

# **DOCTOR APPOINTMENT SYSTEM**

PROJECT REPORT SUBMITTED IN PARTIAL FULFILMENT OF THE  
REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

**BACHELOR OF COMPUTER APPLICATIONS**

**To**

**MARIAN COLLEGE KUTTIKKANAM [AUTONOMOUS]**

**Affiliated to**

**MAHATMA GANDHI UNIVERSITY, KOTTAYAM**

**By**

**ROSHAN SHIBI**

(Reg.No:17UBC149)

**GUIDED BY**

**Ms BENYMOL JOSE**



**DEPARTMENT OF COMPUTER APPLICATIONS**

**MARIAN COLLEGE KUTTIKKANAM [AUTONOMOUS]**

**PEERMADE – 685531**

**MARCH, 2020**

## **DECLARATION**

I, **ROSHAN SHIBI** [Reg.No 17UBC149] certify that the Main project report entitled **“DOCTOR APPOINTMENT SYSTEM”** is an authentic work carried out by me at Marian College Kuttikkanam [Autonomous]. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Signature of the Student :

Name of the Student : Roshan Shibi

Date:

## BONAFIDE CERTIFICATE

This is to certify that this project work entitled “**DOCTOR APPOINTMENT SYSTEM**” is a bonafide record of work done by **ROSHAN SHIBI [Reg.No17UBC149]** at Marian College Kuttikkanam(Autonomous) in partial fulfilment for the award of **Degree of Bachelor of Computer Applications of Mahatma Gandhi University, Kottayam.**

This work has not been submitted elsewhere for the award of any other degree to the best of our knowledge.

Head of the Department

**Dr Rajimol A**

Dept. of Computer Applications

Marian College Kuttikkanam [Autonomous]

Peermade - **685531**

Internal Guide

**Ms Benymol Jose**

Dept. of Computer Applications

Marian College Kuttikkanam

Peermade – **685531**

Submitted for the Viva-Voce Examination held on

**Department Seal**

**External Examiner**

## ACKNOWLEDGEMENT

“Gratitude is a feeling which is more eloquent than words, more silent than silence.” In undertaking this project I needed the direction, assistance and cooperation of various individuals and organizations, which is received in abundance with the grace of God. Without their unconstrained support, this project could not have been completed.

If words are considered as the symbol of approval and token of acknowledgement, then let the following words play the heralding role of expressing my gratitude.

I wish to acknowledge our sincere gratitude to our Manager **Rev Fr James Kozhimala** and Principal **Rev Dr Roy Abraham P, Marian College Kuttikkanam (Autonomous)** for all their efforts and administration in educating me in this premier institution.

I extend our gratitude to **Dr Rajimol A**, Head, Department of Computer Applications, who is a constant source of inspiration and whose advice helped me to complete this project successfully.

I express our deep sense of gratitude to our internal project guide, **Ms Benymol Jose**, for her profound guidance for the successful completion of this project.

I extend my sincere gratitude to **Ms. Rinimol R** for the fruitful training and technical assistance throughout the project development, without which this project would not have become a reality.

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

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

Roshan Shibi

# **ABSTRACT**

# **ABSTRACT**

Doctor appointment system is a smart web application that provides patients or any user an easy way of booking a doctor's appointment online. This is a web application project in asp.net that overcomes the issue of managing and booking appointments according to user's choice or demands. This system is very much useful to save the patients timing and also easy to analyze the doctor availability.

This system is divided into 3 modules :Admin , Doctor and Patient. Admin can view the registered doctors and patients.Admin can add or delete doctors, patients and departments. Doctors can register by giving his necessary details like timings, fee, category, etc. After successful registration, the doctor can log in by giving username and password. The doctor can view the booking request by patients and if he accepts the patient requests the status will be shown as booking confirmed to the patient. The patients must be registered and log in to book a doctor basing the category.Patients can search by doctor name,department, date and time etc.

## Contents

ACKNOWLEDGEMENT .....	iv
ABSTRACT .....	vi
1. INTRODUCTION .....	3
1.1 ABOUT THE PROJECT .....	3
1.1.1 THE PURPOSE AND SCOPE.....	3
2. SYSTEM ANALYSIS.....	6
2.1 PROBLEM DEFINITION.....	6
2.1.1 LIMITATIONS OF THE EXISTING SYSTEM .....	6
2.1.2 ADVANTAGES OF PROPOSED SYSTEM .....	6
2.2 FEASIBILITY ANALYSIS .....	6
2.2.1 ECONOMIC FEASIBILITY.....	7
2.3 RECOMMENDED IMPLEMENTATIONS .....	7
3. SOFTWARE REQUIREMENT SPECIFICATION .....	10
3.1 INTRODUCTION .....	10
3.2 PURPOSE .....	10
3.3 SCOPE.....	10
3.3.1 SYSTEM STATEMENT OF SCOPE.....	10
3.4.1 USER CHARACTERISTICS .....	11
3.5 STATED REQUIREMENTS .....	11
3.5.1 GENERAL REQUIREMENTS.....	11
3.5.2 INPUTS.....	12
3.5.3 PROCESSING .....	12
3.6 EXTERNAL INTERFACE REQUIREMENTS .....	13
3.6.1 USER INTERFACE .....	13
3.6.2 HARDWARE INTERFACES.....	13
3.6.3 SOFTWARE INTERFACES .....	13
4. SYSTEM DESIGN.....	15
4.1 INTRODUCTION .....	15
4.2 DESIGN METHODOLOGY .....	16
4.2.1 INPUT DESIGN .....	16
4.2.2 OUTPUT DESIGN .....	17
4.2.3 CODE DESIGN.....	18
4.2.4 DATABASE DESIGN .....	18
4.3 MAJOR DESIGN CONSTRAINTS .....	19

4.3.1 GENERAL CONSTRAINTS .....	19
4.4 SYSTEM ARCHITECTURE AND PROCESS FLOW .....	20
4.5 MODULE DETAILS .....	22
4.6 TABLE DESIGN.....	23
4.7 PERFORMANCE ISSUES .....	26
4.8 SECURITY ISSUES.....	26
5. CODING .....	28
5.1 INTRODUCTION .....	28
5.2 SELECTION OF SOFTWARE.....	28
5.3.1 CODING STANDARDS.....	29
6. TESTING.....	31
6.1 INTRODUCTION .....	31
6.2 TESTING .....	31
6.2.2 TEST PLAN.....	31
6.3 TESTING METHODS .....	32
6.3.1 BLACK BOX TESTING .....	33
6.3.2 WHITE BOX TESTING .....	33
6.3.3 OUTPUT TESTING .....	35
6.3.4 USER ACCEPTANCE TESTING.....	35
6.4 IMPLEMENTATION .....	36
7. MAINTENANCE AND ENHANCEMENT.....	38
7.1 MAINTENANCE .....	38
7.2 ENHANCEMENTS.....	38
8. CONCLUSION .....	40
9. BIBLIOGRAPHY.....	42
APPENDIX.....	44



# INTRODUCTION

# **1. INTRODUCTION**

## **1.1 ABOUT THE PROJECT**

### **EXISTING SYSTEM**

In the existing system the patient needs to visit the doctor for booking. We need to wait and the booking will be done manually. But this system is very hard to handle as there is no option to save the data of patients in a systematic manner.

Any clinic that handles patient records are at a risk of wasting too much time and money on patient scheduling. Since appointment books are limiting and time consuming, the more cancellations and scribbles the more this process confuse and frustrate staff who are managing these. So it is not worthwhile to continue with primitive paper based scheduling system.

### **PROPOSED SYSTEM**

The main objective of the proposed system is to eliminate the limitations of the existing decentralised system. The system eliminates these shortcomings by introducing a centralised application for all the procedures related to a doctor appointment.

The Proposed System consists of functions divided into three types of users: The administrator, doctors and patients. The Admin handles the bookings, doctors, patients, departments and has access to all information. Doctors can manage the bookings, manage patients and patients can book for the doctors by the availability .

#### **1.1.1 THE PURPOSE AND SCOPE**

The purpose of the Doctor appointment system is to automate the existing manual system by the help of computerized equipments and full-fledged computer software, fulfilling their requirements, so that their valuable data can be stored for a longer period with easy accessing and manipulation of the same. The goals of this project are:

- Minimize time and work put into record keeping.
- Minimize the use of paper.
- Maintain an easily searchable and manageable database.

- To satisfy user requirements.
- Remove duplicate data.
- Make calculations on data easier and faster.
- Make it easier for patients to take an appointment.
- Be easy to understand by the user and operator.

# **SYSTEM ANALYSIS**

## **2. SYSTEM ANALYSIS**

### **2.1 PROBLEM DEFINITION**

The current system does use digital solutions to real world problems. But it is not refined. The system is not structured. Even when records are kept digitally for each events, it is done inefficiently and kept of the internet. This makes it harder for handling events and creating innovative event management tactics. The current unstructured collection of subsystems uses third party software like Microsoft Office for keeping records. Most of the financial record keeping is done by the use of pen and paper.

In order to overcome the limitations of the existing system mentioned above, the Doctor Appointment System is proposed. It'll replace the dated system with an enhanced one. Keeping the database online and compiling all these chaotically scattered subsystems into one is the solution to this problem.

#### **2.1.1 LIMITATIONS OF THE EXISTING SYSTEM**

- The existing system is disorganised and chaotic.
- There's no central software for handling the processes.

#### **2.1.2 ADVANTAGES OF PROPOSED SYSTEM**

1. The system avoids redundancy by the use of several type of validation that is the system is enhanced
2. Quick access and processing is the main advantage that forces as to implement the proposed system.
3. The main alteration between the existing system and the new automated system lies in the specialty which reduces the time consumption in an appropriate manner.
4. The system will reduce the amount of paper work require.

### **2.2 FEASIBILITY ANALYSIS**

Feasibility study is a test of a system proposal according to its workability, ability to meet user needs and effective use of resources. The objective of feasibility is not to solve the problem but to acquire a sense of its scope. The main aim of the feasibility study is to test the technical, social and economic feasibility of the system. The feasibility study can be classified into the following categories:

- Economic Feasibility

- Technical Feasibility
- Operational Feasibility

### **2.2.1 ECONOMIC FEASIBILITY**

Economic feasibility is an important factor. Since the existing system is manual on the feasibility for wrong data entry is higher and consumes a lot of time and can occur errors. But the proposed system aims at processing of information's efficiently, thus saving the time. The new system need only a system and which is already available therefore the cost is negligible. Proposed system use validation check so there is no errors. Even though an initial investment has to be made on the software and the hardware aspects, the proposed system aims at processing of information's efficiently. Thus the benefits acquired out of the system are sufficient enough for the project to be undertaken. So the proposed system is economically feasible.

### **2.2.2 TECHNICAL FEASIBILITY**

Technical feasibility deals with hardware as well as software requirements and to what extend it can support the proposed system. The software required is Visual c# and Microsoft SQL server. If the necessary requirements are made available with the system, then the proposed system is said to be technically feasible.

### **2.2.3 OPERATIONAL FEASIBILITY**

The proposed system offers greater of user friendliness combined with greater processing speed. Since the processing speed is very high compared with manual system on that management can take timely actions depending on information's obtained. Proposed system is very easy to handle. Since the workload is also reduced the college authority convenience that the project is operationally feasible

## **2.3 RECOMMENDED IMPLEMENTATIONS**

Two principle sources of data are:

1. Written documents
2. Data from the persons, who are involved in the operation of the system under study.

The different fact finding techniques are:

1. Data Carriers
2. Questionnaires
3. Personal interviews
4. Observations

### **Data carriers**

Data carriers are the best to collect details. So we use a form for collecting patient details. From this form the administrator gets all the details of the patient for eg: Name , Phone , Address etc. This was the primary source of fact finding used for this project.

### **Questionnaires**

Questionnaires are best methods to probe data out of the patients. In this case, questionnaires were not used for data collection as the administration was small in number and they could be asked questions in a more effective interview.

### **Personal interviews**

Personal interview help to understand more about the person whom the data is associated. We had interviewed the head of technical department in the hospital and we get a clear picture of recording of process of information collected, registration of patient and doctor , appointment booking process , management of databases etc.

### **Observations**

A person can understand a lot about a system just by observing it. We had understood the processes done in the clinics. We had observed that there is a lot of paper work is needed for the existing system, which is time consuming and also workload is very high.

# **SOFTWARE REQUIREMENT SPECIFICATION**



## **3. SOFTWARE REQUIREMENT SPECIFICATION**

### **3.1 INTRODUCTION**

Requirements specification is the starting step for the development activities. It is currently one of the weak areas of software engineering. During requirement specification, the goal is to produce a document of the client's requirements. This document forms the basis of development and software validation. The basic reason for the difficulty in software requirements specification comes from the fact that there are three interested parties- the client, the end users and the software developer.

### **3.2 PURPOSE**

The origin of most software systems is in the need of a client, who either wants to automate an existing manual system or desires a new software system. The software system itself is created by the developer. Finally, the completed system will be used by the end users. Thus, there are three major parties interested in a new system: the client, the users and the developer. A basic purpose of software requirements specification is to bridge the communication gap. SRS is the medium through which the client and user needs are accurately specified. Indeed, SRS forms the basis of software development. A good SRS should satisfy all the parties, something very hard to achieve, and involves trade-offs and persuasion.

Another important purpose of developing an SRS is helping the clients understand their own needs. Advantages are:

- An SRS establishes the basis for agreement between the client and the supplier on what the software product will do.
- An SRS provides a reference for validation of the final product
- A high quality SRS is a prerequisite to high-quality software.
- A high quality SRS reduces the development cost.

### **3.3 SCOPE**

#### **3.3.1 SYSTEM STATEMENT OF SCOPE**

Doctor appointment system is a web application that provides an easy way of booking a doctor's appointment online. This overcomes the issue of managing and booking

appointments according to user's choice or demands. This system is very much useful to save the patient's timing and also easy to analyse the doctor's availability.

### **3.4 TECHNICAL OVERVIEW**

#### **3.4.1 USER CHARACTERISTICS**

- The system can be accessed by one admin, several doctors and all registered patients
- The Admin can add doctors, add departments and manage patients. Only admin is allowed to edit these details.
- The Doctors can handle all the booked patients.

### **3.5 STATED REQUIREMENTS**

#### **3.5.1 GENERAL REQUIREMENTS**

The following general requirements were laid out for our project:

- A way in which login process taken place.
- A way in which patient details can be added.
- A way in which information can be provided to the patients.
- A way in which all data could be stored electronically.

The system has 4 main modules divided by users:

#### **1. LOGIN**

- Email id and password is required for the login process. There is login option for the user as well as the admin.
- There is login option for the admin.
- Email id is same for every time.
- Password can also contain both upper case and lower case alphabetics, numbers and special characters.
- Admin will redirected to the admin panel when login with the predefined id and password in the same login form itself.
- Users will be directed to the home page of site when logged in.

## **2. SIGN UP**

The new users need to register in order to appoint for a doctor. This includes several fields:

- Name
- Phone
- Email id
- Create password
- Gender
- Phone

## **3. DOCTOR**

- Panel for all doctors.
- Doctors can approve the appointment from the patient.
- Details of patients can be seen.
- Doctors can add/edit time schedule.

## **4.ADMIN**

- Admin can edit the details of both doctors and patients.
- Admin can add or delete departments.
- Doctor signup process is done by the admin.

### **3.5.2 INPUTS**

The Doctor Appointment System will take booking details from the users. It also takes user feedbacks as inputs.

### **3.5.3 PROCESSING**

- All types of validation for the data entry are carried out
- Specific formats are allowed to the data entry fields
- Booking for the doctor.

## **3.6 EXTERNAL INTERFACE REQUIREMENTS**

### **3.6.1 USER INTERFACE**

All user interfaces will be GUI interfaces. Interfaces are designed to use with ease and without any confusion. The user interface shall have a pleasing appearance and high functionality.

- Suitable design and pleasing colours are selected to design the window page to make the users comfortable to operate the software.
- Component like textboxes, combo boxes and buttons are selected to make it easy to fill with appropriate data.

### **3.6.2 HARDWARE INTERFACES**

The System needs a computer or any other smart phones or devices with network availability to browse into the site. No other external hardware is required.

#### **HARDWARE SPECIFICATION**

Processor	:	Intel Pentium or higher
RAM	:	256 MB or higher
Hard disk drive	:	100MB is required on disk
Keyboard	:	Standard QWERTY keyboard

### **3.6.3 SOFTWARE INTERFACES**

#### **SOFTWARE SPECIFICATION**

Operating system	:	Windows 10
DBMS	:	Microsoft SQL Server 2013
Tool used	:	.NET

# **SYSTEM DESIGN**

## **4. SYSTEM DESIGN**

### **4.1 INTRODUCTION**

The purpose of design phase is to plan a solution of the problem specified by the analysis phase. This phase is the first step in moving from the problem domain to solution domain.

System design describes the desired features and operation in detail, including screen layouts, business rules, process diagrams, pseudo code and other documentation.

In this phase, the software's overall structure and its nuances are defined. In terms of the client/server technology, the number of tiers needed for the package architecture, input design, output design, the database design, the data structure design etc. are all defined in this phase. Analysis and design are very crucial in the whole development cycle. Any glitch in the design phase could be very expensive to solve in the later stage of the software development. So much care is taken during this phase.

The logical system of the product and the physical characteristics of the system are designed during this phase. The operating environment is established, major resources. Everything requiring user input or approval must be documented and reviewed by the user. The physical characteristics of the system are specified and a detailed design is prepared.

The subsystem identified during design are used to create a detailed structure of the system. Each subsystem is partitioned into one or more design units or modules. Detailed logic specifications are prepared for each software module. The logic of the module is usually specified in a high-level design description language, which is independent of the target language in which the software will eventually be implemented.

A good design must consider:

- Prompt: should be simple and clear to intuitively lead the user to an expected outcome.
- Memory load: Studies show that, under normal circumstances, users have a short-term memory of approximately six words. Ideally, the number of choices of users to select should be four or less. Otherwise, callers become confused and forget the choices presented to them.

- Service reaches ability: It is not pleasant for a person to go through a large number of steps before he reaches a service. Users start to get impatient with more than five steps. Minimize the number of steps a caller must take to reduce frustration.
- Navigation: Provide a way to navigate back and forth between various dialogue steps. The user should be able to go to different parts of the dialog easily.
- Phonetic similarity: Provide a clear set of choices for user to select. Avoid choices with similar pronunciation.
- Error handling: Humans make mistakes. Graceful error handling decreases dependency on operators.
- User update: Let the user know what is going on and keep him engaged.

For the general design one or more potential design are proposed and broadly sketched. Then these alternatives are presented to the users, who choose the design that best suits their requirements while staying within the project constraints.

Detailed design stage is specification for the user interface, database, programs, hardware, and training and system documentation. Several structured techniques are used during the design phase. To design the software components, the designer transforms the automated processes in the physical data flow diagram into a program structure chart, which decomposes software processes into detailed modules and shows control path between modules.

## **4.2 DESIGN METHODOLOGY**

### **4.2.1 INPUT DESIGN**

In the input design, the user oriented inputs are converted into computer recognizable format. The collection of input data is the most expensive part of the system in terms of equipment used, time and number of users involved. Input design is the process of converting user oriented inputs to a computer based format. The goal of designing input data is to make data entry as easy, logical and free from errors as possible.

Input design is the link between the information system and the users and the step necessary to put transaction data into a usable form for processing. Instructing the computer to read data from a written printed document can activate the activity of putting data into the computer for processing or it can occur by keying data directly into the system. The design of input focuses on controlling the amount of input required,

controlling the errors, avoid delay extra steps, and keeping the process simple. System analysis decides the following input design details:

- What data to input
- What medium is to use
- How the data is arranged and coded
- Data items and transaction needing validation to detect error occurs.

Activities performed as part of input design are:

- Data recording
- Data verification
- Data conversion
- Data validation
- Data correction

#### **4.2.2 OUTPUT DESIGN**

Output design is a process that involves designing necessary outputs that have to be used by various users according to requirements. Designing computer should proceed in well thought out manner. The term output means any information produced by the information system whether printed or displayed. When analyst design computer output they identified the specific output that is needed to meet the requirement.

Computer is the most important source of information to the users. Efficient intelligent output design should improve the system relationship with the user and help in decision making. When designing the output, 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 acceptable format

The output design is the key to the success of any system. Output is the key between the user and the sensor. The output must be concerned to the system's working, as it should. Output design consists of displaying specification and procedures as data



presentation. User is never left with the confusion as to what is happening without appropriate error and acknowledges message being received.

### **4.2.3 CODE DESIGN**

The coding step is a process that transform design into programming language. It translates a detail design representation of software into a programming language realization. The code design should be done in such a way that the lines of code used in the software should be minimum for the specified design of the solution. The coding should be in modularized manner.

When code is placed in a module, one may hide it from view and give those executable statements a name (the name of the function or procedure). Information hiding is a good thing when it enhances the understanding of a program by letting to focus on a higher level of abstraction. Information hiding is a bad thing when it obscures one's understanding of a program. This usually happens when the name for the module is not chosen accurately.

In this software, the modularized approach is used. Different functions are created for different operations. The name of the module is chosen such a way that it describes what it does, i.e. the name gives the action performed by the module

### **4.2.4 DATABASE DESIGN**

The details about the relevant data that came into lay in the system are identified according to the relationship the tables are designed by following the standard database design methods. The dative for each data in the table is defined. For optimum design of database to have better response time, to have data integrity, to avoid the redundancy and for security of the database tables created and analysed.

A database system can be defined as a representation of an information system in a computer. The general theme behind a database is to handle information as an integrated whole. 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 user. In database design, several specific objectives are considered:

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

The scheme is the view that helps us the DBMS decide what data in storage it should act upon as requested by the application program. The subschema is concerned with a relatively small part of scheme. In database design, several views of data must be considered along with the persons who use them. The logical view is what the data look like, regardless of how they stored. The physical view is the way data exists in physical storage. It deals with how data are stored, accessed or related to other data in storage. The logical view are the users view the programmer's view and the overall logical view, called a schema.

This project has used a main database having different tables, based on which the operations can perform well.

## **4.3 MAJOR DESIGN CONSTRAINTS**

### **4.3.1 GENERAL CONSTRAINTS**

#### **Time**

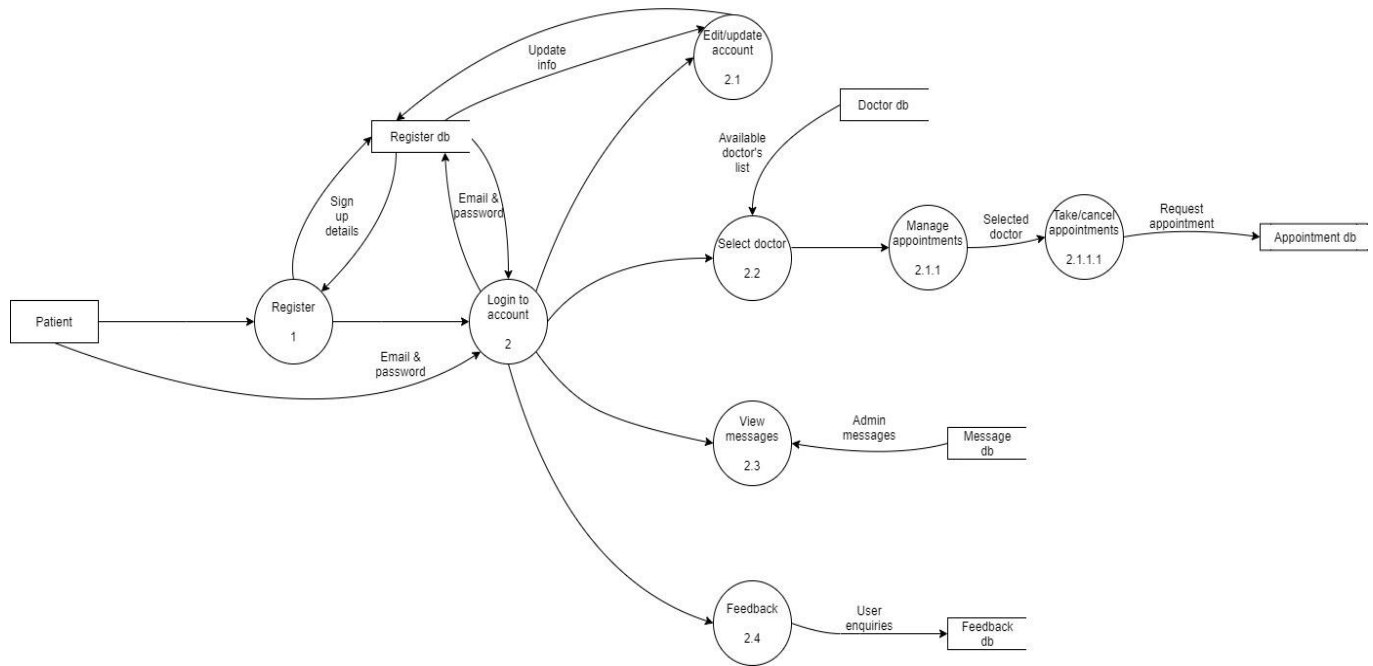
Time is so far the biggest restriction or constraint for this project. Only one semester worth of time is available for the completion of this project. The time constraint must be kept in mind throughout the project life cycle.

#### **Employee Skills**

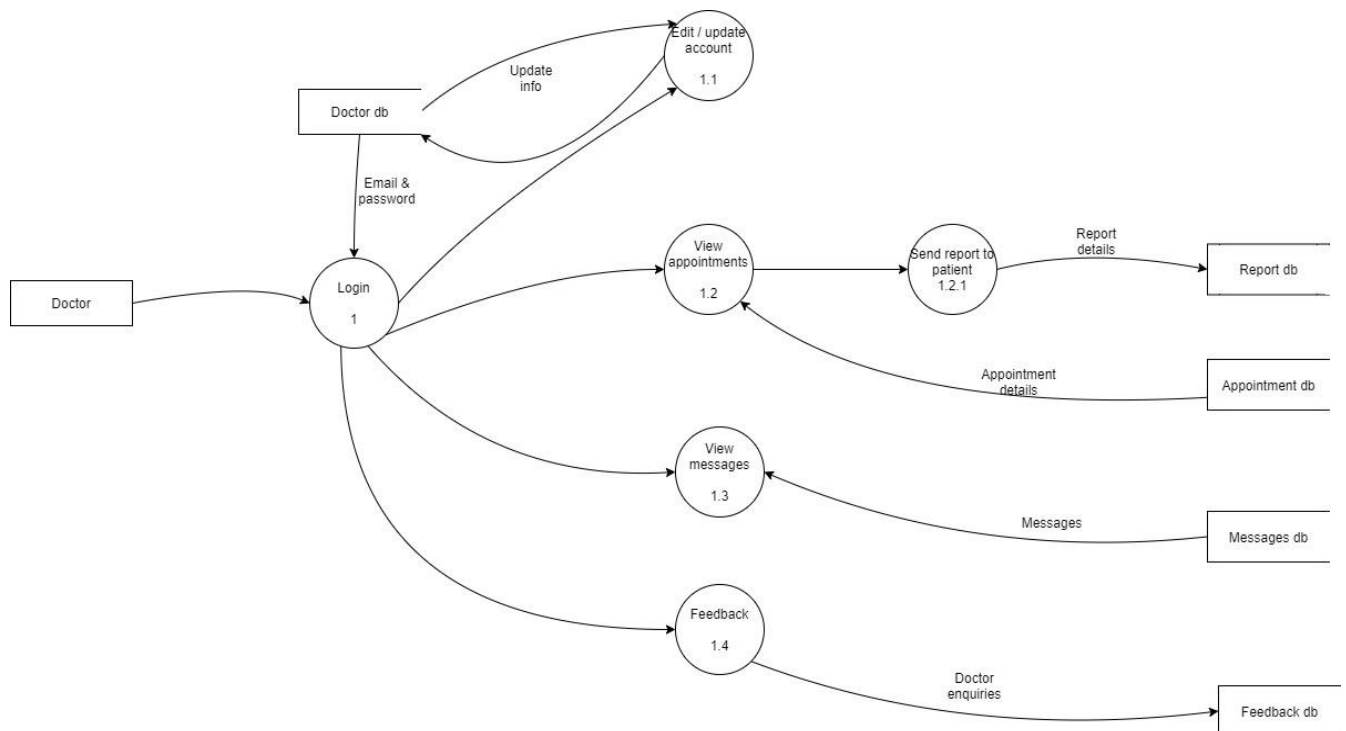
Employees programming and design skills is also one of the restriction. It does not have as big of an impact on the project.

## 4.4 SYSTEM ARCHITECTURE AND PROCESS FLOW

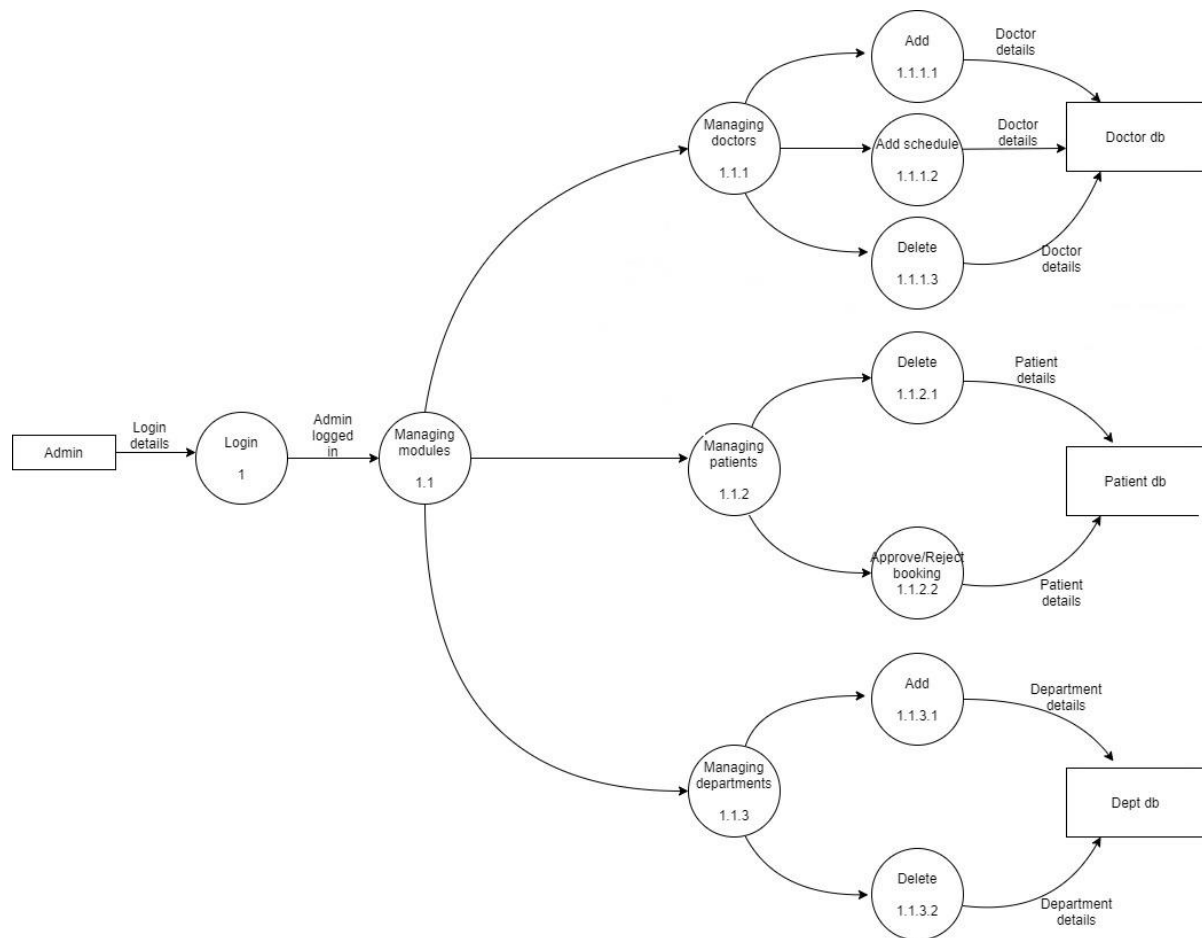
### Patient



## Doctor



## Admin



## 4.5 MODULE DETAILS

The system has 3 main modules divided by users and 3 separate modules for security and data retrieval.

### 1. Admin

This module has the responsibility of adding/editing doctor, patient and department details.

## 2. Doctor

The doctor can view the patients list that booked for that particular doctor. Doctor can also schedule the appointment by updating the profile.

## 3. Patients

The patients can book for a doctor in an available date of doctor. Patient can cancel or view the current bookings.

## 4. Login

The login module ensures the security of the system. It takes the Email-ID and password and verifies it against the database.

## 5. Appointment

In this module the patient can book for the particular doctor in the available date.

## 6. Sign up

The new users need to sign up to the website in order to book for a doctor.

## 7. Feedback

Patients and doctors can send feedback to the admin about the website if any enquiries or suggestions.

## 4.6 TABLE DESIGN

A table is a collection of data about a specific topic. It makes data more efficient and reduces data entry errors. There are several tables used in this project.

### 1. Table Name: tbl\_signup

This table is to store Sign up details.

Sl. No	Field Name	Data Type	Description
1	Id	int	Id for patient
2	Name	nvarchar(max)	Name of the patient
3	Age	nvarchar(max)	Age of the patient
4	Phone	nvarchar(max)	Phone no. of patient

5	Gender	nvarchar(max)	Gender of patient
6	Email	nvarchar(max)	Email of patient
7	Createpassword	nvarchar(max)	Create password for patient

## 2.Table Name: tbl\_doctorregister

This table is to store the details of doctor

Sl. No	Field Name	Data Type	Description
1	Id	Int	Id of doctor
2	Name	nvarchar(max)	Name of doctor
3	Address	nvarchar(max)	Address of doctor
4	Phone	nvarchar(max)	Phone no. of doctor
5	Email	nvarchar(max)	Email of doctor
6	Deptname	nvarchar(max)	Department of doctor
7	Createpassword	nvarchar(max)	Create password for doctor
8	Timein	nvarchar(max)	Opening time
9	Timeout	nvarchar(max)	Closing time
10	Gender	nvarchar(max)	Gender of doctor
11	Age	nvarchar(max)	Age of doctor
12	Location	nvarchar(max)	Location of clinic

## 3.Table Name: tbl\_dept

This table is to store the departments.

Sl. No	Field Name	Data Type	Description
1	Id	int	Id of department
2	Deptname	nvarchar(max)	Department name

#### 4.Table Name: tbl\_booking

This table is to store the booking details.

Sl. No	Field Name	Data Type	Description
1	Id	int	Id of the appointment
2	Memid	int	Id of the booked patient
3	Doctorid	int	Id of the doctor whom the patient is booked
4	Time	nvarchar(max)	Time of booking
5	Status	nvarchar(max)	Status of booking
6	Date	nvarchar(max)	Date of booking
7	Bdate	nvarchar(max)	Time schedule
8	Bookingdate	nvarchar(max)	Consulting date

#### 5.Table Name:tbl\_feed

This table is to store the feedbacks of patients.

Sl. No	Field Name	Data Type	Description
1	Id	int	Id of the patient
2	Name	nvarchar(max)	Name of the patient
3	Email	nvarchar(max)	Email of patient
4	Message	nvarchar(max)	Feedback of patient

#### 6.Table Name:tbl\_feeddoctor

This table is to store the feedbacks of patients.

Sl. No	Field Name	Data Type	Description
1	Id	int	Id of the doctor



2	Name	nvarchar(max)	Name of the doctor
3	Email	nvarchar(max)	Email of doctor
4	Message	nvarchar(max)	Feedback of doctor

#### **7.Table Name:tbl\_notification**

This table is to store the messages that admin send to doctors and patients.

<b>Sl. No</b>	<b>Field Name</b>	<b>Data Type</b>	<b>Description</b>
1	Id	int	Id of the message
2	Message	nvarchar(max)	Message that admin want to send
3	Type	nvarchar(max)	To which user the admin want to send

## **4.7 PERFORMANCE ISSUES**

This system should have at least 256 MB of RAM. The OS must be Windows XP or higher.

## **4.8 SECURITY ISSUES**

Only authorised access is permitted to this software. The login is protected by username and password.

# **CODING**

## **5. CODING**

### **5.1 INTRODUCTION**

Coding section is where the magic happens. All the planning and the designing done in the previous sections come to life in this section. After this section can only the programmer enjoy the result of his/her hard work when he/she runs the program for the first time.

### **5.2 SELECTION OF SOFTWARE**

#### **Visual Studio 2013**

Visual Studio 2013 is an event driven programming language and integrated development environment (IDE) developed by Microsoft. It is a fairly easy programming language to use. Graphical user interface programs can be easily created using this IDE. Many components are provided in the IDE which users can use to create programs. Users can also build their own controls and add them to the program. Therefore, this language is extensible and the program written in it can be later updated using new controls.

#### **SQL Server 2008**

**SQL (Structured Query Language)** is a domain-specific language used in programming and designed for managing data held in a relational database management system(RDBMS), or for stream processing in a relational data stream management system(RDSMS). It is particularly useful in handling structured data where there are relations between different entities/variables of the data. SQL offers two main advantages: first, it introduced the concept of accessing many records with one single command; and second, it eliminates the need to specify how to reach a record, e.g. with or without an index.

### **5.3 CODING PHASE**

The goal of the coding or programming phase is to translate the design of the system produced during the design phase into code in a given programming language, which can be executed by a computer and that performs the computation specified by the design.

The coding phase affects both testing and maintenance profoundly. The coding phase does not affect the structure of the system; it has great impact on the internal structure of modules, which affects the testability of the system.

The goal of the coding phase is to produce clear simple programs. The aim is not to reduce the coding effect, but to program in a manner so that testing and maintenance costs are reduced. Programs should not be constructed so that they are easy to write; they should be easy to read and understand. Reading programs is a much more common activity than writing programs. Hence, the goal of the coding phase is to produce simple programs that are clear to understand and modify.

### **5.3.1 Coding Standards**

The standard used in the development of the system is Microsoft Programming standards. It includes naming conventions of variables, constants and objects, standardized formats for labelling and commenting code, spacing, formatting and indenting.

#### **Naming Conventions**

The controls are prefixed to indicate their functions. The webforms are prefixed with the function they perform like add/edit/delete, textboxes are prefixed with txt, command buttons with cmd, label boxes with lbl, list boxes with li, comboboxes with cb, Date Time Pickers with DTP, Gridview control with dgv and so on.

#### **Labels and comments**

The functions of each control is labelled clearly in the GUI. The code also includes comments so that other developers using the source code in future might understand the module functions better. The functions in the code behind file and the class module are clearly labelled with the function they perform. The condition checks are also labelled.

# **TESTING & IMPLEMENTATION**

## **6. TESTING**

### **6.1 INTRODUCTION**

Software testing is a critical element of software quality assurance and represents the ultimate review of specifications design and coding. Testing presents an interesting anomaly for the software. Testing is a quality measure process, which reveals the errors in the program. During testing, the program is executed with a set of test cases and the output of the program for the test cases is evaluated to determine if the program is performing as it is expected. Testing plays a very critical role in determining the reliability and efficiency of the software and it is a very important stage in software development.

### **6.2 TESTING**

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based systems. Although each tests have a different purpose, all work to verify that all system elements have been properly integrated and perform allocated functions.

System testing is done in order to ensure that the system developed doesn't fail at any point. Before implementations, the system is tested with experimental data to ensure that it will meets the specified requirements, special tests data are input for processing and results examined.

#### **6.2.2 TEST PLAN**

- **Preparation of Test Data**

Taking various kinds of test data does the testing. Preparation of test data plays a vital role in the system testing. After preparing, the test data the system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and correction are also noted for future use.

Two kinds of test data were collected and used:

- **Using live test data**

Live test is those that are actually extracted from organization files. After a system is partially constructed, programmers or analyst often ask users to key in a set of data from their normal activities. Then, the system person uses this data as a way to

partially test the system. In order instance, programmers or analysts extract a set of live data from the files and have entered themselves.

- **Using artificial test data**

Artificial test data are created solely for test purpose, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information system department, make possible the testing of all login and control paths through the program.

The most effective test program use artificial test data generated by person other than those who wrote the program.

In this project invalid data was entered to test whether the program would break or not. These invalid data entries were randomly generated using random people. Many people were given the software for testing the program. They use gibberish values to test if every validation holds strong.

## **6.3 TESTING METHODS**

Testing is generally done at two levels-testing of individual modules and testing of the entire system. During system testing, the system is used experimentally to ensure that the software does not fail that is, that it will run according to its specifications and the results examined. A limited number of uses may be allowed to use the system so analysis can see whether they use it in unforeseen ways. It is preferable to discover any surprise before the organization implements the system and depends on it.

Testing is done throughout system development at various stages. It is always a good practice to test the system at many different levels at various intervals, that is, sub systems, program modules as work progresses and finally the system as a whole. During testing the major activities are concentrated on the examination and modification of the source code. Usually, this testing is to be performed by the person other than the person who has really coded it. This is done in order to ensure more complete and unbiased testing for making the software more reliable

There are two types of testing:

- Black box testing
- White box testing

### **6.3.1 BLACK BOX TESTING**

The concept of the black box is used to represent a system whose inside workings are not available for inspection. In a black box, the test item is treated as “black”, since its logic is unknown; all that is known is what goes in and what comes out, or the input and output.

### **6.3.2 WHITE BOX TESTING**

In white box testing, the internal logic of the modules is considered. Following levels of testing are performed for the developed project:

#### **6.3.2.1 Unit testing**

This involves the tests carried out on modules, which make up a system. This is also called as program testing. Unit testing focuses on the modules, independent of one another, to locate errors. The program should be tested for correctness of logic applied and should detect errors in coding. Before proceeding one must make sure that all the modules are working independently.

The modules present in Doctor Appointment System were tested independently using live and artificial data to check if they could act truly independent of one another. Though the data entered from another module is retrieved, the functions and logic of each module is truly independent.

The admin module was created and tested first because this is the module which is least coupled with other modules. It shares less features with the other module. Then the staff and student modules were tested.

#### **6.3.2.2 System Testing**

The system testing is conducted on a complete, integrated system to evaluate the system's compliance with its specified requirement. It falls within scope of black box testing so no knowledge of inner design or logic is needed. As a rule, system testing takes, as its input, all of the integrated software components that have passed integration testing and also the software system itself integrated with any applicable hardware system. The purpose of the integration testing is to detect any inconsistencies between software units.

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commence. The logical design and the physical design should be thoroughly and continually examined on paper ensure that they will work when implemented.



System testing was implemented in this project with the final product. Various invalid logins were tested to check the integrity of the system. Prevention methods for the SQL injection vulnerability was also tested. The three types of users and their granted privileges were tested and verified. Also there were tests to verify if all the functionalities specified in the system requirement specification was indeed present in the system for all modules.

#### **6.3.2.3 Integration testing**

Integration testing is a systematic technique for constructing the program structure, while at the same time conducting tests to uncover errors associated with interfacing. This is the program is constructed and tested in small segments, which makes it easier to isolate and the following common types of integration problems may be observed:

- Version mistakes
- Data integrity violations
- Overlapping function
- Resource problems especially in memory handling
- Wrong type of parameter in function calls

#### **6.3.2.4 Validation testing**

At the culmination of the integration testing, the software was completely assembled as a package, interfacing errors have been uncovered and corrected and a final series of software validation testing began.

In validation testing we test the system functions in a manner that can be reasonably expected by customer, the system was tested against system requirement specification. Different unusual inputs that the users may use were assumed and the outputs were verified for such unprecedented inputs. Deviation or errors discovered at this step are corrected prior to the completion of this project with the help of user by negotiating to establish a method for resolving deficiencies. Thus, the proposed system under consideration has been tested by using validation testing and found to be working satisfactorily. Validation checking is performed on the: -

**Numeric Field:** - The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error message. The individual modules are checked for

accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested module are integrated into a single system.

**Character Field:** - This field can only contain letters from A-Z and a-z. It is useful for name, address fields and so on.

**Check Null Fields:** - Before entering values into the database or when updating, a validation is done to check whether any NULL fields are present.

**Contact number Fields:** - A number only field with a limit of 10 digits.

**Email fields:** - It accepts only valid email as input

All the necessary validation checks were verified to see if invalid data ever enters into the database. Null values in fields were also treated as invalid values.

### 6.3.3 OUTPUT TESTING

After performing validation test, the next phase is output test of the system, since no system could be useful if it does not produce the desired output in the desired format. By consideration the format of the report/output was generated or displayed and was tested. Here output format was considered in two ways: one is on the screen and other as images that are downloaded and kept.

Grid View control was mainly used to retrieve data on the screen. The patient details, booking details, doctor details etc. could be retrieved using this control. The conflict resolution and primary key duplicate values errors were checked. These errors were handled by the error control mechanisms.

### 6.3.4 USER ACCEPTANCE TESTING

User acceptance test of a system is the key factor for the success of the system. The system under consideration was listed for user acceptance by keeping constant touch with the perspective user of the system at the time of design, development and making changes whenever required. This was done with the regards of the following points: -

- Input screen design
- Output design

Users from each of the 3 user types (Admin, Doctor and Patient) are present in the system. The admin, doctor and patient are part of the core customer base so they must be part in the user acceptance testing. Experts in the field who are part of the computer

department also checked and verified if all functionalities given in the SRS were implemented perfectly and if all validations and error handling are done properly.

## **6.4 IMPLEMENTATION**

Implementation is the stage of the project when the theoretical design is turned into a working system. The implementation stage is a system 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 the changeover method.

The first task in implementation is planning deciding on the methods and time scale to be adopted. Once the planning has been completed the major effort is to ensure that the programs in the system are working properly when the staff has been trained. The complete system involving both computer and user can be executed effectively. Thus the clear plans are prepared for the activities.

A training session will be conducted for the admin . A parallel conversion implementation plan is adopted because the dataset in question is not a very large one. Therefore, the labour cost and maintenance cost is negligible to keep both the systems active is negligible. The new system will work parallel with the existing system.

# **ENHANCEMENT & MAINTENANCE**

## **7. MAINTENANCE AND ENHANCEMENT**

### **7.1 MAINTENANCE**

This software can be modified as need occurs. Maintenance includes all the activities after installation of the software that is performed to keep the system operational. The process of maintenance involves:

- Understanding the existing software
- Understand the effect of change
- Test for satisfaction

This software requires little to no maintenance. During the testing phase most maintenance duties are performed. If a maintenance requirement occurs, it can be solved with ease.

### **7.2 ENHANCEMENTS**

This software is divided into modules. Therefore, adding extra functionalities is simple. Enhancements can be added and newer versions can be created with ease as requirements increase and technologies develop.

The modular nature of this software enables seamless integration with future modules. Any function of a Doctor Appointment System can be automated using this system. It can be developed as an independent module and then added to this system with relatively little work.

The software developed is very flexible and much functionality can be added to it, to enhance its performance. These options can be included to improve the efficiency of the software. In future new functionalities can be added to improve the performance. This system can include correct locations of the clinic or hospital. This will help patients to arrive at the clinic or hospital without any difficulties.

# CONCLUSION

## **8. CONCLUSION**

In this project, first an attempt has been made to find the need for the system. To fulfill the needs, a detailed study has been designed in such a way that the system is user friendly and easy to use. This particular system has been designed in an attractive manner, so that even a user with minimum knowledge can be able to operate the system easily.

The system is developed with scalability in mind. All modules in this have been tested separately and put together to form the system. Finally the system is tested with real data and everything work successfully. Thus the system has fulfilled all the objectives identified and is able to replace the existing manual system.

The advantage of this system is that, the package can be easily being incorporated with any other package. In future the package can be developed further to act as virtual manager caring out all the operation.

# **BIBLIOGRAPHY**



## 9. BIBLIOGRAPHY

1. Ajson. "Full Calendar – A Complete Web Diary System for JQuery and C# MVC." *CodeProject*, CodeProject, 14 Sept. 2016, [www.codeproject.com/Articles/638674/Full-calendar-A-complete-web-diary-system-for-jQue](http://www.codeproject.com/Articles/638674/Full-calendar-A-complete-web-diary-system-for-jQue).
2. McManus, Jeffrey P., and Chris Kinsman. *C# Developer's Guide to ASP. NET, XML and ADO. NET*. Addison-Wesley Longman Publishing Co., Inc., 2001.
3. Arora, Dipti. "Learning Management System (LMS) Using C#, ASP. Net and SQL SERVER." (2018).
4. "Using SQL Server with .NET | .NET | Google Cloud." *Google*, Google, [cloud.google.com/dotnet/docs/getting-started/using-sql-server](https://cloud.google.com/dotnet/docs/getting-started/using-sql-server).

### WEB REFERENCES

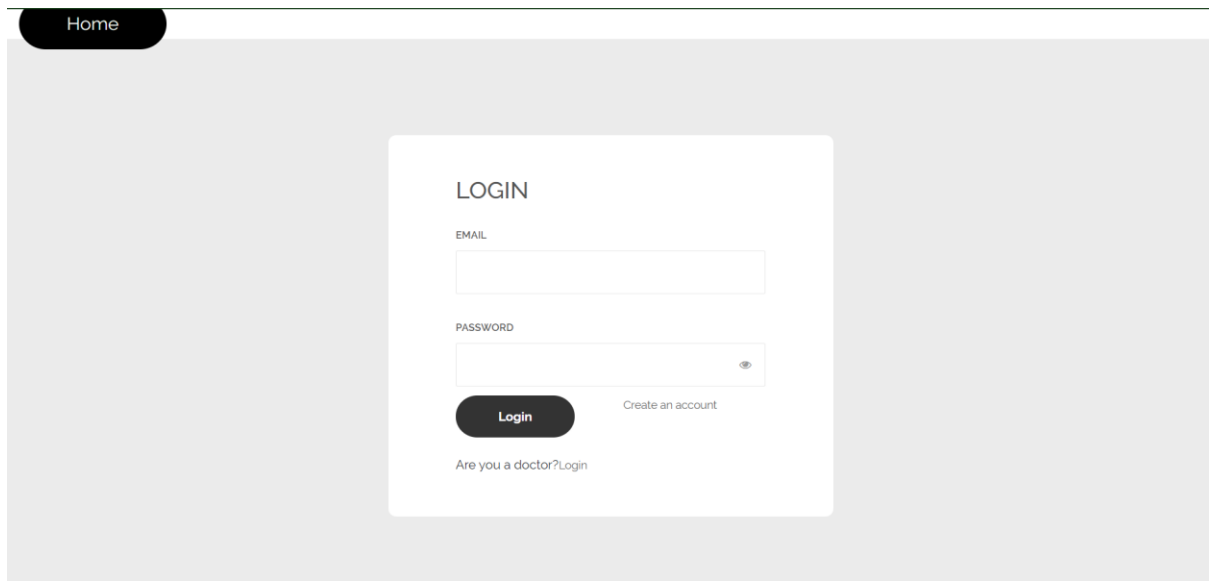
- [www.w3schools.com/asp/webpages\\_intro.asp](http://www.w3schools.com/asp/webpages_intro.asp)
- [www.tutorialspoint.com](http://www.tutorialspoint.com)
- [www.javatpoint.com](http://www.javatpoint.com)

# **APPENDIX**

# APPENDIX

## Screenshots

### LOG IN PAGE



The screenshot shows a login page with a dark blue header containing a 'Home' button. The main content area is light gray and features a white login form. The form has a title 'LOGIN', an 'EMAIL' input field, a 'PASSWORD' input field with a toggle icon, a dark blue 'Login' button, and a 'Create an account' link. Below the form is a link that says 'Are you a doctor? Login'.

Home

LOGIN

EMAIL

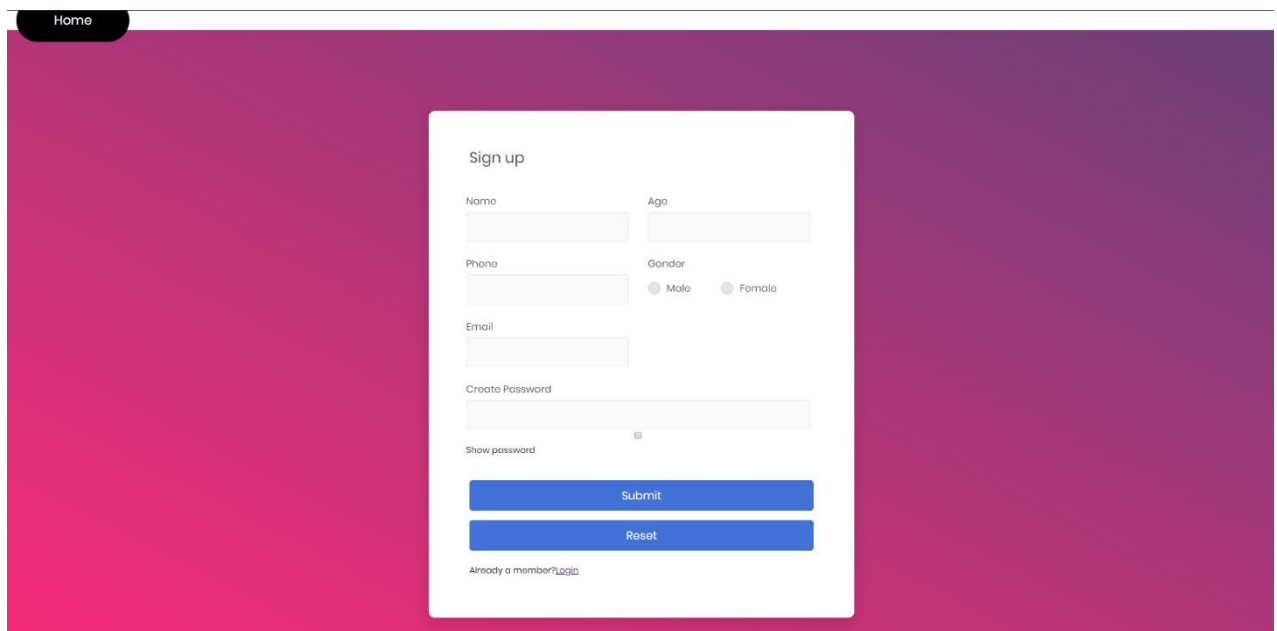
PASSWORD

Login

Create an account

Are you a doctor? Login

### SIGN UP PAGE



The screenshot shows a sign-up page with a dark blue header containing a 'Home' button. The main content area has a purple-to-pink gradient background and features a white sign-up form. The form has a title 'Sign up', input fields for 'Name', 'Age', 'Phone', 'Email', and 'Create Password', a 'Gender' section with 'Male' and 'Female' radio buttons, a 'Show password' toggle, a blue 'Submit' button, a blue 'Reset' button, and a link that says 'Already a member? Login'.

Home

Sign up

Name

Age

Phone

Gender

Male

Female

Email

Create Password

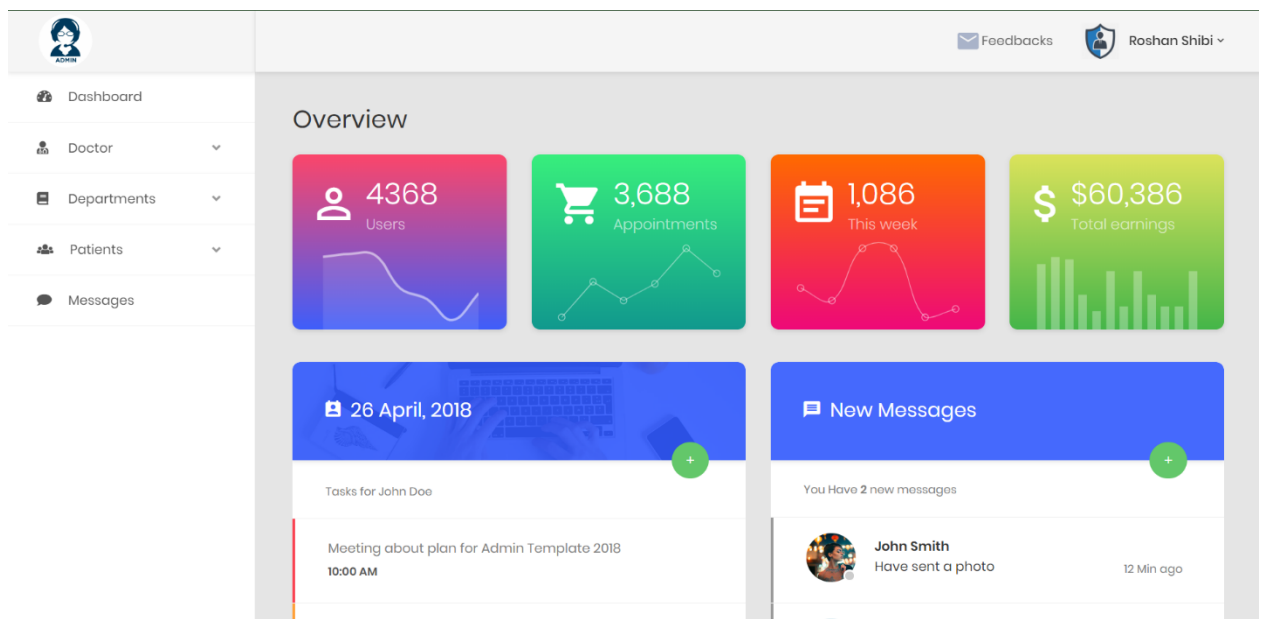
Show password

Submit

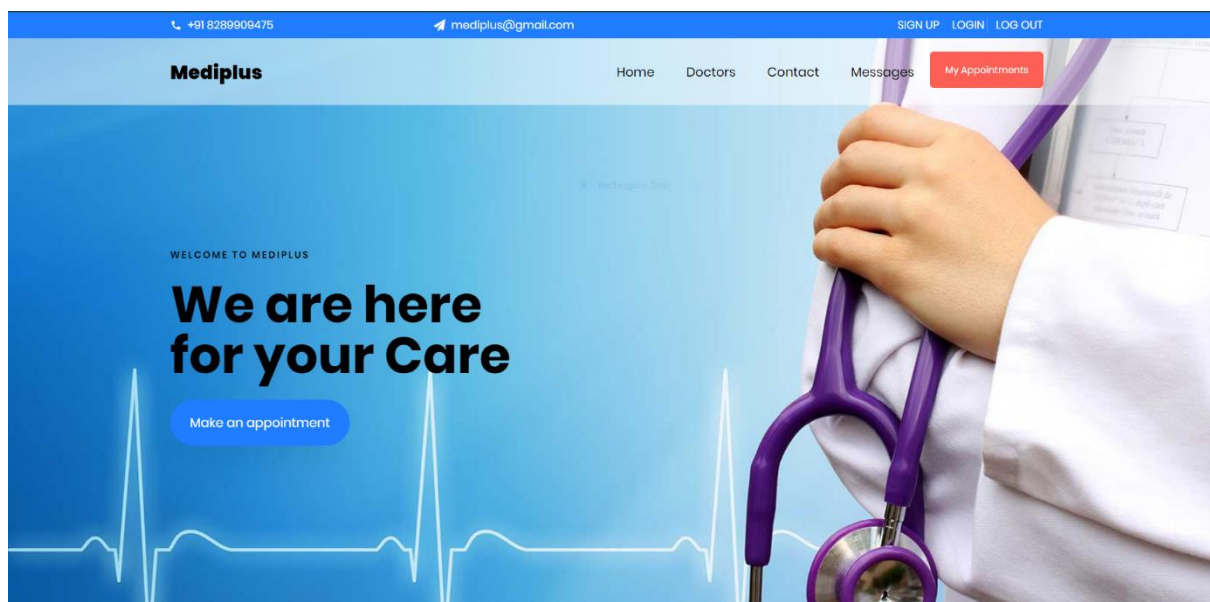
Reset

Already a member? Login

## ADMIN PAGE



## HOME PAGE



## DOCTOR PANEL

