Contents

[Chapter No.1 5](#_Toc428591383)

[Introduction 6](#_Toc428591384)

[1.1Product 6](#_Toc428591385)

[1.2. Purpose: 6](#_Toc428591386)

[1.3 Scope: 6](#_Toc428591387)

[1.4. System’s Goals: 7](#_Toc428591388)

[1.5.General Objective: 7](#_Toc428591389)

[1.6 Specific Objective: 7](#_Toc428591390)

[1.7.Risk Management: 8](#_Toc428591391)

[1.8. Software Features*:* The following are the Online Appointment System features: 8](#_Toc428591392)

[1.9 Planning assumptions: 8](#_Toc428591393)

[1.10 Project document outline: 8](#_Toc428591394)

[Chapter No.02 10](#_Toc428591395)

[2.1 Areas of study: 11](#_Toc428591396)

[2.2 ABOUT THE TOPIC*:* 11](#_Toc428591397)

[Chapter No.03 13](#_Toc428591398)

[3.1 Introductions 14](#_Toc428591399)

[3.2 Existing System 14](#_Toc428591400)

[*3.*3 Description of the Proposed System*:* 15](#_Toc428591401)

[*3*.4 Data Flow Diagram of the proposed system: 15](#_Toc428591402)

[3.5 HTML & CSS as front-end 16](#_Toc428591403)

[3.6 PHP as Middle-End 17](#_Toc428591404)

[3.7 MySQL 18](#_Toc428591405)

[*3*.8 Methodology: 18](#_Toc428591406)

[3.9 Problem definition*:* 19](#_Toc428591407)

[3.8.1Challenges: 20](#_Toc428591408)

[3.8.2 Goals: 20](#_Toc428591409)

[3.8.3 Opportunities: 20](#_Toc428591410)

[3.9 Methodology Models*:* 21](#_Toc428591411)

[3.9.1 Waterfall Methodology*:* 21](#_Toc428591412)

[3.9.2Prototype Model: 24](#_Toc428591413)

[Fig 3.5.2 25](#_Toc428591414)

[3.9.3 Incremental: 26](#_Toc428591415)

[3.9.4 Rapid Application Development Method(RAD): 26](#_Toc428591416)

[3.9.5Iterative model diagram: 28](#_Toc428591417)

[Chapter No.04 30](#_Toc428591418)

[*4.*1 System Design: 31](#_Toc428591419)

[4.2 Database design: 32](#_Toc428591420)

[4.3. Table Design E Doctor Database has 7 Tables to Store Data and Information about the online Doctor Appointment: 32](#_Toc428591421)

[4.3.1 Table User Table is used to Stores the details of users who login and view to our website. 33](#_Toc428591422)

[4.3.2 Academic record Table is used to Stores the details of Doctor about his/her specialization and profile and Viewed by User. 34](#_Toc428591423)

[4.3.3 Plan Table is used to Stores the information of doctors and company’s about their account login authentication through payment method. 34](#_Toc428591424)

[4.3.4 History record table is used to Stores the details of user’s profile and their appointment history through notification. 35](#_Toc428591425)

[4.3.5 Schedule Table is used to Stores the detail information about doctor ‘s clinic timing and appoint the days for appointments. 35](#_Toc428591426)

[4.3.6 Appointment Table is used to Stores the detail information of Appointments 36](#_Toc428591427)

[4.3.7 Login Table Stores the username and Password for login in Admin Panel. 36](#_Toc428591428)

[4.4 Database Design 37](#_Toc428591429)

[4.4.1Tables Relationship Diagram: 38](#_Toc428591430)

[4.5 Entity Relationship Diagram*:* 39](#_Toc428591431)

[*4*.6 System Architecture: 40](#_Toc428591432)

[4.6.1patient: 40](#_Toc428591433)

[4.6.2 doctor: 40](#_Toc428591434)

[4.6.3 Company 41](#_Toc428591435)

[4.7 System sequence diagram 42](#_Toc428591436)

[4.7.1 Scenarios.1 login : 42](#_Toc428591437)

[4.7.2 Scenarios history: 43](#_Toc428591438)

[4.7.3 Scenarios. appointment 43](#_Toc428591439)

[4.7.4 Scenarios.4 Edit Account: 44](#_Toc428591440)

[4.7.5 Scenarios. plan : 45](#_Toc428591441)

[4.7.6 Scenarios. specialty 45](#_Toc428591442)

[4.7.7 Scenarios. logout 46](#_Toc428591443)

[Chapter No.05 47](#_Toc428591444)

[5.1 Functional requirements: 48](#_Toc428591445)

[5.2 Nonfunctional Requirements: 49](#_Toc428591446)

[ Performance Requirements: 49](#_Toc428591447)

[ Safety Requirements: 49](#_Toc428591448)

[5.3. Software Quality Attributes 50](#_Toc428591449)

[5.4 Hardware Requirement: 50](#_Toc428591450)

[5.5 Software Requirements: 51](#_Toc428591451)

[*5.*6 Feasibility Studies: 51](#_Toc428591452)

[5.7Use Case diagram: 52](#_Toc428591453)

[5.7.1 Use case admin 53](#_Toc428591454)

[5.8 Use-Case: Administration: 54](#_Toc428591455)

[5.9 Use case Registration 55](#_Toc428591456)

[5.10 Use case Create profile: 56](#_Toc428591457)

[5.11. Use case Patient: 57](#_Toc428591458)

[5.12 Interface Diagram: 58](#_Toc428591459)

[5.12.1 Main header 58](#_Toc428591460)

[5.12.2From index page. 58](#_Toc428591461)

[5.12.3 If user /patient visit the E doctor then take appointment with doctor. 59](#_Toc428591462)

[5.12.6 If you are a doctor then register from this form: 62](#_Toc428591463)

[5.12.7 About doctors Education 64](#_Toc428591464)

[5.12.8 65](#_Toc428591465)

[About Doctors clinic 65](#_Toc428591466)

[5.12.8 Doctor’s Account Plan 66](#_Toc428591467)

[5.12.9 After choosing the plan let’s come to start practice with E doctor. And go to your profile page. 67](#_Toc428591468)

[5.12.10 If you’re a company then join now 68](#_Toc428591469)

[5.12.11Then register from this form. 68](#_Toc428591470)

[5.12.12 After that company choose admin plan. 69](#_Toc428591471)

[5.12.13 Then go to his own profile page. And take appointments with doctor . 70](#_Toc428591472)

[Chapter No.06 71](#_Toc428591473)

[6.1 Implementation*:* 72](#_Toc428591474)

[6.2 Parallel Implementation: 72](#_Toc428591475)

[6.3-Testing Techniques: 72](#_Toc428591476)

[6.3.1 Unit testing: 73](#_Toc428591477)

[6.3.2 Integration testing: 73](#_Toc428591478)

[6.3.3 Black Box testing: 73](#_Toc428591479)

[6.3.4 White Box testing*:* 74](#_Toc428591480)

[6.4 Test cases: 74](#_Toc428591481)

[6.4.1Admin Module Test Cases: 75](#_Toc428591482)

[6.4.2Test case Manage Doctor: 76](#_Toc428591483)

[6.4.3Test Case Manage schedule: 77](#_Toc428591484)

[6.4.4Test Case Company Module Test Case: 78](#_Toc428591485)

[6.5 Test case 79](#_Toc428591486)

[6.5.1 Test Case 1 : Registration 79](#_Toc428591487)

[6.5.2 Test Case 2 : Login 79](#_Toc428591488)

[6.5.3Test Case 3 : User profile 80](#_Toc428591489)

[6.5.4 Test Case 4 :Changing Password 80](#_Toc428591490)

[Chapter No.07 81](#_Toc428591491)

[7.1 Conclusion: 82](#_Toc428591492)

[7.2 Possible Extension 82](#_Toc428591493)

[7.2.1 Advantages 83](#_Toc428591494)

[7.2.3 Disadvantages 83](#_Toc428591495)

[References 84](#_Toc428591496)

# 

# Chapter No.1

*Introduction*

# Introduction

# 1.1 Product

**Online Doctor’s Appointment** , the Online System to create a user friendly interface and to bridge a gap between the patient and the doctor. This system caters to all the latest technologies, it has been developed to educate our people as well as to save their lives.

# 1.2. Purpose:

An Online Doctor Appointment offers Health Professionals a more efficient and convenient way for patients to reserve appointments in the clinic. It’s the future in appointment scheduling in clinic. Patients can book an appointment that is convenient for them, which clinic’s staffs are able to control and maintain patients’ information through a computerized process.

Online Doctor Appointment has web application which requires patients to register with their username and password before they can access the features. Once a patient has logged in to the account, he can already view the doctors’ profile along with their schedules and set up an appointment. The patient can request for an appointment at any time, showing real time appointments.

# 1.3 Scope:

Now a days people are seeking for a convenient way of being reserved on their appointments. So do in Online Doctor appointment (reservation) process. In this situation therefore, Online Doctor Appointment System is a good project to ease on difficult and too time consuming manual appointment reservation processes. Online Doctor Appointment is a centralized web portal for health appointments where patients find and book an appointment with their health practitioner online. The system is synchronized with reservation system and saves front office staff countless interruptions during their already busy day and helps provide a more customer friendly experience to patients.

The system cover all the basic modules include Doctors profile module, clinic information maintenance module, complete Appointments and notification module, patient profile maintenance module, patient appointment module. Company module includes company medicines, advertisement of their product and Appointments with doctors. The Online Doctor Appointment System has the following scope:

* This website is openly for patients, who can join us and takes Appointments with doctors.
* The online scheduler which contains all the doctors’ available time of appointment.
* Information update, review, and maintenance.
* For patients, only information related to their reservation will be displayed through their account website.
* For clinic staffs, the information of both doctors and patients can be reviewed on system’s main server.
* The system increases new patient traffic, saves staff time and helps reduce your cancellations as you save time and money with lower phone bills and greater efficiency.

# 1.4. System’s Goals:

Our main goal is to serve the humanity and to save their precious and invaluable lives. This

web based application communicates with the doctor and patient. And creates a healthy interaction and reliability between the doctor and the patient.

# 1.5.General Objective:

The main objective of this study is to develop an Online appointment reservation system.

# 1.6 Specific Objective:

* To provide a convenient way of appointment reservation for patients.
* To automated notification generation.
* To computerized the patients’ information review and maintenance.
* To give clinic staffs an easy way in doing information maintenance and updates

# 1.7.Risk Management:

Upon the completion of the project, the whole team will analyses risks. Based on this analysis, the project supervisor and project manager will identify any improvements that can be made to manage those risks and develop risk management process to minimize its impact on project.

* Planning assumptions: At every stage during planning assumptions are made which, if not valid, may put the plan at risk.
* Unexpected events.
* Required hardware not available within the time or not available in the market.
* Faults in required hardware.
* Required Internet not available within the time.
* Hacking Account.

# 1.8. Software Features*:* The following are the Online Appointment System features:

* Online website and online scheduler for Appointment (Booking).
* Account username and password for company, doctors, and patients.
* Database for information storage (staffs’, doctors’, and patients’ information).
* Security features for changing of account username and password.
* Notification Generated about Appointments.
* User-interface and web application.

# 1.9 Planning assumptions:

At every stage during planning assumptions are made which, if not valid, may put the plan at risk.

* Unexpected events.
* Required Internet not available within the time.
* Forget User name and password

# 1.10 Project document outline:

* Documentation will contain the following:
* Introduction
* Background
* Problem definition and goals
* System requirements
* Architectural design implementation and testing
* Conclusion and possible extension

Appendices

# 

# Chapter No.02

*Background*

# 

# 2.1 Areas of study:

Different areas of studies of computer science are involved in the development of this project like software engineering, Advance programming.

We use this tool:

* HTML
* CSS
* Java Script
* J Query
* Bootstrap
* PHP
* My SQL

# 2.2 ABOUT THE TOPIC*:*

A **website** is a collection of Web pages, images, videos and other digital assets that is hosted on one or several Web servers, usually accessible via the Internet, cell phone or a LAN.

The pages of websites can usually be accessed from a common root URL called the homepage, and usually reside on the same physical server. The URLs of the pages organize them into a hierarchy, although the hyperlinks between them control how the reader perceives the overall structure and how the traffic flows between the different parts of the sites.

A webpage is a document, typically written in plain text interspersed with formatting instructions of Hypertext Markup Language (HTML,XHTML). A webpage may incorporate elements from other websites with suitable markup anchors. Webpages are accessed and transported with the Hypertext Transfer Protocol (HTTP), which may optionally employ encryption (HTTP Secure, HTTPS) to provide security and privacy for the user of the webpage content. The user's application, often a web browser, renders the page content according to its HTML markup instructions onto a display terminal.

A website requires attractive design and proper arrangement of links and images, which enables a browser to easily interpret and access the properties of the site. Hence it provides the browser with adequate information and functionality about the organization, community, network etc.

A dynamic website is one that changes or customizes itself frequently and automatically.

Server-side dynamic pages are generated "on the fly" by computer code that produces

The HTML and CSS there are a wide range of software system such as CGI, JAVA SERVLTES and JAVA server pages (JSP), Active server pages and code cold fusion (CFML) that are available to generate dynamic web systems and dynamic sits various web application frameworks and web templates systems are available for general-use –programming languages like PHP, Perl, PYTHON and Ruby, to make it faster and easier to create complex dynamic websites.

A site can display the current state of a dialogue between users, monitor a changing situation, or provide information in some way personalized to the requirements of the individual user. Dynamic HTML uses JavaScript code to instruct the web browser how to interactively modify the page contents.

Early web sites had only text, and soon after, images. Web browser plug ins were then used to add audio, video, and interactivity (such as for a rich Internet application that mirrors the complexity of a desktop application like a word processor). Examples of such plug-ins are Microsoft Silver light, Adobe Flash, Adobe Shockwave, and applets written in Java. HTML 5 includes provisions for audio and video without plugging. JavaScript is also built into most modern web browsers, and allows for web site creators to send code to the web browser that instructs it how to interactively modify page content and communicate with the web server if needed.

# Chapter No.03

*System Analysis*

# 3.1 Introductions

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements on the system. System analysis is a problem solving activity that requires intensive communication between the system users and system developers.

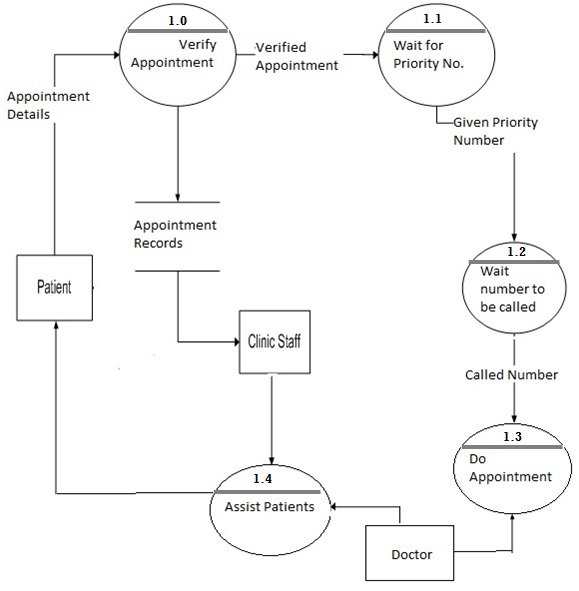
System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. The system analyst plays the role of an interrogator and deep into the working of the present system. The system is viewed as a whole and the inputs to the system are identified. The outputs from the organization are traced through the various processing that the inputs phase through the organization.

A detailed study of these processes must be made by various techniques like Interviews, Questionnaires etc. The data collected by these sources must be examined to arrive to a conclusion. The conclusion is about the understanding of how the system works. This system is called the existing system. Now, the existing system is subjected to close study 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. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable changes are made. This loop ends as soon as the user is satisfied with the proposal.

# 3.2 Existing System

The clinic uses a traditional manual file processing, which is too time consuming. Appointments are being set based on the doctor’s available time prior to patients’ visit on clinic. Therefore, patients are not assured if their preferred schedule is still available. The clinic staff will handle patients’ on their visit and they will be given a stab which has the priority number of their appointment.

The patient will wait on his priority number to be called. If he is unattended (e.g. he went outside because he assumed that other appointments will last long and therefore he will wait too much), his number will be passed by and he needs to wait another available time for him to be entertained.



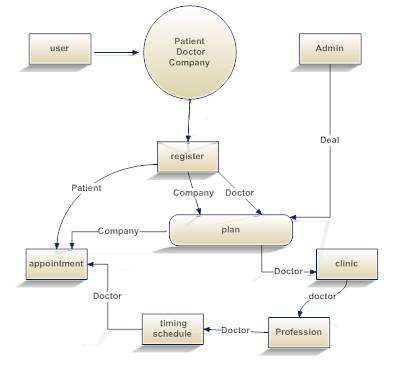
# *3.*3 Description of the Proposed System*:*

The proposed system which is Online Doctor Appointment System will provide ease on patients’ information maintenance. It provides a web application which has scheduler to automate the appointment reservation process.

If the reserved patient was unable to confirm his appointment within the time based on the clinic’s policy, the Doctor will be the one responsible for the adjustment of time schedule. He could serve walk-in patients prior to the number of canceled appointments.

# *3*.4 Data Flow Diagram of the proposed system:

A Data Flow Diagram is a graphical representation of the "flow" of data through an [information system](https://en.wikipedia.org/wiki/Information_system), modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of [data processing](https://en.wikipedia.org/wiki/Data_processing)

****

We developed our project by using the **three tier architecture** which uses the following languages.

* 1. HTML & CSS as front-end
  2. PHP as middle-end
  3. MYSQL as back-end

# 3.5 HTML & CSS as front-end

HTML or HyperText Markup Language is the standard markup language used to create web pages.

HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>). HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent empty elements and so are unpaired, for example <img>. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags).

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a markup language rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages.

Web browsers can also refer to Cascading Style Sheets (CSS) to define the look and layout of text and other material. The W3C, maintainer of both the HTML and the CSS standards, encourages the use of CSS over explicit presentational HTML.

# 3.6 PHP as Middle-End

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 (the Zend Engine) is now produced by The PHP Group. While PHP originally stood for Personal Home Page, it now stands for PHP: Hypertext Preprocessor, which is a recursive acronym.

PHP code can be simply mixed with HTML code, or it can be used in combination with various template 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 command-line interface (CLI) capability and can be used in Standalone Graphical application.

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.

# 3.7 MySQL

MySQL "My S-Q-L" officially, but also called "My Sequel" is (as of March 2014) the World’s second most widely used open-source relational database management system (RDBMS). It is named after co-founder Michael Widenius's daughter My. The SQL phrase stands for Structured Query Language. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL,Perl /PHP/Python." Free-software-open source projects that require a full-featured data base management system often use MySQL

# *3*.8 Methodology:

This project is implementing by using three-tier architecture which easy in future maintenance and to protect the E Doctor data in a secure way. The tool that uses to develop this system is PHP and for the database storing is MYSQL.

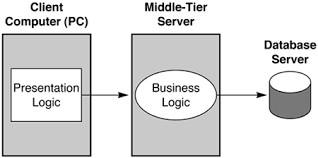
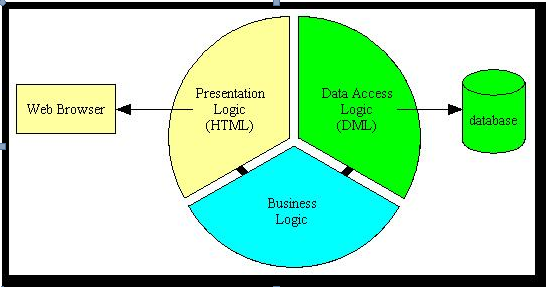


Figure 1



# 3.9 Problem definition*:*

We developed this product for online dairy farm. Only Online registered user perform different functionality

**There are three types of problems:**

1-Challenges:

2-Goals:

3-Opportunities:

## 3.8.1Challenges:

This basic term of the project to see what type of the problem is occurs in the future and what the challenge is face in admin side. In problem definition the biggest Challenges is competitor because competitor is that person of your who make the website and give many assertive to the user in low price. So competitor is always thought about user benefits.

So we always thought about the competitor mind we add many of the menus in our website because user never go any other web site he is only use our website and use facility. The other called is user interest. Because if we are given the user benefits the user never visits our website. If the website design is easily to use and also user friendly the user visit many time.

## 3.8.2 Goals:

The next step is finding the basic goal of our project. Is the goal of our project is find and the next step is give the benefits the peoples. The owner of the website is totally observed the people mind and sees what the people interest is. So the owner of the website have a specific goals to use in the future time because the if the person work to see the goals the interest is increase to do new.

## 3.8.3 Opportunities:

Opportunities are also challenge of the project. Because in opportunities the owner is thought what kind of the website make and the benefits in use future. If our project has different and easily to use the opportunities is increase.

Our website Online Doctor Appointment so there different kind of website and large number of the people want to use this website because many of the kind of websites make but the Functionalities is so less and the people not like these website beside this our website have large number of Functionalities and people like this. So our online doctor appointment website has greater kind of functionalities user easily work in this website.

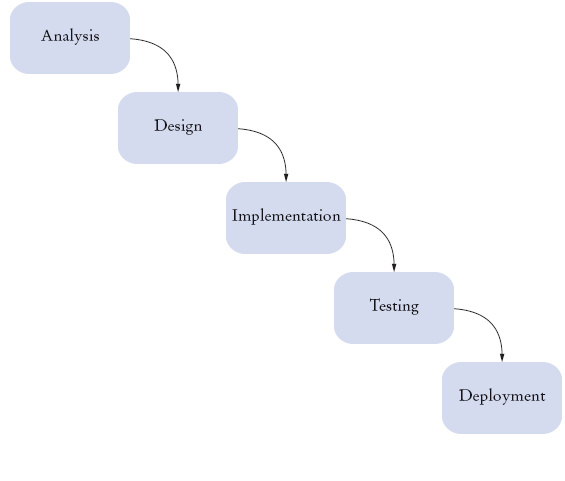
# 3.9 Methodology Models*:*

There are some popular methodologies used. Some of them are following:

* Water Fall Model
* Prototype Model
* Iterative Model
* Spiral Model
* Agile Model
* RAD Model

## 3.9.1 Waterfall Methodology*:*

The Waterfall Model was first Process Model to be introduced. It is also referred to as a linear-sequential lifecycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can be in and there is no overlapping in the phases.

Waterfall model is the earliest SDLC approach that was used for software development

* A waterfall model is easy to follow.
* It can be implemented for any size project.
* Every stage has to be done separately at the right time so you cannot jump stages
* Testing is done at every stage.

A waterfall model helps find problems earlier on. Requirements will be set and these wouldn't be changed. If requirements may change the Waterfall model may not work. Difficult to estimate time and cost for each stage of the development process.

**Analysis:**

Requirements analysis in systems engineering and software engineering, encompasses those tasks that go into determining the needs or conditions to meet for a new or altered product, taking account of the possibly conflicting requirements of the various stakeholders, analysing, documenting, validating and managing software or system requirements.

**Design:**

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

solution domain. The design of a system is perhaps the most critical factor affecting the quality of the software, and has a major impact on the later phases, particularly testing and maintenance. The output of this phase is the design document. This document is similar to a blue print or plan for the solution, and is used later during implementation, testing and maintenance.

**Coding:**

Once the design is complete, most of the major decisions about the system have been made. The goal of the coding phase is to translate the design of the system into code in given programming language. For a given design, the aim of this phase is to implement the design in the best possible manner. The coding phase affects both testing and maintenance profoundly. A well written code reduces the testing and maintenance effort. Since the testing and maintenance cost of software are much higher than the coding cost, the goal of coding should be reduce the testing and maintenance effort. Hence, during coding the focus should be on developing programs that are easy to write. Simplicity and clarity should be strived , during the coding phase.

**Testing:**

Testing is the major quality control measure employed during software development. Its basic function is to detect errors in the software. During requirement analysis and design, the output is a document that is usually textual and non-executable. After the coding phase, computer programs are available that can be executed for testing phases. This implies that testing not only has to uncover errors introduced during coding, but also errors introduced during the previous phases. Thus, the goal of testing is to uncover requirement, design or coding errors in the programs.

**Implementation and Maintenance:**

Maintenance includes all the activity after the installation of software that is performed to keep the system operational. As we have mentioned earlier, software often has design faults. The two major forms of maintenance activities are adaptive maintenance and corrective maintenance. It is generally agreed that for large systems, removing all the faults before delivery is extremely difficult and faults will be discovered long after the system is installed. As these faults are detected, they have to be removed.

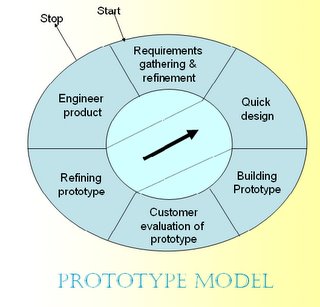
Maintenance activities related to fixing of errors fall under corrective maintenance. Removing errors is one of the activities of maintenance. Maintenance also needed due to a change in the environment or the requirements of the system. The introduction of a software system affects the work environment. This change in environment often changes what is desired from the system. Furthermore, often after the system is installed and the users have had a chance to work with it for some time, requirements that are not identified during requirement analysis phase will be uncovered. This occurs, since the experience with the software helps the user to define the needs more precisely. There might also be changes in the input data, the system environment and output formats. All these require modification of the software. The maintenance activities related to such modification fall under adaptive maintenance.

## 3.9.2Prototype Model:

The Prototyping Model is a systems development method (SDM) in which a prototypeis built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system or product can now be developed.

This model works best in scenarios where not all of the project requirements are known in detail ahead of time. It is an iterative, trial-and-error process that takes place between the developers and the users.

**Prototype Model Diagram:**



# Fig 3.5.2

There are several steps in the Prototyping Model:

The new system requirements are defined in as much detail as possible. This usually involves interviewing a number of users representing all the departments or aspects of the existing system

1. A preliminary design is created for the new system.
2. A first prototype of the new system is constructed from the preliminary design.
3. The users thoroughly evaluate the first prototype, noting its strengths and weaknesses, what needs to be added, and what should to be removed.
4. The first prototype is modified, based on the comments supplied by the users, and a second prototype of the new system is constructed.
5. The first prototype is modified, based on the comments supplied by the users, and a second prototype of the new system is constructed

## 3.9.3 Incremental:

Incremental development is a staging and scheduling strategy in which various parts of the system are developed at different times or rates and integrated as they are completed.



**Advantages of Incremental model:**

1. Generates working software quickly and early during the software life cycle.
2. More flexible
3. Easier to test and debug during a smaller iteration
4. Customer can respond to each built
5. Lowers initial delivery cost

## 3.9.4 Rapid Application Development Method(RAD):

Rapid Application Development (RAD) is an incremental process model that emphasises a short development cycle. Uses high speed adaptation of waterfall model in which rapid development is achieved using a component based construction approach. Suitable when requirements are well understood and project scope is constrained.

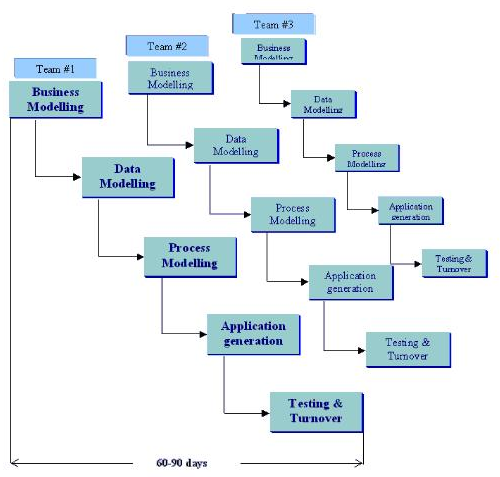
**Maps on to earlier generic framework**.

Communication works to understand the business problem and information characteristics. Planning is essential because multiple software teams work in parallel on different system functions. Modelling has 3 phases: business modelling, data modelling and process modelling. Construction emphasises use of pre-existing software components and the application of automatic code generation.

**Deployment establishes basis for subsequent iterations.**

Time constraints demand scalable scope. If business application can be modularised in such a way that each major business function can be completed in less than 90 days, then RAD is applicable. Each business function can be tackled by a separate RAD team and then integrated to form a whole.

**Diagram of RAD-Model:**

**Figure**

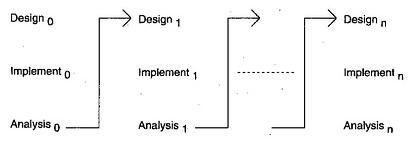
**Adapting methodology:**

We use iterative methodology for our project. It was best for our project

**Reasons for Choosing the Methodology**:

An iterative life cycle model does not attempt to start with a full specification of requirements. Instead, development begins by specifying and implementing just part of the software, which can then be reviewed in order to identify further requirements. This process is then repeated, producing a new version of the software for each cycle of the model.

## 3.9.5Iterative model diagram:



**Fig 4.5We use this methodology because if user wants to change the requirements at any condition then we should able to improve the product as according to new requirements of User.**

* In iterative model we are building and improving the product step by step.
* In iterative model we can get the reliable user feedback
* In iterative model Major requirements must be defined; however, some details can evolve with time.
* **it**  allows developers to break down the task of developing a system into a series of smaller tasks

# 

# Chapter No.04

*Architectural Design*

# 

# *4.*1 System Design:

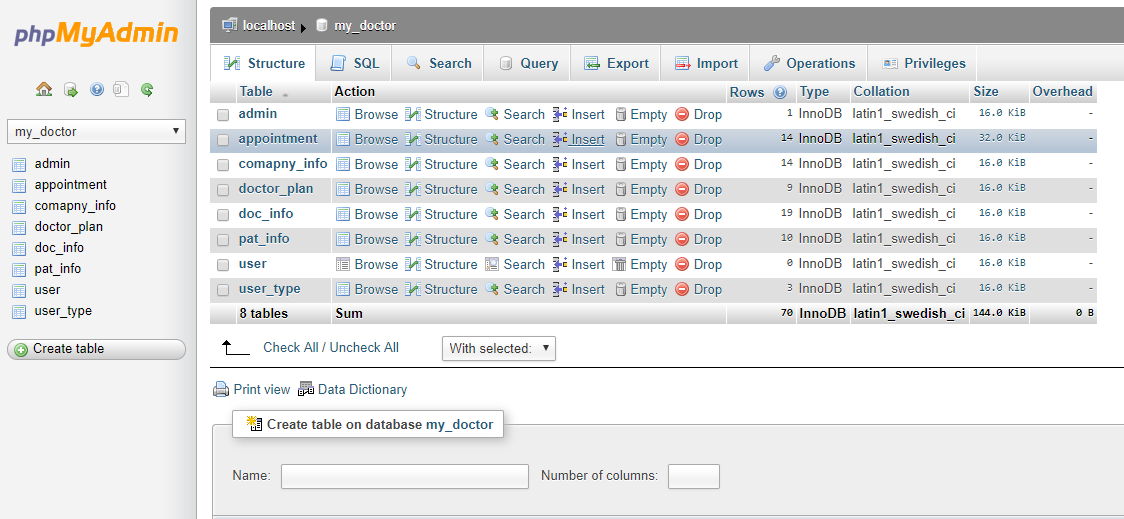
System design is the solution to the creation of a new system. This phase is composed of several systems. This phase focuses on the detailed implementation of the feasible system. It emphasis on translating design specifications to performance specification. System design has two phases of development logical and physical design.

During logical design phase the analyst describes inputs (sources), outputs (destinations), databases (data sources) and procedures (data flows) all in a format that meats the uses requirements. The analyst also specifies the user needs and at a level that virtually determines the information flow into and out of the system and the data resources. Here the logical design is done through data flow diagrams and database

design.

The physical design is followed by physical design or coding. Physical design produces the working system by defining the design specifications, which tell the programmers exactly what the candidate system must do. The programmers write the necessary programs that accept input from the user, perform necessary processing on accepted data through call and produce the required report on a hard copy or display it on the screen.

# 4.2 Database design:

****

**Figure 4.2**

# 

# 4.3. Table Design My Doctor Database has 7 Tables to Store Data and Information about the my\_doctor:

|  |  |
| --- | --- |
| No | Table name |
| 1 | Admin |
| 2 | Appointment |
| 3 | Company\_info |
| 4 | Doc\_info |
| 5 | Pat\_info |
| 6 | User |
| 7 | User\_type |

**Table 4.3**

## 

## 

## 4.3.1 Table User Table is used to Stores the details of users who login and view to our website.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Field name** |  |  | **Type** |  |  | **Constraints** |  |  | **Null** |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | User\_ID |  |  | Int |  |  | Auto Increment |  |  | No |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | First\_name |  |  | Varchar |  |  | None |  |  | No |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Last\_name |  |  | Varchar |  |  | None |  |  | No |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | User\_type |  |  | Varchar |  |  | None |  |  | No |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Password |  |  | Varchar |  |  | None |  |  | No |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Email |  |  | Varchar |  |  | None |  |  | No |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | Address |  |  | Varchar |  |  | None |  |  | No |  |  |
|  | Gender |  |  | Varchar |  |  | None |  |  | No |  |  |
|  | Role |  |  | Varchar |  |  | None |  |  | No |  |  |
|  | Country |  |  | Varchar |  |  | None |  |  | No |  |  |
|  | City |  |  | Varchar |  |  | None |  |  | No |  |  |
|  | Postal \_code |  |  | Varchar |  |  | None |  |  | No |  |  |
|  | Picture |  |  | Varchar |  |  | None |  |  | No |  |  |
|  | Status |  |  | Varchar |  |  | None |  |  | No |  |  |

**Table 4.3.1**

## 4.3.2 Academic record Table is used to Stores the details of Doctor about his/her specialization and profile and Viewed by User.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Field Name** |  |  | **Type** |  |  | **Constraints** |  |  | **Null** |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Acadmic\_ID** |  |  | **Int** |  |  | **Auto increment** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  | **User\_id** |  |  | **Int** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Name** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Degree** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Employee** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  | **Specialization** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |
|  |  |  |  |  | ***Table 4.3.2*** | | |  |  |  |  |  |

## 4.3.3 Plan Table is used to Stores the information of doctors and company’s about their account login authentication through payment method.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Field Name** |  |  | **Type** |  |  | **Constraints** |  |  | **Null** |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Plan\_id** |  |  | **Int** |  |  | **Auto increment** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **User\_id** |  |  | **Int** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Name** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Type** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  | **Payment\_method** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Currency** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  | **Time stamp** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  | **Atus** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |
|  |  |  |  |  | ***Table 4.3.3*** | | |  |  |  |  |  |

## 

## 4.3.4 History record table is used to Stores the details of user’s profile and their appointment history through notification.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Field Name** |  |  | **Type** |  |  | **Constraints** |  |  | **Null** |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **History\_ID** |  |  | **Int** |  |  | **Auto increment** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **User\_id** |  |  | **Int** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Company\_info** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Patient\_info** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Profile** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  | **Notification** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |
|  |  |  |  |  | ***Table 4.3.4*** | | |  |  |  |  |  |
|  |  |  |  |  |  | | |  |  |  |  |  |

## 4.3.5 Schedule Table is used to Stores the detail information about doctor ‘s clinic timing and appoint the days for appointments.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Field Name** |  |  | **Type** |  |  | **Constraints** |  |  | **Null** |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Schudule \_ID** |  |  | **Int** |  |  | **Auto increment** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **User\_id** |  |  | **Int** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Start-time** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **End \_time** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Timing** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |
|  | **Saturday** |  |  | **Varchar** |  | **None** | |  |  | **No** |  |  |
|  | **Sunday** |  |  | **Varchar** |  | **None** | |  |  | **No** |  |  |
|  | **Monday** |  |  | **Varchar** |  | **None** | |  |  | **No** |  |  |
|  | **Thuesday** |  |  | **Varchar** |  | **None** | |  |  | **No** |  |  |
|  | **Wednesday** |  |  | **Varchar** |  | **None** | |  |  | **No** |  |  |
|  | **Thirsday** |  |  | **Varchar** |  | **None** | |  |  | **No** |  |  |
|  | **Friday** |  |  | **Varchar** |  | **None** | |  |  | **No** |  |  |

**Table 4.3.5**

## 4.3.6 Appointment Table is used to Stores the detail information of Appointments

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Field Name** |  |  | **Type** |  |  | **Constraints** |  |  | **Null** |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Appointment\_ID** |  |  | **Int** |  |  | **Auto increment** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **User\_id** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Schedule\_id** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Type** |  |  | **varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |
|  |  |  |  |  | ***Table 4.3.6*** | | |  |  |  |  |  |

## 

## 4.3.7 Login Table Stores the username and Password for login in Admin Panel.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Field Name** |  |  | **Type** |  |  | **Constraints** |  |  | **Null** |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Admin\_ID** |  |  | **Int** |  |  | **Auto increment** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Encrypted\_username** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **Encrypted \_password** |  |  | **Varchar** |  |  | **None** |  |  | **No** |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  | |  |  |  |  |  |
|  |  |  |  |  | ***Table 4.3.7*** | | |  |  |  |  |  |

# 4.4 Database Design

The overall objective in the development of database technology has been to treat data as an organizational resource and as an integrated whole. DBMS allow data to be protected and organized separately from other resources. Database is an integrated collection of data. The most significant form of data as seen by the programmers is data as stored on the direct access storage devices. This is the difference between logical and physical data.

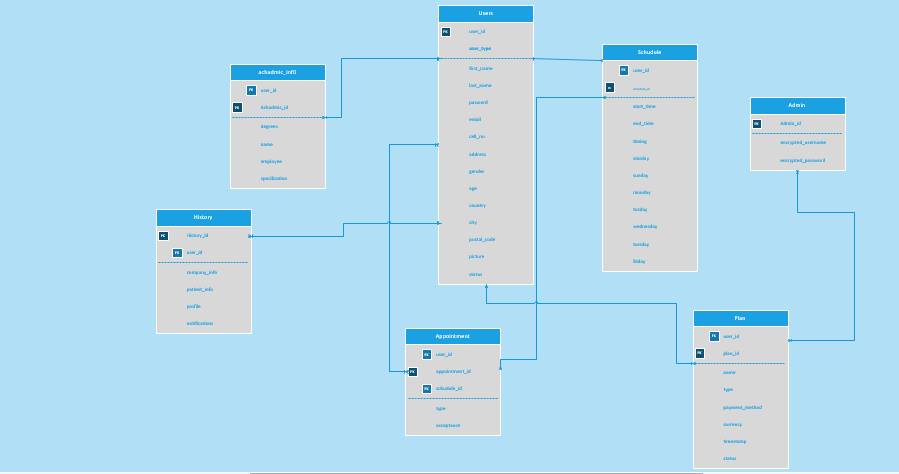
Database files are the key source of information into the system. It is the process of designing database files, which are the key source of information to the system. The files should be properly designed and planned for collection, accumulation, editing and retrieving the required information.

The organization of data in database aims to achieve three major objectives: -

* Data integration.
* Data integrity.
* Data independence.

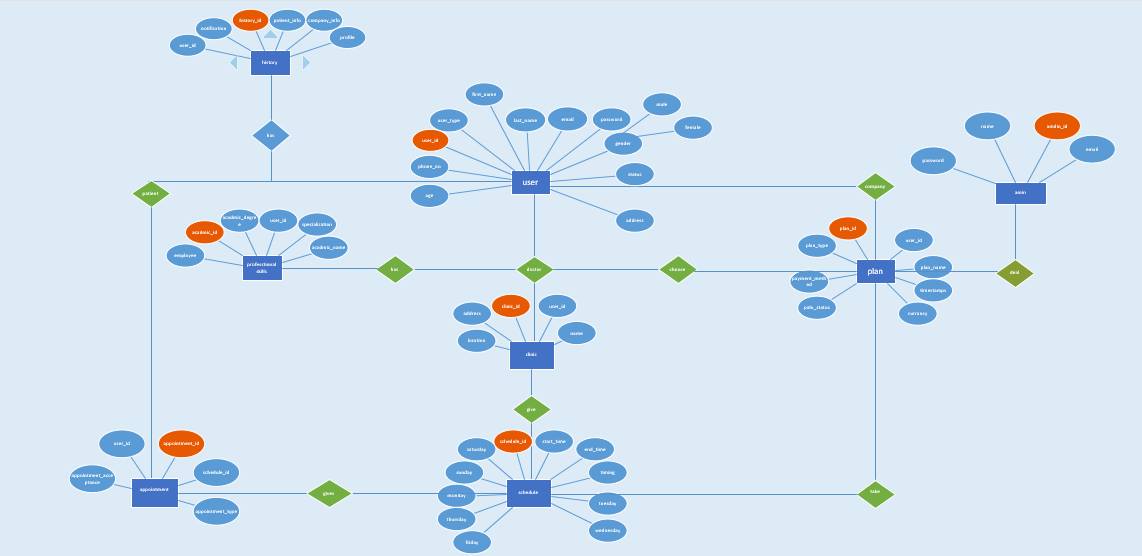
The proposed system stores the information relevant for processing in the MS SQL SERVER database. This database contains tables, where each table corresponds to one particular type of information. Each piece of information in table is called a field or column. A table also contains records, which is a set of fields. All records in a table have the different set of fields with different information. There are primary key fields that uniquely identify a record in a table. There are also fields that contain primary key from another table called foreign keys.

## 4.4.1Tables Relationship Diagram:



## 

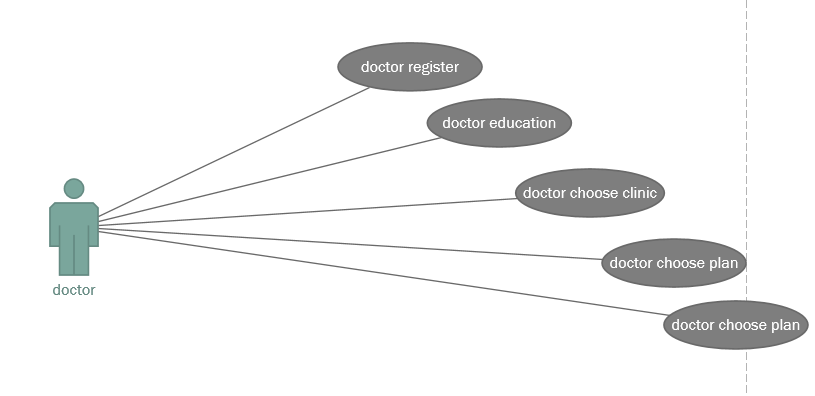
## 4.5 Entity Relationship Diagram*:*

****

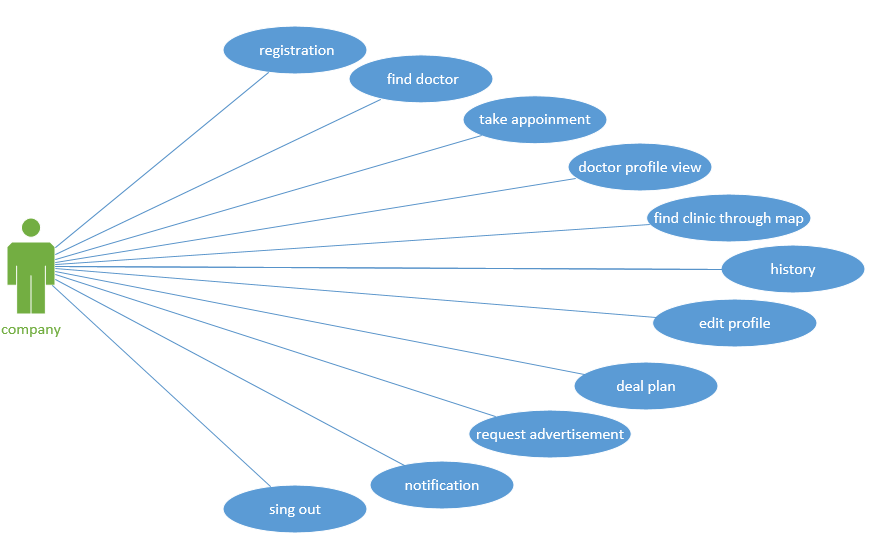
# *4*.6 System Architecture:

## 4.6.1patient:

## C:\Users\UmairIlyas\Desktop\Capture.PNG4.6.2 doctor:

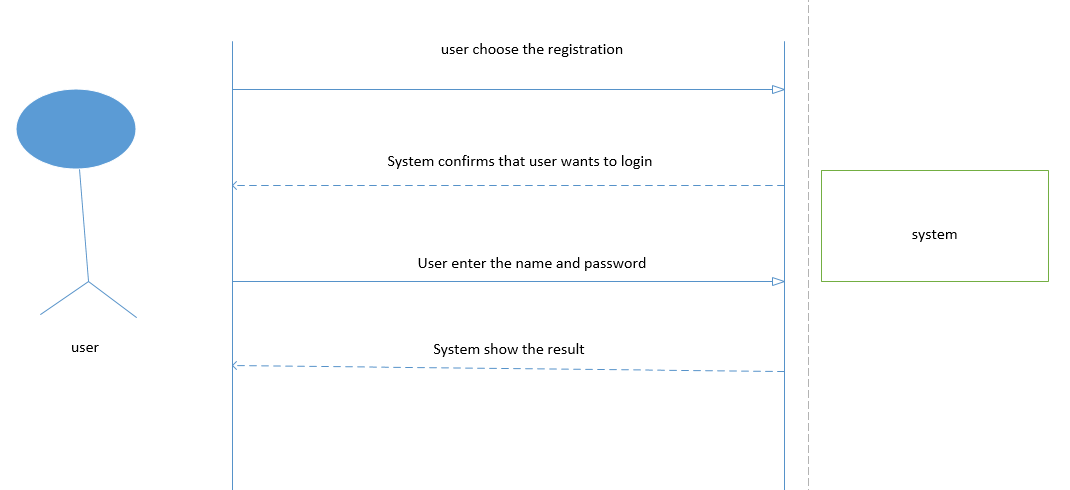
****

## 4.6.3 Company

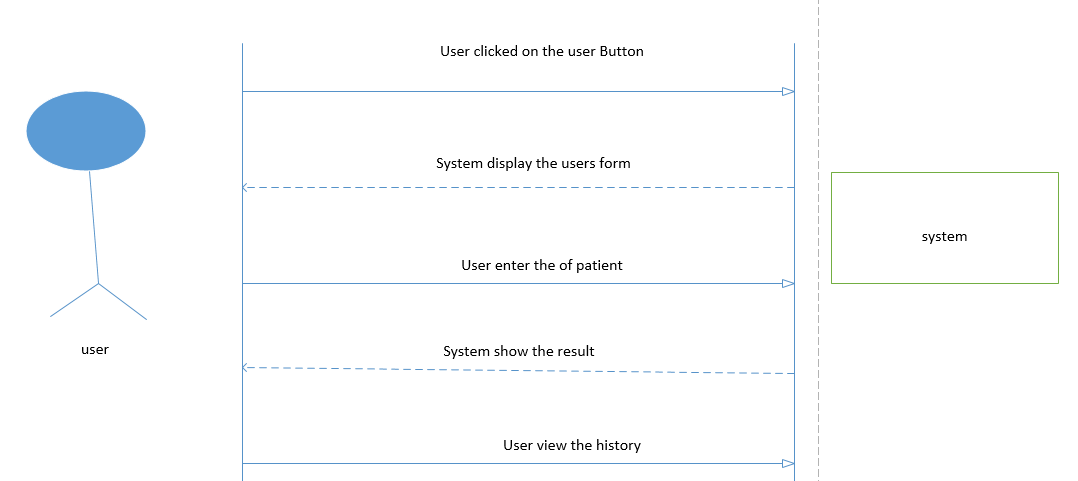
****

# 4.7 System sequence diagram

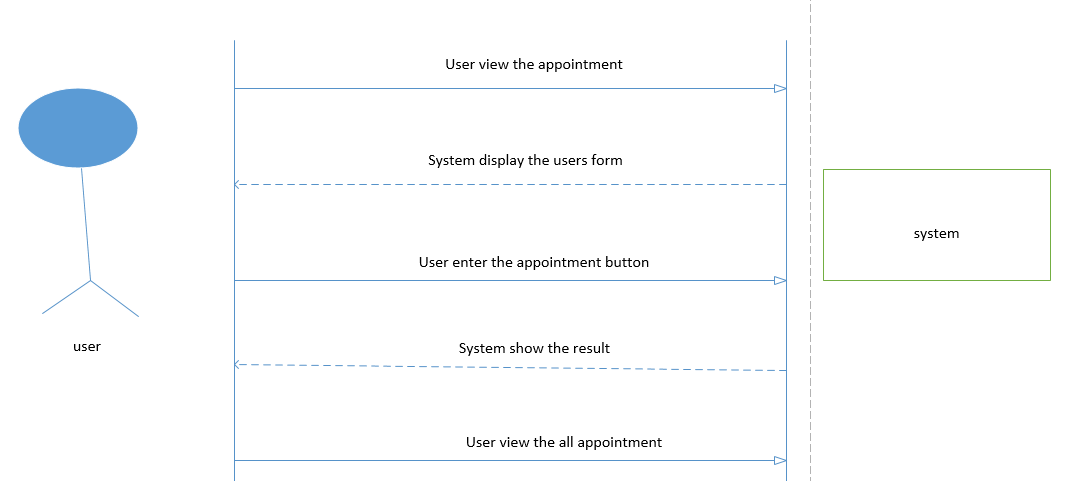
## 4.7.1 Scenarios.1 login :

****

## 4.7.2 Scenarios history:

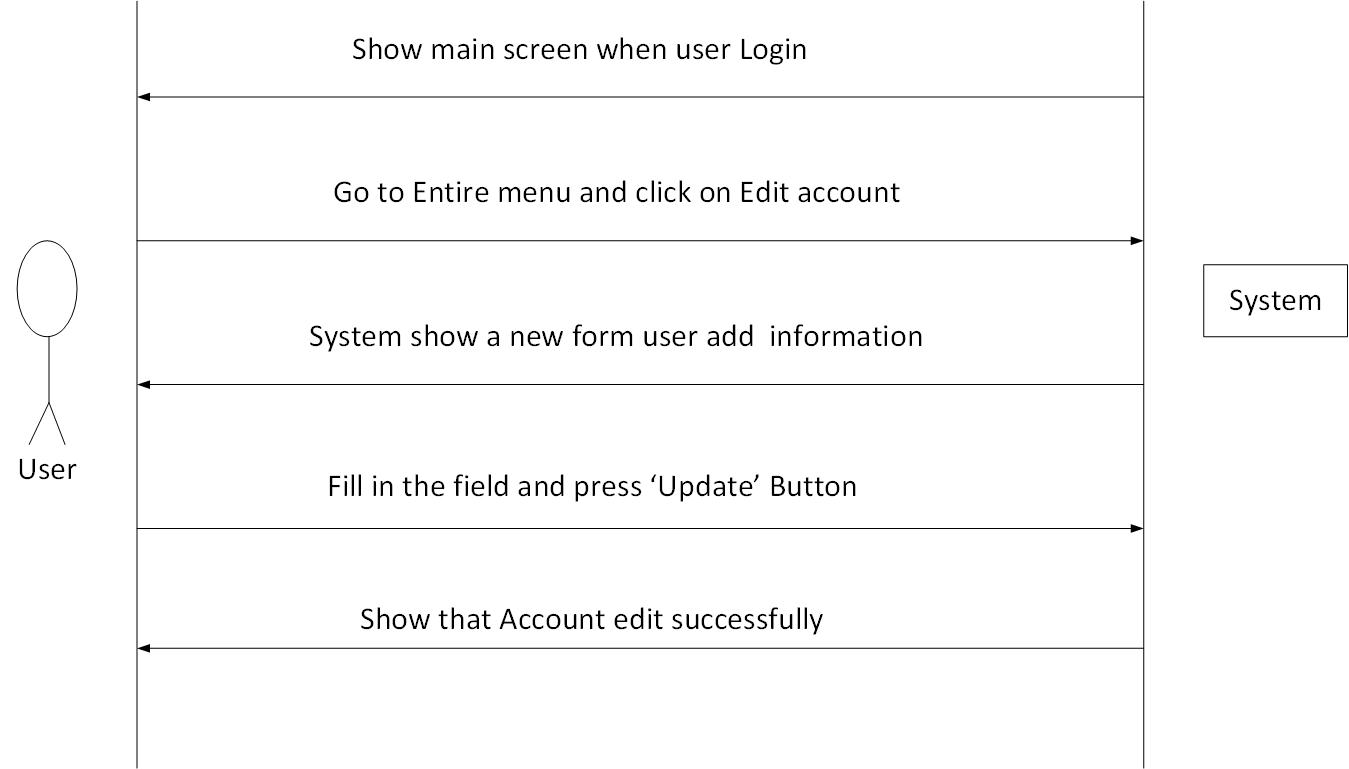
****

## 4.7.3 Scenarios. appointment

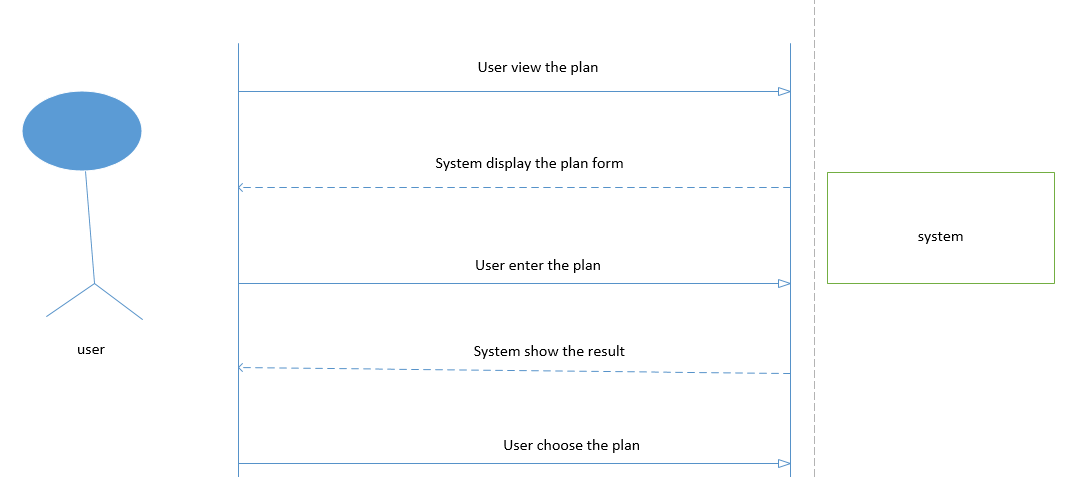


## 

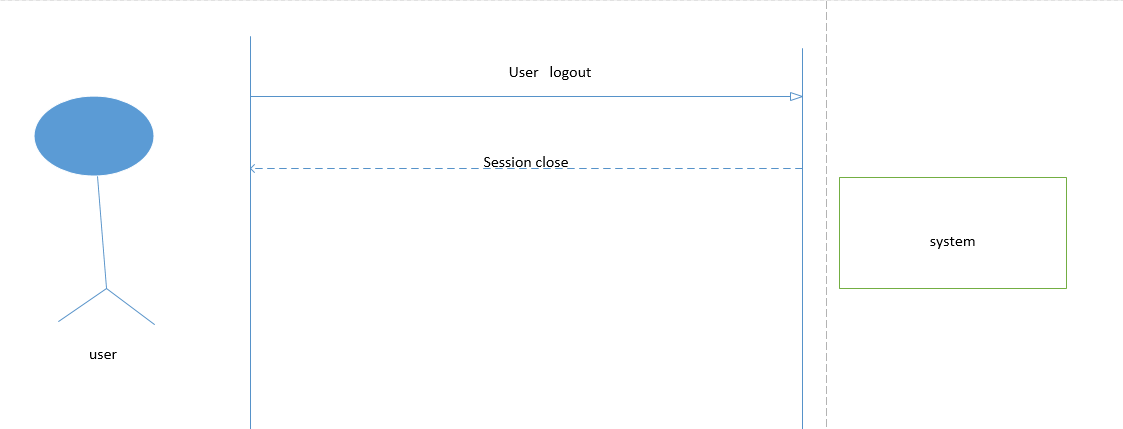
## 4.7.4 Scenarios.4 Edit Account:

****

## 4.7.5 Scenarios. plan :

****

## 4.7.6 Scenarios. logout

****

# 

# 

# Chapter No.05

*System Requirements*

# 5.1 Functional requirements:

* **To Manage Doctors Accounts**

This system will keep different Doctors Accounts.

* **To Manage Doctors Academic and Clinic record**

To manage Doctors Academic record is used to stores the detail of the doctors about his /her specialization and profile and viewed y user.

* **Patient info**

Will save all patients record information and make profile.

* **Company info**

Online doctor appointment web application make patients profile and save history of Appointments’

* **To manage schedule**

This section will manage schedule timing of doctors for patients and company appointments the schedule section will manage start timing and about doctors clinic timing and appoint the day for appointment.

* **To manage plan**

This module Plan Table is used to Stores the information of doctors and company’s

about their account login authentication through payment method **.**

* **To manage History**

History record Table is used to Stores the details of user’s profile and their appointment .history through notification

* **To manage Appointments**

Appointment Table is used to Stores the detail information of Appointments.

* **To search Doctors by location**

Patients can search nearby a doctors clinics.

# 5.2 Nonfunctional Requirements:

# Performance Requirements:

The System will take minimum time to access data from database.

# Safety Requirements:

To prevent data loss in case of system failure, the result of data that are insert and update till then have to be saved in online.

* **Security Requirements**:

It needs the authentic user name and password to secure our system privacy. If anyone wants to use it or hack it he/she will not allow using the system without authentic user name and password.

* **Speed and Latency Requirements:**

There is no speed and latency requirements.

* **Reliability and Availability Requirements:**

This application will be 100% reliable and available.

* **Scalability or Extensibility Requirements:**

The application will be scalable for any number of users and will be designed in such a way that it can be enhanced.

* **Longevity Requirements:**

The technology that will be used to develop PHP, JavaScript, CSS, MYSQL server. These technologies are actively playing the vital role in the development of this web technology.

* **Security Requirements**

**Access Requirements**

|  |  |
| --- | --- |
| **Admin** | Only admin (after successful login) can access the system |
| **Doctor** | Only authorized doctors (after successful login) can search patient the patient history |

**User interface:**

Interface of this product is very easy to use for user. User can easily move from one link to another link. He can insert, update, delete values easily and also check the animals status easily. He will not get any kind of difficulty in inserting, updating and deleting the values.

# 5.3. Software Quality Attributes

The most significant and glittering aspects and attributes of the system are: it is highly adaptable and available. The information which travels through it is extremely correct and secured. Admin works prudently to ensure its authenticity and maintainability. Its portability and robustness is represented by its online availability.

# 5.4 Hardware Requirement:

* Processor Pentium IV
* Clock Speed 5Ghz
* System Bus 32 Bit
* RAM 256MB
* Hard Disk 20GB
* Monitor SVGA
* Internet 512 Kbps
* Keyboard
* Mouse

# 

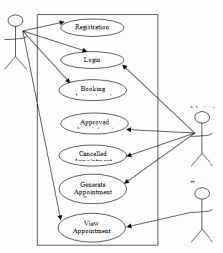
# 5.5 Software Requirements:

* Operating System Window Xp,7,8
* Browser front end Google Chrome, Mozilla Firefox, Opera Mini, Safari
* Front End Layer HTML,CSS
* Web Server PHP 2.4 :WAMPSERVER
* Server Side Scripting PHP
* Client Side Scripting Java Script
* Connection Protocol TCP/IP HTTP

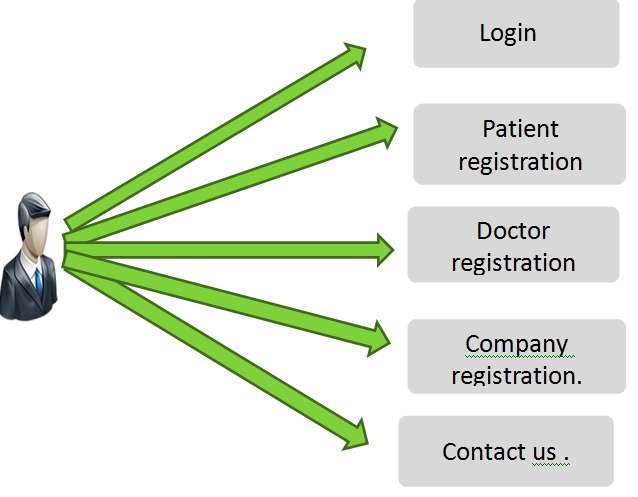
# *5.*6 Feasibility Studies:

A feasibility study is an evaluation and analysis of the potential of the proposed project which is based on extensive investigation and research to support the process of decision making. We firstly do feasibility study of the complete project. And specify the all user Requirements.

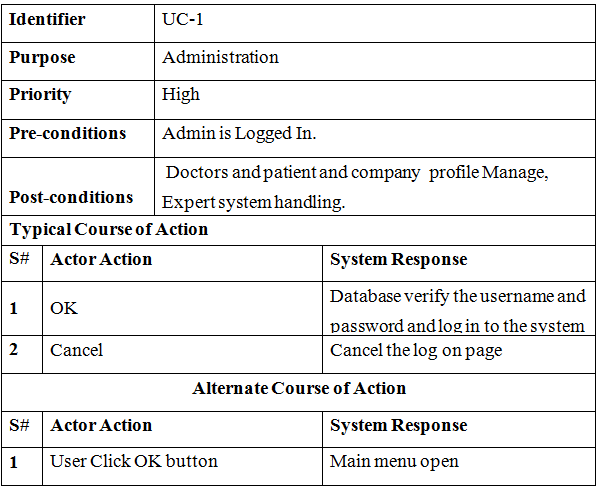
# 5.7Use Case diagram:



## 5.7.1 Use case admin

******

# 5.8 Use-Case: Administration:

****

# 

# 5.9 Use case Registration

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-1 | |
| **Purpose** | | Administration | |
| **Priority** | | High | |
| **Pre-conditions** | | Admin is Logged In. | |
| **Post-conditions** | | Doctors and patient and company profile Manage,  Expert system handling. | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | OK | | Database verify the username and  password and log in to the system |
| **2** | Cancel | | Cancel the log on page |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | User Click OK button | | Main menu open |

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-2 | |
| **Purpose** | | Registration | |
| **Priority** | | High | |
| **Pre-conditions** | | User would have internet and access to website | |
| **Post-conditions** | | Registered to software | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Select body part | | GUI of body parts will be open |
| **2** | Enter patient information | | Doctor registration successfully |
| **3** | Report generated | | Send to admin |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click on Expert System | | Expert system will be open |

# 

# 5.10 Use case Create profile:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identifier** | | UC-3 | | |
| **Purpose** | | Create profile doctor | | |
| **Priority** | | High | | |
| **Pre-conditions** | | Doctor is Logged In. | | |
| **Post-conditions** | | User would have internet and access to website | | |
| **Typical Course of Action** | | | | |
| **S#** | **Actor Action** | | **System Response** | |
| **1** | Select Body Part | | Create profile page open | |
| **2** | Start see appointment | | Patient check | |
|  |  | |  | |
| **Alternate Course of Action** | | | | |
| **S#** | **Actor Action** | | | **System Response** |
| **1** | User’s search option, if the patient  Revisited | | | Start searching using zip code |

# 5.11. Use case Patient:

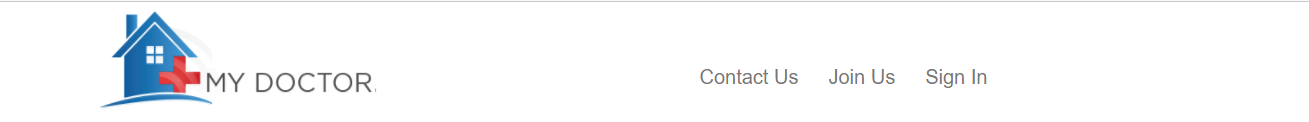
|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-4 | |
| **Purpose** | | Visit the patient profile . | |
| **Priority** | | Medium | |
| **Pre-conditions** | | Should have Internet facility | |
| **Post-conditions** | | Ability to view the patient profile . | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Open website | | Start browsing |
| **2** | Click on doctor profile | | Show list of patient profile |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click on Doctors | | Show list of doctors and GUI  interface to search doctors timing  and address. |

# 5.12 Interface Diagram:

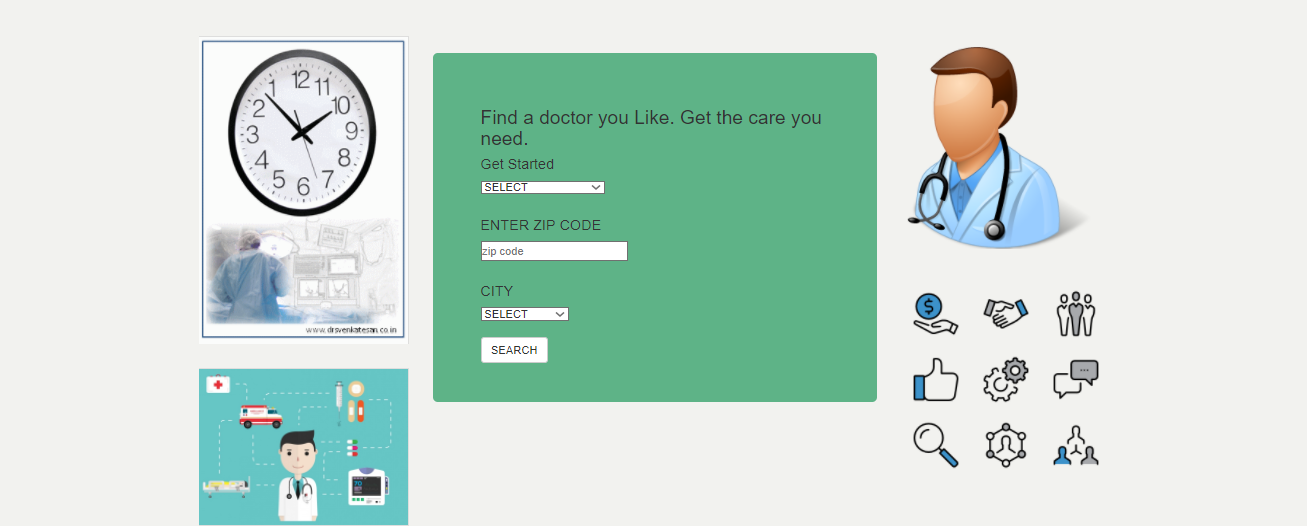
User interface is the front end designs that user first see when he/she logs into system.

**Some Screen shots are following:**

## 5.12.1 Main header

****

## 5.12.2From index page.

****

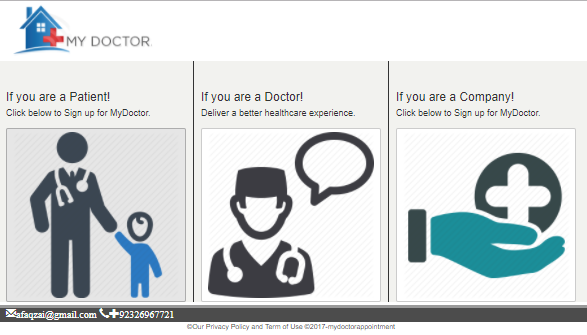
## 

## 5.12.3 If user /patient visit the E doctor then take appointment with doctor.

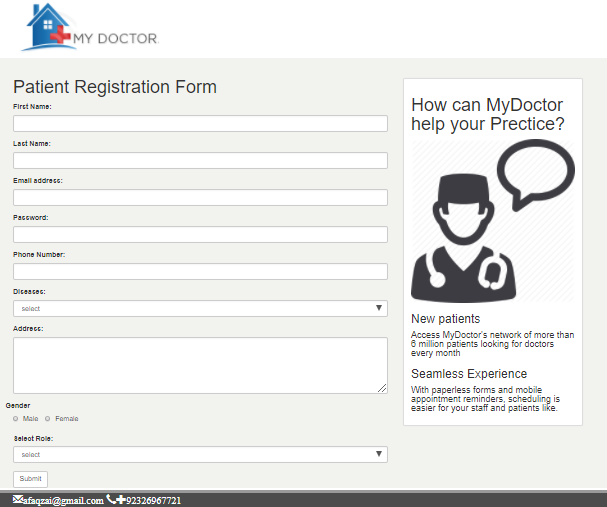
****

**5.12.4 Registration / join Now Form**

**When user click on join now**

****

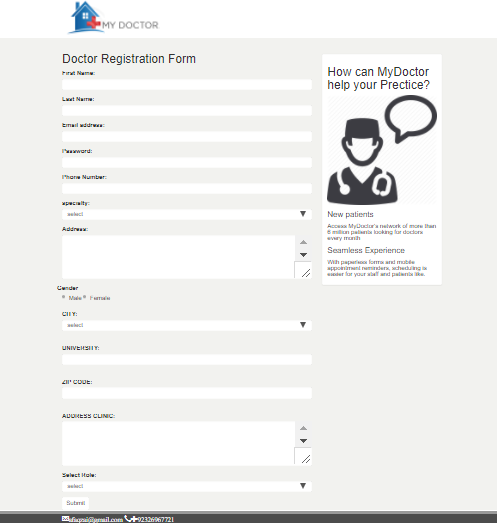
**5.12.5 If you are a new patient then register from this form:**

****

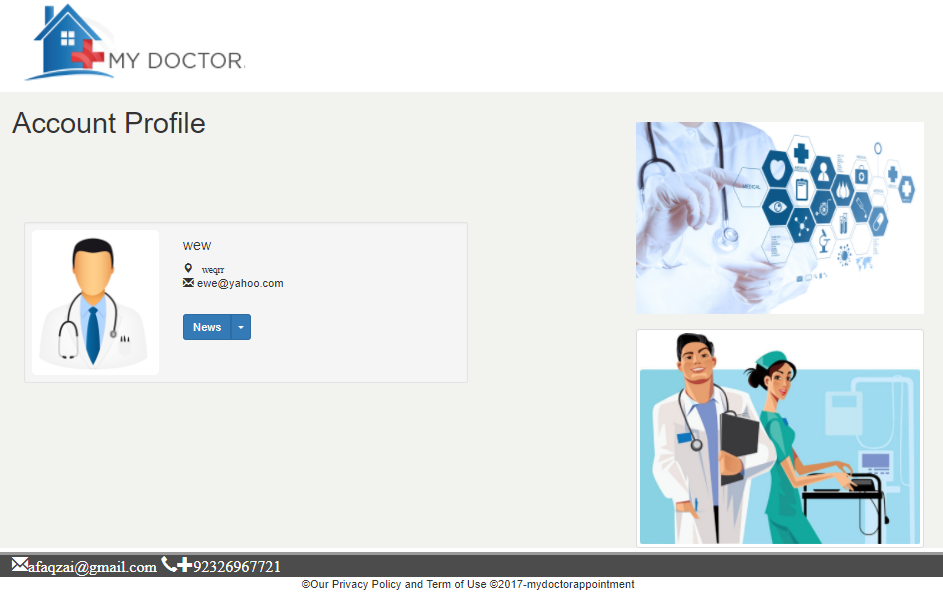
**After that go to Appointment …..**

## 

## 5.12.6 If you are a doctor then register from this form:

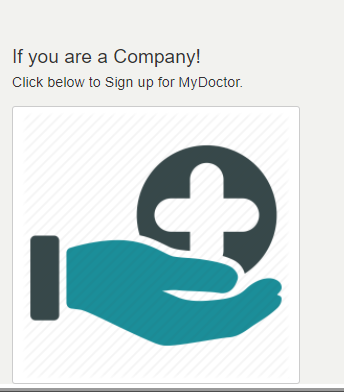


**5.12.6 After join now doctor make profile and complete the following steps.**

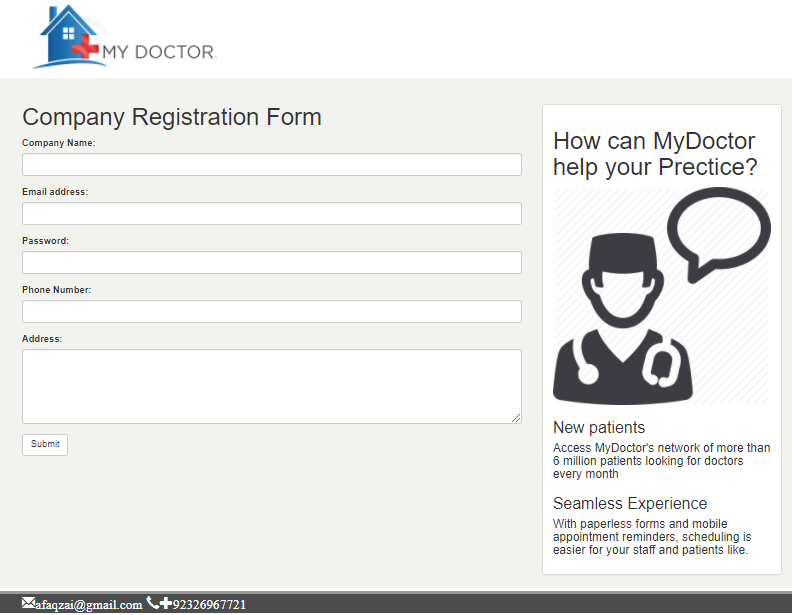
****

## 

## 5.12.10 If you’re a company then join now

****

## 5.12.11Then register from this form.

****

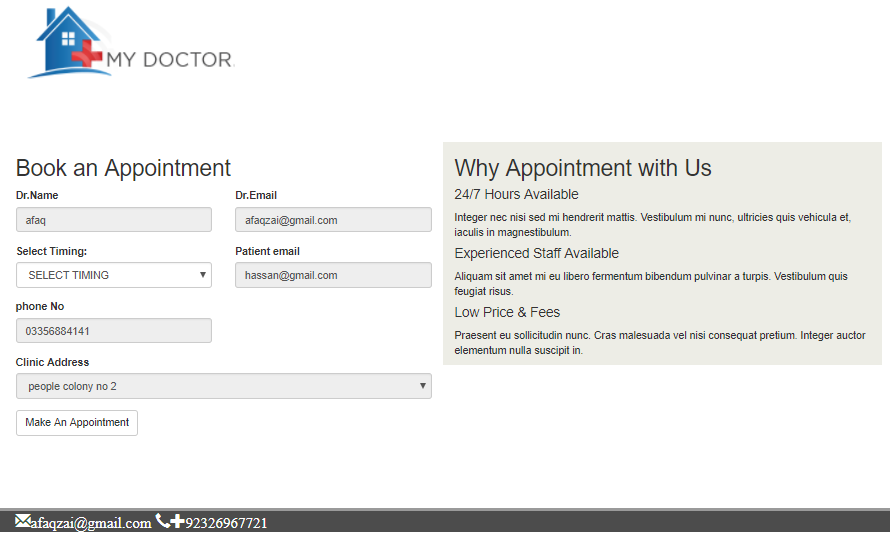
## 

## 5.12.1 Then go to his own profile page. And Search the doctor

## 

****

## 5.12.13 take appointments with doctor

****

# Chapter No.06

*Implementation and Testing*

# 

# 

# 6.1 Implementation*:*

Implementation is a procedure of practically deployment of your system on the required hardware and Software along with proper tools. Here in our case we use Parallel Implementation.

# 6.2 Parallel Implementation:

Is a type of implementation in which the developed system runs parallel to the old manual System for the specific time period until user satisfied by the performance of the developed system.

# 6.3-Testing Techniques:

Test is the name given to that focuses on web applications. Test means to test the project deeply and correction of the errors .if the test of the project is good and correct then the project run easily and correct manner .so the test of the project is very important implementation means that run the project easily without the errors.

So the complete implement is very important .so we test all the modules one by one test First we test one of the module and check this module work properly. For example our project E doctor we make the admin panel module and check to challenge quickly and completely.

The next step is test the user side works if the user enter our website what factor can be involve and what kind of difficulty can be face. Test method is very important because if the project run and many errors are accurse and the website is not completely manage easily. So module by module test is helpful to your project run.

Also test equipment of the project run correctly .one of the important part of the test is server test. Server test means if the user enter the website never face any type of problem and quickly enter in website if large number of user is enter in the same time to see do not slow down the server speed. Because if the server speed slow down many chance to website close all pages.

Analyze the working of developed system after implementing is known as testing there are few techniques we use, they include:

* Unit Testing
* Integration Testing
* Black Box Testing
* White Box Testing

## 6.3.1 Unit testing:

Unit test is a method by which individual test units of source code in unit test smallest part of the module test one by one. If any errors are accurse then correct these errors on by one. Because in unit test individual code and procedure may be test one by one Unit tests are short code fragments. Unit tests are typically written and run to ensure that code meets its design and behaves as intended so unit test is also called the module test.

So that each of the modules test one by one any error is occur to correct the errors module by module. So many of the modules are use in our website like Admin Panel. Milking information breeding, etc .So all the module checks one by one Admin panel is very important because the entire website is control.

Admin is always checkout the menus and check all the user activity and see the account balance. Unit test is test the speed of the website burden because many users are entering our website the speed of the server is slowdown.

## 6.3.2 Integration testing:

Integration test on the other hand is a different form of test in which the interaction between two or more units is explicitly tested. Integration means to merge of two or more modules and then test and what the error show or not and what the link or the pages are work correctly or not .So all the page of the website is link to each other.

Integration test a good point is this if the one module is work correct the link module is also work correctly. So Admin panel is one module they are link to the user account because every person are enter our website they directly connect in our admin so the admin are see all the activity of the user and also user address .Hence we test multiple probability modules for proper working of the system.

## 6.3.3 Black Box testing:

The technique of testing without having any knowledge of the interior workings of the application is Black Box testing.

The tester is oblivious to the system architecture and does not have access to the

Source code. Typically, when performing a black box test, a tester will interact with the system's user Interface by providing inputs and examining outputs without knowing how and where the inputs are worked upon.

## 6.3.4 White Box testing*:*

White box testing is the detailed investigation of internal logic and structure of the code. White box Testing is also called glass testing or open box testing. In order to perform white box testing on an Application, the tester needs to possess knowledge of the internal working of the code. The tester needs to have a look inside the source code and find out which unit/chunk of the code is Behaving inappropriately

# 6.4 Test cases:

Test causes means to test your project in external person. Those people check your entire module one by one and find the errors and want to correction of your mistakes. If any module is not work easily to check there errors .So it is very important to check all the modules before the test cases. Test cases are formal examinations designed to validate certain functionality from the legacy system that should be preserved in the migrated application.

## 6.4.1Admin Module Test Cases:

|  |  |
| --- | --- |
| **Preconditions** | Admin logged to the system. If the user name and password is invalid  then it will not be proceed. And redirected back to the same page. |
| **Actions** | Open the **Doctor.**  Update existing **Doctor**.  Delete **Doctor**.  Click on the Submit. |
| **Expected Results** | **Doctor** management operation perform successfully.  If admin enter data then the desired changing occurs.  Admin has not any validation on text boxes. |
| **Tested By** | Afaq khan |
| **Result** | Pass |
| **Identifier** | TC-2 |
| **Priority** | Medium |
| **Input data** | **Doctor** selection |
| **Short description** | This process is perform to manage **admin**. |

|  |  |
| --- | --- |
| **Preconditions** | Website opens in browser. Enter URL of admin login, then login page  show.. |
| **Actions** | Open the Login Page  Enter the Admin Id and Password    Click on the login |
| **Expected Results** | Admin page will be Open  If username or password field is empty or wrong, it will not proceed  and redirect back |
| **Tested By** | Umair ilyas |
| **Result** | Pass |
| **Identifier** | TC-1 |
| **Priority** | Medium |
| **Input data** | Click login |
| **Short description** | This process is perform to access admin panel and perform |

## 6.4.2Test case Manage Doctor:

|  |  |
| --- | --- |
| **Preconditions** | Admin logged to the system. If the user name and password is invalid  then it will not be proceed. And redirected back to the same page. |
| **Actions** | Open the **Doctor.**  Update existing **Doctor**.  Delete **Doctor**.  Click on the Submit. |
| **Expected Results** | **Doctor** management operation perform successfully.  If admin enter data then the desired changing occurs.  Admin has not any validation on text boxes. |
| **Tested By** | Afaq khan |
| **Result** | Pass |
| **Identifier** | TC-2 |
| **Priority** | Medium |
| **Input data** | **Doctor** selection |
| **Short description** | This process is perform to manage **Doctor** profiles. |

## 

## 6.4.4Test Case Company Module Test Case:

|  |  |
| --- | --- |
| Preconditions | Website opens in browser. Select Company Login(Click here to access  Company login), then login page show. |
| Actions | Open the Login Page  Enter the Company Id and Password  Click on the Login |
| Expected Results | Company page will be Open  If username or password field is empty or wrong, it will not proceed  and redirect back. |
| Tested By | Afaq khan |
| Result | Pass |
| Identifier | TC-6 |
| Priority | High |
| Input data | Click login |
| Short description | This process is perform company |

# Chapter No.07

*Conclusion and possible Extension*

# 7.1 Conclusion:

With the development of web based technology online doctor’s appointment system, patients are able to book and manage their own appointment with ease. They will be reminded of their appointments via notification that will be promptly sent to them before their appointment date. The system itself also provides a quick view of their appointment at the appointment page. The system is mainly designed to facilitate appointment booking between the patient and the health personnel and company. The system delegates some administrative work to the patients by allowing them to manage their own appointment and personal profiles. Time will not be wasted on converting paper-based appointment record into electronic-based.

In this system clinic areas are available the system will provide users with multiple clinics areas of different location

The objective of this research is that experts should feel good by providing immediate access to the public, the benefits of their expert opinion and vice versa. Methodology of this project is that top medical experts and their trusted professional Doctors will be available online and the patients will have access to the by online registration in their own area anytime, anywhere.

# 7.2 Possible Extension

* In future we can be build mobile phone App of online Doctors appointment of this project.
* Hardware can be enhanced in order to fully automate the system.
* In addition we can be build SMS Service of Appointment.

## 7.2.1 Advantages

This system helps to reduce the waiting time of the patient.

User can select the appointment time according to his preference.

Available and booked slots are shown in effective graphical user interface.

## 7.2.3 Disadvantages

It requires an internet connection.

It requires large database.

# References

 [http://www.stackoverflow.com](http://www.stackoverflow.com/)

 [http://www.webmd.com](http://www.webmd.com/)

 [http://stackoverflow.com/questions/18650972/javascript-validation-for-both-non-](http://stackoverflow.com/questions/18650972/javascript-validation-for-both-non-empty-and-number-alphabets)

empty-and-number-alphabets

 [http://www.daniweb.com/web-development/javascript-dhtml-](http://www.daniweb.com/web-development/javascript-dhtml-ajax/threads/312069/validation-using-javascript-for-alphabet-only)

ajax/threads/312069/validation-using-javascript-for-alphabet-only

<http://www.smartwebby.com/DHTML/phone_no_validation.asp>

 [http://stackoverflow.com/questions/4471401/getting-value-of-html-checkbox-from-](http://stackoverflow.com/questions/4471401/getting-value-of-html-checkbox-from-onclick-onchange-events)

onclick-onchange-events

<http://stackoverflow.com/questions/3862098/call-javascript-on-checkbox-onclick>

 [http://stackoverflow.com/questions/19580392/javascript-getting-my-textbox-to-display-](http://stackoverflow.com/questions/19580392/javascript-getting-my-textbox-to-display-a-variable)a-variable

<http://www.w3schools.com/tags/att_textarea_form.asp>

<http://www.w3schools.com/tags/att_textarea_maxlength.asp>

 [http://stackoverflow.com/questions/24305278/php-version-upgrade-from-5-3-to-5-4-x-](http://stackoverflow.com/questions/24305278/php-version-upgrade-from-5-3-to-5-4-x-which-creates-issue-in-login-validation)which-creates-issue-in-login-validation

 [http://techblog.wimgodden.be/2012/03/04/php-5-4-compatibility-checks-using-](http://techblog.wimgodden.be/2012/03/04/php-5-4-compatibility-checks-using-php_codesniffer/)

php\_codesniffer/

 [http://stackoverflow.com/questions/24294513/i-have-an-issue-with-login-validation-in-](http://stackoverflow.com/questions/24294513/i-have-an-issue-with-login-validation-in-php)php

<http://www.php-login.net/>

 [http://stackoverflow.com/questions/20025637/login-and-validation-in-php-with mysqli](http://stackoverflow.com/questions/20025637/login-and-validation-in-php-with-mysqli)

<http://w3schools.invisionzone.com/index.php?showtopic=46435>

<http://www.plus2net.com/php_tutorial/php_login_script2.php>

<http://www.phpro.org/tutorials/Basic-Login-Authentication-with-PHP-and-MySQL.html>

<http://stackoverflow.com/questions/384276/how-to-create-search-engines-like-google>

<http://www.thesitewizard.com/sitepromotion/search-engine-friendly.shtml>

 http://mrbool.com/how-to-create-your-own-search-engine-with-php-and-mysql/25836

 [http://www.w3schools.com](http://www.w3schools.com/)

 [http://www.tizech.com](http://www.tizech.com/)

 [http://www.killerphp.com](http://www.killerphp.com/)

 [http://www.php.net](http://www.php.net/)