Faculty of Engineering and Applied Science

ENGR 4941U Capstone Systems Design for ECSE II

Remote and Mobile Healthcare System for Home Care

R4: Acceptance Testing and Results Report

Team Members:
Mamun Hossain, 100553073
Gobikah Balaruban, 100742539
Manreet Kaur, 100766207
Aaditya Rajput, 100622434

Faculty Advisor: Dr. Mikael Eklund
Capstone Coordinator (Fall 2022 and Winter 2023): Dr. Vijay Sood

Abstract

Our project's goal is to create a remote mobile healthcare system whilst incorporating different IoT aspects to further benefit the world of remote mobile healthcare. However, in order to do this there are many limitations that have been imposed, such as technological limitations, server limitations, and security protocols that may be in place. In order to combat these challenges, our group will be creating a mobile healthcare unit using an Arduino Mega 2560 that will be connecting to a cloud hosting service which is Digital Ocean, whilst it containing Apache scripts for web pages, PHP scripts to enable the web pages to be usable for the mobile application, and MySQL acting as a database to store all the information. We will also be using a SEN-11574 for PPG ratings, a MAX30102 biosensor for blood oxygen levels, and a 3 lead system being controlled by an AD8232 to monitor ECG signals.

This report focuses on our project design and the changes we have made from the first semester leading into this second semester and the necessary system developments alongside testing that we had to make along the way. The product initially had used an Arduino UNO to control all the hardware inputs that was connected in tandem with a Raspberry Pi Model 3 to enable a web server within the cloud limitation of a Raspberry Pi, however due to hardware limitations we had come to a conclusion that it would not be an efficient solution to the cloud server as to what we had initially thought.

To ensure the success of our project, alongside with the tests in this report, we will be conducting several different types of tests, including unit tests, integration tests and UI tests, alongside having a detailed design indicating the process our system will undergo. This will allow us to be able to identify and fix any major issues that will occur before delivering the final prototype.

Acknowledgements

We would like to express our sincere gratitude to our Capstone Coordinator and Capstone Professor for their unwavering support, guidance, and encouragement throughout our project. Their expertise and feedback have been invaluable to our success.

Our Capstone Professor, Dr. Mikael Eklund, provided us with insightful advice, and expert knowledge, and helped us to remain on track with our objectives. His mentorship and regular feedback have been invaluable to the success of our project.

We also extend our heartfelt appreciation to our Capstone Coordinator, Dr. Vijay Sood, for his unwavering support and guidance throughout the project. He is always available to offer assistance and support in all areas of the project, and we are grateful for his dedication and commitment to our success.

We would also like to extend our thanks to our colleagues and fellow students who provided valuable feedback and support during the development of our project. Their contributions and insights helped us to refine our ideas and develop a more robust and effective system.

Finally, we would like to thank our families and loved ones for their unwavering support and encouragement throughout our academic journey. Their love and support have been a constant source of strength and inspiration, and we are deeply grateful for everything that they have done for us.

Table of Contents

List of Tables

1 Introduction

Healthcare is a crucial part of our everyday lives, and with the increasing use of technology, healthcare services are becoming more and more important as each day passes, especially through the usage of mobile applications. Because of the rapid advancements in technology, it has become possible to connect software components to medical grade hardware components and develop different types of healthcare systems. In this regard, the aim of this project will be to create a reliable and efficient healthcare system that is user friendly and highly available. This report will contain the system design and unit testing process that will allow us to further our understanding in this regard.

1.2 Problem Statement

The problem that we have is that we are trying to create a mobile healthcare system with the integration of our Arduino Mega system, where we will be using DigitalOcean as the cloud hosting platform, Apache to run the webpages, PHP to communicate with the webpages and MySQL to hold receive the data from the Arduino and redistribute that data to the mobile application. The goal is to develop a highly efficient and robust system that will allow users to access healthcare services on their mobile devices. This report should provide comprehensive details on the system design and the implementation, including system architecture and unit testing results.

1.3 Overview of the Report

The remainder of this report is organized as follows. Section 2 presents the Acceptability Planning, Section 3 presents the Acceptability tests, Section 4 showcases the test results, and finally issues and defects are presented in Section 5, and contributions in Section 6.

2 Acceptability Planning

The acceptance test will focus on meeting non-functional as well as the functional requirements that were not tested, or unable to have been tested in the integration testing. The acceptance criteria will be determined based on the use cases as well as how the system is used and if it is being utilized to the functions required.

There will be no more hardware required for the acceptance testing aside from the hardware that is already currently defined within our project. There will be PHP code applied to connect the Arduino 2560 Mega with the sensors involved (AD8232 Heart Rate Monitor, SEN11574 Pulse Sensor, MAX302100 Pulse Oximeter) that will use a GET request to load and store the information into the database that is written in SQL using MySQL that is being hosted in the Digital Ocean cloud hosting service, that will use a POST request to send the information to the Android application that is using Android Studio. User acceptance testing will also need to be used to test software to ensure that data is being read properly. This will involve a user that understands the requirements of the entire system and application and will ensure that the requirements are being met or not and to point out any issues or missing components from the integration testing phase.

2.1 Testing Approach

The testing approach has been changed intensively since the beginning of Report 1. However, the testing approach at this time will consist of the following:

- Functional tests
- Performance tests

Because the prototype does not currently contain a high level of integrity, we as a team are concerned with how well the system will manage to meet the current requirements as we had exclaimed for the end user in Report 1. In order to achieve this, we will be focusing intensely on testing the current prototype at component and system level. We will be testing the performance of the sensors and ensuring that they are giving appropriate values, as well as being able to have a connection from the Arduino system to the cloud hosting service, storing it into the database, and then having that information be retrieved from the database to the mobile application and displaying the appropriate values with the required measurements. However, with regards to Report 1, our requirements specifications have changed greatly. As it currently stands, the following will define our refined specifications requirements.

2.2 Updated Requirements Specifications

Requirement	**Description**
System Requirement	
1	The system should be able to take heart rate

	readings using the AD8232 heart rate monitor, the SEN11574 pulse sensor, and the MAX30102 pulse oximeter chip at a sampling rate of 100Hz.
2	The system should be able to store the heart rate, ECG, and blood oxygen readings in a database in the appropriate format for each of the sensors using SQL.
3	The system should be able to connect to a Digital Ocean cloud hosting service using GET requests that have been defined with the PHP scripts.
4	The system should be able to retrieve the heart rate data from the database stored on the Digital Ocean cloud hosting service using POST requests initiated by the mobile application.
5	The system should be able to provide real-time heart rate data to the user through the mobile application.
6	The system should be secure and should not be vulnerable to any hacking attempts.

2.1 System Requirements

Requirement	**Description**
Functional Requirements	
1	The system should have an Arduino Mega 2560 board connected to an AD8232 heart rate monitor, a SEN11574 pulse sensor, and a MAX30102 pulse oximeter chip.
2	The system should be able to take heart rate readings at a sampling rate of 100Hz using the sensors.
3	The system should be able to connect to a

	Digital Ocean cloud hosting service using GET requests.
4	The system should be able to store the heart rate data in a database in the appropriate format for each of the sensors using SQL.
5	The system should be able to retrieve the heart rate data from the database using POST requests initiated by the mobile application.
6	The system should have a mobile application that will allow users to login and retrieve their heart rate data.
7	The system should provide real-time heart rate data to the user through the mobile application.

2.2 Functional Requirements

Requirement	**Description**
Non Functional Requirements	
1	The system should be reliable and should not have any downtime.
2	The system should be secure and should not be vulnerable to any hacking attempts.
3	The system should be able to handle a large amount of data without any performance issues.
4	The system should be user-friendly and easy to use.

5	The system should be scalable, meaning additional sensors can be added to the system in the future.

2.3 Non Functional Requirements

Although we have updated our current specifications for the requirements needed, we have not yet reached the point where we will have every acceptance test available at this time. However, the current testing uses the data that we currently have from the current state of the prototype components, and the testing will be used based on simulations based off the teams testing.

3 Acceptability Tests

3.1 Scenarios

The following table depicts the various scenarios that were created from the stakeholder requirements that had been specified in Report 1, Section 4 to Section 6, and the acceptance test coverage for each scenario. However, our Scenarios have changed a considerable amount from then and will be shown as much as seen below.

ID	User Story	Requirements	Associated Acceptance Test
[S-1]	As a user, I want my device to function	The user should have the Arduino wirelessly connected to the network and have all the sensors working appropriately. The user should have access to all the sensors in a clear way. The system should be functioning and have necessary power.	
[S-2]	As a user, I want to take a medical reading from any sensor I want	The user should be able to have access to any sensor and be able to take a reading in real time	1,6,7
[S-3]	I want to be able	The user should be able to take the	2,5,6,7,9

	take my device with me outside and have my readings be taken continuously even if I am not connected to the cloud network	device remotely and be able to take measurements of their data, while it stores on the Arduino, and when reconnecting to the the network that is shared with the cloud server, it should be able to send timestamps of real time data while the user was outside.	
[S-4]	I want to see the status of the sensors on the app	The user should be able to take any reading with a sensor and should see a change in their reading	1,3,9
[S-5]	As a user, I want to be able to see my previous health data	The user should be able to see previous data in the application to check their health readings in accordance to the timestamps	1,2,3,5,6-9
[S-6]	As a user I want a mobile application that is fully functional	The user should be able to run the application on a mobile application	2,4,5,7,8,9
[S-7]	As a user I want to navigate through the application easily	The user should be able to login, logout, and navigate through the pages	7,8,9
[S-8]	As a user I want to be able to share the data with my doctor	The user should be able to store the data and be able to share with a healthcare practitioner	2,3,5,6,7,9

Table 3.1: Use cases and related requirements

ID	Title	Steps	Inputs	Outputs	Acceptance Criteria
1	Take health readings from sensor	1: Login to Mobile Application	Sensor inputs (be able to use the sensor	Sensor reading based off	Once the user is taking the reading, it should

		2: Go to selected sensor 3. For required sensor, place either finger on device or apply 3 lead	correctly)	sensor being used	be able to update in real time with timestamps with the desired reading
2	Login to application with user specific information	1: Have a unique user ID and password 2: Have information be saved in database	Correct ID being supplied	ID being initialized by the database	Once the user has attempted to login, the login should either be successful or unsuccessful based off the user credentials
3	View specific sensor data	1: Have separate databases for different data 2: Have the correct parameters for each specified data	Data is accurate with the correct parameters	Data is being sent to the user	Once the user accesses the data logs based off the timestamps, it should allow for accurate and consistent data
4	Suggestions for different data	1: Login on mobile application 2: Go to suggestions dashboard	Any suggestions the user has	Suggestions will be displayed	Once the user signs in and uploads data about different sensors they want to have in the system, the system will acknowledge it
5	Data from a specific timestamp	1: Login to mobile application 2: Go to desired sensor dashboard for the data that wants to be viewed	Health data from the user	The correct data parameters for the data that the user is seeking for in that timestamp	The mobile application should display the correct data within the correct parameters for the health section that they are seeking data from with respect to the required

					timestamp
6	Database is retrieving and sending data properly	1: Data is being sent to cloud hosting service and stored 2: Data is being retrieved from cloud hosting service and being displayed	Data storage	Data is being stored and retrieved with the necessary POST and GET api calls	The cloud hosting service should be able to use GET api calls to retrieve the sensor data from the Arduino MEGA 2560 and should store that data, and then when the application demands the information based on the unique ID of the user, should be able to retrieve any data using a POST api call.
7	Mobile application works	Non-functional requirement	N/A	N/A	There should be no issues launching the android application
8	Easy to use	Non-functional requirement	N/A	N/A	User should be able to navigate throughout the application easily
9	Backend is functioning	Non-functional requirement	N/A	N/A	There should be no major errors in the application

Table 3.2: Functional and Non-Functional Acceptance Tests

4 Test Results

ID	Results	Date	Status	Comments
1	Currently unavailable as the feature is dealing with bugs that need to be fixed	03-07-2023	**Fail**	Currently in the processing of being debugged
2	Users were able to login with unique IDs and passwords with some minor issues	03-07-2023	**Pass**	Test passed successfully as intended
3	Currently working on the GET requests to allow for connection to Arduino sensors and POST requests to view on mobile application with cloud hosting service	03-07-2023	**Fail**	Still in the process of being implemented
4	Suggestion box has been implemented in the mobile application	03-07-2023	**Pass**	Test passed successfully
5	Due to the data not being processed as a result this has not implemented as of yet	03-07-2023	**Fail**	Still in the process of being implemented
6	A suggestion has been made to change Arduino script with some more unique IDs in the script to enable PHP cohesion to retrieve unique user information	03-07-2023	**Fail**	Still in the process of being implemented
7	Mobile application very bare bones and needs a significant amount of work, only login has been complete	03-07-2023	**Fail**	Still in the process of being implemented
8	Will require mobile	03-07-2023	**Fail**	Still in the process of

	application completion to ensure this			being implemented
9	Will need to have sensor data to be connected properly. Only have specific tables set up.	03-07-2023	**Fail**	Still in the process of being implemented

Table 4.1 Acceptance tests Results and Status

5 Issues and Defects

Within the current system, all the issues within the system have been addressed and identified amongst all the members of the team. All the issues and defects have been taken note of in a master list that has been recorded and is a private document that has only been made available for the members of the team. The issues are recorded as a major task that needs to be solved for the task that is being affected.

5.1 Arduino Sensors - Noisy readings

There are currently some issues with the Arduino sensors that are being connected as there is a large issue with having very noisy readings in the sensors. This is currently being looked into as a configuration issue, but if persists, will need to be replaced with a new sensor of the same type if it proves to be a hardware issue. We suspect that the configuration is wrong and it is in the wrong voltage, or it has been used in the wrong voltage previously and as a result has damaged the components. That being said, the error is being observed for the readings, but the readings are being made, and as time passes the likelihood of coming to a positive resolution is highly possible.

5.2 Database - GET and POST Requests

There is currently an issue with the GET requests as they have not been configured to retrieve the data from the Arduino and store the information, nor have the POST requests been configured to send the information. The databases have been configured but need much more refinement to better fit the sensor readings within their parameters. The currently working process is the user login database that allows a user to login and have a unique ID. As we progress into the coming weeks we will almost certainly have reached a positive conclusion.

5.3 Mobile Application

The mobile application that we had for the prototype has a lot more refinement that needs to be completed however the login info has been configured and can allow for unique IDs to login, but there is still a significant amount that needs to be put in to read the database and have the information be accessed as per the users needs. With some more time, we will be able to have the mobile application working at full functionality.

6 Contribution Matrix

	People			
Task	**Mamun Hossain**	**Gobikah Balaruban**	**Manreet Kaur**	**Aaditya Rajput**
Updated System Requirements	60%	10%	15%	15%
Use Cases	20%	20%	30%	30%
Functional and Non Functional Acceptance Tests	20%	30%	30%	20%
Acceptance test results	10%	30%	40%	20%
Issues and Defects	20%	20%	20%	40%
Mobile Application Development	10%	10%	20%	60%

Cloud hosting Development	10%	30%	50%	10%
Backend Development	10%	40%	40%	10%
Hardware Development	80%	5%	10%	5%