however this is planned to be implemented as a search feature for patients. The last weakness is that there is not a large amount of user feedback and help provided within the web application. With the current implementation there are not many opportunities for feedback. An invalid login attempt is the only occurrence of feedback. Some areas where feedback could be included is creating an appointment or creating appointment notes, this feedback can be successful or failed interactions.

Data Collection, Analysis and Backend

There are several technical considerations with regards to data collection and analysis that were analyzed with appropriate theory and modelling. Such considerations include hardware requirements and sampling frequency for data collection as well as time complexity of analysis algorithms. With regards to hardware requirements, the mobile application was developed in Flutter and supports iOS (11-15) and Android (19-30) operating systems. Both an accelerometer and gyroscope are required for recording linear and angular acceleration in three dimensions. There are constraints associated with battery consumption while recording data, however, the chest mount allows for one to record data with a phone that is charging (with an extended charging cable).

For data collection, an appropriate sampling frequency was determined using the Nyquist-Shannon sampling theorem. This is an important consideration as the data needs to capture breathing rate while minimizing the total amount of data (avoid collecting unnecessary/useless data) and the

impact on battery consumption. A sample calculation for the minimum sampling frequency is given in Appendix B. This assumes a maximum breathing rate of 20 breaths per minute, however sampling frequency was chosen conservatively [10]. The results of this analysis lead to using a sampling frequency of 1 Hz for accelerometer and gyroscope data. In addition to the theory, the sampling frequency was determined appropriate in practice by assessing collected data which was able to describe subject breathing rate sufficiently.

Another consideration that was analyzed was the time complexity of the analysis algorithms. Both algorithms for classifying sleep position and estimating the breathing rate have a time complexity of O(n). This is the case as both implemented algorithms involve single iterations through the full sequence of accelerometer and gyroscope data. Identifying sleep position is the simpler algorithm which involves a single iteration through the data in which each point is classified based on the x and z acceleration vector components. The time complexity for identifying breathing rate could potentially be improved, however, this would come at the cost of the accuracy of duration in sleep positions. Such a method would not require evaluating each point but rather only considering points following sections of movement. It was decided that this potential increase in efficiency was not worth the risk of lower accuracy and implementation complications. Estimating breathing rate takes a series of steps which also include preprocessing. Data preprocessing requires a single iteration through the data which checks for regions of rapid and excessive movement. Other steps including finding local extrema (comparing neighbouring data points) in the y acceleration component, computing time intervals between points, and filtering the data all require single iterations and have complexities of O(n). Thus, the time complexity of the full analysis is on the order of O(n).

Regarding the backend server, there were a total of 30 functions that were written. The worst time complexity amongst all of the functions was of the order O(E * D) E is the number of events in the database and D is the number of diagnostics associated with the event. Aside from that, the most common time complexity for each of the functions was O(N) where N is the largest number of elements being processed within that function. We were able to minimize the time complexity by relying on the key-value pair nature of the NoSQL database and by designing the database schema in a way such that the information looking for is always available or acquirable via direct look-ups instead of iterations. For example: Normally, if you would want to find a patient's doctor, you would need to iterate over the set of all doctors to find the one that manages the specified patient. However, we made sure to include the doctorId in the patient object so that we could avoid that kind of processing and vice-versa. In a way, our database collections serve and act as a graph with each node pointing to each other. This consumes more space, but it ensures that our processing and look-up speeds are fast.

Database

The largest contributor/consumer of our data will be the Diagnostic Object. Each object will consume roughly 1.4MB of data (Appendix B, Database Storage). The rest of the objects are negligible compared to the size that this object consumes since the rest are simply around 30-60 bytes per object (Appendix B). The size of the database will scale depending on how many Diagnostic objects are being created. We assume that patients will consistently record data for

roughly 1-month before enough data is collected to confidently establish a trend/ diagnosis. This means there would be roughly 42MB of data for each patient when the doctor assesses it. The size will then scale depending on the number of patients. We predict that we will not need more than 100GB of storage space. We have based this prediction based on the assumption that we acquire and maintain 1000 users (~50GB) within the first year of operation and assuming we back-up the data as well (the other 50GB).

With regards to the database, we plan on migrating our entire infrastructure to Azure or AWS and this provides us with many long-term storage options. Firstly, both are HIPPA compliant and have their own built-in securities. Secondly, we plan on moving patient data to long-term storage if the patient has not recorded data or interacted with the platform in over 3 months. This will reduce the size of our database and ensures only relevant data is being used/stored.

Prototype Testing and Validation

Usability testing for mobile and web applications consisted of the user examination test method. This method uses natural observation of a user as they attempt to interact with a product [8]. As users interact with the system their actions are recorded with emphasis placed on when the users encounter an issue [8]. The type of observation was unobtrusive. Any questions from users were answered with questions to guide discovery and learning. Feedback was only given when limitations due to the current prototype blocked testers. It is also desirable to conduct an interview to catch subjective feedback from the users such as attitudes and preferences associated with the product [8]. The usability tests will be explored more in detail within this section.

Mobile Prototypes and Validation

Multiple iterations of the mobile application were developed over the course of the project including a prototype to demonstrate feasibility, intermediate designs and prototypes, and the final application. The first prototype was developed as an IOS application in Swift with the goal of demonstrating the feasibility of using a smart phone to collect accelerometer and gyroscope data. Feasibility tests were successful and proved that a smart phone was suitable hardware for meeting the design criteria of the device. To allow for cross-platform development, subsequent iterations of the mobile application were developed using Flutter.

An initial mockup of the mobile app was created in Figma (Appendix C, Figure 26). This concept sketch was discussed in an interview with the stakeholder to ensure fundamental requirements and desirable features were encompassed. Once the requirements were elicited, the first iteration of the Flutter app was developed following the revised Figma sketch (Appendix C, Figure 26). During parts of development, pair programming was conducted to evaluate the project status and ascertain discrepancies from requirements. Usability tests were conducted after the completion of iteration one. Consenting participants were presented with the mobile app and no specific instructions. Fluent usage of the app is defined as being able to navigate to all screens and successfully start and end a recording. The time for participants to achieve fluent usage was measured; a usability rating and feedback was also collected. Implementing the feedback received from the usability testing led to the final iteration of the mobile app. The participants were asked to provide a second

usability rating with the results summarized in table 2, Appendix C. As a final test, a walkthrough was performed to ensure the application behaved as intended.

Web Prototypes and Validation

The first iteration of the web application contained very few visual elements that enhanced user experience but provided the framework for the full application. Images of the first web prototype can be found within Appendix C Figure 27, 28, 29. The primary focus of this prototype was to ensure the database was connected properly to the web application. All the necessary functions that require access to the database were tested to ensure the proper data was retrieved or stored. During this process, modifications were made to the database, notably extra endpoints were added to better control the web application. Once the important features of the web applications were able to be implemented it was presented to the primary stakeholder and technical supervisor. At this meeting it was suggested to shift the focus from refinement of functionality to enhancing the user experience for the next prototype. It was noted that the prototype was very difficult to interpret which led to the use of human computer interface knowledge for the final design.

The second prototype (final design) built is an overhaul to the user interface of the first prototype. The functionality of the first prototype was carried over to the second prototype and tested once the visual elements were updated. The main type of testing for this iteration was usability testing to make adjustments to better suit end users. Each test performed had participants learn the web application. The time it took for participants to navigate to all pages of the web app and view a test patients recorded data was tracked and overall ratings and comments were collected. The result of these tests can be found in Appendix C, Table 3. Using the feedback from the tests, changes were made such as making buttons clearer, adjusting colour schemes, making text larger and changing icons. As discussed in the analysis section these changes were made according to signifiers and affordances.

Server Prototypes and Validation

There were multiple iterations of the backend server. The first iteration had a total of 4 services: Patient, Doctor, Event, Diagnostics. Each service was responsible for managing the data dedicated to section. However, it was soon discovered that there were dependencies between the data and certain modifications to the data collection of a particular service would impact the data collection of another linked service. Therefore, there needed to be a change in the structure of the services. However, attempting to modify the data in the linked service would lead to a circular dependency error.

The second iteration resolved this error by combining all of the services into 1 Management Service that was responsible for managing the state for the data that was linked together. This also made sense because the data collections were interdependent on each other. For example: When a patient is deleted from the system, all relevant information to the patient must be purged.

Regarding validation, we made sure that tests were executed for the server. Live manual testing of each function and its interaction with the database were executed and recorded. Manual integration testing was also executed using a service known as Postman which ensured that all endpoints were operating as expected and returned the expected responses to any consumer.

Data Analysis Prototypes & Validation

The data analysis modules were subject to multiple iterations over the course of the project. The first iteration of analysis was implemented separately from the system in Python. The rationale for using Python for data analysis is that it has several libraries that can promote computational efficiency and simplicity in the implementation. This original script had to be executed externally from the system and included minimal data processing before using methods to identify sleep position and breathing rate. Initially, the plan was to export the Python script as a Flask API endpoint that could be accessed by the web application or connected to the database. After exploring this option, a design decision was made to instead translate the analysis from Python to C# such that it could directly be integrated with backend. While this design decision increased the complexity of the analysis methods, its benefits in simplifying the system architecture outweighed all drawbacks. This is because the additional time in computation it takes to perform in C# is much faster than the time it would take for the C# server to ping the Flask server to obtain the results of the analysis. This is because the request and response time would be indefinitely longer than computation time.

The second iteration of the analysis modules were implemented in Python and C# and integrated into the backend of the system. The same methods for classifying sleep position and estimating breathing rate from the first iteration were used in addition to further data preprocessing steps. Preprocessing seeks to identify regions in data that correspond to noise and subject movement and ensure that such sequences do not skew breathing rate estimates. See Appendix B for sample data and output graphs as well as flow charts for analysis algorithms.

Verification activities were conducted to assess the accuracy of sleep position classification and estimation of breathing rate output by the analysis modules. Testing was conducted by having a subject wear the mounting device and record data on the mobile application. To assess accuracy of sleep position classification, the subject lay in bed, as they would for normal usage of the device, and was instructed lay in a sequence of sleeping positions. The sequence was specified prior to testing and each position was maintained for a known time interval before transitioning the to next position. The sequence and duration of the known sleeping positions were then compared to those output from the analysis modules. Before testing, there was a high degree of confidence in classification accuracy given the known orientation of the phone with respect to the subject's body. This was reflected in testing as 100% accuracy was achieved.

To assess accuracy of breathing rate estimates, the subject maintained a single sleeping position and inhaled and exhaled for specified time intervals (ex: 3-second inhale followed by 3-second exhale) repeated over two minutes. Inhalation and exhalation times were synchronized with timer that was visible for the subject to follow. Trials were repeated with different breathing rates and sleep positions. The known breathing rates were then compared with the estimates output from the analysis. The mean squared error (MSE), average relative error, and expected range for average absolute error were 0.44 $\text{breaths}^2/\text{min}^2$, 3.4%, and 0.41-0.69 breaths/min, respectively. See Appendix B for testing details and results.

Social, Environmental and Financial Factors

The problem analysis demonstrated that the current standard for diagnosing sleep apnea is unsustainable. Social factors are derived from the need to attend a facility to have a test or multiple tests performed. These tests require a large amount of time that can take away from a patient's social life. A patient must sacrifice time away from friends/family as well as personal time, which can have negative effects on their mental health. Therefore, the new design was created to be more convenient. The use of a personal cell phone and mobile application allow for a patient to perform the test when they desire, instead of having to schedule appointments. Also, if more data is needed, they can also make another recording easily. The mounting system and operation of the app were designed to be used without the aid of a clinician. However, the inclusion of basic instructions allows for another individual to help the patient or for the patient to figure out themselves. Another social factor that was taken into consideration was the patient sleeping with other individuals or pets. While the device was designed to not disrupt the patient's sleep it also needs to not disrupt anybody around them. Since data is being recorded with a mobile device the solution is not very noisy and the pouch of the mounting device helps to reduce any noise. The mounting device is also only restricted to the patient, there are no external attachment points, and it is very compact, so it does not occupy a large space.

Environmental factors to consider is the amount of power used to perform polysomnography. Polysomnography uses multiple different devices to measure different aspects of sleep and therefore can consume large amounts of power. Although the exact power usage of polysomnography is not defined, an attempt to reduce the amount of power to be used is still impactful. The current proposed solution only requires the power of a mobile device and computer to record and access data. It is important to note more testing is required to determine if results are the same quality as polysomnography. As discussed above the proposed solution was meant to be used at the patient's chosen location (most likely to be their home). This addresses an environmental problem indirectly. Since a patient does not need to travel to a facility, they will not need to use resources (gas/fossil fuels) which is important if they require multiple tests to be performed.

Financially, the focus was reducing the cost of the solution compared to polysomnography. Many people may decide to not get a diagnosis due to high costs. In countries with public healthcare some of this cost can be covered while in countries without public healthcare it could be entirely paid for by the patient. The proposed solution is designed to be an at home solution. This solution has costs associated with hosting a database and website, along with a mounting device. Therefore, the design decisions associated with these factors were compared to results observed by a study. This study found the average cost for a patient to do a lab test is $1 840 (95% CI $1 660, $2 015) USD and $1 575 (95% CI $1 439, $2123) USD [11]. These values were used as references in the decision-making process. Another side effect of performing tests at home is the reduction of costs used for transportation between home and a testing facility and was also taken into consideration.

Codes and Standards

Using Health Canada's medical device classification criteria, a conservative estimate for the overall system is a Class I medical device. According to Health Canada, a medical device is any

instrument or component used to treat, diagnose, or prevent a disease or abnormal physical condition [12]. The wearable sleep-analysis device and system are not intended to replace the expertise of a physician or respond in real time to patients, but rather assist physicians in the diagnostic process. To determine whether the system should be classified as a medical device, Health Canada's criteria for software as a medical device were consulted. Although most of the software exclusion criteria were met, the role of the system in contributing to diagnosis was deemed too significant to exclude it from being a medical device [13].

Before development began the two relevant standards were identified to help inform development and safety considerations. The first was ISO 14971 which addresses risk management and includes considerations for software as a medical device. The second was IEC 62304 which addresses safety particularly for medical device software. Given that the wearable device is low risk, the standards were not used extensively but rather to ensure certain risks were not overlooked. Such risks and potential hazards are addressed in the risk assessment section and hazard analysis in Appendix D.

Risk Assessment

In the early stages of development, a failure mode and effect analysis (FMEA) was performed and used as a guide to put in place measures to reduce risk. The FMEA can be found in Appendix D.

Device Risks:

As outlined in the FMEA about the recording device, the mains concerns are heat, electricity and radio waves (failure modes 1-3). Since the device will be operational recording and sending data over a prolonged period (6-8 hours) it can generate a large amount of heat. To mitigate burns or discomfort due to heat a mounting device with a pouch was selected for use. This pouch prevents the device from being in direct contact with the patient. The pouch also protects pets/sleeping partners as well as exterior surfaces from the heat as well. The pouch does also contain ventilation that allows for some of the heat to be dissipated which protects the device itself from overheating and causing damage to internal systems. With these modifications implemented the heat risk is decreased significantly however a warning about heat will need to be implemented in the mobile application to warn users. It would also be beneficial to implement an emergency shutdown if the device becomes too hot. With the current mounting system, the risks of electrical injury and RF waves were not mitigated other than warnings provided in the applications. A patient must consult their clinician before they can be warned about these risks and instructed on proper use. To address failure mode 4, a mount was selected with 2 plastic buckles and adjustable straps. Limiting the number of buckles reduces the number of pinch points on the device. These buckles are secure but are also not difficult to undo if pinching occurs. The straps being adjustable by the user will ensure they are comfortable before they go to sleep and not too restrictive. From initial tests of the device users found it to be comfortable but different designs can also be tested to find the optimal mounting solution.

Data Risks:

The risks that have been previously raised are false positives/false negatives (failure mode 6) and data breaches (failure mode 5). The current implementation has been developed to address these concerns, however more work is required to severely decrease these risks.

False positive and false negative results can be detrimental as they can harm the patient in terms of their health and financially as well. As result, while patients can see their raw data from the recording to ensure the recording was successful, an assessment is not presented to the patient from the mobile application. This prevents patients from acting based on the result given by the algorithms of the solution. A patient may attempt to interpret the data themselves and this cannot be prevented therefore a warning will be implemented into the application. The clinician can view the preliminary assessment in the web application; however, it will be mandated that they must interpret the data themselves and not base diagnosis solely on the result. To further reduce the risk of false positives and negatives more testing and analysis would be required to find parameters to reach a desirable success rate. If a high success rate is required a clinician will still be required to fully validate this assessment.

Information that is stored by the current implementation for users includes name, photo, email, phone number and birthday. Clinicians also have the address of their clinic and records of appointments between patients stored. The raw data and data analysis results are also kept in the database. To reduce the risk of this data being leaked security measures were put in place to increase security of the mobile and web application. Both applications require username and password access. The current implementation does not enforce restrictions on the password such as the use of capitalization or symbols and has no multi-factor authentication. Passwords do not expire after a period. These measures could be implemented to further reduce the risk of unwanted users having access to data. Also, included with the current implementation, a patient must be registered by their clinician during an appointment to begin the use of the device. Therefore, the clinician can instruct patients about the necessity to record and send data on secure networks to prevent data from their information being intercepted. Lastly, while not yet implemented the database does not delete user data as planned to increase security. To implement this feature a doctor will need to designate data (users, appointments, recording, etc.) that they will not use and after a short period of time will be deleted.

Next Steps:

This capstone project is still in its infancy. There are several more steps required before this can become a standalone and marketable product. Firstly, the development is not entirely complete. The next step would be to acquire audio information as the person sleeps and feed that data along with the positional data being acquired into a Deep Learning Network with the purpose of identifying sounds as the person sleeps that could be indicative of apnea (snoring, farting, coughing). This would help the next team in directly diagnosing sleep apnea. Secondly, we would need to migrate our entire infrastructure to a particular cloud provider (Azure or AWS) and leverage their ecosystem to deploy the product live. They would handle the load management, traffic management, down-time management. Aside from development, we would need to develop integration tests to ensure the components can function alone to a degree where it would still be user-friendly (not crash, lag, time-out).

Works Cited

[1] S. L. Worley, "The extraordinary importance of sleep," *Pharmacol Ther*, vol. 43, no. 12, 2018.

[2] S. Javaheri *et al.*, "Sleep Apnea: Types, Mechanisms, and Clinical Cardiovascular Consequences," *Journal of the American College of Cardiology*, vol. 69, no. 7. 2017. doi: 10.1016/j.jacc.2016.11.069.

[3] Motamedi KK, McClary AC, Amedee RG. Obstructive sleep apnea: a growing problem. Ochsner J. 2009 Fall;9(3):149-53. PMID: 21603432; PMCID: PMC3096276.

[4] L. C. Markun and A. Sampat, "Clinician-Focused Overview and Developments in Polysomnography," *Current Sleep Medicine Reports*, vol. 6, no. 4. 2020. doi: 10.1007/s40675-020-00197-

[5] I. M. Rosen *et al.*, "Clinical use of a home sleep apnea test: An updated American academy of sleep medicine position statement," *Journal of Clinical Sleep Medicine*, vol. 14, no. 12, 2018, doi: 10.5664/jcsm.7540.

[6] B. A. Edwards, A. Wellman, and R. L. Owens, "PSGs: More than just the AHI," *Journal of Clinical Sleep Medicine*, vol. 9, no. 6. 2013. doi: 10.5664/jcsm.2738.

[7] R. D. Chervin, D. L. Murman, B. A. Malow, and V. Totten, "Cost-utility of three approaches to the diagnosis of sleep apnea: Polysomnography, home testing, and empirical therapy," *Ann Intern Med*, vol. 130, no. 6, 1999, doi: 10.7326/0003-4819-130-6-199903160-00006.

[8] C. Gong, "Human-computer interaction: The usability test methods and design principles in the human-computer interface design," in *Proceedings - 2009 2nd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2009*, 2009. doi: 10.1109/ICCSIT.2009.5234724.

[9] G. Chao, "Human-computer interaction: Process and principles of human-computer interface design," in *Proceedings - 2009 International Conference on Computer and Automation Engineering, ICCAE 2009*, 2009. doi: 10.1109/ICCAE.2009.23.

[10] Chourpiliadis C, Bhardwaj A. Physiology, Respiratory Rate. [Updated 2022 Sep 12]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2022 Jan-. Available from: https://www.ncbi.nlm.nih.gov/books/NBK537306/

[11] R. D. Kim *et al.*, "An economic evaluation of home versus laboratory-based diagnosis of obstructive sleep apnea," *Sleep*, vol. 38, no. 7, 2015, doi: 10.5665/sleep.4804.

[12] "About Medical Devices," Canada.ca, 27-Jan-2020. [Online]. Available: https://www.canada.ca/en/health-canada/services/drugs-health-products/medical-devices/about-medical-devices.html. [Accessed: 08-Dec-2022].

[13] "Guidance Document: Software as a Medical Device (SaMD): Definition and Classification," Canada.ca, 18-Jun-2020. [Online]. Available: https://www.canada.ca/en/health-canada/services/drugs-health-products/medical-

devices/application-information/guidance-documents/software-medical-device-guidance-document.html. [Accessed: 08-Dec-2022].

Appendices:

Appendix A - Final Design:

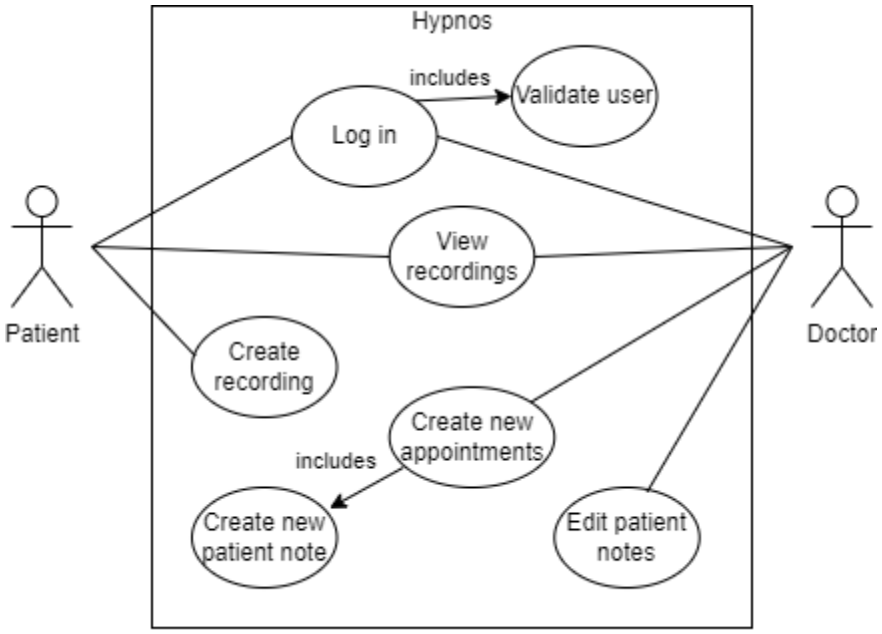


Figure 5: Use case diagram of entire Hypnos system.

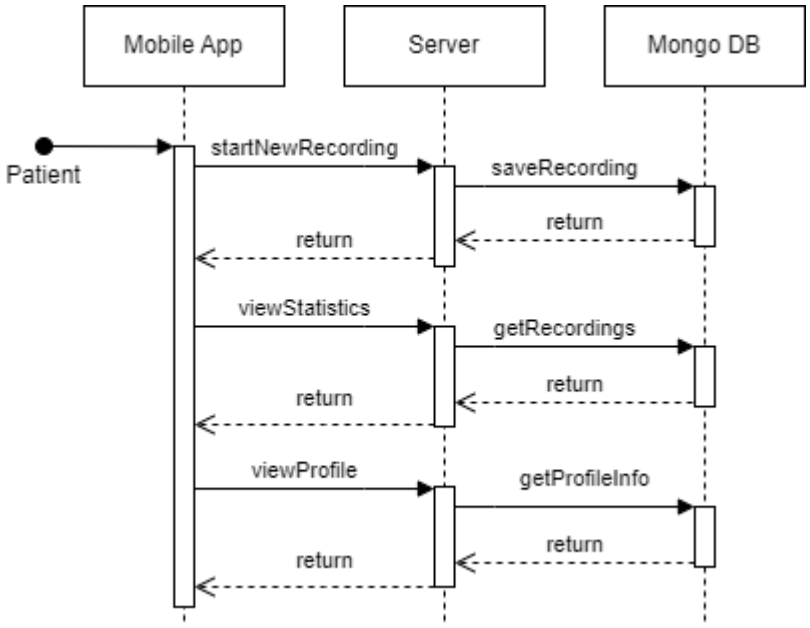


Figure 6: Sequence diagram outlining patient interaction with the system (mobile app).

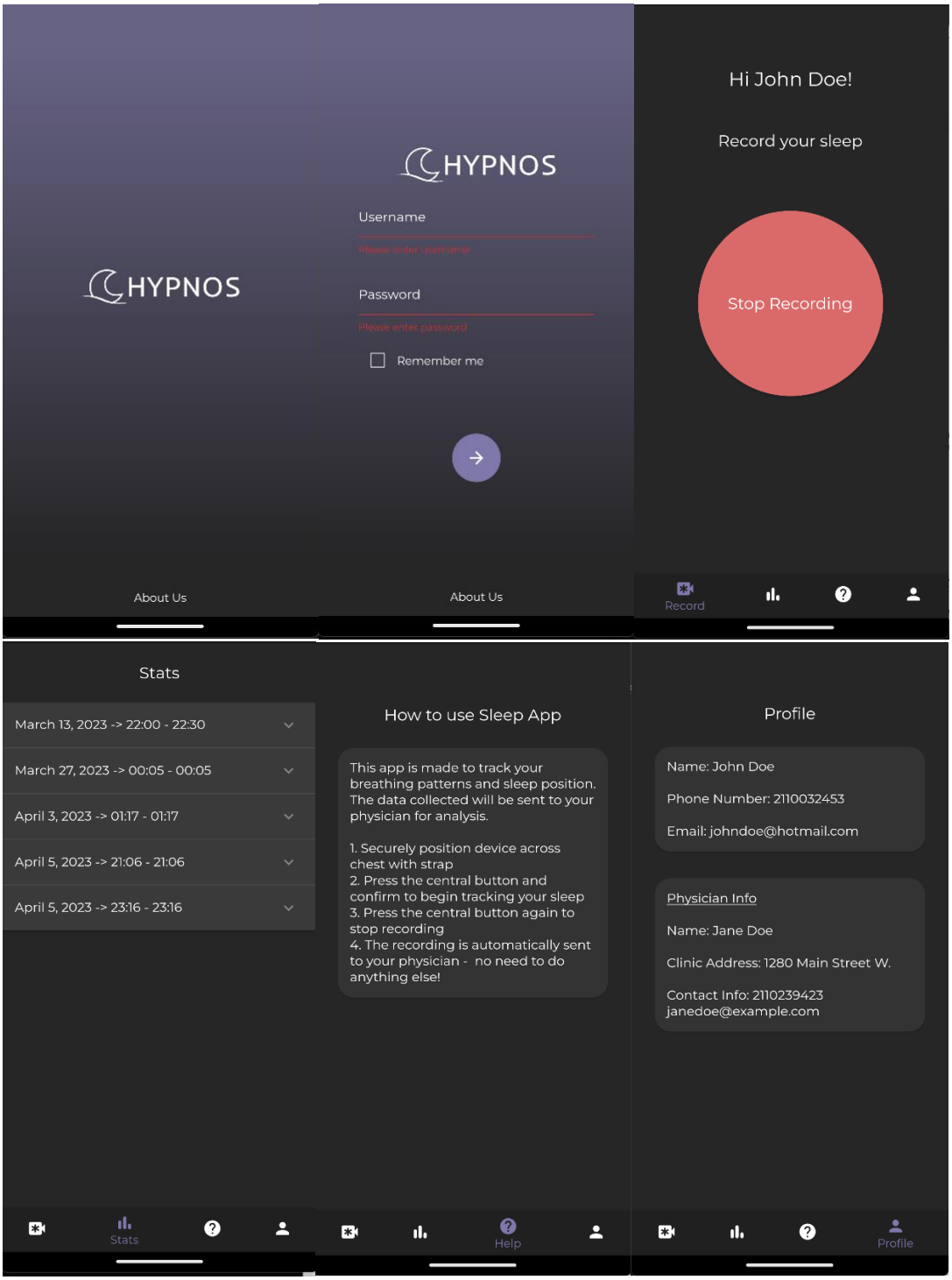


Figure 7: Smartphone app screens - title, invalid login, stop recording, statistics, help, profile (from left to right).

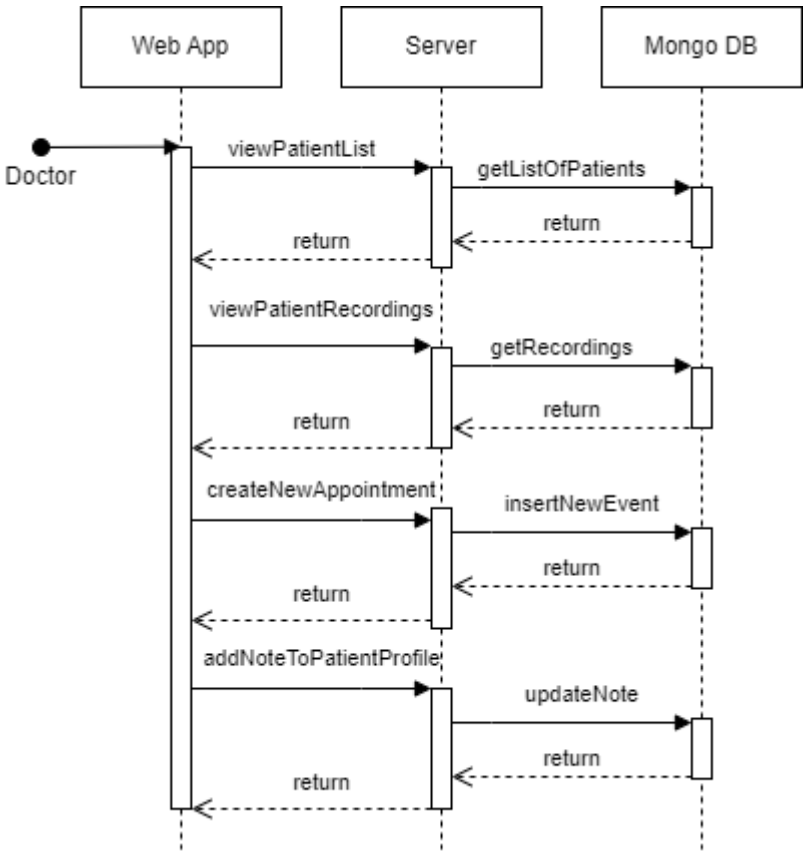


Figure 8: Sequence diagram outlining doctor interactions with the system (web app).

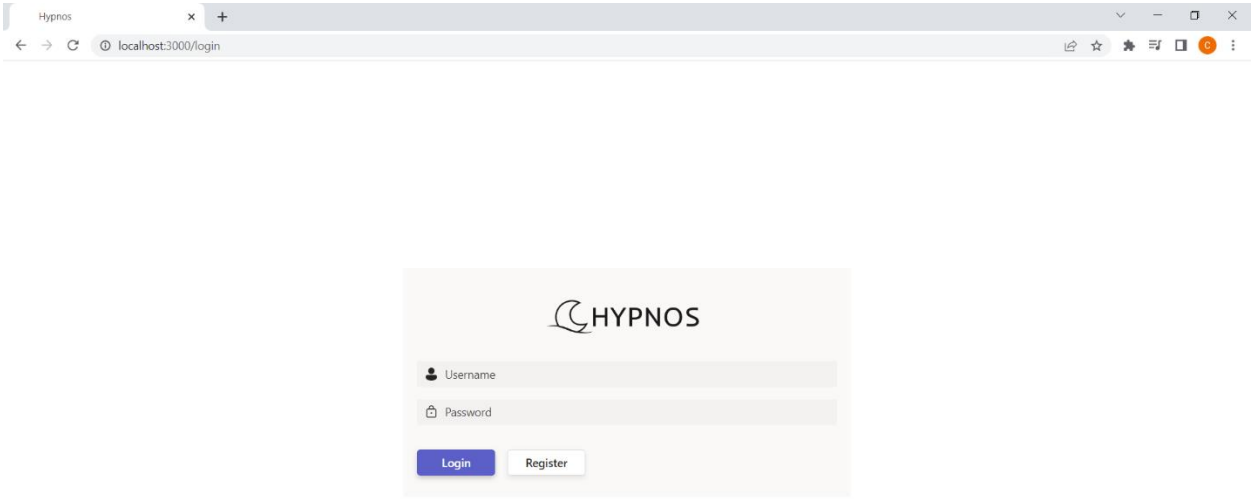


Figure 9: Final Login screen.

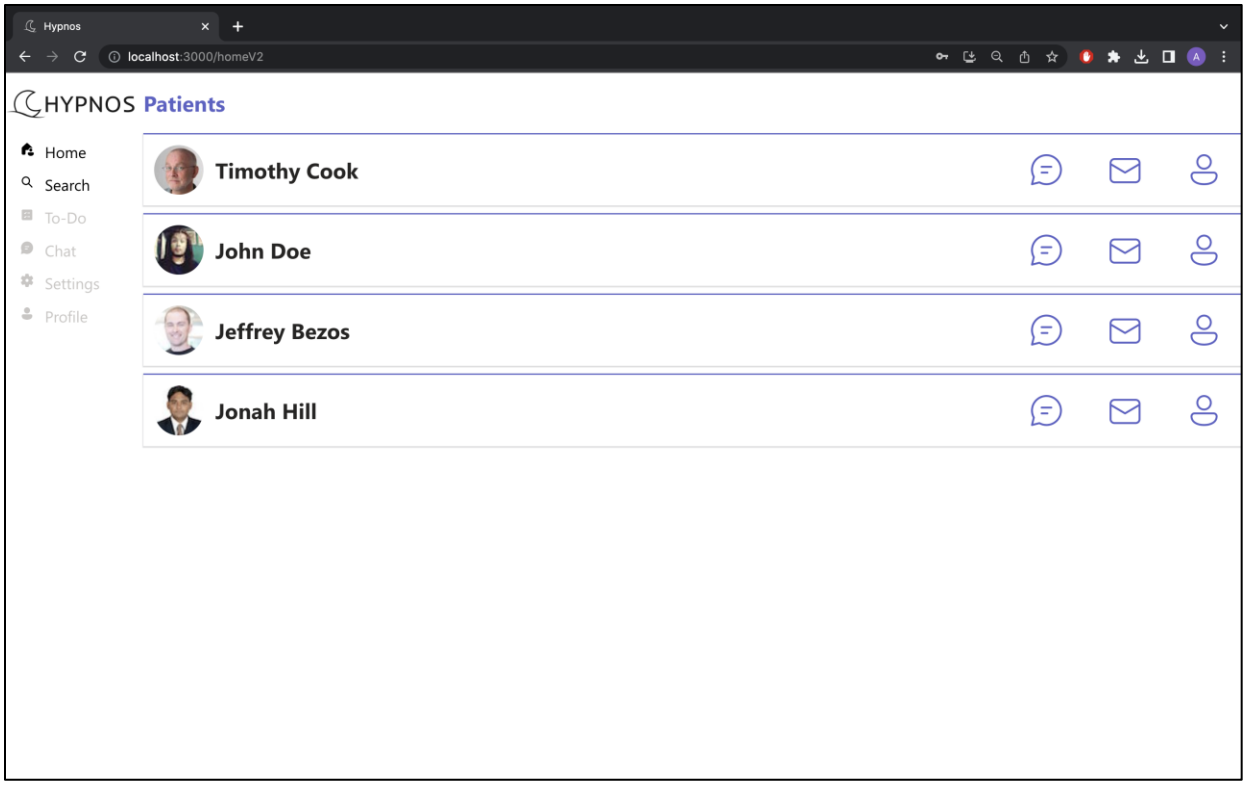


Figure 10: Final Patient List screen.

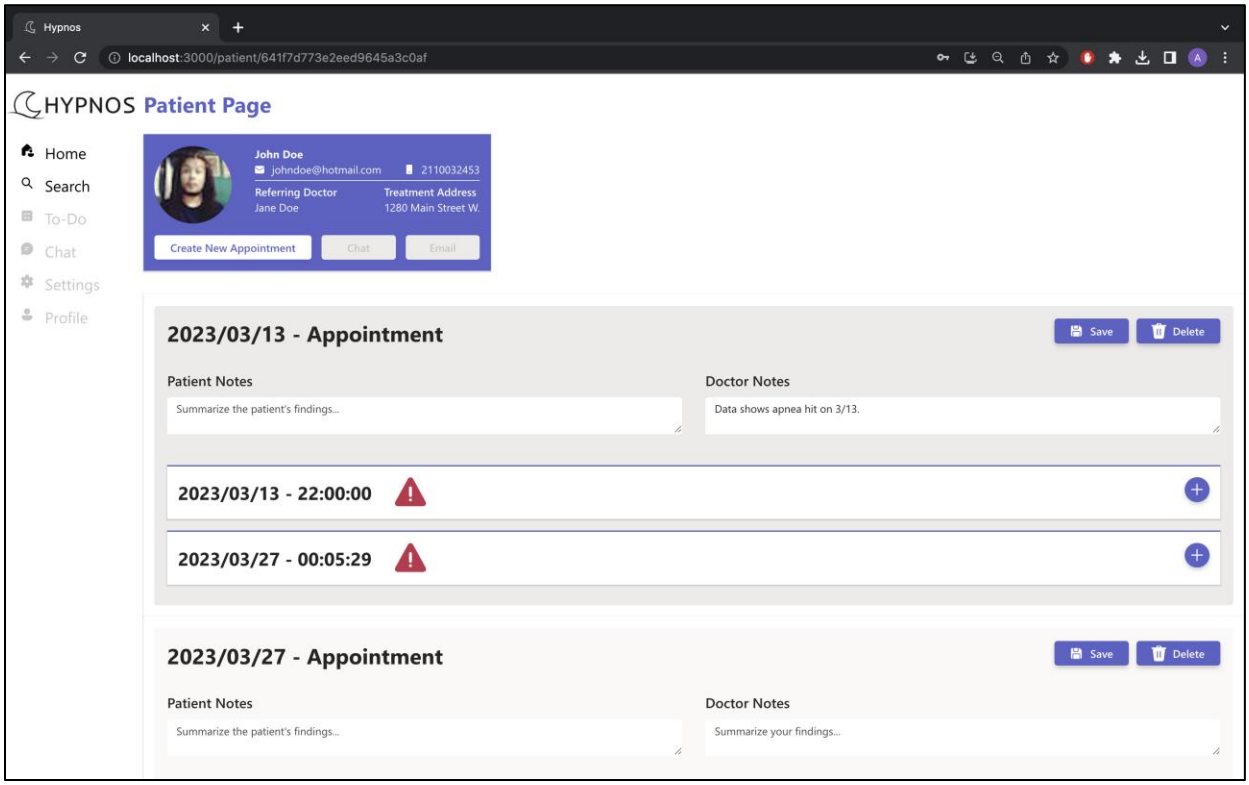


Figure 11: Final Patient Data screen.

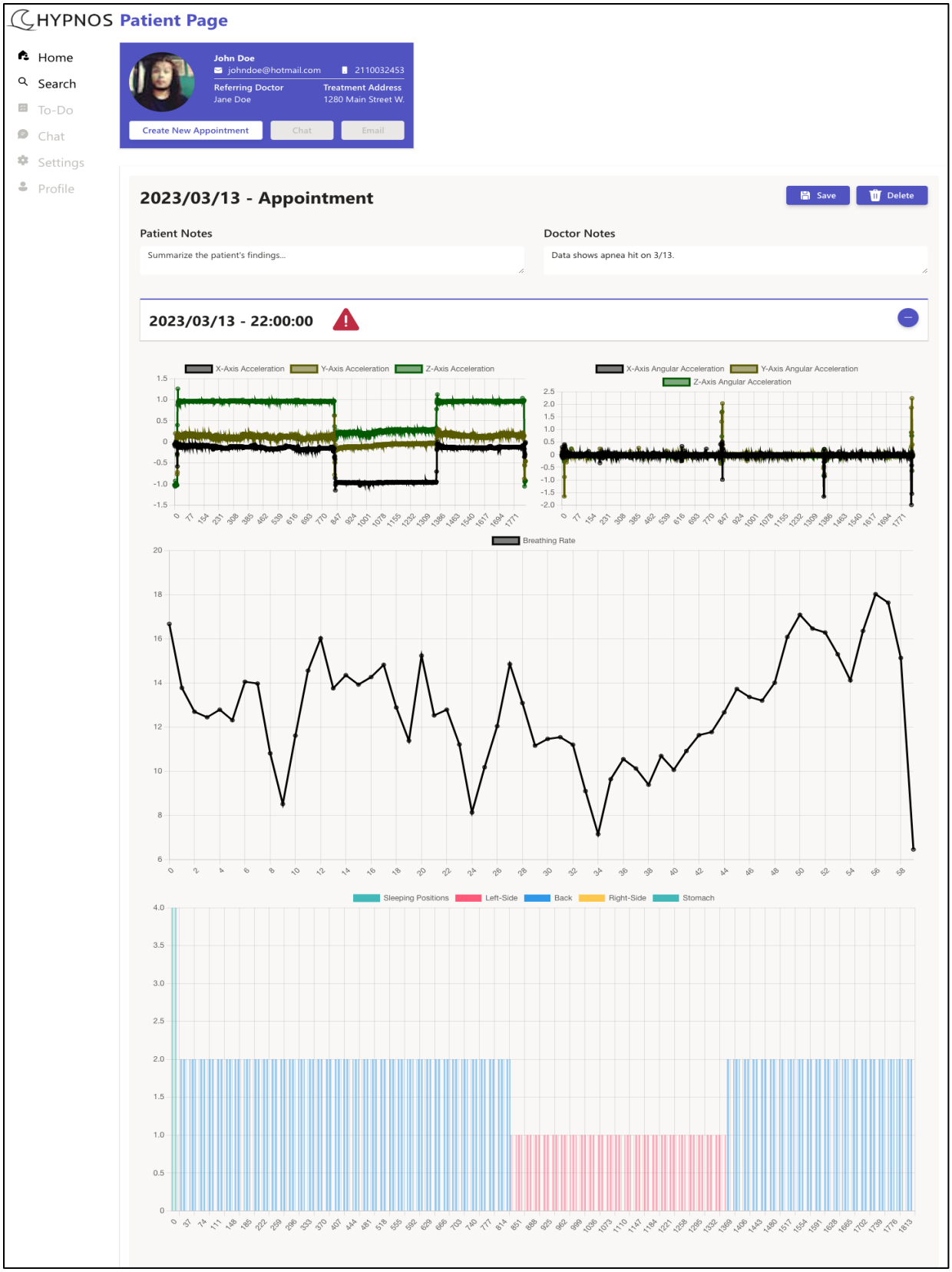


Figure 12: Final Patient Data screen expanded.

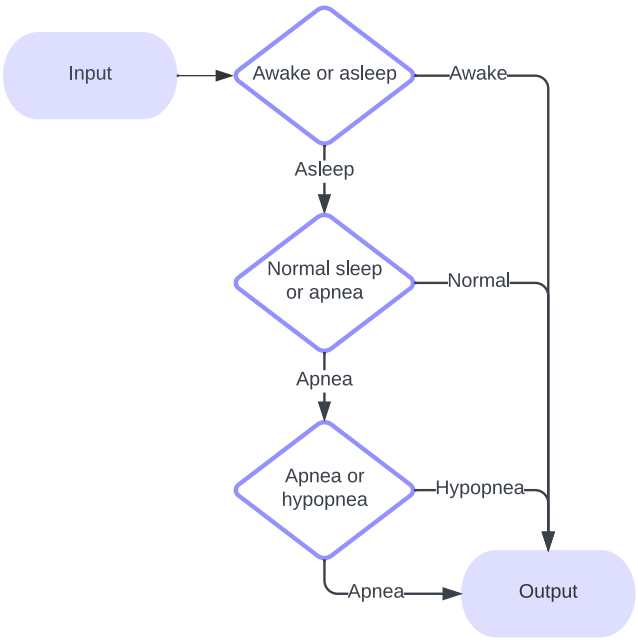


Figure 13: Binary classification algorithm.

Figure 14: Mounting system with recording device.

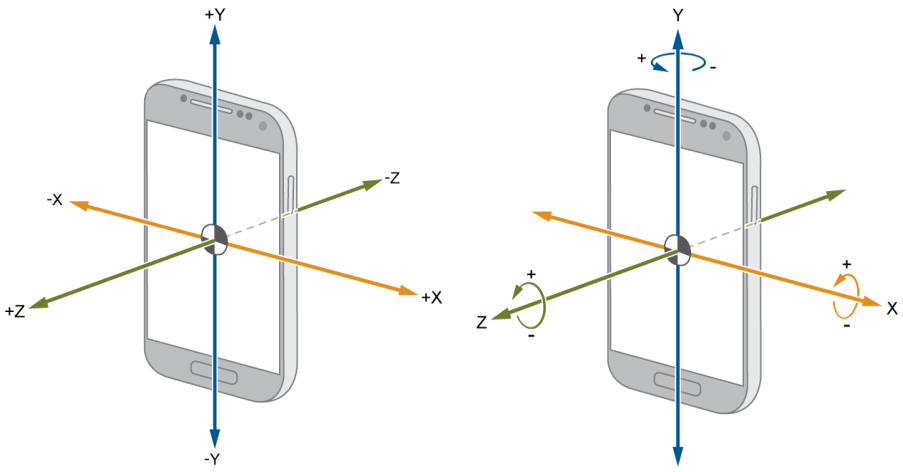


Figure 15: Directional axes for recording devices.

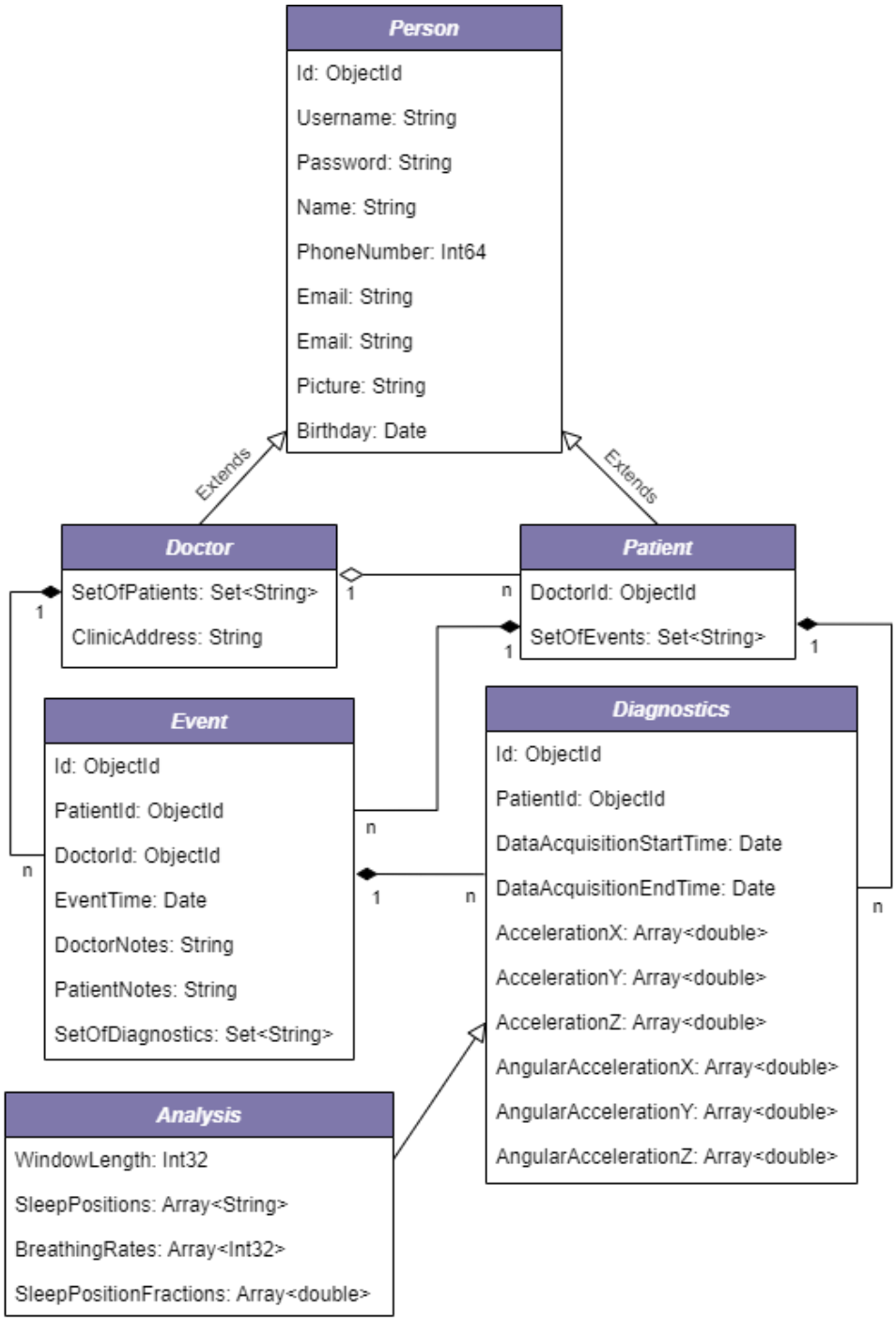


Figure 16: Database relationship diagram.

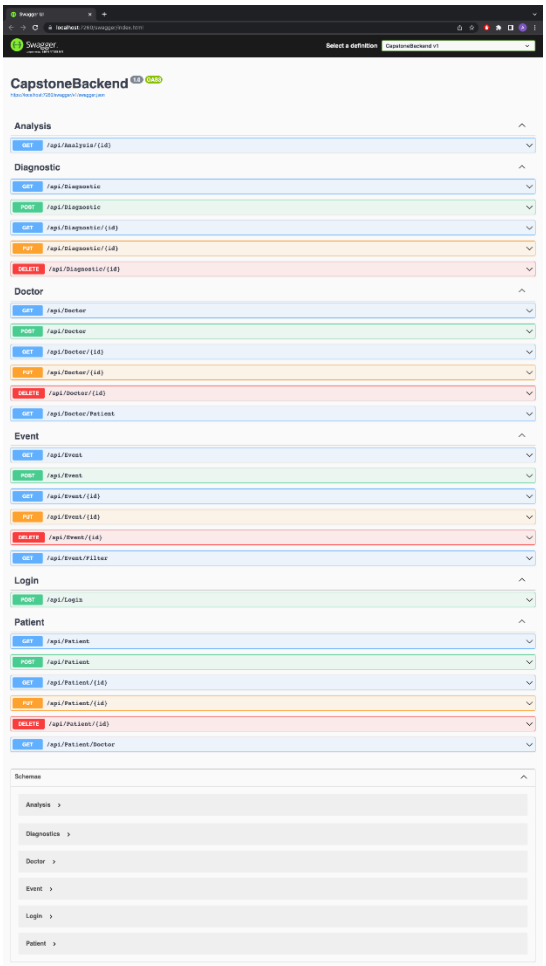


Figure 17: Swagger Documentation for Backend Server Endpoints

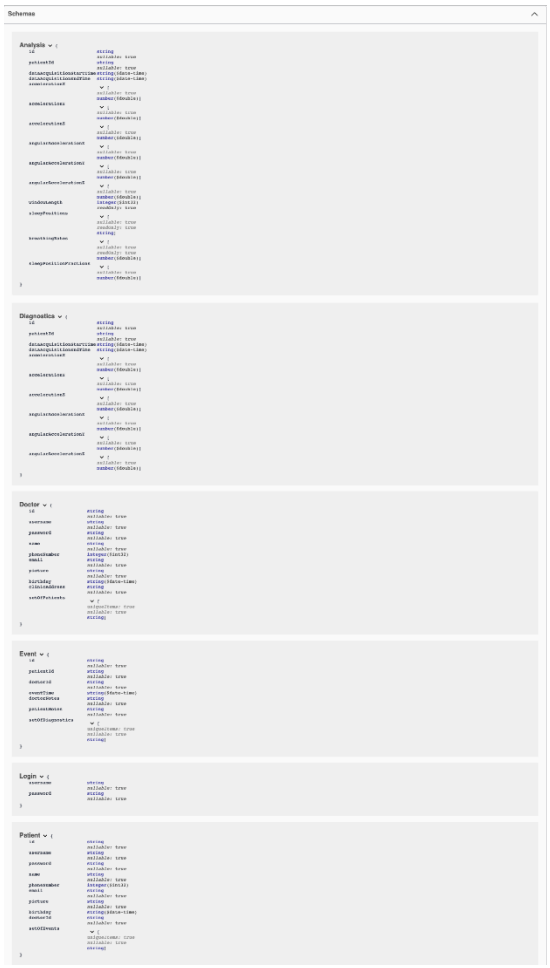


Figure 18: Swagger Documentation for Backend Server Schemas

Appendix B - Analysis

Sample Data graphs:

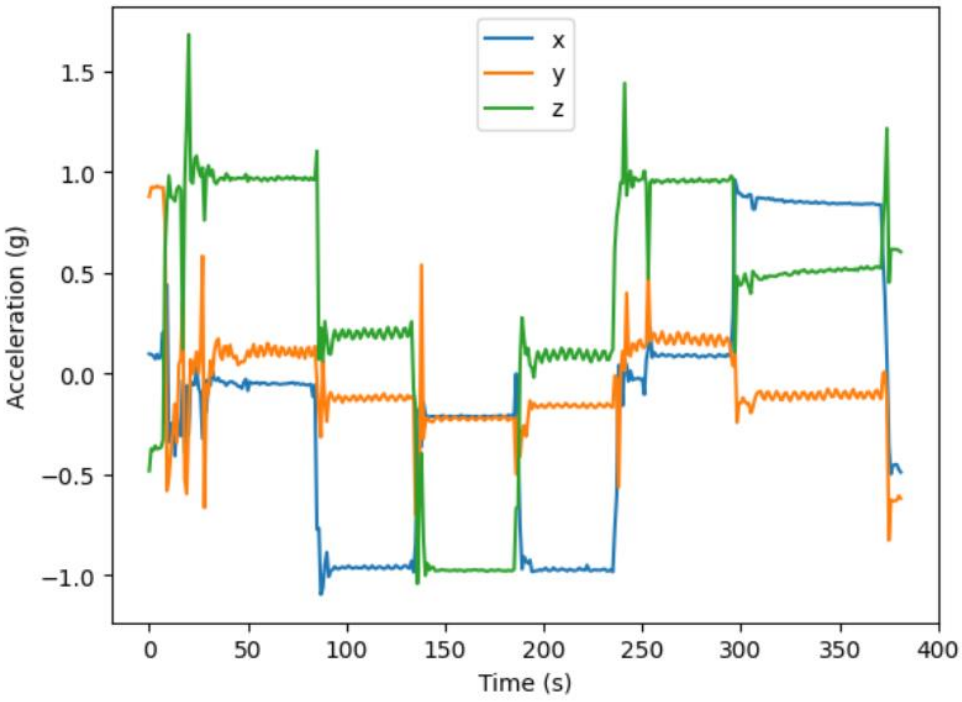


Figure 19: Linear acceleration in three dimensions.

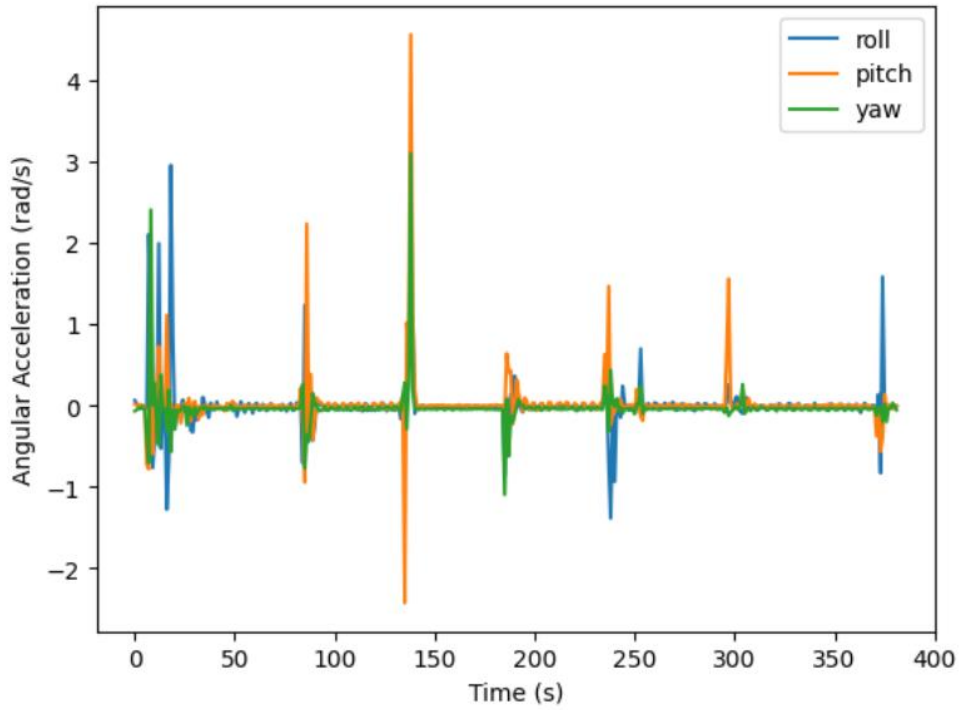


Figure 20: Angular acceleration in three dimensions.

Sample Analysis Output graphs:

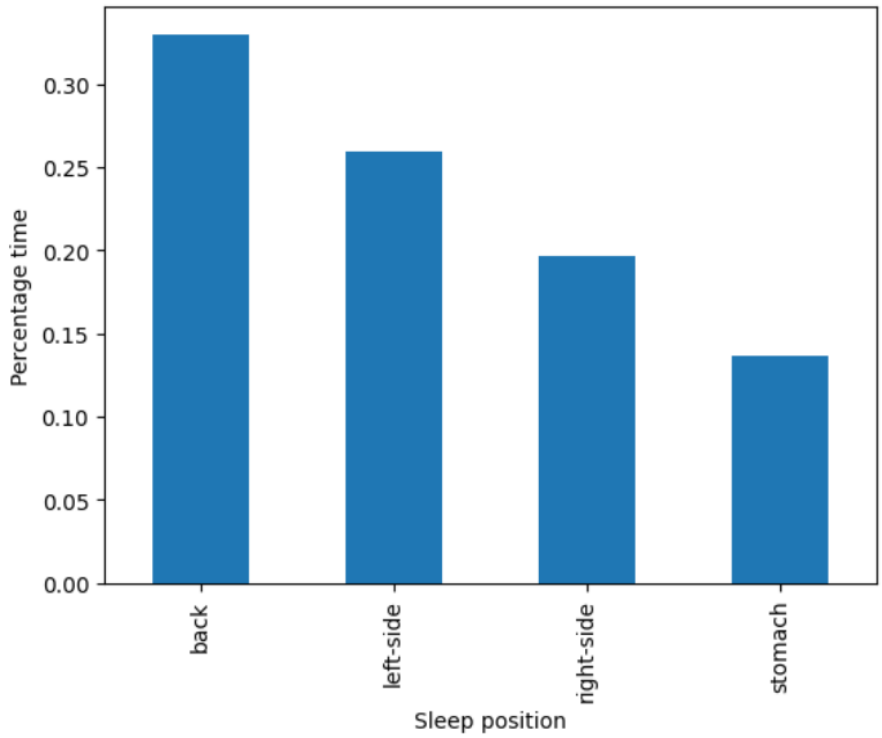


Figure 21: Sleep position distribution

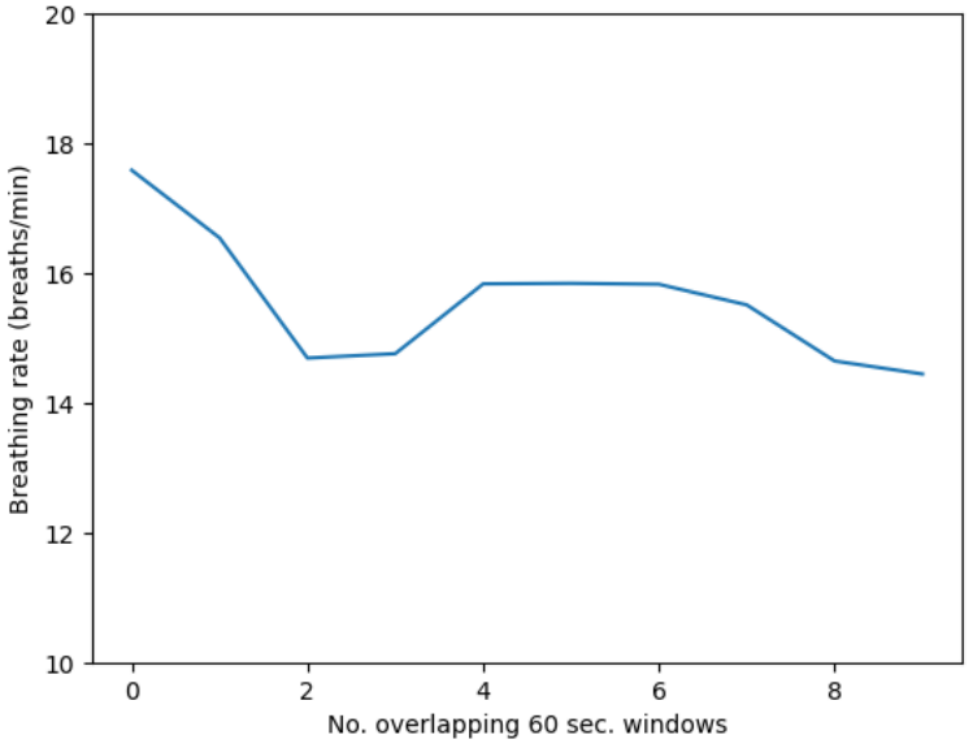


Figure 22: Breathing rate over 60-second averaging windows

Analysis Algorithm Flowcharts

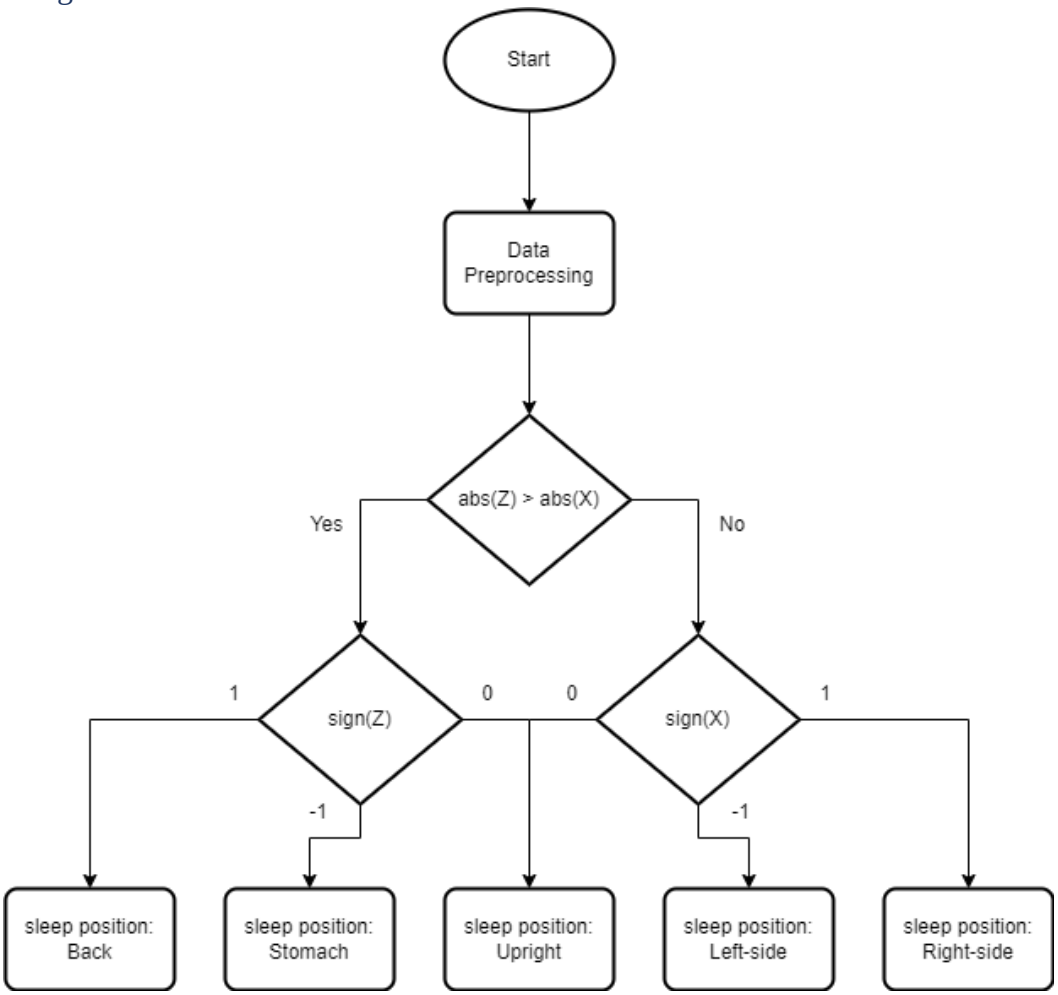


Figure 23: Sleep position identification flowchart

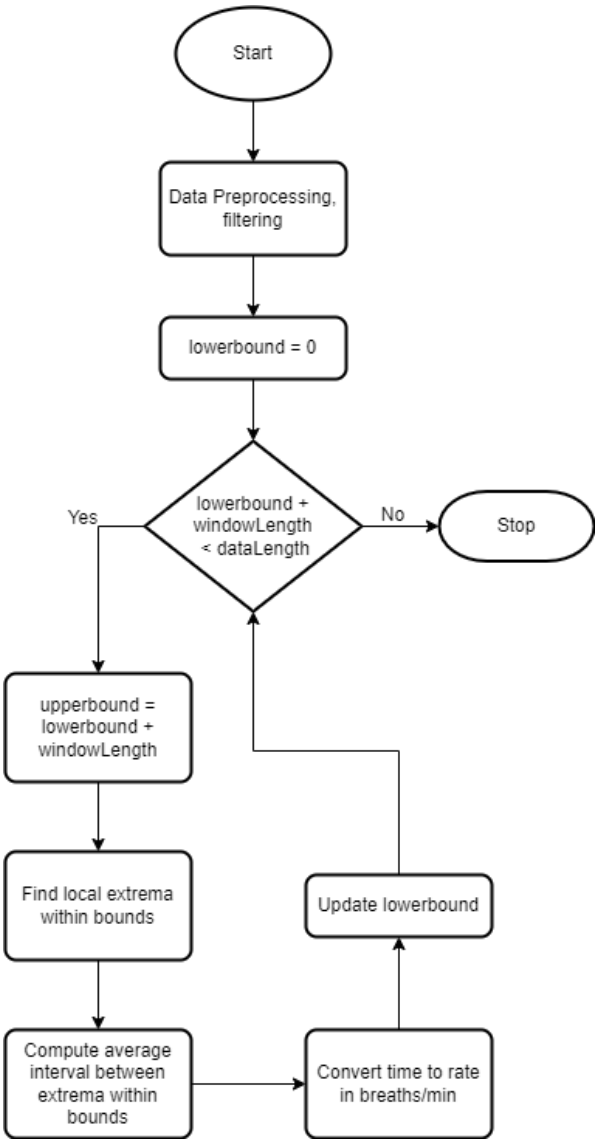


Figure 24: Breathing rate estimation flowchart

Database Storage

Diagnostics Schema is 1.3MB {

Id: 12 Bytes

PatientId: 12 Bytes

DataAcquisitionStartTime: 12 Bytes (DateTime is 8 bytes + 4 bytes for overhead)

DataAcquistionEndTime: 12 Bytes

AccelerationX: 230408 bytes (8 hours * 60 minutes/hour * 60 seconds/ minute * 1 point/ second * 8 bytes/ point)

AccelerationY: 230408 bytes

AccelerationZ: 230408 bytes

AngularAccelerationX: 230408 bytes

AngularAccelerationY: 230408 bytes

AngularAccelerationZ: 230408 bytes

}

Sampling Frequency Calculations & Justification:

Assume maximum breathing rate of 20 breaths per minute.

$$f_{breath} = 20\ breaths/min$$

$$f_{breath} = 0.33\ Hz$$

Using Nyquist-Shannon theorem, the Nyquist frequency must be greater than the highest frequency in the signal being sampled and half of the sampling frequency.

$$f_{Nyquist} > f_{breath}, f_{Nyquist} = 0.5 f_{sampling}$$

Given that the maximum breathing rate is 0.33 Hz, a possible Nyquist frequency is 0.4 Hz. Thus, the sampling frequency would be 0.8 Hz.

$$f_{sampling} = 2 f_{Nyquist}$$

$$f_{sampling} = 2(0.4) = 0.8\ Hz$$

The sampling frequency was set to 1 Hz as it meets the requirements, based on the Nyquist-Shannon theorem, to avoid aliasing and translates well to data interpretation and visualization.

Sample Analysis Verification Results:

Sleeping position:

Defined Sequence: supine/back (0), left-side (3), prone/stomach (1), left-side (3), supine (0), right-side (2)

Output Sequence: supine/back (0), left-side (3), prone/stomach (1), left-side (3), supine (0), right-side (2)

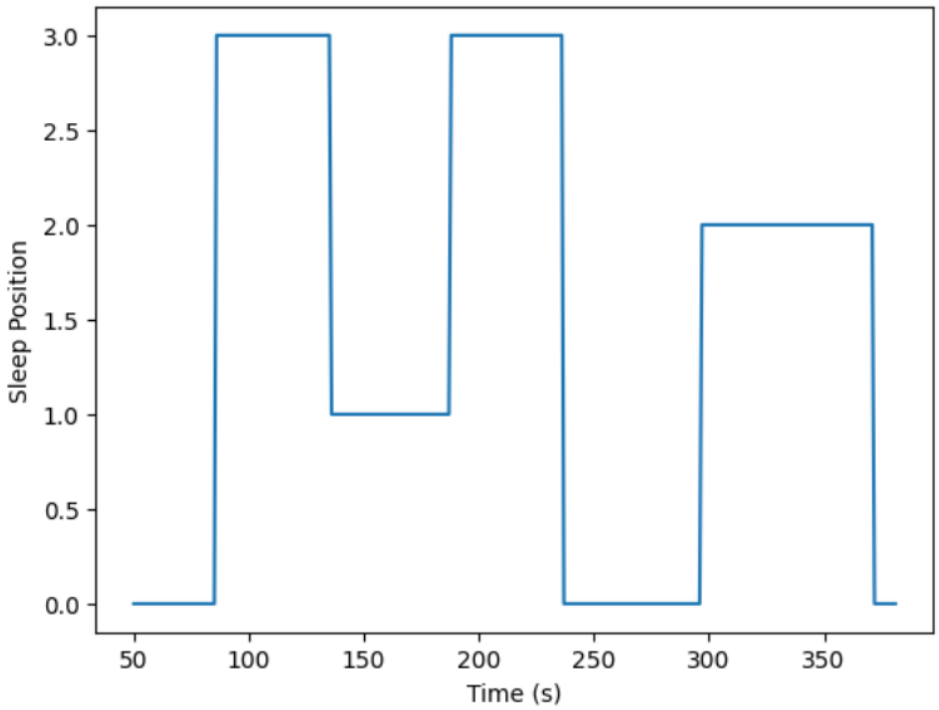


Figure 25: Sample sleeping position graph.

Breathing rate:

Table 1: Breathing rate test results.

Estimate No.	Position (maintained for 2 mins)	True breath rate (breaths/min)	Output breath rate (breaths/min)	Absolute error (breaths/min)	Relative error
1	Supine	10	11.74	1.74	0.174
2		10	10	0.0	0.0
3		10	10	0.0	0.0
1	Supine	15	15	0.0	0.0
2		15	15	0.0	0.0
3		15	14	1.0	0.067
1	Left-side	10	9.64	0.36	0.036
2		10	9.64	0.36	0.036
3		10	9	1.0	0.1
1	Left-side	15	15	0.0	0.0
2		15	15	0.0	0.0
3		15	15	0.0	0.0

Mean squared error:

$$MSE = \sum_{i=1}^{n} \frac{(y_i - \hat{y}_i)^2}{n}$$

where y_i is the true rate and $\widehat{y_i}$ is the output rate.

The resulting MSE is 0.44 breaths^2/min^2.

Absolute error:

$$E = |y_i - \widehat{y_i}|$$

Relative error:

$$RE = \frac{|y_i - \widehat{y_i}|}{y_i}$$

Average relative error: 0.034

Given the expected range of breathing rate of 12-20 breaths/min the average absolute error is expected to range between 0.41 to 0.69 breaths/min.

Appendix C - Prototypes and Testing

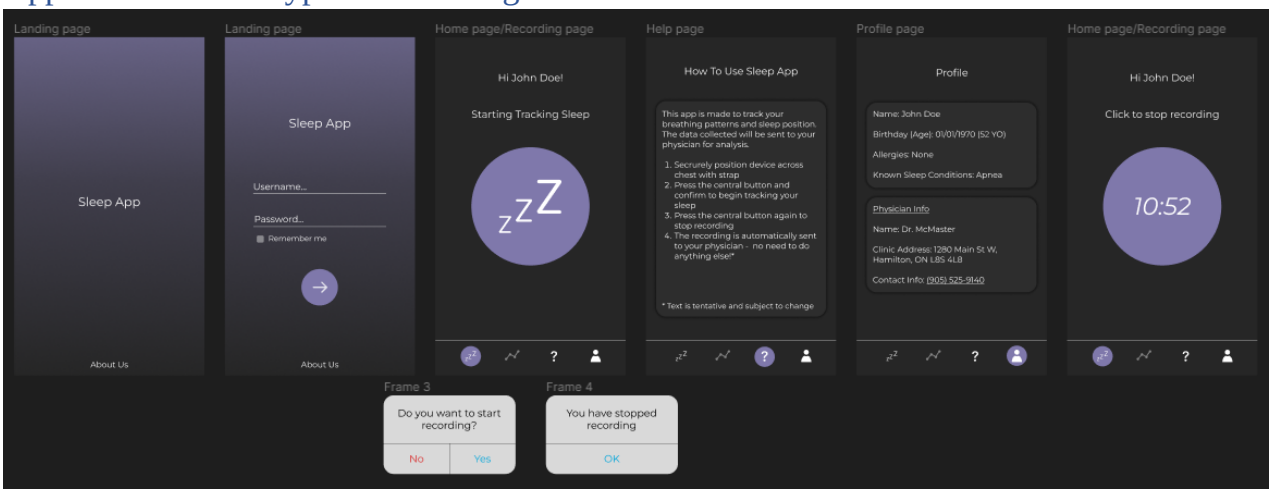
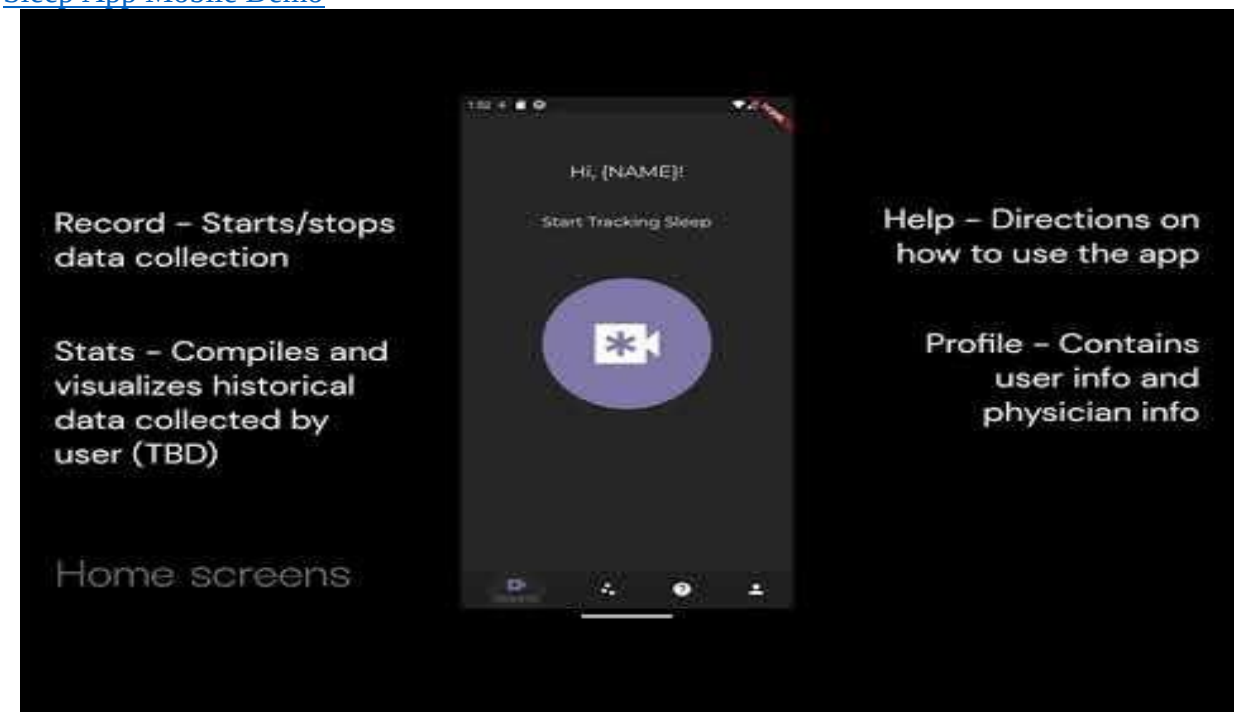


Figure 26: Figma sketch of mobile app; First iteration of mobile app.

[Sleep App Mobile Demo]



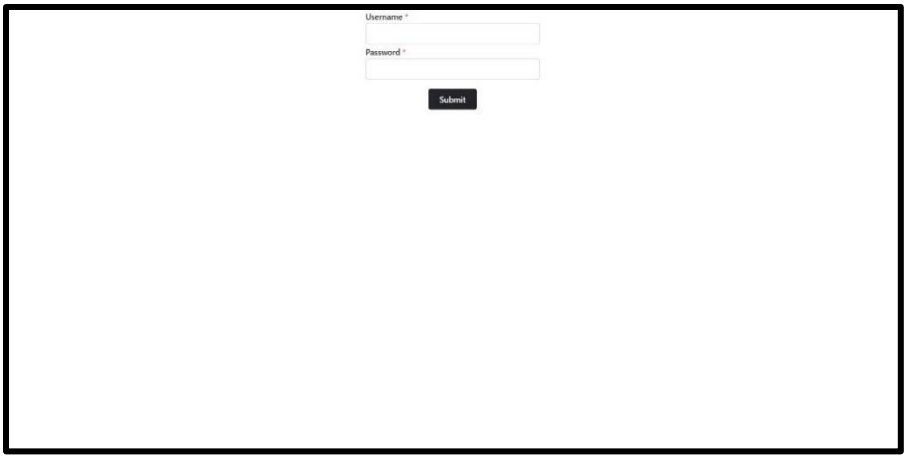


Figure 27: Prototype Login Screen.

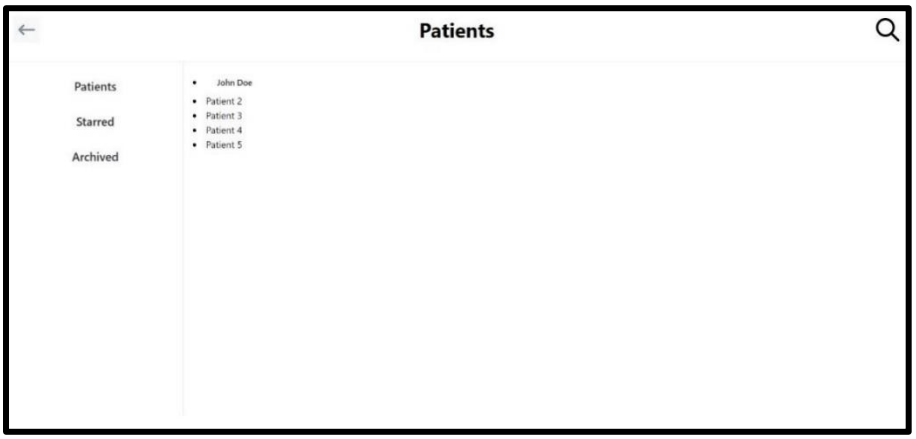


Figure 28: Prototype Patient List Screen.

Figure 29: Prototype Patient Data Screen.

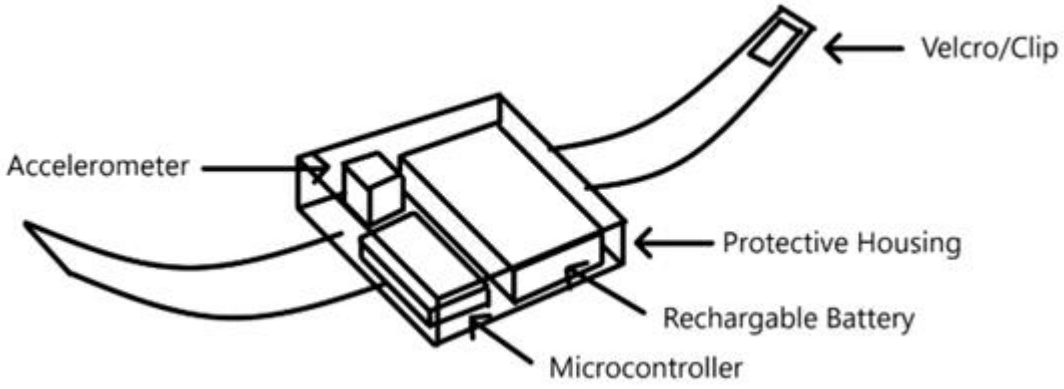


Figure 30: Sketch of custom wearable sleep analysis device.

Table 2: Mobile usability testing results.

Participant	Time to learn (m:ss.ms)	Initial Rating (/10)	Final Rating (/10)	Feedback
1	1:24.91	9	10	Stop recording button hard to distinguish between start.
2	1:30.11	8	9	Landing page should be touch anywhere to continue. Scale graph based on length of recording.
3	0:55.33	8.5	9	Stats icon is not indicative of what the page does. Stats heading confusing with all the numbers, look into changing date to text.
4	0:58.52	9	9.5	Graphs may be confusing for user as the acceleration and gyroscope data is a bit abstract.
5	1:28.25	8	8.5	The delay between opening the stats page and when it loads is quite long; feels like app crashed.
Average	1:15.42	8.5	9.2	

Table 3: Web usability testing results.

Participant	Time to learn (m:ss.ms)	Initial Rating (/10)	Final Rating (/10)	Feedback
1	1:05.84	8	9	Login section could be larger and an option to save login credentials would be convenient.
2	1:10.67	7.5	9	Unclear how to open-up graphs. Enable zooming and panning on the graph; more precision needed.
3	0:30.14	9	9.5	Symbols are very logical and easy to navigate around. Theme could be more consistent.
4	0:47.66	8	8.5	Went to the wrong page, wasn't clear on where to go. Include a quick start guide.
5	0:38.24	8.5	10	Very easy to read and see everything. Clean, simple theme.
Average	0:50.51	8.2	9.3	

Appendix D - Failure Mode Effect Analysis

Table 4: Failure Mode Effect Analysis (FMEA Table).

Design Function	Failure Modes	Effects of Failure	Causes of Failure	Recommended Action
Record information over a long period of time while attached to the user.	1. Recording device can become very hot.	Burns, injuries, discomfort and negatively affect sleep.	a. Device has a lot of processes occurring for long periods of time. b. Insufficient ventilation to allow heat to dissipate.	a. Provide extra insulation. b. Detect operating temperatures for shutoff. c. Provide documentation and warnings. a. Provide safer methods for charging while in use such as storage for a battery pack.
	2. Device needs to be charged while in use.	Electrocution, fire hazard, strangulation due to wiring.	a. Charging cord can constrict due to movement while asleep. b. Charging cord is damaged/becomes damaged.	b. Provide documentation and warnings about hazards. a. Provide warnings about use with other devices and ensure clinicians are aware of potential use.
	3. Interferes with other medical devices.	Failure of other medical devices such as pacemakers.	a. Frequencies of recording devices are close in proximity to other medical devices.	a. Use securing methods that reduce the risk of being pinched. b. Provide extra padding for comfort. c. Provide instructions on how the device should be mounted.
	4. Mounting device is improperly fit for the user.	Pinching, restriction of blood flow and/or respiration. Negatively affects sleep.	a. Clips or other parts used to secure the mount can be shut on user. b. Mount is secured too tightly on the user.	
Store and transfer patient data safely	5. Data is compromised by malicious attack.	Attackers have obtained personal data about users.	a. Unsecured network used for data transfer. b. Database security is breached	a. Encourage users to only use secure networks. b. Encourage users to change login credentials often. c. Delete data that has not been accessed for a long period of time.
Provide preliminary assessment of sleep conditions.	6. False positives and false negative assessments are made.	Patients will use preliminary assessment to seek an inappropriate treatment plan.	a. Improper data acquisition. b. Assessment methods are inappropriate	a. Always require clinician approval for assessments. b. Train system intensely and reduce error.