

G NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCE
(for women), Shaikpet.
SOFTWARE ENGINEERING LAB

1. Introduction
 - 1.1. Purpose
 - 1.2. Scope
 - 1.3. Definitions, Acronyms and Abbreviations
 - 1.4. Objectives
 - 1.4.1. Main objective
 - 1.4.2. Specific objectives
 - 1.5. References
 - 1.6. Overview
2. General description
 - 2.1. Product perspective
 - 2.2. Product function summary
 - 2.3. User characteristics
 - 2.4. General constraints
 - 2.5. Assumption and dependencies
3. Specific Requirements
 - 3.1. Functional requirements
 - 3.2. External interface requirements
 - 3.3. Performance requirements
 - 3.4. Design constraints
 - 3.5. Attributes
 - 3.6. Other requirements
4. Appendices
5. Index

G NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCE
(for women), Shaikpet.
SOFTWARE ENGINEERING LAB

1. Introduction

1.1 Purpose

The demand for emergency medical diagnosis is particularly acute in developing nations like India, where access to trained and certified medical workers is limited. Considering the impact of the COVID-19 pandemic, it has dramatically changed how outpatient care is delivered in healthcare practices. Researches confirmed that 70 thousand cases of death, which happen yearly in the world, were because of the misprescribing of the drug itself or its dose (overdose or lower amount).

1.2 Scope

Many of the patients have some troubles when they visit the doctor to identify suitable treatment for them. After the doctor has diagnosed the disease, another problem appears; the drug might be written nonspecific. Since the choice of the drug depends on the patient's disease history and whether he had any chronic diseases, careful assessment should be done as the drug is given. Expert systems mimic the problem-solving activity of human experts in specialized domains by capturing and representing expert knowledge. This project employs expert systems, including a knowledge database and a user-friendly interface from which accurate conclusions are derived.

The project's scope is restricted to regularly prescribed medications for general clinical illnesses. The system provides the physician with a list of available alternative medicines, their possible side effects when used alone and in combination with other medications, and dosages and precautions.

1.3 Definitions, Acronyms and Abbreviations

S.No	Stakeholder	Acronyms	Definition
1.	Patient	P	Patient is one of the naive end users with a medical report.
2.	Doctor	D	Doctor is another end user who deals with severe patient appointments.
3.	Administrator	A	Administrator is responsible for dealing with general requirements.
4.	System Maintenance	S	System Maintenance is responsible for performing systems checks and assuring the non-functional requirements.

G NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCE
(for women), Shaikpet.
SOFTWARE ENGINEERING LAB

5.	Government	G	Government affects the project through its policies and regulations.
----	------------	---	--

1.4 Objective

1.4.1 Main Objective:

The main objective of this project is to deliver reasoned advice to the customer at a level comparable to that provided by a human expert, offering a 24/7 service from the comfort of their own homes in a cost-effective manner.

1.4.2 Specific Objectives:

1. To build a predictive model that preserves prediction accuracy.
2. To assist clinicians by giving medical decision-making strategies.
3. To assist the illiterate and disabled with natural language and translation.
4. To establish two distinct user login/sign-up portals for doctors and patients.
5. To provide a disclaimer about the prescribed medications and to recommend alternatives to improve and maintain good health.
6. To refer a patient to a doctor if they are experiencing severe symptoms.
7. To keep track of the users' previous medical histories.
8. To maintain an expert system that is proactive and responds to every facet of the situation.
9. To create an expert system that is simple to use and comprehend.

1.5 References

- IEEE Std 830-1998 (Revision of IEEE Std 830-1993)
- IEEE Std 828-1998, IEEE Standard for Software Configuration Management Plans.
- IEEE Std 610.12-1990, IEEE Standard Glossary of Software Engineering Terminology.
- IEEE Std 730-1998, IEEE Standard for Software Quality Assurance Plans.
- R. Pressman 5E, reference book

1.6 Overview

This software is responsible for automating the task of prescribing medicines. It makes a reliable diagnosis after analysing user symptoms and pre medical history and prescribes medicines to users efficiently.

2.General Description

2.1 Product Perspective

The product helps the users to enter the detailed information about patient's symptoms and their past medical history. According to the symptoms and details given by the user, the expert system shall prescribe the medicines to them.

It also enables the doctor to view the profiles of the patients and can also help patients with severe conditions and prescribe medicines to them.

2.2 Product Function Summary

- The main objective of this product is to reduce the manual work of both the users and the doctors.
- This product is developed to broaden the accessibility of medical services to people round the clock from any corner of the world.
- This software is capable of managing multiple queries from the users and also providing the multiple solutions for those queries so that the users can pick the best solutions.

2.3 User Characteristics

We have 3 levels of users.

a. User module: In the user module, users (mostly patients) will log into the system, enter the information regarding their health issues, submit the queries and search for their healthcare solutions.

b. Doctor module: In the doctor module, doctor logs into the system, refers the patient's medical condition and prescribes the required medicines. This also helps as a wonderful tool for their daily practice.

2.4 General Constraints

- The user should have a valid user id, details and password.
- Users with valid identification proof are allowed to register in this software.
- The user cannot get accessed unless and until he is registered with the valid user information.
- The users cannot get access to the doctor's account.

2.5 Assumptions and Dependencies

All the data entered will be correct and up to date. This software product is developed using python and GUI as a front end. SQL is used as the back end which is supported by Windows.

3. Specific Requirements

3.1 Functional requirements

3.1.1 Doctor

- The system should validate the entire name and password and log the doctor into the system.
- The system shall enable doctors to provide detailed information about diseases, their symptoms and medicines to be used.

3.1.2 Patient

Functional Requirements of Patients:

- The system should validate the details and log the patient into the system.
- Patient must enter all the symptoms that occur for him/her for the expert system to prescribe medicines.
- Patient must also provide his/her complete medical history.
- Patient must receive a snapshot of the prescription.
- The system should go through the database and prescribe the medicines and send a copy to the patient and the pharmacist.
- Pharmacist contact details must be shared with the patient.
- Patient's dashboard has to be updated for future use.
- The system is user friendly and feasible for patients.
- If the medicines aren't available then an alternative should be suggested.

Non - functional Requirements of Patients:

Privacy:

- Patient's details will be confidential.
- Patient's details and medical history won't be shared with any third person.

User friendly:

- Patients are notified whenever required. It is convenient to use.
- The system is highly responsive and easy to use.

3.1.3 Admin

- The expert system is managed by an administrator, who overlooks the activities of the expert system.
- The administrator must have the main control and is also responsible for making the

G NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCE
(for women), Shaikpet.
SOFTWARE ENGINEERING LAB

updates in the expert system on a regular basis so that the patients will not have any access to obsolete data.

- The administrator must send notifications and inform about the latest features of the expert systems to the patients and pharmacists.
- The patients and pharmacists should be able to contact the administration in case of any problems or queries.
- The administrator must maintain a record of all the patients, pharmacists and doctors registered on his database.
- The administrator can exercise the authority to remove any pharmacist from his system if he doesn't comply with the protocols of the expert system.
- The administrator must have a unique dashboard on his expert system through which he can perform all his functions.

Non-functional requirements of Administrator:

Security:

- The administrator must have a special Sign-in to the expert system, which directs him to the admin dashboard, from which he can manage all the activities of the expert system.
- The administrator must have a two-factor authentication (i.e. OTP validation) every time he tries to log-in into the expert system, this feature will ensure high security to his account.

User-friendly:

- The administrator must easily access the records of all the users (patients, pharmacists) from any location.

Interoperability:

- The administrator must take the responsible for updating the Medical Database for prescribing medicines through the help of doctors.
- The administrator must notify the developer about the system errors and bugs, so that the developers can fix it timely and avoid inconvenience to users.

3.1.4 System management

- The developer must be responsible for developing the expert system.
- The developer must simultaneously develop new updates and feature in order to make the expert system easier and more convenient to access.
- The developer must check the system errors and bugs regularly and fix them accordingly.

G NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCE
(for women), Shaikpet.
SOFTWARE ENGINEERING LAB

- The developer must be able make any kind of changes to the structure of the expert system according to the wants of the users or the administrator.

Non-functional requirements of Developer:

Interoperability:

- The developer must notify the administrator of the changes or updates he makes, who in turn will let the users know about it.

The developer must be reliable and must be able to provide security to all of its users, which would prevent any kind of security breach in the expert system.

Performance Requirements

Security

- The expert system is highly secured
- Every user will have their own private medical records in their own modules which are confidential and must acquire the user's permission to be accessed.

Reliability

- Every user must log in to access through their account to access any data or to acquire any prescription required.
- The expert system also offers a very reliable medicines delivery service to its users from their nearby pharmacies.
- The system also provides user support for queries.

Scalability

- The expert system has the ability to accommodate any increasing number of users, number of SQL statements that can run and provide results simultaneously.

Response Time

- The Expert system must provide every response to its users within milliseconds.

User Interface

- Every user (say, patients, doctors, pharmacists, administrator) have their own kind of

User Interface.

- They differ from each other because every stakeholder has his own different set functions on an expert system.
- Every system provides the use of icons and toolbars.

G NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCE
(for women), Shaikpet.
SOFTWARE ENGINEERING LAB

Throughput

- The system may easily handle 10,000 users requests for prescriptions per hour.
- But can only provide delivery services of medicines to a maximum of 7000 patients per hour via expert system.

Recovery Time

- The expert system must have recovery time ranging from a minimum of 1 hour to a maximum of 24 hours depending on the error occurred.

3.1.5 Government

- The government is responsible for issuing securities and policies of the product.
- It also promotes the license of the product and enables the functioning of the stakeholders.

3.2 External interface requirements

3.2.1 User Interfaces

Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.

3.2.2 Hardware Interfaces

Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.

G NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCE
(for women), Shaikpet.
SOFTWARE ENGINEERING LAB

3.2.3 Software Interfaces

Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.

3.2.4 Communications Interfaces

Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.

3.3 Performance requirements

Security

- The expert system is highly secured
- Every user will have their own private medical records in their own modules which are confidential and must acquire the user's permission to be accessed.

Reliability

- Every user must log in to access through their account to access any data or to acquire any prescription required.
- The expert system also offers a very reliable medicines delivery service to its users from their nearby pharmacies.
- The system also provides user support for queries.

Scalability

- The expert system has the ability to accommodate any increasing number of users, number of SQL statements that can run and provide results simultaneously.

Response Time

- The Expert system must provide every response to its users within milliseconds.

G NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCE
(for women), Shaikpet.
SOFTWARE ENGINEERING LAB

User Interface

- Every user (say, patients, doctors, pharmacists, administrator) have their own kind of User Interface.
- They differ from each other because every stakeholder has his own different set functions on an expert system.
- Every system provides the use of icons and toolbars.

Throughput

- The system may easily handle 10,000 users requests for prescriptions per hour.
- But can only provide delivery services of medicines to a maximum of 7000 patients per hour via expert system.

Recovery Time

- The expert system must have recovery time ranging from a minimum of 1 hour to a maximum of 24 hours depending on the error occurred.

3.4 Design constraints

3.4.1 Software design constraints

- Standards for design,coding,naming,etc.
- Software interfaces
- Use a specific application package
- Constraints on program size,data size etc.

3.4.2 Hardware design constraints

- Specific type of hardware ,reliability requirements
- Hardware interfaces
- Requirements for spare capacity or spare performance

3.4.3 User interface design constraints

- Features of operator/user with details of working environment
- Any special features required

3.5 Other requirements

- Security
- Safety
- Environmental
- Reusability
- Training