Department of Computer Science and Engineering
University of Texas at Arlington

βeta βlockers

Wireless Blood Pressure Monitor

System Test Plan

Ian Jungmann
Hao Nguyen
Scott Phan
Benjamin Bloom
Sabreen Alabedredha

Table of Contents

Table of Contents 2
Document Revision History 4
List of Figures 5
List of Tables 6
1. Introduction 8
1.1. Document Overview 8
1.2. Product Overview 8
1.3. Project Scope 8
2. References 10
2.1. Overview 10
2.2. System Requirements Specification 10
2.3. Architectural Design Specification 21
2.4. Detailed Design Specification 23
3. Test Items 33
3.1 Overview 33
3.2 Hardware Tests 34
3.3 Unit Tests 34
3.4 Component Tests 41
3.5 Integration Tests 43
3.6 System Validation Tests 44
4. Risks 46
4.1 Overview 46
4.1 Table of Risks 46
5. Features to Be Tested 47
5.1 Overview 47
5.2 Customer Requirements 47
5.3 Package Requirements 51
5.5 Safety Requirements 53
5.6 Maintenance and Support Requirements 53

5.7 Security Requirements 54
5.8 Other Requirements 56
6. Features Not To Be Tested 57
6.1 Overview 57
6.2 Customer Requirement 57
6.3 Package Requirement 58
6.4 Performance Requirement 59
6.5 Safety Requirement 60
6.6 Maintenance and Support Requirement 61
6.7 Security Requirement 61
6.8 Other Requirement 62
7. Testing Approaches 64
7.1 Overview 64
7.2 Strategy 64
7.3 Configurations 65
7.4 Tools 65
7.5 Core Functionalities 65
7.6 Metrics 66
8. Item Pass/Fail criteria 67
8.1 Overview 67
8.2 Hardware Tests 67
8.3 Unit Tests 68
8.4 Component Tests 70
8.5 Integration Tests 70
8.6 Validation Tests 71
9. Test Deliverables 72
9.1 Overview 72
9.2 Deliverables 72
9.3 Bugs and Defects 73
9.4 Test Code 73
10. Test Schedule 74
11. Approval 75

Document Revision History

Revision Number	Revision Date	Description	Rationale
1.0	03/24/2014	First Draft	Version to be reviewed by peers
2.0	4/1/2014	Baseline Version	Updated version that reflects peer critique

List of Figures

Figure 2 - 1: Architecture Diagram 21
Figure 2 - 2: Detailed Design Diagram 24
Figure 3 - 1: Relational Diagram 33
Figure 7 - 1: Test Document Excel Example 64

List of Tables

Table 2 - 1: Customer Requirements 10
Table 2 - 2: Package Requirements 13
Table 2 - 3: Performance Requirements 14
Table 2 - 4: Safety Requirements 16
Table 2 - 5: Maintenance and Support Requirements 17
Table 2 - 6: Security Requirements 18
Table 2 - 7: Other Requirements 20
Table 2 - 8: Client Layer Data Flow 25
Table 2 - 9: Client Layer Requirement Mapping 26
Table 2 - 10: Transport Layer Data Flow 27
Table 2 - 11: Transport Layer Requirement Mapping 28
Table 2 - 12: Business Logic Layer Data Flow 29
Table 2 - 13: Business Logic Layer Requirement Mapping 30
Table 2 - 14: Data Storage Layer Data Flow 31
Table 2 - 15: Data Storage Layer Requirement Mapping 32
Table 3 - 1: Hardware Tests 34
Table 3 - 2: Browser Display Unit Tests 34
Table 3 - 3: Web Validation Unit Tests 34
Table 3 - 4: Chart Generation Unit Tests 35
Table 3 - 5: Application UI Unit Tests 35
Table 3 - 6: BPM Validation Unit Tests 35
Table 3 - 7: Data Packet Processing Unit Tests 35
Table 3 - 8: Bluetooth Communication Unit Tests 36
Table 3 - 9: Request Encryption Unit Tests 36
Table 3 - 10: Response Decryption Unit Tests 36
Table 3 - 11: Client/Server Communication Unit Tests 36
Table 3 - 12: Request Decryption Unit Tests 37
Table 3 - 13: Permission Authentication Unit Tests 37
Table 3 - 14: Response Encryption Unit Tests 37
Table 3 - 15: Request Delegation Unit Tests 37
Table 3 - 16: Analysis Controller Unit Tests 38
Table 3 - 17: Analysis Command Unit Tests 38
Table 3 - 18: Data Preparer Unit Tests 38
Table 3 - 19: Page Generator Unit Tests 39
Table 3 - 20: Data Controller Unit Tests 39
Table 3 - 21: Cryptography Unit Tests 39
Table 3 - 22: Database Get Unit Tests 40
Table 3 - 23: Database Update Unit Tests 40
Table 3 - 24: Client Layer Component Test 41

Table 3 - 25: Transport Layer Component Test 41
Table 3 - 26: Business Logic Layer Component Test 42
Table 3 - 27: Data Storage Layer Component Test 42
Table 3 - 28: Integration Tests 43
Table 3 - 29: System Validation Tests 44
Table 4 - 1: Risk Table 46
Table 7 - 1: Success and Failure Metrics 66
Table 8 - 1: Hardware Tests Pass/Fail Criteria 67
Table 8 - 2: Unit Tests Pass/Fail Criteria 68
Table 8 - 3: Component Tests Pass/Fail Criteria 70
Table 8 - 4: Integration Tests Pass/Fail Criteria 70
Table 8 - 5: Validation Tests Pass/Fail Criteria 71
Table 11 - 1: Approval Table 75

1. Introduction

1.1. Document Overview

The System Test Plan (STP) provides details regarding how the Wireless Blood Pressure Monitoring System will be tested to ensure that requirements have been met and the final product is of acceptable quality. Using the team's SRS, ADS, and DDS as references, the team can construct tests and testing criteria that are specifically designed to prove the functionality and quality of a given component of the system, whether the component be a small portion of the overall system, such as a module, a slightly larger component (like a subsystem or an entire layer), or even the system as a whole. Together, these tests should prove that the system functions as it is meant to (as specified by the ADS and the DDS), and that the product provides the functions and services that were required by the customer (the requirements, as specified in the SRS).

1.2. Product Overview

The Wireless Blood Pressure Monitoring System is meant to provide doctors and patients an easy way to exchange data and information about a patient's health. By using the product's Bluetooth-enabled blood pressure monitor and an Android phone, patients will be able to measure their blood pressure and send the data wirelessly to the system's server. The system's web service will then be able to display the data in various forms (different charts and graphs, tables, etc.), give the patient information about their general health (basic advice, alerts in case of concerning trends), and provide a way for the patient's doctor to be able to view this information, to allow them to easily keep track of their patients' blood pressure without having to schedule appointments.

1.3. Project Scope

The Wireless Blood Pressure Monitoring System will be able to transfer a patient's blood pressure reading to the system's servers by using the provided blood pressure monitor and an Android phone with the system's application installed. Doctors will have the ability to create a profile for themselves on the web service, as well as profiles for their patients (though it is ultimately the patient who will be in control of their own profile). Various charts and graphs will be generated from the provided data in a way that will allow both patients and doctors to more easily visualize different trends and statistics, so that it may be easier to recognize problems and make adjustments accordingly. An advice system shall be put in place that will look at the patient's blood pressure results, as well as other data that may have been collected about them (new prescriptions, existing conditions, etc.), and make general observations and give feedback to the patient based on the data, though these are meant to be general suggestions and should not be substituted for a doctor's advice. An alert system shall be provided that will alert a patient's doctor that a patient may be at risk if the system detects a concerning trend or a problem (based off of various calculations). Patients will also be able to generate alerts themselves if they so choose.

Because of the sensitive nature of much of the data the system is collecting, security will play a major part in the system. Data encryption, permission assignments, secure transferring of data across the network, and other security features need to be in place, so that doctors and patients will feel safe when using the system, and so that the system is HIPAA compliant.

2. References

2.1. Overview

The tests and test plans documented in the TPS have been designed to fit the elements of the system specified in previous documents (the SRS, ADS, and DDS). The system will be tested to ensure that it fits the requirements specified in the SRS, and that the components of the system function properly and interact correctly, as detailed in the ADS and the DDS. The following sections will provide the key portions of those documents that need to be considered when designing the test plan.

2.2. System Requirements Specification

The SRS details the requirements for the project that have been set by the various stakeholders (team members, sponsors, etc.). These requirements specify things such as general functionalities and features, as well as more detailed things such as latency times and safety features. Testing to ensure that our system meets these requirements will allow the team to make a final determination as to whether the product is acceptable or not.

2.2.1 Customer Requirements

Table 2 - 1: Customer Requirements

SRS No.	Requirement	Description	Priority
3.1	Non-invasive Measurements	Blood Pressure Monitor will take readings in a non-invasive manner.	1 – Critical
3.2	Oscillometric Method	Blood Pressure Monitor will use the oscillometric method to evaluate blood pressure.	1 – Critical
3.3	Automatic Inflation/Deflation	Blood Pressure Cuff will automatically inflate and deflate.	1 – Critical
3.4	Cuff Location	The Blood Pressure Cuff will be worn on the upper or lower arm.	1 – Critical
3.5	User Authentication	An authentication system will be used before all sessions to ensure user security.	1 – Critical
3.6	Blood Pressure Reading	The device will measure the blood pressure of the user it is attached to.	1 – Critical
3.7	View Results	Users will be able to view their results on the monitor.	1 – Critical
3.8	Multiple Readings	Users will be able to record and submit multiple readings in a session.	2 – High
3.9	Result Transmission and Storing	The monitor will transmit the blood pressure results to the website's database for storage.	1 – Critical
3.10	Web Service	A web service will be created to display results, handle users, and house other features.	1 – Critical
3.11	Account Types	The web service will support two types of accounts: patients and doctors.	1 – Critical

3.12	Doctor Account Creation	Doctors will be able to create their own account on the web service.	1 – Critical
3.13	Patient Account Creation	Doctors will create patient profiles and have them connected to their own. The doctor will enter medical information and other key information about the patient's health. The system, with the doctor's permission, will then send an email to the patient's email address with temporary login information. The patient will then change their login credentials, add their personal information (if they wish), and confirm the changes.	1 – Critical
3.14	Patient's Doctor	A person who wishes to use this system as a patient must be affiliated with a doctor or caretaker who has an account on the web service.	1 – Critical
3.15	Doctor Limit	A patient may only have one doctor account tied to their account on the web service. If the patient wishes to have more doctors view their data, those doctors will have to use the patient's viewing account to see the information.	1 – Critical
3.16	Doctor Editing	Doctor accounts can have their information edited by the account holder.	3 – Moderate
3.17	Patient Editing	Profile information (email, password, full name) can be edited by the given patient. Medical information will be edited by the patient's doctor.	1 – Critical
3.18	View Charts and Graphs	All registered users on the site will be able to view various charts and graphs on the website that describe the patient's blood pressure conditions, overall health, and other factors, assuming that they have access to said patients information.	1 – Critical
3.19	Annotations	Patients registered on the site will be able to make notes about particular readings of theirs on the website, to relay information such as changes in their health or lifestyle.	2 – High
3.20	Patient Viewing Profile	All patients will have a viewing profile, a way in which friends, family members, other doctors, or anyone else the patient wishes to give access to can view how the patient is doing and other information. This profile is separate from the standard Patient profile, it will be tied to the account with a separate set of login credentials, and it is extremely limited in functionality. The patient will be able to change the login credentials for this at any time to restrict access.	3 – Moderate

3.21	Create Alerts	Alerts can be created, either automatically by the system if it detects a problem or a concerning trend, or by a patient if they want to alert their doctor about something, which will be sent to the doctor for later viewing.	2 – High
3.22	View Alerts	Doctors will be able to view alerts and the issue they pertain to.	2 – High
3.23	Messaging System	Doctors and patients can send messages to one another to be read later (like an email system), as a way of addressing things not covered by the alert system and the annotation system.	4 – Low
3.24	Multiple Users	The blood pressure monitor will be able to support multiple users on the same device.	4 – Low
3.25	Login to Access Data	Users will have to log into the website in order to view any data.	1 – Critical
3.26	HIPPA	Product will comply with all HIPAA standards (as well as other standards that apply to medical data) to ensure customer privacy. This will be covered in detail in the Security Requirements section.	1 – Critical
3.27	Advice System	The website will be able to give patients health information, advice, and general suggestions to keep their blood pressure in check by looking at general patterns and user information	1 – Critical
3.28	Manual Upload of Information	In the case where a patient's wireless connection goes down or begins to suffer problems, the device will store the patient's readings for a set amount of time. The patient will then be able to manually upload the readings using a computer, if they feel that the wireless connection will not resume working in the near future.	5 – Future

2.2.2 Package Requirements

Table 2 - 2: Package Requirements

SRS No.	Requirement	Description	Priority
4.1	Blood Pressure Cuff	The product will be packaged with a standard blood pressure cuff. The cuff will weigh between ½ -1 pound, and will be fitted for arm circumference of 9-14 inches	1 – Critical
4.2	Blood Pressure Monitor	The product will come packaged with a monitor which is attached to the cuff. It will be the interaction point between the patient and the system. The patient can authenticate, take blood pressure reading, view current readings in the session, and send blood pressure readings to the website. The Monitor will be a 3 x 4 inch display and will weigh no more than 6 ounces.	1 – Critical
4.3	Android Phone	The product will require an Android phone, which will not be packaged with the product. The phone will use the Android application to interact with the blood pressure monitor device.	2 – High
4.4	Android Application	An android application will be provided via Google play store. The application will allow the user to interact with the blood pressure cuff to authenticate, take blood pressure reading, and send the readings to the website.	2 – High
4.5	Software Packaging	All software will be installed prior to packaging. No set up will be required by the Patient.	1 – Critical
4.6	Website	The team will create a website that will receive blood pressure readings from the device. These readings will be displayed on the website in chart and graph format. The patient will be able to annotate, view advice regarding the readings, send and receive alert messages as well as contact their doctor through the website.	1 – Critical
4.7	Battery	The device will be operated via battery power. Batteries will not be packaged with the device.	3 – Moderate
4.8	Connection Cable	Connection cable will be provided to manually upload the data on to a computer and uploaded to the website if wifi is not available.	5 – Future
4.9	User Manual	A user manual will be provided with the product. The manual will contain detailed information regarding the product, website, and how to operate the device.	2 – High

2.2.3 Performance Requirements

Table 2 - 3: Performance Requirements

SRS No.	Requirement	Description	Priority
5.1	Storage Conditions	The blood pressure monitor shall maintain the performance requirements specified in this document after being stored for 24 hours at 21°C (70°F), give or take 10°C variance, and at a relative humidity lower than 90% (noncondensing).	2 – High
5.2	Operating Conditions	The blood pressure monitor shall maintain the safety and performance characteristics specified in this document during operation over the following ranges of environmental conditions: a) Temperatures between 11°C and 31°C; b) Relative humidity under 90% (noncondensing)	2 – High
5.3	Operational Battery Life	The blood pressure monitor shall maintain the performance characteristics specified in this document for at least 30 minutes while running the measurement acquisition process continuously.	2 – High
5.4	WLAN Authentication Latency	The blood pressure monitor interface to the end-user's wireless local area network (WLAN) will take no longer than 10 seconds to attempt a successful connection. After 10 seconds, the blood pressure monitor will cease the connection activity.	2 – High
5.5	WLAN Connection Failure Response Time	The blood pressure monitor interface to the end-user's wireless local area network (WLAN) will take no longer than 10 seconds, intervene appropriately, and notify the end-user of the current status of the WLAN connection.	2 – High
5.6	Web Authentication Latency	The blood pressure monitor interface to the internet and to the web service will take no longer than 30 seconds to attempt a login. After 30 seconds, the system will cease the login activity.	2 – High
5.7	Blood Pressure Monitor Consistency Regarding Battery Life	The blood pressure monitor will still be able to take consistent readings until the device detects that the battery's voltage is too low to continue normal operation. The variation in blood pressure readings shall be confined to within 15% accuracy.	2 – High
5.8	Blood Pressure Monitor Accuracy	The blood pressure measurements obtained using the monitor shall not deviate more than 10% at any measurement range from an existing commercial blood pressure monitor.	2 – High

5.9	Blood Pressure Monitor Mechanical Robustness	The blood pressure monitor shall be able to maintain its performance and safety requirements when being physically handled during normal use.	2 – High
5.10	Blood Pressure Monitor Startup Delay	The blood pressure monitor shall be able to boot up from a cold start to a ready state for the user within 30 seconds.	3 – Moderate

2.2.4 Safety Requirements

Table 2 - 4: Safety Requirements

SRS No.	Requirement	Description	Priority
6.1	Maximum Cuff Pressure	For devices intended for public, home, or other unsupervised use, a means of limiting cuff pressure shall be provided so that the maximum cuff pressure will never exceed 300 mmHg. For devices intended for use under professional supervision, a means of limiting cuff pressure shall be provided so that the maximum cuff pressure will never exceed 300 mmHg or 30 mmHg above the upper limit of the instrument's manufacturer-specified operating range, whichever is lower. In addition, the device shall incorporate means to ensure that cuff pressure will not be maintained above 15 mmHg for longer than 3 minutes (min).	1 – Critical
6.2	Cuff Pressure Release	An easily accessible and clearly labeled means of allowing the user to deflate the cuff shall be provided. During the rapid exhaust of the pneumatic system with fully-opened valve, the time for the pressure reduction from 260 mmHg to 15 mmHg shall not exceed 10 s.	1 – Critical
6.3	Electrical Safety	The blood pressure monitor shall not be able to deliver a harmful shock to its user. The voltage range throughout the device will be limited to 12 volts. The device will also only use DC type power as its primary source of electrical power.	2 – High
6.4	Eligible Users	The blood pressure monitor shall not be intended for neonatal users. It will be only intended for those who understand how to operate the device safely.	2 – High
6.5	Blood Pressure Monitor Mechanical Safety	The blood pressure monitor, primarily the wearable components, shall not be able to easily induce physical injury to the user through its physical form.	2 – High

2.2.5 Maintenance and Support Requirements

Table 2 - 5: Maintenance and Support Requirements

SRS No.	Requirement	Description	Priority
7.1	User Manual	A general user manual will be created that will instruct the end-user on how to safely operate the blood pressure monitor and access his or her information on the website.	2 – High
7.2	Blood Pressure Monitor Battery Replacement	The battery on the blood pressure monitor shall be replaceable, as it will be found on the consumer marketplace. The battery shall also be able to be removed easily from the device by the targeted audience without compromising the safety and performance requirements.	2 – High
7.3	Web Service Software Portability	The web service software shall be able to be deployed via an automated script that sets up the server software (e.g. Apache, MySQL, etc.) and preloads system templates such as database schemas and defaults. This is intended to ease deployment onto the client's intended web servers.	2 – High
7.4	Web Service Software Update Effects	The web service will be able to be able to be modified without having to reboot the operating system or web server (e.g. Apache). There is no guarantee that the web application itself will not restart.	3 – Moderate
7.5	Web Service Administration Layer	The web service shall have a portal for the system administrators to manage the basics of the web portal so that the source of the web application does not need to be modified and redeployed. This includes website-wide notifications/alerts, the ability to enable/disable certain services, and enabling/disabling the entire web application.	3 – Moderate

2.2.6 Security Requirements

Table 2 - 6: Security Requirements

SRS No.	Requirement	Description	Priority
8.1	Data Transport Encryption	The web service shall encrypt all data, particularly PHI, that is transmitted between the blood pressure monitor and the data collecting web server. This requirement is primarily focused on the level of the communication between monitor-client and the web server and not the WLAN connection (see 8.5).	1 – Critical
8.2	Authorization	The collected PHI data will only be accessible to authorized personnel using audited access controls. Mechanisms will be designed and implemented to ensure that data can be accessed only by appropriate users. This requirement may rely on storage encryption (section 8.4) to prevent unauthorized access, as well as on the web hosting vendor to comply with the HIPAA Business Associate (section 8.7) to maintain control over low-level access to the web server's functionality.	1 – Critical
8.3	Integrity	The system will attempt to prevent any PHI that will be collected and stored from the user from unauthorized modification or corruption and ensure its validity. This requirement may rely on storage encryption (section 8.4) and authorization protocols (section 8.2) to deter unauthorized modification.	2 – High
8.4	Storage Encryption	All PHI data that is being stored or archived will be encrypted and intended for storage of a minimum of five years. In general, the data will be encrypted using symmetric encryption of at least AES-128 strength. The encrypted data can only be decrypted and accessible by users with the appropriate keys. The keys will further be protected using a form of public key encryption. This requirement would also further enforce the protocols of authorization (section 8.2) on PHI data.	2 – High
8.5	Disposal Guarantee	PHI data, when found necessary to remove from the system, will be effectively purged. This guarantees that when the system removes the data from its oversight, it would be thoroughly deleted from itself and all of the system's underlying dependencies. If a data backup system is implemented (section 8.7), then this requirement will be implemented completely on this system as well. This means that the backups must be thoroughly cleansed of or recompiled without the appropriate data.	4 – Low

8.6	WLAN Connection Security	The wireless network connection between blood pressure monitor and the WLAN router shall communicated through an encryption of at least a symmetric encryption of AES-128 strength.	4 – Low
8.7	Data Backup	The data collected by the web portion of the system will be backed up in case of an emergency or accidental deletion situation. The backup must be securely handled (i.e. encrypted and placed under an access control paradigm).	5 – Future
8.8	Business Associate	The system will have a HIPAA Business Associate Agreement with every vendor that has access to the data. The vendor, in the scope of the project, is the company that will host the system's web application. The web hosting company will have all the data that is being transferred and stored, so putting an agreement in place will ensure that the company will follow the HIPAA requirements for the data and the servers. This guarantees that the web hosting company will provide an infrastructure that meets HIPAA compliance requirements. The development team will then be responsible for applying secure programming methodologies to design and manage the website so that its functionality is HIPAA compliant.	5 – Future

2.2.7 Other Requirements

Table 2 - 7: Other Requirements

SRS No.	Requirement	Description	Priority
9.1	Web Browser Compatibility	The web interface shall be accessible and usable through popular web browsers for the user, caretaker/physician, and administrators. Such web browsers would include Internet Explorer, Mozilla Firefox, Google Chrome, and Safari.	1 – Critical
9.2	Blood Pressure Monitor Source Compatibility	The blood pressure monitor device's source code shall be compatible with Windows 7 development environments for implementation, deployment, and maintenance.	1 – Critical
9.3	Web Service Source Compatibility	The source code of the web service functionality shall be compatible with Windows 7 and popular Linux development environments for implementation, deployment, and maintenance.	1 – Critical
9.4	Smartphone Device Constraint	Any implementation of functionality for mobile devices, mainly smartphones, will be constrained to Android 3.0+ and Unix-type operation systems. This is due to their accessibility and availability of their tools. The choice of Android 3.0 and up is due to their available and compatible feature set for this project.	4 – Low
9.5	Multiple Health Modules for Website	The website will be able to support many different sets of data from many different pieces or hardware, such as blood sugar readings or weight measurements, to allow for a robust web service that will allow doctors to easily track patients with a wide variety of conditions or concerns.	5 – Future
9.6	Advanced Authentication Capabilities	The monitor will have advanced authentication capabilities, to better ensure that the correct person is wearing the device at all times.	5 – Future
9.7	Remote Control for Blood Pressure Monitor	The blood pressure monitor will come with a control that will be used to control the system, allowing for more portability and removing the need for a phone or a large box to support the system.	5 – Future

2.3. Architectural Design Specification

The ADS documents the high-level architecture for the product, the general concept of the various components of the system and how they will interact with one another. While the components in the ADS are not broken down as far as they could be, they do show how the larger, more general parts of the system are supposed to work. In order to ensure that the system functions properly, and that it is producing acceptable results, the team must test these components to verify that each one is working as it should, and that the interactions between the components are behaving correctly.

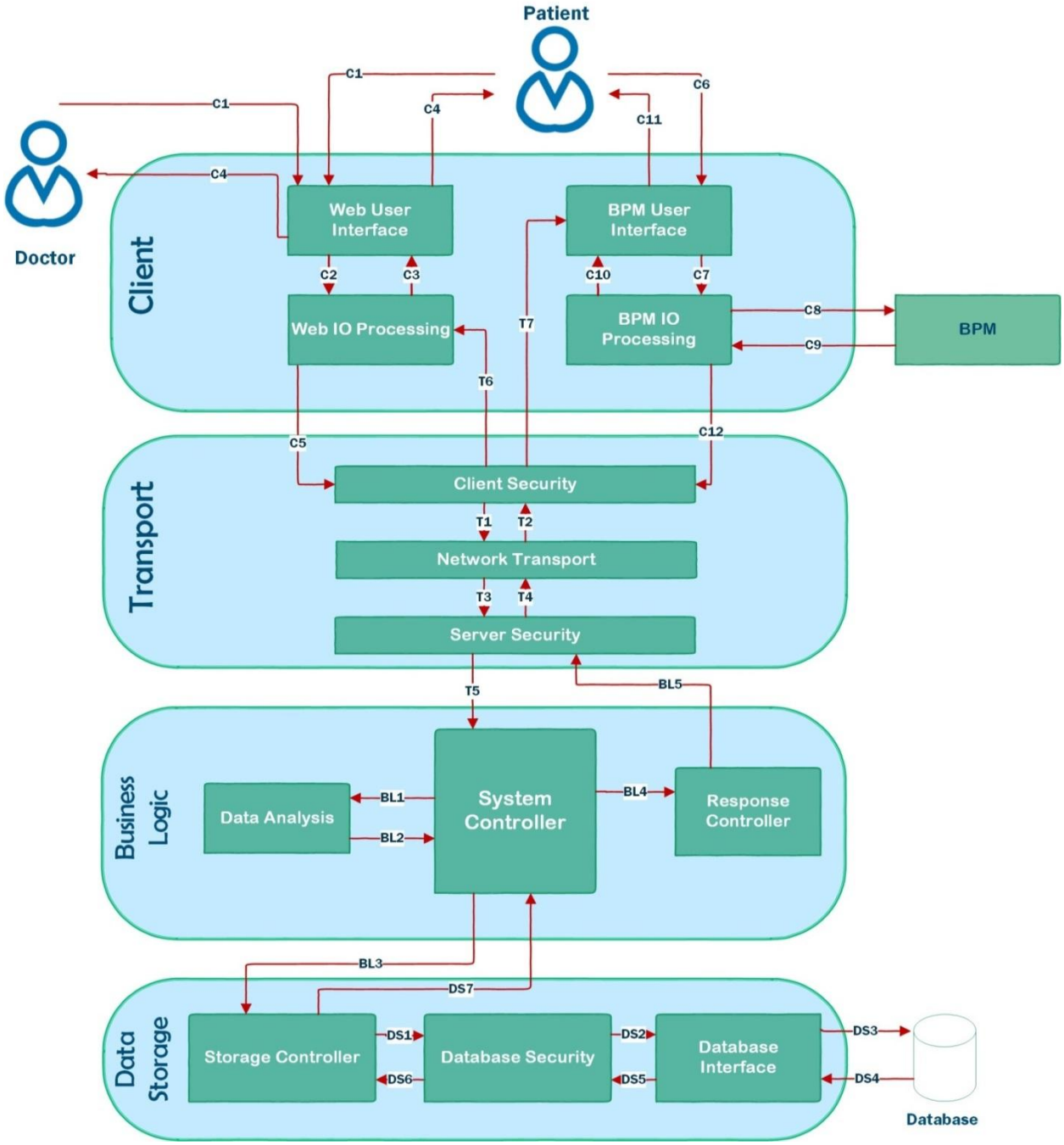

Figure 2 - 1: Architecture Diagram

2.3.1 Client Layer

The Client Layer is meant to deal with all things related to the portions of the system that users will interact with, specifically the BPM and the website. The layer takes its name from the client portion of the Client-Server model upon which we are basing our architecture, and will act similarly to a client in the Thin Client model. This layer's main focus is providing a consistent, user friendly interface that will allow users to consume the data that they need, as well as to input information and receive the output that results from this. In addition, this layer will handle some basic processing of input that needs to be handled on the client end of the system, either for safety or performance purposes, such as validating input to eliminate bad data, as well as processing the signals from the pressure sensors in the cuff and responding appropriately. The Client Layer will take in input directly from the user or hardware, whether that be sensor data, configuration changes, blood pressure readings, or others, and will either directly respond to the user or hardware, or will pass the data down to the Transport Layer, at which point it will wait for the Transport Layer to send response data for it to display.

2.3.2 Transport Layer

The Transport Layer, as its name suggests, handles transporting data from the client to the server and back (in our architecture, from the Client Layer to the Business Logic Layer). Its main purpose is to make sure that this data reaches its destination in a safe and secure manner. To do this, the layer has to deal not only with the actual transporting of the data through the use of various network protocols, but also with securing the data to make sure that everything going through the network cannot be intercepted and used in harmful ways. It also needs to make sure that the data being delivered to the receiver is able to be used, which means going through another layer of security after transporting it to restructure the data to make it useful. The Transport Layer takes its name from the major functions it provides. It takes a request from the Client Layer and passes it to the Business Logic Layer and it gets a response from the Business Logic Layer and passes it back to the Client Layer. To put it in simple terms, it packages the materials, ships them, and then un-package them. This is one of the simpler layers of the system's architecture, but due to the team's emphasis on security and privacy, it is a key layer, and because of the team's choice to structure the architecture in a Client-Server fashion, this layer provides the crucial connection between the two.

2.3.3 Business Logic Layer

The Business Logic Layer is tasked with doing the bulk of the work on the server side of the architecture. Specifically, it will be doing data process for the system, such as data analysis and formatting. This layer is the brain of the system, it figures out what the client is requesting, completes the request, and returns any response necessary. The layer does not deal with getting data, or with the saving or storage of data. Its goal is to amass the information necessary from other sources, process the information, and push the newly processed data back up to the Transport Layer. Some of this data will come from the Transport Layer, as mentioned previously. Other data will come from the Data Storage Layer. The Business Logic Layer is not concerned with what actual data is needed from the Data Storage Layer, it will send the Data Storage Layer a request that specifies which process it needs to run, along with any input needed, and the Data Storage Layer will respond with whatever data is needed to complete that process, or an error message if need be. If the Business Logic Layer just needs to store more data in the database, it will send the data to the Data Storage Layer to handle, and will expect a confirmation message in response.

2.3.4 Data Storage Layer

The Data Storage Layer is the lowest layer of our system, and it has a very simple, but very important job. The Data Storage Layer must dictate how data is stored in and retrieved from the database. Without the Data Storage Layer, either the Business Logic Layer would have to pick up the slack, resulting in a very complex and dense layer, or there would be no interfacing with the database, meaning that no data could be saved, and any features that rely on long-term data would be lost. The Data Storage Layer is the heart of the system, it determines who gets what data and distributes it to keep the system functional. The Data Storage Layer can only be accessed through the Business Logic Layer. These two layers work together to form the core of the server model one would see in a Client-Server Model. The Business Logic Layer handles what the server needs to do. The Data Storage Layer handles what the server needs in order to do that. The Business Logic Layer will send the Data Storage Layer a request that states what the Business Logic Layer needs to do (what process it wants to run) and any input data that would accompany this, the Data Storage Layer figures out exactly what the Business Logic Layer needs to do this, it interacts with the database to get or store this data, and is pushes the data or the database response back to the Business Logic Layer.

2.4. Detailed Design Specification

The DDS further breaks down the architecture specified in the ADS. This is the furthest the architecture and the design of the system get broken down, which result in the smallest components of the system, modules. These modules, like the layers and subsystems they are derived from, each have their own purpose and responsibilities, and by each doing their proper task and interacting with each other, the system should then operate in the way it is intended. Because these modules are the smallest components of the system, it is crucial that these be tested to ensure that they are functioning properly, as it is impossible to ensure that the larger components of the system work without first proving that the modules that male them up work.

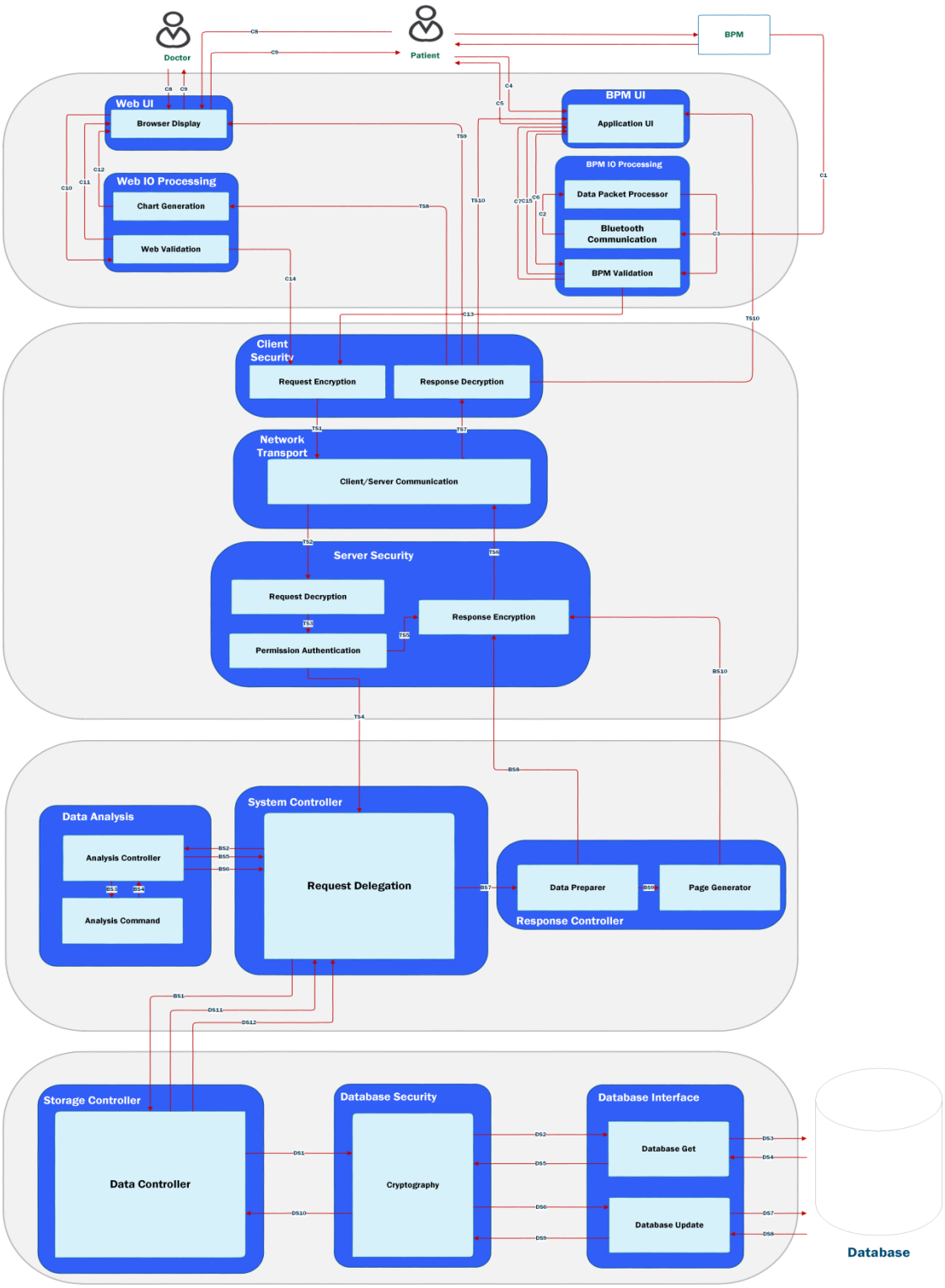

Figure 2 - 2: Detailed Design Diagram

2.4.1 Client Layer Data Flow

Table 2 - 8: Client Layer Data Flow

Data Flow	Description
C1	Data Packet with raw blood pressure reading
C2	Data Packet with raw blood pressure reading
C3	String formatted translation of Data Packet.
C4	User inputs
C5	Success | Fail message out put to screen.
C6	User inputs extracted from elements.
C7	Boolean Valid
C8	User inputs
C9	Visual Response
C10	HTML elements (forms, buttons, etc.) that contains user inputs
C11	Boolean for Invalid, or N/A for Valid
C12	JavaScript chart element to be inserted into the web page
C13	HttpConnection Object that contains the user request
C14	HttpConnection Object that contains the user request
C15	Boolean Invalid

2.4.1.1 Client Layer Requirements Mapping

Table 2 - 9: Client Layer Requirement Mapping

Req. No.	Requirement Name	Browser Display	Web Validation	Chart Generation	App UI	BlueTooth Comm.	Data Packet Proc.	BPM Validation
3.5	User Authentication	X	x		x			x
3.7	View Results	X			x			
3.8	Multiple Readings				x	X	x	x
3.9	Result Transmission and Storage					X	x	x
3.1	Web Service	X	x	x				x
3.11	Account Types	X						
3.12	Doctor Account Creation	X	x					
3.13	Patient Account Creation	X	x					
3.14	Patient's Doctor	X						
3.15	Doctor Limit	X						
3.16	Doctor Editing	X	x					
3.17	Patient Editing	X	x					
3.18	View Charts and Graphs	X		x				
3.19	Annotations	X	x	x				
3.2	Patient Viewing Profile	X						
3.21	Create Alerts	X	x					
3.22	View Alerts	X						
3.23	Messaging System	X	x					
3.24	Multiple Users	X			x			x
3.25	Login to Access Data	X	x		x			x
3.26	HIPAA	X	x	x	x	X	x	x
3.27	Advice System	X						
3.28	Manual Upload				x			x
8.1	Data Transport Encryption							
8.2	Authorization	X	x		x			x
8.3	Integrity	X	x	x	x	X	x	x
8.4	Storage Encryption							
8.5	Disposal Guarantee							
8.6	WLAN Connection Security							
8.7	Data Backup							

2.4.2 Transport Layer Data Flow

Table 2 - 10: Transport Layer Data Flow

Data Flow	Description
T1	HttpConnection Object that contains the user request, which has now been encrypted
T2	HTTPServletRequest and HTTPServletResponse Object
T3	HTTPServletRequest and HTTPServletResponse Object
T4	HTTPServletRequest and HTTPServletResponse Object
T5	HTTPServletRequest and HTTPServletResponse Object (with error page if invalid)
T6	HTTPServletRequest and HTTPServletResponse Object
T7	Web Page or data to be decrypted
T8	Data that has been formatted in JSON
T9	Web page of which the browser will redirect to or data (now decrypted) from initial request
T10	Boolean response of true or false

2.4.2.1 Transport Layer Requirement Mapping

Table 2 - 11: Transport Layer Requirement Mapping

Req. No.	Requirement Name	Request Encryption	Response Decryption	Client/Server Comm.	Response Encrytion	Request Decryption	Permission Auth.
3.5	User Authentication	X	x	x	x	x	x
3.7	View Results						
3.8	Multiple Readings						
3.9	Result Transmission and Storage	X	x	x	x	x	x
3.1	Web Service	X	x	x	x	x	x
3.11	Account Types						x
3.12	Doctor Account Creation	X		x		x	x
3.13	Patient Account Creation	X		x		x	x
3.14	Patient's Doctor						x
3.15	Doctor Limit						x
3.16	Doctor Editing	X		x		x	x
3.17	Patient Editing	X		x		x	x
3.18	View Charts and Graphs			x			x
3.19	Annotations			x			x
3.2	Patient Viewing Profile			x			x
3.21	Create Alerts			x			x
3.22	View Alerts			x			x
3.23	Messaging System			x			x
3.24	Multiple Users						x
3.25	Login to Access Data			x			x
3.26	HIPAA	X	x	x	x	x	x
3.27	Advice System			x			x
3.28	Manual Upload			x			x
8.1	Data Transport Encryption	X	x		x	x	
8.2	Authorization	X		x			x
8.3	Integrity	X	x	x	x	x	x
8.4	Storage Encryption						
8.5	Disposal Guarantee						
8.6	WLAN Connection Security	X	x	x	x	x	
8.7	Data Backup						

2.4.3 Business Logic Layer Data Flow

Table 2 - 12: Business Logic Layer Data Flow

Data Flow	Description
BS1	StorageDataPool object
BS2	AnalysisDataPool object
BS3	AnalysisDataPool object
BS4	AnalysisCommand object
BS5	AnalysisController object
BS6	AnalysisDataPool object (same sent in)
BS7	HTTPServletRequest, HTTPServletResponse, DataController, and AnalysisController objects
BS8	Response data (if new page not required)
BS9	HTTPServletRequest and HTTPServletResponse objects
BS10	Translated web page

2.4.3.1 Business Logic Layer Requirement Mapping

Table 2 - 13: Business Logic Layer Requirement Mapping

Req. No.	Requirement Name	Request Delegation	Analysis Controller	Analysis Command	Data Preparer	Page Generator
3.5	User Authentication	x	x	x	x	x
3.7	View Results					
3.8	Multiple Readings					
3.9	Result Transmission and Storage		x	x	x	x
3.1	Web Service	x	x	x	x	x
3.11	Account Types	x				
3.12	Doctor Account Creation	x			x	x
3.13	Patient Account Creation	x			x	x
3.14	Patient's Doctor	x				
3.15	Doctor Limit	x				
3.16	Doctor Editing	x			x	x
3.17	Patient Editing	x			x	x
3.18	View Charts and Graphs	x			x	x
3.19	Annotations	x			x	x
3.2	Patient Viewing Profile	x			x	x
3.21	Create Alerts	x	x	x	x	x
3.22	View Alerts	x			x	x
3.23	Messaging System	x			x	x
3.24	Multiple Users	x				
3.25	Login to Access Data	x			x	
3.26	HIPAA	x			x	x
3.27	Advice System	x	x	x	x	x
3.28	Manual Upload	x				
8.1	Data Transport Encryption					
8.2	Authorization	x				
8.3	Integrity	x	x	x	x	x
8.4	Storage Encryption					
8.5	Disposal Guarantee					
8.6	WLAN Connection Security					
8.7	Data Backup					

2.4.4 Data Storage Layer Data Flow

Table 2 - 14: Data Storage Layer Data Flow

Data Flow	Description
DS1	Database command object
DS2	Database command object w/encrypted data
DS3	SQL Statement
DS4	ResultSet Object
DS5	Database Command object with encrypted result
DS6	Database command object w/encrypted data
DS7	SQL Statement
DS8	ResultSet Object
DS9	Database Command object with encrypted result
DS10	Updated Database command object
DS11	StorageDataPool object
DS12	StorageDataPool object

2.4.4.1 Data Storage Layer Requirement Mapping

Table 2 - 15: Data Storage Layer Requirement Mapping

Req. No.	Requirement Name	Data Controller	Cryptography	DB Get	DB Update
3.5	User Authentication	x	x	X	X
3.7	View Results				
3.8	Multiple Readings				
3.9	Result Transmission and Storage	x	x		
3.1	Web Service	x	x	X	X
3.11	Account Types	x			
3.12	Doctor Account Creation	x	x		X
3.13	Patient Account Creation	x	x		X
3.14	Patient's Doctor	x			
3.15	Doctor Limit	x			X
3.16	Doctor Editing	x	x	X	X
3.17	Patient Editing	x	x	x	X
3.18	View Charts and Graphs	x		x	
3.19	Annotations	x	x		X
3.2	Patient Viewing Profile	x		x	
3.21	Create Alerts	x	x		X
3.22	View Alerts	x		x	
3.23	Messaging System	x		x	X
3.24	Multiple Users	x		x	X
3.25	Login to Access Data	x		x	
3.26	HIPAA	x	x	x	X
3.27	Advice System	x		x	X
3.28	Manual Upload	x	x		X
8.1	Data Transport Encryption				
8.2	Authorization	x	x		
8.3	Integrity	x	x	x	X
8.4	Storage Encryption		x		
8.5	Disposal Guarantee	x	x		
8.6	WLAN Connection Security				
8.7	Data Backup	x	x	x	X

3. Test Items

3.1 Overview

The following section will provide details about how the team plans to test the system. Starting with the hardware, going to unit tests, and then gradually moving up to include more components at every stage, the team will ensure that each portion of the system is working individually and in collaboration with each other to ensure that the system is functioning properly, and to verify that the system meets the standards set forth at the beginning of the project.

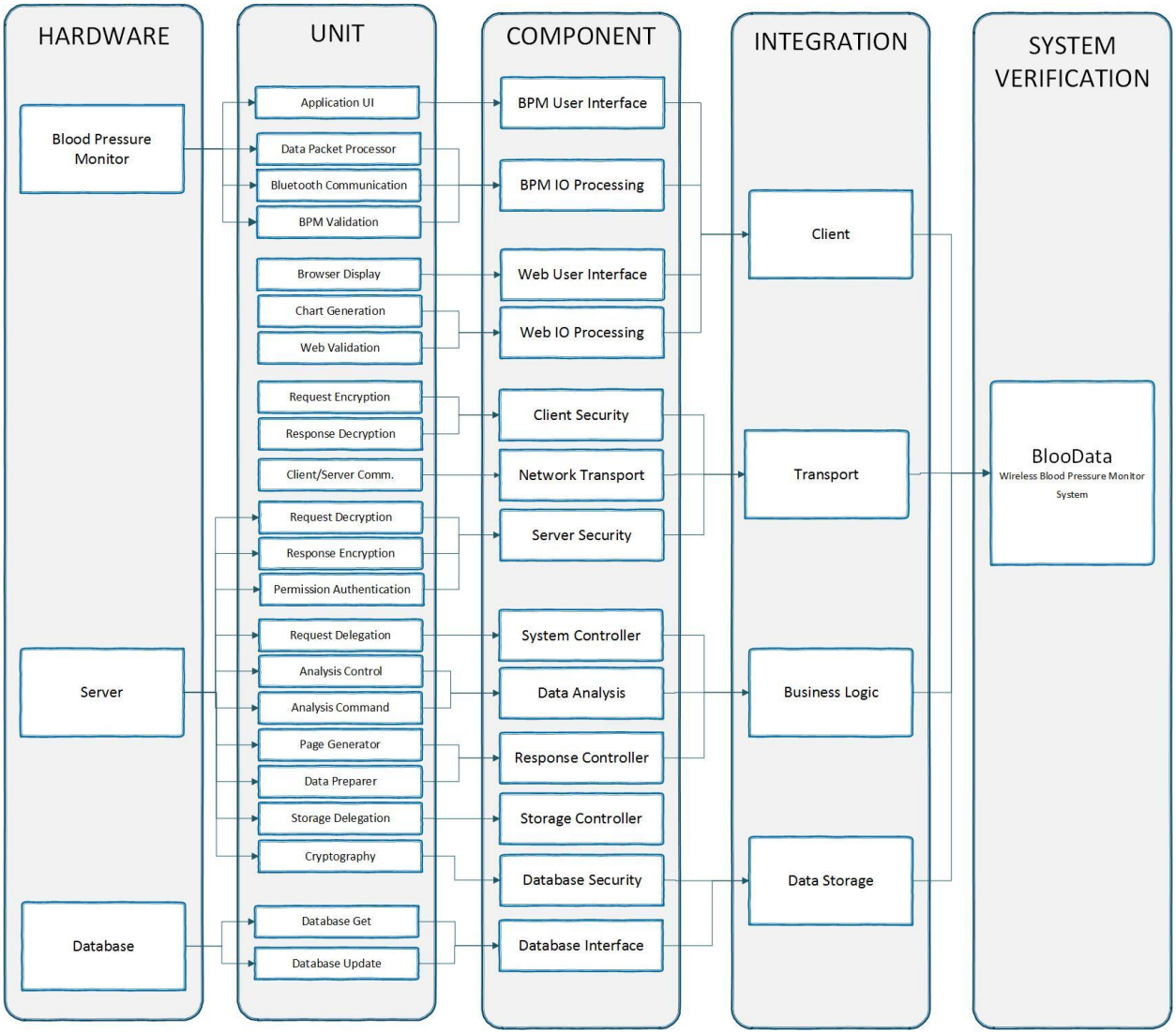

Figure 3 - 1: Relational Diagram

3.2 Hardware Tests

Table 3 - 1: Hardware Tests

ID	Hardware	Input	Output	Test	Risk
H1	BPM	• Start button	• Blood pressure reading	Push start button, confirm that BPM initiates and completes a blood pressure session.	High
H2	BPM	• Send button	• Data packet	Push send button, data packets are sent to the android application.	High
H3	Database	• SQL statement	• ResulSet • Sql.exception	Execute SQL statement, confirm that data is stored and retrieved successfully	High
H4	Server	• Http Request	• Http Response	Send page request, confirm that sever has responded	High

3.3 Unit Tests

3.3.1 Client Layer Unit Test

Table 3 - 2: Browser Display Unit Tests

ID	Module	Input	Output	Test	Risk
CBD1	Browser Display	• Mouse	• Correct web page • animation	Click hyperlink	High
CBD2	Browser Display	• Keyboard	• Text display	Type into form fields	High
CBD3	Browser Display	• Mouse	• Correct web page • Animation	Click submit button	High
CBD4	Browser Display	• Mouse	• Animation	Click element on web page	High

Table 3 - 3: Web Validation Unit Tests

ID	Module	Input	Output	Test	Risk
CWV1	Web Validation	• Form elements	• Boolean false if invalid • Boolean true if valid	Input invalid data into field, submit Input valid data into field, submit	High

Table 3 - 4: Chart Generation Unit Tests

ID	Module	Input	Output	Test	Risk
CCG1	Chart Generation	• JSON objects • JSON arrays	• Chart element based on patient data	Request graph for particular patient	High

Table 3 - 5: Application UI Unit Tests

ID	Module	Input	Output	Test	Risk
CAU1	Application UI	• Touch input	• Correct views • Animation	Touch elements on the android screen	High
CAU2	Application UI	• Keyboard	• Text display	Type into text fields and input is displayed on the android screen	High
CAU3	Application UI	• Touch input	• Text dialog	Click send data button, and responds with a success or fail message	High

Table 3 - 6: BPM Validation Unit Tests

ID	Module	Input	Output	Test	Risk
CBV1	BPM Validation	• Text field	• Error message if invalid • None if valid	Input invalid data into field, submit Input valid data into field, submit	High

Table 3 - 7: Data Packet Processing Unit Tests

ID	Module	Input	Output	Test	Risk
CDPP1	Data Packet Processing	• Byte packet	• Hexadecimal integer representing the byte	Input Valid byte packets, and verify translation to hexadecimal is accurate Input Invalid byte packets, and verify that the invalid bytes are handled correctly and proper message is displayed	High

Table 3 - 8: Bluetooth Communication Unit Tests

ID	Module	Input	Output	Test	Risk
CBC1	Bluetooth Communication	• Byte stream	• Byte packet	• Send data to Bluetooth, and verify the byte stream has been properly packaged	High

3.3.2 Transport Layer Unit Test

Table 3 - 9: Request Encryption Unit Tests

ID	Module	Input	Output	Test	Risk
TRQE1	Request Encryption	• HttpConnection object	• Encrypted HttpConnection object	Send HTTP Request, check request packets on client	Medium

Table 3 - 10: Response Decryption Unit Tests

ID	Module	Input	Output	Test	Risk
TRSD1	Response Decryption	• Response Decryption	• Web page or data to be decrypted	Send HTTP Response, check response packets on client	Medium

Table 3 - 11: Client/Server Communication Unit Tests

ID	Module	Input	Output	Test	Risk
TCSC1	Client/Server Communication	• Encrypted HttpConnection object	• Encrypted web page or data to be sent to the client	Send HTTP request, make sure server receives request, and make sure client receives HTTP response	Medium

Table 3 - 12: Request Decryption Unit Tests

ID	Module	Input	Output	Test	Risk
TRQD1	Request Decryption	• Encrypted HTTPServletRequest object • HTTPServletResponse object	• Unencrypted HTTPServletRequest object • HTTPServletResponse object	Send HTTP Request, check request packets on server	Medium

Table 3 - 13: Permission Authentication Unit Tests

ID	Module	Input	Output	Test	Risk
TPA1	Permission Authentication	• Unencrypted HTTPServletRequest object • HTTPServletResponse object	• Forwards to Request Delegation • Error page	• Send request with valid role • Send request with invalid role	High

Table 3 - 14: Response Encryption Unit Tests

ID	Module	Input	Output	Test	Risk
TRSE1	Response Encryption	• HTTPServletRequest object • HTTPServletResponse object	• Encrypted HTTPServletRequest object • HTTPServletResponse object	Send HTTP Response, check response packets on server	Medium

3.3.3 Business Logic Layer Unit Test

Table 3 - 15: Request Delegation Unit Tests

ID	Module	Input	Output	Test	Risk
BLRD1	Request Delegation	• Httpservletrequest object • Httpservletresponse object	• DataController • AnalysisController objects.	Send HTTP Request, ensure correct servlet being accessed	High
BLRD2	Request Delegation	• Httpservletrequest • Httpservletresponse object	• DataPool object(s)	Send HTTP Request, ensure correct DataPools being successfully constructed	High

Table 3 - 16: Analysis Controller Unit Tests

ID	Module	Input	Output	Test	Risk
BLACL1	Analysis Controller	• AnalysisDataPool object	• AnalysisCommand objects	Send AnalysisDataPool, ensure that correct command objects are built based on patient data	High
BLACL2	Analysis Controller	• AnalysisDataPool object	• AnalysisController object	Send AnalysisDataPool, ensure that results being stored correctly	High

Table 3 - 17: Analysis Command Unit Tests

ID	Module	Input	Output	Test	Risk
BLACM1	Analysis Command	• AnalysisDataPool object	• AnalysisCommand object with set result attribute	Send AnalysisDataPool, ensure that result is correct based on data	Medium

Table 3 - 18: Data Preparer Unit Tests

ID	Module	Input	Output	Test	Risk
BLDP1	Data Preparer	• AnalysisController • DataController	• JSON Object • JSON Array	Send AnalysisController and DataController, ensure that objects are formatted correctly	High
BLDP2	Data Preparer	• AnalysisController • DataController • Httpservletrequest • Httpservletresponse objects	• Web page • JSON Objects	Send AnalysisController, DataController, Httpservletrequest, and Httpservletresponse objects, ensure that correct page or data is dispatched	High
BLDP3	Data Preparer	• AnalysisController • DataController • Httpservletrequest • Httpservletresponse objects	• Httpservletrequest • Httpservletresponse	Send AnalysisController, DataController, Httpservletrequest, and Httpservletresponse objects, ensure that data is being loaded into the response correctly	High

Table 3 - 19: Page Generator Unit Tests

ID	Module	Input	Output	Test	Risk
BLPG1	Page Generator	• Httpservletrequest • Httpservletresponse objects	• JSP page	Send Httpservletrequest and Httpservletresponse objects, ensure that pages are correctly translated	High

3.3.4 Data Storage Layer Unit Test

Table 3 - 20: Data Controller Unit Tests

ID	Module	Input	Output	Test	Risk
DSDC1	Data Controller	• StorageDataPool object	• DataCommand objects	Send StorageDataPool, ensure that correct command objects are built	High
DSDC2	Data Controller	• StorageDataPool object	• DataController object	Send StorageDataPool, ensure that results being stored correctly	High

Table 3 - 21: Cryptography Unit Tests

ID	Module	Input	Output	Test	Risk
DSC1	Cryptography	• DataCommand object	• DataCommand object with encrypted data	Send DataCommand object to store data, ensure that data is encrypted properly	Medium
DSC2	Cryptography	• DataCommand object	• DataCommand object with decrypted data	Send DataCommand object to get PHI data, ensure that data being returned is properly decrypted	Low

Table 3 - 22: Database Get Unit Tests

ID	Module	Input	Output	Test	Risk
DSDG1	Database Get	• StorageDataPool object	• DataCommand object	Send StorageDataPool, ensure that data is being retrieved successfully	High

Table 3 - 23: Database Update Unit Tests

ID	Module	Input	Output	Test	Risk
DSDU1	Database Update	• StorageDataPool object	• DataCommand object	Send StorageDataPool, ensure that data is being stored successfully	High

3.4 Component Tests

3.4.1 Client Layer Component Test

Table 3 - 24: Client Layer Component Test

ID	Subsystem	Input	Output	Test	Risk
CWU1	Web UI	• User input	• Updated display	Check that subsystem response correctly to user actions	High
CWI1	Web IO Processing	• HTML elements	• Boolean True/False	Check that invalid data is not being accepted	High
CWI2	Web IO Processing	• JSON Object	• Chart element	Check that charts are being generated	High
CBU1	BPM UI	• User input	• Updated display	Check that subsystem response correctly to user actions	High
CBI1	BPM IO Processing	• User input • BPM packets	• Boolean True/False	Checks that all data coming in from user and from BPM is valid	High

3.4.2 Transport Layer Component Test

Table 3 - 25: Transport Layer Component Test

ID	Subsystem	Input	Output	Test	Risk
TCS1	Client Security	• Http Request	• HTTP Response	Check that data going to server is encrypted, and data going to client is decrypted	Medium
TNT1	Network Transport	• Http Request	• HTTP Response	Check that data is being sent between the client and the server	High
TSS1	Server Security	• Http Request	• HTTP Response	Check that data going to server is decrypted, and data going to client is encrypted	Medium

3.4.3 Business Logic Layer Component Test

Table 3 - 26: Business Logic Layer Component Test

ID	Subsystem	Input	Output	Test	Risk
BLSC1	System Controller	• Httpservletrequest • Httpservletresponse objects	• Function calls to other subsystems	Check that the subsystem is making calls to other subsystems appropriately	High
BLDA1	Data Analysis	• AnalysisDataPool object	• AnalysisController object	Check that analysis results match the input data	High
BLRC1	Response Controller	• AnalysisControll • DataController • Httpservletrequest Httpservletresponse objects	• Web page or JSON object	Check that the server is responding with the appropriate web page or data	High

3.4.4 Data Storage Layer Component Test

Table 3 - 27: Data Storage Layer Component Test

ID	Subsystem	Input	Output	Test	Risk
DSSC1	Storage Controller	• StorageDataPool object	• DataController object	Check that correct data and database response messages are being received	High
DSDS1	Database Security	• DataCommand object	• Encrypted DataCommand object	Check that data being stored is encrypted, and data being retrieved is decrypted	High
DSDI1	Database Interface	• StorageDataPool object	• ResultSet • Error message	Check that data is being stored and retrieved correctly	High

3.5 Integration Tests

Table 3 - 28: Integration Tests

ID	Features	Input	Output	Test	Risk
I1	Account Creation	• Form with account information	• User home page	Tester will enter account information. System should validate input, send to server, store data, and respond with the account's home page	High
I2	Advice	Advice button click	User advice page	Tester will log into system and click on Advice button. System will send request to server, check that tester has role permission for that page, gather tester's data, analyze the data, and send back advice.	High
I3	Blood Pressure Reading	• Login credentials • Bluetooth address • BP reading	• Login success message (or error) • Pairing success message • BP reading • Graph	Tester will login on the system's application. They will then click the Start button on the BPM, take their blood pressure, pair the BPM with their phone, send the data to their phone, and then transmit the data to the server. Finally, the tester will login on the website, click on the Graphs link, and see the new reading added to their latest charts.	High

3.6 System Validation Tests

Table 3 - 29: System Validation Tests

ID	Requirements	Output	Test	Risk
V1	• 3.5 • 3.6 • 3.7 • 3.8 • 3.9 • 3.10 • 3.21 • 3.22 • 8.1 • 8.2 • 8.3	• BP reading • Transmission message • Health alert (optional)	Tester will take a blood pressure reading, and send it to their phone and then to the server. If necessary, an alert will be sent to the doctor's profile. Tester will be able to see result on the BPM and the website. The doctor will see an alert for the tester if generated.	High
V2	• 3.5 • 3.10 • 3.11 • 3.12 • 3.13 • 3.14 • 3.16 • 3.17 • 8.1 • 8.2 • 8.3 • 8.4	• User home page • Error message	Tester will create a doctor account if a doctor, or finish creating a patient account if a patient (doctor will begin setting up patient accounts). Tester will then go to the Edit Account page, and edit information about the account. Tester will verify that the account data has been encrypted and stored in the database.	High
V3	• 3.5 • 3.10 • 3.18 • 3.19 • 8.1 • 8.2 • 8.3	• Chart page	Tester will login to their account and click on the Chart link. Tester will then select various graphs and ensure that they are accurate. Tester will select a reading on the chart, enter an annotation, and submit. The tester should then see the annotation when selecting that reading.	High
V4	• 3.5 • 3.10 • 3.27 • 8.1 • 8.2 • 8.3	• Advice page	Tester will login to their account and click on the Advice link. System will then gather the tester's health data and analyze it for any concerning trends or readings. The tester will then see the results of the analysis.	High

V5	• 3.6 • 3.7 • 3.8 • 5.8	• Pass/Fail	Tester will take multiple readings to confirm that the BPM is precise in its results. The tester will then take blood pressure readings on other BP monitors to ensure that the system BPM's readings is accurate.	High
V6	• 3.5 • 5.4 • 5.5 • 5.6 • 8.1 • 8.2	• Time taken	Tester will pair the BPM with an Android phone and time how long it takes to connect. Tester will then login to their account and time how long it takes to successfully authenticate.	Low

4. Risks

4.1 Overview

Testing is meant to help find and fix problems in a project so that the final product is functional and acceptable to users. However, problems can occur during the process of testing, which can be just as bad as the problems that it intends to fix. In order to avoid causing more issues and possibly doing irreversible damage to the product, the team has identified certain risks that need to be addressed and avoided as much as possible. The following section will discuss some of these risks.

4.1 Table of Risks

Each risk has been assigned a severity rating of high, moderate, or low. Severe risks have the possibility of hindering the testing process (by preventing the team from testing certain parts of the systems). More moderate risks will slow down testing to a degree, but can be worked around. Risks with little severity will most likely have little impact on the testing process.

Table 4 - 1: Risk Table

Risk	Impact	Severity	Management Plan
BPM stops functioning properly	BPM testing will be halted, could affect the delivery of the final product.	High	Only perform test that comply with the restrictions in the user's manual.
Server no longer functions	Testing can be done locally, but integration becomes a problem, and delivery of the final product is put at risk	High	Code carefully and make sure server is in the proper environment.
Fixing errors causes further problems	Testing on that particular portion of the system cannot be completed until a solution is reached. Regression testing will need to be done, possibly more than once.	Moderate	Carefully regression testing whenever a change is made
Impossible to fully test security	Problems may exist that, despite the team's best efforts, slipped through the tests. Will not always affect the testing or the delivery, but may affect the acceptability of the product.	Low	Create tests that check that security is adequate for this project
All planned tests do not get run	As with the previous risk, the acceptability of the final product may be called into question.	Low	Track all create tests and assign specific people to perform the tests

5. Features to Be Tested

5.1 Overview

The following features have been derived from the System Requirement Specification, and will be thoroughly tested for compliance and to ensure the system has satisfied both function and nonfunctional requirements. These sections will cover what feature is to be tested a brief description requirement and the approaches to testing the requirement as well as the risk level associated with the failure of the test.

5.2 Customer Requirements

5.2.1 Non-Invasive Measurements

Description: The blooData device will be able to take blood pressure measurement in a non-invasive method.

Risk: High

Test Approach: The team and sponsor will do multiple blood pressures reading session. After which we and the sponsor will evaluate the invasiveness of the cuff.

5.2.2 Oscillometric Method

Description: The blooData device will use the oscillometric method to evaluate blood pressure

Risk: High

Test Approach: The team before construction of the device will ensure that the blood pressure cuff used in the construction of the blooData device has implemented oscillometric measuring methods for blood pressure.

5.2.3 Automatic Inflation/Deflation

Description: Blood Pressure Cuff will automatically inflate and deflate.

Risk: High

Test Approach: The team will test the blooData device upon initial request to take a blood pressure reading will automatically inflate/deflate without any further instructions.

5.2.4 Cuff Location

Description: The Blood Pressure Cuff will be worn on the upper or lower arm.

Risk: High

Test Approach:

5.2.5 User Authentication

Description: An authentication system will be used before all sessions to ensure user security.

Risk: High

Test Approach: The team will run multiple test cases on the blooData application to ensure that the system will require the user to login before starting a blood pressure reading session.

5.2.6 Blood Pressure Reading

Description: An authentication system will be used before all sessions to ensure user security.

Risk: High

Test Approach: The team will run multiple test cases on the blooData application to ensure that the system will require the user to login before starting a blood pressure reading session.

5.2.7 View Results

Description: Users will be able to view their results on the blooData application

Risk: High

Test Approach: The team will start a blood pressure reading session, take multiple blood pressures, send them to the blooData application, and verify that the readings are properly displayed on the applications.

5.2.8 Multiple Readings

Description: Users will be able to record and submit multiple readings in a session

Risk: High

Test Approach: The team will start a blood pressure reading session. Upon completion of the initial reading the system will allow for a second blood pressure reading. After multiple blood pressure readings have been completed, the user will be able to submit all the readings to the blooData application.

5.2.9 Result Transmission and Storing

Description: The blooData application will transmit the blood pressure results to the website's database for storage.

Risk: High

Test Approach: The team will execute multiple blood pressure reading sessions, and manually submit them. Upon submission we will verify that all sessions have been properly saved to the database.

5.2.10 Account Types

Description: The web service will support two types of accounts: patients and doctors.

Risk: High

Test Approach: The team will run multiple test to manually and automatically create doctors and patient accounts and ensure that website will properly create and register the accounts into the database.

5.2.11 Account Editing

Description: Profile information (email, password, full name) can be edited by the given patient. Medical information will be edited by the patient's doctor.

Risk: High

Test Approach: The team will create new accounts and login to existing account and edit personal profile information and verify that the changes to the account has been properly changed in the database.

5.2.12 View Charts and Graphs

Description: All registered users on the site will be able to view various charts and graphs on the website that describe the patient's blood pressure conditions, overall health, and other factors, assuming that they have access to said patients information.

Risk: High

Test Approach: The team will login to existing account and click on the charts web page and view the blood pressure reading displayed in charts/graph format. We will verify that the readings displayed on the charts are valid and are properly labeled and marked on the charts/graph.

5.2.13 Annotations

Description: Patients registered on the site will be able to make notes about particular readings of theirs on the website, to relay information such as changes in their health or lifestyle.

Risk: High

Test Approach: The team will login to existing account and click on the charts web page. Select a blood pressure reading chart, select a blood pressure reading, and input a unique annotation to the blood pressure reading. The process will be repeated multiple time and after finish of annotation, the team will click each of the readings and validate that each of the annotations have been properly saved and linked to the correct reading.

5.2.14 Create Alerts

Description: Alerts can be created, either automatically by the system if it detects a problem or a concerning trend, or by a patient if they want to alert their doctor about something, which will be sent to the doctor for later viewing.

Risk: High

Test Approach: The team will run multiple manually insert multiple blood pressure readings into a patient account half of which will be “healthy” and “unhealthy” readings. Upon insertion of the blood pressure reading the team will login to the doctor account and verify that the system has alerted the doctor account on the “unhealthy” readings of that patient. Then the team will also login to the patient account and be able to set various alert on “healthy” readings.

5.2.15 Advice System

Description: The website will be able to give patients health information, advice, and general suggestions to keep their blood pressure in check by looking at general patterns and user information.

Risk: High

Test Approach: The team will enter multiple blood pressure readings into an account, login to the account and request advice. Upon receiving advice notification from our system the team will verify the semantics of the advice according to the medical data inputted into the account.

5.3 Package Requirements

5.3.1 Blood Pressure Cuff

Description: The product will be packaged with a standard blood pressure cuff. The cuff will weigh between ½ -1 pound, and will be fitted for arm circumference of 9-14 inches.

Risk: High

Test Approach: The team will weigh the cuff and verify that the weight is with ½ -1 pound, and fit the cuff on various arms of members on our team and a few selected individual to cover all arm width.

5.3.2 Android Application

Description: An android application will be provided via Google play store. The application will allow the user to interact with the blood pressure cuff to authenticate, take blood pressure reading, and send the readings to the website.

Risk: High

Test Approach: The team will install the android application on multiple phones. After which we will initiate multiple tests by login to the android application, take multiple blood pressure reading with the application, and send the readings from the application to the server. After which the team will verify that the readings have been properly saved into the database.

5.3.3 Software Packaging

Description: All software will be installed prior to packaging. No set up will be required by the Patient.

Risk: High

Test Approach: The team will have the sponsor test our system by creating an account on our web services and download our application. From which we will have the sponsor take

5.3.4 User Manual

Description: A user manual will be provided with the product. The manual will contain detailed information regarding the product, website, and how to operate the device.

Risk: Low

Test Approach: The team will verify that the device used in storing the manual has properly stored the documentation by manual extracting the manual from the device. Verify that it is indeed our documentations and is the correct version of release.

5.4 Performance Requirements

5.4.1 Web Authentication Latency

Description: The android application interface to the end-user's wireless local area network (WLAN) will take no longer than 10 seconds to attempt a successful connection. After 10 seconds, the android application will cease the connection activity.

Risk: High

Test Approach: The team will execute multiple connection tests in different area and time the time it takes to connect to the wifi in that region.

5.4.2 WLAN Connection

Description: The android application interface to the end-user's wireless local area network (WLAN) will take no longer than 10 seconds to attempt a successful connection. After 10 seconds, the blood pressure monitor will cease the connection activity and notify the end-user of the current status of the WLAN connection.

Risk: Medium

Test Approach: The team will execute multiple connection tests in different area and time the time it takes to connect to the wifi in that region. As well as test the android application in definite non connectable are and verify that the application properly notify the user of connection failure.

5.4.3 Bluetooth Connection

Description: The android application interface to the BPM will take no longer than 10 seconds to attempt a successful connection. After 10 seconds, the android application will cease the connection activity and notify the end-user of the current status of the Bluetooth connection.

Risk: Medium

Test Approach: The team will execute multiple connection tests in different area and time the time it takes to connect to the wifi in that region. As well as test the android application in definite non connectable are and verify that the application properly notify the user of connection failure.

5.4.4 Blood Pressure Cuff Accuracy

Description: The blood pressure measurements obtained using the monitor shall not deviate more than 10% at any measurement range from an existing commercial blood pressure monitor.

Risk: High

Test Approach: The team will run multiple test sessions on the blood pressure cuff. Each reading will be compared to another standard blood reading machine (or manual method) and ensure that the cuff have a consist, accurate, and reasonable reading.

5.5 Safety Requirements

5.5.1 Blood Pressure Cuff Mechanical Safety

Description: he blood pressure monitor, primarily the wearable components, shall not be able to easily induce physical injury to the user through its physical form.

Risk: High

Test Approach: The team will each individually wear the cuff, and evaluate the quality of the comfort.

5.5.2 Cuff Pressure Release

Description: An easily accessible and clearly labeled means of allowing the user to deflate the cuff shall be provided. During the rapid exhaust of the pneumatic system with fully-opened valve, the time for the pressure reduction from 260 mmHg to 15 mmHg shall not exceed 10 s.

Risk: High

Test Approach: The team will run multiple reading sessions and prematurely stop the cuff before its full pressure point, and after its full pressure point.

5.6 Maintenance and Support Requirements

5.6.1 Web Service Software Portability

Description: The web service software shall be able to be deployed via an automated script that sets up the server software (e.g. Apache, MySQL, etc.) and preloads system templates such as database schemas and defaults. This is intended to ease deployment onto the client's intended web servers.

Risk: High

Test Approach: The team will package the web service software, database script, and server configurations. We will then load them into different machines and test to validate that each machine can replicate our service on them.

5.7 Security Requirements

5. 7.1 Integrity

Description: The system will attempt to prevent any PHI that will be collected and stored from the user from unauthorized modification or corruption and ensure its validity. This requirement may rely on storage encryption (section 8.4) and authorization protocols (section 8.2) to deter unauthorized modification.

Risk: High

Test Approach: The team will test to make sure each data inputted into the database will be the same as when it's retrieved, for both unencrypted and encrypted insert/update's.

5. 7.2 Authorization

Description: The collected PHI data will only be accessible to authorized personnel using audited access controls. Mechanisms will be designed and implemented to ensure that data can be accessed only by appropriate users. This requirement may rely on storage encryption (SRS section 8.4) to prevent unauthorized access, as well as on the web hosting vendor to comply with the HIPAA Business Associate (section 8.7) to maintain control over low-level access to the web server's functionality.

Risk: Critical

Test Approach: The team will test access restriction of each user type (guest, patient, and doctor). Each user type will test its individual functionally and ensure each user type has access to their specific user type.

5. 7.3 Storage Encryption

Description: All PHI data that is being stored or archived will be encrypted and intended for storage of a minimum of five years. In general, the data will be encrypted using symmetric encryption of at least AES-128 strength. The encrypted data can only be decrypted and accessible by users with the appropriate keys. The keys will further be protected using a form of public key encryption. This requirement would also further enforce the protocols of authorization (section 8.2) on PHI data.

Risk: High

Test Approach:

5.7.4 Disposal Guarantee

Description: PHI data, when found necessary to remove from the system, will be effectively purged. This guarantees that when the system removes the data from its oversight, it would be thoroughly deleted from itself and all of the system's underlying dependencies. If a data backup system is implemented (section 8.7), then this requirement will be implemented completely on this system as well. This means that the backups must be thoroughly cleansed of or recompiled without the appropriate data.

Risk: Low

Test Approach: The team will delete all records of the specified patient/doctor who wish to be removed from our record. Each record corresponding to the patient/doctor will be deleted from the database and the team wills manual check the database to ensure no information is missed during the cascading delete.

5.7.5 Data Transport Encryption

Description: The web service shall encrypt all data, particularly PHI, that is transmitted between the blood pressure monitor and the data collecting web server. This requirement is primarily focused on the level of the communication between monitor-client and the web server and not the WLAN connection (see section 8.5).

Risk: High

Test Approach: The team will ensure the data is sent through HTTPS and all packets will be check for encryption with a 3rd party tool.

5.8 Other Requirements

6.8.1 Web Browser Compatibility

Description: The web interface shall be accessible and usable through popular web browsers for the user, caretaker/physician, and administrators. Such web browsers would include Internet Explorer, Mozilla Firefox, Google Chrome, and Safari.

Risk: Low

Test Approach: The team will visit the various pages of the website using different browsers, and ensure that the key features of the website are functioning and displaying as they were meant to. The animations and other features will be tested as much as possible, although guaranteeing that every item works for all supported browsers will be almost impossible.

5.8.2 Web Service Source Compatibility

Description: The source code of the web service functionality shall be compatible with Windows 7 and popular Linux development environments for implementation, deployment, and maintenance.

Risk: High

Test Approach: The team shall load the source code on to different computers with different OS(including Window 7 and Linux) and ensure that the source code can be properly modified and ran in the given environment.

5.8.3 Smartphone Device Constraint

Description: Any implementation of functionality for mobile devices, mainly smartphones, will be constrained to Android 3.0+ and Unix-type operation systems. This is due to their accessibility and availability of their tools. The choice of Android 3.0 and up is due to their available and compatible feature set for this project.

Risk: Medium

Test Approach: The team will test install the android application on multiple android devices. The devices will each run android version 3.0+ and the team will ensure that the application will work on all smart phone devices(for advance and new android devices this test will not be applicable).

6. Features Not To Be Tested

6.1 Overview

Despite the team's best efforts, portions of the project will either not be delivered or will not be able to be tested, due to time and budget constraints. Certain features were set aside for a future release date in order to ensure that the rest of the project was fully functional and acceptable to users. Some components of the project cannot be tested because failing the test would leave the team in a worse position than simply not testing them. Some features cannot be fully tested due to their complexity or because they rely partially on how the user handles the product. The following section details the features that will not be tested in the product to be delivered.

6.2 Customer Requirement

6.2.1 Patient Viewing Profile

Description: All patients will have a viewing profile, a way in which friends, family members, other doctors, or anyone else the patient wishes to give access to can view how the patient is doing and other information. This profile is separate from the standard Patient profile, it will be tied to the account with a separate set of login credentials, and it is extremely limited in functionality. The patient will be able to change the login credentials for this at any time to restrict access.

Reason: This requirement will not be part of the upcoming release of the project therefore cannot be tested.

6.2.2 Messaging System

Description: Doctors and patients can send messages to one another to be read later (like an email system), as a way of addressing things not covered by the alert system and the annotation system.

Reason: This requirement will not be part of the upcoming release of the project there fore cannot be tested.

6.2.3 HIPAA

Description: Product will comply with all HIPAA standards (as well as other standards that apply to medical data) to ensure customer privacy. This will be covered in detail in the Security Requirements section.

Reason: This requirement will not be fully tested because of the time constraint of the project as well as full compliance with HIPPA standard cannot be met with our current development time.

6.2.4 Manual Upload of Information

Description: In the case where a patient's wireless connection goes down or begins to suffer problems, the device will store the patient's readings for a set amount of time. The patient will then be able to manually upload the readings using a computer, if they feel that the wireless connection will not resume working in the near future.

Reason: This requirement will not be part of the upcoming release of the project there fore cannot be tested.

6.3 Package Requirement

6.3.1 Blood Pressure Monitor

Description: The product will come packaged with a monitor which is attached to the cuff. It will be the interaction point between the patient and the system. The patient can authenticate, take blood pressure reading, view current readings in the session, and send blood pressure readings to the website. The Monitor will be a 3 x 4 inch display and will weigh no more than 6 ounces.

Reason: This requirement will not be tested due a design choice of the team to construct a android application instead of building a monitor.

6.3.2 Android Phone

Description: The product will require an Android phone, which will not be packaged with the product. The phone will use the Android application to interact with the blood pressure monitor device.

Reason: This requirement will not be tested because this requirement is for required by the user to use our product.

6.3.3 Battery

Description: The device will be operated via battery power. Batteries will not be packaged with the device.

Reason: This requirement will not be tested because this is a requirement directed toward our user.

6.3.4 Connection Cable

Description: Connection cable will be provided to manually upload the data on to a computer and uploaded to the website if wifi is not available.

Reason: This requirement will not be part of the upcoming release of the project there fore cannot be tested.

6.4 Performance Requirement

6.4.1 Storage conditions

Description: The blood pressure monitor shall maintain the performance requirements specified in this document after being stored for 24 hours at 21°C (70°F), give or take 10°C variance, and at a relative humidity lower than 90% (noncondensing).

Reason: This specific feature will be tested by everyday use within the normal environment of our test area. Any further climate testing we have chosen to withhold testing due to possible destruction of the advice and within our time and budget is not an option.

6.4.2 Operation Conditions

Description: The blood pressure monitor shall maintain the safety and performance characteristics specified in this document during operation over the following ranges of environmental conditions:

a) Temperatures between 11°C and 31°C;

b) Relative humidity under 90% (noncondensing)

Reason: This specific feature will be tested by everyday use within the normal environment of our test area. Any further climate testing we have chosen to withhold testing due to possible destruction of the advice and within our time and budget is not an option.

6.4.3 Operational Battery Life

Description: The blood pressure monitor shall maintain the performance characteristics specified in this document for at least 30 minutes while running the measurement acquisition process continuously.

Reason: This specific feature testing has been done by the manufacture due to budget and time constraint the team will not test this feature.

6.4.4 Blood Pressure Monitor Consistency Regarding Battery Life

Description: The blood pressure monitor will still be able to take consistent readings until the device detects that the battery's voltage is too low to continue normal operation. The variation in blood pressure readings shall be confined to within 15% accuracy.

Reason: This specific feature testing has been done by the manufacture due to budget and time constraint the team will not test this feature.

6.4.5 Blood Pressure Monitor Mechanical Robustness

Description: The blood pressure monitor shall be able to maintain its performance and safety requirements when being physically handled during normal use.

Reason: This specific feature testing has been done by the manufacture due to budget and time constraint the team will not test this feature.

6.4.6 Blood Pressure Monitor Startup Delay

Description: The blood pressure monitor shall be able to boot up from a cold start to a ready state for the user within 30 seconds.

Reason: This specific feature testing has been done by the manufacture due to budget and time constraint the team will not test this feature.

6.5 Safety Requirement

6.5.1 Maximum Cuff Pressure

Description: For devices intended for public, home, or other unsupervised use, a means of limiting cuff pressure shall be provided so that the maximum cuff pressure will never exceed 300 mmHg. For devices intended for use under professional supervision, a means of limiting cuff pressure shall be provided so that the maximum cuff pressure will never exceed 300 mmHg or 30 mmHg above the upper limit of the instrument's manufacturer-specified operating range, whichever is lower. In addition, the device shall incorporate means to ensure that cuff pressure will not be maintained above 15 mmHg for longer than 3 minutes (min).

Reason: This specific feature will be tested by everyday use within the normal environment of our test area. Any further climate testing we have chosen to withhold testing due to possible destruction of the advice and within our time and budget is not an option. As this requirement cannot be tested regressively we entrust the quality of this feature to the manufacture of the product with which we purchased the device from.

6.5.2 Eligible Users

Description: The blood pressure monitor shall not be intended for neonatal users. It will be only intended for those who understand how to operate the device safely.

Reason: This requirement will not be tested because this is a requirement directed toward our user and we cannot directly test who will be using our device after release of our product.

6.5.3 Electrical Safety

Description: The blood pressure monitor shall not be able to deliver a harmful shock to its user. The voltage range throughout the device will be limited to 12 volts. The device will also only use DC type power as its primary source of electrical power.

Reason: This specific feature testing has been done by the manufacture due to budget and time constraint the team will not test this feature.

6.5.4 Blood Pressure Monitor Mechanical Safety

Description: The blood pressure monitor, primarily the wearable components, shall not be able to easily induce physical injury to the user through its physical form.

Reason: This specific feature testing has been done by the manufacture due to budget and time constraint the team will not test this feature.

6.6 Maintenance and Support Requirement

6.6.1 Web Service Software Update Effects

Description: The web service will be able to be able to be modified without having to reboot the operating system or web server (e.g. Apache). There is no guarantee that the web application itself will not restart.

Reason: This requirement will not be tested because of by our design we will have separate version of code to both run on our server and modify on our local machine.

6.6.2 Web Service Administration Layer

Description: The web service shall have a portal for the system administrators to manage the basics of the web portal so that the source of the web application does not need to be modified and redeployed. This includes website-wide notifications/alerts, the ability to enable/disable certain services, and enabling/disabling the entire web application.

Reason: This requirement will not be part of the upcoming release of the project there fore cannot be tested.

6.7 Security Requirement

6.7.1 WLAN Connection Security

Description: The wireless network connection between blood pressure monitor and the WLAN router shall communicated through an encryption of at least a symmetric encryption of AES-128 strength.

Reason: This requirement will be met through the designing of our database schema and further secure programing technique in our server side code.

6.7.2 Data Backup

Description: The data collected by the web portion of the system will be backed up in case of an emergency or accidental deletion situation. The backup must be securely handled (i.e. encrypted and placed under an access control paradigm).

Reason: This requirement will not be test because a full database backup will not be available in our release.

6.7.3 Business Associate

Description: The system will have a HIPAA Business Associate Agreement with every vendor that has access to the data. The vendor, in the scope of the project, is the company that will host the system's web application. The web hosting company will have all the data that is being transferred and stored, so putting an agreement in place will ensure that the company will follow the HIPAA requirements for the data and the servers. This guarantees that the web hosting company will provide an infrastructure that meets HIPAA compliance requirements. The development team will then be responsible for applying secure programming methodologies to design and manage the website so that its functionality is HIPAA compliant.

Reason: This requirement will not be tested due to time and budget constraint.

6.8 Other Requirement

6.8.1 Multiple Health Modules for Website

Description: The website will be able to support many different sets of data from many different pieces or hardware, such as blood sugar readings or weight measurements, to allow for a robust web service that will allow doctors to easily track patients with a wide variety of conditions or concerns.

Reason: This requirement will not be part of the upcoming release of the project there fore cannot be tested.

6.8.3 Advanced Authentication Capabilities

Description: The monitor will have advanced authentication capabilities, to better ensure that the correct person is wearing the device at all times.

Reason: This requirement will not be part of the upcoming release of the project there fore cannot be tested.

6.8.4 Remote Control for Blood Pressure Monitor

Description: The blood pressure monitor will come with a control that will be used to control the system, allowing for more portability and removing the need for a phone or a large box to support the system.

Reason: This requirement will not be part of the upcoming release of the project there fore cannot be tested.

7. Testing Approaches

7.1 Overview

The following section will detail how the team plans to organize their efforts in order to test the system in an efficient and controlled manner. This will include tools that will be used to test the system, the process the team will go through to execute a test, and other things that will be needed in order to test the system as quickly and correctly as possible.

7.2 Strategy

Due to the compressed nature of this projects timeframe, the team will be unable to follow the conventional testing layout. The team will be testing during the development of the system. Features will be developed, tested, then integrated together in pieces, then finally assembled together and tested as a whole. The testing will focus more on the objects being created and passed through the system.

The testing will begin in hardware, which consists of three components: the BP monitor, the web server, and the database. Each will be testing in turn to determine that they are functioning correctly and will work properly within the project. This will be followed by module testing and component testing. As the modules are developed, they will each be tested to ensure they are processing their respective object and data correctly. Once the modules and components pass there testing, they will be hooked together and put through integration testing. Integration testing will be performed by testing features. A particular feature will be run; if it fails the team will know that one of the modules involved is not working properly. Once the entire system has passed integration testing it will move to acceptance testing. All stakeholders will examine the functionality of the products and verify that it meets the requirements.

Tests will be tracked on an Excel spreadsheet. The document will be divided into the different sections to be tested. When a test is created it will be placed in the document using the following format.

Test ID	Test Desription	Inputs	Expected Outputs	Tester	Date	Result	Bug ID
BLRD2	Test Request Delegator to ensure building DataPools correctly	Page requested: "../charts" UserID: "testA12"	Servlet should build ChartStorageDataPool and ChartAnalysisDataPool	Ian Jungmann	4/11/2014	Fail	CWV1

Figure 7 - 1: Test Document Excel Example

All inputs for each test will be documented. All tests will be assigned to a specific person. There will be a section to make any necessary comments related to the test itself and any further clarification needed to be made about the result of the test.. This testing log will be kept online where all team members can access it.

Whenever a defect is discovered, the team will put the affected section through regression testing. These will consist of the same tests that were used to test integration. The features that the corrected section is part of will be tested. If the feature functions correctly, it will pass regression testing.

7.3 Configurations

Many of the documented tests will need to be run multiple times to ensure that the feature being tested is working properly under different conditions. Certain modules will need to be tested with different sets of input to verify that they are responding properly, which can range from changing one or two key values (such as logging into the system) to entering a completely different set of complex data (such as the analysis system). This will be handled mostly at the Unit Testing level, to ensure that the modules are working under all circumstances. Web pages will be tested using different browsers to ensure that they are functioning in the correct manner under all of the supported web browsers. When running tests that require multiple configurations, the team will analyze the given modules algorithms and functions to generate different sets of input that should check the different results that are to be expected. The input will be documented, as well as the output, so that the team is aware of on what input the module is having problems if an error occurs.

7.4 Tools

The team is planning to use the following tools to aid in testing:

- Wireshark
- JUnit
- TestLink
- Excel

7.5 Core Functionalities

- Take a blood pressure reading
- Create an account
- Log into the system
- Advice system
- Alert system
- Chart generation

7.6 Metrics

Some of the metrics for success that our team will be using are different from the traditional percent complete layout. Our project has several features that have varying levels of complexity. For example, the advice system, which can start very simple, but has the potential to become very extensive. For these features we have decided to give a ranking of our results. A rating of fail will be given if the feature does not meet the requirements laid out in the SRS. A rating of pass will be given if the feature meets our sponsor's expectations for basic functionality. A rating of excellent will be given if the feature exceeds the expectations of our sponsor. Here are the metrics we will be using.

Table 7 - 1: Success and Failure Metrics

Metric	Success	Failure
Security features	80% or more of the documented security tests pass (most vulnerabilities have been solved)	< 80% of the documented security tests pass (there are still many vulnerabilities)
Core functionality	100% passing rating	A failure rating on any
Complexity of Algorithms	Acceptance by sponsor	Not acceptable to sponsor

8. Item Pass/Fail criteria

8.1 Overview

The following section will detail how the team will determine if individual tests pass or fail. This will allow the team to clearly dictate what is acceptable and what is not in regards to a given portion of the system, so that the success or failure of a test is clearly defined.

8.2 Hardware Tests

Table 8 - 1: Hardware Tests Pass/Fail Criteria

Hardware	Pass criteria	Fail Criteria
BPM	• Device turns on, takes a reading, and displays the results on the screen • Device connects to Bluetooth and transmits to the android device	• Device does not turn on/does not take a reading • Device cannot connect to Bluetooth
Database	• Can successfully connect and send SQL statements to the database • The correct result sets or exceptions are returned	• Unable to connect to the database • Receive corrupted data from the database
Server	• Server starts up and displays content • Can connect and transfer file to the server	• Cannot connect to the server.

8.3 Unit Tests

Table 8 - 2: Unit Tests Pass/Fail Criteria

Test ID	Pass criteria	Fail criteria
CBD1	• The correct web page is shown	• Incorrect/ no web page shown
CBD2	• Correct text is displayed	• Incorrect text is show
CBD3	• Correct web page or animation is shown	• Incorrect/ nothing is shown
CBD4	• Animation is shown	• Animation not shown or incorrect
CWV1	• Validation performs correctly	• Bad input is let through
CCG1	• Correct charts are displayed	• Incorrect/ no chart is display
CAU1	• Elements respond appropriately	• Elements do not respond appropriately
CAU2	• Text displays correctly	• Text does not display correctly
CAU3	• Correct View is displayed • Correct response message is displayed	• Incorrect response message is displayed • Data is not sent
CBV1	• Valid input is allowed through, bad input is stopped	• Valid input is flagged as invalid, or invalid input is being accepted
CDPP1	• Translation occurs successfully, and invalid packets are handled properly	• Problems translating or handling invalid packets
CBC1	• Packets are being received by the phone	• Phone does not receive packets, or packets are corrupted
TRQE1	• Packets are encrypted properly	• Packets not encrypted
TRSD1	• Packets are decrypted properly	• Packets not decrypted
TCSC1	• Data is flowing to the server and back to the client	• Server is unreachable, or data is not being sent properly.
TRQD1	• Packets are decrypted properly	• Packets not decrypted
TPA1	• Requests are being forwarded to servlets with correct permissions, and error pages are being sent with incorrect permissions	• Requests are forwarded when user does not have permissions, or error pages are being sent to legal users
TRSE1	• Packets are encrypted properly	• Packets not encrypted
BLRD1	• Request goes to the correct servlet	• Request is directed to an incorrect servlet
BLRD2	• DataPool is successfully constructed	• Errors occur while constructing DataPool
BLACL1	• The correct command objects are built based on the input	• Command objects are constructed that should not have been, or fail to be constructed when they should have been
BLACL2	• Results are being generated and stored correctly	• Errors with the storing of results
BLACM1	• Results of the command match the input and are correct	• Incorrect results
BLDP1	• Objects are all in the correct format	• Objects fail to get formatted, or do not fit the needed format
BLDP2	• The correct page or piece of data is returned	• Incorrect pages or pieces of data are being returned
BLDP3	• JSP is able to generate (thus all data is successfully loaded)	• Errors with the generating of JSP pages

BLPG1	• JSP pages are being correctly translated based on the input	• Errors translating the JSP page, or resulting web elements are incorrect
BLDC1	• Correct command objects are built	• Command objects are constructed that should not have been, or fail to be constructed when they should have been
BLDC2	• Results are being generated and stored correctly	• Errors with the storing of results
BLC1	• Data needing encryption is properly encrypted (look in database to confirm)	• Data is not correctly encrypted
BLC2	• Data needing decryption is decrypted properly (system is able to work with the data)	• Data is not decrypted properly
BLDG1	• Data is retrieved from the database	• The system fails to retrieve the data
BLDU1	• Data is stored in the database	• Database fails to store data

8.4 Component Tests

Table 8 - 3: Component Tests Pass/Fail Criteria

Test ID	Pass criteria	Fail criteria
CWU1	• UI responds appropriately	• UI responds incorrectly
CWI1	• Invalid data is rejected	• Invalid data is accepted
CWI2	• Charts are being generated	• Errors generating charts
CBU1	• UI responds appropriately	• UI responds incorrectly
CBI1	• Invalid data is rejected	• Invalid data is accepted
TCS1	• Data going to server is encrypted, data going to client is decrypted	• Errors encrypting or decrypting.
TNT1	• Data is being passed	• Data is not flowing correctly
TSS1	• Data going to client is encrypted, data going to server is decrypted	• Errors encrypting or decrypting.
BLSC1	• Servlets are calling the correct components at the correct time	• Servlets are not delegating properly, components are being called at the wrong time or without need
BLDA1	• Analysis results match the data being input	• Analysis gives incorrect results
BLRC1	• The server responds with the correct web page or data	• The wrong web page or data is sent
DSSC1	• The correct data and database messages are being received	• Incorrect data, unexpected error messages, or other issues arise
DSDS1	• Data needing encryption is being encrypted when going to the database, and is decrypted when coming from the database	• Data is being stored with no encryption, or is being pulled into the Business Logic layer still encrypted
DSDI1	• Data is being stored and retrieved successfully	• Problems interacting with the database (issues getting or storing data, errors when connecting, etc.)

8.5 Integration Tests

Table 8 - 4: Integration Tests Pass/Fail Criteria

ID	Pass Criteria	Fail Criteria
I1	• Account is successfully created when entering valid data.	• Account not created successfully (meaning errors have occurred in the process at some point)
I2	• Advice is displayed is calculated correctly	• Advice is not shown or does not match the data
I3	• Reading is successfully taken and the results are stored in the database	• Reading does not get to the database or is stored incorrectly. Bluetooth connection fails. User cannot see result on the website (for whatever reason).

8.6 Validation Tests

Table 8 - 5: Validation Tests Pass/Fail Criteria

ID	Pass Criteria	Fail Criteria
V1	• Blood pressure is successfully sent to the server, and an alert is generated if the data calls for it	• Problems connecting the phone to the BPM, problems sending the data, problems logging in, or problems generating an alert (if appropriate)
V2	• Account is successfully created, and information is edited without issue	• Account cannot be created, errors when trying to resume the setup of a patient account, or issues with trying to edit account information
V3	• Charts are displaying correctly and are based on the correct data	• Problems accessing patient data, the database retrieves the wrong data, charts do not display properly
V4	• Patient sees advice that is correct based on their blood pressure data	• Patient cannot see advice, or the patient's advice does not accurately reflect their blood pressure data and other data.
V5	• BPM precision and accuracy are within 10% standard deviation	• BPM precision and accuracy are not within the set range
V6	• Phone should connect to the BPM within 10 seconds, and the phone should connect to the server within 30 seconds	• Phone does not successfully connect within these times.

9. Test Deliverables

9.1 Overview

This section will describe what the team will deliver with the system in regards to their testing efforts. This will provide supervisors, sponsors, and customers with interest in the project with a general concept of what to expect in terms of documentation, logs, and other files that will be used to test the system and keep track of the results.

9.2 Deliverables

9.2.1 System Test Plan

This document will be provided overview of what is to be tested, how it will be approaches, and overall detail of the requirement of the test plan.

9.2.2 Test Cases Specifications

Each test case specification will include the follow items:

- Test Case ID: A unique ID assign to each test case.
- Description: A general description of what component of the system is being tested.
- Valid input range: A range of inputs that is expected to be accepted by the component.
- Valid output range: A range of outputs that is to be expected to be product by the component with regards to the “Valid input range”.
- Invalid input range: A range of inputs that is semantically not expected to not be accepted by the component.
- Process: A step by step instruction on the execution of the test.

9.2.3 Test Case Result

Each of the following test cases result will be documented with the following items after execution:

- Test Case ID: A unique ID assign to each test case.
- Date of Test: Date of which the test was executed.
- Name of Tester: Name of the person that executed the test.
- Inputs: All inputs used during the test.
- Expected Outputs: Expected Output produce with regards to the input provided to the component after execution of the test.
- Actual Outputs: Actual Output produce from the input provided to the component after execution of the test.
- Result of Test (Pass/Fail): Final evaluation of the test Pass or Fail.
- Tester Comments: Any comments the tester wish to provide for future reference or explanation for the results of the test run.
- Bug ID: A unique ID assign to each Bug detected by this test case.

9.3 Bugs and Defects

All bugs and defects that are found during testing will be documented and achieved. Each Bug or Defect will be recorded with the following:

- Bug ID: A unique ID assign to each Bug
- Test Case ID: The test case ID that generated the bug/defects.
- Name of Tester: Name of the person that executed the test.
- Time Stamp: Time of which the bug/defect was detected.
- Severity: The level of priority of the given fault/bug(High, Medium, Low).
- Status: The current status of the bug/defect (pending, resolved, removed)
- Description: A general description of the bug/defect.
- Inputs: All inputs used during the test.
- Expected Outputs: Expected Output produce with regards to the input provided to the component after execution of the test.
- Actual Outputs: Actual Output produce from the input provided to the component after execution of the test.
- Sever Log: Any error message that is being generated by the server because of bugs/defects.
- Stack Trace: Any stack trace (or error logs) output generated by testing environment/program.
- Tester Comments: Any comments the tester wish to provide for future reference or explanation.

9.4 Test Code

The code for all automated testing shall be provided and document for future regression testing. This code will include any codes written for unit, component, and integration testing. All code will be properly commented and documented and delivered with the final product.

10. Test Schedule

Table 10 - 1: Test Schedule

Task Name	Planned Start Date	Planned Due Date
Hardware Testing	Thu 3/27/14	Mon 3/31/14
Blood Pressure Monitor Testing	Tue 3/25/14	Mon 3/31/14
Database Testing	Tue 3/25/14	Mon 3/31/14
Server Testing	Tue 3/25/14	Mon 3/31/14
Account System Test	Fri 3/28/14	Fri 4/4/14
Login System Test	Fri 3/28/14	Fri 4/4/14
Android Application Test	Fri 3/28/14	Fri 4/4/14
Graph System Test	Wed 4/2/14	Sat 4/5/14
System Integration Test 1	Wed 4/2/14	Mon 4/7/14
Advice System Test	Fri 4/11/14	Fri 4/25/14
Alert System Test	Fri 4/11/14	Fri 4/18/14
System Security Test	Fri 4/11/14	Fri 4/25/14
System Integration Test 2	Fri 4/18/14	Thu 5/1/14
System Validation Test	Fri 4/25/14	Thu 5/1/14

11. Approval

Table 11 - 1: Approval Table

Name	Role	Signature	Date
Dimitrios Zikos	Project Sponsor		
Mike O'Dell	Project Supervisor		
Ian Jungmann	Team Lead		
Benjamin Bloom	Team Member		
Hao Nguyen	Team Member		
Sabreen Alabedredha	Team Member		
Scott Phan	Team Member		