CS2004 BDS-4A FAST-NUCES Dr. Ali Afzal Malik

**Software Project Documentation**

**Version 3**

**MedWeb**

**Client: Government M. Nawaz Shareef Teaching Hospital**

**Team #02**

|  |  |  |
| --- | --- | --- |
| **Member Name** | **Member Roll #** | **Primary Responsibility** |
| Muhammed Ahmad | 21L-5617 | Team Manager(DSD , Test Cases + Back End ) |
| Sohaib Ahmad | 21L-5635 | Backend Implementation + DSD + Test Cases + Revision History |
| Ghulam Mustafa | 21L-6221 | DSD + Test Cases + Front End |
| Zenab Rizwan | 21L-5640 | IV&V |
| Ramis Baig | 21L-6209 | DSD + Test Cases + Front End + Risk Report |

Important Note: This template has been adapted from the SRS Template by Karl E. Wiegers which is available for download at ***http://www.processimpact.com/***.

# Table of Contents

**Table of Contents .......................................................................................................................... ii**

**Revision History ........................................................................................................................... iii** **Glossary ........................................................................................................................................ iv** **1.** **Introduction ..............................................................................................................................1** 1.1 Product ......................................................................................................................................... 1

1.2 Scope ............................................................................................................................................ 1

1.3 Business Goals ............................................................................................................................. 1 1.4 Document Conventions ................................................................................................................ 1

1.5 References .................................................................................................................................... 1

1. **Overall Description ..................................................................................................................3** 2.1 Product Features .......................................................................................................................... 3 2.2 User Classes and Characteristics ................................................................................................. 3

2.3 Operating Environment ................................................................................................................ 3 2.4 Design and Implementation Constraints ...................................................................................... 3

2.5 Assumptions and Dependencies .................................................................................................. 4

1. **Functional Requirements ........................................................................................................6** 3.1 Use-Case 1 ................................................................................................................................... 6
   1. Use-Case 2 (and so on) ............................................................... **Error! Bookmark not defined.**
2. **Nonfunctional Requirements ................................................................................................14** 4.1 Performance Requirements ........................................................................................................ 14
   1. Security Requirements ............................................................................................................... 14
3. **Other Requirements ..............................................................................................................15** **6.** **Analysis Models ......................................................................................................................16** 6.1 Use Case Diagram (UCD) ......................................................................................................... 16
   1. Analysis Class Diagram (ACD) ................................................................................................. 17
   2. Data Flow Diagram (DFD) ........................................................................................................ 17
4. **Design Models.........................................................................................................................18** 7.1 Architectural Design .................................................................................................................. 18
   1. Design Class Diagram (DCD) ................................................................................................... 18
   2. Design Sequence Diagrams (DSDs) .......................................................................................... 18
5. **GUI Design .............................................................................................................................19** **9.** **Test Cases ...............................................................................................................................20** **10.** **IV & V Report ........................................................................................................................21** **11.** **Risk Report .............................................................................................................................22** **12.** **Activity Timesheet .................................................................................................................23**

**13.** **Updated Project Plan .............................................................................................................24**

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name(s)** | **Date** | **Reason(s) For Change(s)** | **Version** |
| Muhammad Ahmad | 20/02/2023 | Less use of shall/must statement | 0.1 |
| Ramis Baig | 22/02/2023 | Linus OS must not be included in the Software requirement | 0.2 |
| Saad Imran | 23/02/2023 | Miscalculation of risk exposure | 0.3 |
| Muhammad Ahmad | 28/02/2023 | The users are the actors instead of the patient | 0.4 |
| Ramis Baig | 22/02/2023 | Correction in Analysis Class Diagram (ACD) | 1.1 |
| Saad Imran | 23/02/2023 | Correction in Data Flow Diagram (DFD | 1.2 |
| Sohaib Ahmad | 28/02/2023 | Changing in Gui color theme | 1.3 |
| Zenab Rizwan | 24/03/2023 | Recalculation in the risk exposure assessment | 1.4 |
| Ghulam Mustafa | 26/03/2023 | Point out security vulnerabilities in the n login process  that needed to be addressed | 1.5 |
| Sohaib Ahmad | 28/03/2023 | Added a new feature to allow users to skip  login. | 1.6 |
| Muhammad Ahmed | 01/04/2023 | Correction of Phase 1 | 2.0 |
| Sohaib Ahmed | 25/04/2023 | Refined use cases and activity diagrams to improve system modeling | 2.1 |
| Muhammad Ahmed | 5/1/2023 | Correction of Phase 2 | 3.0 |

# Glossary

**HTML (Hyper Text Markup Language):** HTML is a standard language used for web designing.

**CSS (Cascading Style Sheets):** CSS is used to control the visual representation of web pages.

**Java Script:** JavaScript is a programming language used to create interactive and dynamic web pages. It is often used in combination with HTML and CSS.

**PHP (Hypertext Processor**: PHP is also used to create dynamic web pages but the difference between PHP and JavaScript is that the code on PHP s first executed on the server and then the web page is sent to the user’s browser.

**Apache:** It is an open-source web server used to serve websites and web applications.

**Bootstrap:** It is a front-end web development framework.

**IIS (Internet Information Services) :** It is a software developed by Microsoft that is used to host and manage websites.

**API (Application Programming Interface) :** It is used for building software applications that specify how different components should interact with each other.

**HTTPS (Hyper Text Transfer Protocol Secure) :** It is a secure version of HTTP protocol used for transferring data over the internet.

**Edge Browser:** Microsoft developed Edge Browser. It was designed to be fast, secure and compatible with web standards.

**UML (Unified Modeling Language) :** It is a graphical language used for modeling software systems.It is used to display different aspects of the software including its structure, behavior and interactions**.**

**SQL Server:** It is a relational database management system used to store, manage and retrieve data as requested by software applications.

# 1. Introduction

## 1.1 Product

Our project is of a local-scale database it is a software application designed for medicine-related websites. The software has two main features - a medicine recommender system that allows users to search for medicines based on their symptoms, and a blog management system that allows medical experts to add, update and delete blog articles related to medicine. The software is likely to be useful for medical professionals, healthcare providers, and people who are looking for information on medicines and their uses.

## 1.2 Scope

The medicine website will serve as a platform for users to search for recommendations for medicine based on their disease or symptom. Users will be able to enter a keyword or phrase related to their condition, and the website will provide recommendations for medicine that can help treat that condition. The website will also provide information about each medicine, including dosage, side effects, and any other relevant information.

Additionally, the website will have a feature for medical experts or managers to create and manage blog posts related to medicine and health. They will be able to log in to a dashboard where they can create new posts, edit existing ones, and delete old ones. These blog posts can be informative, educational, or promotional in nature and can cover a wide range of topics related to medicine and healthcare. Overall, the scope of this medicine website is to provide users with a platform to find recommendations for medicine and access to educational and informative blog posts related to medicine and healthcare.

## 1.3 Business Goals

Our medicine website has several business goals that benefit our users. We aim to provide accurate information to help users make informed healthcare decisions, create a community of healthcare professionals and enthusiasts, offer personalized medicine recommendations, provide a userfriendly interface, and offer a platform for feedback and engagement. By achieving these goals, we hope to establish ourselves as a trusted source of information and a valuable resource for healthcare professionals and enthusiasts. Other users of our website will be able to benefit from possible. The website is not just limited to one organization; rather it will have features that are in demand for many pharmaceutical websites

.

## 1.4 Document Conventions

Bold characters exhibit significant information.

## 1.5 References

Pharmaceuticals, C. (2018). *CCL Pharmaceuticals - Healthy, Happy Life*. Retrieved from CCL Pharmaceuticals:

http://www.cclpharma.com/

Pharmacy, S. (2018). *Selmore Pharmacy*. Retrieved from Selmorepharma.com:

https://www.selmorepharma.com/

Tech, A. (2014). *Non-Functional Requirements*. Retrieved from Aakash Tech Support Docs:

https://aakashtechsupportdocs.readthedocs.io/en/latest/nonfunc.html

# 2. Overall Description

## 2.1 Product Features

**Home Page**: Displays an overview of the website, including links to other pages such as About Us, Blog, and Contact Us.

**About Us Page:** Provides information about the website and its mission.

**Blog Page**: Allows users to search for medicine blogs and filter results based on age range, and view blog posts.

**Contact Us Page:** Allows users to submit messages to the website's contact email address.

**Blog Manager Page**: Allows the manager to add, edit, and delete blog posts.

**Manager Login/Logout:** Allows the manager to log in/log out to their account.

Overall, the website provides a platform for users to access information on medicine-related topics through blog posts, and for the manager to manage those blog posts through the Blog Manager page. Users can also contact the website via the Contact Us page, while the manager has access to additional features such as adding, editing, and deleting blog posts through the Manager Login.

## 2.2 User Classes and Characteristics

1. Patients/Doctors who will search up medicines.
2. Managers who would be able to manage the blog.

## 2.3 Operating Environment

The medicine website is a web-based application developed using a combination of front-end and back-end technologies. It requires the following environment to operate:

**Hardware Platform**: The website is designed to operate on standard desktop and laptop computers, as well as mobile devices such as smartphones and tablets.

**Operating System:** The website is designed to be platform-independent and can be accessed from any operating system that supports web browsing, including Windows.

**Web Server:** The website is built using HTML, CSS, JavaScript, and PHP, and requires a web server that supports these technologies. Commonly used web servers include Apache, Nginx, and IIS.

**Database Management System:** The website stores data in a relational database management system, such as MySQL or SQL Server, and requires a database server to operate.

**Web Browser:** The website is designed to be accessed through web browsers such as Google Chrome and Microsoft Edge.

**Additional Software**: The website may use additional software libraries, frameworks, or APIs to enhance its functionality, such as jQuery, Bootstrap, or Google Maps API.

Overall, the medicine website requires a standard web development environment, including a web server, database management system, and web browser. It is designed to operate on a variety of hardware and operating systems and does not require any specialized software or components.

## 2.4 Design and Implementation Constraints

**Regulatory Policies**: The medicine website must comply with regulatory policies related to the healthcare industry, such as HIPAA (Health Insurance Portability and Accountability Act), which governs the privacy and security of patient information. The website must ensure that user data is protected and secure, and that any communication or transactions are encrypted.

**Technologies and Databases**: The medicine website is developed using HTML, CSS, JavaScript, PHP, and a relational database management system such as MySQL or SQL Server. The development team must have expertise in these technologies and databases to ensure the website is built effectively and efficiently.

**Design Conventions and Programming Standards**: The website must adhere to design conventions and programming standards set by the customer or the development team, to ensure consistency and maintainability of the codebase. The development team must follow coding best practices such as modularization, commenting, and error handling.

**Hardware and Memory Limitations:** The website must be designed to operate efficiently within the available hardware and memory limitations. The development team must optimize the website's performance to ensure it is responsive and quick to load, even on slower connections.

**Security Considerations:** The website must be developed with security in mind, to prevent any unauthorized access or data breaches. The development team must implement security measures such as encryption, access control, and firewalls to ensure that user data is protected.

**Communication Protocols:** The website must communicate securely with other applications or APIs, using protocols such as HTTPS or SSL to ensure that data is encrypted during transmission.

## 2.5 Assumptions and Dependencies

**Assumptions:**

**User Age Filter**: The website assumes that users will enter their age to filter the blog content based on their age range. The filtering may not work effectively if users do not provide their age.

**Website Traffic**: The website assumes that there will be a certain amount of traffic at any given time, and is designed to handle this expected amount of traffic. However, if the website experiences a sudden spike in traffic, it may slow down or crash.

**User Authentication:** The website assumes that only managers will have access to add, delete and update blogs. It assumes that user authentication is reliable and secure, and only authorized personnel will be able to access the website's management functionalities.

**Dependencies:**

**Development Environment:** The project is dependent on the availability and functionality of the development environment, including hardware, software, and other tools needed to develop and test the website. Any issues with the development environment may delay or affect the project's progress.

**Operating Environment:** The project is dependent on the operating environment in which the website will operate. The website is designed to operate on Chrome and Edge browsers, and on the Windows operating system. Any changes or issues with the operating environment may affect the website's performance.

**User Input:** The website is dependent on accurate and consistent user input, including search terms and filter parameters. Any errors or inconsistencies in user input may affect the website's search and filtering functionalities.

# 3. Functional Requirements

## 3.1 Use-Case 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-1(Search Bar) | |
| **Purpose** | | To get easy access to our contents | |
| **Priority** | | High | |
| **Actors** | | Users, Admins | |
| **Pre-conditions** | | Visiting the homepage | |
| **Post-conditions** | | The website displays a list reverent medicine blogs | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click On Search Bar | | Will display a blinking cursor |
| **2** | Typing | | Shows the related contents |
| **3** | Pressing Enter | | Shows the entered content |
| **4** | Clicking On the Recommended Content | | Takes you to that specific content |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click On Search Bar | | Taking time for processing due to the slow speed of internet |
| **2** | Typing | | Shows a message of unavailability of the content |
| **3** | Pressing Enter | | Taking Time for processing due the slow speed of internet |

**Table 1: UC-1**

### 3.2 Use-Case 2

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-2(Login) | |
| **Purpose** | | To get easy access to our website | |
| **Priority** | | High | |
| **Actors** | | Users, Admins | |
| **Pre-conditions** | | Searching our website on the internet | |
| **Post-conditions** | | Displays our main page | |
|  | | **Typical Course of Action** | |
| **S#** | **A** | **ctor Action** | **System Response** |
| 1 | Click On Login | | Shows two boxes for e-mail and password entry. |
| **2** | Enter password | | Verifies the password and allows you to access the website |
|  | **Alternate Course of Action** | | |
| **S#** | **Actor Action** | | **System Response** |
| 1 | Click on Login | | Takes time to load due to the slow speed of internet. |
| **2** | Enter password | | If the password is wrong, it again asks for the password. |

**Table 2: UC-2**

### 3.3 Use-Case 3

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-3(View Home Page) | |
| **Purpose** | | User View the home page | |
| **Priority** | | Low | |
| **Actors** | | Users | |
| **Pre-conditions** | | The user navigates the website or checking the host | |
| **Post-conditions** | | The website displays the homepage with links to other pages | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click On “Home Page” | | Takes you to the main page |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click On “Home Page” | | Shows waiting message due to the slow speed of internet |

**Table 3: UC-3**

### 3.4 Use-Case 4

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-4(Submit Contact us form) | |
| **Purpose** | | To contact with us | |
| **Priority** | | Low | |
| **Actors** | | User | |
| **Pre-conditions** | | The fills up the contact us form and clicks the submit button | |
| **Post-conditions** | | The website sends an email to the website’s contact email address with the user’s message | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click ”Contact US” | | Takes to the page where you can submit your feedback |
| **2** | Enter name | | Save your name in database |
| **3** | Enter phone number | | Save your phone number in database |
| **4** | Enter email | | Save your email ID in database |
| **5** | Select medicine | | Select the clicked medicine |
| **6** | Enter your message | | Save your message |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click “Contact US” | | Shows waiting message due to the slow speed of internet |
| **2** | Enter “Name” | | Show error message due to some storage problem |
| **3** | Enter “E-mail” | | Show error message due to some storage problem |
| **4** | Select “Medicine” | | There is no such medicine available in database |
| **5** | Enter your message | | Show error message due to some storage problem |

**Table 5: UC-4**

### 3.5 Use-Case 5

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-5(View About Us) | |
| **Purpose** | | To read the information About us page | |
| **Priority** | | Low | |
| **Actors** | | Users | |
| **Pre-conditions** | | The user clicks on the About Us link in the navigation bar | |
| **Post-conditions** | | The website displays the About Us page that includes various charts | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click On “About Us” | | Takes a view to the About US page |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click On “About Us” | | Shows waiting message due to the slow speed of internet |
| **2** | Click “Read More” | | There is no as much knowledge available about the page |

**Table 5: UC-5**

### 3.6 Use-Case 6

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-6(Filter Medicine Blog by Age) | |
| **Purpose** | | To show the relevant medicine according to age | |
| **Priority** | | High | |
| **Actors** | | Users | |
| **Pre-conditions** | | The user selects the age from the filter drop down and clicks the filter button | |
| **Post-conditions** | | The website displays a list of medicines blogs relevant to the selected age range | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click On “Filter” | | Open the filter page |
| **2** | Click checkbox of age | | Press Sumit/ Confirm button |
| **Alternate Course of Action** | | | |

|  |  |  |
| --- | --- | --- |
| **S#** | **Actor Action** | **System Response** |
| **1** | Click On “Filter” | Shows waiting message due to the slow speed of internet |
| **2** | Enter age | Error message by entering the faulty age or duplicate age |

**Table 6: UC-6**

### 3.7 Use-Case 7

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-7(View Blogs Post) | |
| **Purpose** | | To show the relevant Blogs | |
| **Priority** | | Low | |
| **Actors** | | Users | |
| **Pre-conditions** | | The user clicks on a blog post from the list searches result. | |
| **Post-conditions** | | The website displays the selected blogs post | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click the name of medicine | | Show the blog related to the clicked medicine |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click the name of medicine | | Shows waiting message due to the slow speed of internet |
| **2** | Click the name of medicine | | There is no blog related to that medicine |

**Table 7: UC-7**

### 3.8 Use-Case 8

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-8(Add Blogs Post) | |
| **Purpose** | | Add a new blog | |
| **Priority** | | Low | |
| **Actors** | | Manager | |
| **Pre-conditions** | | The manager logs in to the website and navigates to the blog manager page | |
| **Post-conditions** | | The website displays the blogs manager page with a form to add a new blog post | |
|  | | **Typical Course of Action** | |
| **S#** | **A** | **ctor Action** | **System Response** |
| **1** | Click save |  | Blog is successfully added in database |
|  | | **Alternate Course of Action** | |
| **S#** | **A** | **ctor Action** | **System Response** |
| **1** | Click save |  | Error in storage as there is no enough space |

**Table 8: UC-8**

### 3.9 Use-Case 9

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-9(Edit Blog Page) | |
| **Purpose** | | Update the existing blog | |
| **Priority** | | Low | |
| **Actors** | | Manager | |
| **Pre-conditions** | | The manager logs in to the website and navigates to the blog manager page | |
| **Post-conditions** | | The website displays the blogs manager page with a list of existing blog post and options to edit each one | |
|  | | **Typical Course of Action** | |
| **S#** | **A** | **ctor Action** | **System Response** |
| **1** | Click save |  | Blog is successfully Updated in database |
|  | | **Alternate Course of Action** | |
| **S#** | **A** | **ctor Action** | **System Response** |
| **1** | Click save |  | Storage problem /limited same |

**Table 9: UC-9**

### 3.10 Use-Case 10

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-10(Delete Blogs Post) | |
| **Purpose** | | Delete the existing blog | |
| **Priority** | | Low | |
| **Actors** | | Manager | |
| **Pre-conditions** | | The manager logs in to the website and navigates to the blog manager page | |
| **Post-conditions** | | The website displays the blogs manager page with a form to delete each one | |
|  | | **Typical Course of Action** | |
| **S#** | **A** | **ctor Action** | **System Response** |
| **1** | Click save |  | Successfully deleted the blog |
|  | | **Alternate Course of Action** | |
| **S#** | **A** | **ctor Action** | **System Response** |
| **1** | Click save |  | Shows waiting message due to the slow speed of internet |

**Table 10: UC-10**

### 3.11 Use-Case 11

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-11(View contact Us) | |
| **Purpose** | | To view the all messages that are received | |
| **Priority** | | Low | |
| **Actors** | | Manager | |
| **Pre-conditions** | | The manager logs in to the website and navigates to the blog manager page | |
| **Post-conditions** | | Click the contact us link | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click the contact us link | | Show all message that are received |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click the contact us link | | Shows waiting message due to the slow speed of internet |

**Table 11: UC-11**

### 3.12 Use-Case 12

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-12(Logout) | |
| **Purpose** | | To keep the Account secure | |
| **Priority** | | High | |
| **Actors** | | User, Manager | |
| **Pre-conditions** | | The User/Manager clicks on the logout button in the navigation bar | |
| **Post-conditions** | | The website logs out the user/manager and redirects the login-page | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click logout | | Successfully logout from the website and open the home page |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click logout | | Shows waiting message due to the slow speed of internet |

**Table 12: UC-12**

# 4. Nonfunctional Requirements

## 4.1 Performance Requirements

To constrain the speed of operation of a website, consider the following types of performance requirements:

Response requirements: This refers to the time it takes for the website to respond to user input or requests. For example, the website will respond quickly when a user searches for a particular illness or medication. To constrain response times, we set a requirement such as "The website must -respond to user input within 2 seconds."

Throughput requirements: This refers to the number of requests the website will handle at any given time. To constrain throughput, we set a requirement such as "The website be able to handle at least

1000 requests per minute."

Timing requirements: This refers to the time it takes for the website to perform specific tasks or operations. For example, the website will be able to display search results within a certain amount of time. To constrain timing, we will set a requirement such as "The website must display search results within 5 seconds of the user entering a search query."

## 4.2 Security Requirements

* Security requirements are included in a system to ensure:
  + Unauthorized access to the system and its data is not allowed
  + Ensure the integrity of the system from accidental or malicious damage Examples of security requirements are:
  + The system’s data administrator may only change the access permissions for system data.
  + All system data must be backed up every 24 hours and the backup copies stored in a secure location that is not in the same building as the system
  + All external communications between the system’s data server and clients must be encrypted.
* The system shall not permit operation unless the operator guard is in place.
* The system shall not allow the sedative dose delivered to the patient to be greater than the maximum value, which is determined by the patient’s physician.

# 5. Other Requirements

**5.1 DATA BASE REQUIREMNETS**

Our medicine website will use a relational database management system, either SQL Server or PostgreSQL, to store and manage data related to medicine recommendations, user profiles, blog articles, comments, and other relevant information. The database management system will be scalable, secure, reliable, and provide backup and recovery capabilities. It will also have performance tuning and easy administration features.

**5.2 HARDWARE REQUIREMENTS**

**Hardware Requirements:**

Our medicine website will require a minimum and recommended hardware specification for optimal performance. The following specifications are recommended:

**Minimum Hardware Requirements:**

Processor: 1.8 GHz dual-core or higher

Memory: 4 GB RAM or higher

Storage: 100 GB hard disk or higher

Network: 100 Mbps or higher

Recommended Hardware Requirements:

Processor: 2.4 GHz quad-core or higher

Memory: 8 GB RAM or higher

Storage: 500 GB hard disk or higher

Network: 1 Gbps or higher

It is important to note that the actual hardware requirements may vary depending on the specific needs of our medicine website, such as the number of users, the amount of data being processed, and the level of website traffic. Therefore, we will continually monitor the performance of the website and adjust the hardware requirements as needed to ensure optimal performance and user experience. Our team will conduct regular assessments to determine the current and future needs of the website and make any necessary adjustments to the hardware specifications.

### 5.3 SOFTWARE REQUIREMENTS

Development tools consist of HTML CSS JavaScript, Node JS and database will be SQL Server or Postgre SQL. Browser will be the Chrome or Edge.

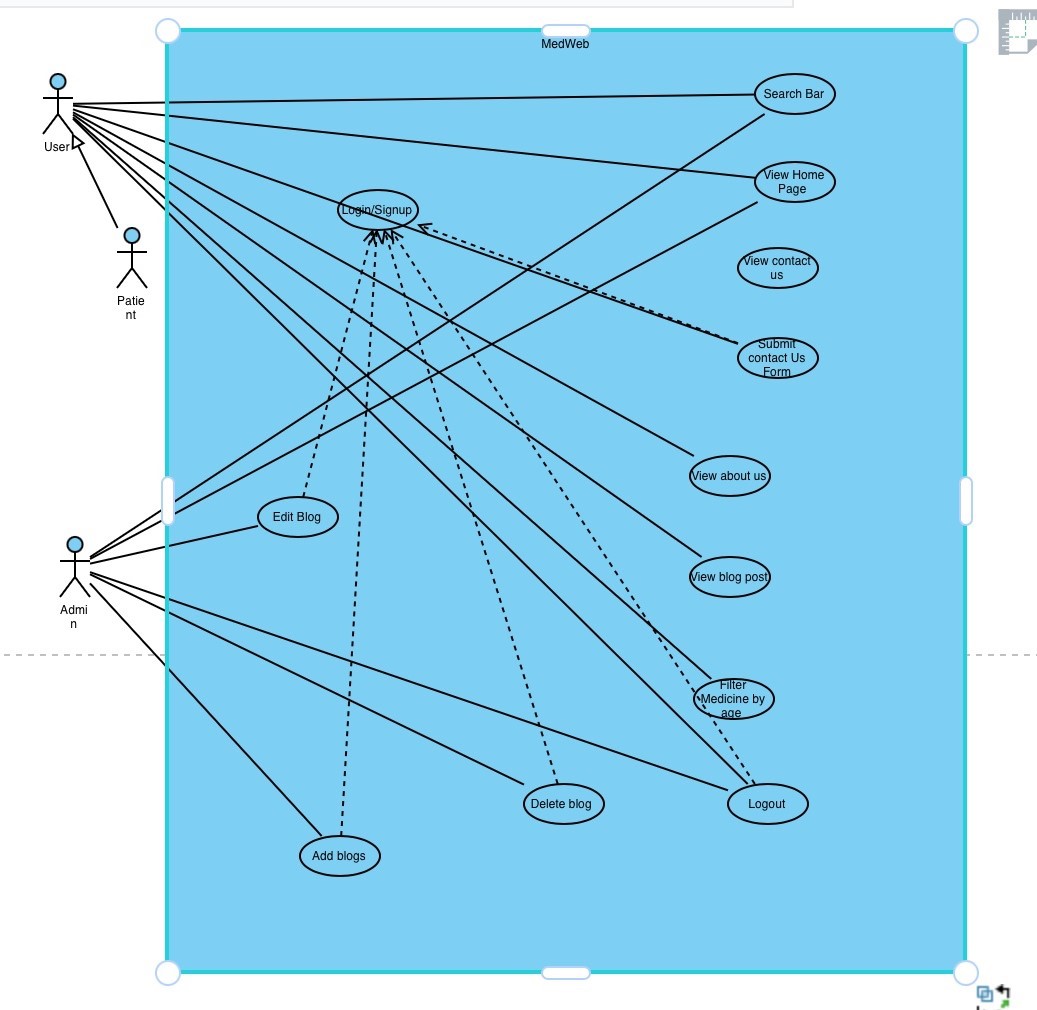
### 5.4 COMMUNICATION INTERFACE REQUIREMENTS

The website should have a responsive design that works on a variety of devices and screen sizes. This will require implementing a flexible layout that adjusts to different screen widths and resolutions.

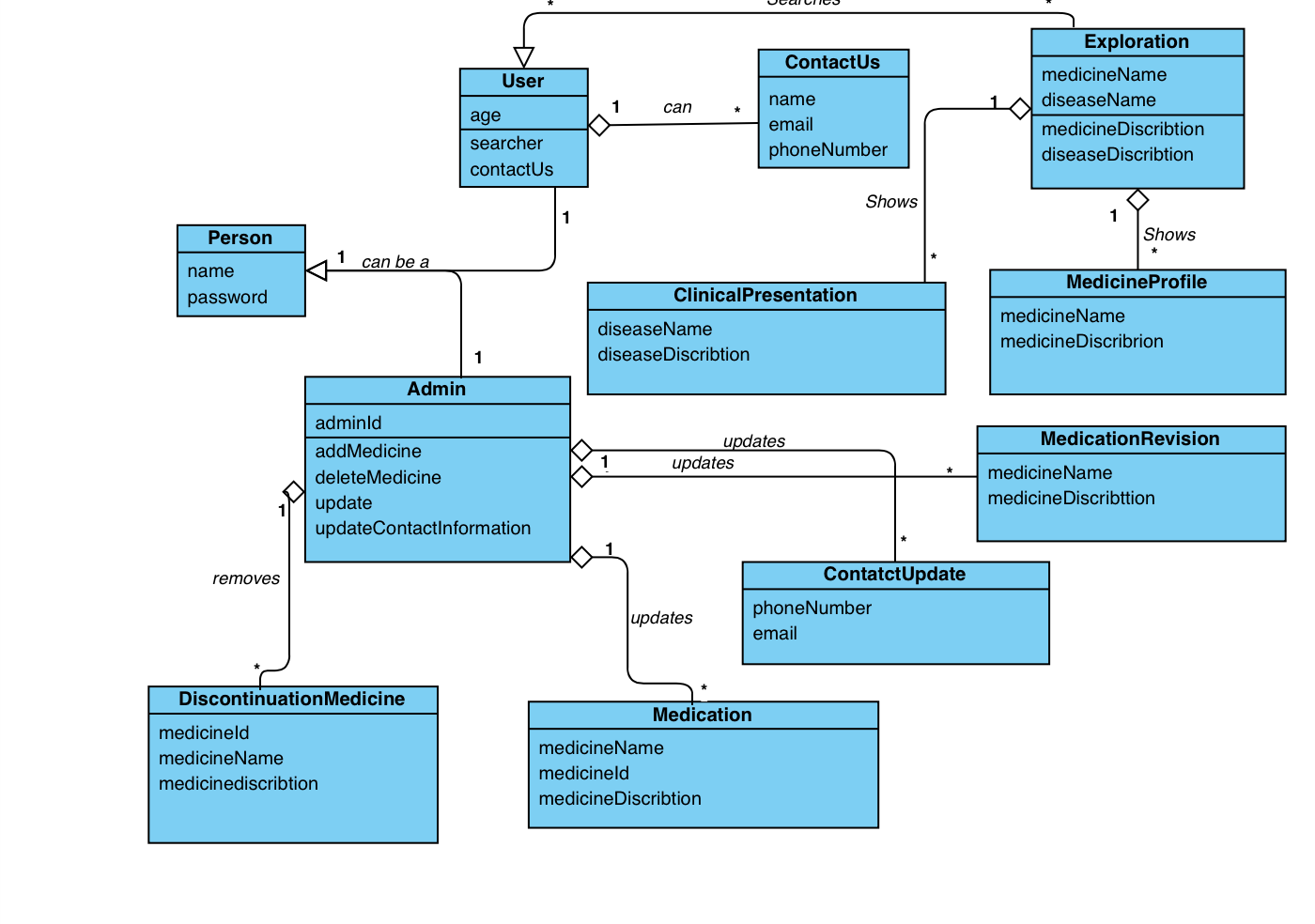
The website should use HTTP or HTTPS to communicate with the server-side code, depending on whether secure communication is required.

# Analysis Models

## 6.1 Use Case Diagram (UCD)



## 6.2 Analysis Class Diagram (ACD)



## 6.3 Data Flow Diagram (DFD)

**

# 7 Design Models

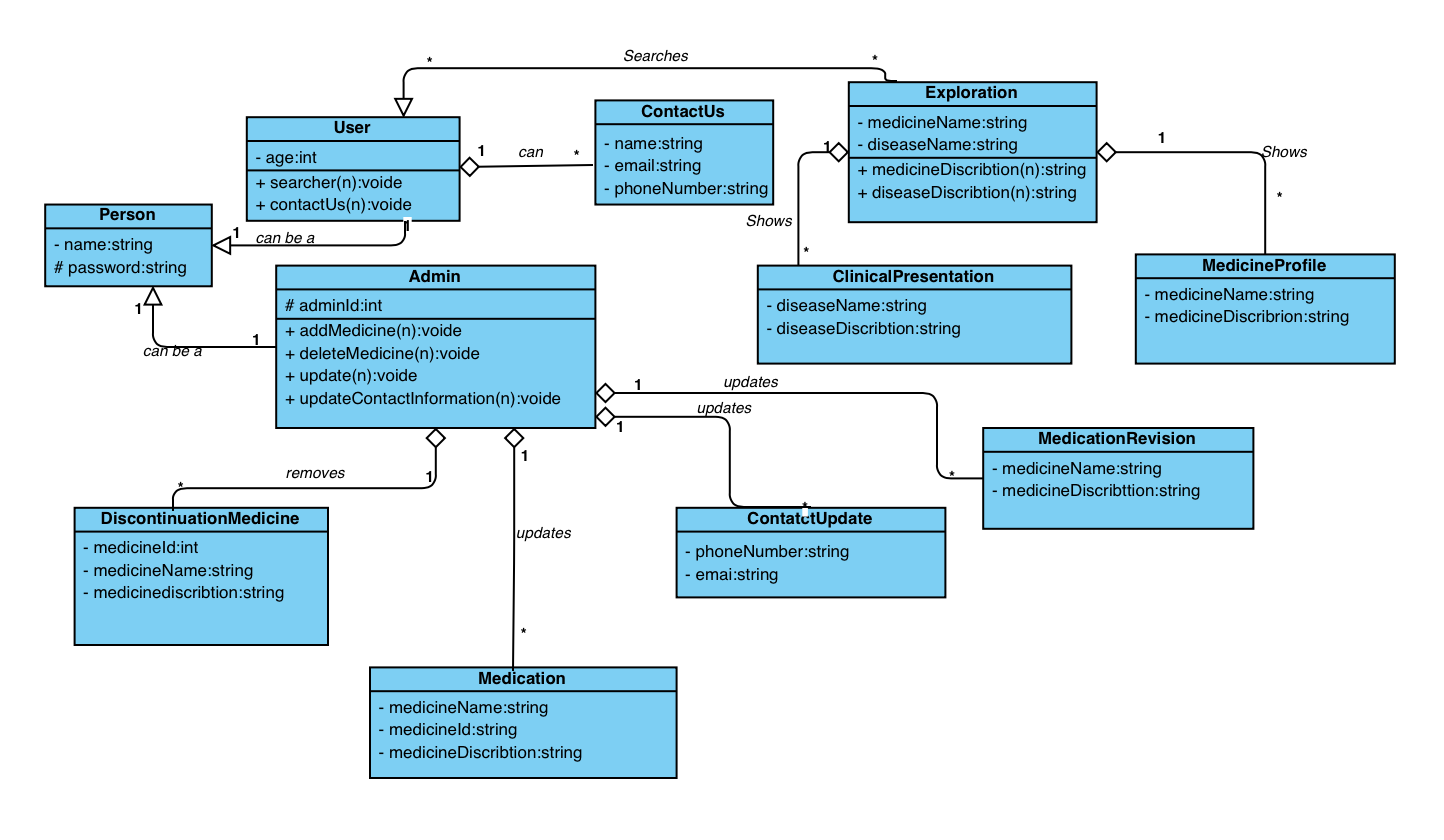
## 7.1 Architectural Design

Layered Architectural Style is provided below:

A screenshot of a computer

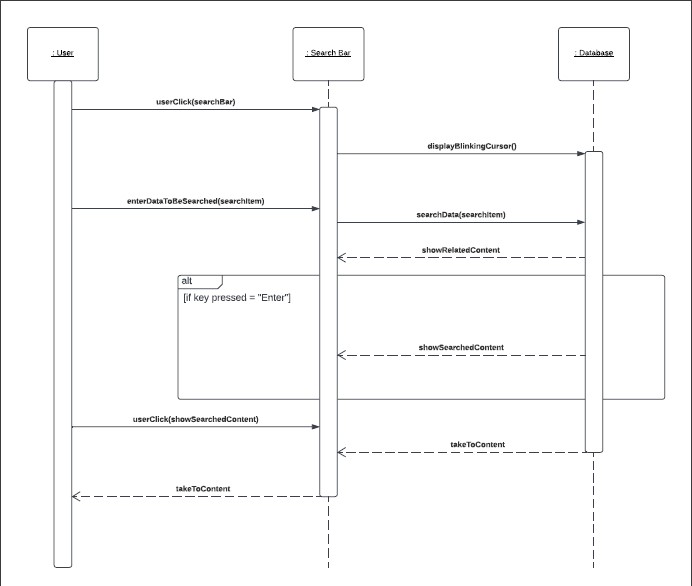
Description automatically generated with medium confidence

## 7.2 Design Class Diagram (DCD)

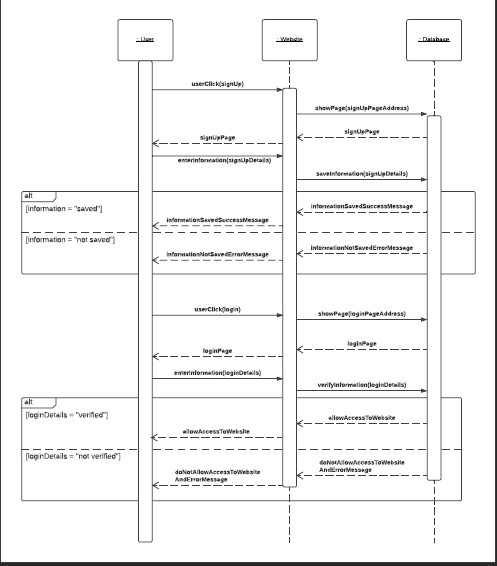


## 7.3 Design Sequence Diagrams (DSDs)

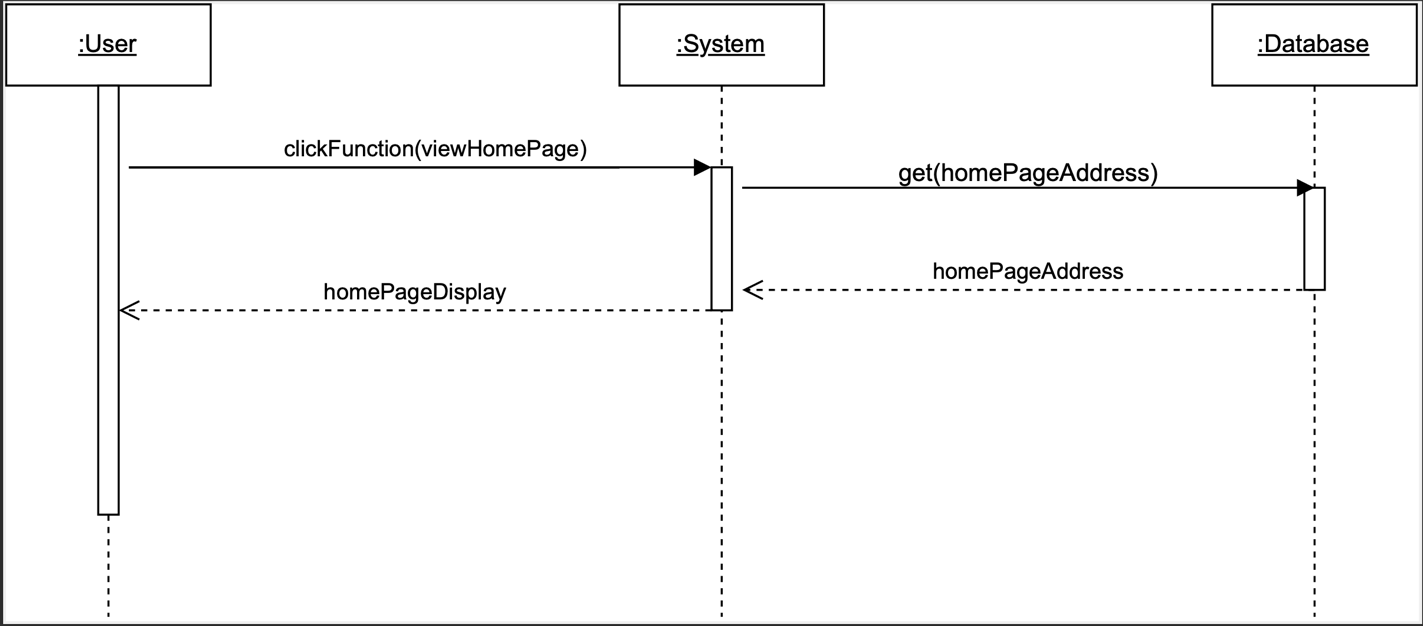
**Use-Case 1:**



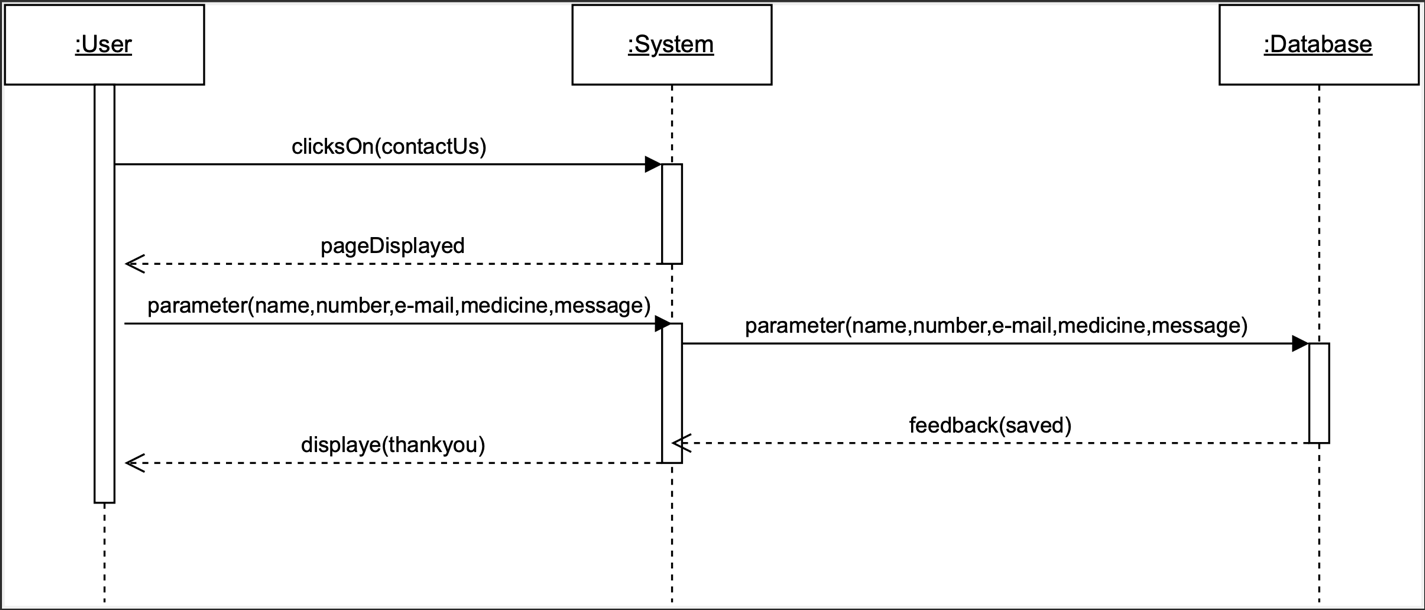
**Use-Case 2:**



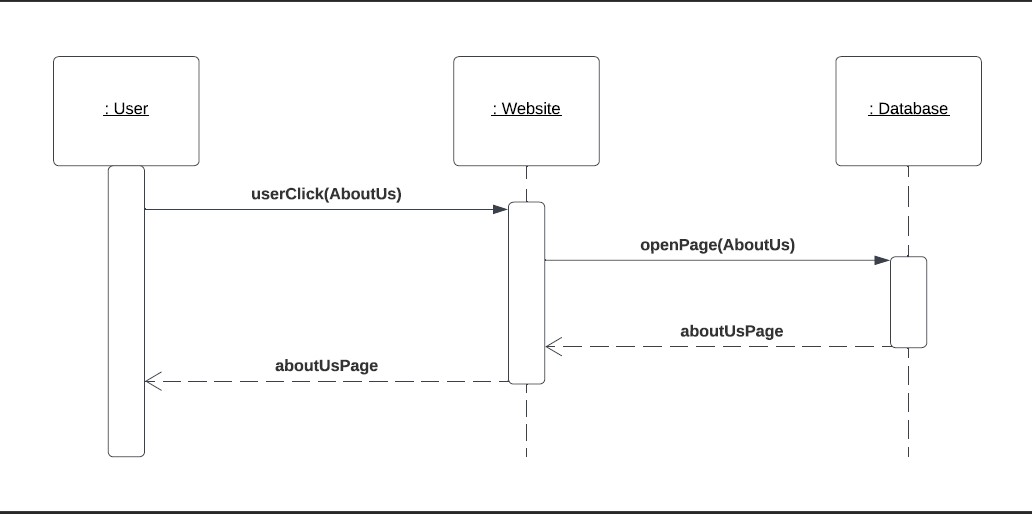
# Use-Case 3:



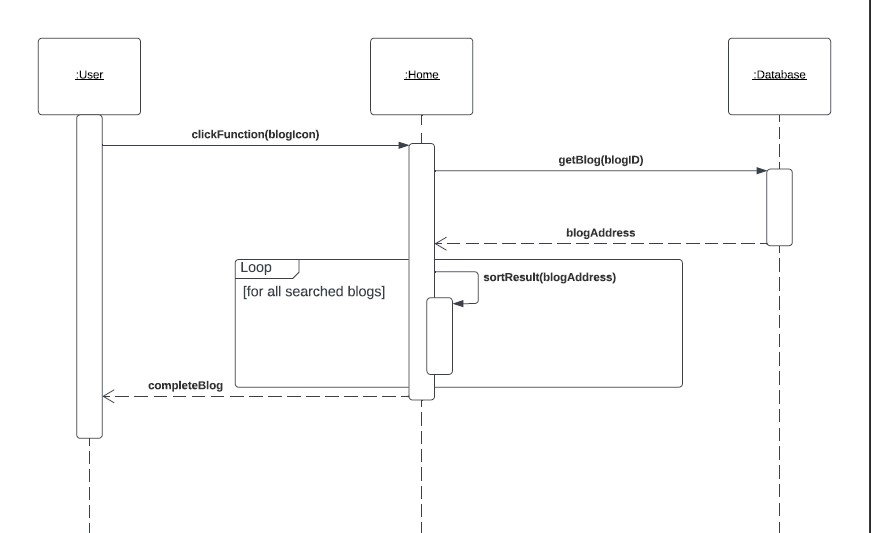
# Use-Case 4:

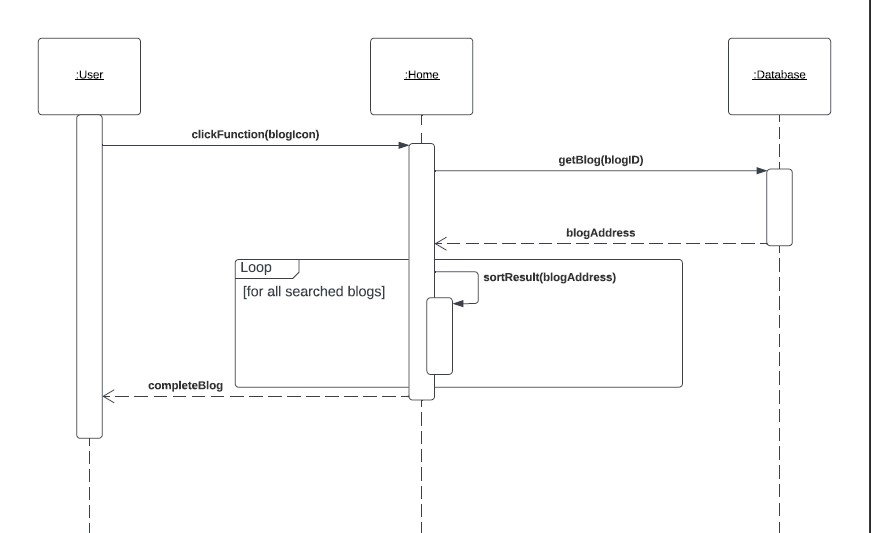


**Use-Case 5:**

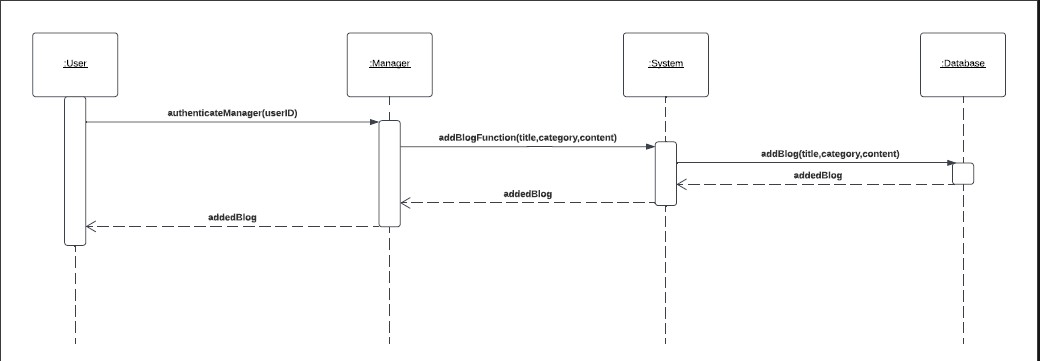
****

**Use-Case 6:**

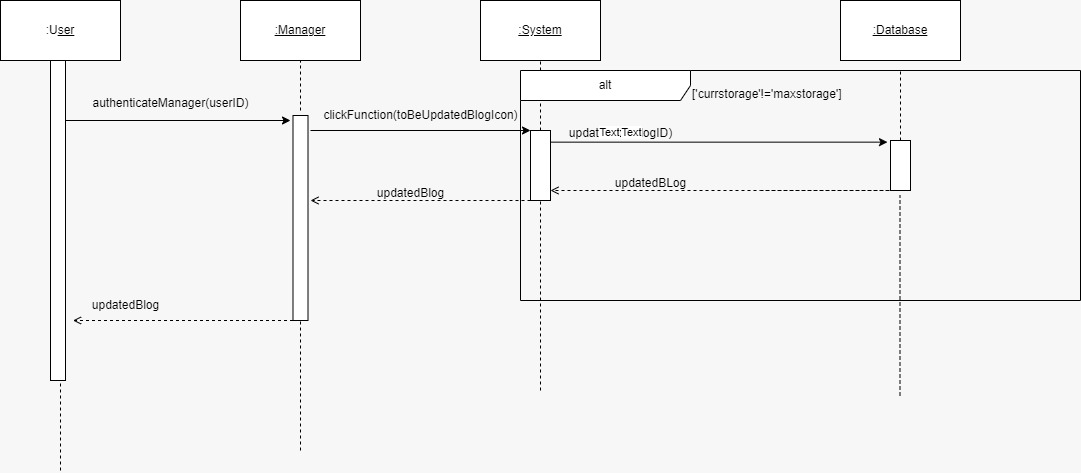
****

**Use-Case 7:**

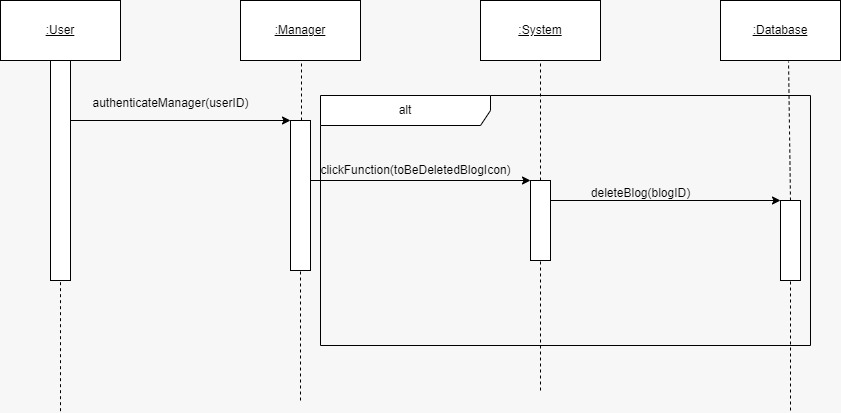
**Use-Case 8:**

****

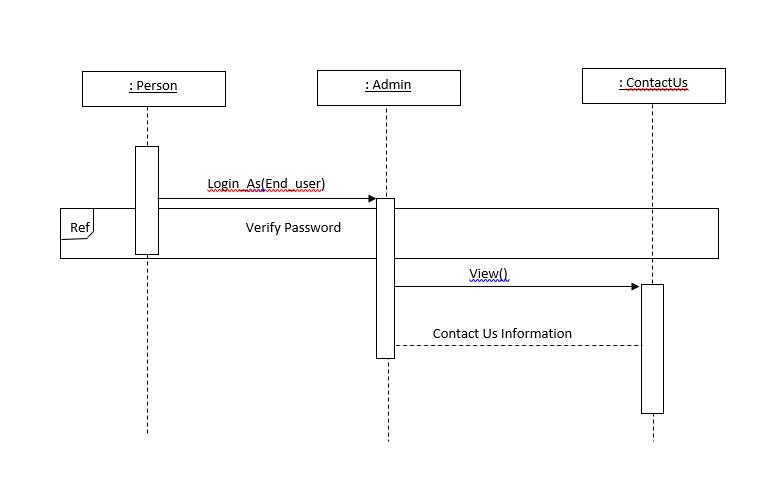
**Use-Case 9:**

****

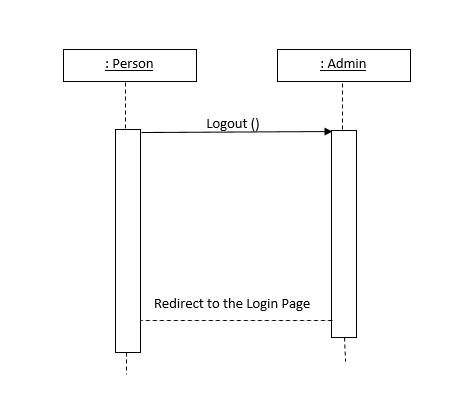
**Use-Case 10:**

****

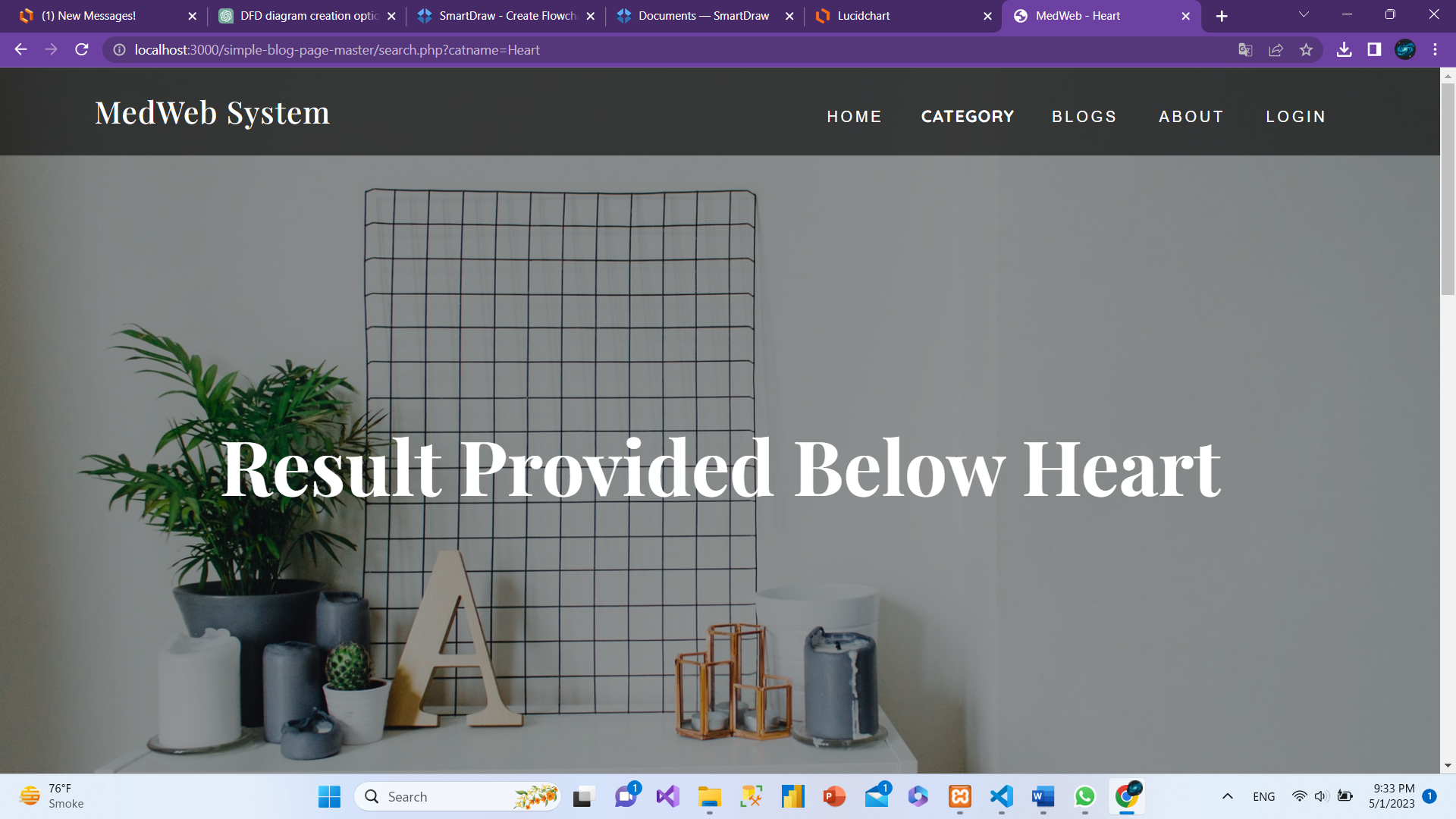
**Use-Case 11:**

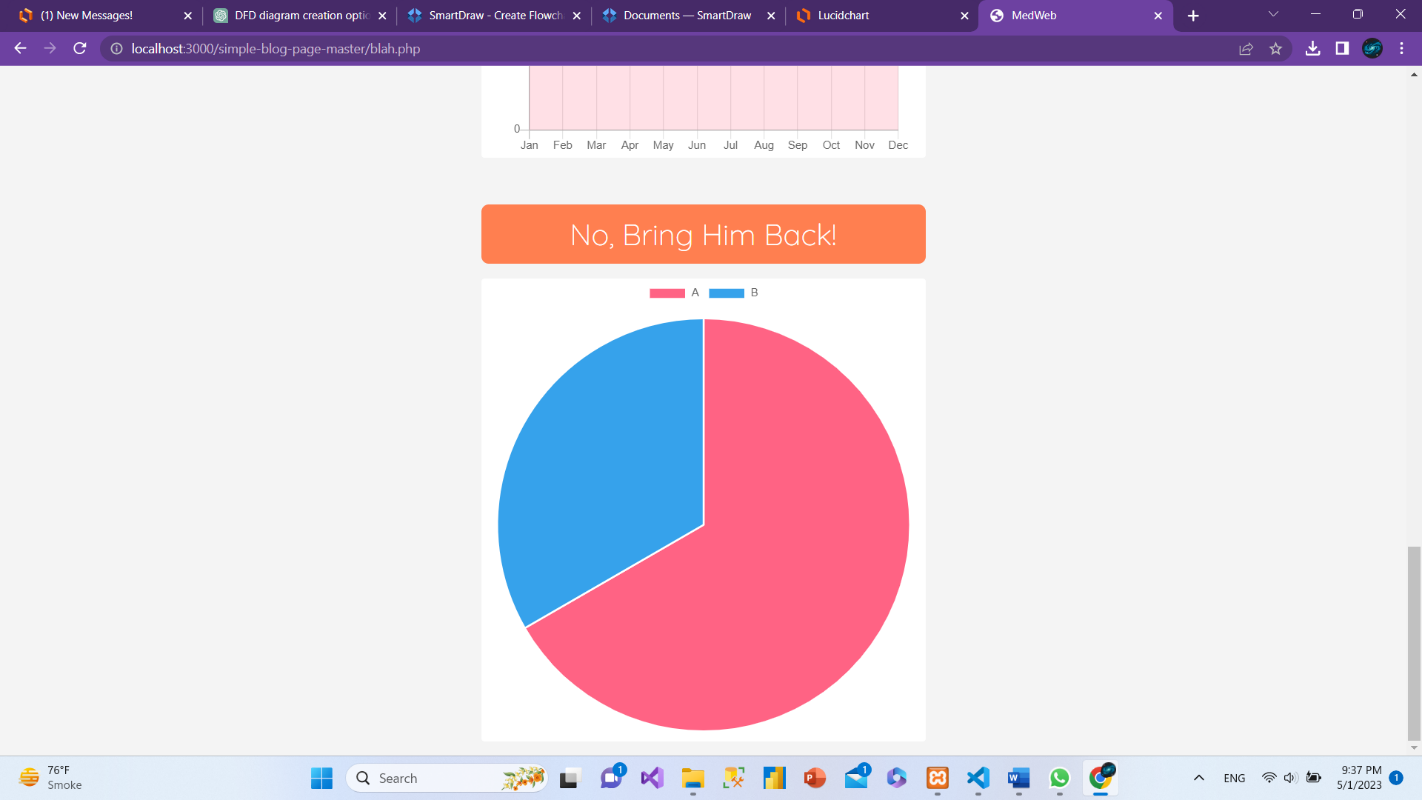
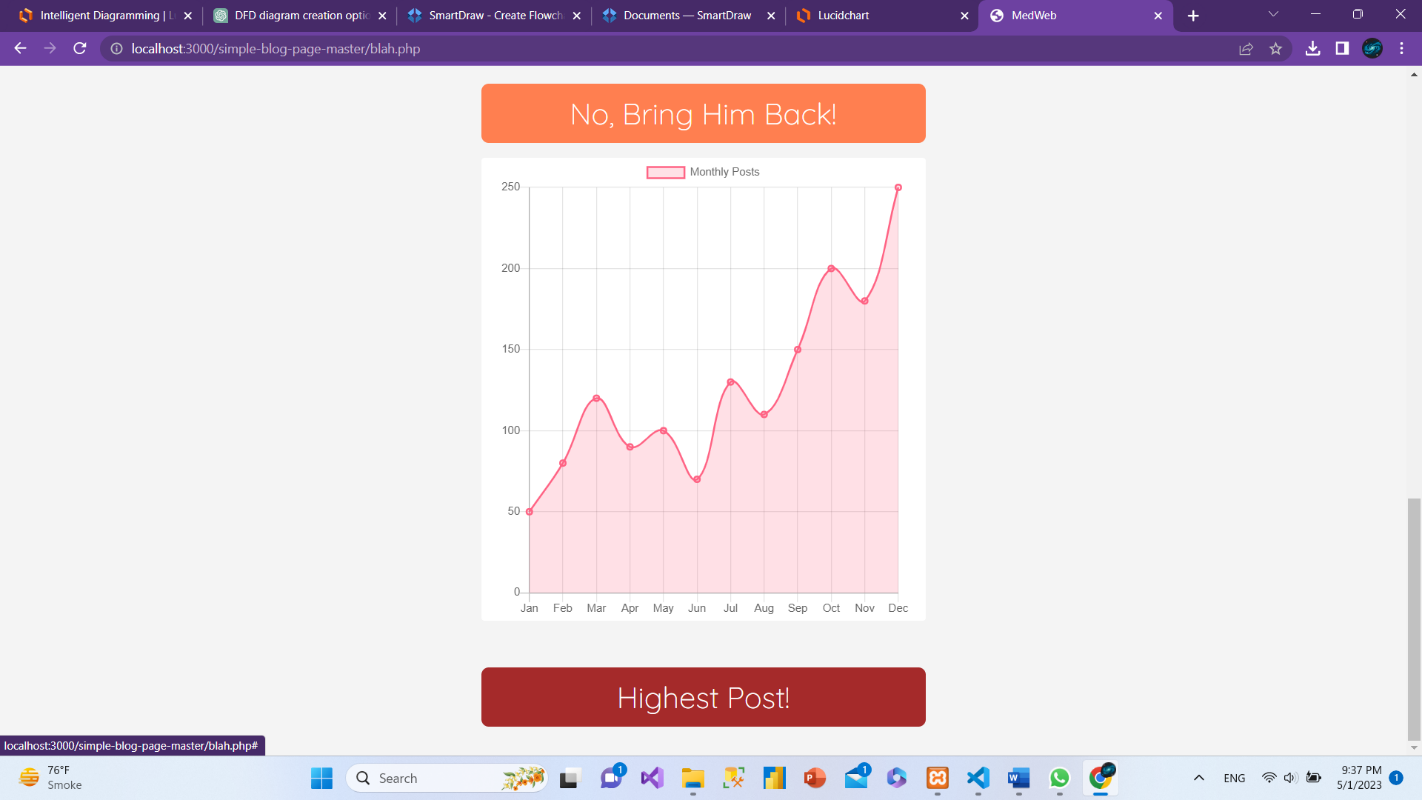
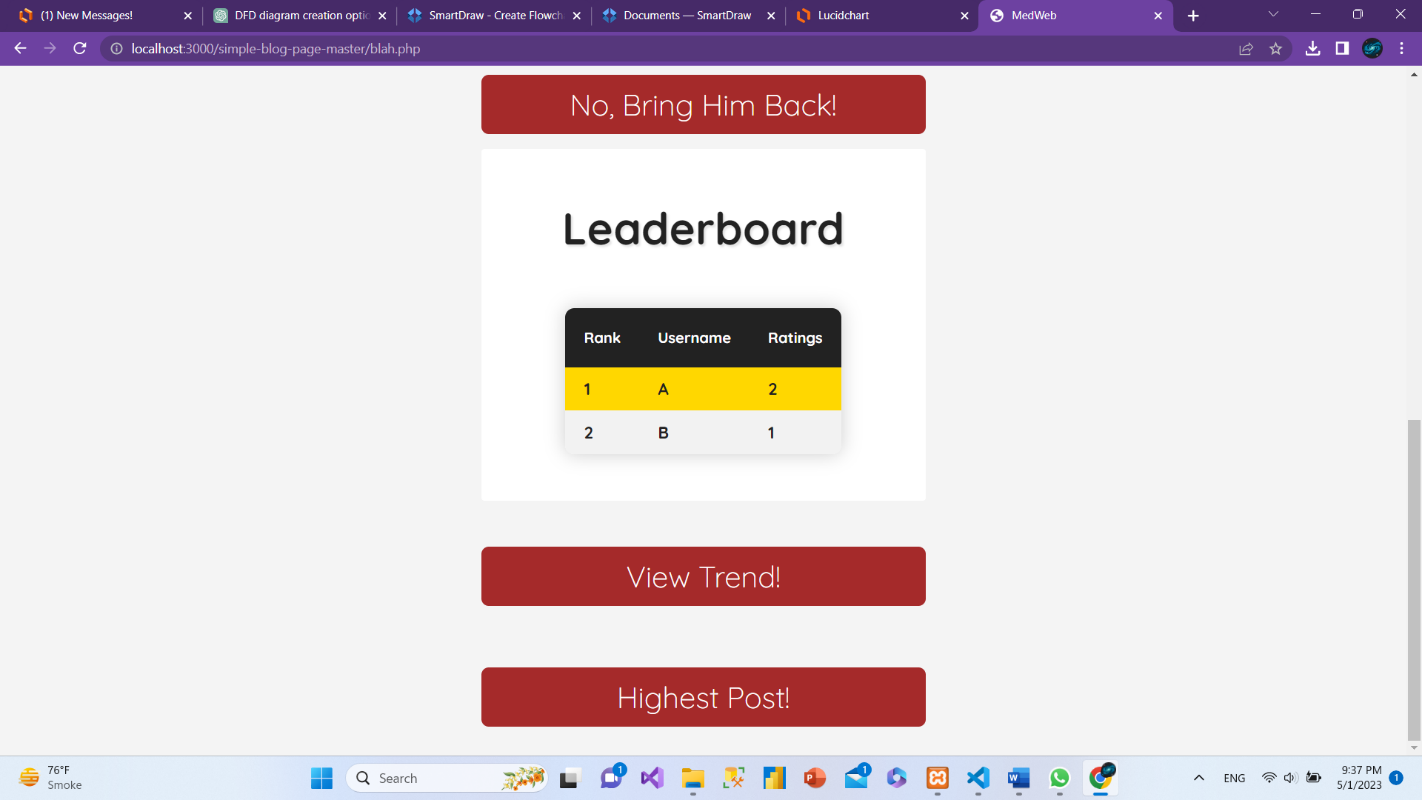
****

**Use-Case 12:**

****

# 8 GUI Design Graphical user interface, application Description automatically generatedGraphical user interface, application Description automatically generatedGraphical user interface, text Description automatically generatedGraphical user interface, application Description automatically generatedGraphical user interface, application Description automatically generatedGraphical user interface, text, application Description automatically generatedGraphical user interface, application Description automatically generatedGraphical user interface, application Description automatically generatedGraphical user interface Description automatically generatedGraphical user interface Description automatically generated





# 

# 

# 9. Test Cases

|  |  |
| --- | --- |
| **Identifier** | TC-1 |
| **Priority** | Low |
| **Related requirements(s)** | UC-9 |
| **Short description** | This tests the action of updating the website |
| **Pre-condition(s)** | The manager logs in to the website and navigates to the blog manager page |
| **Input data** | The manager clicks on “Edit Blog” |
| **Detailed steps** | 1. The manager clicks on the “Edit Blog” button on the navigation bar. 2. The website displays the blog manager page with a list of existing blog posts and options to edit each one of them. 3. The manager edits the desired blog by adding or removing information. 4. The manager clicks on ‘save’ |
| **Expected result(s)** | The blog is successfully updated. |
| **Post-condition(s)** | The manager can click on ‘Edit Blog’ again to update the website or move to another page using navigation bar. |

**Table 1: TC-1**

|  |  |
| --- | --- |
| **Identifier** | TC-2 |
| **Priority** | Low |
| **Related requirements(s)** | UC-9 |
| **Short description** | This tests the action of viewing the “Edit Blog” page when it is not working |
| **Pre-condition(s)** | The manager logs in to the website and navigates to the blog manager page. |
| **Input data** | The user clicks on “Edit Blog” |
| **Detailed steps** | 1. The user clicks on the “Edit Blog” button on the navigation bar. 2. The website displays the blog manager page with a list of existing blog posts and options to edit each one of them. 3. The manager edits the desired blog by adding or removing information. 4. The manager clicks on ‘save’ |
| **Expected result(s)** | The blog is not updated due to storage issues |
| **Post-condition(s)** | Shows the ‘storage full’ message. |

**Table 2: TC-2**

|  |  |
| --- | --- |
| **Identifier** | TC-3 |
| **Priority** | High |
| **Related requirements(s)** | UC-10 |
| **Short description** | Tests the action of successfully deleting an existing blog page. |
| **Pre-condition(s)** | The manager logs in to the website and navigates to the blog manager page. |
| **Input data** | Click on “Edit Blog” |
| **Detailed steps** | 1. Click on “Edit Blog” on the navigation bar. 2. The website displays the blog manager page with a form to delete each one. 3. The manager deletes the desired blog. 4. The manager clicks on ‘save’. |
| **Expected result(s)** | The blog is successfully deleted. |
| **Post-condition(s)** | The manager can click on ‘Edit Blog’ again to delete a blog post or move to another page using navigation bar. |

**Table 3: TC-3**

|  |  |
| --- | --- |
| **Identifier** | TC-4 |
| **Priority** | High |
| **Related requirements(s)** | UC-10 |
| **Short description** | Tests the action of unsuccessful attempts of deleting a blog post from the website. |
| **Pre-condition(s)** | The manager logs in to the website and navigates to the blog manager page. |
| **Input data** | Click on “Edit Blog” |
| **Detailed steps** | 1. Click on “Edit Blog” on the navigation bar. 2. The website displays the blog manager page with a form to delete each one. 3. The manager deletes the desired blog. 4. The manager clicks on ‘save’. |
| **Expected result(s)** | The blog page is not deleted due to the slow speed of internet. |
| **Post-condition(s)** | Shows a waiting message due to the slow speed of internet. |

**Table 4: TC-4**

|  |  |
| --- | --- |
| **Identifier** | TC-5 |
| **Priority** | Low |
| **Related requirements(s)** | UC-11 |
| **Short description** | This tests the action of viewing the “Contact Us” Page |
| **Pre-condition(s)** | The user has logged in to the website |
| **Input data** | The user clicks on “Contact Us” |
| **Detailed steps** | 1. The user clicks on the “Contact Us” button on the navigation bar. 2. Fill the form with the required information 3. Click on Submit |
| **Expected result(s)** | The feedback is submitted to the admin. |
| **Post-condition(s)** | Users can again fill out the form for “Contact Us” or move to another page using navigation bar |

**Table 5: TC-5**

|  |  |
| --- | --- |
| **Identifier** | TC-6 |
| **Priority** | Low |
| **Related requirements(s)** | UC-11 |
| **Short description** | This tests the action of viewing the “Contact Us” page when it is not working |
| **Pre-condition(s)** | The user has logged in to the website |
| **Input data** | The user clicks on “Contact Us” |
| **Detailed steps** | 1. The user clicks on the “Contact Us” button on the navigation bar. 2. Fill out the form with the required information 3. Click on Submit |
| **Expected result(s)** | The contact form is not submitted |
| **Post-condition(s)** | Shows the waiting message. |

**Table 6: TC-6**

|  |  |
| --- | --- |
| **Identifier** | TC-7 |
| **Priority** | High |
| **Related requirements(s)** | UC-12 |
| **Short description** | Tests the action of successfully logging out from the website. |
| **Pre-condition(s)** | The user is logged in to the website. |
| **Input data** | Click on “Logout” |
| **Detailed steps** | 1. Click on “Logout” on the navigation bar. |
| **Expected result(s)** | The User is logged out of the website |
| **Post-condition(s)** | The home Page is displayed |

**Table 7: TC-7**

|  |  |
| --- | --- |
| **Identifier** | TC-8 |
| **Priority** | High |
| **Related requirements(s)** | UC-12 |
| **Short description** | Tests the action of unsuccessful attempts of logging out from the website. |
| **Pre-condition(s)** | The user is logged in to the website. |
| **Input data** | Click on “Logout” |
| **Detailed steps** | 1. Click on “Logout” on the navigation bar. |
| **Expected result(s)** | The user is not logged out from the website sue to slow speed of internet. |
| **Post-condition(s)** | Shows a waiting message. |

**Table 8: TC-8**

|  |  |
| --- | --- |
| **Identifier** | TC-9 |
| **Priority** | Low |
| **Related requirements(s)** | UC-5 |
| **Short description** | This tests the action of viewing the “About Us” information. |
| **Pre-condition(s)** | The user is on the website’s home page. |
| **Input data** | Click on “About Us” |
| **Detailed steps** | 1. Click on “About Us” on the navigation bar. |
| **Expected result(s)** | The page is successfully loaded.  The “About Us” information is displayed. |
| **Post-condition(s)** | User can view more information by clicking “Read More”. |

**Table 9: TC-9**

|  |  |
| --- | --- |
| **Identifier** | TC-10 |
| **Priority** | Low |
| **Related requirements(s)** | UC-5 |
| **Short description** | This tests the action of displaying more information where “Read More” is pressed. |
| **Pre-condition(s)** | The user is on the home page of the website. |
| **Input data** | Click on “About Us” |
| **Detailed steps** | 1. Click on “About Us” button on navigation bar. 2. Click on “Read More” |
| **Expected result(s)** | The “About Us” information is displayed.  More information is displayed when “Read More” is pressed. |
| **Post-condition(s)** | User can go to any page using navigation bar. |

**Table 10: TC-10**

|  |  |
| --- | --- |
| **Identifier** | TC-11 |
| **Priority** | Low |
| **Related requirements(s)** | UC-5 |
| **Short description** | This tests the action of viewing “About Us” when there is no/slow internet connection. |
| **Pre-condition(s)** | User is on the website’s home page. |
| **Input data** | Click on “About Us”. |
| **Detailed steps** | 1. Click on “About Us” |
| **Expected result(s)** | The website does not take the user to “About Us” page. |
| **Post-condition(s)** | Displays the waiting message. |

**Table 11: TC-11**

|  |  |
| --- | --- |
| **Identifier** | TC-12 |
| **Priority** | Low |
| **Related requirements(s)** | UC-5 |
| **Short description** | This tests the action of viewing “Read More” when there is no further information |
| **Pre-condition(s)** | User is on the website’s home page. |
| **Input data** | Click on “About Us” |
| **Detailed steps** | 1. Click on “About Us” on the navigation bar. 2. Click on “Read More” |
| **Expected result(s)** | No changes are made i.e. no new information is displayed. |
| **Post-condition(s)** | Shows the message “no as much information available”. |

**Table 12: TC-12**

|  |  |
| --- | --- |
| **Identifier** | TC-13 |
| **Priority** | High |
| **Related requirements(s)** | UC-6 |
| **Short description** | This tests the action of displaying medicines according to the age. |
| **Pre-condition(s)** | The user is logged in to the website |
| **Input data** | Age = 10 |
| **Detailed steps** | 1. Click on the “Filter” drop-down menu 2. Click on “Age” 3. Enter the Age. |
| **Expected result(s)** | Medicines are displayed according to the entered age. |
| **Post-condition(s)** | Users can see the relevant medicines or can search for more. |

**Table 13: TC-13**

|  |  |
| --- | --- |
| **Identifier** | TC-14 |
| **Priority** | High |
| **Related requirements(s)** | UC-6 |
| **Short description** | This tests the action of displaying medicines according to the age when there is no/slow speed of internet. |
| **Pre-condition(s)** | The user is logged in to the website. |
| **Input data** | Age = 10 |
| **Detailed steps** | 1. Click on “Filter” drop-down menu. |
| **Expected result(s)** | The contents of drop-down menu are not displayed/loaded. |
| **Post-condition(s)** | Shows a waiting message due to the slow/no internet. |

**Table 14: TC-14**

|  |  |
| --- | --- |
| **Identifier** | TC-15 |
| **Priority** | High |
| **Related requirements(s)** | UC-6 |
| **Short description** | This tests the action of displaying medicines according to the age when faulty input i.e. age is given. |
| **Pre-condition(s)** | The user has logged in to the website. |
| **Input data** | Age = -20 |
| **Detailed steps** | 1. Click on “Filter” drop-down menu. 2. Click on “Age” 3. Enter the age |
| **Expected result(s)** | No content is displayed.  Shows an error message. |
| **Post-condition(s)** | User can again enter the correct age. |

**Table 15: TC-15**

|  |  |
| --- | --- |
| **Identifier** | TC-16 |
| **Priority** | High |
| **Related requirements(s)** | UC-1 |
| **Short description** | This will be help to user to search “content” |
| **Pre-condition(s)** | The user will signup or login first |
| **Input data** | Name of medicine and disease |
| **Detailed steps** | 1. User will login or signup 2. Search the medicine or disease |
| **Expected result(s)** | website will show medicine related to disease |
| **Post-condition(s)** | The website show a list of reverent medicine |

**Table 16: TC-16**

|  |  |
| --- | --- |
| **Identifier** | Tc-17 |
| **Priority** | High |
| **Related requirements(s)** | UC-2 |
| **Short description** | At this page the user will login or sign up |
| **Pre-condition(s)** | User must have internet |
| **Input data** | User will enter name and password |
| **Detailed steps** | 1. User search website 2. To search user must have internet 3. User will login or sign up by completing requirements |
| **Expected result(s)** | Login or sign up page will appear |
| **Post-condition(s)** | The main page of website will appear |

**Table 17: TC-17**

|  |  |
| --- | --- |
| **Identifier** | TC-18 |
| **Priority** | Medium |
| **Related requirements(s)** | UC-7, SRS section 3.2 (View Blogs Post) |
| **Short description** | This test case is to verify that the website displays the selected blog post when the user clicks on a blog post from the list search results. |
| **Pre-condition(s)** | * The user is on the search results page. * The search results contain at least one blog post related to the clicked medicine. |
| **Input data** | * A list of blog posts related to the clicked medicine. * Click event on the blog post. |
| **Detailed steps** | 1. User clicks on the name of the medicine from the list of search results. 2. The website retrieves the list of blog posts related to the clicked medicine. 3. User clicks on one of the blog posts from the list of related blog posts. |
| **Expected result(s)** | * The website displays the selected blog post related to the clicked medicine. * The displayed blog post contains relevant information about the medicine. * The displayed blog post is easy to read and understand. * The user can navigate back to the search results page |
| **Post-condition(s)** | * The user has successfully viewed the selected blog post related to the clicked medicine. * The user can continue to search for other medicines or blog posts. |

**Table 18: TC-18**

|  |  |
| --- | --- |
| **Identifier** | TC-19 |
| **Priority** | High |
| **Related requirements(s)** | UC-8, SRS section 3.2.1 (Blog Post Storage) |
| **Short description** | Test adding a new blog post when there is enough space in storage. |
| **Pre-condition(s)** | Manager is logged in and navigated to the blog manager page. |
| **Input data** | Valid blog post information (e.g. title, content, tags) |
| **Detailed steps** | 1. Click on the "Add Blog Post" button on the blog manager page. 2. Fill out the form with valid blog post information. 3. Click the "Save" button. |
| **Expected result(s)** | Blog post is successfully added to the database and the blog manager page is displayed with the new blog post added to the list. |
| **Post-condition(s)** | New blog post is successfully added to the database. |

**Table 19: TC-19**

|  |  |
| --- | --- |
| **Identifier** | TC-20 |
| **Priority** | Medium |
| **Related requirements(s)** | UC-8, SRS section 3.2.1 (Blog Post Storage) |
| **Short description** | Test adding a new blog post when there is not enough space in storage. |
| **Pre-condition(s)** | Manager is logged in and navigated to the blog manager page. |
| **Input data** | Valid blog post information (e.g. title, content, tags) that exceeds the remaining storage capacity. |
| **Detailed steps** | 1. Click on the "Add Blog Post" button on the blog manager page. 2. Fill out the form with valid blog post information that exceeds the remaining storage capacity. 3. Click the "Save" button. |
| **Expected result(s)** | An error message is displayed indicating that there is not enough space to add the blog post. |
| **Post-condition(s)** | No new blog post is added to the database. |

**Table 20: TC-20**

|  |  |
| --- | --- |
| **Identifier** | TC-21 |
| **Priority** | Low |
| **Related requirements(s)** | UC-3 |
| **Short description** | This tests the action of viewing the home page. |
| **Pre-condition(s)** | The user logs in to the website and navigates the website |
| **Input data** | The user clicks on view home page |
| **Detailed steps** | 1. The user clicks on the “Home Page” button on the navigation bar. 2. The website displays the main page. |
| **Expected result(s)** | The home page is successfully viewed. |
| **Post-condition(s)** | The user can view the home page with links to other pages. |

**Table 21: TC-21**

|  |  |
| --- | --- |
| **Identifier** | TC-22 |
| **Priority** | Low |
| **Related requirements(s)** | UC-3 |
| **Short description** | This tests the action of viewing the “Home Page” when the speed of internet is slow. |
| **Pre-condition(s)** | The user logs in to the website and navigates the website. |
| **Input data** | The user clicks on view home page |
| **Detailed steps** | 1. The user clicks on the “Home Page” button on the navigation bar. 2. The website displays the waiting message |
| **Expected result(s)** | The home page is not displayed |
| **Post-condition(s)** | Shows waiting message. |

**Table 22: TC-22**

|  |  |
| --- | --- |
| **Identifier** | TC-23 |
| **Priority** | Low |
| **Related requirements(s)** | UC-2 |
| **Short description** | Tests the action of successfully contacting us. |
| **Pre-condition(s)** | The user fills up the contact us form and clicks on submit button. |
| **Input data** | Click on “Contact us form”.  Name=Ali  Phone number=0323445566  [Email=ali@gmail.com](mailto:Email=ali@gmail.com) |
| **Detailed steps** | 1. Click on “Contact us form” on the navigation bar. 2. The website displays the page where you can submit your feedback. 3. User enters name. 4. Saves name in database. 5. User enters phone number 6. Saves phone number in database. 7. User enters email. 8. Saves email ID in database. 9. User selects medicine. 10. Selects the clicked medicine. 11. User enters message. 12. Saves your message |
| **Expected result(s)** | The user is able to contact us successfully. |
| **Post-condition(s)** | The website sends an email to the website’s contact email address with the user’s message. |

**Table 23: TC-23**

|  |  |
| --- | --- |
| **Identifier** | TC-24 |
| **Priority** | Low |
| **Related requirements(s)** | UC-4 |
| **Short description** | Tests the action of successfully contacting us. |
| **Pre-condition(s)** | The user fills up the contact us form and clicks on submit button. |
| **Input data** | Click on “Contact us form”.  Name=Ali  Phone number=0323445566  [Email=ali@gmail.com](mailto:Email=ali@gmail.com) |
| **Detailed steps** | 1. Click on “Contact us form” on the navigation bar. 2. Shows waiting message due slow internet speed. 3. User enters name. 4. Shows error message due to storage issues. 5. User enters phone number 6. Shows error message due to storage issues. 7. User enters email. 8. Shows error message due to storage issues. 9. User selects medicine. 10. There is no such medicine in database. 11. User enters message. 12. Shows error message due to storage issues. |
| **Expected result(s)** | The user is not able to contact us successfully. |
| **Post-condition(s)** | Error message is displayed |

**Table 24: TC-24**

# 10. IV & V Report

**IV & V Resource**

Zenab Rizwan 21L-5640

Name Roll # Signature

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S#** | **Defect Description** | **Origin Stage** | **Status** | **Fix Time** | |
| **Hours** | **Minutes** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | Name of identifier was wrong in test cases | Test case | Fixed | 00 | 20 |
| 2 | Name of system and database was missing in DSD | DSD | Fixed | 00 | 50 |
| 3 | Risk calculation was wronged | Risk report | Fixed | 1 | 15 |
| 4 | Order of risk exposure was wrong | Risk report | Fixed | 1 | 00 |

**Table 6: List of non-trivial defects**

# 11. Risk Report

**1Project Risks**

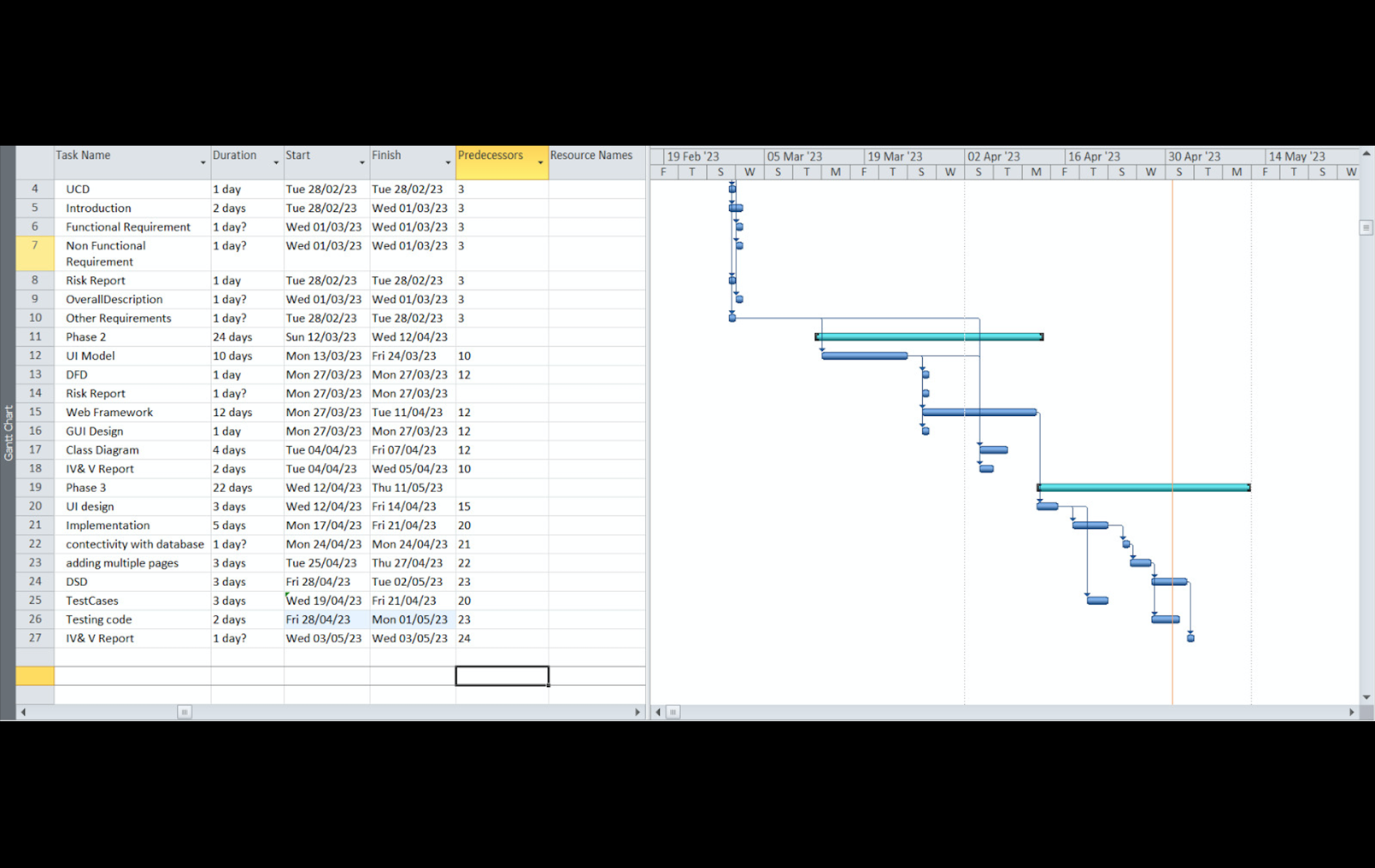
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Risk Description** | **Impact**  **(1 – 10)** | **Probability (0 – 1)** | **2Risk Exposure** | **Weeks Active** | **Mitigation Strategy** |
| Inaccurate or unreliable information negatively impacting the success of the project | 10 | 0.9 | 9 | 2 | Up to date data  Data should come from reliable sources and provided by medical expert |
| Non-compliance with applicable laws, regulations, and licenses leading to legal liabilities, financial penalties, reputational damage, and potential project delays or cancellations | 9 | 0.8 | 7.2 | 6 | Proper documentation should be available from WHO approval |
| Cyber-attack, unauthorized access, or other security incidents resulting in data loss, theft, or damage, compromising the confidentiality, integrity, and availability of sensitive information, and potentially exposing the organization to legal and financial liabilities, loss of reputation | 8 | 0.6 | 4.8 | 1 | Implementing encryptions and firewalls  Code review |
| Negative user feedback due to poor user experience, usability issues, or other factors resulting in decreased customer satisfaction, reduced user adoption, and potential brand damage | 7 | 0.4 | 2.8 | 4 | * Provide clear guidelines. * Responds promptly. * Take actions |
| Risk of reduced customer trust and loyalty due to poor website usability, design, and functionality, leading to decreased website traffic | 7 | 0.3 | 2.1 | 1 | Friendly or intuitive |

# 12. Activity Timesheet

*<Time is cumulative for all team members. However, a separate timesheet shall be used for each phase.>*

|  |  |  |
| --- | --- | --- |
| **Activity** | **Time** | |
| **Hours** | **Minutes** |
| Requirements Engineering | 6 | 25 |
| Analysis and Design | 14 | 00 |
| Implementation | 15 | 00 |
| Testing | 2 | 00 |
| Deployment | 3 | 00 |
| Project Management | 1 | 40 |
| IV & V | 2 | 10 |

# 13 Updated Project Plan



**Project Manager**

Muhammad Ahmad 21L-5617

Name Roll # Signature