

# **VIRTUAL MEDICO**

## **A PROJECT REPORT**

**submitted by**

**MOHAMED RASHID**

**ICE21MCA-2022**

**to**

the APJ Abdul Kalam Technological University  
in partial fullfilment of the requirements for the award of the degree

**of**

Master of Computer Applications



**Department of Computer Applications**  
Ilahia College of Engineering and Technology  
Muvattupuzha-695016

MAY 2023

# **DECLARATION**

I undersigned hereby declare that the project report titled "**VIRTUAL MEDICO**" submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of **Asst.Prof. Sulfath PM**. This submission represents my ideas in my words and where ideas or words of others have been included. I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity as directed in the ethics policy of the college and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and/or University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title.

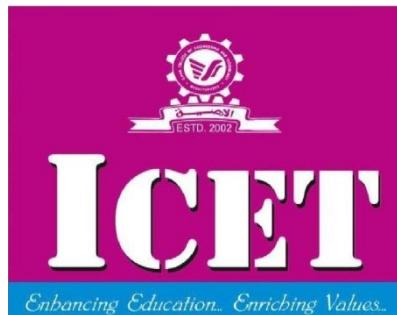
Place : Mulavoor

**MOHAMED RASHID**

Date :

**ILAHIA COLLEGE OF ENGINEERING AND TECHNOLOGY**  
**MULAVOOR P.O MUVATTUPUZHA**

**DEPARTMENT OF COMPUTER APPLICATIONS**



**CERTIFICATE**

This is to certify that the report entitled **VIRTUAL MEDICO** submitted by **MOHAMED RASHID (ICE21MCA-2022)** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications is a bonafide record of the project work carried out by him under my guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

**Project Supervisor**

Prof.Sulfath PM

**Head of the Department**

Prof. Anoop R

**Project Coordinator**

Prof. Sheena KM

Submitted for the viva-voce held on \_\_\_\_\_ at ICET, Muvattupuzha.

**External Examiner**

# **ACKNOWLEDGEMENT**

First and for most I thank **GOD** almighty and my parents for the success of this project. I owe a sincere gratitude and heart full thanks to everyone who shared their precious time and knowledge for the successful completion of my project.

I am extremely thankful to **Prof. Dr. KA Navas**, Principal, Ilahia College of Engineering and Technology for providing me with the best facilities and atmosphere which was necessary for the successful completion of this project.

I am extremely grateful to **Prof. Anoop R**, HOD, Dept of Computer Applications, for providing me with best facilities and atmosphere for the creative work guidance and encouragement.

I express my sincere thanks to **Asst. Prof. Sheena KM**, Department of Computer Applications, Ilahia College of Engineering and Technology for his valuable coordination and advice that aided in the successful completion of my project.

I express my sincere thanks to **Asst. Prof. Sulfath PM**, Department of Computer Applications, Ilahia College of Engineering and Technology for his valuable guidance, and advice that aided in the successful completion of my project.

I profusely thank other Asst. Professors in the department and all other staffs of ICET, for their guidance and inspirations throughout my course of study.

I owe my thanks to my friends and all others who have directly or indirectly helped me in the successful completion of this project. No words can express my humble gratitude to my beloved parents and relatives who have been guiding me in all walks of my journey.

**MOHAMED RASHID**

## **ABSTRACT**

‘Virtual Medico’ is a Website for medical healthcare system . Website provide services like taking online appointment of doctors, online pharmacy, prediction of lung cancer using machine learning and managing staff details in a hospital. Hospitals can make use of this website to add information of their facilities and doctor’s. Users can search for Hospitals and take appointment of doctors and avail services like searching medicines online. Website also provides users option to provide feedback and for registering complaints. Website Admin can add details of Hospitals and approve registrations of pharmacies. The best part of my website is the feature which helps prediction of lung cancer which has been implemented with the help of machine learning techniques. Lung cancer is one of the most common diseases for human beings everywhere throughout the world. Early identification of this disease is the main conceivable approach to enhance the possibility of patients’ survival. In this website , K-Nearest-Neighbors algorithm has been used, for lung cancer prediction. Technology has been a saviour for the medical profession, not just in innovations to improve medication but in other departments too. The management department, for example, now has an automated system where patients can be reminded of their appointments. If you’re a healthcare provider and are just hearing this, stay put and say goodbye to all the losses you’ve incurred from forgotten appointments. It has now gone beyond self-scheduled appointments to reminding patients of their appointments. Don’t be left behind. Reminder systems have proved to improve patient turn up as well as follow up on well-being after treatment. The doctors are, therefore, able to diligently attend to their patients. Here’s what you need to know about automated patient reminders. The system also helps patients to take medicines at time. ‘Virtual Medico’ has been designed considering these trends in the field of health care.

# Contents

|   |      |
|---|------|
| <b>1 INTRODUCTION</b>   | viii |
| <b>2 PROBLEM DEFINITION AND MOTIVATION</b>  | 2    |
| 2.1 Existing System . . . . .   | 2    |
| 2.1.1 Limitations Of Existing System . . . . .  | 2    |
| 2.2 Proposed System . . . . .   | 3    |
| 2.2.1 Advantages of Proposed System . . . . .   | 3    |
| <b>3 LITERATURE REVIEW</b>  | 5    |
| 3.1 Lung Cancer Prediction using Extended KNN Algorithm IEEE 13 April 2022 . . .                                  | 5    |
| 3.2 Lung Cancer Prediction using Machine Learning: A Comprehensive Approach IEEE 3 April 2021 . . . . .           | 5    |
| 3.3 Managing a virtual hospital IEEE paper 06 August 2022 . . . . .   | 6    |
| 3.4 Chen et al. (2022) presented a fuzzy system using kNN (FkNN) for Parkinson's disease (PD) diagnosis . . . . . | 6    |
| <b>4 SYSTEM ANALYSIS</b>  | 7    |
| 4.1 Introduction . . . . .  | 7    |
| 4.2 Module Description . . . . .  | 7    |
| 4.2.1 Product Functions . . . . .   | 10   |
| 4.3 Feasibility Study . . . . .   | 10   |
| 4.3.1 Economic Feasibility . . . . .  | 10   |
| 4.3.2 Technical Feasibility . . . . .   | 11   |
| 4.3.3 Operational Feasibility . . . . .   | 11   |

|  |           |
|--|-----------|
| <b>5 SYSTEM REQUIREMENTS</b>                     | <b>12</b> |
| 5.1 Hardware Requirements . . . . .              | 12        |
| 5.2 Software Requirements . . . . .              | 12        |
| 5.3 Software . . . . .                           | 13        |
| 5.3.1 PyCharm . . . . .                          | 13        |
| 5.3.2 Flask . . . . .                            | 14        |
| 5.3.3 Backend MYSQL . . . . .                    | 14        |
| 5.3.4 Frontend Python . . . . .                  | 16        |
| 5.4 Functional Requirements . . . . .            | 16        |
| 5.5 Non Functional Requirements . . . . .        | 17        |
| 5.5.1 Performance Requirements . . . . .         | 17        |
| 5.5.2 Quality Requirements . . . . .             | 18        |
| <b>6 SYSTEM DESIGN</b>                           | <b>19</b> |
| 6.1 Introduction To System Design . . . . .      | 19        |
| 6.1.1 Logical design . . . . .                   | 20        |
| 6.1.2 Physical design . . . . .                  | 20        |
| 6.2 Methodology . . . . .                        | 20        |
| 6.2.1 Algorithm For Creating The Model . . . . . | 21        |
| 6.3 Input Desing . . . . .                       | 22        |
| 6.3.1 Easy Data Input . . . . .                  | 22        |
| 6.3.2 Data Validation . . . . .                  | 22        |
| 6.3.3 User Friendliness . . . . .                | 23        |
| 6.3.4 Consistent Format . . . . .                | 23        |
| 6.3.5 Interactive Dialogue . . . . .             | 23        |
| 6.4 Architecture Diagram . . . . .               | 23        |
| 6.5 Data Flow Diagrams . . . . .                 | 24        |
| 6.6 Database Design . . . . .                    | 29        |
| <b>7 CODING</b>                                  | <b>35</b> |
| 7.1 Introduction To Coding . . . . .             | 35        |
| 7.1.1 Goals of Coding . . . . .                  | 35        |
| 7.1.2 Coding Standards . . . . .                 | 36        |

|           |  |           |
|-----------|--|-----------|
| 7.1.3     | Coding Guidelines . . . . .                        | 36        |
| <b>8</b>  | <b>SYSTEM TESTING</b>                              | <b>38</b> |
| 8.1       | Testing and Testing Methods . . . . .              | 38        |
| 8.1.1     | Unit Testing . . . . .                             | 39        |
| 8.1.2     | Integration Testing . . . . .                      | 40        |
| 8.1.3     | Validation Testing . . . . .                       | 41        |
| 8.1.4     | Output Testing . . . . .                           | 41        |
| 8.1.5     | User Acceptance Testing . . . . .                  | 41        |
| 8.1.6     | System Testing . . . . .                           | 42        |
| <b>9</b>  | <b>SYSTEM IMPLEMENTATION</b>                       | <b>43</b> |
| 9.1       | Introduction To System Implementation . . . . .    | 43        |
| 9.2       | The common approaches for implementation . . . . . | 44        |
| 9.2.1     | Parallel Conversion . . . . .                      | 44        |
| 9.2.2     | Direct Conversion . . . . .                        | 44        |
| 9.2.3     | Pilot Conversion . . . . .                         | 44        |
| 9.2.4     | User Training . . . . .                            | 44        |
| 9.2.5     | Coding . . . . .                                   | 45        |
| 9.2.6     | Code Plans . . . . .                               | 46        |
| <b>10</b> | <b>SYSTEM MAINTENANCE</b>                          | <b>47</b> |
| 10.1      | Introduction To System Maintenance . . . . .       | 47        |
| 10.1.1    | Corrective Maintenance . . . . .                   | 47        |
| 10.1.2    | Adaptive Maintenance . . . . .                     | 48        |
| 10.1.3    | Perfective Maintenance . . . . .                   | 48        |
| 10.1.4    | Preventive Maintenance . . . . .                   | 48        |
| <b>11</b> | <b>RESULT AND DISCUSSION</b>                       | <b>49</b> |
| 11.1      | Advantages and Limitations . . . . .               | 49        |
| 11.1.1    | Advantages . . . . .                               | 49        |
| 11.1.2    | Limitations . . . . .                              | 50        |
| <b>12</b> | <b>CONCLUSION</b>                                  | <b>51</b> |

|                                    |           |
|------------------------------------|-----------|
| <b>13 FUTURE SCOPE</b>             | <b>52</b> |
| <b>14 APPENDIX</b>                 | <b>53</b> |
| 14.1 Sample Source Code . . . . .  | 53        |
| 14.2 Sample screenshoots . . . . . | 61        |

# List of Figures

|      |                                |    |
|------|--------------------------------|----|
| 6.1  | Architecture Diagram . . . . . | 24 |
| 6.2  | LEVEL 0 . . . . .              | 24 |
| 6.3  | LEVEL 1 : ADMIN . . . . .      | 25 |
| 6.4  | LEVEL 2 : USER . . . . .       | 26 |
| 6.5  | LEVEL 3 : HOSPITAL . . . . .   | 27 |
| 6.6  | LEVEL 4: DOCTOR . . . . .      | 28 |
| 6.7  | LEVEL 5: PHARMACY . . . . .    | 28 |
| 6.8  | LOGIN . . . . .                | 29 |
| 6.9  | HOSPITAL . . . . .             | 29 |
| 6.10 | NOTIFICATION . . . . .         | 30 |
| 6.11 | PHARMACY . . . . .             | 30 |
| 6.12 | FEED BACK . . . . .            | 30 |
| 6.13 | COMPLAINT . . . . .            | 31 |
| 6.14 | DOCTOR . . . . .               | 31 |
| 6.15 | FACILITY . . . . .             | 32 |
| 6.16 | DEPARTMENT . . . . .           | 32 |
| 6.17 | BOOKING . . . . .              | 32 |
| 6.18 | USER REGISTER . . . . .        | 33 |
| 6.19 | MEDICINE . . . . .             | 33 |
| 6.20 | PREDICTION . . . . .           | 34 |
| 14.1 | LOGIN PAGE . . . . .           | 61 |
| 14.2 | ADMIN HOME . . . . .           | 61 |
| 14.3 | USER HOME . . . . .            | 62 |
| 14.4 | USER REGISTRATION . . . . .    | 62 |

|                                       |    |
|---------------------------------------|----|
| 14.5 HOSPITAL HOME . . . . .          | 63 |
| 14.6 DOCTOR HOME . . . . .            | 63 |
| 14.7 USER BOOKIN PAGE . . . . .       | 64 |
| 14.8 LUNG CANCER PREDICTION . . . . . | 64 |
| 14.9 PHARMACY HOME . . . . .          | 65 |
| 14.10PHARMACY REGISTRATION . . . . .  | 65 |

# List of Tables

|     |   |    |
|-----|---|----|
| 8.1 | Unit test cases and results . . . . .   | 39 |
| 8.2 | Integration cases and result . . . . .  | 40 |
| 8.3 | System test cases and results . . . . . | 42 |

# Chapter 1

## INTRODUCTION

‘Virtual Medico’ is a website for medical healthcare system. This webpage is an aid for Hospitals, Pharmacies and General Public. The best part of ‘Virtual Medico’ is the feature which helps prediction of lung cancer which has been implemented with the help of machine learning techniques. At ‘Virtual Medico’ you can take doctor appointment, check packages and facilities in hospitals, search medicine availability, online doctor consultation, and many more. The main features in this website include:

### **Admin Module**

Admin user is there in this site. Admin has the features to add hospital information, verify pharmacy registration, view feedbacks and manage complaints.

### **Hospital Module**

Hospitals can add packages and facilities available in that hospital. They can add Doctor information

### **Online Pharmacy**

Pharmacies can register in this site and can update information about medicines available

### **User Module**

People can register in this site and make use of the features provided by ‘Virtual Medico’. Users can view hospital and doctor information, make appointments online, search for medicine availability in nearby pharmacies and make use of **Lung Cancer Prediction** feature which has been configured with the help of machine learning techniques.

Lung cancer is one of the most common diseases for human beings everywhere throughout the world. Early identification of this disease is the main conceivable approach to enhance the possibility of patients' survival. In this website , a k-Nearest-Neighbors technique, for lung cancer prediction which a genetic algorithm is applied for the efficient feature selection to reduce the dataset dimensions and enhance the classifier pace, is employed for diagnosing the stage of patients' disease. The application provides a reminder module for both patients and doctors and user can also view nearest doctor and take booking. Technology has been a savior for the medical profession, not just in innovations to improve medication but in other departments too. The management department, for example, now has an automated system where patients can be reminded of their appointments. If you're a healthcare provider and are just hearing this, stay put and say goodbye to all the losses you've incurred from forgotten appointments. It has now gone beyond self-scheduled appointments to reminding patients of their appointments. The suggested techniques provide a noble quality tool to predict lung tumor classification and play a major role, particularly in the finding and classification of medical data. The literature reports a number of lung cancer diagnosis systems which predict normal and abnormal lung cancers with the support of SVM.

# **Chapter 2**

## **PROBLEM DEFINITION AND MOTIVATION**

In addition to serving the needs of a broader consumer base, virtual health can provide an opportunity to improve care and healthcare value for chronic disease patients in a way that could also position health systems to succeed in risk-based reimbursement models.

Due to the sevior condition like pandemic disease, majority of the people are refraining to go to hospitals. Even though you cannot physically see a doctor .In that situation Online doctors can work great for treating non emergency conditions such as minor bacterial infections, the cold and flu, and many mental health conditions. They are also useful for managing chronic conditions and getting prescriptions filled and refilled, without having to go to the doctor.

### **2.1 Existing System**

There are a variety of apps available on both ios and Android phones where in you can consult them for various issues. These apps not have the option to cater to your physical health and to consult certified mental health professionals if need .And actually there is no existing system currently as online lung cancer detection.

#### **2.1.1 Limitations Of Existing System**

- For direct consultation of doctors and for searching all medicine required lot of time .
- There is no online prescription.

- No possible for exact booking with correct time by online.
- There is no lung cancer prediction .

## 2.2 Proposed System

Proposed system consist only website. Here we have Online doctor work great for treating non emergency conditions such as minor bacterial infections, the cold and flu, and many mental health conditions. They are also useful for managing chronic conditions and getting prescriptions filled and refilled, without having to go to the doctor. And if there is a emergency condition we can book the doctor for specific time and we find the nearest medical shop that the medicine is available and we can predict lung cancer which is a common disease we don't want to waste time

Website Features

1. Signup
2. Online Booking
3. Online pharmacy
4. Manage hospital
5. Adding doctor
6. Predict lung cancer
7. Receive prescription from doctor
8. Feedback
9. Complaint registration

### 2.2.1 Advantages of Proposed System

- Secure login for admin, user, doctor, pharmacy and hospital.
- Registration for admin, pharmacy and user.
- Hospital add the doctors and facilities of hospital.
- Admin can approve pharmacy, add notification, view feed back.
- The admin can see all the list of the user and doctors and pharmacy.

- Here provides an option to modify the information about the Hospital,pharmacy and user.
- The user can book for their specialised doctors and view all facilities of hospital.
- Patients can receive online prescription.
- User can see near by pharmacy.
- User can search medicine availability.
- Use can add feed back.
- It is time consuming, faster access and user friendly.
- Easy to update the details and view details.
- To provide a facility for the user to change password

# **Chapter 3**

## **LITERATURE REVIEW**

### **3.1 Lung Cancer Prediction using Extended KNN Algorithm IEEE 13 April 2022**

Among several different types of cancer the one that causes high mortality in every country is lung carcinoma. The possibility of survival from this deadly disease can be enhanced by identifying cancer at an early stage. This paper focuses on an Extended version of the KNN Algorithm that is used for the prediction of lung carcinoma based on the Computed Tomography (CT) - Images given as the input. The 2-D image undergoes a Modified Gabor Filtration technique wherein the images are used to extract the features for Edge Detection. This further undergoes Feature Extraction followed by Binarization which is fed as Production data to the Machine Learning model. Based on the Extended KNN Algorithm, the model evaluates the testing data and corresponding predictions are made. The model predicts the Cancer Stage based on the input CT - Image which is passed to the doctor for further medication.

### **3.2 Lung Cancer Prediction using Machine Learning: A Comprehensive Approach IEEE 3 April 2021**

prominent cause of cancer-related mortality throughout the globe is “Lung Cancer”. Hence beforehand detection, prediction and diagnosis of lung cancer has become essential as it expedites

and simplifies the consequent clinical board. To erect the progress and medication of cancerous conditions machine learning techniques have been utilized because of its accurate outcomes. Various types of machine learning algorithms(ML) like Naive Bayes, Support Vector Machine (SVM), Logistic regression, Artificial Neural Network (ANN), have been applied in the healthcare sector for analysis and prognosis of lung cancer. In this review, factors that cause lung cancer and application of ML algorithms are discussed up to date and also draws special attention to their relative strengths and weaknesses. This paper will help the researchers to quickly go through the related literature instead of referring to the many papers.

### **3.3 Managing a virtual hospital IEEE paper 06 August 2022**

The paper discusses the concept of the virtual hospital and telemedicine as a part of daily hospital operations. The concept of the virtual hospital is analyzed by using business, application and technology point of views. In addition to the hospital oriented operations the operational level activities, i.e. mobile hospital, independent living, care of wellness and treatment chain, form an essential part of it. A common feature of all of these operations is the central role of telecommunication to implement the operative systems. The virtual hospital may not only be seen as a collection of new technical tools at the hospital; above all there is a question on the adoption of new operations model to achieve effectivity and better quality from both the hospital and the patient point of view

### **3.4 Chen et al. (2022) presented a fuzzy system using kNN (FkNN) for Parkinson's disease (PD) diagnosis**

Machine learning involves several algorithms such as k-Nearest Neighbors (kNN), support vector machine (SVM), Naive Bayes (NBs), classification tree (C4.5), gradient boosting machines (GBM), etc. While each of these algorithms processes data differently, in this section, a few recently proposed machine learning candidates in the area of malignant growth finding are reviewed chronologically. The proposed methodology is an enhancement of the kNN method provides a background for the kNN method.

# **Chapter 4**

## **SYSTEM ANALYSIS**

### **4.1 Introduction**

System analysis is performed by many parties of the system development . Overall, the system analyst is the person that controls activities in this phase. This person must have a strong technical and analytical background. The system analyst has the task of investigating the current business processes and determines the requirements of the new system. So this person needs to have some business acumen and "CSI" investigation skills in addition, they should be able to understand the capabilities of technology. The system analyst needs to have strong communication skills to run these interviews.

SDLC, Software Development Life Cycle is a process used by software industry to design, develop and test high quality software. The SDLC aims to produce high quality software that meets or exceeds customer expectations, reaches completion within times and cost estimates. SDLC is the acronym of Software Development Life Cycle. It is also called as Software development process.

### **4.2 Module Description**

This project contain five modules following are the five modules .

#### **Admin**

- login
- Add hospital
- Manage hospital

- Add notification
- Approve pharmacy
- View feed back
- Manage complaint

## Hospital

- Login
- Add doctors
- Manage doctors
- Add staff
- Manage staff
- Add facility
- Mange facility
- Add department
- Manage department
- Online booking

### **Doctor**

- Login
- View profile
- View patients
- View booking
- Add complaint
- Add feedback

### **User**

- Register
- Login
- View doctor
- Add booking
- View nearest pharmacy
- Purchase medicine
- View department and facility
- Add complaint
- View reply
- Add feedback
- Predict lung cancer

### **Pharmacy**

- Register
- Login

- View profile
- Add medicine details
- Add feedback

#### 4.2.1 Product Functions

- User can take prescription from a doctor and book specialized doctors.
- Train model using knn for lung cancer prediction.
- User can see near by pharmacy and know medicine availability.
- Developing a user interface.
- Admin,hospital,doctor,user and pharmacy can login.

### 4.3 Feasibility Study

The feasibility study is not warrantied system in with economic justification is obvious technical risk is low, no legal problems are expected and no reasonable alternative exists. Three essential aspects are involved in the feasibility study promotions of preliminary investigation Technical, Economic, and Operational feasibility.

- Economic Feasibility
- Technical Feasibility
- Operational Feasibility

#### 4.3.1 Economic Feasibility

Economic analysis is the most frequently used method for evaluating the effectiveness of a candidate system. More commonly known as benefit analysis, the procedure is to determine the benefit and saving that are expected from a candidate system and compare them with the term of time by automating the process or report generation. Here all the processes being done without including additional cost to employ other resources than available with the existing internet

infrastructure. There is no need for the initial expense with this project. Hence it can be said that, this package is economically feasible.

#### **4.3.2 Technical Feasibility**

The technical feasibility centers on the existing system hardware, software, etc. and to what intent if supports the proposed system. Since the “VIRTUAL MEDICO” system have all the require software such as python, pycharm, Microsoft SQL Server. The hardware such as core I3 class processor with 80GB Hard disk, Hence it is technically feasible.

#### **4.3.3 Operational Feasibility**

An estimate should be made to know how strong the reaction of user is likely to have towards the new system. Since this system ready to use in the World Wide Web, So the system is operationally feasible. As this package is technically, economically, and functionally feasible, the system is judged feasible. Viewing the collected information, recommendation and justification, conclusion is made of the proposed system.

# **Chapter 5**

## **SYSTEM REQUIREMENTS**

### **5.1 Hardware Requirements**

- Processor : Intel Core i3 and above.
- Storage : 512 GB Hard Disk space
- Memory : 4 GB RAM

### **5.2 Software Requirements**

- Operating System : windows 7 and above
- Front end: Python
- Back end: SQL server
- IDE : PyCharm
- Web Browser : Internet Explorer/Google Chrome/Firefox
- Frame Work : Flask
- Libraries used : Opencv, Mediapipe, numpy, Fastapi, Sqlalchemy etc

## 5.3 Software

### 5.3.1 PyCharm

PyCharm is an Integrated Development Environment (IDE) used for computer programming, specifically for the Python language. Python is an interpreter, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy-to-learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed. Python is a multi-paradigm programming language. Object-oriented programming and structured programming are fully supported, and many of its features support functional programming and aspect-oriented programming (including by meta-programming and meta-objects (magic methods)). Many other paradigms are supported via extensions, including design by contract and logic programming. Python uses dynamic typing, and a combination of reference counting and a cycle-detecting garbage collector for memory management. It also features dynamic name resolution (late binding), which binds method and variable names during program execution.

#### Features Of PyCharm

- Coding assistance and analysis, with code completion, syntax and error highlighting, linter integration, and quick fixes.
- Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages.
- Python refactoring: including rename, extract method, introduce variable, introduce constant, pull up, push down and others.
- Support for web frameworks: Django, web2py and Flask.
- Integrated Python debugger.

- Integrated unit testing with line-by-line code coverage.
- Google App Engine Python development.

### 5.3.2 Flask

Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Extensions are updated far more regularly than the core Flask program.

#### Major Features Of Flask

- Contains development server and debugger.
- High flexibility.
- Integrated support for unit testing.
- It is super easy to deploy Flask in production (Flask is 100 WSGI 1.0 compliant”).
- Support for secure cookies (client side sessions).
- Extensive documentation.
- Extensions available to enhance features desired.

### 5.3.3 Backend MYSQL

SQL SERVER 2008 SQL Server is a relational database management system that's capable of handling large amounts of data and many concurrent uses while preserving data integrity and providing many advanced administrations and data distribution capabilities. The SQL Server components act as a gateway between the clients and the physical data. The easiest and fastest way to enter data in to a table is by using SQL Server Management studio. Microsoft SQL server 2008 is fully featured relational database management system (RDBMS) that offers a variety of

administrative tools to ease the burdens of database development maintenance and administration. It consists of five more frequently used tools: Enterprise Manager, Query Analyzer, SQL Profiler, Service Manager and Data Transformation Service.

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database, it is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). There are at least a dozen different editions of Microsoft SQL Server aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users. Its primary query languages are T-SQL and ANSI SQL.

Microsoft SQL Server is a widely accepted standard database sublanguage used in querying, updating and managing relational databases. SQL Server 2000 has features that will import, transform and export data from multiple sources, from Oracle to text files. It can also act as the backbone to Business Intelligence infrastructure, as an XML processing tool, or even to process natural language queries. It has the ability to be "clustered" so that it can automatically fail over to another system in case of a catastrophe.

It can also serve as a reporting server front-end for users. The Code Editor component of SQL Server Management Studio contains integrated script editors for authoring Transact-SQL, MDX, DMX, and XML/A. The Object Explorer component of SQL Server Management Studio is an integrated tool for viewing and managing objects in all server types. SQL Server manager includes features such as Supports most administrative tasks for SQL Server, A single, integrated environment for SQL Server Database Engine management and authoring, Dialogs for managing objects in the SQL Server Database Engine, Analysis Services, and Reporting Services, that allows you to execute actions immediately, send them to a Code Editor, or script them for later execution, Non-modal and resizable dialogs allow access to multiple tools while a dialog is open, A common scheduling dialog that allows you to perform action of the management dialogs at a later time.

#### **It includes the following features:**

- Internet Integration.
- Scalability and availability.
- Enterprise-level database features.

- Ease of installation , deployment and use of warehousing .
- Query Analyzer and Sql profiler.
- Data definition and data manipulation languages and database keys.

### 5.3.4 Frontend Python

Python is developed by Guido Van Rossum. Guido van Rossum started implementing python in 1989 . python is a very simple programming language so even if you are new to programming , you can learn python with out facing any issues.

#### Features Of Python Programming Language

- **Readable:** Python is very readable language.
- **Easy To Learn:** Learning python easy as this is a expressive and high level programming language , which means it is easy to understand the language and thus easy to learn.
- **Cross Platform:** Python is available and can run various operating system such as Mac,windows,Linux,Unix etc. this makes it a cross platform and portable language.
- **Open Source:** Python is a open source programming language.
- **Large Standard Library:** Python comes with a large standard library that has some handy codes and functions which we can use while writing code in python .
- **Free:** Python is free to download and use .This means you can downloaded it for free and use it in your application. Python is a example of a FLOSS , which means you can freely distribute copies of this software , read its source code and modify it.
- **Advanced Features:** Supports generators and list comprehension. We will cover this features later .

## 5.4 Functional Requirements

Functional requirements define what a system is supposed to do. The system should perform the following functionalities

- Secure login for admin, user, doctor, pharmacy and Hospital.
- Registration for admin, pharmacy and user.
- Hospital add the doctors and facilities of hospital.
- Admin can approve pharmacy, add notification, view feed back.
- The admin can see all the list of the user and doctors and pharmacy.
- Here provides an option to modify the information about the Hospital, pharmacy and user.
- The user can book for their specialised doctors and view all facilities of hospital.
- User can receive online prescription.
- User can see near by pharmacy.
- User can search medicine availability.
- User can add feed back.
- It is time consuming, faster access and user friendly.
- Easy to update the details and view details.
- To provide a facility for the user to change password

## 5.5 Non Functional Requirements

### 5.5.1 Performance Requirements

- Accuracy : The system should keep its accuracy in operation and its user-friendly character.
- Speed : The system should be able to provide speed.
- Low cost: This method is both inexpensive to implement and user-friendly.
- Less Time consuming: In comparison to the existing system, it takes much less time.
- User Friendly: This proposed system is extremely user-friendly, and it allows for the creation of a pleasant environment.

### 5.5.2 Quality Requirements

- Scalability : All of the functional requirements will be met by the product.
- Maintainability : The system must be easy to maintain. It should retain backups in case of system failures and report its activity on a regular basis.
- Reliability : The maximum amount of downtime that is tolerable should be set as high as possible. The mean time between failures, in other words, should be as long as possible. If the system fails, the time it takes to restore the system backup should be minimal.
- Availability: This system is readily available since the software's essential requirements are readily available.
- High- Functionality: Because they are very adaptive, this system is highly functional in any context.

# **Chapter 6**

# **SYSTEM DESIGN**

## **6.1 Introduction To System Design**

The most creative and challenging phase of the system development is the system design. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Design goes through the logical and physical stages of development. In designing a new system, the analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. The first step is to determine how the output is to be produced and in what format. Second input data and master files have to be designed to meet the requirements of the proposed output. The operational phases are handled through program construction and testing. Finally details related to justification of the system and an estimate of the impact of the candidate system on the user and the organization are documented and evaluated by the management.

Design of a system can be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Thus system design is a solution, “how to” approach to the creation of a new system. The design step provides a data design, architectural design and a procedural design. The data design transforms the information domain created during analysis into the data structure that will be required to implement the software. The architectural design defines the relationship among major structural components and procedural description of the software. Source code is generated and testing conducted to integrate and validate the software. System design goes through two phases of development:

- Logical design
- Physical design

### 6.1.1 Logical design

The part of the design process that is independent of any specific hardware or software platform is referred to as logical design. During logical design, all functional features of the system chosen for development in analysis phase are described independently of any computer platform. Logical design concentrates on the business aspects of the system and tends to be oriented to a high level of specificity.

### 6.1.2 Physical design

Physical design is the part of the design phase in which the logical specifications of the system from logical design are transferred into technology-specific details from which all programming and system construction can be accomplished. As a part of the physical design, analysts design the various parts of the system to perform the physical operation necessary to facilitate data capture, processing, and information output.

## 6.2 Methodology

In this project.we use a the creation of the model knn algorithm for lung cancer prediction . K-Nearest Neighbour (KNN) Algorithm is proposed for the detection which is a non parametric method. This optimization algorithm allows physicians to identify the nodules present in the CT lung images in the early stage hence the lung cancer.

- **Feature extraction:** Feature extraction is the most important part of any machine learning task and so is the case with this. Here I specify the cancer of the lungs.
- **Training:** A kNN is used for training. A common type of lung image and image recognition and classification is the done by support vector machine.
- **Testing:** I have tested the produced model with the remaining dataset in the testing phase. The data set is input into the produced model, and the results are saved for the model evaluation and error analysis that follows.

- For classification support vector machine is used the data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector
- **Model evaluation and error analysis:** For the error calculation, the results of the testing data are combined with their original values. For calculating the model's efficiency, a variety of statistical measures can be used. Accuracy, precision, recall, kappa score, and other metrics are used. I proceed with the developed model if the findings of these quantitative assessments are acceptable. If the results are bad, the model should be regenerated with more characteristics in order to achieve the optimal outcome.

The user interface is the second phase of the project. Python are used to create the user interface. This section of the project is concerned with the user. This is how the input text is fed into the server. The returned findings are also shown in the user program. The user interface is designed in such a way that it is simple and easy to understand. In order to deliver a better user experience, I employ responsible HTML designs that include Bootstrap.

### 6.2.1 Algorithm For Creating The Model

- Dataset is split into training and testing datasets in which 80% is used for training and the rest for testing to obtain the better result.
- The model is created using KNN. The images with labelled landmarks will be trained and identified.
- K-Nearest Neighbour is one of the simplest Machine Learning algorithms based on Supervised Learning technique.
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suited category by using K- NN algorithm.
- K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

- K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.
- It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.
- KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.
- The testing dataset is feeded into the created model and their results are noted down.
- The result of testing dataset evaluated using the created model is then compared with the actual values of the testing dataset to evaluate the efficiency of the model.

## 6.3 Input Design

The first step in system design is to design input and output within predefined guidelines. In input design, user originated inputs are converted into computer based format. In output design, the emphasis is on producing the hard copy of the information requested or displaying the output on a screen in a predefined format. The following features have been incorporated into the input design of the proposed system.

### 6.3.1 Easy Data Input

Data entry has been designed in a manner much similar to the source documents. Appropriate messages are provided in the message area, which prompts the user in entering the right data. Erroneous data inputs are checked at the end of each screen entry

### 6.3.2 Data Validation

The input data is validated to minimize errors in data entry. For certain data specific codes have been given and validation is done which enables the user to enter the required data or correct them if they entered wrong codes.

### 6.3.3 User Friendliness

User is never left in a state of confusion as to what is happening, instead appropriate error and acknowledge messages are sent. Error maps are used to indicate the error codes and specific error messages.

### 6.3.4 Consistent Format

A fixed format is adopted for displaying the title messages. Every screen has line, which displays the operation that can be performed after the data entry. They are normally done at the touch of a key.

### 6.3.5 Interactive Dialogue

The system engages the user in an interactive dialogue. The system is able to extract missing or omitted information from the user by directing the user through appropriate messages, which are displayed.

## 6.4 Architecture Diagram

An architecture diagram is a visual representation that shows how components of a software system are physically implemented. It depicts the overall structure of the software system, as well as the relationships, restrictions, and boundaries that exist between each component.

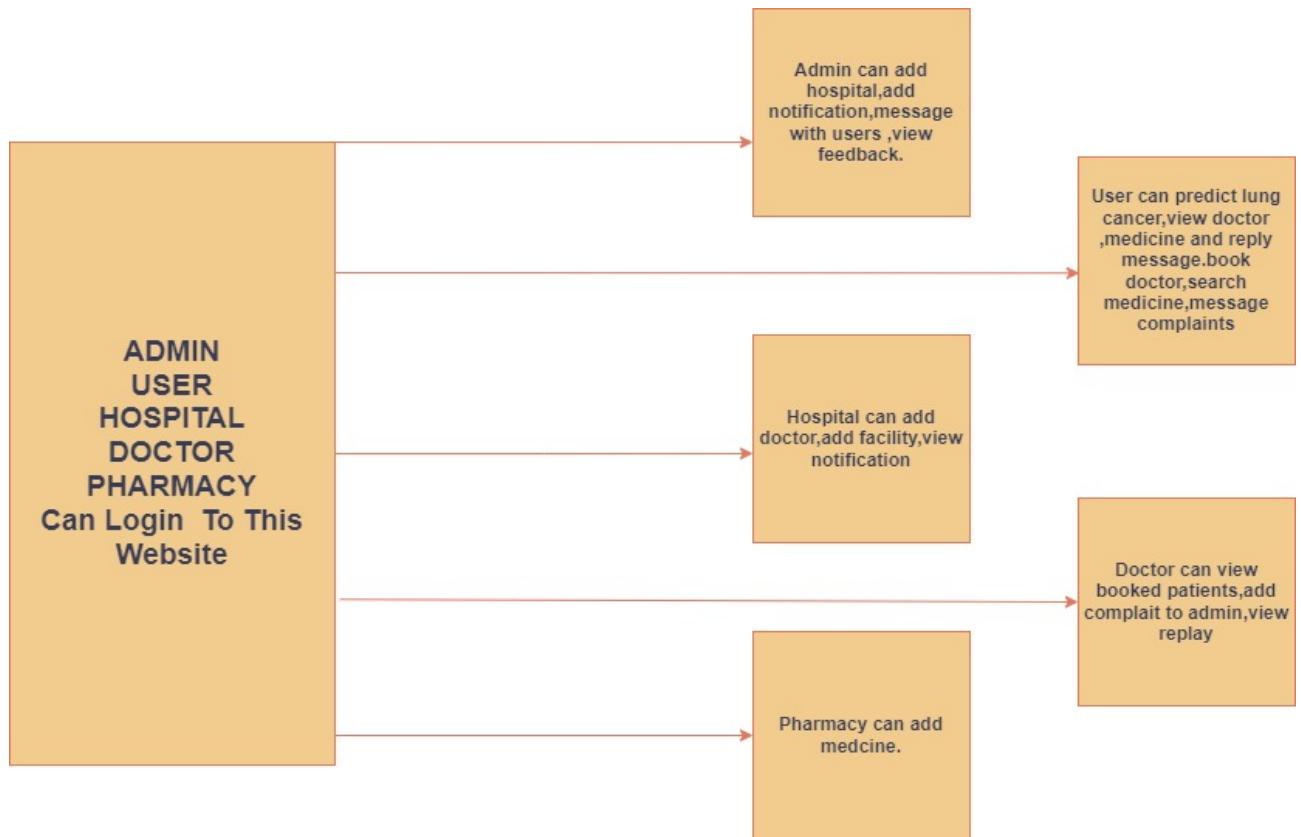


Figure 6.1: Architecture Diagram

## 6.5 Data Flow Diagrams

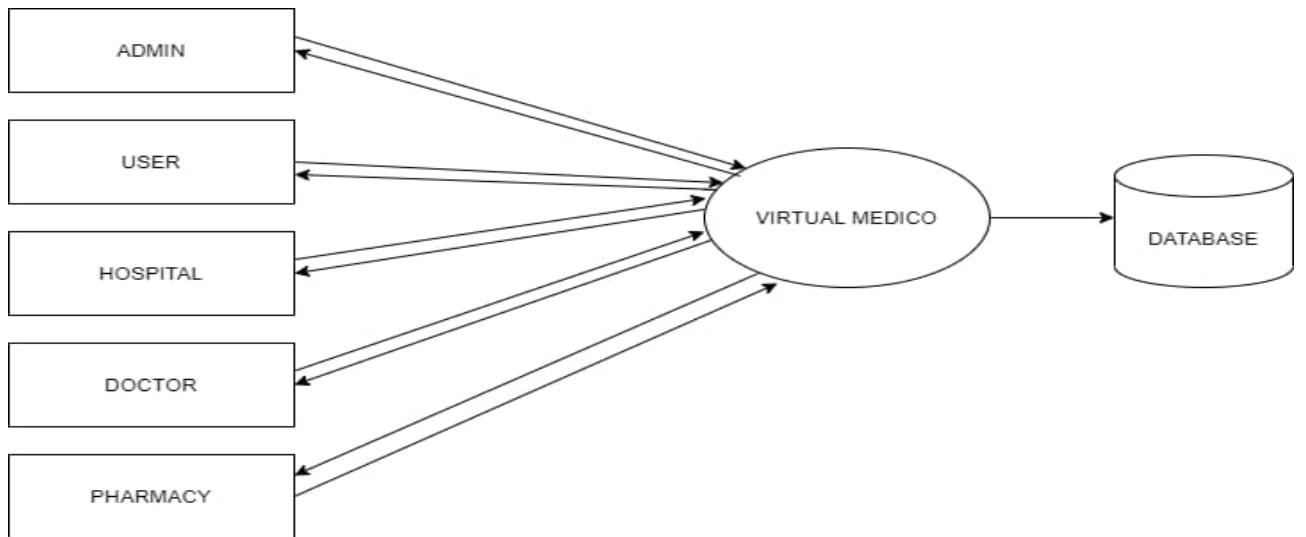


Figure 6.2: LEVEL 0

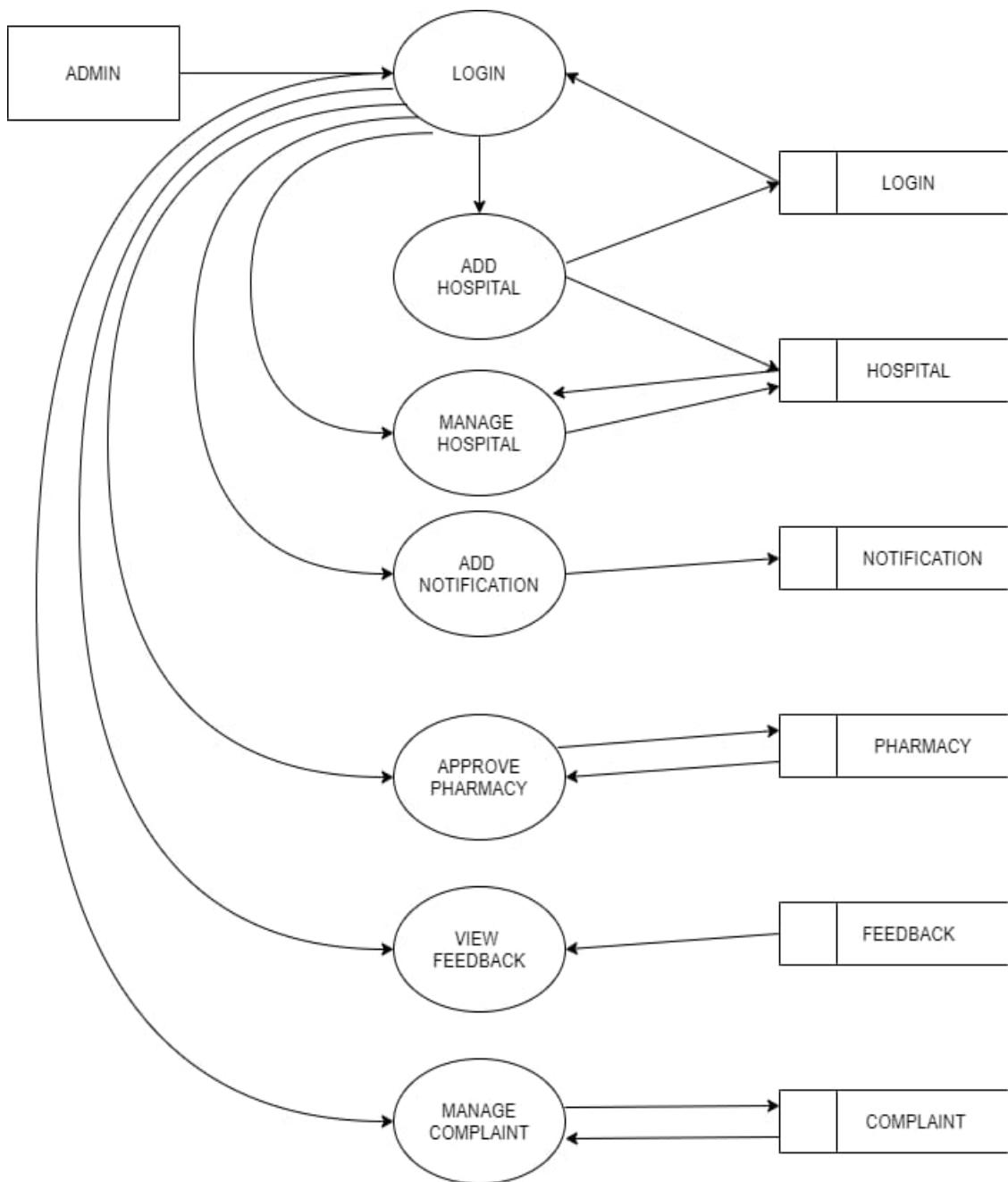


Figure 6.3: LEVEL 1 : ADMIN

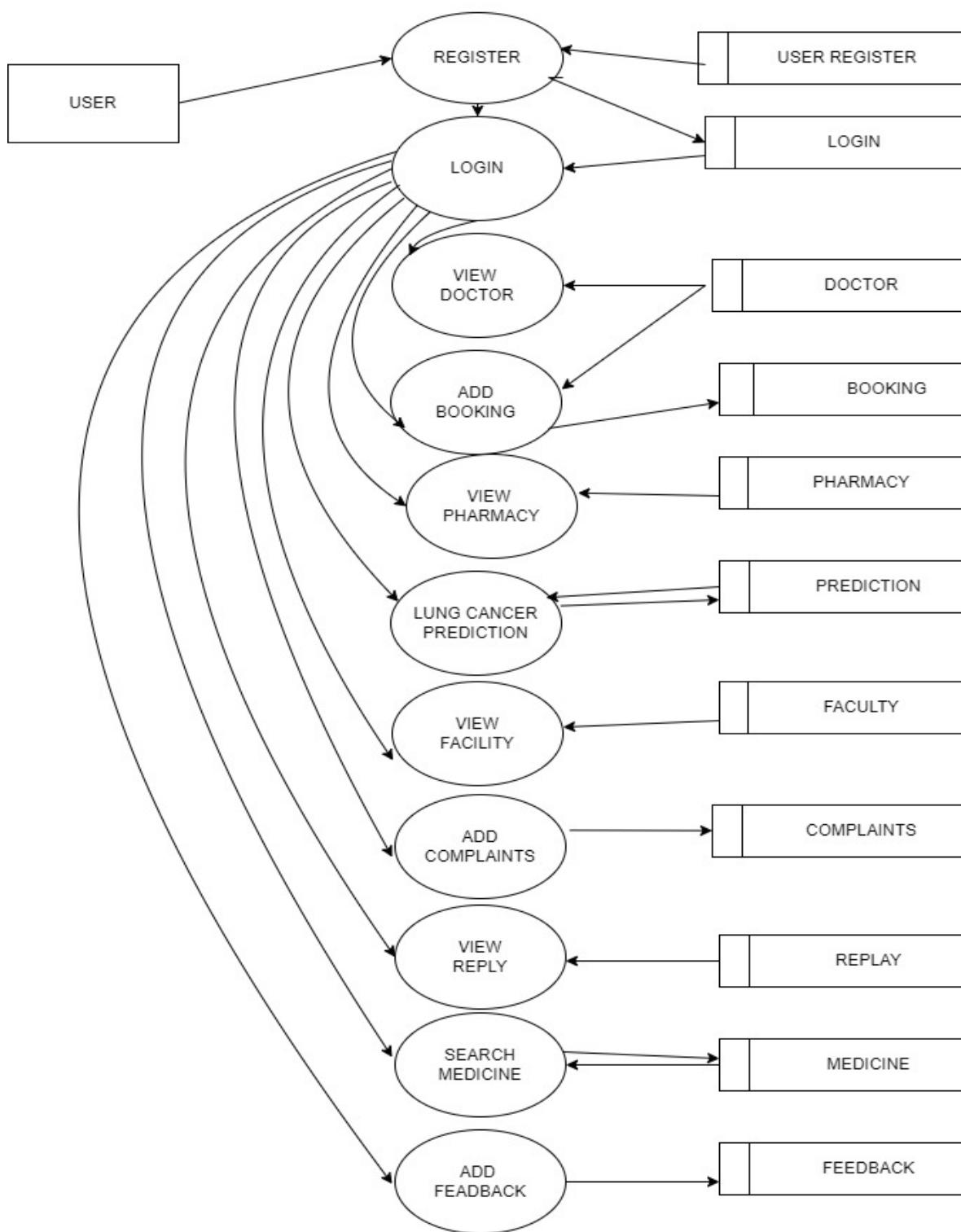


Figure 6.4: LEVEL 2 : USER

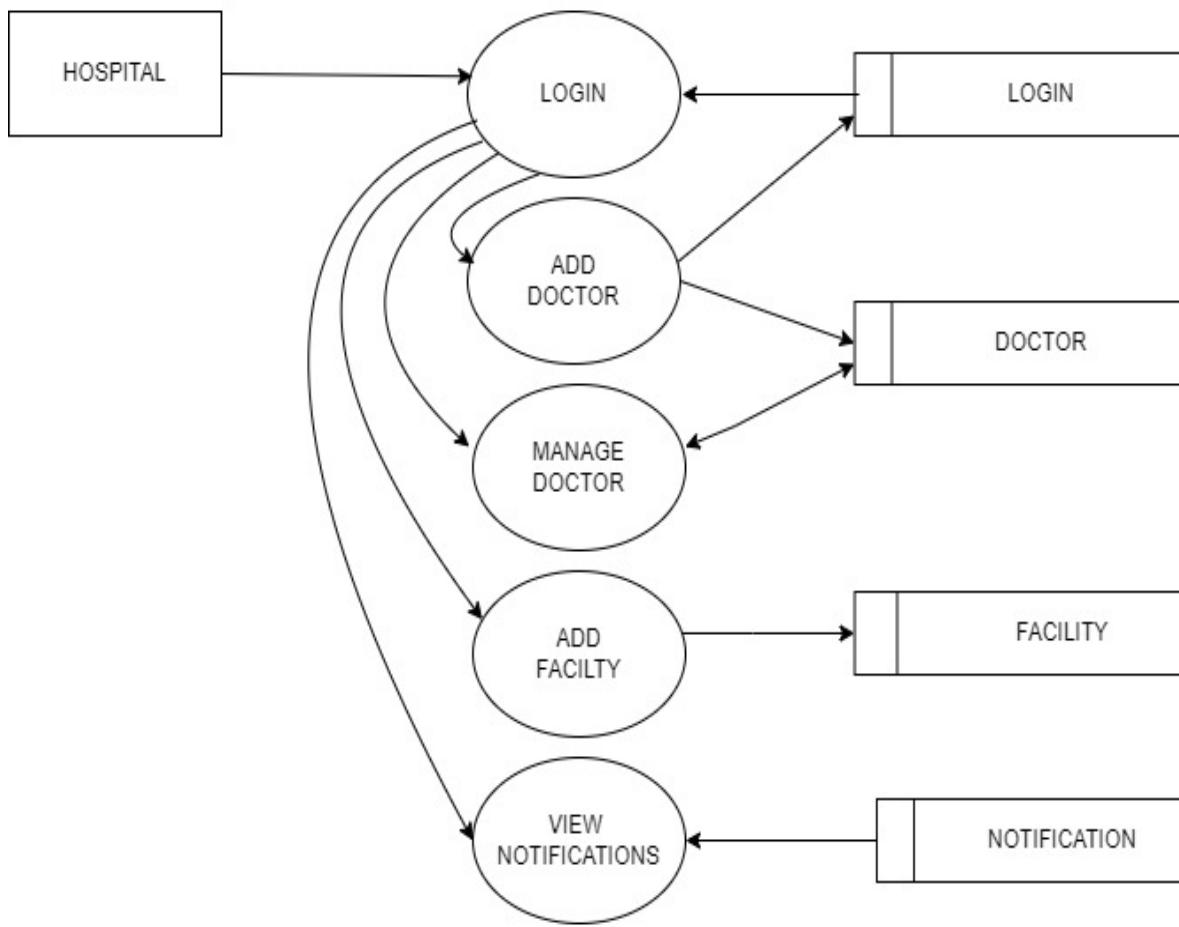


Figure 6.5: LEVEL 3 : HOSPITAL

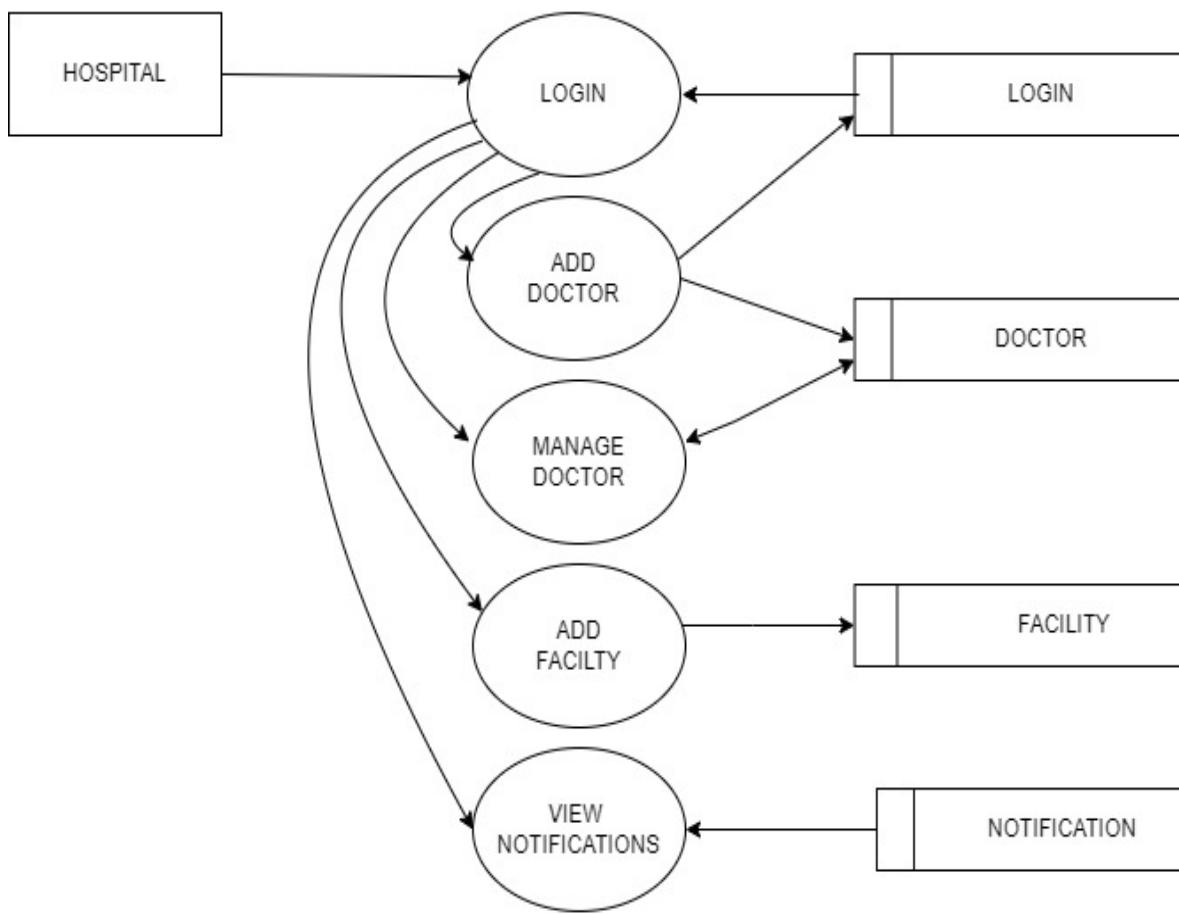


Figure 6.6: LEVEL 4: DOCTOR

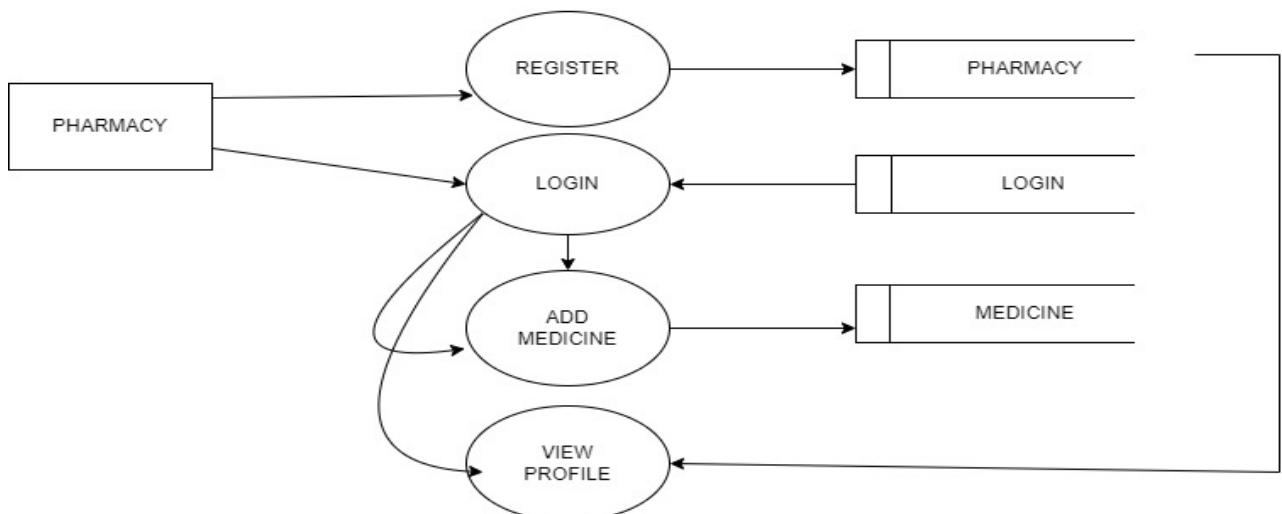


Figure 6.7: LEVEL 5: PHARMACY

## 6.6 Database Design

| <b>Column name</b> | <b>Datatype</b> |
|--------------------|-----------------|
| Login id           | int()           |
| username           | varchar(50)     |
| password           | varchar(50)     |
| User type          | varchar(50)     |

Figure 6.8: LOGIN

| <b>Column name</b> | <b>Datatype</b> |
|--------------------|-----------------|
| Hospital id        | int()           |
| Login id           | bigint()        |
| Name               | varchar(50)     |
| place              | varchar(50)     |
| post               | varchar(50)     |
| pin                | varchar(50)     |
| email              | varchar(50)     |
| phone              | Varchar(50)     |

Figure 6.9: HOSPITAL

| <b>Column name</b> | <b>Datatype</b> |
|--------------------|-----------------|
| Notification id    | int()           |
| Notification       | bigint()        |
| Date               | bigint()        |

Figure 6.10: NOTIFICATION

| <b>Column name</b> | <b>Datatype</b> |
|--------------------|-----------------|
| Pharmacy id        | int()           |
| Login id           | varchar(50)     |
| Name               | varchar(50)     |
| Place              | varchar(50)     |
| Post               | varchar(50)     |
| pin                | varchar(50)     |
| License no         | varchar(50)     |
| email              | varchar(50)     |
| phone              | varchar(50)     |

Figure 6.11: PHARMACY

| <b>Column name</b> | <b>Datatype</b> |
|--------------------|-----------------|
| Feedback id        | varchar(max)    |
| Login id           | double          |
| feedback           | Varchar(50)     |
| date               | Bigint()        |

Figure 6.12: FEED BACK

| <b>Column name</b> | <b>Datatype</b> |
|--------------------|-----------------|
| Complaint id       | Int()           |
| Login id           | varchar(50)     |
| Complaint          | varchar(50)     |
| Date               | varchar(50)     |
| Reply              | varchar(50)     |

Figure 6.13: COMPLAINT

| <b>Column name</b> | <b>Datatype</b> |
|--------------------|-----------------|
| Doctor_id          | int()           |
| Login_id           | varchar(50)     |
| Fname              | varchar(50)     |
| Lname              | varchar(50)     |
| Gender             | Varchar(50)     |
| Dob                | Date()          |
| Qualification      | Varchar(50)     |
| Specilizaton       | Varchar(50)     |
| Place              | Varchar(50)     |
| Post               | Varchar(50)     |
| Pin                | Varchar(50)     |
| Email              | Varchar(50)     |
| Phone              | Varchar(50)     |

Figure 6.14: DOCTOR

| <b>Column name</b> | <b>Datatype</b> |
|--------------------|-----------------|
| Facility_id        | int()           |
| Login_id           | Varchar(50)     |
| Facility           | varchar(50)     |
| Description        | varchar(50)     |

Figure 6.15: FACILITY

| <b>Column name</b> | <b>Datatypes</b> |
|--------------------|------------------|
| Department id      | Int()            |
| Login id           | Varchar(50)      |
| name               | Varchar(50)      |
| Description        | Varchar(50)      |

Figure 6.16: DEPARTMENT

| <b>Column name</b> | <b>Data type</b> |
|--------------------|------------------|
| Booking id         | Int()            |
| Doctor id          | Varchar(50)      |
| User id            | Varchar(50)      |
| Date               | Varchar(50)      |

Figure 6.17: BOOKING

| <b>Column name</b> | <b>Data type</b> |
|--------------------|------------------|
| User id            | Int()            |
| Login id           | Int()            |
| Fname              | Varchar(50)      |
| Lname              | Varchar(50)      |
| Gender             | Varchar(50)      |
| Dob                | Date()           |
| Place              | Varchar(50)      |
| Post               | Varchar(50)      |
| Pin                | Varchar(50)      |
| Phone              | Varchar(50)      |

Figure 6.18: USER REGISTER

| <b>Column name</b> | <b>Data type</b> |
|--------------------|------------------|
| Medicine id        | Int()            |
| Pharmacy id        | Varchar(50)      |
| Medicine Name      |                  |
| Description        | Varchar(50)      |
| Photo              | Varchar(50)      |
| Price              | Varchar(50)      |
| Exp date           | Varchar(50)      |
| Dosage             | Varchar(50)      |

Figure 6.19: MEDICINE

| Column name   | Data type   |
|---------------|-------------|
| Prediction id | Int()       |
| Login id      | Varchar(50) |
| image         | Varchar(50) |
| Date          | Varchar(50) |

Figure 6.20: PREDICTION

# **Chapter 7**

## **CODING**

### **7.1 Introduction To Coding**

The coding is the process of transforming the design of a system into a computer language format. This coding phase of software development is concerned with software translating design specification into the source code. It is necessary to write source code internal documentation so that conformance of the code to its specification can be easily verified.

Coding is done by the coder or programmers who are independent people than the designer. The goal is not to reduce the effort and cost of the coding phase, but to cut to the cost of a later stage. The cost of testing and maintenance can be significantly reduced with efficient coding.

#### **7.1.1 Goals of Coding**

- To translate the design of system into a computer language format: The coding is the process of transforming the design of a system into a computer language format, which can be executed by a computer and that perform tasks as specified by the design of operation during the design phase.
- To reduce the cost of later phases: The cost of testing and maintenance can be significantly reduced with efficient coding. Making the program more readable: Program should be easy to read and understand. It increases code understanding having readability and understandability as a clear objective of the coding activity can itself help in producing more maintainable software.

### 7.1.2 Coding Standards

General coding standards refers to how the developer writes code, so here we will discuss some essential standards regardless of the programming language being used.

**The following are some representative coding standards:**

- Indentation: Proper and consistent indentation is essential in producing easy to read and maintainable programs.
- Indentation should be used to: Emphasize the body of a control structure such as a loop or a select statement. Emphasize the body of a conditional statement Emphasize a new scope block
- Inline comments: Inline comments analyze the functioning of the subroutine, or key aspects of the algorithm shall be frequently used.
- Rules for limiting the use of global: These rules file what types of data can be declared global and what cannot.
- Structured Programming: Structured (or Modular) Programming methods shall be used. "GOTO" statements shall not be used as they lead to "spaghetti" code, which is hard to read and maintain, except as outlined line in the FORTRAN Standards and Guidelines.
- Naming conventions for global variables, local variables, and constant identifiers: A possible naming convention can be that global variable names always begin with a capital letter, local variable names are made of small letters, and constant names are always capital letters.
- Error return conventions and exception handling system: Different functions in a program report the way error conditions are handled should be standard within an organization. For example, different tasks while encountering an error condition should either return a 0 or 1 consistently.

### 7.1.3 Coding Guidelines

General coding guidelines provide the programmer with a set of the best methods which can be used to make programs more comfortable to read and maintain. Most of the examples use

the C language syntax, but the guidelines can be tested to all languages.

**The following are some representative coding guidelines recommended by many software development organizations**

- 
- Line Length: It is considered a good practice to keep the length of source code lines at or below 80 characters. Lines longer than this may not be visible properly on some terminals and tools. Some printers will truncate lines longer than 80 columns.
- Spacing: The appropriate use of spaces within a line of code can improve readability.
- The code should be well-documented: As a rule of thumb, there must be at least one comment line on the average for every three-source line.
- The length of any function should not exceed 10 source lines: A very lengthy function is generally very difficult to understand as it possibly carries out many various functions. For the same reason, lengthy functions are possible to have a disproportionately larger number of bugs.
- Do not use goto statements: Use of goto statements makes a program unstructured and very tough to understand.
- Inline Comments: Inline comments promote readability.
- Error Messages: Error handling is an essential aspect of computer programming. This does not only include adding the necessary logic to test for and handle errors but also involves making error messages meaningful.

# **Chapter 8**

## **SYSTEM TESTING**

### **8.1 Testing and Testing Methods**

Testing is an activity to verify that a correct system is being built and is performed with the intent of finding faults in the system. However not restricted to being performed after the development phase is complete, but this is to carry out in parallel with all stages of system development, starting with requirements specification. Testing results, once gathered and evaluated, provide a qualitative indication of software quality and reliability and serve as a basis for design modification if required. A project is said to be incomplete without proper testing.

System testing is a process of checking whether the developed system is working according to the original objectives and requirements. The system should be tested experimentally with test data so as to ensure that system works according to the required specification. When the system is found working, test it with actual data and check performance.

**The testing procedure that has been used as follows:**

- Unit Testing
- Integration Testing
- Validation Testing
- Output Testing
- User Acceptance Testing
- System Testing

### 8.1.1 Unit Testing

The first level of testing is called as unit testing. Here the different modules are tested and the specification produced during design for the modules. Unit testing is essential for verification of the goal and to test the internal logic of the modules. Unit testing is conducted to different modules of the project. Errors were noted down and corrected down immediately and the program clarity was increased. The testing was carried out during the programming stage itself. In this step each module is found to be working satisfactory as regard to be expected out from the module.

| Sl No | Procedures                           | Expected result                            | Actual result    | Pass or Fail |
|-------|--------------------------------------|--|------------------|--------------|
| 1     | all login and registration           | These should be done                       | Same as expected | Pass         |
| 2     | lung cancer prediction               | should predict whether cancer is yes or no | same as expected | Pass         |
| 3     | view doctor,pharmacy,replay,feedback | showing these                              | Same as expected | Pass         |
| 4     | add complaints ,feedback             | Adding these                               | Same as expected | Pass         |
| 5     | Booking doctors on time shedule      | Is booking working                         | Same as expected | Pass         |

Table 8.1: Unit test cases and results

### 8.1.2 Integration Testing

The second level of testing includes integration testing. It is a systematic testing of constructing structure. At the same time tests are conducted to uncover errors with the interface. It need not to be the case, that software whose modules when run individually showing results will also show perfect results when run as a whole. The individual modules are tested again and the results are verified. The goal is to see if the modules integrated between the modules. This testing activity can be considered as testing the design and emphasizes on testing modules interaction.

| Sl No | Procedures                                  | Expected result          | Actual result    | Pass or Fail |
|-------|---|--------------------------|------------------|--------------|
| 1     | Integration of data collected from the user | like searching ,book-ing | Same as expected | Pass         |
| 2     | Upload the image for lung cancer prediction | Is all are working       | Same as expected | Pass         |

Table 8.2: Integration cases and result

### **8.1.3 Validation Testing**

The next level of testing is validation testing. Here the entire software is tested. The reference document for this process is the requirement and the goal is to see if the software meets its requirements. The requirement document reflects and determines whether the software functions as the user expected. At culmination of integration testing, software is completely assembled as a package and corrected and a final series of software test validation test begins. The proposed system under construction has been tested by using validation testing and found to be working satisfactory. Data validation checking is done to see whether the corresponding entries made in different tables are done correctly. Proper validation checks are done in case of insertion and updating of tables, in order to see that no duplication of data has occurred. If any such case arises proper warning message will be displayed. Double configuration is done before the administrator deletes a data in order to get positive results and to see that o data have been deleted by accident.

### **8.1.4 Output Testing**

The output of the software should be acceptable to the system user. The output of requirement is defined during the system analysis. Testing of the software system is done against the output and the output testing was completed with success.

### **8.1.5 User Acceptance Testing**

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system procedures operate to system specification and the integrity of the vital data is maintained.

### 8.1.6 System Testing

System Testing is a level of software testing where a complete and integrated software is tested. The purpose of this test is to evaluate the system's compliance with the specified requirements.

| Sl No | Procedures   | Expected result  | Actual result    | Pass or Fail |
|-------|--|--|------------------|--------------|
| 1     | Outputs are the correct result based on the input. | Is lung cancer ,booking,complaint and feedback and searching medicine and doctor prescription is working | Same as expected | Pass         |

Table 8.3: System test cases and results

# **Chapter 9**

## **SYSTEM IMPLEMENTATION**

### **9.1 Introduction To System Implementation**

System implementation is the final phase that putting the utility into action. Implementation is the state in the project where theoretical design turned into working system. Implementation involves the conversion of a basic application to complete replacement with a computer system. It is the process of converting to a new or revised system design into an operational one. During the design phase, the products structure, its undergoing data structures, the general algorithms and the interfaces and control data linkages needed to support communication among the various sub structures were established.

Implementation process is simply a translation of the design abstraction into the physical realization, using the language of the target architecture. Implementation includes all those activities that take place to convert from the old system to the new. The new system may be totally new replacing an existing manual or automated system, or it may be major modification to an existing system. In either case, proper implementation is essential to provide a reliable system to meet organizational requirements.

**There are three types of implementations:**

- Implementation of a computer system to replace a manual system.
- Implementation of a new computer system to replace an existing one.
- Implementation of a modified application to replace an existing one, using the same computer.

## 9.2 The common approaches for implementation

### 9.2.1 Parallel Conversion

In parallel conversion the existing system and new system operates simultaneously until the project team is confident that the new system is working properly. The outputs from the old system continue to be distributed until the new system has proved satisfactorily parallel conversion is a costly method because of the amount of duplication involved

### 9.2.2 Direct Conversion

Under direct conversion method the old system is discontinued altogether and the new system becomes operational immediately. A greater risk is associated with direct conversion is no backup in the in the case of system fails.

### 9.2.3 Pilot Conversion

A pilot conversion would involve the changing over of the part of the system either in parallel or directly. Use of the variation of the two main methods is possible when part of the system can be treated as a separate entity.

### 9.2.4 User Training

After the system is implemented successfully, training of the user is one of the most important subtasks of the developer. For this purpose user manuals are prepared and handled over to the user to operate the developed system. Thus the users are trained to operate the developed system. In order to put new application system into use.

**The following activities were taken care of:**

- Preparation of user and system documentation.
- Conducting user training with demo and hands on.
- Test run for some period to ensure smooth switching over the system.

**The major implementation procedures are:**

- Test plans
- Training
- Conversion

### **Test Plans**

The implementation of a computer based system requires that the test data be prepared and the system and its elements be tested in a structured manner.

### **Training**

The purpose of training is to ensure that all the personal who are to be associated with the computer based business system possesses the necessary knowledge skills. As the system provides user friendliness only basic training is needed.

### **Conversion**

It is the process of performing all of the operations that results directly in the turn over of the new system to the user. Conversion has two parts: The creation of a conversion plan at the start of the development phase and the implementation of the plan throughout the development phase. The creation of a system change over plan at the end of the development phase and the implementation of the plan at the beginning of the operation phase.

## **9.2.5 Coding**

When large volumes of data are being handled, it is important that items must be identified, stored or selected easily and quickly. To accomplish this each data item must have a unique identification and must be related to other item of data of the same type. Code can provide briefs identification of data item and replace longer description that would be more specific and manipulated. A code has a group of characters used to identify an item of data.

### 9.2.6 Code Plans

The code plans identifies the particular characteristics that need to be continued with in the code, whatever the nature of code they must be:

- Expandable
- Precise
- Convenient
- Meaningful

# **Chapter 10**

# **SYSTEM MAINTENANCE**

## **10.1 Introduction To System Maintenance**

The maintenance is an important activity in the life cycle of a software product. Maintenance includes all the activities after the installation of software that is performed to keep the system operational. The maintenance phase of a software life cycle is the time period in which a product performs useful work. Maintenance is classified into four types:

- Corrective Maintenance
- Adaptive Maintenance
- Perfective Maintenance
- Preventive Maintenance

### **10.1.1 Corrective Maintenance**

Corrective maintenance refers to changes made to repair defects in the design, coding, or implementation of the system. Corrective maintenance is often needed for repairing processing or performance failures or making changes because of previously uncorrected problems or false assumptions. Most corrective maintenance problems surface soon after the installation. When corrective maintenance problems surface, they are typically urgent and need to be resolved to curtail possible interruptions in normal business activities.

### **10.1.2 Adaptive Maintenance**

Adaptive maintenance involves making changes to an information system to evolve its functionality or to migrate it to different operating environment adaptive maintenance is usually less urgent than corrective maintenance because of business and technical changes typically occur some period of time.

### **10.1.3 Perfective Maintenance**

Perfective maintenance involves making enhancements to improve processing performance, interface usability, or to add desired, but not necessarily required, system features. Many system professionals feel that perfective maintenance is not really the maintenance but new development.

### **10.1.4 Preventive Maintenance**

Preventive maintenance is the only maintenance activity which is carried out without formal maintenance request from the user. When a software company or maintenance agency realizes that the methodologies used in a program have become obsolete, it may decide to develop or modify parts of the program, which do not confirm to the current trends. Of these types, more time and money is spending on perfective than on corrective and adaptive maintenance together.

# **Chapter 11**

## **RESULT AND DISCUSSION**

The project's main goal was to present My webpage named as "Virtual Medico". This webpage is useful for peoples that required for health services due to the some critical situation or some pandemic diseases, majority of the people are refraining to go to hospitals. Even though you cannot physically see a doctor .Here we have Online doctor work great for treating non emergency conditions such as minor bacterial infections, the cold and flu, and many mental health conditions. They are also useful for managing chronic conditions and getting prescriptions filled and refilled, without having to go to the doctor.And if there is a emergency condition we can book the doctor for specific time and we find the nearest medical shop that the medicine is available and we can predict lung cancer which is a common disease we don't want to waste time .

### **11.1 Advantages and Limitations**

The proposed system is a machine learning model to predict the lung cancer and take doctor prescription for non emergency situations and booking the specialized doctors for emergency situation for time saving and to search the availability of medicine in our near by pharmacy .

#### **11.1.1 Advantages**

- Lung cancer prediction : we can predict by providing a jpeg image of Cect thorax scan .
- Simple doctor consultation: we can consult doctor for non emergency situation and get prescription .

- Booking doctors: we can book specialized doctors for emergency condition with specific time.
- Pharmacy : we can search nearby pharmacy and seach the availability of the medicine.
- Feed back: Feed back option for the hearing the user recommendation.

### **11.1.2 Limitations**

- There is no video conferencing for live consultation . If we include that it will more effective for the users .
- Medicine delivery is not a feature.
- This is only web application so it is not user friendly for other platform.

# **Chapter 12**

## **CONCLUSION**

The ‘Virtual Medico’ is developed using python. While developing the system a conscious effort has been made to create and develop a software package, making use of available tools, techniques and resources that would generate a proper system. While making the system, an eye has been kept on making it as user-friendly, as cost effective and as flexible as possible. As such one may hope that the system will be acceptable to any user and will adequately meet his/her needs. It saves time and gives easy access for already stored information.

The “Virtual Medico” after being tested and was found to be achieving what is meant for. The system is found to be 100 percentage error free and ready for implementation. The system has been designed in such a way that it can be modified with very little effort when such a need arises in the future. The system has been found to work efficiency and effectively.

# **Chapter 13**

## **FUTURE SCOPE**

In this proposed system 'VIRTUAL MEDICO' the user can consult doctor and get pre-scription and he can search pharmacy and medicine according to desire . Here user can predict the lung cancer and if necessary he can book the doctor for live consultation.

The scope of the project will be vast enough as it contains a Virtual Doctor who itself treat the patients (users) and will prescribe medicines and if user can order the medicines and will receive it at his doorstep. In future it can added as to find out the medicines available near to the pharmacies from where the user is using the app (through position navigation) and it will sort the price of the medicine available in the pharmacies and helps the user to buy the medicine at the lowest price available near to him .Currently there is no online medicine purchasing in future we can add online purchasing and delivery . It is good to have video doctor consultation for live experience and can also include E-mail and SMS facility for all booking as a alert . For make it user friendly, we can develop a mobile application.

# Chapter 14

## APPENDIX

### 14.1 Sample Source Code

#### Web Code

```
from flask import*
from werkzeug.utils import secure_filename
from src.myknn import prep
from src.dbop import *
import smtplib
from email.mime.text import MIMEText
app=Flask(__name__)
app.secret_key="hsptl"
@app.route('/',methods=['get','post'])
def main():
    return render_template('login.html')
@app.route('/adminhome',methods=['get','post'])
def adminhome():
    return render_template('adminhome.html')
@app.route('/hospitalhome',methods=['get','post'])
def hospitalhome():
    return render_template('hospitalhome.html')
@app.route('/doctorhome',methods=['get','post'])
```

```
def doctorhome():
    return render_template('doctorhome.html')

@app.route('/login',methods=['get','post'])
def login():
    username=request.form['textfield']
    password=request.form['textfield2']
    rd+=" res=select(qry)
    print(qry)
    print(res)
    if res is None:
        >'''
    else:
        if res[3]=='admin':
            return :
        elif res[3]=='doctor':
            session['lid'] = str(res[0])
        elif res[3]=='pharmacy':
            session['lid'] = str(res[0])
        elif res[3]=='user':
            session['lid'] = str(res[0])
    @app.route('/viewhospital', methods=['get', 'post'])
    def viewhospital():
        qry="select * from hospital"
        res=selectall(qry)
        return render_template('viewhospital.html',val=res)
    @app.route('/viewnotification', methods=['get', 'post'])
    def viewnotification():
        qry="select * from notification"
```

```

res=selectall(qry)

    return render_template('viewnotification.html',val=res)
@app.route('/approvepharmacy', methods=['get', 'post'])
def approvepharmacy():
    q="SELECT `pharmacy`.* FROM `pharm

        return render_template('approvepharmacy.html',val=res)
@app.route('/approvepharmacy1', methods=['get', 'post'])
def approvepharmacy1():
    id=request.args.get('id')
    q="update login set type='pharmacy' where id="+id
    iud(q)

def approvepharmacy2():
    id=request.args.get('id')
    q="update login set type='reject' where id="+id
    ,

@app.route('/viewfeedback', methods=['get', 'post'])
def viewfeedback():
    return render_template('viewfeedback.html',val=res)
@app.route('/addfeedback', methods=['get', 'post'])
def addfeedback():def approvepharmacy2():
    id=request.args.get('id')
    q="update login set type='reject' where id="+id
    ,

@app.route('/viewfeedback', methods=['get', 'post'])
def viewfeedback():
    return render_template('viewfeedback.html',val=res)
@app.route('/addfeedback', methods=['get', 'post'])
def addfeedback():
    return render_template('addfeedback.html')

```

```

@app.route('/useraddfeedback', methods=['get', 'post'])
def useraddfeedback():
    feedback=request.form['textfield']
    '/managecomplaint', methods=['get', 'post'])
def managecomplaint():
    te('managecomplaint.html',val=res)
@app.route('/addhospital', methods=['get', 'post'])
def addhospital():
    return render_template('addhospital.html')
@app.route('/addhospitall', methods=['get', 'post'])
def addhospitall():
    try:
        name=request.form['textfield']
        place=request.form['textfield2']
        post=request.form['textfield3']
        pin=request.form['textfield4']
        email=request.form['textfield5']
        phone=request.form['textfield6']
        password=request.form['textfield8']
        username=request.form['textfield7']
        q="insert into login values (NULL,'" +username+ "', '" +password+
        "', 'hospital') " id=iud(q)
        q="insert into hospital values(NULL, '" +str(id)+ "', '" +name+ "' ,
        '" +place+ "', '" +post+ "', '" +pin+ "', '" +email+ "', '" +phone+ "') "
        iud(q)
        return redirect(url_for('addhospital'))
    except Exception as e:
        return '''<script>alert("already exist");window.location="/addhospital
#intro"</script>'''
@app.route('/delete_hospital', methods=['get', 'post'])
def delete_hospital():
    id=request.args.get('id')

```

```

q="delete from hospital where login_id=%s"
val=str(id)

iud2(q,val)

q = "DELETE FROM 'login' WHERE 'id'=%s"
val = str(id)

iud2(q, val)
q="DELETE FROM  'login' WHERE 'id' IN(SELECT 'login_id' FROM 'doctor' WHERE
'hid'=%s)"iud2(q, val)
q="DELETE FROM 'doctor' WHERE 'login_id' IN(SELECT 'login_id' FROM 'doctor'
WHERE 'hid'=%s)" iud2(q, val)
,,
@app.route('/addnotification', methods=['get', 'post'])

def addnotification():
    return render_template('addnotification.html')

@app.route('/addnotification1', methods=['get', 'post'])

def addnotification1():
    notification=request.form['textarea']
    q="insert into notification values(NULL,'" +notification+"',curdate())"
    iud(q)
    print("Couldn't setup email!!" + str(e))

    print(msg)
    msg['Subject'] = 'Appointment Info'
    msg['To'] = email
    msg['From'] = 'projectmailsample@gmail.com'
    try:
        gmail.send_message(msg)
    except Exception as e:
        print("COULDN'T SEND EMAIL", str(e))

@app.route('/finishedbooking', methods=['get', 'post'])

```

```

def finishedbooking():
    id=request.args.get('id')
    s e:
        re
# if __name__=="__main__":
app.run(debug=True)

```

## KNN

```

import math
from tkinter import Tk, Label, Entry
import numpy as np
from collections import Counter
from src.featureextract import glcm_feat
class CKNN:
    def __init__(self):
        self.accurate_predictions = 0
        self.total_predictions = 0

        with open('labels.dat') as f:
            lines = f.readlines()
            lines=[s.strip('\n') for s in lines]
training_data=np.loadtxt("sample.data",
dtype=float,delimiter=" ")
        self.training_set= { '1':[],'2':[]}
        #Split data into training and test for cross validation
        #training_data = lbls[: len(lbls)]
        test_data = []#[-int(test_size * len(dataset)):]
#Insert data into the training set
cnt=0

        for record in training_data:

```

```

        st=lines[cnt][0]
        cnt+=1
        self.training_set[st[-1]].append( record[:])
def predict(self, to_predict, k = 1):
    distributions = []
    for group in self.training_set:
        i=0
        # print(group,'group')
        for features in self.training_set[group]:
            euclidean_distance = np.linalg.norm(np.array(features)- np.array(to_predict))

            distributions.append([euclidean_
                instance, group])
    # print(distributions)
    results = [i[1] for i in sorted(distributions)[:k]]
    result = Counter(results).most_common(1)[0][0]
    # print("rs",results,self.training_set.keys())
    confidence = Counter(results).most_common(1)[0][1]/k
    return result, confidence

def prep(filename):
    feat=glcm_feat(filename)
    knn = CKNN()
    res=knn.predict(feat)
    return str(res[0])

```

## Booking

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.
w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

```

```
<title>Untitled Document</title>
</head>
<body>
<form id="form1" name="form1" method="post" action="">
    <div style="width:60%;">
        <table width="200" border="1" id="customers">tr>
            <th colspan="2"><div align="center" ><h2>Add Complaint</h2></div> <tr>
                <td>name</td>
                <td><label>
                    <input type="text" name="textfield" pattern="[A-Z a-z.]{3,25}" />
                </label></td>
            </tr>
            <tr>
                <td>date</td>
                <td><label>
                    <input type="date" name="textfield3" maxlength="1990-01-01"
                           min="1990-01-01" />
                </label></td>
            </tr>
            <tr>
                <td colspan="2"><div align="center">
                    <label>
                        <input type="submit" name="Submit" value="register" />
                    </label>
                </div></td>
            </tr>
        </table>
    </div>
</form>
</body>
</html>
```

## 14.2 Sample screenshots



Figure 14.1: LOGIN PAGE



Figure 14.2: ADMIN HOME



| View Doctor    |                |               |                |              |                               |
|----------------|----------------|---------------|----------------|--------------|-------------------------------|
| Name           | specialization | Hospital      | Place          | Phone Number |                               |
| Dr Amala Anoop | ortho          | Baby memorial | Kozhikode      | 9876543210   | <a href="#">view schedule</a> |
| Dr.Ravi varma  | neurologist    | alshifa       | perinthalmanna | 9876546784   | <a href="#">view schedule</a> |
| Dr. Geetha Das | surgeon        | alshifa       | perinthalmanna | 9876546784   | <a href="#">view schedule</a> |

Figure 14.3: USER HOME

Login   Pharmacy reg   User reg
Virtual Medical Staff

### Register

|          |  |
|----------|--|
| fname    | <input type="text"/>   |
| lname    | <input type="text"/>   |
| gender   | <input checked="" type="radio"/> male <input type="radio"/> female |
| dob      | <input type="text"/> dd/mm/yyyy <input type="button" value=""/>    |
| place    | <input type="text"/>   |
| email    | <input type="text"/>   |
| pin      | <input type="text"/>   |
| phone    | <input type="text"/>   |
| username | <input type="text"/>   |
| password | <input type="text"/>   |

[Register](#)

Figure 14.4: USER REGISTRATION



### Manage Doctor

| name           | dob        | gender | qualification | email                  | pin    |                                 |
|----------------|------------|--------|---------------|------------------------|--------|---------------------------------|
| Dr.Ravi varma  | 1987-02-03 | Male   | MBBS,MD       | ravivarman@gmail.com   | 679540 | <a href="#">delete Schedule</a> |
| Dr. Geetha Das | 1987-04-01 | Female | MBBS          | geethadas123@gmail.com | 678543 | <a href="#">delete Schedule</a> |

[add new](#)

Figure 14.5: HOSPITAL HOME



### View Booking

| Patient Name   | Phone      | Date       | Time     |   |
|----------------|------------|------------|----------|---|
| Mohamed Rashid | 8281883581 | 2023-04-25 | 12:00:00 | <a href="#">Accept</a> <a href="#">Reject</a> |
| Mohamed Rashid | 8281883581 | 2023-04-25 | 21:53:00 | <a href="#">Accept</a> <a href="#">Reject</a> |
| hazna sherin   | 9539951238 | 2023-04-25 | 21:53:00 | <a href="#">Accept</a> <a href="#">Reject</a> |

Figure 14.6: DOCTOR HOME



| View Doctor    |                |               |                |              |                     |          |
|----------------|----------------|---------------|----------------|--------------|---------------------|----------|
| Name           | specialization | Hospital      | Place          | Phone Number | Date Time           | Status   |
| Dr Amala Anoop | ortho          | Baby memorial | Kozhikode      | 9876543210   | 2023-04-25 21:53:00 | Finished |
| Dr Amala Anoop | ortho          | Baby memorial | Kozhikode      | 9876543210   | 2023-04-25 9:00:00  | accepted |
| Dr Amala Anoop | ortho          | Baby memorial | Kozhikode      | 9876543210   | 2023-04-25 12:00:00 | accepted |
| Dr Amala Anoop | ortho          | Baby memorial | Kozhikode      | 9876543210   | 2023-04-25 21:53:00 | rejected |
| Dr. Geetha Das | surgeon        | alshifa       | perinthalmanna | 9876546784   | 2021-07-09 7:21:00  | accepted |
| Dr Amala Anoop | ortho          | Baby memorial | Kozhikode      | 9876543210   | 2023-04-25 9:00:00  | pending  |

Figure 14.7: USER BOOKIN PAGE

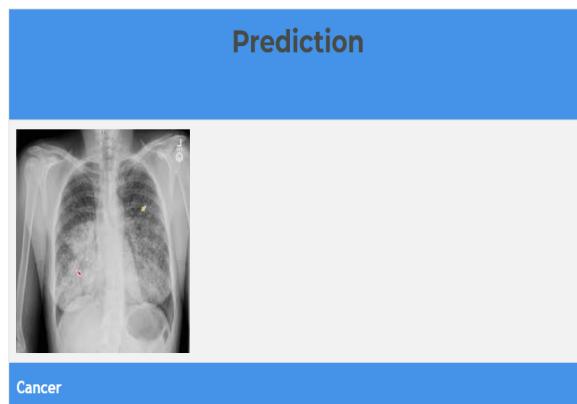


Figure 14.8: LUNG CANCER PREDICTION



| View Medicine                |                |         |       |            |                |
|------------------------------|----------------|---------|-------|------------|----------------|
| medicinename                 | description    | photo   | price | expdate    | dosage         |
| cancerio                     | for lug cancer | th.jpeg | 45    | 2023-04-27 | 1 tablet a day |
| calpol                       | no             | th.jpeg | 44    | 2022-03-31 | nill           |
| <a href="#">add medicine</a> |                |         |       |            |                |

Figure 14.9: PHARMACY HOME

| Pharmacy Register        |                      |
|--------------------------|----------------------|
| name                     | <input type="text"/> |
| place                    | <input type="text"/> |
| post                     | <input type="text"/> |
| pin                      | <input type="text"/> |
| email                    | <input type="text"/> |
| phone                    | <input type="text"/> |
| License No               | <input type="text"/> |
| Username                 | <input type="text"/> |
| Password                 | <input type="text"/> |
| <a href="#">register</a> |                      |

Figure 14.10: PHARMACY REGISTRATION

# Bibliography

- [1] Lung Cancer Prediction using Extended KNN Algorithm  
<https://ieeexplore.ieee.org/document/9753689> IEEE 13 April 2022.
- [2] Managing a virtual hospital  
<https://ieeexplore.ieee.org/document/547838>
- [3] e-Pharm Assist: The Future Approach for Dispensing Medicines in Smart Cities.  
<https://ieeexplore.ieee.org/document/9105747>. IEEE 02 June 2020
- [4] Lung Cancer Prediction using Machine Learning: A Comprehensive Approach IEEE 3 April 2021.  
<https://ieeexplore.ieee.org/document/9074947>.
- [5] S. Lung cancer Prediction and Classification based on Correlation Selection method Using Machine Learning Techniques Doi: 10.48161/Issn.2709-8206