ATMIYA UNIVERSITY

RAJKOT

A

Report On

Mediband

Under subject of

MAJOR PROJECT

B.TECH, Semester – VII

(Computer Engineering)

Submitted by:

1. Khushi R. Anadkat 190002003
2. Megha K. Rank 190002095

Prof. Nirali Borad

(Faculty Guide)

Prof. Tosal M. Bhalodia

(Head of the Department)

Academic Year
(2022-23)

CANDIDATE'S DECLARATION

We hereby declare that the work presented in this project entitled "**Mediband"** submitted towards completion of project in **7th Semester** of B. Tech. (Computer Engineering) is an authentic record of our original work carried out under the guidance of "**Prof. Nirali Borad".**

We have not submitted the matter embodied in this project for the award of any other degree.

Semester: 7th
Place: Rajkot

Signature:

Khushi Anadkat (190002003)
Megha Rank (190002095)

ATMIYA UNIVERSITY

RAJKOT

CERTIFICATE

Date:

This is to certify that the "**Mediband**" has been carried out by **Megha Rank** under my guidance in fulfillment of the subject Major Project in COMPUTER ENGINEERING (7th Semester) of Atmiya University, Rajkot during the academic year 2022-23.

Prof. Nirali Borad — **(Project Guide)**

Prof. Tosal M. Bhalodia — **(Head of the Department)**

ATMIYA UNIVERSITY

RAJKOT

CERTIFICATE

Date:

This is to certify that the "**Mediband**" has been carried out by **Khushi Anadkat** under my guidance in fulfillment of the subject Major Project in COMPUTER ENGINEERING (7th Semester) of Atmiya University, Rajkot during the academic year 2022-23.

Prof. Nirali Borad

(Project Guide)

Prof. Tosal M. Bhalodia

(Head of the Department)

ACKNOWLEDGEMENT

We have put in many efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

We are highly indebted to **Prof. Nirali Borad** for her guidance and constant supervision as well as for providing necessary information regarding the Major Project titled **"Mediband".** We would like to express our gratitude towards staff members of Computer Engineering Department, Atmiya University for their kind co- operation and encouragement which helped us in completion of this project.

We even thank and appreciate our colleague in helping us develop the project and people who have willingly helped us out with their abilities.

Khushi Anadkat (190002003)

Megha Rank (190002095)

ABSTRACT

The product “Mediband” as a part of this project is an exclusive approach towards facilitating an economic health band which combines all major medical features in one wristband. The objective of this project has been designed by keeping in mind the senior citizens and the fact that “old age needs so little but needs that little so much.” The ever growing swift world has often put us in a situation where the elderly people have to be left alone at home owing to our much work outside but that must not prevent us from fulfilling the responsibilities and taking well care of our elderly. Thus this Mediband is an attempt to monitor the health of the elderly even from outside the home. A single band wore by the elderly will give many health updates to the caretaker on mobile phone thus making it easy for the caretaker to look after his elderly along with managing outside work. Also the band will provide reminders to the patient through different forms. In such a way the elderly will always stay connected to the caretaker. After all “Old age needs no money or share. Just some love and care.”

INDEX

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Purpose

This project provides a very useful and helpful product the Mediband. The sole purpose of this product is to track the health of an elderly person who is away from the guardian for a while. It provides several advantages for the better of both the patient/elderly as well as the guardian. The product is specially designed to ease the job of the guardian who can monitor the health of his/her elderly from anywhere even while working outside home and the elderly who can be at home alone without fear of the need. However the mediband does not only target these senior citizens alone but can also be used by any other patients or other people.

1.2 Scope

The scope of such Medibands being used seems to be high. Who would not prefer working and taking care of the loved ones at the same time. Since this Mediband fulfills such a purpose it is highly assumable that such Medibands will be used by the majority. However with pros also come the cons. Since the product is a collaboration of hardware and software and technology can at several times fail to produce correct results, there are chances to system failures at time. Yet that does not reduce the scope of it being used. In fact these errors when identified would assist in developing a more precise and enhanced product for a better use.

1.3 Embedded Systems

- Embedded means something that is attached to another thing. An embedded system is a combination of computer hardware and software designed for a specific function. Embedded systems may also function within a larger system. The systems can be programmable or have a fixed functionality.
- It is also known as an integrated system due to its combination of hardware and software (also known as *Firmware*). It comprises of intelligent computing devices that are surrounding us. The devices may be a smartphone, smart watches, smart home devices, medical equipment, security alarms, IoT products, etc.

- The decreasing cost of processing power, combined with the decreasing cost of memory and the ability to design low-cost systems on chip, has led to the development and deployment of embedded computing systems in a wide range of application environments.

- The product Mediband is one such application of embedded system. It consist of the hardware for making of the band as well software that designed the whole communication system that takes place between the guardian and the elderly using the suitable programming language. Since C is one of the main and important programming languages used in embedded systems, it has been used a lot in designing of this product. Thus equal efforts have been put in development and analysis of both the software and the hardware part.

1.4 Technology and tools

Designing embedded software is the real task at hand which doesn't come prepackaged; you have to design it yourself according to your system requirements. These tools are required for designing different applications of embedded systems. The basic tools include:

Editor

- The writing editor leads the **first step** in the process of writing the embedded software.
- We use an editor for **writing source code.**
- The source code for the application or we can say embedded software is written in **C, C++** language according to the requirement and complexity of the embedded system you are going to make.
- Your **Notepad++** can act as your text editor meanwhile more advanced options include **Geany**, **Atom, Sublime Text and Coffee Cup** among many others.

Compiler

- After we are done with writing the source code for the application, the **second step** for designing the embedded system software is carried out with the help of a compiler, which is an important **embedded component**.
- **Compiler acts like a convertor, as it converts high-level programming language code into an object code.**
- Object code is easily digestible for the computer to process because it is written in a low-level programming language.
- You can use **Keil C51** for this purpose; it is immensely famous among its users.

Assembler

- **The assembler takes its name for converting the object code which was previously written in assembly language into machine language.**

- This conversion marks the **third step** in designing the embedded system's software.
- Now you must be thinking, why we converted the object code into machine language? The answer is simple, our computer functions on binary language, which is made up of 0 and 1 digit only. To process the source code into the language understandable by the system, we need to convert it into machine language.
- **GNU assembler, FASM assembler, NASM assemblers are** popularly used by the programmers for this purpose.

Linker

- **The linker is another tool used for designing embedded system software; it is used for linking several different object codes into a single executable program.**
- You must be thinking, why we write the object code in chunks? Why don't we write it as a whole? The answer is really simple, to make the task of writing a program easier, we write it in chunks, codes for different tasks or functions are written separately and then linked together at the end with the help of a linker to convert it into a single program.
- On the other hand, some of the object codes are already present in the library, you can pick the one you need, and use it accordingly.
- **GNU Linker** is free software available for this purpose

Libraries

We define a library as a pre-written volatile resource containing templates, codes, configuration data, documents, classes and many other related things used for the development of software or any other computer program.

- In embedded system software development libraries can be found in C, and C++, we can use them for writing our own program.
- You simply need to download the library you need, search out the code you need, copy the syntax and you are good to go!

Debugger

A bug is a flaw or error in the code of the computer program, which can cause runtime errors and syntax errors in the system making it behave abnormally.

- Now you can easily tell, what is the job of a debugger? A debugger goes through the entire written program looking for the errors.
- After testing the whole program, a debugger identifies the wrong code with the help of Breakpoints.
- You can later fix the Breakpoints and test your program again if your program for the embedded system is working well or not!

Simulator

- This is the last tool for developing embedded system software.
- What do you do, when you want to check how your written program is interacting with the hardware of your embedded system for executing the designated task? You run a simulation with the help of a simulator!
- A simulator provides information about how the written program i.e your code would perform in a real-time situation.
- You can test your system by altering the input values and analyze the outcomes.
- Results are generated in the form of written data when a simulation is run.
- Simulators prove quite helpful especially in designing real-time embedded systems.
- Software available for simulation includes **Analogic, Matlab, Simio, and Arena** among many others.

Back End: Back End technologies used in the Product are:

1. C Language:

- C is a procedural programming language. It was initially developed by Dennis Ritchie in the year 1972. It was mainly developed as a system programming language to write an operating system. The main features of the C language include low-level memory access, a simple set of keywords, and a clean style; these features make C language suitable for system programming like an operating system or compiler development.
- Many later languages have borrowed syntax/features directly or indirectly from the C language. Like syntax of Java, PHP, JavaScript, and many other languages are mainly based on the C language.
- Although not originally designed for embedded software development, the C language allows a range of programming styles from high-level application code down to direct low-level manipulation of hardware registers. As a result, C has become the most popular programming language for embedded systems today.
- C provides optimized machine instructions for the given input, which increases the performance of the embedded system. Most of the high-level languages rely on libraries, hence they require more memory which is a major challenge in embedded systems. Since C does none of that, there is little to no overhead.

2. C++:

- C++ is a general-purpose programming language and is widely used nowadays for competitive programming. It has imperative, object-oriented and generic programming features. C++ runs on lots of platforms like Windows, Linux, Unix, Mac etc.

- C++ is a cross-platform language that can be used to create high-performance applications. C++ was developed by Bjarne Stroustrup, as an extension to the C language. It gives programmers a high level of control over system resources and memory. The language was updated 4 major times in 2011, 2014, 2017, and 2020 to C++11, C++14, C++17, C++20.

- C++ is one of the world's most popular programming languages. It can be found in today's operating systems, Graphical User Interfaces, and embedded systems. C++ is an object-oriented programming language which gives a clear structure to programs and allows code to be reused, lowering development costs. It is portable and can be used to develop applications that can be adapted to multiple platforms. It is fun and easy to learn!

- C++ was developed as an extension of C, and both languages have almost the same syntax. The main difference between C and C++ is that C++ support classes and objects, while C does not.

2. PROJECT MANAGEMENT

2.1 Project Planning:

Project Planning is concerned with identifying and measuring the activities, milestonesand deliverables produced by the project. Project planning is undertaken and completed sometimes even before any development activity starts. Project planning consists of followingessential activities:

- Scheduling manpower and other resources needed to develop the system.
- Staff organization and staffing plans.
- Risk identification, analysis, and accurate planning.
- Estimating some of the basic attributes of the project like cost, duration and efforts. The effectiveness of the subsequent planning activities is based on the accuracy of these estimations.
- Miscellaneous plans like quality assurance plan, configuration management plan, etc.

Project management involves planning, monitoring and control of the people, process, and theevents that occurs as the software evolves from a preliminary concept to an operational implementation. Cost estimation is a relative activity that is concerned with the resources required to accomplish the project plan.

2.2 Project Scheduling:

The scheduling is the peak of a planning activity, a primary component of software project management. When combined with estimation methods and risk analysis, scheduling establishes a roadmap for project management. The characteristics of the project are used to adapt an appropriate task set for doing work.

2.3 Risk Management:

Risk management consists of a series of steps that help a software development team to understood and manage uncertain problems that may arise during the course of software development and can plague a software project.

Risks are the dangerous conditions or potential problems for the system which may damage the system functionalities to very high level which would not be acceptable at any cost. So in order to make our system stable and give its 100% performance we must have identify thoserisks, analyze their occurrences and effects on our system and must prevent them to occur.

2.3.1 Risk Identification:

Risk identification is a first systematic attempt to specify risks to project plan,
Scheduling resources, and project development. It may be carried out as a team process usingbrainstorming approach.

Technology risk:

Technical risks concern implementation, potential design, Interfacing, testing, andmaintenance problems

- Database Corruptness
- Garbage Collection

People Risks:

These risks are concerns with the team and its members who are taking part in developingthe system.

- Leaking an important data
- Failure of the administration
- Lack of knowledge
- Lack of clear product vision.
- Technical staff conflict
- Poor communication between people.

Tools Risks:

These are more concerned with tools used to develop the system

- Tools containing virus.

General Risks:

General Risks are the risks, which are concerned with the mentality and resources.

- Lack of resources can cause great harm to efficiency and timely productivity.
- Rapidly changing requirements.
- Changes in requirements can cause a great harm to implementation, designing and schedule of developing the system.
- Insufficient planning and task identification.

2.3.2 Risk Analysis

"Risk analysis = risk assessment + risk management + risk communication." Risk analysis is employed in its broadest sense to include:

Risk assessment:

It involves identifying sources of potential harm, assessing the likelihood that harm will occur and the consequences if harm does occur.

For this project It might be:-

- System Crash.

Risk management:

Evaluates which risks identified in the risk assessment process require management andselects and implements the plans or actions that are required to ensure that those risks are controlled.

Precautions taken to make risks minimal are as under:-

- Periodical backups are taken to avoid major loss in case of system crash.

Risk communication:

It involves an interactive dialogue between stakeholders and risk assessors and risk managers which actively informs the other processes

Steps taken for risk communication is as under:-

- Probability of certain risks is negotiated with client.
- All the possible risks are listed out during communication and project is developed taking care of that risks.

3. SYSTEM REQUIREMENTS STUDY

3.1 Hardware and Software Requirement

This lists minimum requirements to run this system efficiently.

3.1.1 Hardware Requirements for building product

Components Required	It's Function
30-Pin ESP32 Devkit	Create an accessible way for software developers to enter the world of microcontroller programming.
MPU6050 Accelerometer Gyroscope	It helps us to measure velocity, orientation, acceleration, displacement and other motion like features.
Graphic LCD 84x48 - Nokia 5110	Provide all necessary functions for display
MAX30102 Pulse Oximeter and Heart Rate Sensor	Measures blood oxygen level and the heart rate.
Pushbutton Switch - 12mm Square	They make or break the circuit
Piezo buzzer	Used to produce a tone, alarm or sound
Coloured LEDs	In our case will sense a signal and convert it into a light signal.
Lipo battery	Power other devices

Table 3.1.1.1 Hardware Requirement

3.1.2 Software Requirements for coding

For which	Software
Operating System	Windows7/8/10, Linux
Coding	Any Editor that complies, runs, debugs program
Coding Language	C/C++

Table 3.1.2.1 Software Requirements

3.1.3 Guardian's Requirements

For which	Requirement
Tracking Health	Any smart phone with required app

Table 3.1.3.1 Guardian's Requirements

3.2 Constraints

3.2.1 Hardware Limitations

The major hardware limitations faced by the system are as follows:

If the appropriate hardware is not there like the components mentioned above or they are corrupt:

- There will be problem in the proper functioning of the Mediband
- It may incorrectly measure various health related functions and thus may lead to miscommunication of information.
- The product structure already designed may be compromised

3.2.2 Reliability Requirements

The product has too many components to be assembled in a very small band which will either tend to make band heavy or they might look ugly if not placed and packed properly. Thus utmost care must be taken in assembling of the components in a single band which must not compromise with the comfort of the elderly. Also the code that runs the hardware must be tested and should run properly so that it does not lead to incorrect results.

The Reliability requirements are the validations used to protect the system against one or more incorrect activities. Without proper validation of the system, the failure possibilities of it grow higher so it is must to understand the proper validation of the system and must implement them. All the required validator controls spend very good role to keep the system secure from any unauthorized or incorrect information. In all these validation actions if system found one or more entries violating validation rules then user will be warned by proper error messages and the details or the record is not going to be saved until corrections are made to them.

4. SYSTEM ANALYSIS

4.1 Study Current System

Implementation is the stage where the theoretical design is turned into a working system. It is the most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively.

The system can be implemented only after thorough testing is done and if it is found towork according to the specification.

It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over and an evaluation of change over methods apart from planning. Two major tasks of preparing the implementation are education and training of the users and testing of the system.

The more complex is the system being implemented, the more involved will be the systems analysis and design effort required just for implementation.

The implementation phase comprises of several activities. The required hardware and software acquisition is carried out. The system may require some software to be developed. Forthis, programs are written and tested. The user then changes over to his new fully tested systemand the old system is discontinued.

4.2 Problem and weakness of similar product that already exist in market

- All necessary features not clubbed in one product.
- Less scope for the seniors or elderly person.
- Not economical with respect to the type and number of features.

4.3 Requirements of New System

4.3.1 User Requirements:

The user requirement for this system is to make the system fast, flexible, less prone toerror, reduce expenses, save the time, specially designed for elderly and provide as many functionality it can.

4.3.2 System Requirements:

Functional System Requirement:

This section gives a functional requirement that applicable to the product Mediband.

There are three sub modules in this phase.

1. Patient module.
2. Guardian module.
3. System module.

The functionality of each module is as follows:

Patient module: The user or the patient using this band must first wear the Mediband so that system can track patient health and band can assist the user for various activities.

- The Mediband will show time to the patient.
- The patient must keep wearing the band all time to stay connect with his/her guardian or caretaker.
- The system will keep tracking the patient's health and send notification and status to the guardian.
- The patient will get reminders of all his/her tasks due as set by the guardian.
- The patient has the facility of an emergency button which can be pressed in case he/she feels the urgent need of the guardian.

Guardian module: The guardian will connect the Mediband with phone to receive health updates.

- The guardian can set tasks that need to be reminded to patient in his/her absence.
- The guardian will monitor patient health on the phone in the app from far place.
- Guardian will receive notification for health and location and posture of the patient from system
- Guardian can analyze patient health based on the statistics it receives and act accordingly.
- Guardian will also receive an emergency alert from system in case patient needs an immediate help

System module: The system module will establish connection between the Guardian and the patient and keep them connect all the time.

- System will show time to patient
- System will remind patient of tasks to be completed on time
- It will diagnose patient health and record values.
- Detect location and posture of patient.
- Send notification of health, location, posture and emergency to guardian

Non-Functional System Requirements:

i. EFFICIENCY REQUIREMENT:

- When the system is used the patient and guardian can effectively take benefits of all the functionality the Mediband provides.

ii. RELIABILITY REQUIREMENT:

- The system should provide a reliable environment to both patient and guardian. Proper notifications and reminders should reach both of these.

iii. USABILITY REQUIREMENT:

- The Mediband product is designed for user friendly environment and ease of use.

iv. IMPLEMENTATION REQUIREMENT:

- Implementation of the system done with the use of C programming language that will code the system for the hardware of this embedded system and the hardware together in co-ordination with the code will work towards functioning of the product.

v. DELIVERY REQUIREMENT:

- The whole system is expected to be delivered in six months of time with a weekly evaluation by the project guide.

4.4 Feasibility Study

The feasibility study of any system is mainly intended to study and analyze the proposed system and to decide whether the system under consideration will be viable or not after implementation. That is it determines the usability of the project after deployment. To come to result a set of query is answered keeping the efficiency of the software and its impact on the domain for which it was developed.

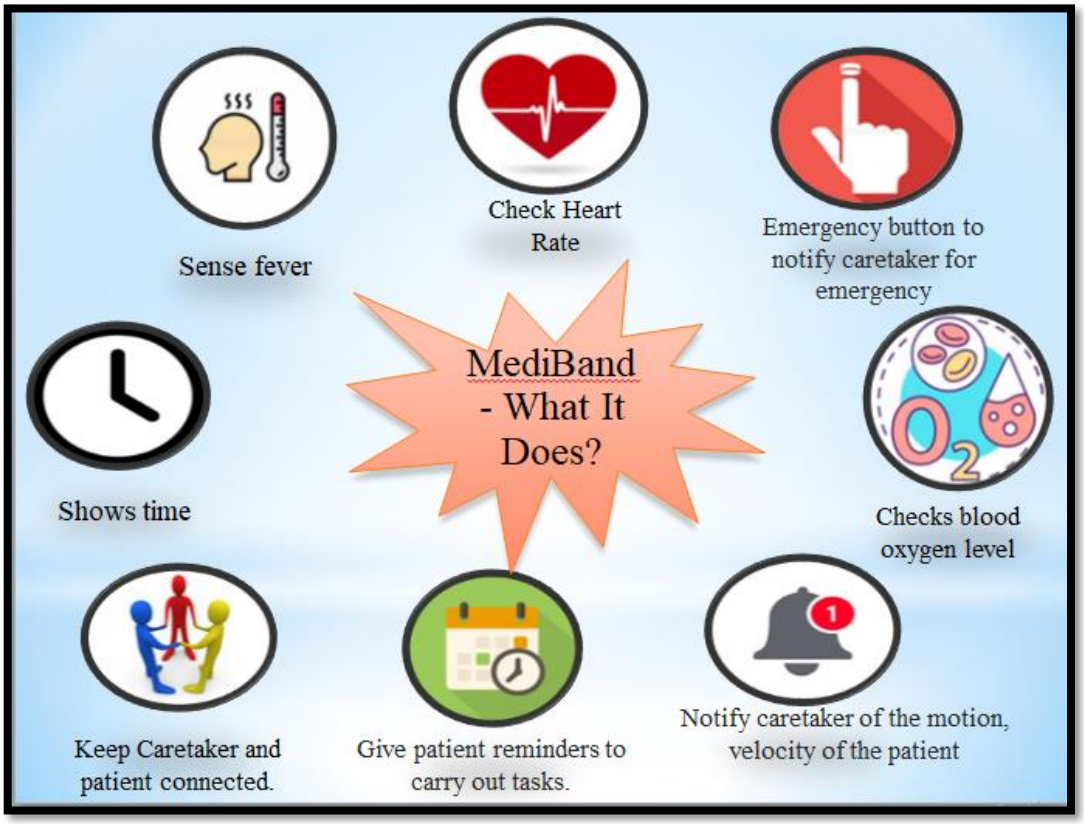
Technical Feasibility:

In technical feasibility, we study all technical issues regarding the proposed system. It is mainly concerned with the specifications of the equipment and the software, which successfully satisfies the end-user's requirement. The technical needs of the system may vary accordingly but include:

- The feasibility to produce outputs in a given time.
- Response time under certain conditions.
- Ability to process a certain volume of the transaction at a particular speed.
- Facility to communicate data.

4.5 Details of the Product - Mediband

- Features that the Mediband provides are:

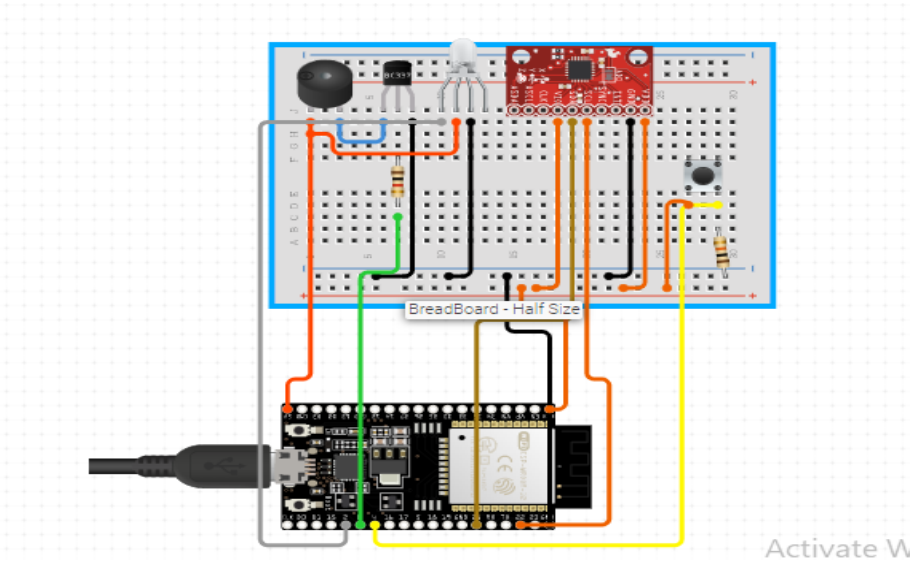


- Advantages of the Mediband are:

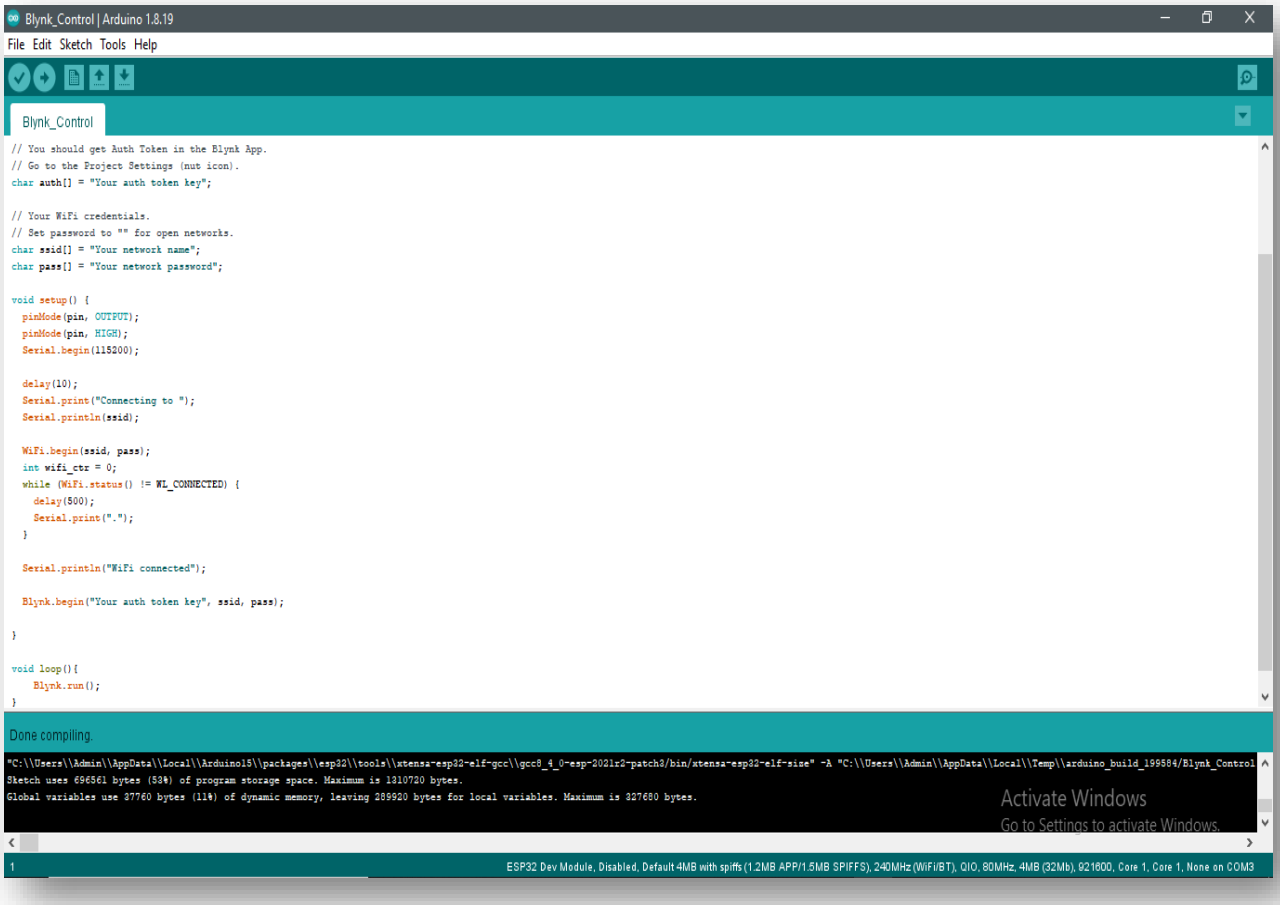
- Include several medical features in single band.
- Targeted for senior citizens but efficient to be used by any person.
- Simple and easy to use.
- Economical

- Innovation in the product is brought by putting sincere efforts and working hard on all the above advantages to set this product apart out in the market from other similar products that already exist but lack these functionalities and innovation.

- Basic circuit diagram for mediband:



- Task reminder through LED control Code:



5. System Design

5.1 Interface Design

5.1.1 Class Diagram

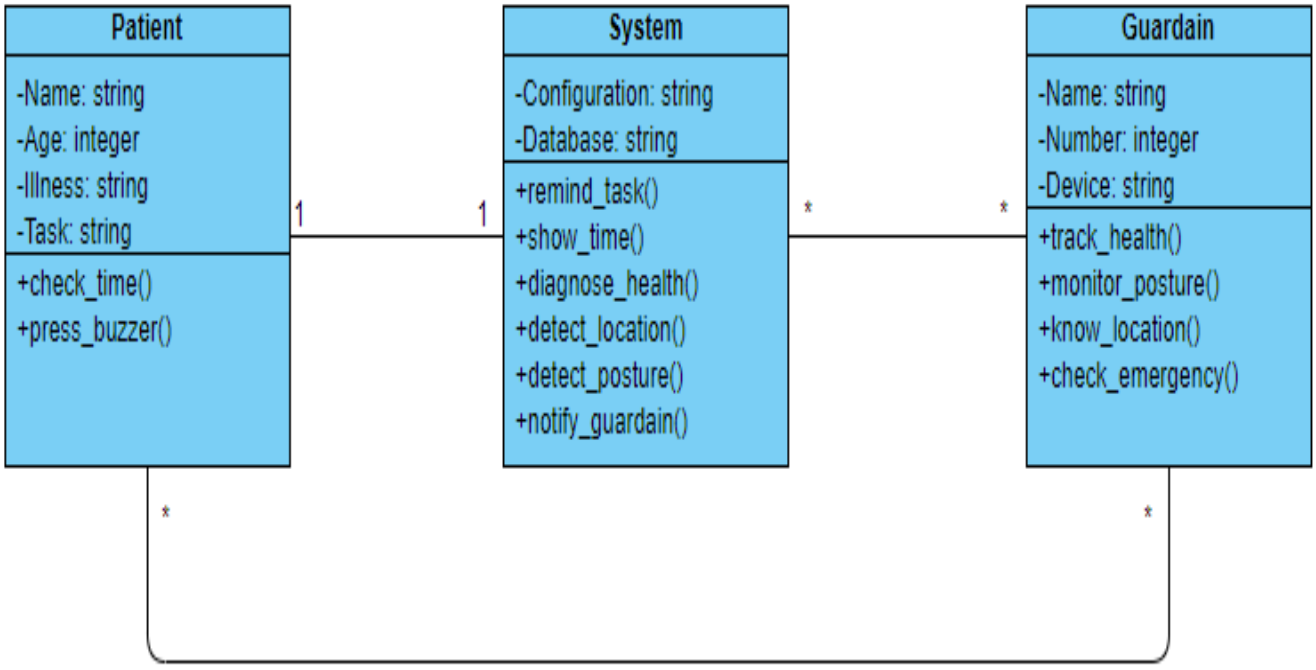


Figure 5.1.1.1 Class Diagram

5.1.2 Use Case Diagram

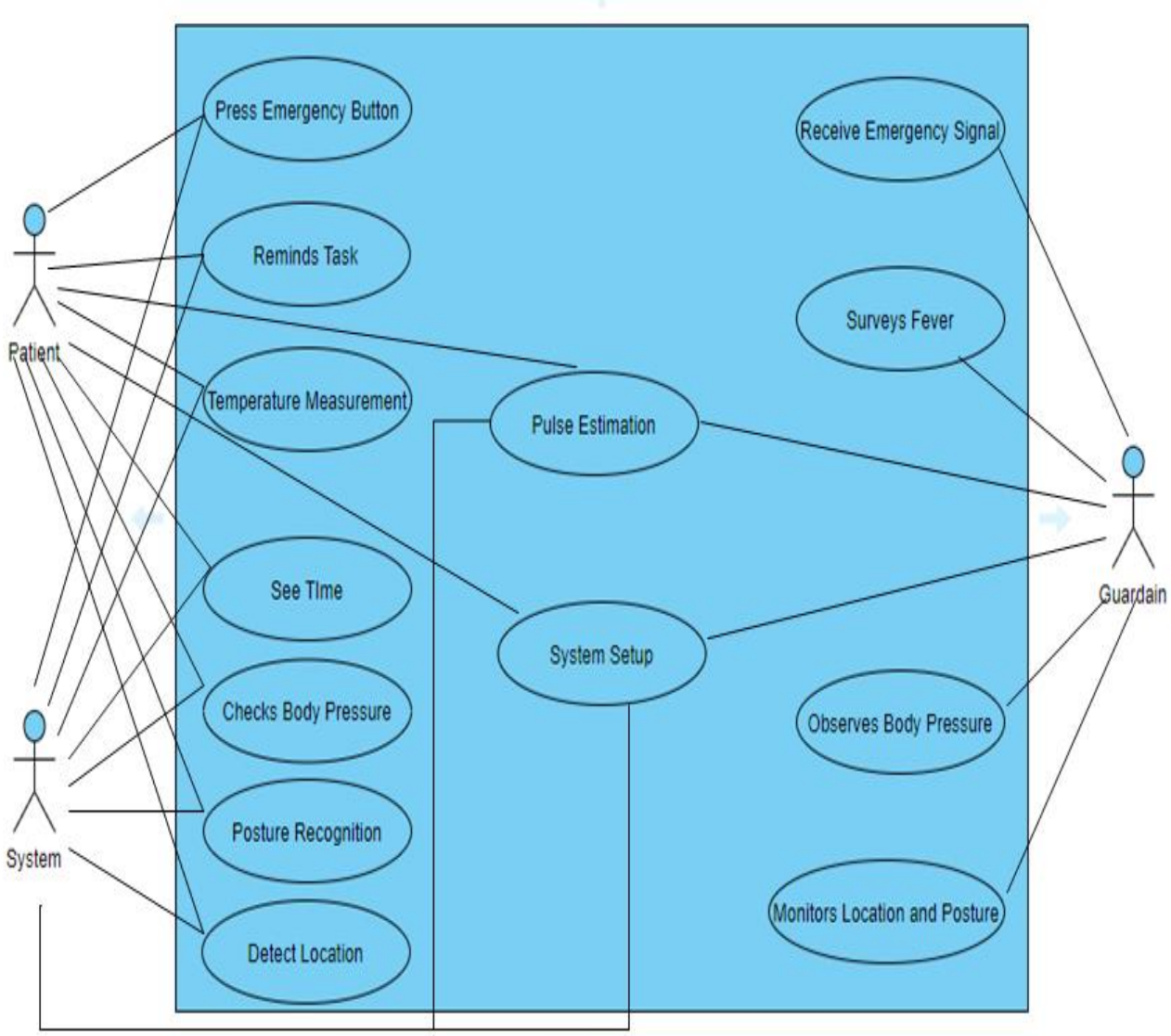


Figure 5.1.2.1 Use Case Diagram

5.1.3 Activity Diagram

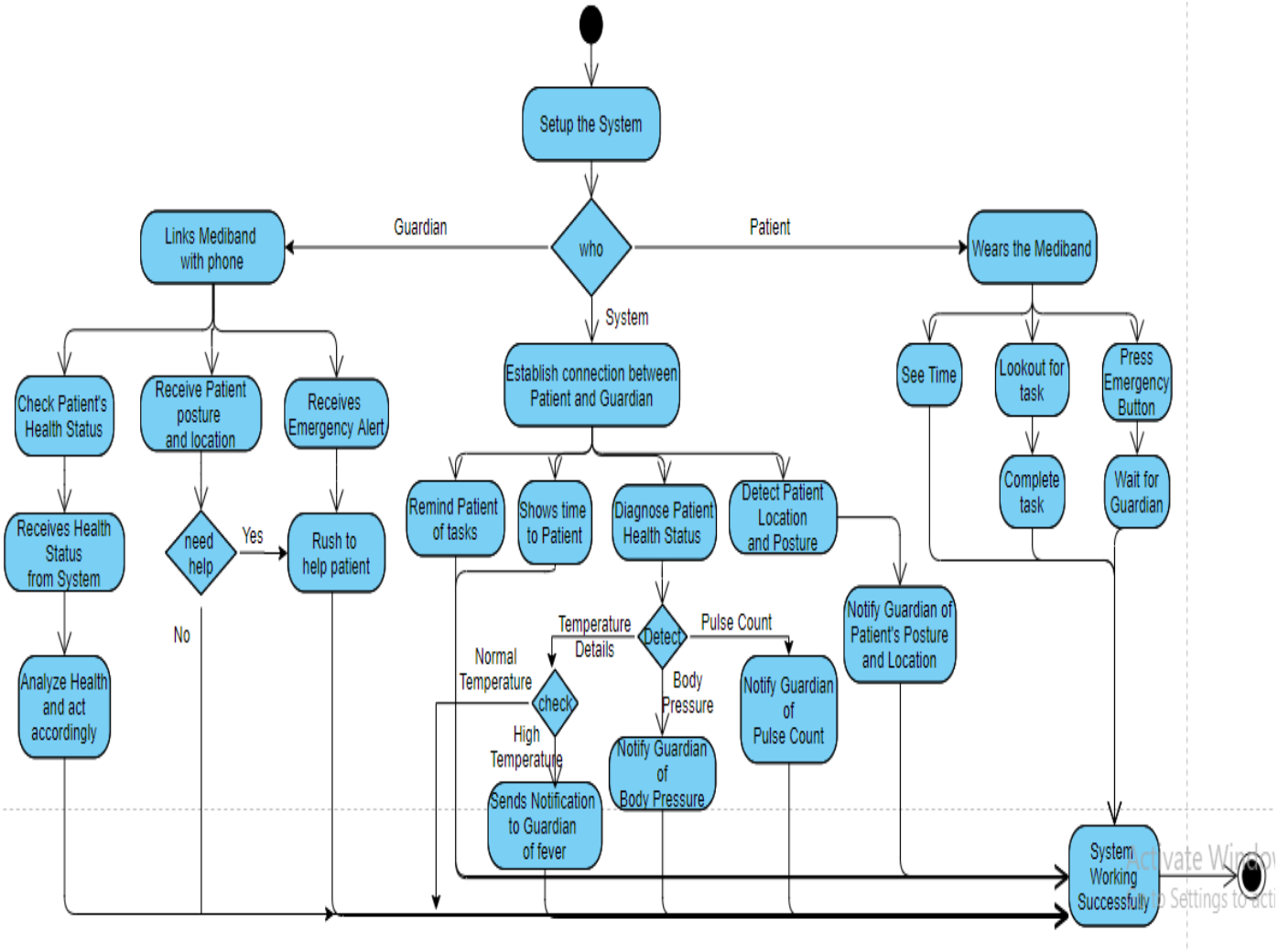


Figure 5.1.3.1 Activity Diagram

5.1.4 Sequence Diagram

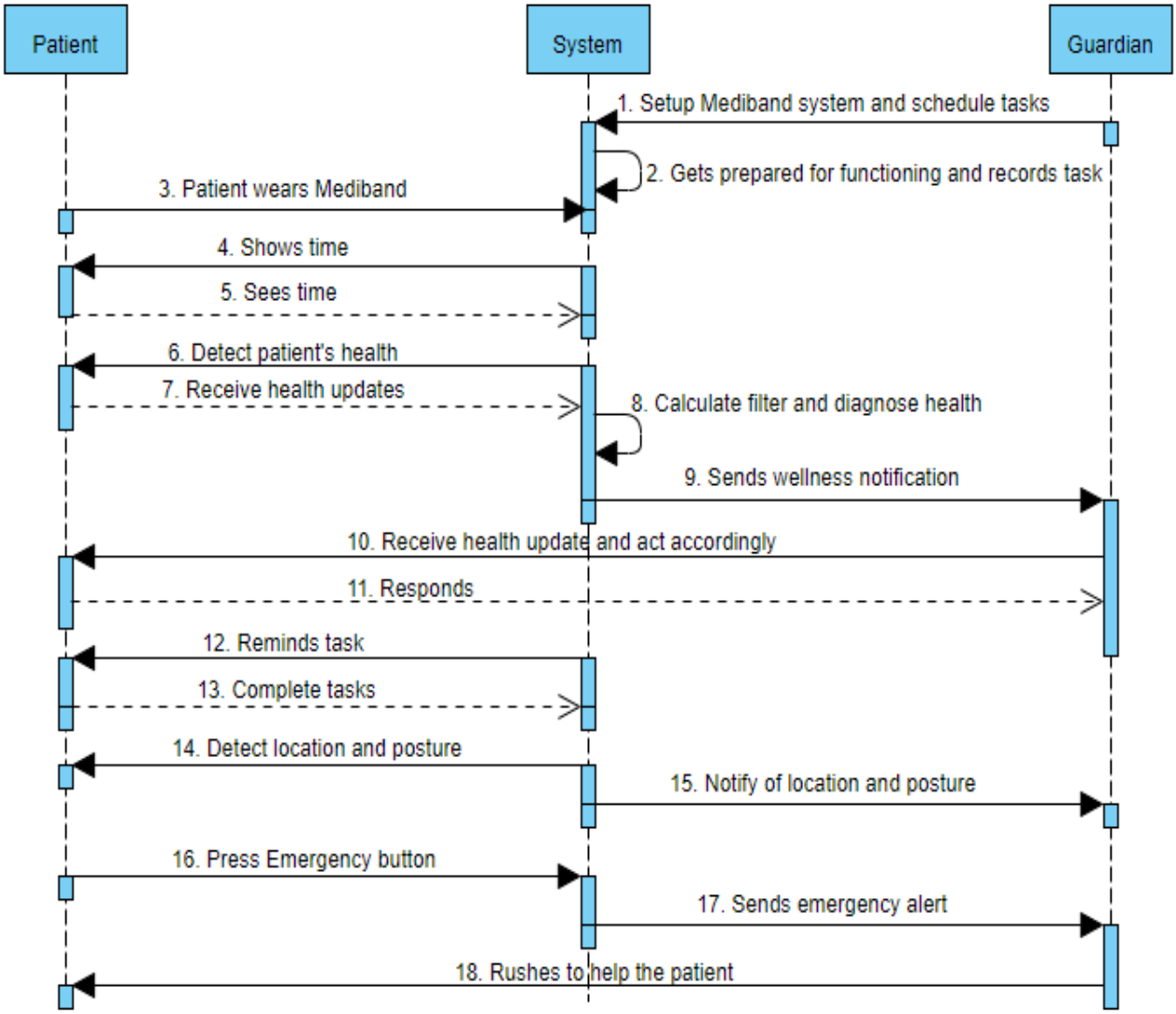


Figure 5.1.4.1 Sequence Diagram

5.1.5 Data Flow Diagram

♦ Context Level DFD

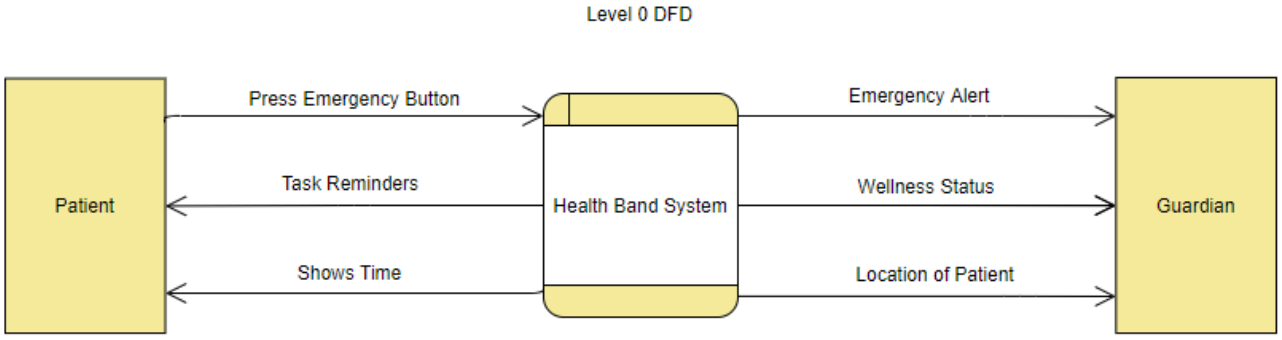


Figure 5.1.5.1 Context level DFD

♦ First Level DFD

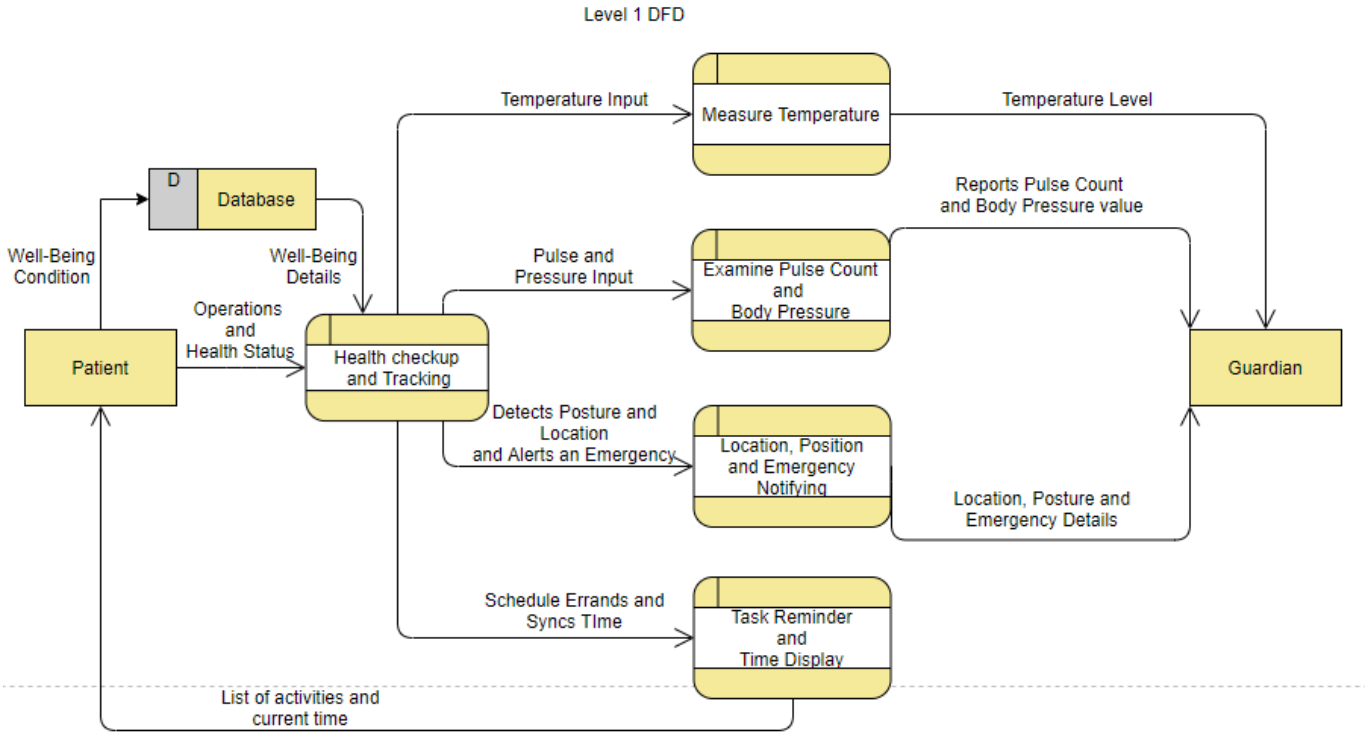


Figure 5.1.5.2 First level DFD

6. Code Implementation

6.1 Implementation Environment

Challenges identified for successful design and implementation of this project are dominated by:

Complexity, reliability/availability, transparent data access; the project was a result of a group consensus. The team was having two members. The team was guided by project manager. The team structure depends on the management style of the organization, the no. of people in the team, their skill levels and the problem difficulty.

6.2 Coding Standards

Normally, good software development organization requires their programmers to maintainsome well-defined and standard style of coding called coding standard.

6.2.1 Comment Standards:

The comment should describe what is happening, how it is being done, what parametersmean, which global are used and which are modified, and any registration or bugs.

The standards I have followed are:

- Comment may also be used in the body of the Cascading style sheets to explain individual sections or lines of codes to easily get access and easily review or manage the classes or properties for the pages.
- Inline comments should be made with the //. Comment style and should be indented atthe same level as the code described.
- For multiple line comments we write between /* ..... */.

7. Testing

7.1 Testing Strategy

A strategy for software testing integrates software test case design method into a well-planned series of steps that result in the successful construction of the software. The strategy provides the roadmap that describes the steps to be conducted as a part of testing, then these steps are planned andthen undertaken, and how much effort, time and resource will be required.

7.2 Testing Method

7.2.1 Unit Testing

The unit testing is meant for testing smallest unit of software. There are two approaches namely bottom-up and top-down. In bottom up approach the last module is tested and then moving towards the first module while top down approach reverses the action. In present work we opt for the first one.

7.2.2 Validation Testing

After the integration testing software is completely assembled as a package, interfacing error have been uncovered and corrected, and then validation testing may begin.Validation can be defined in many ways but a simple definition is what a validation succeeds when software functions in a manner that can be reasonably accepted by the user.

7.2.3 Integration Testing

The integration testing is meant to test all the modules simultaneously because it is possible that all the modules may function correctly when tested individually. But they may not work altogether andmay lead to unexpected outcome.

8. Limitations and Future Enhancement

8.1 Limitations:

Though we tried our best in developing this system but as limitations are part of anysystem so are of our system. Some limitations of Mediband are:

- It may turn out to be a little heavy band since it has been made by assembling of many components.
- Since this is the prototype it may not be as attractive as expected.
- There are several other features which could have been still integrated which this product shall have not accomplished.

8.2 Future Enhancement:

There is always a scope for enhancements in any developed system, especially when our nature of the project is iterative waterfall which allows us to rethink on the method of development to adopt changes in the project. Below mentioned are some of the changes possible in the future to increase the adaptability, and efficiency of the system.

- Making it light weight.
- Making it more attractive and appealing.
- Addition of more features without compromising the quality and effectiveness.
- Create more compact and compatible band.
- Memory audio video text integration
- Able to give instructions(Speaker)
- Keep log of activities

9. Conclusion

The product Mediband is a user friendly product mostly targeting senior citizens which is easy and convenient to operate. The product designed keeping in mind the hectic schedule of the caretakers and the need of the elderly people, seem to make a justice to the idea of this innovation as it is progressing towards a better functioning and aid. No more will caretakers have to worry about their elderly, nor will the elderly have to fear being alone at home. It gives immense pleasure to have had such a thought of designing a product that will be beneficial to the society in a unique way. Yet we believe there is always a room for improvement and thus we look forward towards polishing and enhancing the quality of this product and adding new feasible features to it in future.

10. References

WEBSITES:

https://www.codrey.com

https://internetofthingsagenda.techtarget.com

https://create.arduino.cc

https://circuitdigest.com

https://how2electronics.com

https://www.academia.edu

Books

Software Engineering by Roger Pressman