

# E-CLINIC

*A Project Report Submitted in Partial Fulfilment of the  
Requirements for the Award of the Degree of*

Bachelor of Computer Applications  
By

Aneesha C M 210021089354  
Anto Evaniyose 210021089362

Under the guidance of

Ms. Remya R  
Assistant professor



**SANTHIGIRI**  
COLLEGE OF COMPUTER SCIENCES

Affiliated to MG University and Approved by AICTE

Department of Computer Science

NOVEMBER 2023



# SANTHIGIRI

---

## COLLEGE OF COMPUTER SCIENCES

Affiliated to MG University and Approved by AICTE

### CERTIFICATE

*This is to certify that the report titled **E-Clinic** is a bonafide record of work done by **Aneesha C M (210021089354)** and **Anto Evaniyose (210021089362)**, of Santhigiri College of Computer Sciences in partial fulfillment of the requirements of Fifth Semester of Bachelor of Computer Applications during the year 2023.*

**Rev. Fr. Prof. Dr. Baby Joseph CMI**

**Ms. Amitha Joseph**

**Ms. Remya R**

**Principal**

**HOD**

**Project Guide**

Valued-by:

1.

2.

## **DECLARATION**

We, Aneesha C M and Anto Evaniyose hereby declare that the project report, titled “**E-Clinic**” is a record of original work undertaken by us for the award of the degree of Bachelor Computer Applications. We have completed this project under the guidance of Ms Remya R, Department of Computer Science.

We also declare that this project has not been submitted for the award of any degree. We hereby confirm the originality of the work.

Aneesha C M	210021089354	
Anto Evaniyose	210021089362	

Place: Vazhithala

Date:

## ACKNOWLEDGEMENT

A project is not complete if one fails to acknowledge all who have been instrumental in the successful completion of the project. If words were to be the symbol of undiluted feelings and token of gratitude, then let the words play the heralding role of expressing our gratitude.

First of all, we thank the “**God Almighty**” for his immense grace and blessings in our life and at each stage of this project.

We express our sincere and profound gratitude to **Rev. Fr. Prof. Dr. Baby Joseph CMI**, Principal, Santhigiri College of Computer Sciences, Vazhithala for providing all the facilities during the period of the project.

We extend our gratitude to **Dr. Amitha Joseph**, Head of the Department of Computer Science, who is a constant source of inspiration and whose advice helped us to complete this project successfully.

We express our deep sense of gratitude to our internal project guide, **Ms. Remya R**, Assistant Professor, Department of Computer Science, for her profound guidance for the successful completion of this project.

With great enthusiasm we express our gratitude to all the faculty members of Department of Computer Science for their timely help and support.

Finally, we express our deep appreciation to all my friends and family members for the moral support and encouragement they have given to complete this project successfully.

## ABSTRACT

To develop a web application that allows the users to book tokens for hospitals and manage all other related things online. The main objective of the application online E-Clinic is to allow someone to book their tokens for a particular hospital. Here, user has to login to book tokens . The user can sign in to the hospital site and search for doctors and can easily book. For bookings, the user has to provide information such as booking dates and personal details. The hospital's details are provided and it also includes doctor's profile and overview. The user can update their profile as well as passwords anytime they want from the site. Admin can add/manage hospital and doctor details, bookings, pages and many more. It's easy to operate and understand for the users. This site makes customers easy to get their bookings. The design is pretty simple and the user won't find it difficult to understand, use and navigate. The healthcare industry has seen significant advancements in recent years, with technology playing a pivotal role in improving patient care and accessibility. One such innovation that has transformed the way patients interact with healthcare providers is the online booking E-Clinic. This revolutionary approach has been adopted by our esteemed hospital, and in this article, we will explore the motivation behind implementing an online booking E-Clinic, its benefits, and how it has enhanced patient experiences and healthcare delivery. The proposed system for an online E-Clinic aims to provide a convenient platform for users to book tokens for appointments at a specific hospital.

## LIST OF TABLES

<b>Table No.</b>	<b>Table Name</b>	<b>Page No.</b>
2.1	Sign off table	9
3.1	tbllogin	30
3.2	tbldepartment	31
3.3	tbldistrict	31
3.4	tbllocation	31
3.5	tbldoctorreg	32
3.6	tblpatientreg	33
3.7	tblbook	34
3.8	tblcasehistory	35
3.9	tbltest	36
3.10	tblpayment	37
3.11	Data Flow Diagram Symbols	39
5.1	Test Case	70

## LIST OF FIGURES

<b>Figure No</b>	<b>Figure name</b>	<b>Page No</b>
3.1	Zeroth level DFD for Online token booking system of a hospital	41
3.2	First level DFD for Online token booking system of a hospital	41
3.3	Second level DFD for login	41
3.4	Second level DFD for Department registration	42
3.5	Second level DFD for District Registration	42
3.6	Second level DFD for Location Registration	42
3.7	Second level DFD for Doctor Registration	42
3.8	Second level DFD for Department View	43
3.9	Second level DFD for District View	43
3.10	Second level DFD for Location View	43
3.11	Second level DFD for Doctors View	43
3.12	Second level DFD for patient payment view	44
3.13	Second level DFD for Datewise by Count of Patients in each Doctor Report	44
3.14	Second level DFD for Patient Payment Report	45
3.15	Second level DFD for Doctor Registration Report	45

3.16	Second level DFD for Patient Registration Report	46
3.17	Second level DFD for Count of Doctors in each Department Report	46
3.18	Second level DFD for Case details of Patient	47
3.19	Second level DFD for Test details of Patient	47
3.20	Second level DFD for Doctor Activities	47
3.21	Second level DFD for Patient Registration	48
3.22	Second level DFD for Patient Booking Registration	48
3.23	Second level DFD for Patient Registration	49
3.24	Doctor Registration Form	50
3.25	Button that redirects to pie chart	52
3.26	Pie charts of the patients who have booked each doctors.	53
4.1	Registration form	56
5.1	Unit testing	65
5.2	Unit testing result	65
5.3	Integration testing	67
5.4	Integration testing result	67
5.5	Patient name validation	68
5.6	Patient gender validation	69
8.1	Admin login	75
8.2	Department Registration Page	75

8.3	Doctor Registration Page	76
8.4	District Registration Page	76
8.5	Location Registration Page	77
8.6	Location View	77
8.7	Department View	78
8.8	Doctor View	78
8.9	Lab result	79
8.10	Patient Index Page	79
8.11	Patient login Page	80
8.12	Patient Registration	80
8.13	Doctors View	81
8.14	Department View	81
8.15	Patient view profile	82
8.16	Change Password of patient	82
8.17	Doctor Booking Page	83
8.18	Patient token generation	83
8.19	Booking Payment	84
8.20	Booking payment success	84

8.21	Doctor Home Page	85
8.22	Patient case history	85
8.23	Patient view	86
8.24	View case history	86
8.25	Doctor profile view	87
8.26	Piechart for count of patients in each doctor	87
8.27	Piechart for count of doctors in each department	88
8.28	Datewise doctor booking	88
8.29	Monthwise doctor booking	89
8.30	Datewise patient registration report	89
8.31	Datewise patient payment report	90
8.32	Doctor view	90
8.33	Patient view	91
8.34	Department view	91
8.35	Patient payment report	92

## **ABBREVIATION**

IDE	Integrated Development Environment
CPU	Central Processing Unit
DBMS	Data Base Management System
RDBMS	Relational Data Base Management System
NF	Normal Forms
PK	Primary Key
FK	Foreign Key
DFD	Data Flow Diagram
PHP	Hypertext Preprocessor
SQL	Structured Query Language
WAMP	Windows, Apache, MySQL and PHP
HTML	Hyper Text Markup Language
SDLC	Software Development Life Cycle Models
EHR	Electronic Health Records

## **TABLE OF CONTENTS**

<b>Acknowledgment</b>	<b>iv</b>
<b>Abstract</b>	<b>v</b>
<b>List of Tables</b>	<b>vi</b>
<b>List of Figures</b>	<b>vii</b>
<b>Abbreviations</b>	<b>viii</b>
1. Introduction	1
1.1. Background and motivation	1
1.2. The proposed system	2
1.3. Project Scope	2
1.3.1         Limitations of existing system	2
1.3.2         Advantages of proposed system	3
2. System Analysis	5
2.1. Introduction	5
2.2. Stake holders of this project	6
2.2.1         Admin	6
2.2.2         Doctor	7
2.2.3         Patient	7
2.3. Software requirement specifications	8
2.3.1         Admin	8

2.3.2	Doctor	8
2.3.3	Patient	8
2.4.	Feasibility study	10
2.5.	Software Development lifecycle model	13
2.6.	Hardware and software requirements	14
2.6.1	Software specifications	14
2.6.1.1.	PHP	14
2.6.1.2.	MySQL	15
2.6.1.3.	Wamp Server	18
2.6.1.4.	Windows 10	18
2.6.1.5.	Microsoft Word	19
2.6.1.6.	Smart draw	20
2.6.2	Hardware requirements	21
3.	System Design	23
3.1.	System Architecture	23
3.2.	Module Design	24
3.3.	Database Design	26
3.3.1	Normalization	27
3.3.2	Table Structure	28
3.3.3	Data Flow Diagram	38
3.3.3.1.	Introduction to data flow diagrams	38
3.3.3.2.	Data flow diagram	41
3.4.	Interface Design	49
3.4.1	User interface screen design	49
3.4.2	Output design	51

4.	Implementation	54
4.1.	Coding Standards	54
4.2.	Sample Code	56
5.	Testing	63
5.1.	Test Cases	63
5.1.1	Unit testing	64
5.1.2	Integration testing	66
5.1.3	Validation Testing	68
5.1.4	User acceptance testing	69
5.2.	Test case documents	70
6.	Conclusion	73
7.	References	74
8.	Appendix	75
8.1.	Screenshots	75

## 1. INTRODUCTION

### 1.1 BACKGROUND AND MOTIVATION

This project is leading to the development of improving the efficiency and accessibility of healthcare services. One such innovative concept is the E-Clinic, which combines telemedicine, electronic health records (EHR), and other digital tools to provide patients with convenient and efficient healthcare services. The E-Clinic project is a response to the evolving healthcare landscape, driven by a variety of compelling motivations and challenges that need to be addressed. The primary motivation behind the E-Clinic project is to enhance access to healthcare services. By offering virtual consultations, patients can seek medical advice without having to travel long distances or wait for extended periods. This not only benefits those in remote areas but also people with limited mobility. EClinics offer the convenience of scheduling appointments at times that suit the patient, reducing the need for lengthy waiting times in crowded waiting rooms. This convenience enhances the overall patient experience and encourages more people to seek timely medical advice. E-Clinic project represents a progressive step toward transforming healthcare delivery through technology. Its background lies in addressing the challenges of the traditional healthcare system, and its motivation stems from a desire to improve healthcare access, efficiency, and patient engagement. By embracing this innovative approach, the E-Clinic project aims to usher in a new era of healthcare that prioritizes the needs of patients in an increasingly digital world. E-Clinics integrate electronic health records (EHRs) to maintain and manage patient information securely. This digital approach eliminates the need for physical paperwork and ensures that healthcare providers have immediate access to a patient's medical history, leading to more accurate diagnoses and treatment plans.

## 1.2 THE PROPOSED SYSTEM

The proposed system for an Online E-Clinic aims to provide a convenient platform for users to book tokens for appointments at a specific hospital. This system will offer several benefits, such as streamlining the appointment booking process, reducing wait times, and enhancing overall healthcare access. Users will be able to create accounts, update personal information, and manage their profiles, including medical history and contact details. Users can choose a preferred date and time for their medical appointment and book it through the platform. They may also specify the type of specialist or department they need to consult. Upon successful booking, the system generates a unique token or reference number for the user's appointment. This token will be used for identification at the doctor. Users will receive confirmation notifications and reminders for their upcoming appointments via email. Hospital administrators will have access to a dashboard for managing appointments, allocating resources, and monitoring overall system performance. Robust security measures and data privacy protocols must be in place to protect sensitive medical information. The system should be designed to handle a large volume of appointments efficiently, especially in the case of hospitals with a high patient load. For hospitals that charge fees for appointments, the system should support secure payment processing. In summary, the Online E-Clinic system aims to revolutionize healthcare access by simplifying appointment booking processes, reducing waiting times, and enhancing overall patient experiences. It should be user-friendly, secure, and capable of accommodating a large number of users and hospitals.

## 1.3 PROJECT SCOPE

### 1.3.1 LIMITATIONS OF EXISTING SYSTEM

- Manual activities are very time consuming and a tedious job.
- Deletion and updation of information is somehow troublesome and if possible, it ends up in untidiness.
- It is very difficult to retrieve any particular data.
- Changes are difficult to make.

- The current appointment scheduling system whereby business practices employ a hospital personnel to manually record appointment will lead to patient wastage of time on the telephone while waiting to receive assistance
- A very large amount of data redundancy occurs.
- Report generation is not an easy task and the accuracy of the reports generated cannot be guaranteed.
- Manipulations of information together with calculations are done manually and typically accuracy is compromised.
- Time consuming

In this system all the transactions are done manually. It takes more time to write every record, book a token, payment etc.

- More human effort

The one who controls the whole system must deal with the heavy workload

### **1.3.2 ADVANTAGES OF PROPOSED SYSTEM**

- Enhanced Patient Access  
E-clinics break down geographical barriers, allowing patients to access healthcare services from anywhere with an internet connection. This is especially valuable for individuals in remote or underserved areas who may otherwise struggle to reach a physical clinic.
- Save time
- Reduced Administrative Costs  
Automation of administrative tasks reduces the need for manual labor, saving time and resources.

- Billing and Payments

Automated billing processes minimize errors, improve revenue collection, and simplify financial transactions for both clinics and patients.

- Accurate Electronic Health Records (EHRs)

E-clinics replace traditional paper records with digital EHRs. This ensures that patient information is readily available to authorized healthcare providers, reducing the risk of errors and misplacement of records.

- Less use of manual work.

## 2. SYSTEM ANALYSIS

### 2.1 INTRODUCTION

Software Engineering is the analysis, design, construction, verification and management of technical or social entities. To engineer software accurately, a software engineering process must be defined. System analysis is a detailed study of the various operations performed by the system and their relationship within and module of the system. It is a structured method for solving the problems related to the development of a new system. The detailed investigation of the present system is the focal point of system analysis. This phase involves the study of parent system and identification of system objectives. Information has to be collected from all people who are affected by or who use the system. During analysis, data are collected on the variable files, decision point and transactions handled by the present system. The main aim of system is to provide the efficient and user friendly automation. So the system analysis process should be performed with extreme precision, so that an accurate picture of existing system, its disadvantages and the requirements of the new system can be obtained.

System analysis involves gathering the necessary information and using the structured tool for analysis. This includes the studying existing system and its drawback, designing a new system and conducting cost benefit analysis.

System analysis is a problem solving activity that requires intensive communication between the system users and system developers. The system is studied to the minute detail and analyzed. The system is viewed as a whole and the inputs to the system are identified. The outputs from the organization are traced through various phases of processing of inputs.

There are a number of different approaches to system analysis. When a computer based information system is developed, systems analysis (according to the Waterfall model) would constitute the following steps:

- The development of a feasibility study, involving determining whether a project is economically, technologically and operationally feasible.

- Conducting fact-finding measures, designed to ascertain the requirements of the system's end-users. These typically span interviews, questionnaires, or visual observations of work on the existing system
- Gauging how the end-users would operate the system (in terms of general experience in using computer hardware or software), what the system would be used for and so on.

Techniques such as interviews, questionnaires etc. can be used for the detailed study of these processes. The data collected by these sources must be scrutinized to arrive at a conclusion.

The conclusion is an understanding of how the system functions. This system is called the Existing System. The Existing system is then subjected to close observation and the problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as a proposal which is the Proposed System. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is then presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This is a loop that ends as soon as the user is satisfied with the proposal.

## **2.2 STAKEHOLDERS IN THIS PROJECT**

### **2.2.1 ADMIN**

Admin serve as the backbone of any healthcare institution, and this role holds true in the context of an E-clinic. Their responsibilities are multifaceted and critical to the clinic's overall functioning, and they contribute to various aspects of the clinic's operations. Admin in an E-clinic are responsible for overseeing and managing day-to-day operations. The admin controls the registration of doctor, departments, and location. Admin play a role in shaping the patient experience by overseeing appointment scheduling, managing patient records, and ensuring a user-friendly interface for virtual records. A seamless and userfriendly experience is essential for patient retention and

satisfaction. Admin is the most important stakeholder in E-clinic. It plays a major role for the E-clinic's success by managing various aspects of its operation, including registration process, accuracy, technology, payment, and the patient experience. Their role is complex and multifaceted, involving a deep understanding of healthcare administration, technology, and regulatory matters. He can also check the comments of customer.

He can also check the payment status.

### **2.2.2 DOCTOR**

A Doctor of an Online E-Clinic plays a pivotal role in bridging the gap between healthcare providers and patients in the digital age. This multifaceted position encompasses a wide array of responsibilities and duties aimed at ensuring efficient, accessible, and quality healthcare services for individuals seeking medical care from the comfort of their own. In this comprehensive description, we will delve into the various aspects of the role, its significance, and the impact it has on modern healthcare. Doctors in E-Clinics educate patients about their conditions, treatment plans, and preventive measures. They ensure that patients have a clear understanding of their health and can make informed decisions.

Doctors must manage their schedules, ensuring that appointments are available and patients are seen in a timely manner. Doctors issue digital prescriptions for medications, which patients can easily access and fulfill at their chosen pharmacies.

### **2.2.3 PATIENT**

Patients are the main user and stakeholders of the system. The system is developed to help the patients to book their tokens easily in the online. They can select doctors of their needs from the given categories. The system enables the patients to register to the system online, select the doctor which they want to book at their convenient day and time. Also, the payment can be made through online mode. Patients can browse through a list of hospitals and healthcare facilities, each accompanied by detailed profiles that include information about the services they offer, the expertise of their medical staff, and patient reviews.

## **2.3 SOFTWARE REQUIREMENT SPECIFICATION**

### **2.3.1 ADMIN**

1. This System should have the provision for login using username and password.
2. The system should have the provision to access the home page.
3. The system should have the provision to accept or reject the registered patients.
4. Admin should have the permission to change password.
5. Admin should have the permission to add/view/edit/remove details.
6. Admin should have the permission to add details and modify details in resource page.
7. Admin should have the permission to view the details of daily tasks.
8. Admin should have the permission to update the page.
9. Admin should have the permission to view the registrations.
10. Admin should have the permission to the payment status.
11. Admin should have the permission to add feactures.
12. Admin should have the permission to access the location of patients.
13. The system should have the provision to logout.

### **2.3.2 DOCTOR**

1. Doctors can register on to the website and make their own profiles.
2. They can make changes like edit and delete the profiles.
3. Doctors have to login to the site using username and password.
4. Doctor can add case history of a patient.
5. Doctor can add prescription and other details of a patient.
6. Doctor can view the number of bookings of a day
7. There will be a limit in registration of patients for a doctor in a day.
8. Doctors have a particular number of OP days.

### **2.3.3 PATIENT**

1. Registration shall be provided with a provision to register as a new user.
2. System should allow the customers to login to the site using username and password

3. System should have the provision to change password.
4. The system should have the provision to agree the terms and conditions.
5. The system should have the provision to schedule the date and time.
6. System should allow the customers to perform profile updation .
7. System should allow the patient to select their doctors according to their needs.
8. System should allow the patients to book.
9. System should allow the patients to pay consulting fee.
10. The system should have the provision to view the profile.
11. The system should allow the customers to review the product.
12. The system should have the provision to logout

Table 2.1. Sign off table

<b>Sl. No.</b>	<b>Name &amp; Designation</b>	<b>Date</b>	<b>Accepted (Yes/No)</b>
1	Ms. Remya R Assistant Professor Santhigiri college of Computer Sciences	16-6-2023	Yes
2	Aneesha C M Developer	16-6-2023	Yes
3	Anto Evaniyose Developer	16-6-2023	Yes

## 2.4 FEASIBILITY STUDY

The practical degree to which a project can be carried out successfully is known as feasibility. A feasibility study is conducted to assess the viability of a solution, determining whether or not it is feasible and implementable in the software to meet the requirements. During the feasibility study, factors including resource availability, software development cost estimation, post-creation software benefits to the company, and maintenance costs are taken into account. Establishing the justifications for creating software that is user-acceptable, flexible, and compliant with established standards is the aim of the feasibility study. The following is a list of additional feasibility study goals.

- To evaluate if the program satisfies organizational needs.
- To ascertain if the program can be implemented within the allocated spending limit and time frame while utilizing the available technology.
- To ascertain whether the program can be merged with other programs that are already in use.

When our project guide and our client, Ms. Remya R, informed us about the small project and Word to the Wise for developing the desired product, we were able to develop a general understanding of the features and functions that the software must have.

With reference to this data, we investigate and debate whether developing the intended system and its functionality is viable. The result of this step is a feasibility study report that ought to include sufficient remarks and suggestions.

Technical, operational, and economic feasibility are among the several forms of feasibility that we examined.

### **Technical Feasibility**

Technical feasibility evaluates the available technology and resources (hardware and software) that are needed to fulfill user requirements in the software within the allotted time and budget. To do this, the program development team determines if the available

tools and technology can be updated or added to the program to fulfill certain user requirements. Additionally, technical feasibility carries out the following duties.

- Analyses the technical skills and capabilities of the software development team members.
- Determines whether the relevant technology is stable and established.
- Ascertains that the technology chosen for software development has a large number of users so that they can be consulted when problems arise or improvements are required.

From our vantage point, the web based apps are developed using two languages: PHP, HTML, and the MySQL database. The front end uses PHP, while the back end uses MySQL. Web-based, The Word to the Wise thus it's accessible from any browser. While utilizing these most recent technologies that are popular right now and that many developers use We may declare that our project is technically feasible wherever in the world.

### **Operational Feasibility**

Operational feasibility evaluates how well the necessary software executes a sequence of actions to address user needs and business concerns. The software development team's human resources are necessary for this possibility.

It entails determining through visualization if the software will function both during development and after installation. Additionally, operational viability carries out the upcoming tasks.

- Determines whether the problems anticipated in user requirements are of high priority.
- Determines whether the solution suggested by the software development team is acceptable.
- Analyses whether users will adapt to a new software.

- Determines whether the organization is satisfied by the alternative solutions proposed by the software development team.

We discovered that the client would be happy with our project because we had constant communication with them regarding every aspect of the product. The client's feedback is the most crucial component of an operational feasibility study. Consequently, the software is created entirely in compliance with the client's specifications. We have made use of the industry standards that are currently in effect for the software. Thus, it can be said that this program is practical from an operational standpoint.

### **Economic Feasibility**

The ability of the necessary software to bring in money for a company is determined by its economic feasibility. It includes expenses paid to the software development team, the approximate cost of the necessary hardware and software, the cost of carrying out the feasibility study, and so forth. For this reason, it is crucial to take into account the costs associated with purchases (such hardware purchases) and the actions necessary to complete software development. The advantages that can be obtained by building the program must also be taken into account. If software addresses the following problems, it is considered economically feasible.

- Cost incurred on software development to produce long-term gains for an organization.
- Cost required to conduct full software investigation (such as requirements elicitation and requirements analysis).
- Cost of hardware, software, development team, and training.

Our concept is thought to be economically realistic because it only required very little development costs because the tools and technology were found online. There are no personnel costs because this is a collective student effort. The development period is carefully scheduled and won't interfere with other personal operations or pursuits. Following system development, a handbook will be supplied to the companies making the purchase for training purposes. Since the new system may still be implemented on the current PCs, no new hardware needs to be purchased.

---

## 2.5 SOFTWARE DEVELOPMENT LIFECYCLE MODEL

One of the basic notions of the software development process is SDLC models which stand for Software Development Life Cycle models. SDLC – is a continuous process, which starts from the moment, when it's made a decision to launch the project, and it ends at the moment of its full remove from the exploitation. Software development lifecycle (SDLC) is a framework that defines the steps involved in the development of software. It covers the detailed plan for building, deploying and maintaining the software. SDLC defines the complete cycle of development i.e. all the tasks involved in gathering a requirement for the maintenance of a Product. Some of the common SDLC models are Waterfall Model, V-Shaped Model, Prototype Model, Spiral Model, Iterative Incremental Model, Big Bang Model, Agile Model. We used Agile Model for our Project.

### **Agile Model**

Agile Model is a combination of the Iterative and incremental model. This model focuses more on flexibility while developing a product rather than on the requirement. In the agile methodology after every development iteration, the client is able to see the result and understand if he is satisfied with it or he is not. Extreme programming is one of the practical use of the agile model. The basis of this model consists of short meetings where we can review our project. In Agile, a product is broken into small incremental builds. It is not developed as a complete product in one go. At the end of each sprint, the project guide verifies the product and after his approval, it is finalised. Client feedback is taken for improvement and his suggestions and enhancement are worked on in the next sprint.

Testing is done in each sprint to minimize the risk of any failures.

### **Advantages of Agile Model:**

- It allows more flexibility to adapt to the changes.
- The new feature can be added easily.
- Customer satisfaction as the feedback and suggestions are taken at every stage.

- Risks are minimized thanks to the flexible change process

**Disadvantages:**

- Lack of documentation.
- If a customer is not clear about how exactly they want the product to be, then the project would fail.
- With all the corrections and changes there is possibility that the project will exceed expected time

## **2.6 HARDWARE AND SOFTWARE REQUIREMENTS**

### **2.6.1 SOFTWARE SPECIFICATION**

This project is built upon the latest technology software.

Front end : HTML, JavaScript

Development tool : PHP

Database : My SQL

Web server : WAMP server

Operating System : Windows 10

#### **2.6.1.1PHP**

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. As of January 2013, PHP was installed on more than 240 million websites (39% of those sampled) and 2.1 million web servers. Originally created by Rasmus Lerdorf in 1994, the reference implementation of PHP is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Preprocessor, a recursive acronym.

PHP code can be simply mixed with HTML code, or it can be used in combination with various templating engines and web frameworks. PHP code is usually processed by a PHP interpreter, which is usually implemented as a web server's native module or a Common Gateway Interface (CGI) executable. After the PHP code is interpreted and executed, the web server sends resulting output to its client, usually in form of a part of the generated web page - for example, PHP code can generate a web page's

HTML code, an image, or some other data. PHP has also evolved to include a commandline interface (CLI) capability and can be used in standalone graphical applications.

PHP is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge

### 2.6.1.2 MySQL

MySQL is the world's most popular open source database software, with over 100 million copies of its software downloaded or distributed throughout its history. With its superior speed, reliability, and ease of use, MySQL has become the preferred choice for Web, Web 2.0, SaaS, ISV, Telecom companies and forward-thinking corporate IT Managers because it eliminates the major problems associated with downtime, maintenance and administration for modern, online applications.

Many of the world's largest and fastest-growing organizations use MySQL to save time and money powering their high-volume Web sites, critical business systems, and packaged software — including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, Wikipedia, and Booking.com.

The flagship MySQL offering is MySQL Enterprise, a comprehensive set of productiontested software, proactive monitoring tools, and premium support services available in an affordable annual subscription.

MySQL is a key part of LAMP (Linux, Apache, MySQL, PHP / Perl / Python), the fastgrowing open source enterprise software stack. More and more companies are using LAMP as an alternative to expensive proprietary software stacks because of its lower cost and freedom from platform lock-in.

MySQL was originally founded and developed in Sweden by two Swedes and a Finn: David Axmark, Allan Larsson and Michael "Monty" Widenius, who had worked together since the 1980's. MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by Oracle Corporation.

MySQL is a database management system. A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

MySQL databases are relational. A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to one, onetomany, unique, required or optional, and —pointers|| between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data.

The SQL part of MySQL|| stands for Structured Query Language||. SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a languagespecific API that hides the SQL syntax.

SQL is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, SQL-92|| refers to the standard released in 1992, SQL:1999|| refers to the standard released in 1999, and SQL:2003|| refers to the current version of the standard.

We use the phrase the SQL standard|| to mean the current version of the SQL Standard at any time. MySQL software is Open Source. Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the

source code and change it to suit your needs. MySQL software use the GPL (GNU General Public License).

<http://www.fsf.org/licenses/>, to define what you may and may not do with the software in different situations. If you feel uncomfortable with the GPL need to embed MySQL code into a commercial application, you can buy a commercially licensed version from us.

The MySQL Database Server is very fast, reliable, scalable, and easy to use. If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available.

MySQL can also scale up to clusters of machines, networked together.

MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Although under constant development, MySQL Server today offers a rich and useful set of functions. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet. MySQL Server works in client/server or embedded systems.

The MySQL Database Software is a client/server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs). We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

A large amount of contributed MySQL software is available. MySQL Server has a practical set of features developed in close cooperation with our users. It is very likely that your favourite application or language supports the MySQL Database Server.

### 2.6.1.3 WAMP SERVER

WAMP Server is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your databases. WAMP Server refers to a software stack for the Microsoft Windows operating system, created by Romain Bourdon and consisting of the Apache web server, Open SSL for SSL support, MySQL database and PHP programming language. WAMP Server is a Web development platform on Windows that allows you to create dynamic Web applications with Apache2, PHP, MySQL and MariaDB. WampServer automatically installs everything you need to intuitively develop Web applications. You will be able to tune your server without even touching its setting files. Best of all, WampServer is available for free (under GPL license) in both 32 and 64 bit versions. Wampserver is not compatible with Windows XP, SP3, or Windows Server 2003.

WAMP Server's functionalities are very complete and easy to use so we won't explain here how to use them.

With a left click on WAMP Server's icon, you will be able to:

- manage your Apache and MySQL services
- switch online/offline (give access to everyone or only localhost)
- install and switch Apache, MySQL and PHP releases
- manage your server's settings
- access your logs
- access your settings files
- create alias

### 2.6.1.4 WINDOWS 10

Operating System is defined as a program that manages the computer hardware. An operating system can be viewed as a scheduler, where it has resources for which it has charge. Resources include CPU, memory, I/O device and disk space. In another view, the operating system is a new machine. The third view is that operating system is a

multiplexer which allows sharing of resources provides protection from interference and provides a level of cooperation between users. This project is developed using Windows 10 as the operating system and supports its latest versions. Windows 10 is a series of personal computer operating systems produced by Microsoft as part of its Windows NT family of operating systems. It is the successor to Windows 8.1, and was released to manufacturing on July 15, 2015, and to retail on July 29, 2015. One of Windows 10's most notable features is support for universal apps. Windows 10 also introduced the Microsoft Edge web browser, a virtual desktop system, a window and desktop management feature called Task View, support for fingerprint and face recognition login, new security features for enterprise environments, and DirectX12. Windows 10 received mostly positive reviews upon its original release in July 2015. Critics praised Microsoft's decision to provide a desktop-oriented interfacing line with previous versions of Windows, contrasting the tablet-oriented approach of 8, although Windows 10's touch-oriented user interface mode was criticized for containing regressions upon the touch-oriented interface of Windows 8. Critics also praised the improvements to Windows 10's bundled software over Windows 8.1, Xbox Live integration, as well as the functionality and capabilities of the Cortana personal assistant and the replacement of Internet Explorer with Microsoft Edge. However, media outlets have been critical of changes to operating system behaviours, including mandatory update installation, privacy concerns over data collection performed by the OS for Microsoft and its partners and the adware-like tactics used to promote the operating system on its release.

#### **2.6.1.5 MICROSOFT WORD**

Microsoft Word (or simply Word) is a word processor developed by Microsoft. It was first released on October 25, 1983 under the name *Multi-Tool Word* for Xenix systems. Subsequent versions were later written for several other platforms including IBM PCs running DOS (1983), Apple Macintosh running the Classic Mac OS (1985), AT&T Unix PC (1985), Atari ST (1988), OS/2 (1989), Microsoft Windows (1989), SCO Unix (1994), and macOS (formerly OS X; 2001).

Commercial versions of Word are licensed as a standalone product or as a component of Microsoft Office, Windows RT or the discontinued Microsoft Works suite. Unlike most MS-DOS programs at the time, Microsoft Word was designed to be used with a mouse. Advertisements depicted the Microsoft Mouse, and described Word as a WYSIWYG, windowed word processor with the ability to undo and display bold, italic, and underlined text, although it could not render fonts. It was not initially popular, since its user interface was different from the leading word processor at the time, WordStar. However, Microsoft steadily improved the product, releasing versions 2.0 through 5.0 over the next six years. In 1985, Microsoft ported Word to the classic Mac OS (known as Macintosh System Software at the time). This was made easier by Word for DOS having been designed for use with high-resolution displays and laser printers, even though none were yet available to the general public. Following the precedents of LisaWrite and MacWrite, Word for Mac OS added true WYSIWYG features. It fulfilled a need for a word processor that was more capable than MacWrite. After its release, Word for Mac OS's sales were higher than its MS-DOS counterpart for at least four years.

#### **2.6.1.6 SMARTDRAW**

SmartDraw is a diagram tool used to make flowcharts, organization charts, mind maps, project charts, and other business visuals. SmartDraw has two versions: an online edition and a downloadable edition for Windows desktop.

SmartDraw integrates with Microsoft Office products including Word, PowerPoint, and Excel and G Suite applications like Google Docs and Google Sheets. SmartDraw has apps for Atlassian's Confluence, Jira, and Trello. SmartDraw is compatible with Google Drive, Dropbox, Box, and OneDrive.

Since 1994, the mission of SmartDraw Software has been to expand the ways in which people communicate so that we can clearly understand each other, make informed decisions, and work together to improve our businesses and the world. We accomplish this by creating software and services that make it possible for people to capture and present information as visuals, while being a pleasure to use. In 2019, we took this to the

next level by launching VisualScript, which makes it easy to visualize data in relational formats like trees, flows, and timelines, automatically, without any human input. VisualScript is a relationship visualization platform that empowers organizations to visualize data across siloed ecosystems and gain critical insights in real-time. Today, SmartDraw Software is one of the most sophisticated digital marketing organizations in the world with over 90,000 unique visitors to our website each business day and in excess of 3,000,000 installations of our apps each year. SmartDraw is used by more than half of the Fortune 500 and by over 250,000 public and private enterprises of all sizes around the world. Privately held, SmartDraw Software is headquartered in San Diego, California.

### **2.6.2 HARDWARE REQUIREMENTS**

The selection of hardware configuring is a very task related to the software development, particularly inefficient RAM may affect adversely on the speed and corresponding on the efficiency of the entire system. The processor should be powerful to handle all the operations. The hard disk should have the sufficient to solve the database and the application. Hardware used for development:

CPU	: Intel i5 Processor
Memory	: 4 GB
Cache	: 6 MB
Hard Disk	: 1 TB
Monitor	: 15.6" Monitor
Keyboard	: Standard108 keys Enhanced Keyboard
Mouse	: Optical Mouse

Minimum Hardware Required For Implementation:

CPU	: Pentium IV Processor
Memory	: 256MB Above

Cache : 512 KB Above

Hard Disk : 20 GB Above

Monitor : Any

Keyboard : Any

Mouse : Any

## 3. SYSTEM DESIGN

### 3.1 SYSTEM ARCHITECTURE

A system architecture or system's architecture is the conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures of the system,

System architecture can comprise system components, the externally visible properties of those components, the relationships (e.g. the behaviour) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system. There have been efforts to formalize languages to describe system architecture; collectively these are called architecture description languages (ADLs).

The system architecture can best be thought of as a set of representations of an existing (or to be created) system. It is used to convey the informational content of the elements comprising a system, the relationships among those elements, and the rules governing those relationships. The architectural components and set of relationships between these components that architecture describes may consist of hardware, software, documentation, facilities, manual procedures, or roles played by organizations or people. System architecture is primarily concerned with the internal interfaces among the system's components or subsystems, and the interface between the system and its external environment, especially the user.

The structural design reduces complexity, facilitates change and result in easier implementation by encouraging parallel development of different parts of the system. The procedural design transforms structural elements of program architecture into a procedural description of software components. The architectural design considers architecture as the most important functional requirement. The system is based on the three-tier architecture.

The first level is the user interface (presentation logic), which displays controls, receives and validates user input. The second level is the business layer (business logic) where the

application specific logic takes place. The third level is the data layer where the application information is stored in files or database. It contains logic about to retrieve and update data. The important feature about the threetier design is that information only travels from one level to an adjacent level.

### **3.2. MODULE DESIGN**

Modular programming is a software design technique that emphasizes separating the functionality of a program into independent, interchangeable modules, such that each contains everything necessary to execute only one aspect of the desired functionality. Conceptually, modules represent a separation of concerns, and improve maintainability by enforcing logical boundaries between components.

Different modules of this project include.

#### **1. Authentication**

This module allows the administrator and the user to login to our website. Admin can log into the system by using their corresponding username and password. Admin can perform all the actions in system only after login. Doctors and Patients login into the system by using their corresponding name and password. After when Doctor and Patient is logged in they are able to access the details and connect to the system. This module allows the Admin, Doctors and Patients to change password and also to reset their passwords if they forgot it.

#### **2. Registration**

This module contains all the registration process in the system. This module includes the registrations that can performed by all stake holders. i.e. Admin, Doctors and Patients. Admin can do the doctor registration, department registration ... And the Patients can also register into the website. The Doctors and Patients are needed to be registered into the website. Only the registered Patients can make the booking. While registering he needs to provide basic information including Name, Address, phone number, user id, password etc. This will keep inside the database that Admin can use this data. The admin has the

permission to approve or reject the user's registration. After registration he can view all the information inside the site. This registered information is helpful to create meaningful information's. Registration have many sub modules like department registration, doctor registration, district registration, location registration.

### **3. Activity**

This module includes the viewing and booking of tokens according to the patient's needs. The patients can see the doctor's profile their experience and other related details and can book for that particular doctors conveniently. The patients have many options to select the doctor they want. Firstly, the users have to login to the site using username and password. Patients can select their visiting time as per their need. They should book by giving their personal details and phone number. If the admin approve the booking the customer should pay consulting fee. The admin can see the booking details of a patient. Doctors can also login to the site using username and password. They can also update their profile by editing or deleting data in it. Admin can monitor all the things that happening in the site. After the visit of the patient the doctor can add a case history and test details of the patient if needed.

### **4. Reports**

This module allows the admin to view the details of the site through various reports. This includes pie chart, excel, PDF and some table information. The reports provide the valuable information from the system. In our system all reports are important because the system eliminate the use of manipulation of the paper work. The system generates useful information's. Admin can generate many reports such that list of department, list of doctors and list of patients etc... So that the Admin can easily manage the website. Admin can also generate the report include the list of customers who made more orders, list of the patients who registered for doctors, list of places to which most of the orders was made, list of the food which is order mostly in a week, and also these can be represent in a pie chart. This will helpful for the admin to generate useful information. The reports show the significance of the system.

### 3.3. DATABASE DESIGN

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make information access easy, quick, inexpensive and flexible for the users. The general theme behind a database is to integrate all information. Database design is recognized as a standard of management information system and is available virtually for every computer system.

In database design several specific objectives are considered:

- Ease of learning and use
- Controlled redundancy
- Data independence
- More information at low cost
- Accuracy and integrity
- Recovery from failure
- Privacy and security
- Performance

A database is an integrated collection of data and provides centralized access to the data. Usually, the centralized data managing the software is called RDBMS. The main significant difference between RDBMS and other DBMS is the separation of data as seen by the program and data has in direct access to stores device. This is the difference between logical and physical data.

### 3.3.1 Normalization

Designing a database is complete task and the normalization theory is a useful aid in the design process. The process of normalization is concerned with transformation of conceptual schema into computer representation form. There will be need for most databases to grow by adding new attributes and new relations. The data will be used in new ways. Tuples will be added and deleted. Information stored may undergo updating also. New association may also be added. In such situations the performance of a database is entirely depend upon its design. A bad database design may lead to certain undesirable things like:

- Repetition of information
- Inability to represent certain information
- Loss of information

To minimize these anomalies, Normalization may be used. If the database is in a normalized form, the data can be growing without, in most cases, forcing the rewriting application programs. This is important because of the excessive and growing cost of maintaining an organization's application programs and its data from the disrupting effects of database growth. As the quality of application programs increases, the cost of maintaining the without normalization will rise to prohibitive levels. A normalized database can also encompass many related activities of an organization thereby minimizing the need for rewriting the applications of programs. Thus, normalization helps one attain a good database design and there by ensures continued efficiency of database.

Normalization theory is built around the concept of normal forms. A relation is said to be in normal form if it satisfies a certain specified set of constraints. For example, a relation is said to be in first normal form (1NF) if it satisfies the constraint that it contains atomic values only. Thus, every normalized relation is in 1NF. Numerous normal forms have been defined. Codd defined the first three normal forms.

All normalized relations are in 1NF, some 1NF relations are also in 2NF and some 2NF relations are also in 3NF. 2NF relations are more desirable than 1NF and 3NF are more desirable than 2NF. That is, the database designer should prefer 3NF than 1NF or 2NF. Normalization procedure states that a relation that is in some given normal form can be converted into a set of relations in a more desirable form. We can define this procedure as the successive reduction of a given collection of relations to some more desirable form. This procedure is reversible. That is, it is always possible to take the output from the procedure and convert them back into input. In this process, no information is lost. So it is also called “no loss decomposition”.

### **First Normal Form**

A relation is in first normal form (1NF) if and all its attributes are based on single domain. The objective of normalizing a table is to remove its repeating groups and ensure that all entries of the resulting table have at most single value.

### **Second Normal Form**

A table is said to be second Normal Form (2NF), when it is in 1NF and every attribute in record is functionally dependent upon the whole key, and not just a part of the key.

### **Third Normal Form**

A table is in third Normal Form (3NF), when it is in 2NF and every non-key attribute is functionally dependent on just the primary key.

#### **3.3.2 Table Structure**

Table is a collection of complete details about a particular subject. These data are saved in rows and Columns. The data of each Row are different units. Hence, rows are called RECORDS and Columns of each row are called FIELDS.

Data is stored in tables, which is available in the backend the items and data, which are entered in the input, form id directly stored in this table using linking of database. We can link more than one table to input forms. We can collect the details from the different tables to display on the output.

There are mainly 9 tables in the project. They are,

1. tbllogin
2. tbldepartment
3. tbldistrict
4. tbllocation
5. tbldoctorreg
6. tblpatientreg
7. tblbook
8. tblcasehistory
9. tbltest
10. tblpayment

---

## TABLE STRUCTURE

### 1. Table Name :tbllogin

Description:To store login details.

Table 3.1 tbllogin

Field Name	Data Type	Constraint	Description of Field
loginid	Int(11)	Primary key	To store uniquely identify each stakeholder's and the values are automatically generated
username	Varchar(30)	Unique	To store the username of the stakeholder
password	Varchar(20)	Not null	To store the password of the stakeholder
role	Varchar(10)	Not null	This field is used to store role of each stakeholder
status	Varchar(20)	Not null	This fields identify the log on status of each stakeholder

---

## 2.Table Name :tbldepartment

Description:To store department details.

Table 3.2 tbldepartment

Field Name	Data Type	Constraint	Description of Field
departmentid	Int(11)	Primary key	To store uniquely identify each department and the values are automatically generated
department	Varchar(30)	Not null	To store department name
image	Varchar(100)	Not null	To store department image

## 3.Table Name :tbldistrict

Description:To store district details.

Table 3.3 tbldistrict

Field Name	DataType	Constraint	Description of Field
districtid	Int(11)	Primary key	To store uniquely identify each district and the values are automatically generated
districtname	Varchar(20)	Not null	To store the district name

## 4.Table Name :tbllocation

Description:To store location details.

Table 3.4 tbllocation

Field Name	Data Type	Constraint	Description of Field
Locationid	int(11)	Primary key	To store uniquely identify each location and the values are automatically generated
locationname	Varchar(30)	Not null	To store the location name
districtid	Int(11)	Not null	To store the district id

**5.Table Name :tbldoctorreg**

Description:To store doctor details.

Table 3.5 tbldoctorreg

Field Name	Data Type	Constraint	Description of Field
doctorid	Int(11)	Primary key	To store uniquely identify each doctor and the values are automatically generated
doctornoame	Varchar(30)	Not null	To store doctor's name
departmentid	Int(11)	Foreign key	Used to take reference from tbldepartment
specialization	Varchar(30)	Not null	The area which the doctor concentrated in
qualification	Varchar(30)	Not null	To store the qualification of the doctor
OPdays	Varchar(10)	Not null	To store number of OP days
Patientcount	Int(11)	Not null	To store the total number of patients consulting per day
consultingfee	float	Not null	To store the consultingfee of doctor
OPtime	Varchar(20)	Not null	To store OP time
Photo	Varchar(100)	Not null	To store image of the doctor
loginid	Int(11)	Foreign key	Used to take reference fromtbllogin
gender	Varchar(10)	Not null	To store the gender of doctor
regdate	Date	Not null	To store the registrationdate of the doctor
experience	Varchar(max)	Not null	To Store the experience of the doctor
phonenumer	Bigint(20)	Not null	To store the phone number of the doctor

**6.Table Name :tblpatientreg**

Description:To store patient details.

Table 3.6 tblpatientreg

Field Name	Data Type	Constraint	Description of Field
patientid	Int(11)	Primary key	To store uniquely identify each doctor and the values are automatically generated
patientname	Varchar(20)	Not null	To store the patient name
gender	Varchar(10)	Not null	To store the gender patient
age	Int(11)	Not null	To store the age of patient
regdate	Date	Not null	To store registration date of patient
phone	Bigint(20)	Not null	To store the phone number of patient
email	Varchar(30)	Not null	To store the email id of patient
locationid	Int(11)	Not null	To store the location id of the patient
place	Varchar(20)	Not null	To store the place of the patient
pin	Bigint(20)	Not null	To store the pincode of the patient
loginid	Int(11)	Foreign key	Used to take reference from tbllogin
OPnumber	Varchar(10)	Unique	To store the op number of the patient

## 7.Table Name:tblbook

Description:To store booking details.

Table 3.7 tblbook

Field Name	Data Type	Constraint	Description of Field
bookid	Int(11)	Primary key	To store uniquely identify each booking and the values are automatically generated
doctorid	Int(11)	Foreign key	To store the doctor id
bookdate	Date	Not null	To store the book date
requestdate	Date	Not null	To store the request date
patientid	Int(11)	Foreign key	Used to take reference from tblpatientreg
status	Varchar(50)	Not null	To check whether the user is admin or customer
tokennumber	Int(11)	Not null	To store the token number

---

### 8.Table Name : tblcasehistory

Description: To store the case details of patients

Table 3.8 tblcasehistory

Field Name	Data Type	Constraint	Description of Field
casehistoryid	Int(11)	Primary key	To store uniquely identify each casehistory and the values are automatically generated
OPnumber	Varchar(30)	Not null	To store the OP number
symptoms	Varchar(200)	Not null	To store the symptoms
prescription	text	Not null	To store the prescriptions
description	Varchar(100)	Not null	To store the description
date	Date	Not null	To store the date
did	Int(11)	Foreign key	Used to take reference from tbldoctorreg

---

### 9.Table Name : tbltest

Description: To store the test details of patient

Table 3.9 tbltest

Field Name	Data Type	Constraint	Description of Field
testid	Int(11)	Primary key	To store uniquely identify each test and the values are automatically generated
testname	Varchar(50)	Not null	To store the testname
testdescription	Varchar(60)	Not null	To store the testdescription
casehistoryid	Int(11)	Foreign key	Used to take reference from tblcasehistory
OPnumber	Varchar(50)	Foreign key	Used to take reference from tblpatientreg
result	Varchar(90)	Not null	To store the result

---

## 10.Table Name : tblpayment

Description: To store the payment details of the patients

Table 3.10 tblpayment

Field Name	Data Type	Constraint	Description of Field
paymentid	Int	Primary key	To store uniquely identify each stakeholder's and the values are automatically generated
Paymentdate	Date	Date	To store the paymentdate of the patient
bookid	Int	Foreign key	Used to take reference from tblbook
doctorid	Int	Foreign key	Used to take reference from tbldoctorreg
amount	Int	Foreign key	Used to take reference from tbldoctorreg
status	Varchar(20)	Not Null	This fields identify the payment status of each patient

### 3.3.3 Data Flow Diagram

#### 3.3.3.1 Introduction to Data Flow Diagrams

Data Flow Diagram is a network that describes the flow of data and processes that change, or transform, data throughout the system. This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. Data in files may also flow to processes as inputs.

There are various symbols used in a DFD. Bubbles represent the processes. Named arrows indicate the data flow. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. Each component in a DFD is labelled with a descriptive name. Process names are further identified with a number.

The Data Flow Diagram shows the logical flow of a system and defines the boundaries of the system. For a candidate system, it describes the input (source), outputs (destination), database (files) and procedures (data flow), all in a format that meet the user's requirements.

The main merit of DFD is that it can provide an overview of system requirements, what data a system would process, what transformations of data are done, what files are used, and where the results flow.

This network is constructed by use a set of symbols that do not imply a physical implementation. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities from which data flows to a process, which transforms the data and creates, output-data-flows which go to other processes or external entities or files. External entities are represented by rectangles. Entities supplying data are known as sources and those that consume data are called sinks. Data are stored in a data store by a process in the system. It is a graphical tool for structured analysis of the system requirements. DFD models a system by using external entities

---

from which data flows to a process, which transforms the data and creates, output dataflows which go to other processes or external entities or files. Data in files may also flow to processes as inputs.

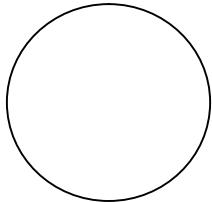
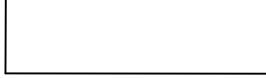
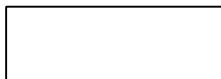
### Rules for a Data Flow Diagram

1. Arrows should not cross each other
2. Squares, circles and files must bear names.
3. Decomposed data flow squares and circles can have same time
4. Choose meaningful names for data flow
5. Draw all data flows around the outside of the diagram

### Basic Data Flow Diagram Symbols

Table 3.11 Data Flow Diagram Symbols

	A data flow is a route, which enables packets of data to travel from one point to another. Data may flow from a source to a process and from data store or process. An arrow line depicts the flow, with arrow head pointing in the direction of the flow.
---	--

	Circles stands for process that converts data in to information. A process represents transformation where incoming data flows are changed into outgoing data flows.
	A data store is a repository of data that is to be stored for use by a one or more process may be as simple as buffer or queue or sophisticated as relational database. They should have clear names. If a process merely uses the content of store and does not alter it, the arrowhead goes only from the store to the process. If a process alters the details in the store, then a doubleheaded arrow is used.
	A source or sink is a person or part of an organization, which enters or receives information from the system, but is considered to be outside the contest of data flow model.

### 3.3.3.2 Data Flow Diagram

Each component in a DFD is labelled with a descriptive name. Process name are further identified with number. Context level DFD is draw first. Then the process is decomposed into several elementary levels and is represented in the order of importance. A DFD describes what data flow (logical) rather than how they are processed, so it does not depend on hardware, software, and data structure or file organization.

A DFD methodology is quite effective; especially when the required design.

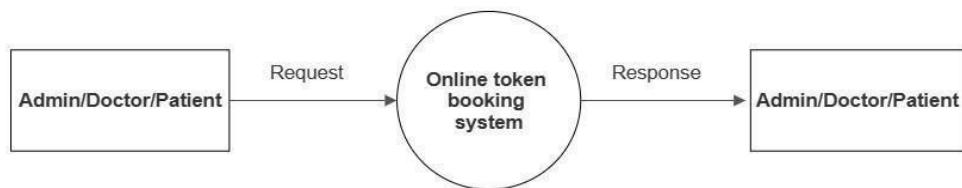


Fig 3.1 Zeroth level DFD for Online Token Booking System

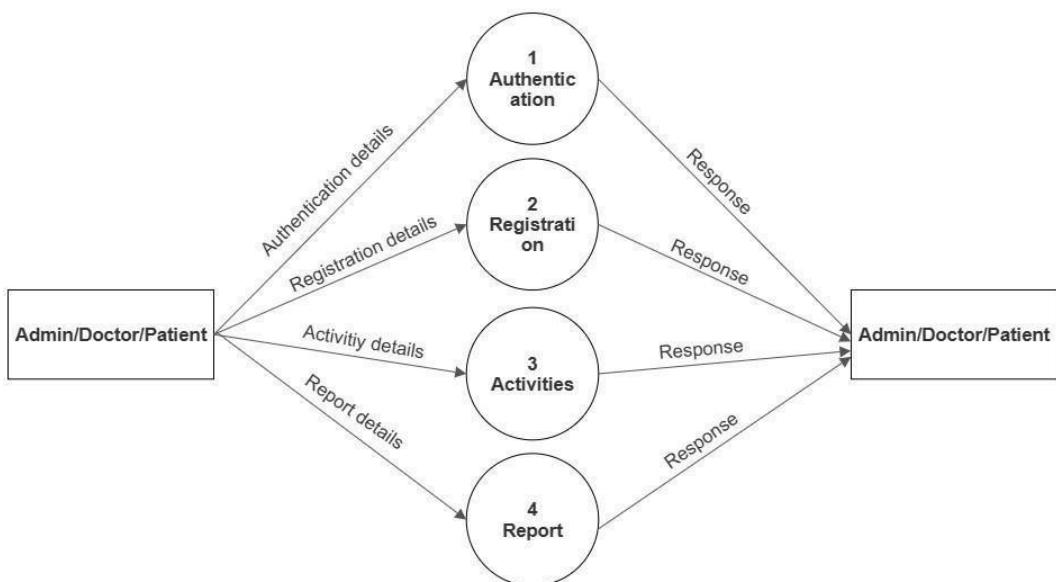


Fig 3.2 First Level DFD for Online Token Booking System

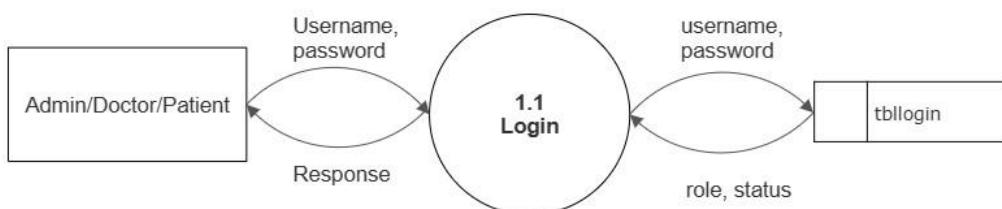


Fig. 3.3 Second level DFD for login

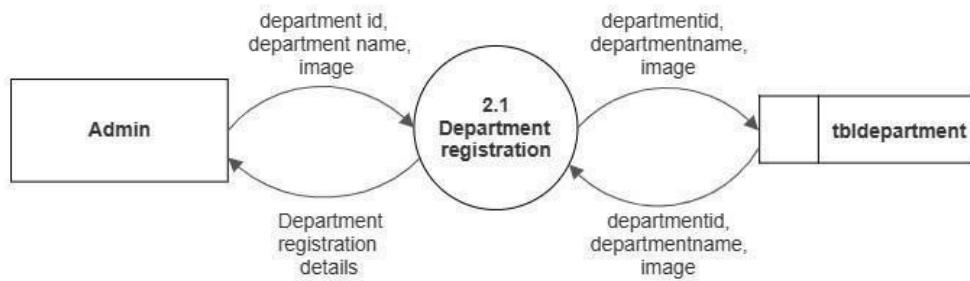


Fig. 3.4 Second level DFD for Department registration

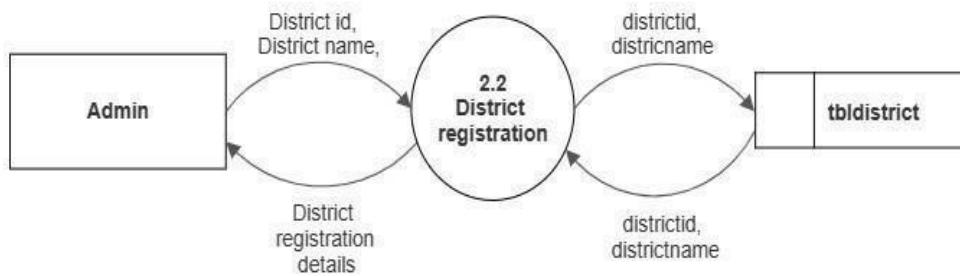


Fig. 3.5 Second level DFD for District registration

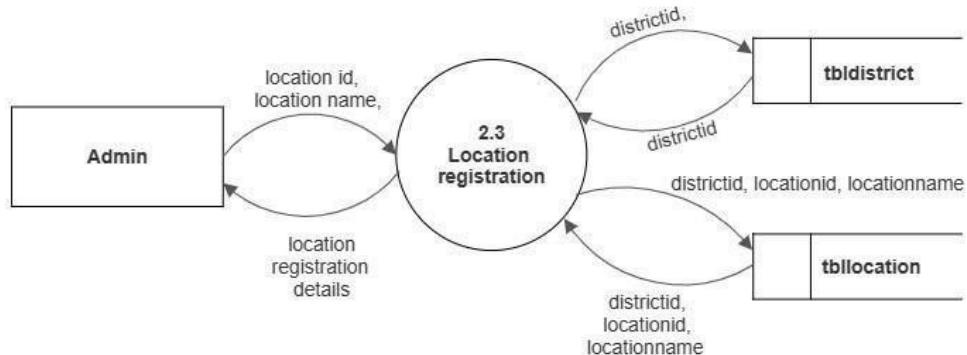


Fig. 3.6 Second level DFD for location registration

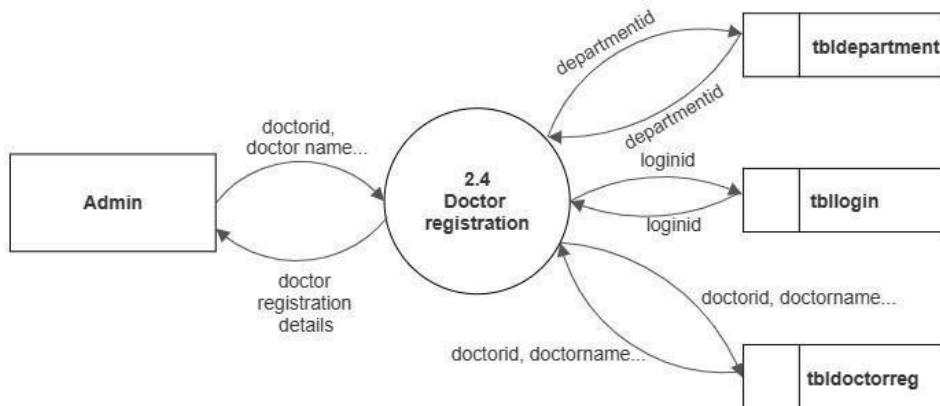


Fig. 3.7 Second level DFD for Doctor registration

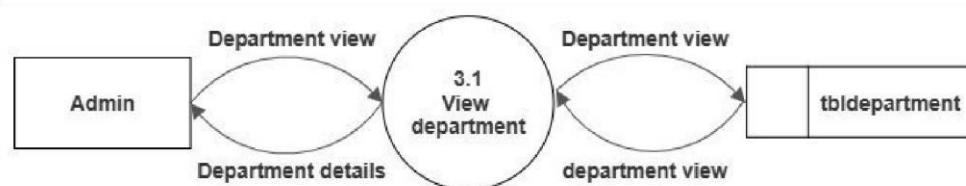


Fig. 3.8 Second level DFD for Department view

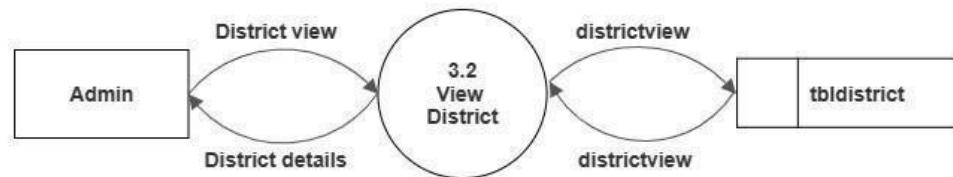


Fig. 3.9 Second level DFD for District view

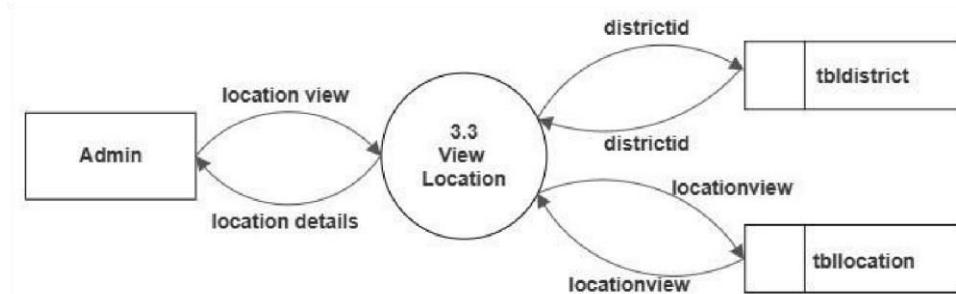


Fig. 3.10 Second level DFD for Location view

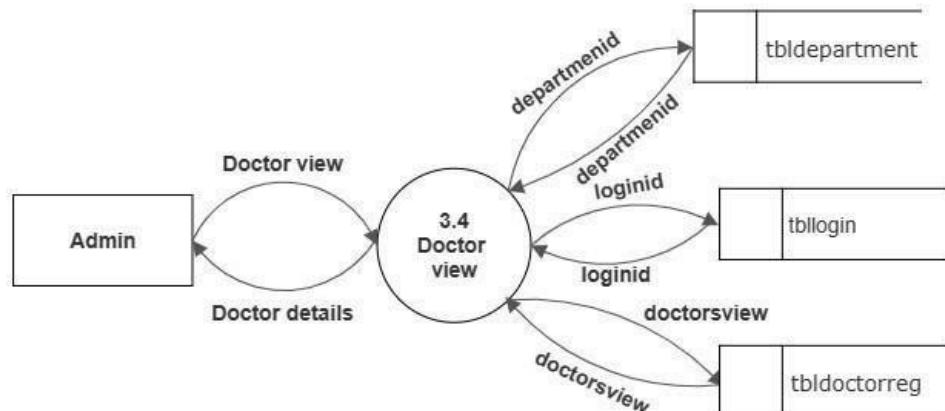


Fig. 3.11 Second level DFD for Doctors view

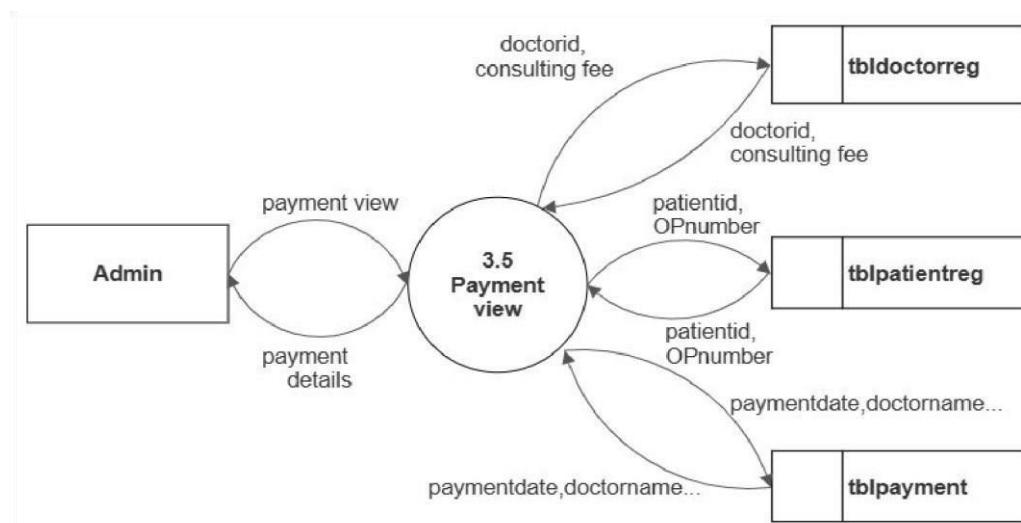


Fig 3.12 Second level DFD for Patient payment view

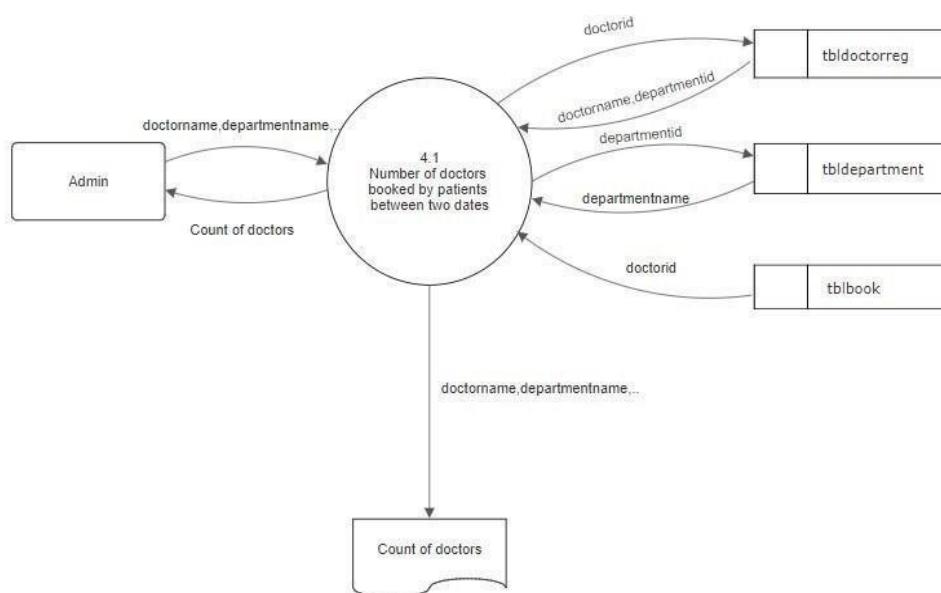


Fig 3.13 Second level DFD for Datewise by Count of Patients in each Doctor Report

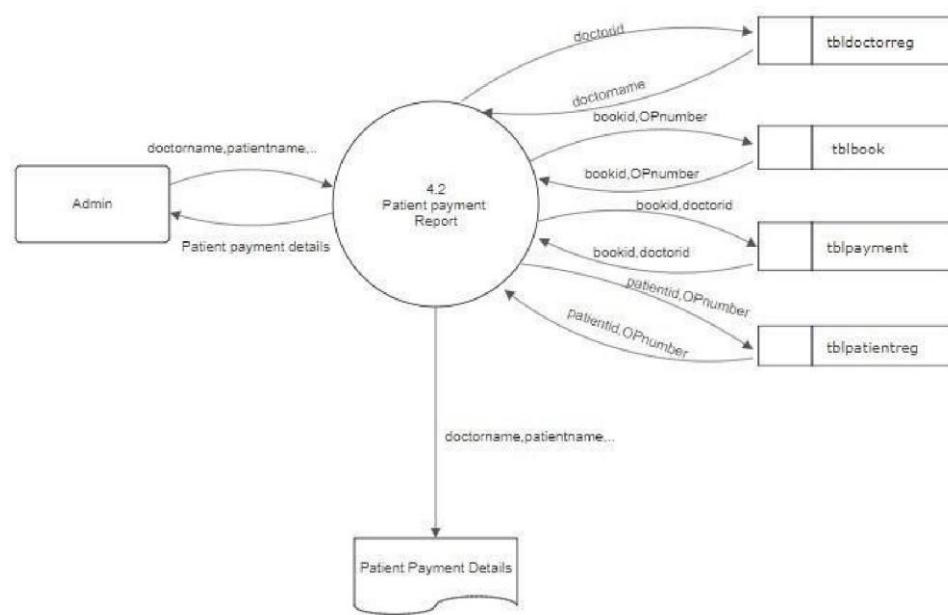


Fig 3.14 Second level DFD for Patient Payment Report

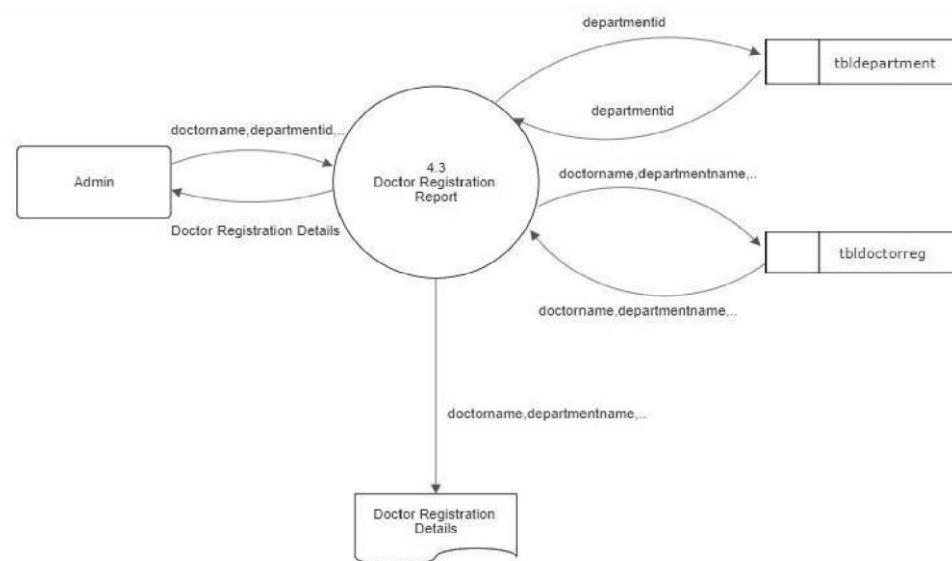


Fig 3.15 Second level DFD for Doctor Registration Report

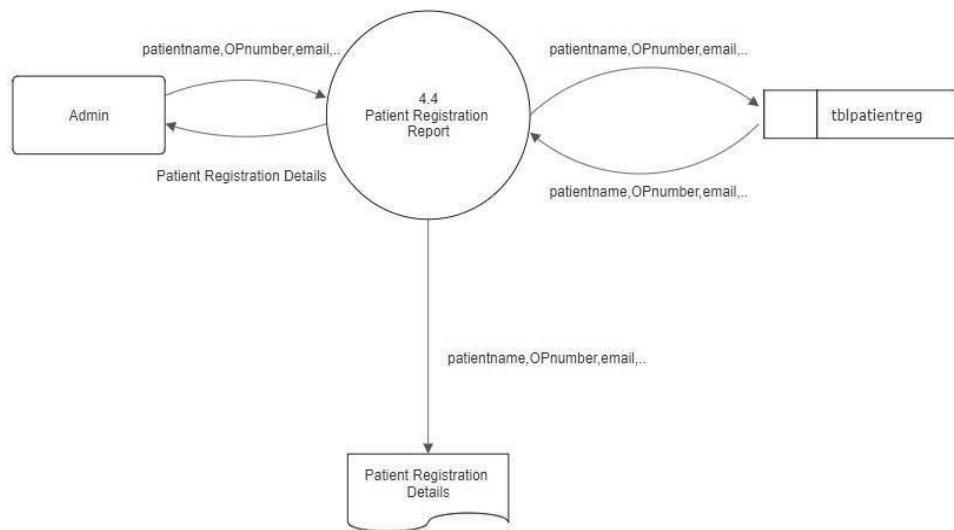


Fig 3.16 Second level DFD for Patient Registration Report

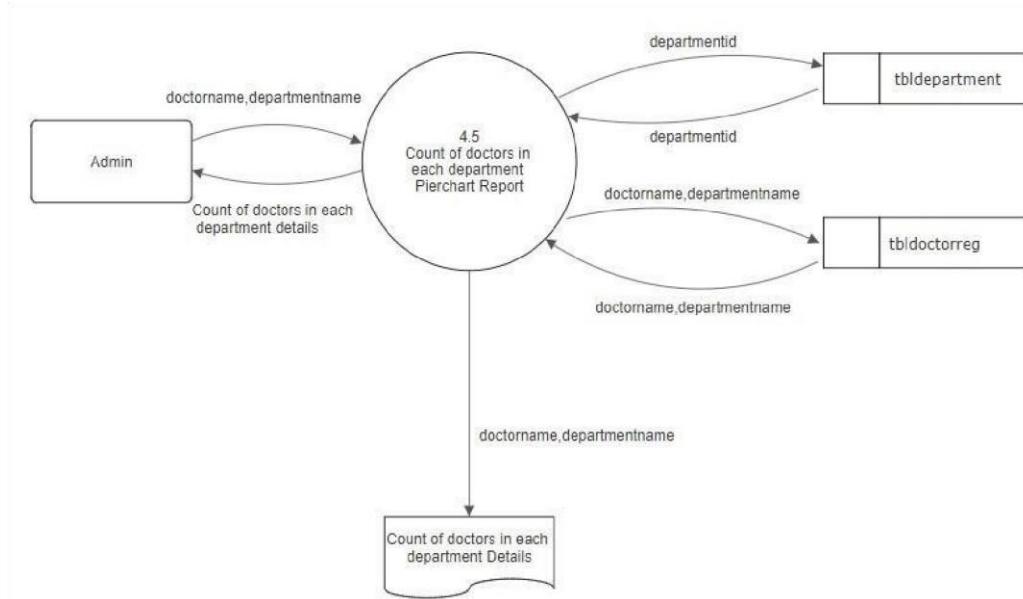


Fig 3.17 Second level DFD for Count of Doctors in each Department Report

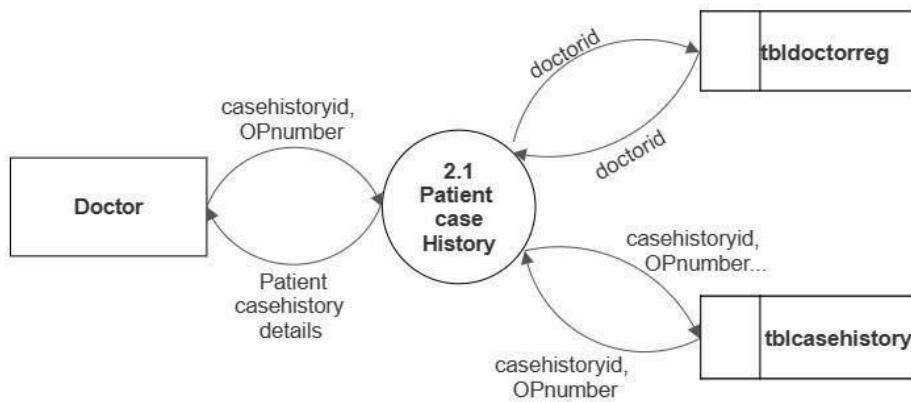


Fig. 3.18 Second level DFD for Case details of Patient

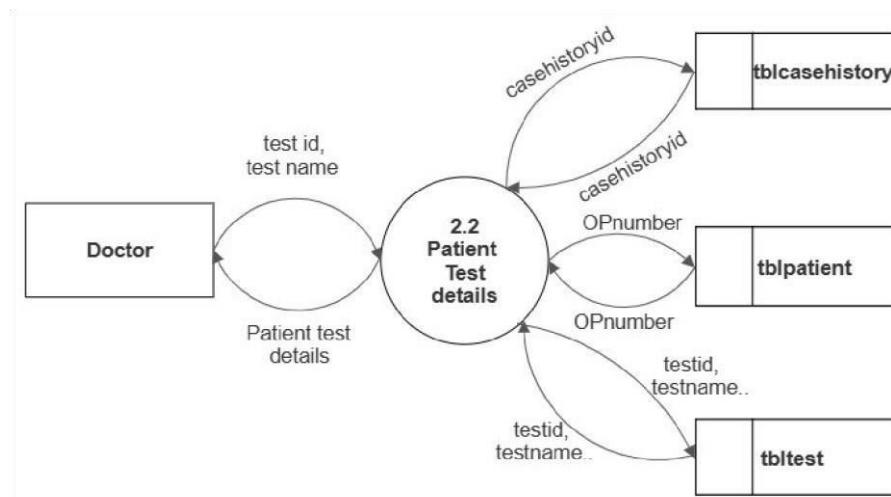


Fig. 3.19 Second level DFD for Test details of Patient

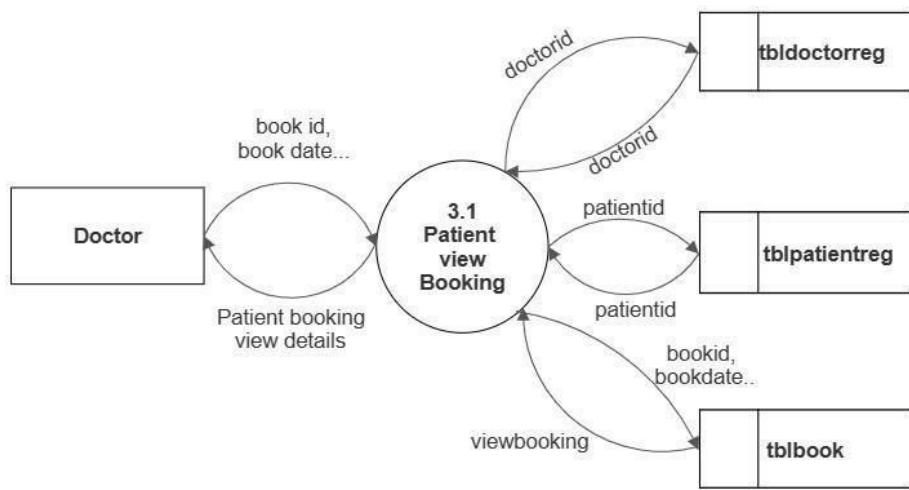


Fig. 3.20 Second level DFD for Doctor activities

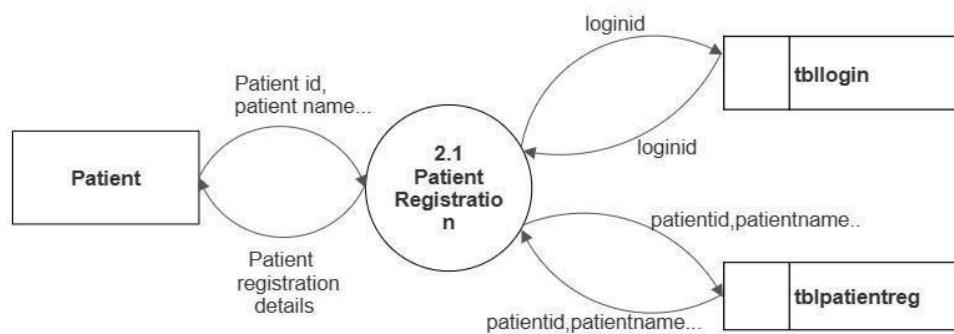


Fig. 3.21 Second level DFD for Patient registration

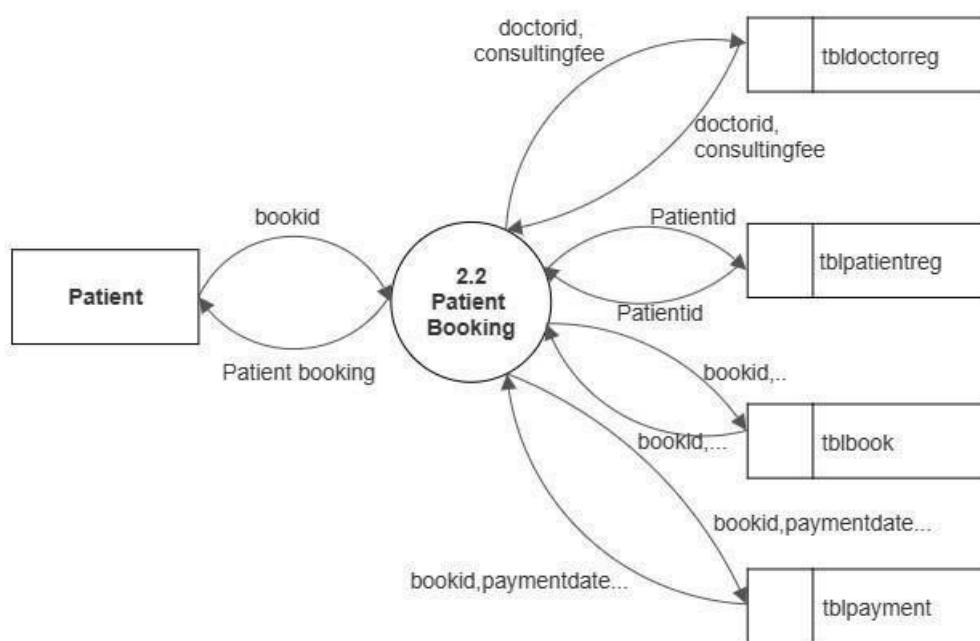


Fig. 3.22 Second level DFD for Patient booking registration

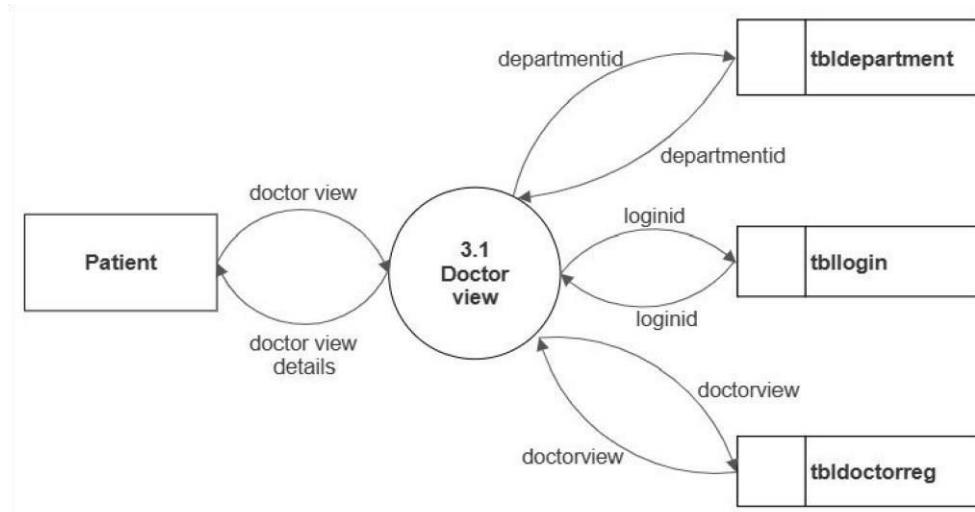


Fig. 3.23 Second level DFD for Patient registration

### 3.4 INTERFACE DESIGN

These modules can apply to hardware, software or the interface between a user and a machine. An example of a user interface could include a GUI, a control panel for a nuclear power plant, or even the cockpit of an aircraft. In systems engineering, all the inputs and outputs of a system, subsystem, and its components are listed in an interface control document often as part of the requirements of the engineering project. The development of a user interface is a unique field.

#### 3.4.1 User Interface Screen Design

The user interface design is very important for any application. The interface design describes how the software communicates within itself, to system that interpreted with it and with humans who use it. The input design is the process of converting the user-oriented inputs into the computer based format. The data is fed into the system using simple inactive forms. The forms have been supplied with messages so that the user can enter data without facing any difficulty. They data is validated wherever it requires in the project. This ensures that only the correct data have been incorporated into system. The goal of designing input data is to make the automation as easy and free from errors as possible. For providing a good input design for the application easy data input and selection features are adopted. The input design requirements such as user friendliness, consistent format and interactive dialogue for giving the right messages and help for the user at right are also considered for development for this project.

Input Design is a part of the overall design. The input methods can be broadly classified into batch and online. Internal controls must be established for monitoring the number of inputs and for ensuring that the data are valid. The basic steps involved in input design are:

- Review input requirements.
- Decide how the input data flow will be implemented.
- Decide the source document.
- Prototype on line input screens.
- Design the input screens.

The quality of the system input determines the quality of the system output. Input specifications describe the manner in which data enter the system for processing.

Input design features can ensure the reliability of the system and produce results from accurate data. The input design also determines whether the user can interact efficiently with the system.

These is a sample input forms:

The screenshot shows a web-based doctor registration form titled "Doctor Registration". The URL in the address bar is "localhost/eclinicnew/admin/doctorreg.php". The form includes fields for "Doctor Name" (with a placeholder "Doctor Name"), "Department Name" (with a dropdown menu), "Image" (with a "Choose File" button and a note "No file chosen"), "Specialization" (text input), "Qualification" (text input), "OP Day" (text input), "Total Count" (text input), "OP Time" (text input), "Phone Number" (text input with placeholder "phone number"), "Gender" (radio buttons for Male and Female), "Registration Date" (text input with value "21-10-2023"), "Experience" (text input), "ConsultingFee" (text input), "Amount" (text input), "Username" (text input), "Password" (text input), and a "Remember me" checkbox. At the bottom right is a blue "Register" button.

Fig 3.24 Doctor registration form

This input form is for doctor registration. It contains textboxes for inputting Doctor name, Specialization, Qualification, OP days, Total count, OP time, Phone number, Experience, Consulting fee, Username and Password . The form also gives a provision for the user to select their Gender and Department name from the select box. After clicking the Create button the doctor has successfully created their profile in the website. The doctor registration form is very important in the project. Each doctor must have to fill the full details that are given in the form to register into the system and log in to it. Each field have its own label that denotes the value need to enter in that box. Also, each textbox has placeholders which helpful for the user to decide the type of value which need to enter in the box. The form also has a button that allows the user to pass the contents entered in the form to the database table. The data entered in the form should be correct according to the type of that field. All labels are arranged in the same alignment line and all boxes to enter values are also in the same line.

### **3.4.2 Output Design**

A quality output is one, which meets the requirements of end user and presents the information clearly. In any system result of processing are communicated to the user and to the other system through outputs. In the output design it is determined how the information is to be displayed for immediate need.

It is the most important and direct source information is to the user. Efficient and intelligent output design improves the system's relationships with the user and helps in decision making. The objective of the output design is to convey the information of all the past activities, current status and to emphasize important events. The output generally refers to the results and information that is generated from the system. Outputs from computers are required primarily to communicate the results of processing to the users.

Output also provides a means of storage by copying the results for later reference in consultation. There is a chance that some of the end users will not actually operate the input data or information through workstations, but will see the output from the system.

Two phases of the output design are:

1. Output Definition
  2. Output Specification
-

Output Definition takes into account the type of output contents, its frequency and its volume, the appropriate output media is determined for output. Once the media is chosen, the detail specification of output documents are carried out. The nature of output required from the proposed system is determined during logical design stage. It takes the outline of the output from the logical design and produces output as specified during the logical design phase.

In a project, when designing the output, the system analyst must accomplish the following:

- Determine the information to present.
- Decide whether to display, print, speak the information and select the output medium.
- Arrange the information in acceptable format.
- Decide how to distribute the output to the intended receipt. Thus, by following the above specifications, a high-quality output can be generated In our projects outputs are generated as pie charts of the patients who have booked each doctors.
- 

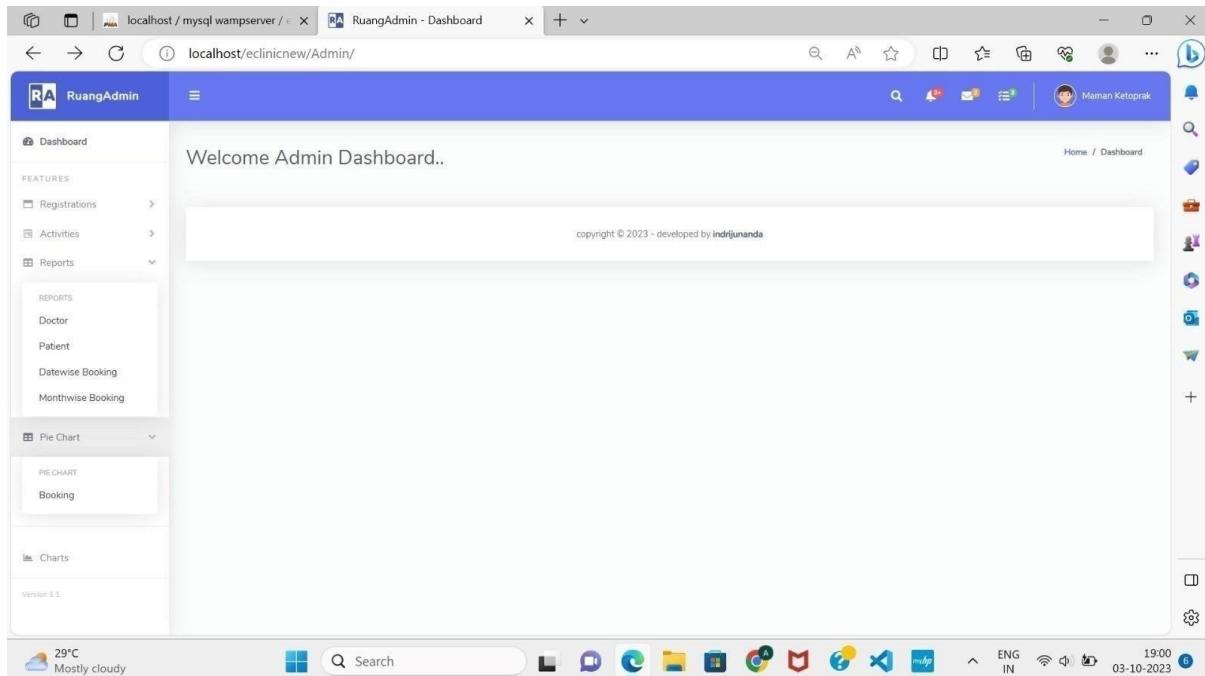


Fig 3.25 Button that redirects to pie chart

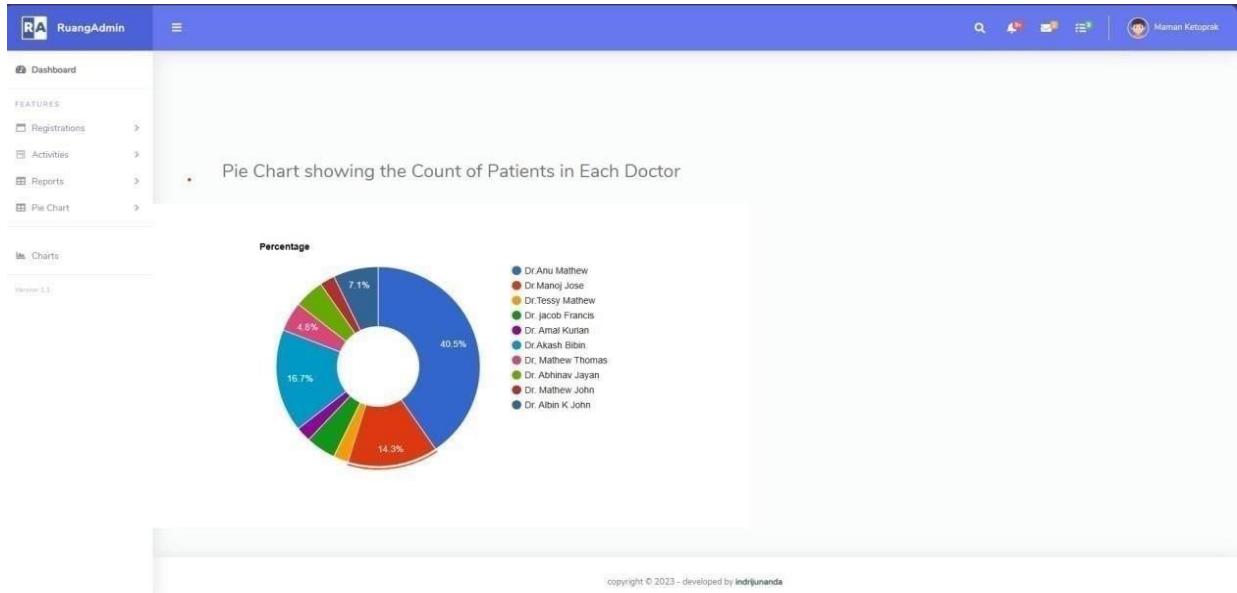


Fig 3.26 Pie charts of the patients who have booked each doctors.

This is a pie chart report which gives the percentage of which doctor got the greatest number of bookings. So it makes easier for the Admin to examine the doctor mostly booked by the patients. Since this report is in the form of pie chart. Graphical representation easily conveys all details in efficient manner. Here we can see that which doctor is mostly booked by the patients. Dr Anu Mathew is booked around 40.5% of total percent. Dr Manoj Jose is booked around 14.3% of total percent. Dr Tessy Mathew is booked 2.4% of total percent. Dr Jacob Francis is booked around 4.8% of total percent. Dr Amal Kurian is booked around 2.4% of total percent. Dr Akash Bibin is booked around 16.7% of total percent. Dr Mathews Thomas booked around 4.8% of total percent. Dr Abhinav Jayan booked around 4.8% of total percent Dr Mathew John booked around 2.4% of total percent. Dr Albin K John booked around 7.1% of total percent. Dr Anu Mathew is in blue color, Dr Manoj Jose is in red color, Dr Tessy Mathew is in yellow color. Dr Jacob Francis is in green color, Dr Amal Kurian is in Violet color. Dr Akash Bibin is in blue color. Dr Mathew Thomas in magenta color. Dr Abhinav Jayan is in green color. Dr Mathew John is in wine red color. Dr Albin K John is in blue color These color variation makes it more attractive.

## 4. IMPLEMENTATION

Implementation is the stage of the project when the theoretical design is turned into a working system. The implementation stage is a systems project in its own right. It includes careful planning, investigation of current system and its constraints on implementation, design of methods to achieve the changeover, training of the staff in the changeover procedure and evaluation of changeover method

### 4.1. CODING STANDARDS

PHP follows few rules and maintains its style of coding. As there are many coders and developers all over the world, so each of them can follow different coding styles and standards but this would have raised great confusion and difficulty for a developer to understand another developer's code. It would have been hard to manage and store the code for future reference. Here is where the coding standards come into play. This not only makes a code easy to read but also makes the code very easy to refer in the future. This makes the code understandable and clearer to decipher, just like a blueprint. This also makes the code more formal and industry or software oriented. Below mentioned are few guidelines that one must follow in order to maintain the standard of PHP coding.

1. **PHP tags:** One must use the PHP standard tags(), rather than the shorthand tags() to delimit the PHP code.
2. **Commenting:** Use of standard C and C++ commenting style i.e., (//) – for single line and /\* \*/ – for multi-line, is highly encouraged and use of Python or Perl style of commenting i.e., (#), is discouraged.
3. **Line length and Indentation:** It is a standard recommendation to not exceed more than 75-85 characters per line of code. One must not use tabs for indentation instead use 4 spaces as it is the standard indenting method in most of the programming languages.
4. **Structuring the control flow statements:** The control flow or conditional statements must be written in such a way so that it could be differentiated from function call statements. While writing if, for, while, switch and other control flow statements there must be one space between the keyword and the opening parenthesis. Example:  
filter\_none edit play\_arrow brightness\_4

```
<?php $n = 5; if ($n  
> 0){ echo  
"Positive";  
} elseif ($n < 0){ echo  
"Negative";  
} else{ echo  
"Zero";  
}  
?
```

?>

Output:

Positive

5. **Function Calls:** While writing a function call statement, there must be no space between the function name and the opening parenthesis. Example:

```
filter_none edit play_arrow brightness_4  
<?php
```

```
echo testFunc(5, 6);
```

```
function testFunc($num1, $num2) {  
$val = $num1 + $num2; return  
$val;  
}
```

?>

Output:

11

**6. Naming Variables:** Here are few conventions that one must follow in order to name

the variables:

- Use of lower case letters to name the variables.
- Use of ‘\_’ to separate the words in a variable.
- Static variable names may be started with a letter ‘s’.
- Global variable names must start with letter ‘g’.
- Use of upper-case letters to define global constants with ‘\_’ as a separator.

**7. Block alignment:** Every block of code and curly braces must be aligned.

**8. Short Functions:** All functions and methods must limit themselves to a single page and must not be lengthy.

## 4.2. SAMPLE CODE

4.1 registration form

```
<?php include("header.php");
?> <?php include("../dboperation.php");
$obj=new dboperation();
$sql="SELECT * FROM tbldepartment";
$res=$obj->executequery($sql);
?>
```

```
<div class="container-fluid" id="container-wrapper">
<div class="d-sm-flex align-items-center justify-content-between mb- 4">
<h1 class="h3 mb-0 text-gray-800">Doctor Registration</h1>
<ol class="breadcrumb">
<li class="breadcrumb-item"><a href="./">Home</a></li>
<li class="breadcrumb-item">Forms</li>
<li class="breadcrumb-item active" aria-current="page">Doctor Registration</li>
</ol>
</div>

<div class="row">
<div class="col-lg-6">
<!-- Form Basic -->
<div class="card mb-4">
<div class="card-header py-3 d-flex flex-row align-items-center justifycontentbetween">
<h6 class="m-0 font-weight-bold text-primary">Doctor Registration</h6>

<a href="doctorsview.php"><button type="submit" class="btn btnprimary">VIEW
DOCTORS</button></a>
</div>
<div class="card-body">
<form action="doctorregaction.php"
method="POST" name="f1"
enctype="multipart/form-data">
<div class="form-group">
<label for="exampleInputdoctorname">Doctor Name</label>
```

```
<input type="text" class="form-control" name="doctorno" id="exampleInputdoctorno" placeholder="Doctor Name" required> </div>

<div class="form-row">
<div class="form-group col-md-6">
<label for="inputEmail4">Department Name</label>
<select id="ddldep" name="ddldep" class="form-group" required>
<option value="0">--select--</option>
<?php while($display=mysqli_fetch_array($res))
{
?>
<option value="<?php echo $display["departmentid"];?>" required><?php echo $display["departmentname"];?></option >
<?php
}
?>

</select>
</div>
</div>

<div class="form-group">
<label for="photo">Image</label>
<input type="file" class="form-control" id="photo" name="photo" alt="Profile" class="brand-image img-square elevation-3" width="70" style="opacity: .8; border: 1px solid gray" required>
</div>
<div class="form-group">
```

```
<label for="exampleInputspecialization">Specialization</label>
<input type="specialization" class="form-control" name="specialization"
id="exampleInputspecialization" required>
</div>
<div class="form-group">
<label for="exampleInputqualification">Qualification</label>
<input type="qualification" class="form-control" name="qualification"
id="exampleInputqualification" required>
</div>
<div class="form-group">
<label for="exampleInputdays">OP Days</label>
<input type="days" class="form-control" name="OPdays" id="exampleInputdays"
required>
</div>
<div class="form-group">
<label for="exampleInputnumber">Total Count</label>
<input type="number" class="form-control" name="totalcount" id="exampleInputnumber"
required pattern="^[0-9]+\$"
title="Enter only digits">
</div>
<div class="form-group">
<label for="exampleInputtime">OP Time</label>
<input type="text" class="form-control" name="OPtime"
id="exampleInputnumber" required>
</div>
<div class="form-group">
<label for="exampleInputphonenumber">Phone
```

```
Number</label>
<input type="consultingfee" class="form-control" name="phone"
placeholder="phone number"
id="exampleInputconsultingfee" required pattern="[0-9]{10}" required title="Must
contain 10 digits">
</div>

<div class="form-group">
<label for="exampleInputgender">Gender</label> <input type="radio"
name="gender" value="Male" required>Male
<input type="radio" name="gender" value="Female" required>Female </div>
<div class="form-group">
<label for="exampleInputregistrationdate">Registration
Date</label>
<input type="date" id="date" name="date" value="php echo date('Y-m-d'); ?"
readonly="true" required/>
</div>

<div class="form-group">
<label for="exampleInputexperience">Experience</label>
<input type="experience" class="form-control"
name="experience"
id="exampleInputexperience" required>
</div>

<div class="form-group">
<label for="exampleInputconsultingfee">ConsultingFee</label>
<input type="consultingfee" class="form-control"
name="consultingfee" placeholder="Amount"
id="exampleInputconsultingfee" required pattern="^[0-9]+\$" title="Enter only digits">
</div>
```

```
<div class="form-group">
<label for="exampleInputusername">Username</label>
<input type="username" class="form-control" name="username"
id="exampleInputusername" placeholder="Username" pattern="[a-z]{5,15}" required
value="" title="Must contain minimum 5 and maximum 15 characters">
</div>

<div class="form-group">
<label for="exampleInputPassword">Password</label>
<input type="password" class="form-control" name="password"
id="exampleInputPassword" placeholder="Password"
pattern="(?=.*\d)(?=.*[az])(?=.*[A-Z]).{8,}" title="Must contain at least one number and one
uppercase and lowercase letter, and at least 8 or more characters" required value="">

</div>

<div class="form-group">
<div class="custom-control custom-checkbox">
<input type="checkbox" class="custom-control-input" id="customControlAutosizing">
<label class="custom-control-label" for="customControlAutosizing">Remember me</label>
</div>
</div>

<button type="register" name="submit" class="btn
btnprimary">Register</button>
<!-- <input required id="submit" class="button" type="submit" name="Submit">
<button type="submit" name="submit" class="btn btn-primary btn-block">
Register </button> -->
</form>
</div>
</div>
```

```
</div> </div>

<!--Row-->

<!--Container Fluid-->

</div>    <?php include("footer.php");?

?>
```

## 5. TESTING

Coding conventions are a set of guidelines for a specific programming language that recommend programming style, practices and methods for each aspect of a piece program written in this language. These conventions usually cover file organization, indentation, comments, declarations, statements, white space, naming conventions, programming practices, programming principles, programming rules of thumb, architectural best practices, etc. These are guidelines for software structural quality. Software programmers are highly recommended to follow these guidelines to help improve the readability of their source code and make software maintenance easier.

### 5.1 TEST CASES

The objective of system testing is to ensure that all individual programs are working as expected, that the programs link together to meet the requirements specified and to ensure that the computer system and the associated clerical and other procedures work together. The initial phase of system testing is the responsibility of the analyst who determines what conditions are to be tested, generates test data, produced a schedule of expected results, runs the tests and compares the computer produced results with the expected results with the expected results. The analyst may also be involved in procedures testing. When the analyst is satisfied that the system is working properly, he hands it over to the users for testing. The importance of system testing by the user must be stressed. Ultimately it is the user must verify the system and give the go-ahead.

During testing, the system is used experimentally to ensure that the software does not fail, i.e., that it will run according to its specifications and in the way users expect it to. Special test data is input for processing (test plan) and the results are examined to locate unexpected results. A limited number of users may also be allowed to use the system so analysts can see whether they try to use it in unexpected ways. It is preferably to find these surprises before the organization implements the system and depends on it. In many organizations, testing is performed by person other than those who write the original programs. Using persons who do not know how certain parts were designed or programmed ensures more complete and unbiased testing and more reliable software.

Parallel running is often regarded as the final phase of system testing. Since the parallel operation of two systems is very demanding in terms of user resources it should be embarked on only if the user is satisfied with the results of testing - it should not be started

---

if problems are known to exist. Testing is the major quality control measure during software development. Its basic function is to detect errors in the software. Thus the goal of testing is to uncover requirement design and coding errors in the program.

Testing is the process of correcting a program with intends of finding an error.

Different types of testing are,

1. Unit Testing
2. Integrated Testing
3. Black Box Testing
4. White Box Testing
5. Validation Testing
6. User Acceptance Testing

### **5.1.1 Unit Testing**

In computer programming, unit testing is a method by which individual units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures are tested to determine if they are fit for use. In this testing we test each module individual and integrated the overall system. Unit testing focuses verification efforts on the smaller unit of software design in the module. This is also known as module testing. The modules of the system are tested separately. The testing is carried out during programming stage itself. In this testing step each module is found to working satisfactory as regard to the expected output from the module. There are some validation checks for verifying the data input given by the user which both the formal and validity of the entered. It is very easy to find error debug the system.

---

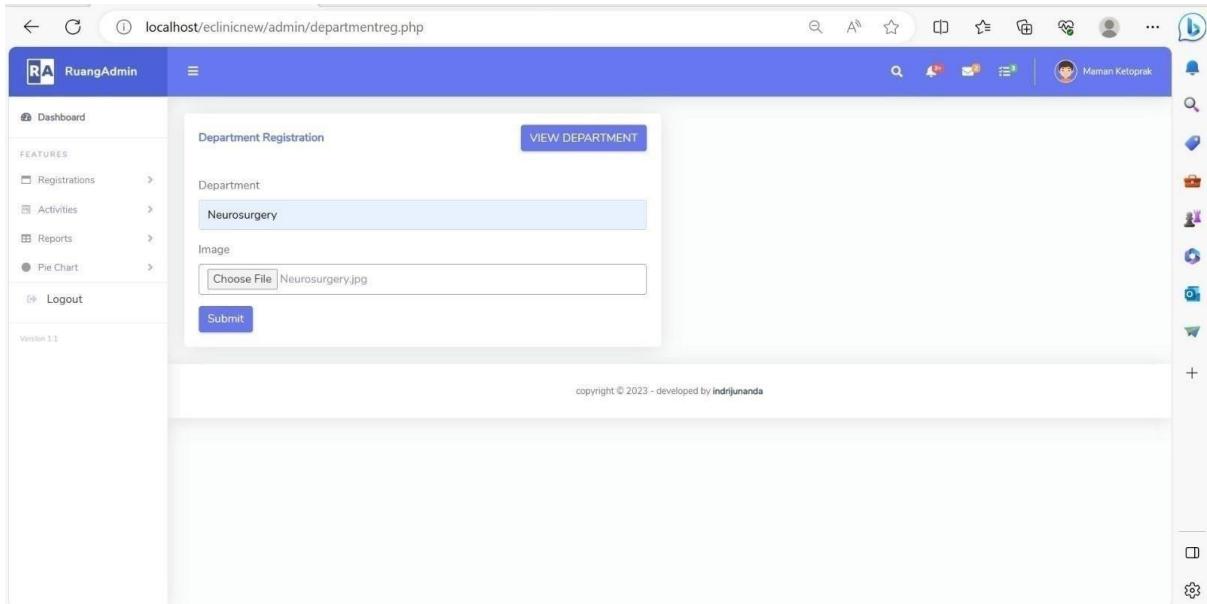


Fig 5.1 Unit testing

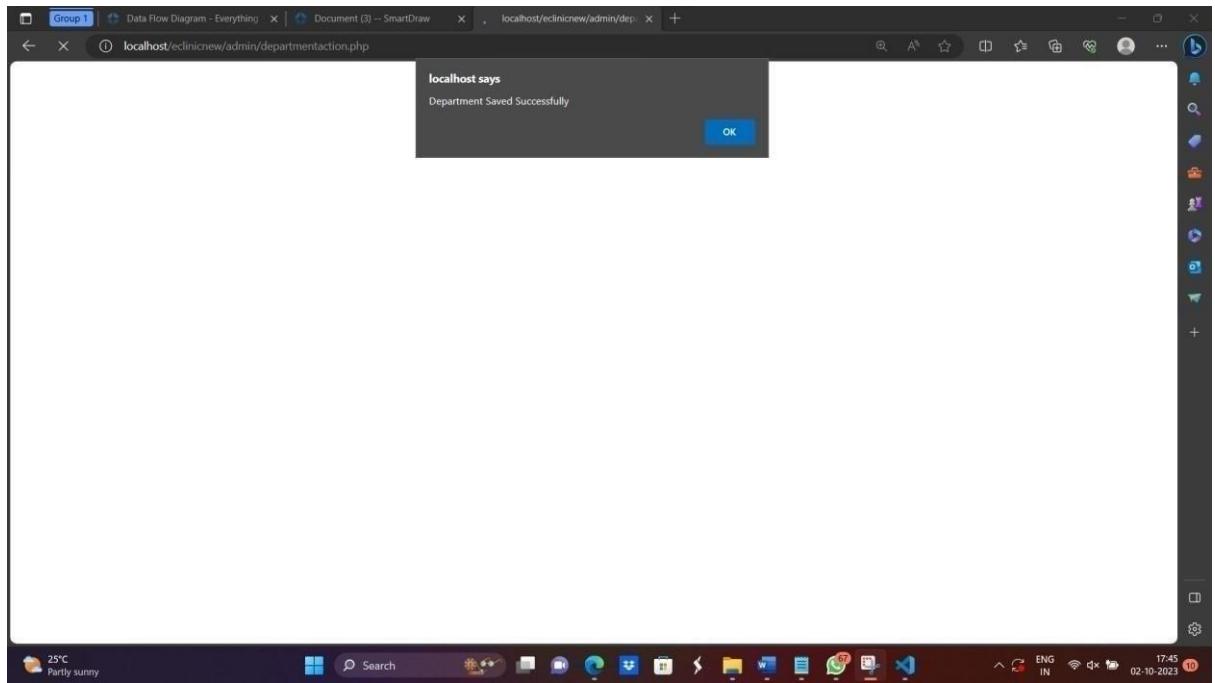


Fig 5.2 Unit testing result

We have continued Unit Testing from the starting of the coding phase itself. Whenever we completed one small sub module, some amount of testing was done based on the requirements to see if the functionality is aligned to the gathered requirements.

### 5.1.2 Integration Testing

Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group. Software components may be integrated in an iterative way or all together ("big bang"). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed. Data can be lost across an interface; one module can have an adverse effect on the other sub functions when combined by, may not produce the desired major functions. Integrated testing is the systematic testing for constructing the uncover errors within the interface. This testing was done with sample data. The developed system has run success full for this sample data. The need for integrated test is to find the overall system performance.

Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested. A component, in this sense, refers to an integrated aggregate of more than one unit. Integration testing identifies problems that occur when units are combined. By using a test plan that requires you to test each unit and ensure the viability of each before combining units, you know that any errors discovered when combining units are likely related to the interface between units. This method reduces the number of possibilities to a far simpler level of analysis. Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system.

SLNO	Location Name	Actions
1	Neyyattinkara	Dropdown
2	Parassala	Dropdown
3	Kattakada	Dropdown
4	Vellarada	Dropdown

Fig 5.3 Integration testing

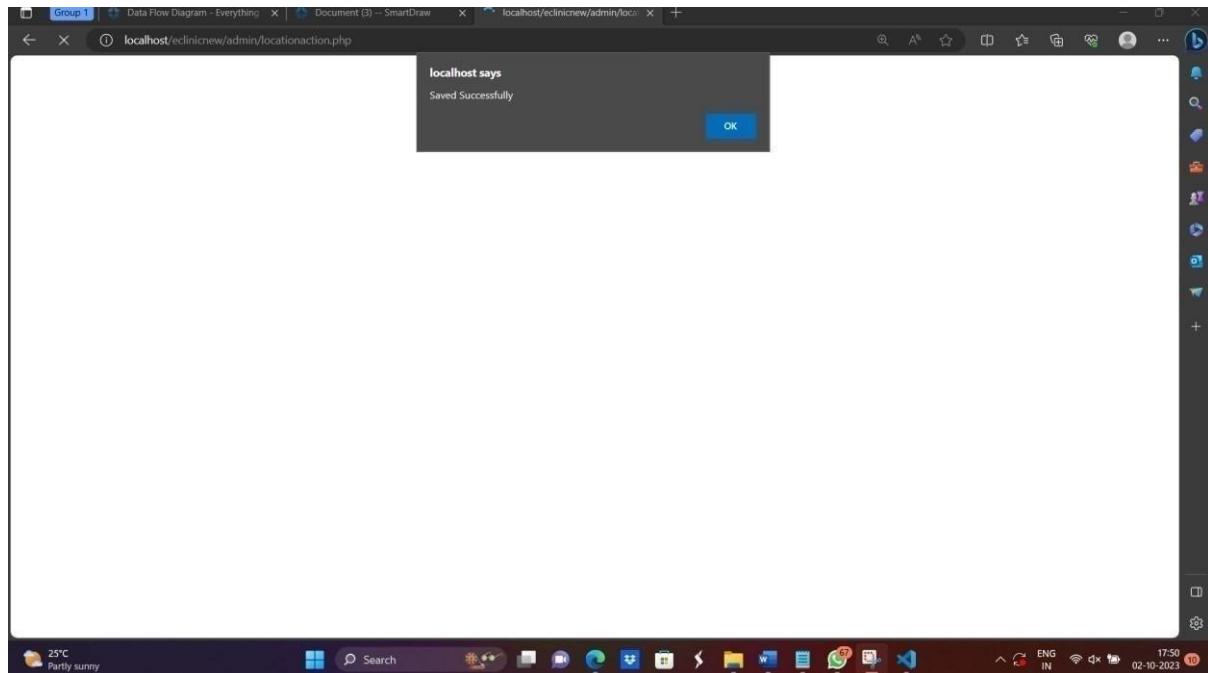


Fig 5.4 Integration testing result

We have performed integration testing whenever we have combined two modules together. When two modules are combined we have checked whether the functionality works correctly or not through integration testing.

### 5.1.3. Validation Testing

At the culmination of Black Box testing, software is completely assembled as a package, interface errors have been uncovered and corrected and final series of software tests, Validation tests begins. Validation testing can be defined many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably accepted by the customer. After validation test has been conducted one of the two possible conditions exists.

1. The function or performance characteristics conform to specification and are accepted.
2. A derivation from specification uncovered and a deficiency list is created.

The screenshot shows the 'Patient Registration Form' on the E-Clinic website. The form includes fields for Name, Age, Mobile Number, Email, Location, Place, Pincode, and Password. A 'Register' button is at the bottom. An error message 'Please fill out this field' is displayed above the Name input field, indicating that the name is required. The rest of the fields appear to be empty or have placeholder text.

Fig 5.5 Patient name validation

The screenshot shows a web-based patient registration form titled "Patient Registration Form". The form includes fields for Name (Anakha), Gender (Male or Female, with a note "Please select one of these options."), Age (20), Mobile Number (9541526398), Email (anakha43@gmail.com), Location (Thodupuzha), Place (Vazhithala), Pincode (686662), and two password fields (Username: anakha, Password: \*\*\*\*\*). A blue "Register" button is at the bottom. At the very bottom, there's a link "Already have an account? Login". The top right corner of the page has links for HOME, PATIENT REGISTRATION, LOGIN, a search icon, and an Appointment button.

Fig 5.6 Patient gender validation

We have given various validations in our forms so that there will be a neat format for the data's that are entered on to the website. We have also given an already existing validation so that the data redundancy is reduced; same data is not entered twice.

#### 5.1.4. User Acceptance Testing

Acceptance Testing is a level of the software testing process where a system is tested for acceptability. User Acceptance testing is the software testing process where system tested for acceptability & validates the end to end business flow. Such type of testing executed by client in separate environment & confirms whether system meets the requirements as per requirement specification or not. UAT is performed after System Testing is done and all or most of the major defects have been fixed. This testing is to be conducted in the final stage of Software Development Life Cycle (SDLC) prior to system being delivered to a live environment. UAT users or end users are concentrating on end to end scenarios & typically involves running a suite of tests on the completed system.

User Acceptance testing also known as Customer Acceptance testing (CAT), if the system is being built or developed by an external supplier. The CAT or UAT are the final confirmation from the client before the system is ready for production. The business customers are the primary owners of these UAT tests. These tests are created by business customers and articulated in business domain languages. So ideally it is collaboration between business customers, business analysts, testers and developers. It consists of test suites which involve

multiple test cases & each test case contains input data (if required) as well as the expected output. The result of test case is either a pass or fail.

## 5.2 TEST CASE DOCUMENTS

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly. The process of developing test cases can also help find problems in the requirements or design of an application. A sample of test case document format is given below.

Table 5.1 Test Case

TC No.	Test Steps	Expected Result	Actual Result	Status	Comment
1	Run application and navigate to Department registration page	Department registration screen is displayed. A field for entering Department name and a field for selecting Department image and a button submit should be present. This page contains another button namely ViewDepartment to show all registered departments.	Department registration screen is displayed. A field for entering Department name and a field for selecting Department image and a button submit should be present. This page contains another button namely View Department to show all registered departments.	Pass	
2	Enter the save button without entering no name to the	A message should be displayed stating that 'Please fill out this field' in beside of	A message should be displayed stating that 'Please fill out this field' in beside of Department Name	Pass	

	Department name field	Department Name field.	field.		
3	Enter the save button after enter on name to the department name field and without selecting an image for the Department Image field.	A message should be displayed stating that 'Please select an image' in beside of Department Image field.	A message should be displayed stating that 'Please select an image' in beside of Department Image field.	Pass	
4	Enter the save button after entering a food category name starting with lowercase	A message have been displayed stating that 'please match the request format.'in beside of Department name	A message have been displayed stating that 'please match the request format.'in beside of Department name	Pass	
5	Enter the save button after entering valid Department name and Department image	A message should be displayed stating that Department Registered successfully.	A message should be displayed stating that Department Registered successfully	Pass	

6	After Registration navigate to view registered Departments screen	Department details is displayed. Registered departments are displayed in a table contain fields SL.NO, Name, Image, Action	Department details is displayed. Registered departments are displayed in a table contain fields SL.NO, Name, Image, Action	Pass	

## 6. CONCLUSION

The project was successfully completed within the time span allotted. All the modules are tested separately and put together to form the main system. Finally, the modules are tested with real data and it worked successfully. Thus, the system has fulfilled the entire objective defined.

This project helps people to find different departments and doctors. Helps people to see the available doctors and can book their tokens according to their needs. This system is a bunch of benefits from the various point of views. As this online application enables the end users to register to the system online, we can select the department and doctors and book tokens according to our convenient time and day. Also, the payment can be made through online and it is secure. The user's details are stored using the electronic media. Also the people can easily view their profile and can make any changes like editing or deleting and this feature is available for doctors also. The user's details are maintained confidentially because it maintains a separate account for each user.

The purpose of Online Token Booking System is to automate the existing manual system by the help of computerized equipment's and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. The required software and hardware are easily available and easy to work with. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information. The aim is to automate its existing manual system by the help of computerized equipment's and fullfledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. This digital platform leverages technology to provide a range of benefits, such as improved patient care, increased administrative efficiency, better decision-making, and enhanced communication between healthcare providers and patients. It allows healthcare providers to maintain electronic health records (EHRs) for each patient, including medical history, treatment plans, medications, and test results. This ensures that patient data is readily accessible, promoting accurate diagnosis and personalized care.

---

## 7. REFERENCES

- [1] K K Aggarwal, Yogesh Singh – Software Engineering Third Edition
  - [2] Programming PHP,Ramus Lerdorf and Kevin Tatore
  - [3] Beginning PHP, Dave W Mercer, Allan Kent, Steven D Nowicki, David Mercer, Dan Squier,Wankyu Choi,Wiley Publishing,Inc
  - [4] Roger S Pressman-Software Engineering: A Practitioner's Approach, Sixth Edition, MCGraw-Hill Higher Education
  - [5] Pamkaj Jalote-An Integrated approach to Software Engineering, Second Edition, Narosa Publishing Company
  - [6] <https://www.stackoverflow.com>
  - [7] <https://www.sqlsolutions.com>
  - [8] <http://ecomputernotes.com/software-engineering/feasibilitystudy>
  - [9] [https://en.wikipedia.org/wiki/Third\\_normal\\_form](https://en.wikipedia.org/wiki/Third_normal_form)
  - [10] <https://www.w3resource.com>
  - [11] <https://www.guru99.com/user-acceptance-testing.html>
  - [12] <http://softwaretestingfundamentals.com/black-box-testing/>
  - [13] <https://www.geeksforgeeks.org/php-coding-standards/>
  - [14] <https://www.geeksforgeeks.org/software-engineering-sdlc-v-model/>
  - [15] [https://www.tutorialspoint.com/sdlc/sdlc\\_quick\\_guide.htm](https://www.tutorialspoint.com/sdlc/sdlc_quick_guide.htm)
  - [16] <http://www.coddletech.com/php>
  - [17] <http://www.datatreesystems.in/technologies/mysql/>
  - [18] <http://www.wampserver.com/en/>
  - [19] [https://en.wikipedia.org/wiki/Microsoft\\_Word](https://en.wikipedia.org/wiki/Microsoft_Word)
  - [20] <https://en.wikipedia.org/wiki/SmartDraw>
-

## 8. APPENDIX

### 8.1. SCREENSHOTS

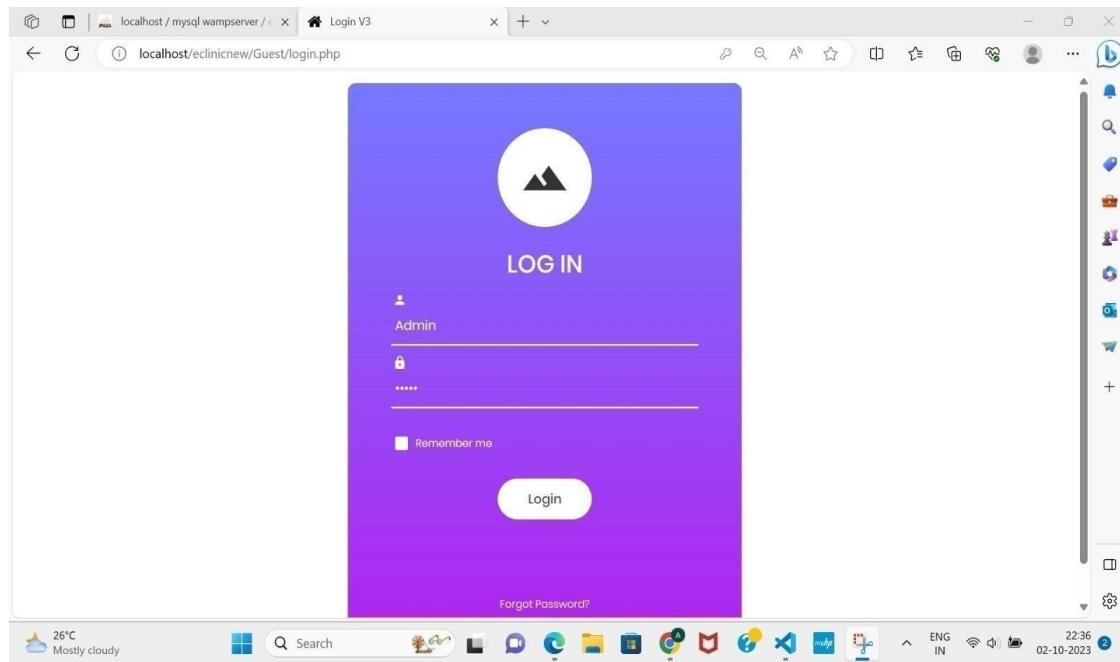


Fig 8.1 Admin login page

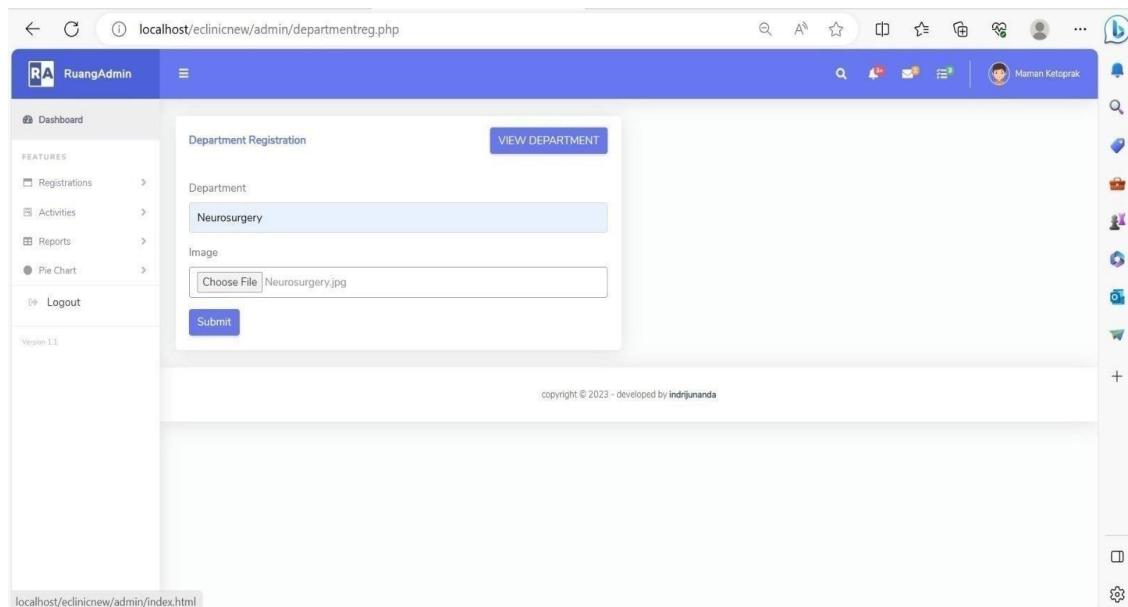
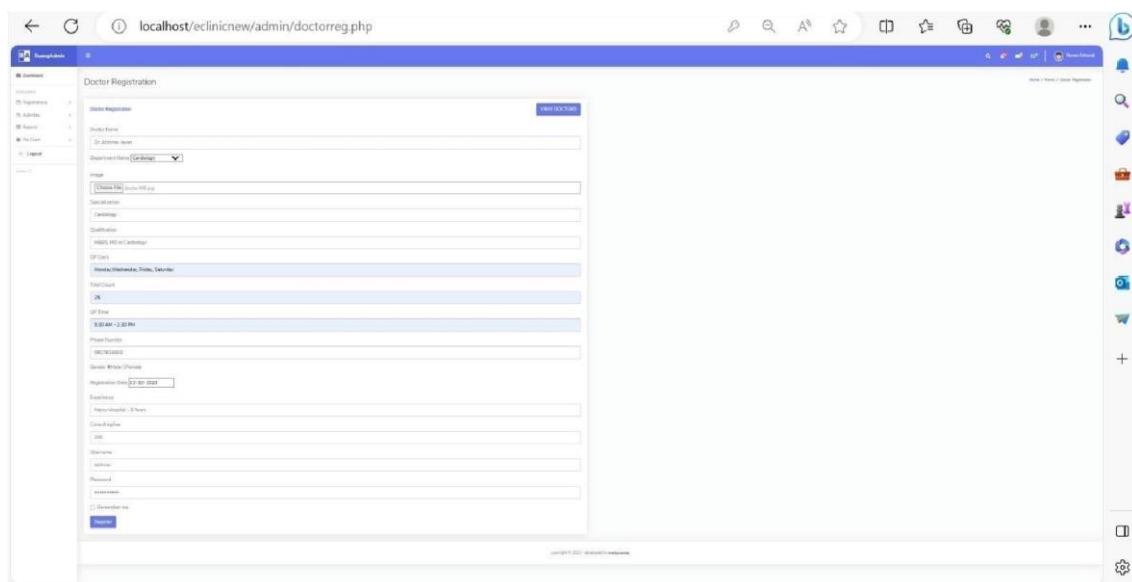


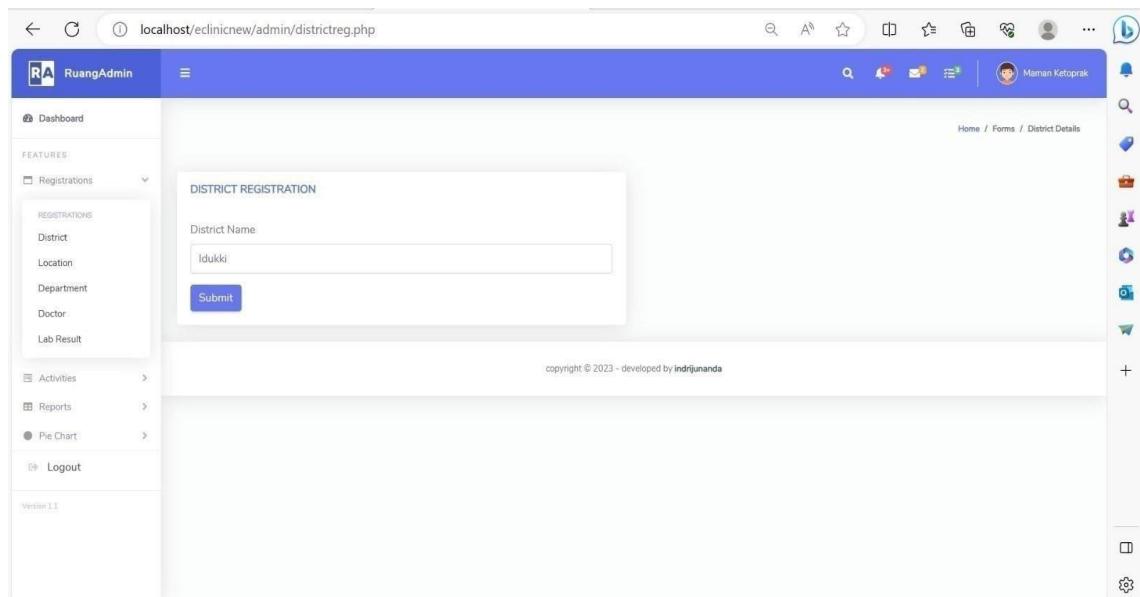
Fig 8.2 Department Registration Page



The screenshot shows a web browser window with a blue header bar containing the URL 'localhost/eclinicnew/admin/doctorreg.php'. The main content area is titled 'Doctor Registration' and contains several input fields:

- Image: A file input field with 'choose file' and 'cancel' buttons.
- Specialization: A dropdown menu showing 'General' and 'Cardiology'.
- Certification: A dropdown menu showing 'MBBS, MD in Cardiology' and 'GP Card'.
- Mobile Number: A text input field with placeholder 'Mobile Number, Doctor, Doctor'.
- Total Cases: A text input field with value '26'.
- GP Time: A dropdown menu showing '9:00 AM - 12:30 PM'.
- Phone Number: A text input field with placeholder '9876543210'.
- Gender: A dropdown menu showing 'Male/Female'.
- Registration Date: A date input field with value '22-01-2023'.
- Address: A dropdown menu showing 'Home Hospital - 3 Years'.
- Qualification: A dropdown menu showing 'MBBS, MD in Cardiology'.
- Experience: A dropdown menu showing '200'.
- Username: A text input field with placeholder 'username'.
- Password: A text input field with placeholder 'password'.
- Confirm Password: A text input field with placeholder 'confirm password'.
- Remember me: A checkbox labeled 'Remember me'.
- Submit: A blue 'Submit' button.

Fig 8.3 Doctor Registration Page

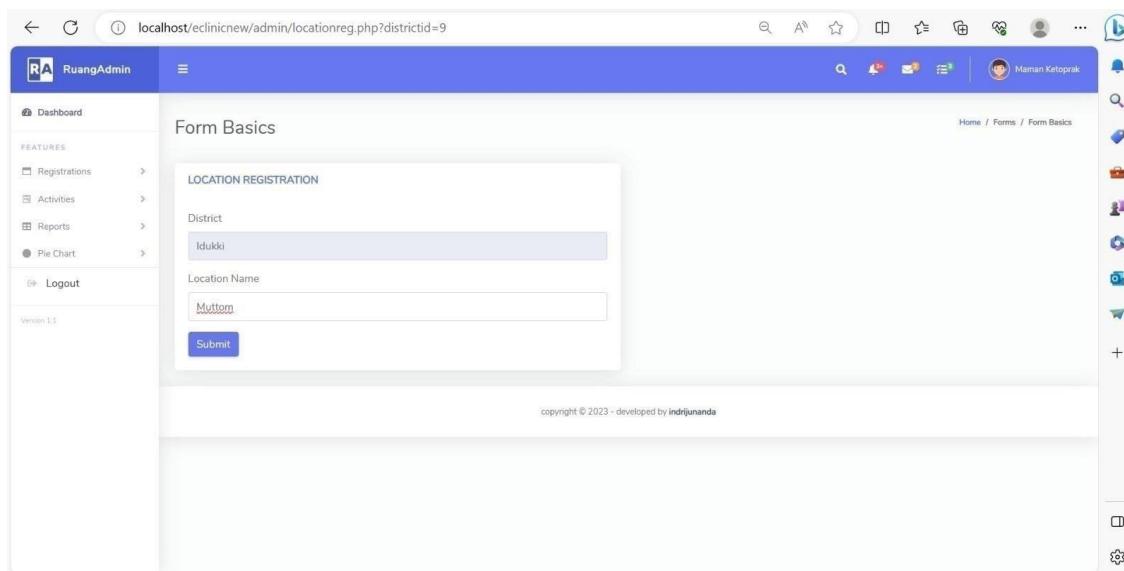


The screenshot shows a web browser window with a blue header bar containing the URL 'localhost/eclinicnew/admin/districtreg.php'. The main content area is titled 'DISTRICT REGISTRATION' and contains a single input field:

District Name:

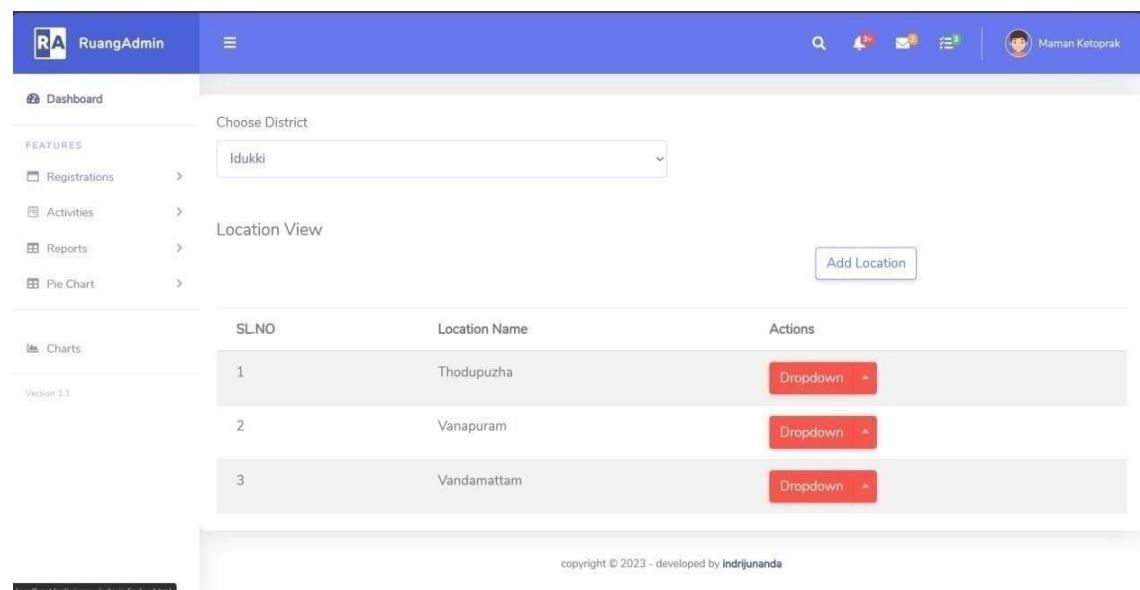
Below the input field is a blue 'Submit' button. At the bottom right of the page, there is a copyright notice: 'copyright © 2023 - developed by indrijunanda'.

Fig 8.4 District Registration Page



The screenshot shows a web application interface titled "RuangAdmin". The main content area is titled "Form Basics" and contains a form for "LOCATION REGISTRATION". The form has two fields: "District" (with "Idukki" selected) and "Location Name" (with "Muttom" entered). A "Submit" button is at the bottom. The left sidebar lists "FEATURES" such as "Registrations", "Activities", "Reports", and "Pie Chart", along with a "Logout" link. The right sidebar includes a user profile for "Maman Ketoprak" and various icons for notifications, search, and other functions.

Fig 8.5 Location Registration Page

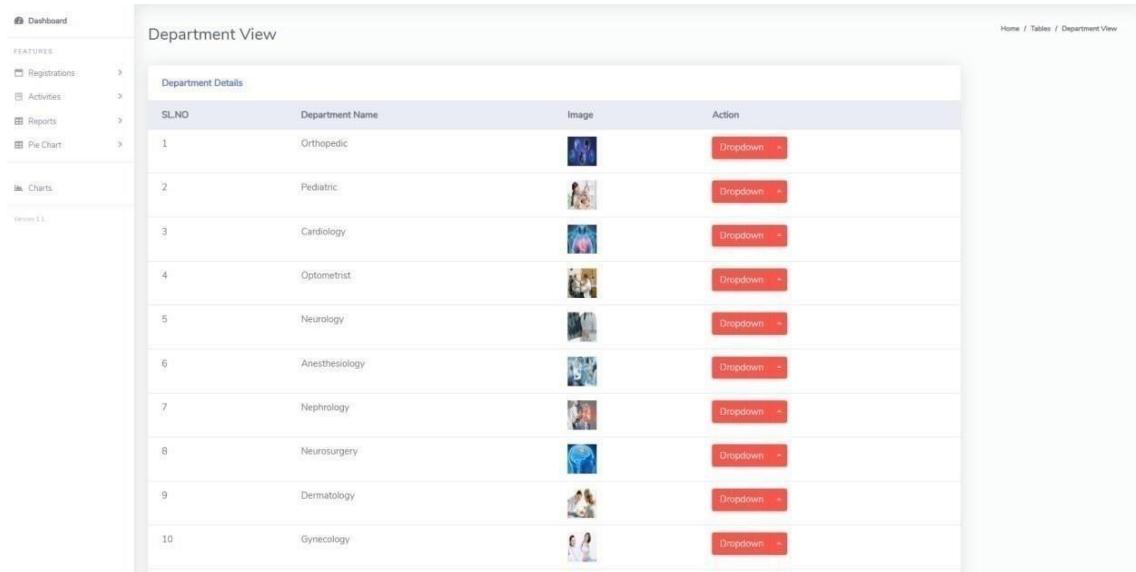


The screenshot shows a web application interface titled "RuangAdmin". The main content area is titled "Location View" and displays a table of locations. The table has columns for "SL.NO", "Location Name", and "Actions". The data is as follows:

SL.NO	Location Name	Actions
1	Thodupuzha	Dropdown ▲
2	Vanapuram	Dropdown ▲
3	Vandamattam	Dropdown ▲

The left sidebar lists "FEATURES" like "Registrations", "Activities", "Reports", and "Pie Chart", along with a "Charts" section and a "Version 1.1" link. The right sidebar includes a user profile for "Maman Ketoprak" and various icons.

Fig 8.6 Location View

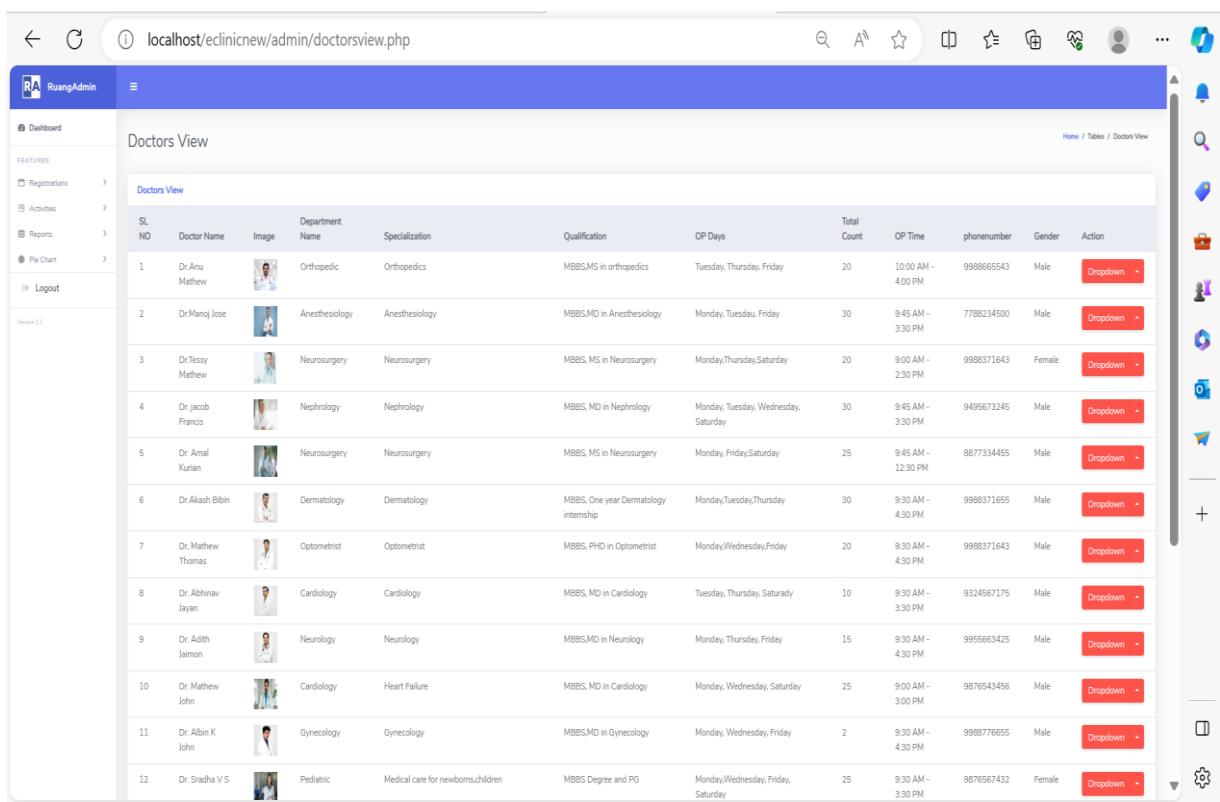


**Department View**

Department Details

SLNO	Department Name	Image	Action
1	Orthopedic		Dropdown
2	Pediatric		Dropdown
3	Cardiology		Dropdown
4	Optometrist		Dropdown
5	Neurology		Dropdown
6	Anesthesiology		Dropdown
7	Nephrology		Dropdown
8	Neurosurgery		Dropdown
9	Dermatology		Dropdown
10	Gynecology		Dropdown

Fig 8.7 Department View



**Doctors View**

SL NO	Doctor Name	Image	Department Name	Specialization	Qualification	OP Days	Total Count	OP Time	phonenumer	Gender	Action
1	Dr.Anu Mathew		Orthopedic	Orthopedics	MBBS,MS in orthopedics	Tuesday,Thursday,Friday	20	10:00 AM - 4:00 PM	9988665543	Male	Dropdown
2	Dr.Manjo Jose		Anesthesiology	Anesthesiology	MBBS,MD in Anesthesiology	Monday,Tuesday,Friday	30	9:45 AM - 3:30 PM	7788234500	Male	Dropdown
3	Dr.Tessy Mathew		Neurosurgery	Neurosurgery	MBBS, MS in Neurosurgery	Monday,Thursday,Saturday	20	9:00 AM - 2:30 PM	9988371643	Female	Dropdown
4	Dr.jacob Francis		Nephrology	Nephrology	MBBS, MD in Nephrology	Monday,Tuesday,Wednesday,Saturday	30	9:45 AM - 3:30 PM	9495673245	Male	Dropdown
5	Dr. Amal Kurian		Neurosurgery	Neurosurgery	MBBS, MS in Neurosurgery	Monday,Friday,Saturday	25	9:45 AM - 12:30 PM	8877334455	Male	Dropdown
6	Dr.Akash Bilin		Dermatology	Dermatology	MBBS, One year Dermatology internship	Monday,Tuesday,Thursday	30	9:30 AM - 4:30 PM	9988371655	Male	Dropdown
7	Dr. Mathew Thomas		Optometrist	Optometrist	MBBS, PhD in Optometrist	Monday,Wednesday,Friday	20	9:30 AM - 4:30 PM	9988371643	Male	Dropdown
8	Dr. Abhinav Jayan		Cardiology	Cardiology	MBBS, MD in Cardiology	Tuesday,Thursday,Saturday	10	9:30 AM - 3:30 PM	9324567175	Male	Dropdown
9	Dr. Adith Jamon		Neurology	Neurology	MBBS,MD in Neurology	Monday,Thursday,Friday	15	9:30 AM - 4:30 PM	9955663425	Male	Dropdown
10	Dr. Mathew John		Cardiology	Heart Failure	MBBS, MD in Cardiology	Monday,Wednesday,Saturday	25	9:00 AM - 3:00 PM	9876543456	Male	Dropdown
11	Dr. Albin K John		Gynecology	Gynecology	MBBS,MD in Gynecology	Monday,Wednesday,Friday	2	9:30 AM - 4:30 PM	9988776655	Male	Dropdown
12	Dr. Sradha V S		Pediatric	Medical care for newborns,children	MBBS Degree and PG	Monday,Wednesday,Friday,Saturday	25	9:30 AM - 3:30 PM	9876567432	Female	Dropdown

Fig 8.8 Doctors View

The screenshot shows the 'View\_LabTest.php' page from the 'localhost/eclinicnew/admin' URL. The page has a blue header bar with the 'RuangAdmin' logo and a search bar. On the left, there's a sidebar with 'FEATURES' sections: 'Registrations', 'Activities', 'Reports', 'Pie Chart', and 'Logout'. The main content area is titled 'Basic Tables' and shows a table titled 'Tables / Basic tables'. The table has columns: Sl.No, PatientName, Email, Contact, Patient RegNo, TestName, Description, DoctorName, and Lab Result. It contains three rows of data:

Sl.No	PatientName	Email	Contact	Patient RegNo	TestName	Description	DoctorName	Lab Result
1	Anju	anju@gmail.com	9988665543	RE_9009	Blood	erghtht	Dr.Anu Mathew	Upload Result
2	Anju	anju@gmail.com	9988665543	RE_9009	Liver Function	aaa	Dr.Anu Mathew	Upload Result
3	Alan	alanevaniyose@gmail.com	9876543456	RE-1028	Blood	RBC Count	Dr. Mathew John	Upload Result

At the bottom of the page, it says 'copyright © 2023 - developed by indrijunanda'.

Fig 8.9 Lab result

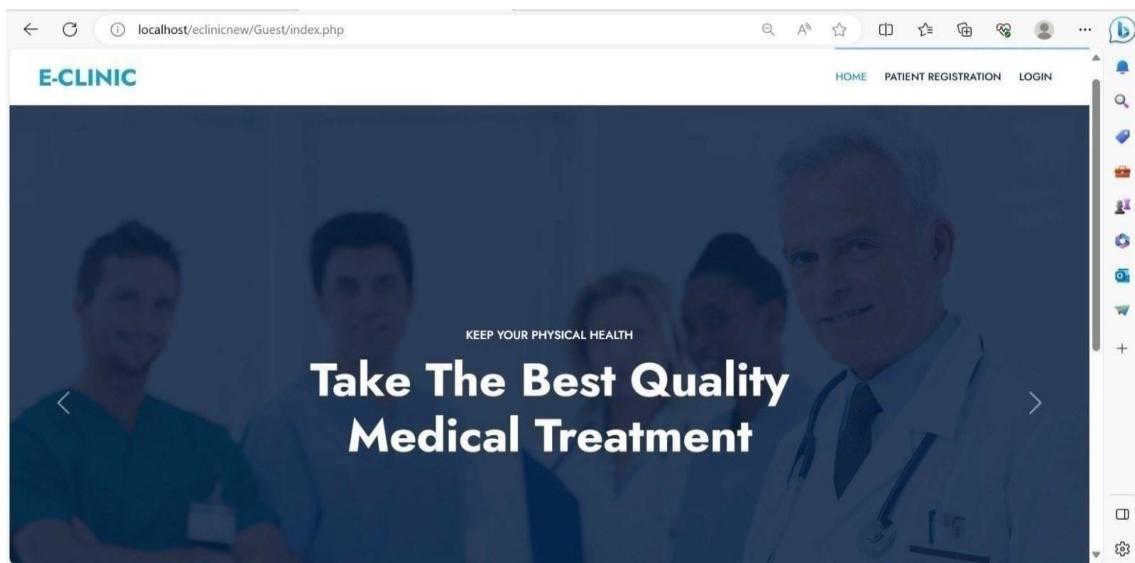


Fig 8.10 Patient index page

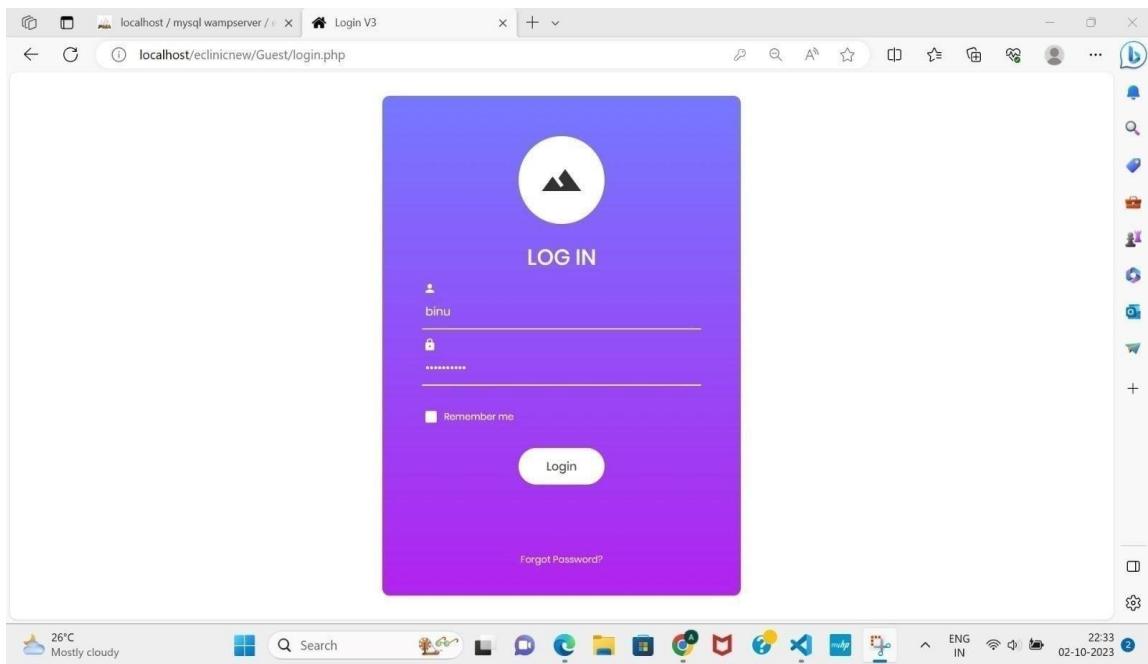


Fig 8.11 Patient Login page

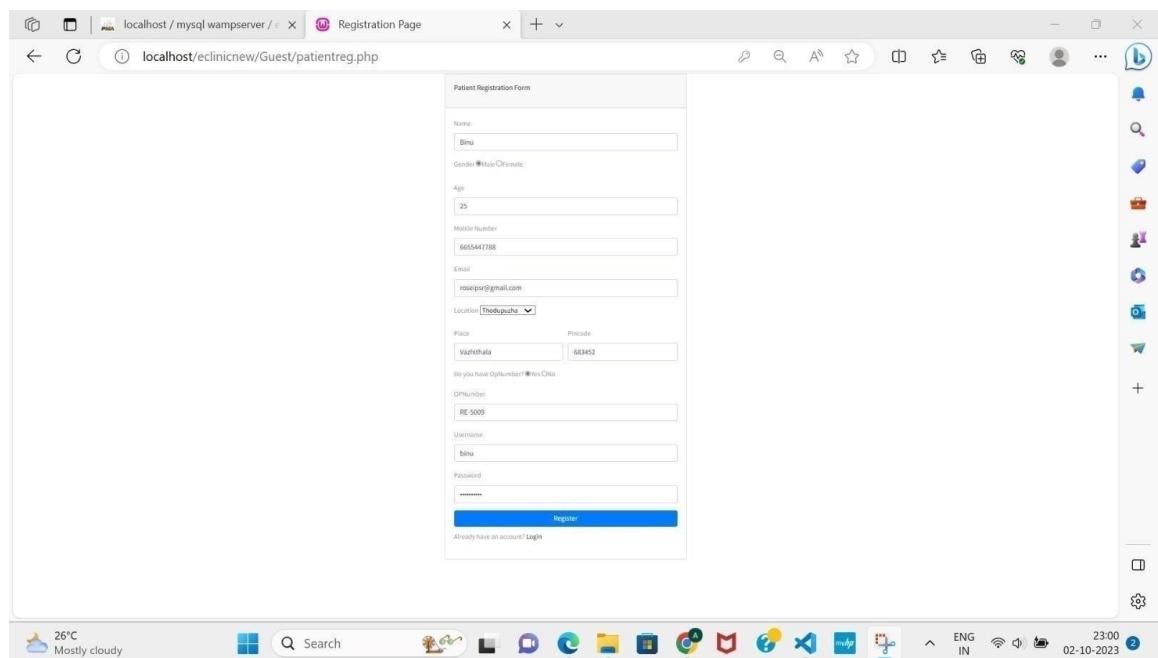


Fig8.12 Patient Registration

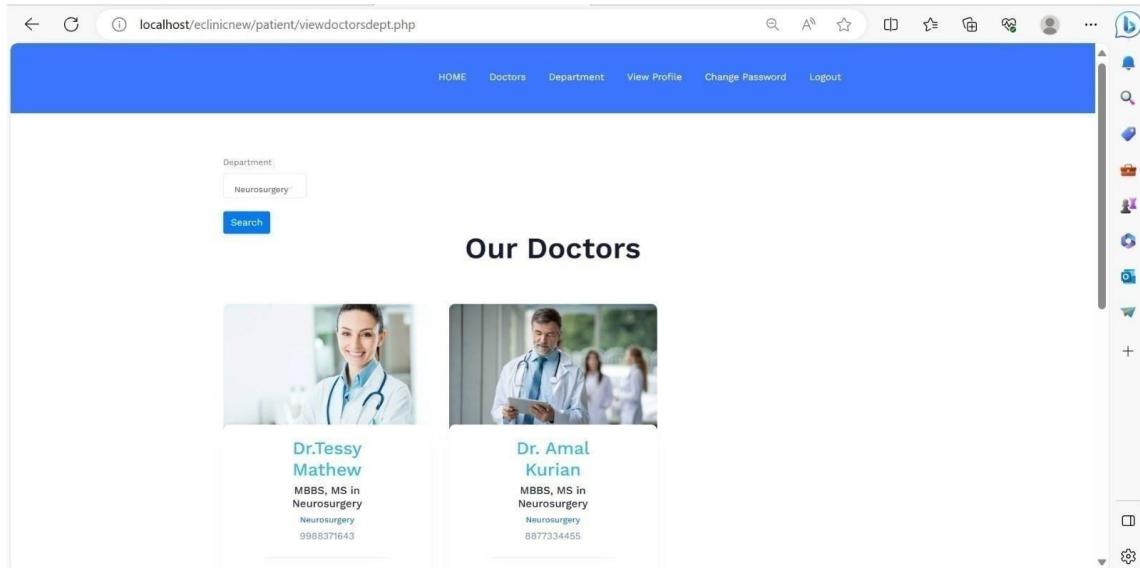


Fig 8.13 Doctors view

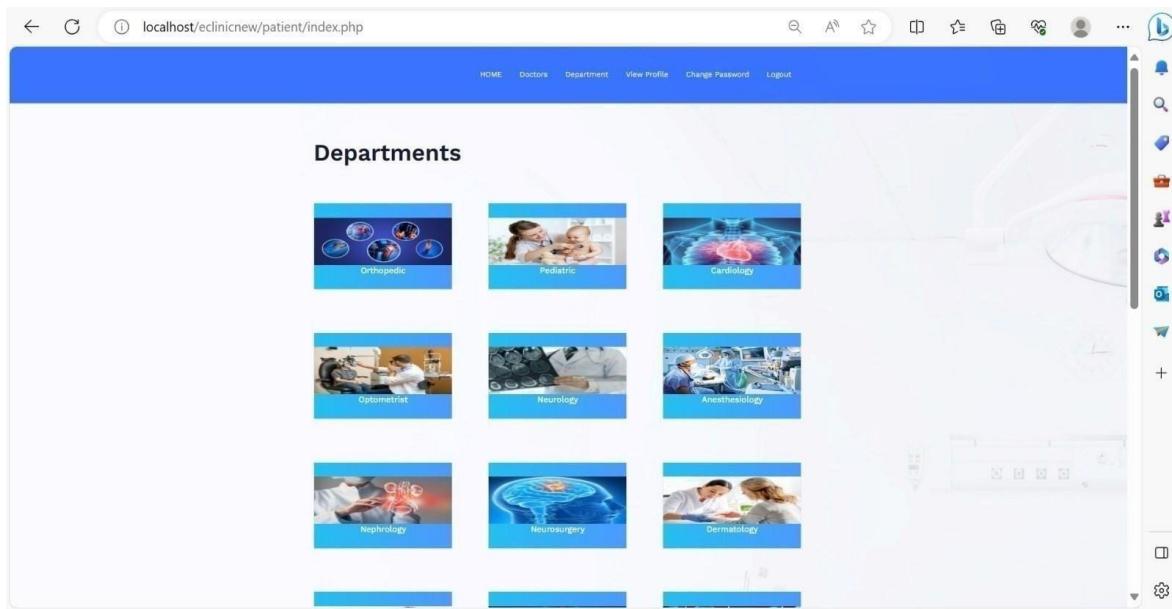


Fig 8.14 Department view

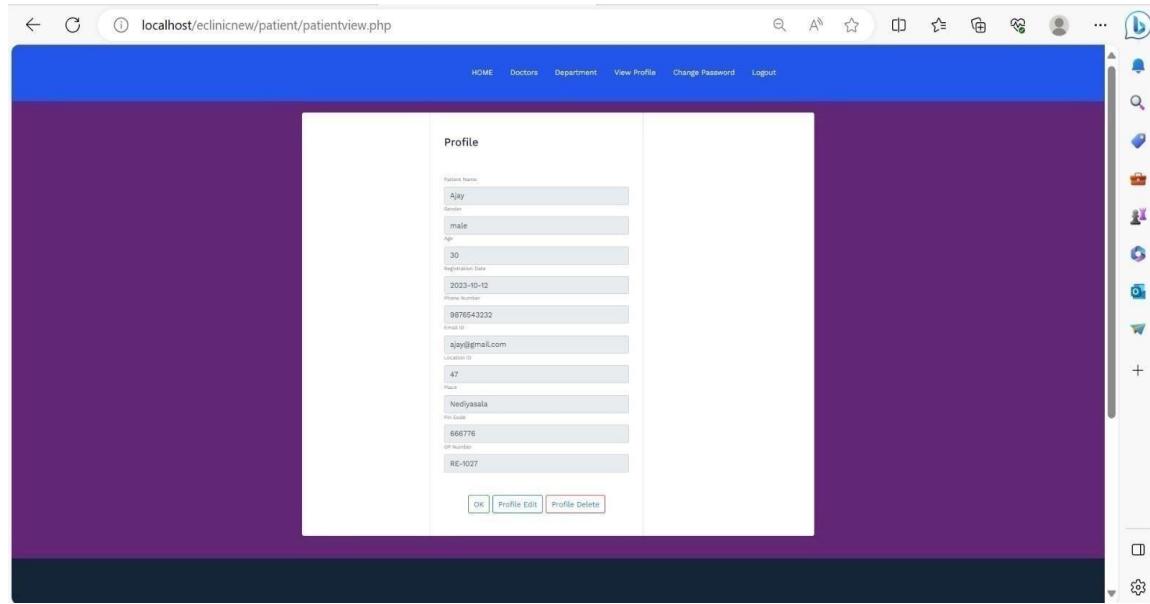


Fig 8.15 Patient view profile

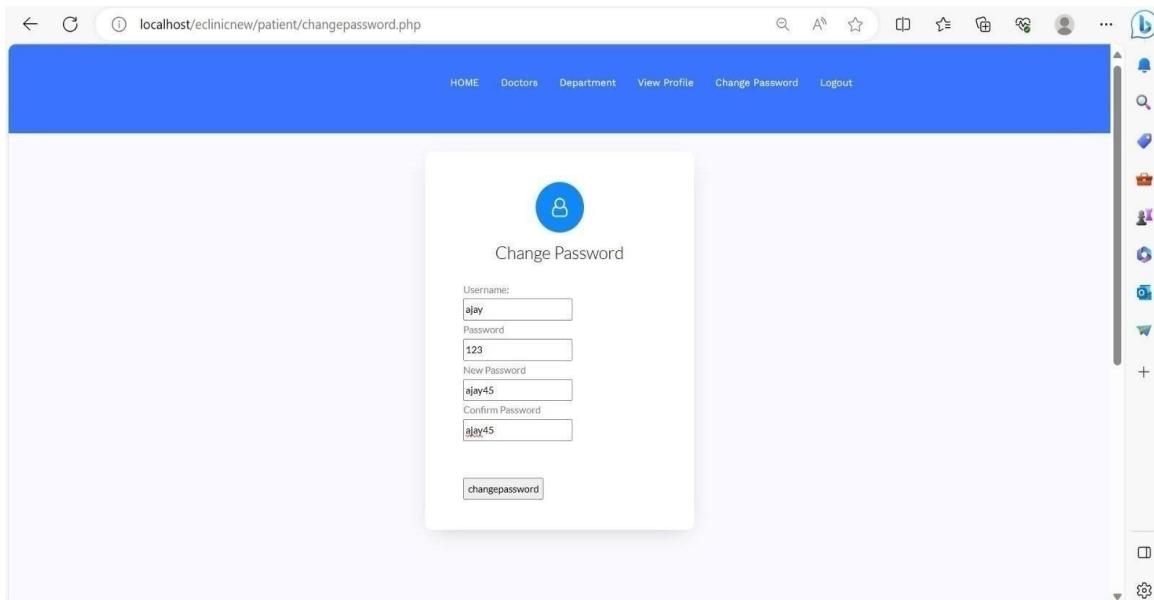


Fig 8.16 Change password of patient

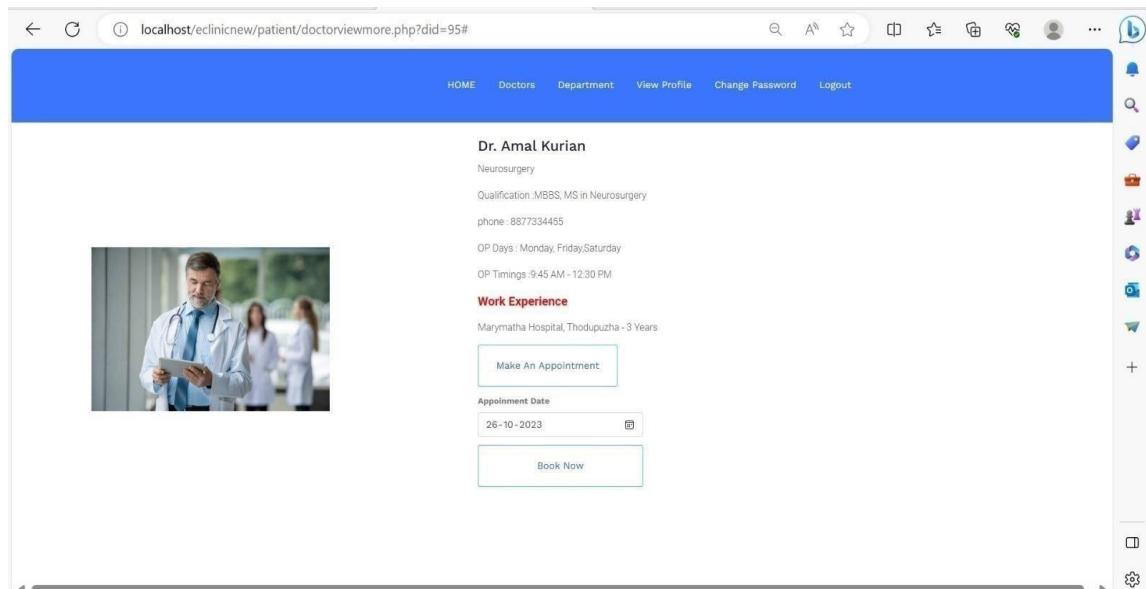


Fig 8.17 Doctor booking page

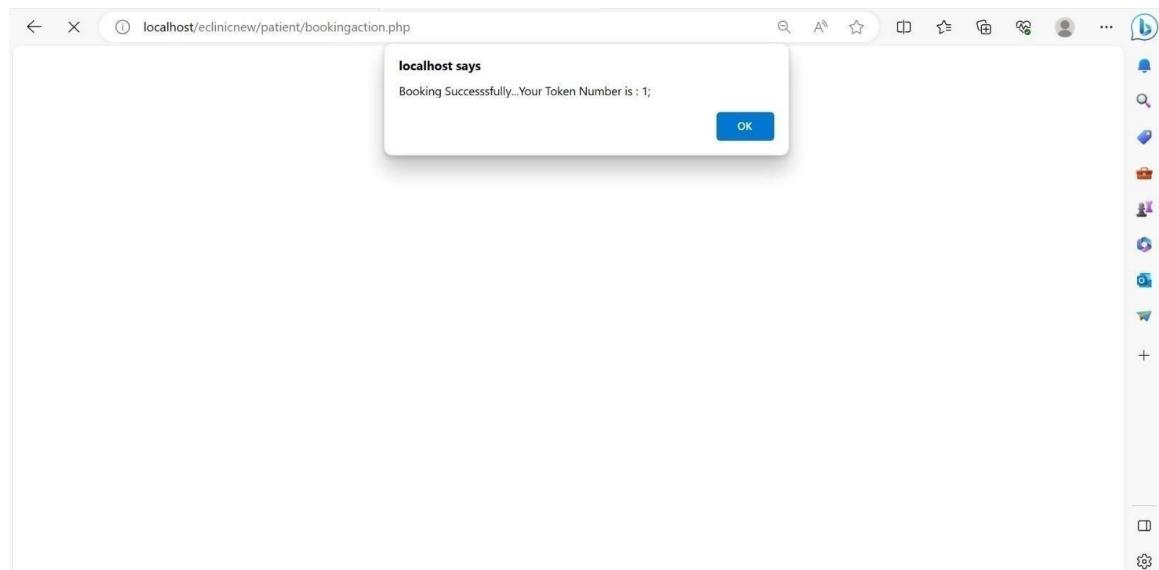


Fig 8.18 Patient token generation

The screenshot shows a web browser window with the URL `localhost/eclinicnew/patient/payment.php?doctorid=95&consultingfee=200&bookid=78`. The main content is titled "Payment Details". It contains the following form fields:

- Person Name: Arun Jose
- Doctor Name: Dr. Arun Jose
- Book Id: 78
- Amount: 200
- Payment Date: 2024-03-29
- Card Number: 548723933847
- Expiry: 12/2030
- cvv/cvc: ...

A large blue "Pay" button is at the bottom right.

Fig 8.19 Booking Payment

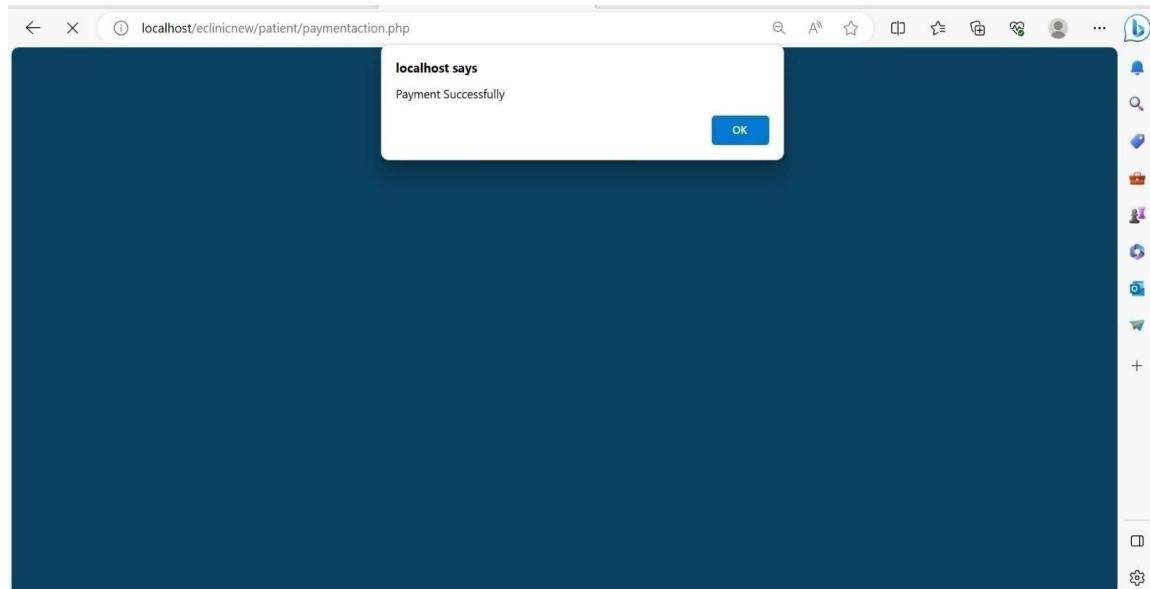


Fig 8.20 Booking Payment Success

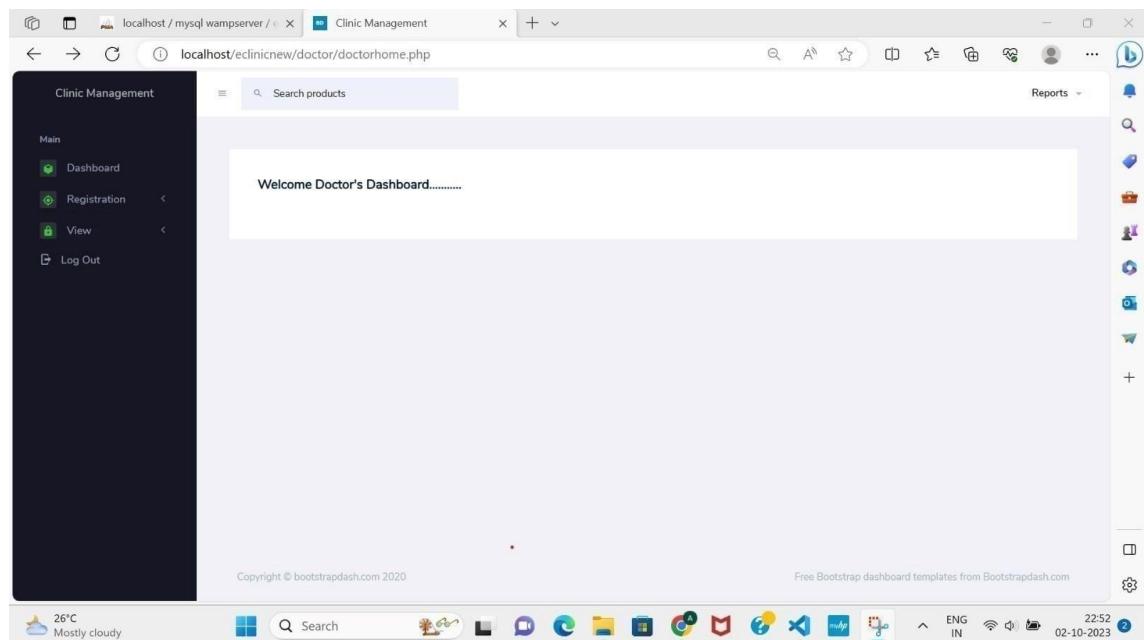


Fig 8.21 Doctor Home Page

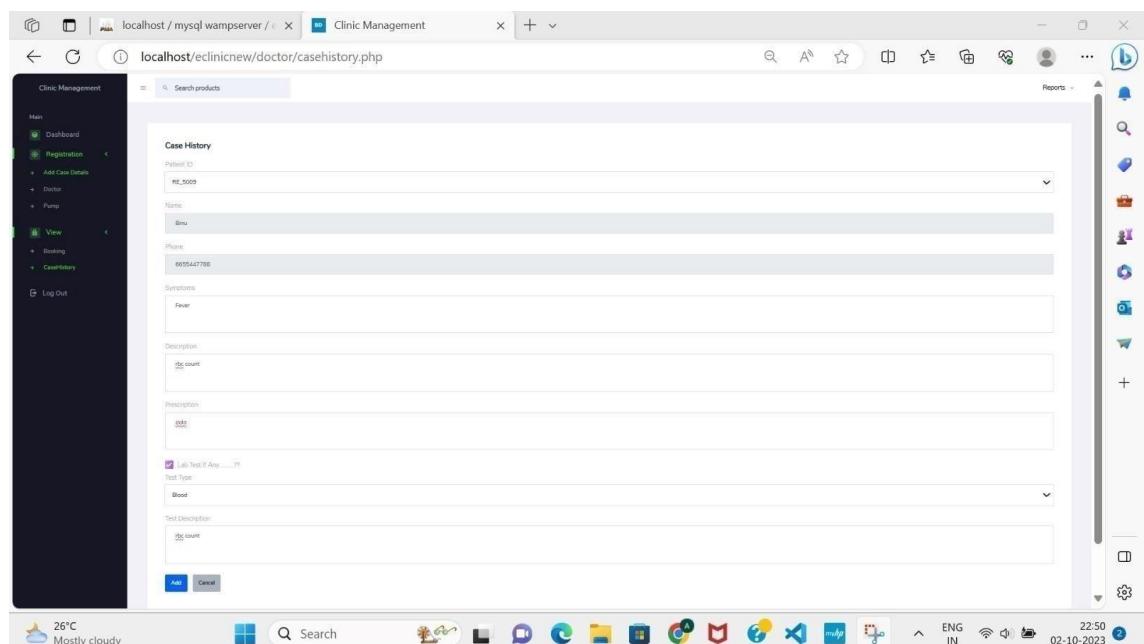


Fig 8.22 Patient case history

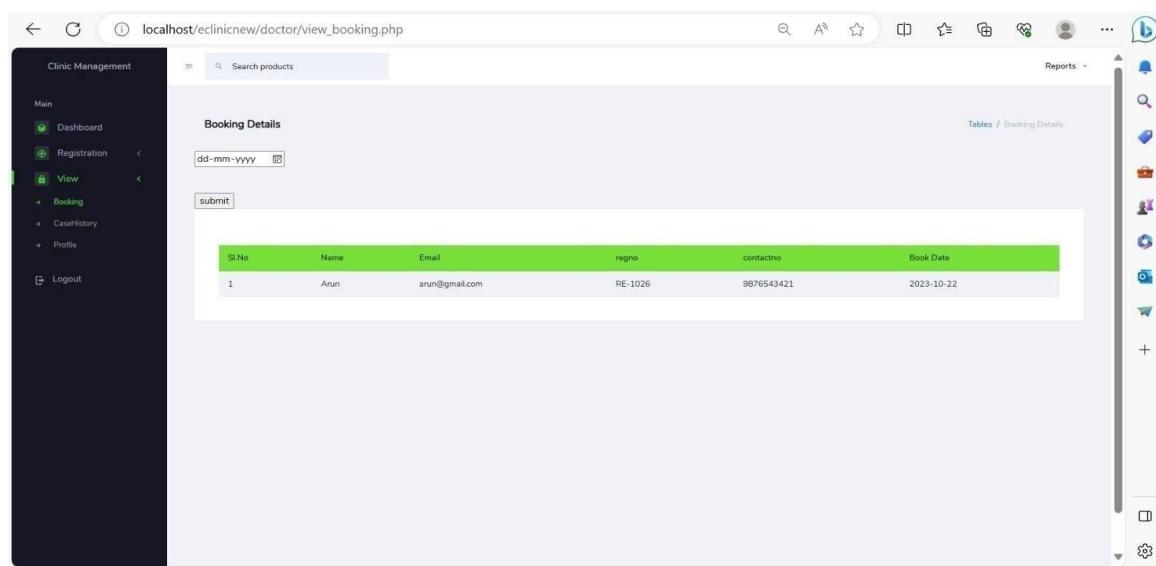


Fig 8.23 Patient view

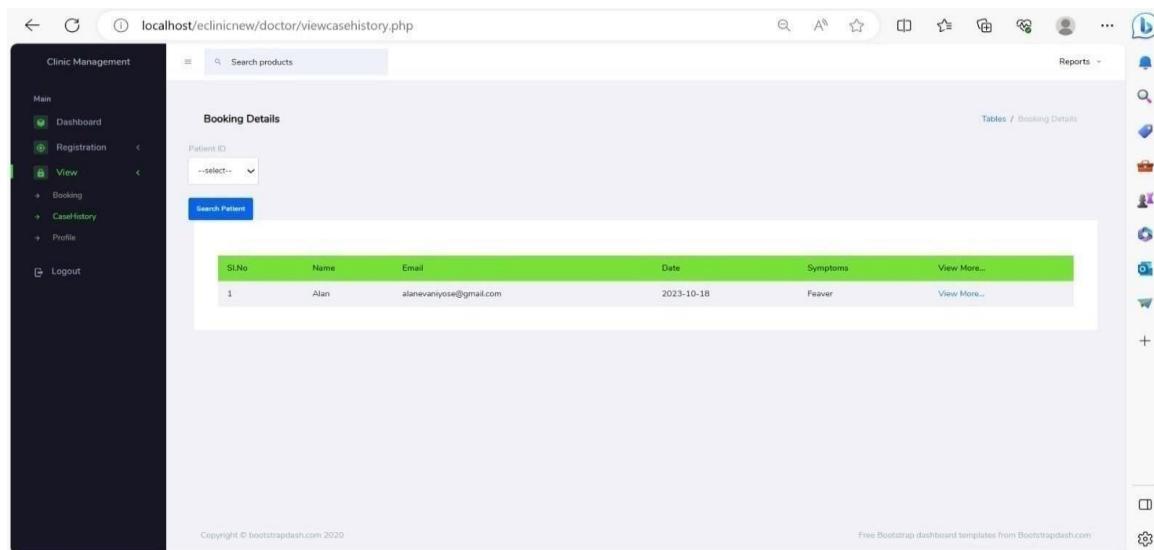


Fig 8.24 View case history

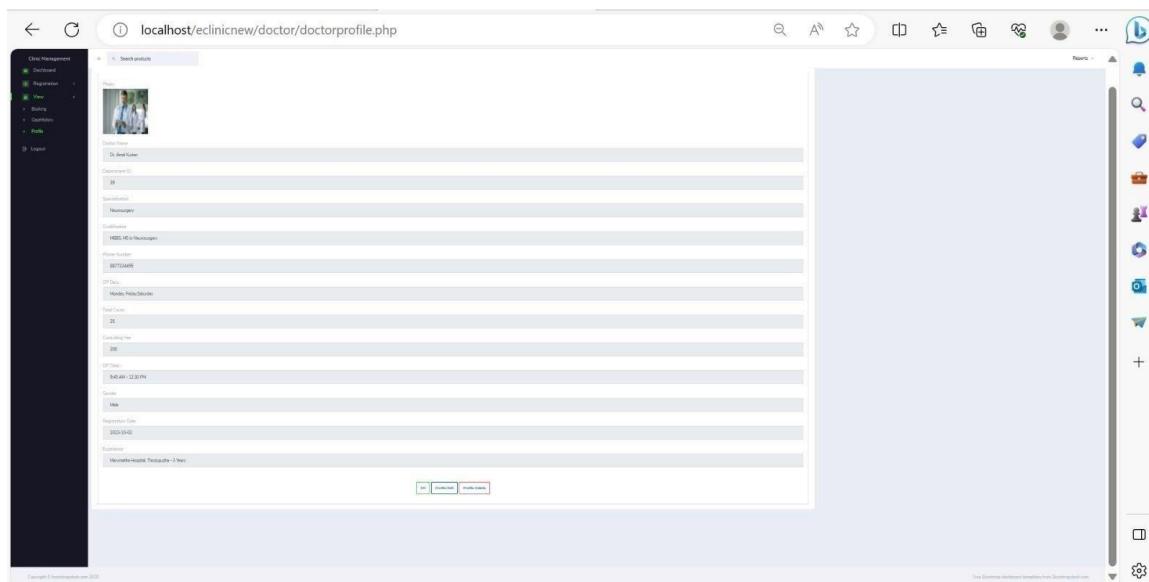


Fig 8.25 Doctor Profile view

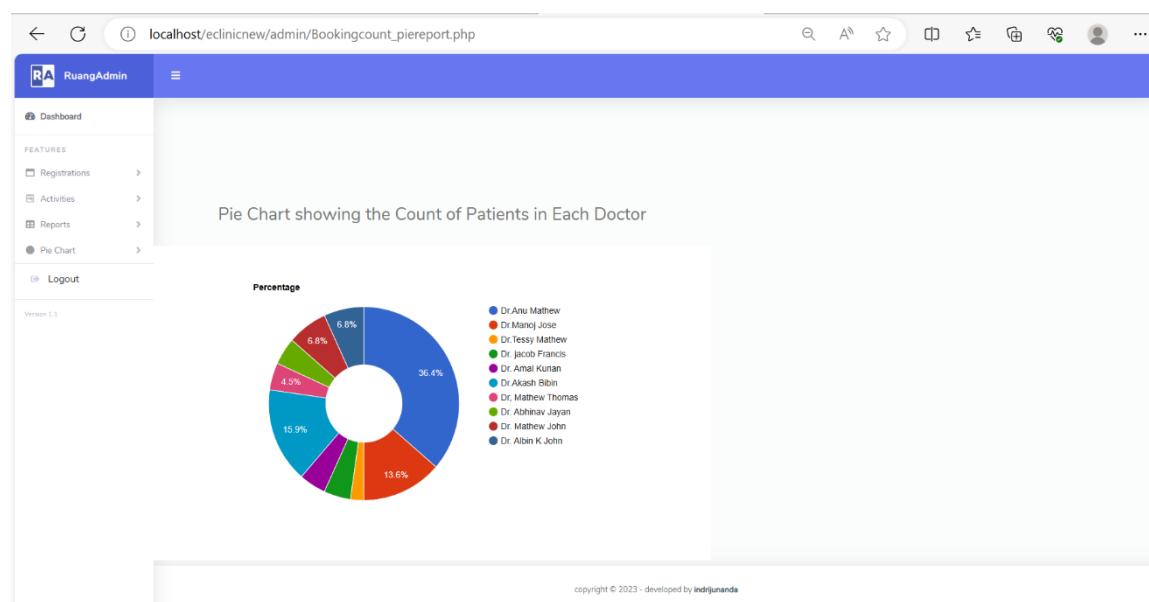


Fig 8.26 Pie chart for count of patients in each doctor

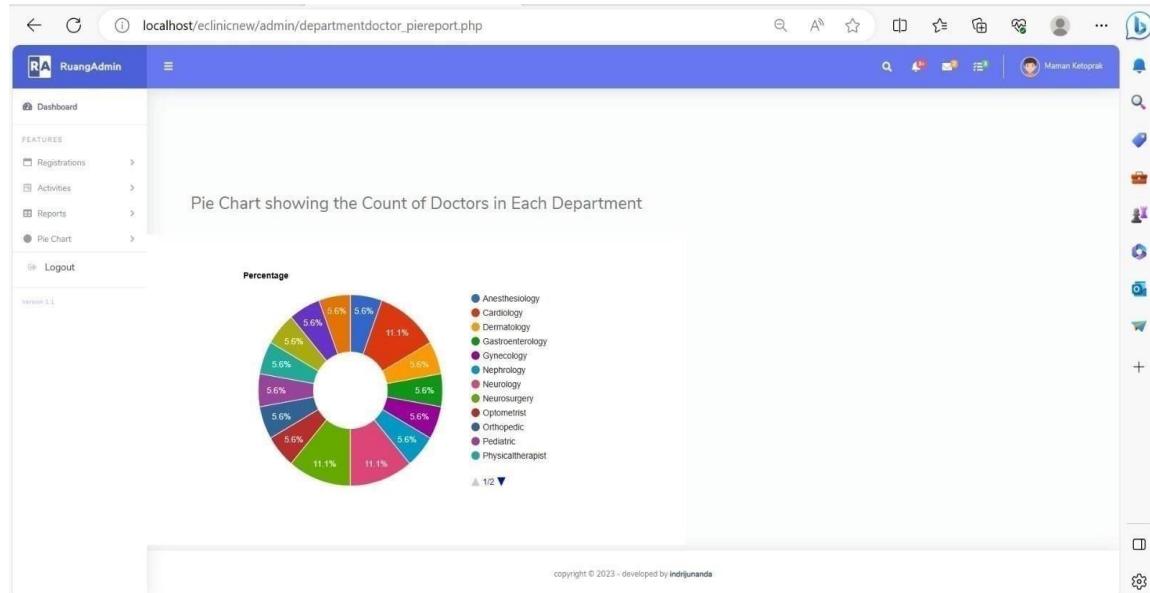


Fig 8.27 Pie chart for count of doctors in each department

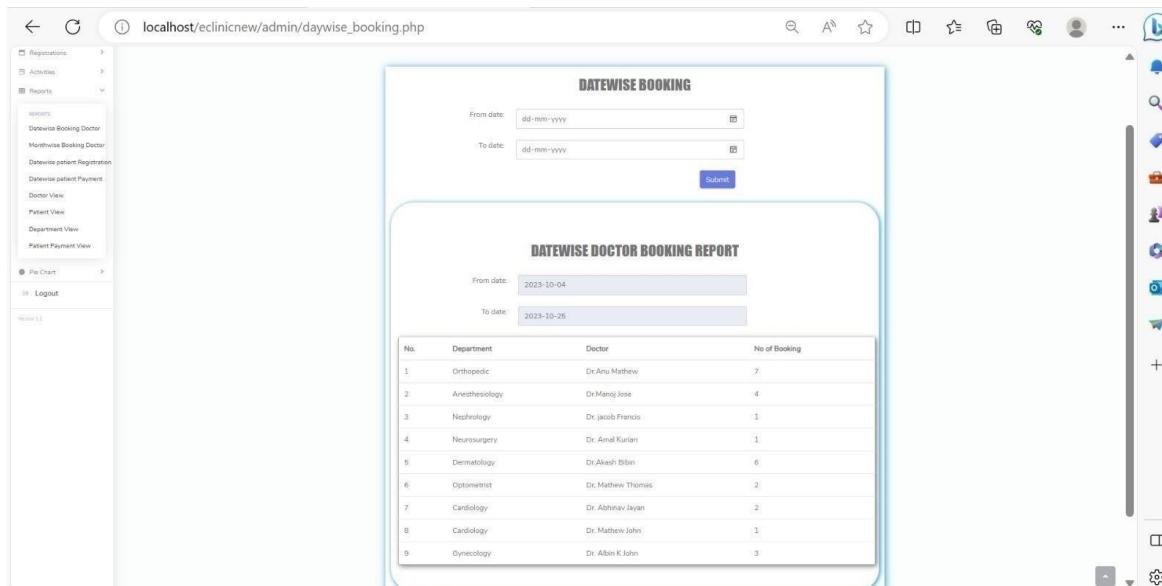


Fig 8.28 Datewise doctor booking

The screenshot shows a web-based administrative interface for 'RuangAdmin' at the URL [localhost/eclinicnew/admin/monthwise\\_booking.php](http://localhost/eclinicnew/admin/monthwise_booking.php). The left sidebar contains navigation links for Dashboard, FEATURES (Activities, Reports, Pie Chart), and Logout. The main content area is titled 'MONTHWISE BOOKING REPORT' and displays a table of doctor bookings between December 2018 and December 2020. The table includes columns for No., Department, Doctor, and No of Booking.

No.	Department	Doctor	No of Booking
1	Orthopedic	Dr Anu Mathew	6
2	Anesthesiology	Dr Mani Jose	2
3	Neurology	Dr Tessy Mathew	1
4	Nephrology	Dr Jacob Francis	1

Fig 8.29 Monthwise doctor booking

The screenshot shows a web-based administrative interface for 'RuangAdmin' at the URL [localhost/eclinicnew/admin/daywise\\_patientreg.php](http://localhost/eclinicnew/admin/daywise_patientreg.php). The left sidebar contains navigation links for Dashboard, FEATURES (Activities, Reports, Pie Chart), and Logout. The main content area is titled 'DATEWISE PATIENT' and displays a table of patient registrations between October 4, 2020, and October 17, 2020. The table includes columns for No., Patient Name, E-Mail, Phone Number, Place, and OF Number.

No.	Patient Name	E-Mail	Phone Number	Place	OF Number
1	Kiran	irw@gmail.com	9897719333	Kochi,Kerala	RK-1237
2	Aruna	arun@gmail.com	9897714555	Vaduthala	RK-1238
3	Arun	arun@gmail.com	12345678987	Vaduthala	RK-1239
4	Arun	arun@gmail.com	12345678987	Vaduthala	RK-1240
5	Hariha	hariha@gmail.com	9876543211	Kochi,Kerala	RK-1241
6	Arun	arun@gmail.com	9876543212	Vaduthala	RK-1242
7	Arun	arun@gmail.com	9876543213	Vaduthala	RK-1243
8	Lulu	luu@gmail.com	9876543211	Mavel	RK-1244
9	Aruna	arunaam123@gmail.com	9876543200	Vaduthala	RK-1245
10	Arun	arun@gmail.com	9876543211	Ranipet	RK-1246
11	Arun	arun@gmail.com	9876543232	Ranipet	RK-1247

Fig 8.30 Datewise patient registration report

The screenshot shows a web-based administration interface for 'RuangAdmin'. The main menu on the left includes 'Dashboard', 'FEATURES' (with sub-options 'Registrations', 'Activities', 'Reports', and 'Pie Chart'), and 'Logout'. The central area displays a 'DATEWISE PATIENT PAYMENT' form with 'From date' and 'To date' fields, and a 'Submit' button. Below it is a 'DATEWISE PATIENT PAYMENT REPORT' section with similar date filters and a table of patient payment details:

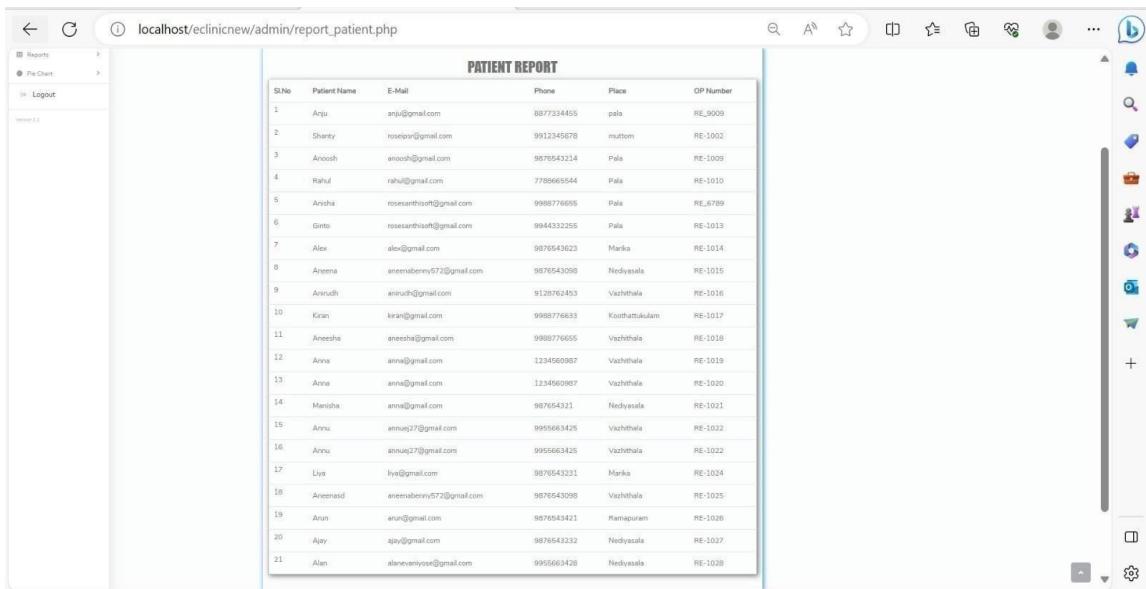
No.	Doctor Name	Patient Name	Book ID	OP Number
1	Dr Anu Mathew	Anresha	72	RE-1018
2	Dr Albin K John	Anus	77	RE-1026
3	Dr Abhinav Jayan	Ajey	71	RE-1027
4	Dr Mathew John	Alan	73	RE-1028

Fig 8.31 Datewise patient payment report

The screenshot shows a 'DOCTOR REPORT' page. The left sidebar has 'Reports' selected. The main content area displays a table of doctor information:

Sl.No	Doctor Name	Department	Phone	Qualification
1	Dr Anu Mathew	Orthopedic	9380660543	MBBSMS in Orthopedics
2	Dr Manoj Jose	Anesthesiology	7705234560	MBBS,MD in Anesthesiology
3	Dr Tessy Mathew	Neurosurgery	9888371643	MBBS, MS in Neurosurgery
4	Dr Jacob Francis	Nephrology	949673245	MBBS, MD in Nephrology
5	Dr. Amal Kurian	Neurosurgery	8077334465	MBBS, MG in Neurosurgery
6	Dr Akash IBBan	Dermatology	9000371655	MBBS, One year Dermatology Internship
7	Dr. Mathew Thomas	Optometrist	9900371643	MBBS, PhD in Optometrist
8	Dr. Abhinav Jayan	Cardiology	9324667119	MBBS, MD in Cardiology
9	Dr. Adith Iammi	Neurology	9955663425	MBBS,MD in Neurology
10	Dr. Mathew John	Cardiology	9876543456	MBBS, MD in Cardiology
11	Dr. Albin K John	Gynecology	9998776655	MBBS,MD in Gynecology
12	Dr. Sradha V S	Pediatric	9876567432	MBBS Degree and PG
13	Dr. Alphonsina Kurian	Neurology	9337654587	MBBS, MD
14	Dr Richu John	Gastroenterology	8925476418	MBBS, MD in Gastroenterology
15	Dr. Isel George	Radiology	7307645932	MBBS, MD in Radiology
16	Dr. Adriano Shine	Urology	9237654593	MBBS, MG Course
17	Dr. Gayathri Devi	Physicaltherapist	9988776628	3 Year Professional DPT Program
18	Elizabeth Jose	Psychiatrist	9337674693	MBBS, Diploma in Psychiatric Medicine(DPM)

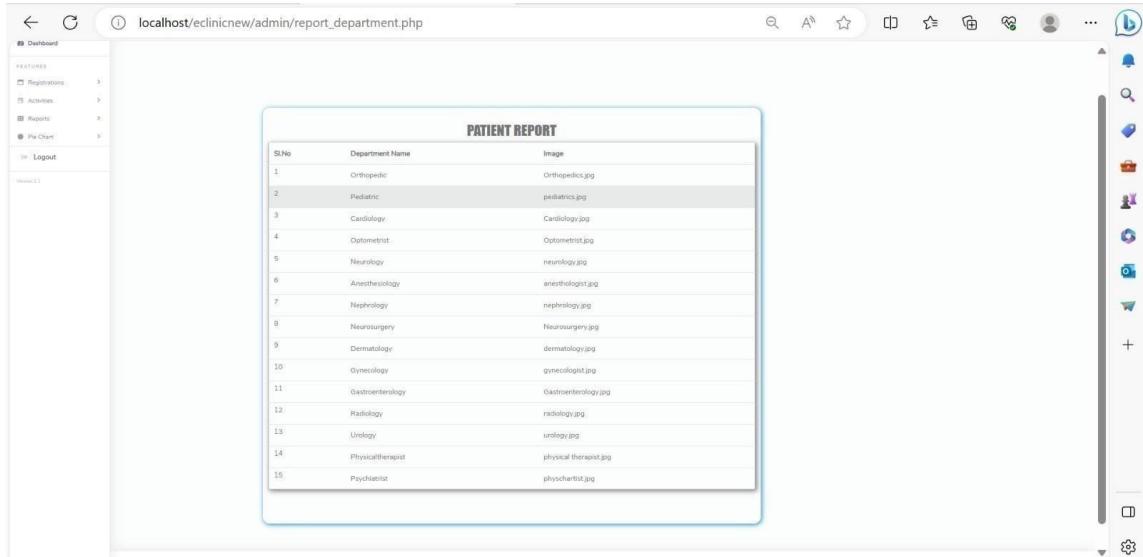
Fig 8.32 Doctor view



The screenshot shows a web-based administrative interface for a clinic. The main content area displays a table titled "PATIENT REPORT". The table has columns for Sl.No, Patient Name, E-Mail, Phone, Place, and OP Number. There are 21 rows of data, each representing a patient record. The interface includes a left sidebar with navigation links like Dashboard, Reports, Pie Chart, and Logout, and a right sidebar with various icons.

Sl.No	Patient Name	E-Mail	Phone	Place	OP Number
1	Anju	anju@gmail.com	8877334655	Pala	RE_9009
2	Shanty	rosepor@gmail.com	9912345678	Muttom	RE-1002
3	Anoush	anoush@gmail.com	9876543214	Pala	RE-1009
4	Rahul	rahul@gmail.com	7788656544	Pala	RE-1010
5	Anisha	roseanthosh@gmail.com	9988776655	Pala	RE-6789
6	Ginto	rossanthosh@gmail.com	9944332255	Pala	RE-1013
7	Alex	alex@gmail.com	9876543233	Marik	RE-1014
8	Areena	areenaberry572@gmail.com	9876543098	Nediyasala	RE-1015
9	Anirudh	anirudh@gmail.com	9128762453	Vazhthalai	RE-1016
10	Kiran	kiran@gmail.com	9988776633	Koothattukulam	RE-1017
11	Aneesha	aneesha@gmail.com	9988776655	Vazhthalai	RE-1018
12	Anna	anna@gmail.com	1234560987	Vazhthalai	RE-1019
13	Anne	anne@gmail.com	1234560987	Vazhthalai	RE-1020
14	Manisha	anna@gmail.com	9876543211	Nediyasala	RE-1021
15	Annu	anna27@gmail.com	995663425	Vazhthalai	RE-1022
16	Annu	anna27@gmail.com	995663425	Vazhthalai	RE-1022
17	Lya	lyla@gmail.com	9876543231	Marik	RE-1024
18	Areenasid	areenaberry572@gmail.com	9876543098	Vazhthalai	RE-1025
19	Arun	arun@gmail.com	9876543221	Manapalam	RE-1026
20	Ajay	ajay@gmail.com	9876543232	Nediyasala	RE-1027
21	Alan	alanenvyone@gmail.com	995663428	Nediyasala	RE-1028

Fig 8.33 Patient view



The screenshot shows a web-based administrative interface for a clinic. The main content area displays a table titled "PATIENT REPORT". The table has columns for Sl.No, Department Name, and Image. There are 15 rows of data, each representing a department record. The interface includes a left sidebar with navigation links like Dashboard, Reports, and Logout, and a right sidebar with various icons.

Sl.No	Department Name	Image
1	Orthopedic	Orthopedics.jpg
2	Pediatric	pediatrics.jpg
3	Cardiology	Cardiology.jpg
4	Ophthalmologist	Ophthalmologist.jpg
5	Neurology	neurology.jpg
6	Anesthesiology	anesthesiologist.jpg
7	Nephrology	nephrology.jpg
8	Neurosurgery	Neurosurgery.jpg
9	Dermatology	dermatology.jpg
10	Gynecology	gynecologist.jpg
11	Gastroenterology	Gastroenterology.jpg
12	Radiology	radiology.jpg
13	Urology	urology.jpg
14	Physical therapist	physical therapist.jpg
15	Psychiatrist	psychiatrist.jpg

Fig 8.34 Department view

The screenshot shows a web-based administration interface for 'RuangAdmin'. The left sidebar contains a 'Dashboard' button and a 'FEATURES' section with 'Registrations', 'Activities', 'Reports', and 'Pie Chart' options. A 'Logout' button is also present. The main content area is titled 'PATIENT PAYMENT REPORT' and displays an 8x6 grid table of patient payment details. The columns are labeled: SL.No, Book ID, Doctor Name, Patient Name, OP Number, and Amount. The data is as follows:

SL.No	Book ID	Doctor Name	Patient Name	OP Number	Amount
1	72	Dr. Anu Mathew	Aneesha	RE-1018	500
2	77	Dr. Albin K John	Arun	RE-1026	1000
3	78	Dr. Amal Kurian	Arun	RE-1026	200
4	71	Dr. Abhinav Jayan	Ajay	RE-1027	200
5	74	Dr. Abhinav Jayan	Ajay	RE-1027	200
6	75	Dr. Albin K John	Ajay	RE-1027	1000
7	73	Dr. Mathew John	Alan	RE-1028	1000
8	76	Dr. Albin K John	Alan	RE-1028	1000

Fig 8.35 Patient payment report