CHAPTER 1

Introduction

Time is most precious commodity a neurologist has. We are developing an appointment scheduler so as to let a person schedule appointment without his presence around. Software requirement specification describes all the details describing the various attributes of appointment scheduler.

It gives description of the purpose and scope of appointment scheduler, about the appointment that would be performed and the basic framework as to how the appointment are intended to be scheduled with respect to the stimulated time and order. This SRS delineates the aspects such as reference and also compares the existing system with the proposed system.

Neurologist are medical doctors who specialize in diagnosing and treating conditions related to central and peripheral nervous system. They mainly check patients who have been referred by primary care provider, but occasionally they come from emergency room, physician referrals or referrals from other specialists.

Effective scheduling lets decision makers allocate available human resources to appointments in a way that ensures on-time completion of related business goals. **Appointment scheduler** software helps you assign an appropriate number of workers to each appointment, set activity time-frames and durations, create job schedules, prioritize todo lists and receive automatic notifications.

1.2 The Benefits of Effective Appointment Scheduling

If there were no detailed schedule, it would be difficult to link appointment assignments to working hours. You would have just a todo list with multiple appointments assigned to you, but you couldn't realize when each one should be started and what deadlines should be met. Then lack of scheduling would make you inefficient at the workplace, and most likely you would just waste your working hours, rather than deliver expected value to your company. Besides, you couldn't collaborate with your colleagues and management.

So the greatest advantage of having a detailed job schedule in place is predictability. Your daily schedule tells you what appointments are to be done, what deadlines are to be met, and what priorities are to be followed. This document allows you to predict your possible steps and plan for effective use of resources, tools and equipment. Other possible benefits of appointment scheduling are as follows:

- Daily work is planned according to strategic priorities
- Decreased turnaround time
- Improved communications and collaboration with personnel
- Avoidance of overloaded and underused resources
- Job delays and schedule deviations are more readily apparent
- An ability to review and forecast personnel effectiveness

1.3 Why Use Appointment Scheduler

All of these benefits can be achieved with use of effective appointment scheduler software. For example, in Appointment Scheduler you can use the Calendar view to create and manage job schedules. The program lets display appointments on each user's calendar, so you can get multiple calendars displayed on a single screen. This feature makes it easier to review and manage several schedules at once. In addition, you can switch between daily, weekly and monthly modes to easily schedule short-term, mid-term and long-term appointments.

1.4 Priorities, Time-frames & Due Dates to Schedule Todo Lists

Priorities, time-frames and due dates help make job scheduling more accurate and aligned with business objectives. Prioritization lets you organize and rank your appointments by their urgency and importance, so that you will know what items in your todo list should come up first. Time-frames tell you expected start and finish time or duration for each todo item - you can use this data to plan your actions more effectively. Due dates can be used to link your todo items to business goals and projects. In Appointment Scheduler you can set different priority levels (from Urgent and Highest to Normal and Lowest), due dates and start & finish time for your appointments. The software lets make your todo list easier to read and understand - you can group all your appointments by the Priority column in the Appointment List view.

Chapter 2

Requirement Analysis

This section lists the software and hardware functional requirements.

2.1 Hardware Requirements:

2.1.1 The Laptop

Table: 2.1

Requirement Number	Requirement Category	Description of the requirement	Implementation Phase
2.1.1.1	Laptop Speed	The laptop may have a processing speed of at least 2.4 Hz.	Phase 1
2.1.1.2	Laptop RAM	The mobile should have RAM, which is a form of data storage that stores data and machine code currently being used. The laptop may have a RAM of at least 4 GB.	Phase 1
2.1.1.3	Laptop OS	The laptop may have Windows OS as its operating system.	Phase 1
2.1.1.4	Laptop Processor	The laptop may have a processor of any company like intel, AMD,etc.	Phase 1
2.1.1.5	Laptop System Type	The laptop may have system type as 64-bit OS, x64-based processor.	Phase 1

2.1.2 The Mobile

Table: 2.2

Requirement Number	Requirement Category	Description of the requirement	Implementation Phase
2.1.2.1	Mobile Speed	The mobile may have a processing speed of at least 2.4 Hz.	Phase 1
2.1.2.2	Mobile RAM	The mobile should have RAM, which is a form of data storage that stores data and machine code currently being used. The mobile may have a RAM of at least 2 GB.	Phase 1
2.1.2.3	Mobile ROM	ROM is a type of non-volatile memory used in computer and all electronic devices. The mobile may have a ROM of at least 16 GB.	Phase 1
2.1.2.4	Mobile OS	The mobile may have Android OS or as its operating system.	Phase 1

2.2 Software Requirements:

2.2.1 The Software

Table: 2.3

Requirement Number	Requirement Category	Description of the requirement	Implementation Phase
2.2.1.1	Database	Appointment scheduling system Shall be installed in the Machine.	Phase 1
2.2.1.2	LAMP	Linux ,Apache,Mysql,PHP server	Phase 1

2.3 Functional requirements:

The Web App:

2.3.1 The Dashboard Module

Table: 2.4

Requirement Number	Requirement Category	Description of the requirement	Implementation Phase
2.3.1.1	Dashboard Page	The user shall be directed to the Dashboard page to Select the option	Phase 1
2.3.1.2	Book Appointment	Here user can book appointment	Phase 1
2.3.1.3	View Appointment	Here user can view booked appointment	Phase 1
2.3.1.4	Cancel Appointment	Here user can cancel booked appointment	Phase 1
2.3.1.5	Doctor's Login	Here Doctor can log in	Phase 1

2.3.2 Doctor's Login Module

Table: 2.5

Requirement Number	Requirement Category	Description of the requirement	Implementation Phase
2.3.2.1	Login Page	The Login page shall be displayed on the load of the web app.	Phase 1
2.3.2.2	Username	The Page shall contain a username Text box to type in the Username.	Phase 1
2.3.2.3	Password	The Page shall contain a Password Text box to type in the Password.	Phase 1
2.3.2.4	Login Button	The page shall contain a Login button below the boxes to enter into the main pages.	Phase 1

2.3.3 The Appointment Module

Table: 2.6

Requirement Number	Requirement Category	Description of the requirement	Implementation Phase
2.3.3.1	Appointment Page	The appointment page shall be Displayed on the load of the web app.	Phase 1
2.3.3.2	First name	The Page shall contain a firstname Text box to type in the First name of the user.	Phase 1
2.3.3.3	Last name	The Page shall contain a lastname Text box to type in the Last name of the user.	Phase 1
2.3.3.4	Email id	The page shall contain a email id Text box to type in the Email id of the user.	Phase 1
2.3.3.5	Mobile Number	The page shall contain a mobile number Text box to type in the Mobile number of the user.	Phase 1
2.3.3.6	Date Of Appointment	The page shall contain a date Text box to type in the date of appointment of user	Phase 1
2.3.3.7	Time Of Appointment	The page shall contain a time of appointment which will set time of appointment of the user.	Phase 1
2.3.3.8	Gender	The page shall contain a gender of user.	Phase 1

2.3.3.9	DOB	The page shall contain a DOB of user.	Phase 1
2.3.3.10	Address	The page shall contain a address of user.	Phase 1
2.3.3.11	Doctor	The page shall contain a doctor's name and their respective hospital.	Phase 1
2.3.3.12	Symptoms	The page shall contain a various symptoms that can be experienced by user.	Phase 1

2.4 Non – Functional requirements:

Table: 2.7

Requirement Number	Requirement Category	Description of the requirement	Implementation Phase
2.4.1	Performance	The system must be interactive and the delays involved must be less. So in every action – response of the system, there are no immediate delays.	Phase 1
2.4.2	Safety	Information transmission should be securely transmitted to the server without any changes in information.	Phase 1
2.4.3	Reliability	As the system provides the right tools for discussion, problem solving it must be made sure that the system is reliable in its operation and for securing the sensitive details.	Phase 1
2.4.4	Usability	As the system is easy to handle and navigates in the most expected way with no delays. In that case the system program reacts accordingly and transverses quickly between its states.	Phase 1

2.4.5	Availability	If the internet service gets disrupted while sending information to the server, the information can be send again for the verification.	Phase 1
2.4.6	Security	The main security concerns is for users account hence proper login mechanism should be used to avoid hacking. Hence, security will be provided by password authentication.	Phase 1

2.5 PLANNING

Cost Estimation:

1. Objectives:

- Link information systems to business needs.
- Learn how to create a system request.
- Understand system feasibility (Applicability).
- Understand how to select a project.
- Become familiar with work breakdown structure, Gantt charts and network diagrams.
- Become familiar how to staff a project.
- Perform a feasibility analysis; revise the system request.
- Approve or decline the project.

2. Feasibility Study:-

Feasibility analysis begins once the goals are defined. It starts by generating broad possible solutions, which are possible to give an indication of what the new system should look like. This is where creativity and imagination are used. Analysts must think up new ways of doing things-generate new ideas. There is no need to go into the detailed system operation yet. The solution should provide enough information to make reasonable estimates about project cost and give users an indication of how the new system will fit into the organization.

1. **Economically Feasibility**

   Development of this application is highly economically feasible. The only thing to be done is making an environment with an effective supervision.

   It is cost effective in the sense that has eliminated the paper work completely. The system is also time effective because the calculations are automated which are made at the end of the month as per the user requirement.

2. **Technical Feasibility**

   The technical requirement for the system is economic and it does not use any other additional Hardware and Software. Technical evaluation must also access whether the existing systems can be upgraded to use the new technology.

3. Operational Feasibility

The system working is quite easy to use and learn due to its simple but attractive interface. User requires no special training for operating the system. Technical Performance includes issues such as determining whether the system can provide the right information for the Department personnel.

	2019	2020	2021
Increased Sales	25,000	40,000	53,000
Reduction in Customer Complaint calls	4,750	5,000	5,500
Total Benefits	**29,750**	**30,000**	**44,000**
PV of Benefit	**34,500**	**31250**	**25000**
Server Cost	1000	1000	1000
Printer	0	0	0
Development Labor	5,00	500	500
Total Development Cost	**25000**	**1000**	**1000**
Hardware	200000	200000	200000
Software	350	350	350
Operational Labor	5000	5000	5000
Total Operational Cost	**10,050**	**10,050**	**10,050**
Total Cost	**30,250**	**16,050**	**16,050**
PV of Cost	**30,250**	**10,031**	**6,270**
Total Project Benefit - Costs	**4,250**	**21,219**	**18,730**
Yearly NPV	**4,250**	**12,735**	**14,733**

Chapter 3

System Design

3.1 DFD 0 DIAGRAM:

Here it is shown the process of booking appointments.

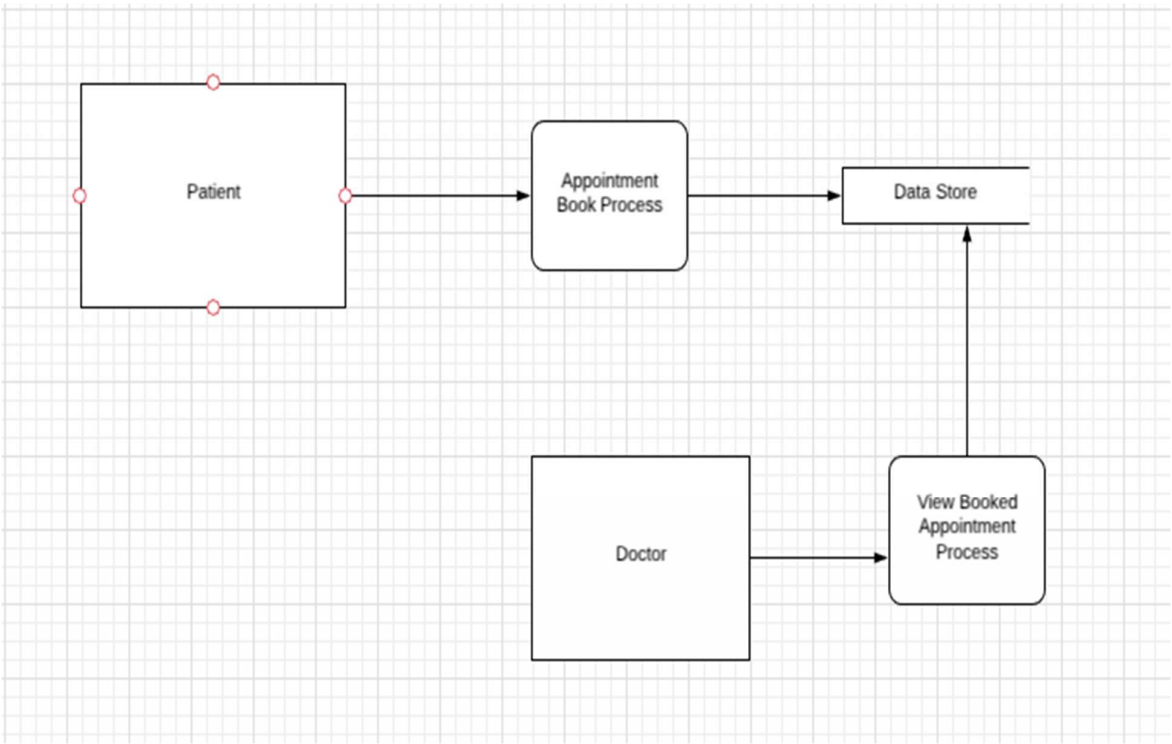

Fig.3.1 DFD 0 Diagram

3.2 SEQUENCE DIAGRAM:

Here diagram shows the working interaction between the user, appointment scheduler & database.

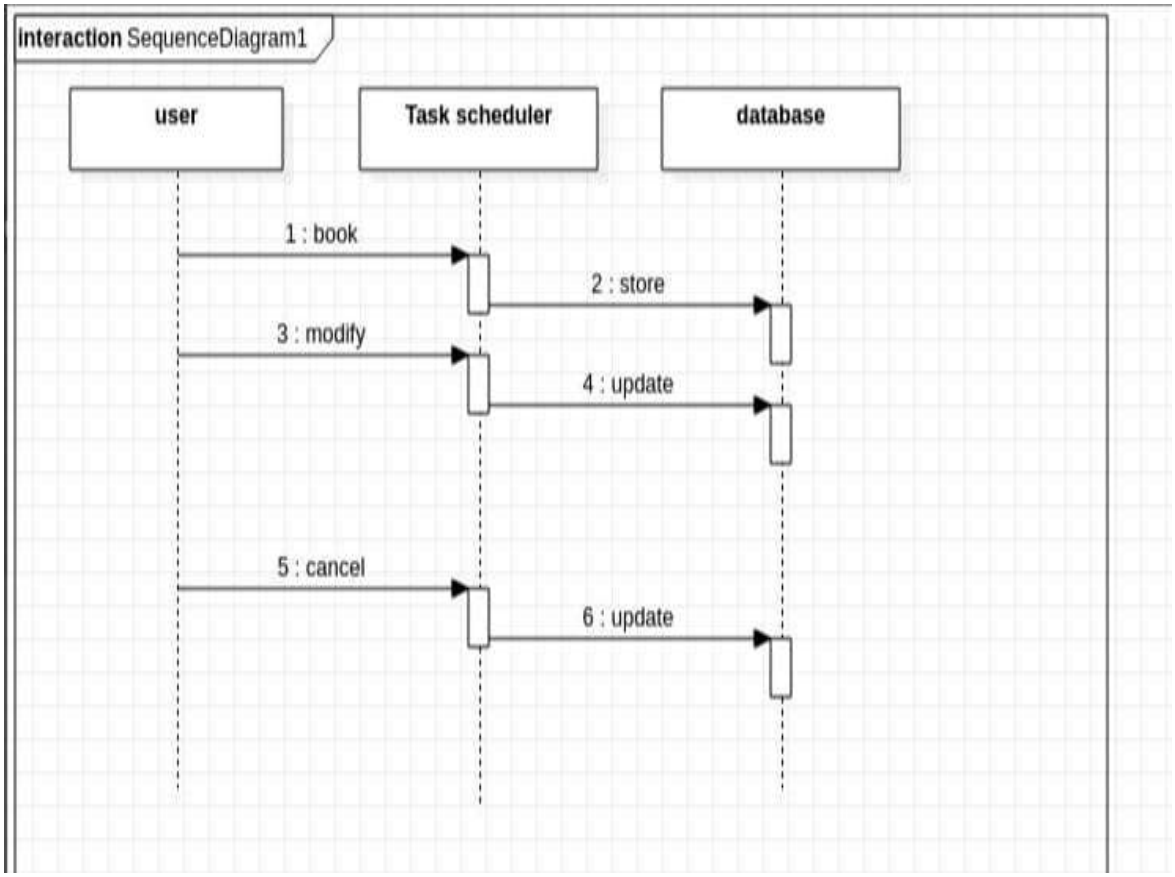

Fig.3.2 Sequence Diagram

3.3 ACTIVITY DIAGRAM:

Homepage has four options such as book apppointment, view appointment, cancel appointment & doctors login and each one of them has their own work flow.

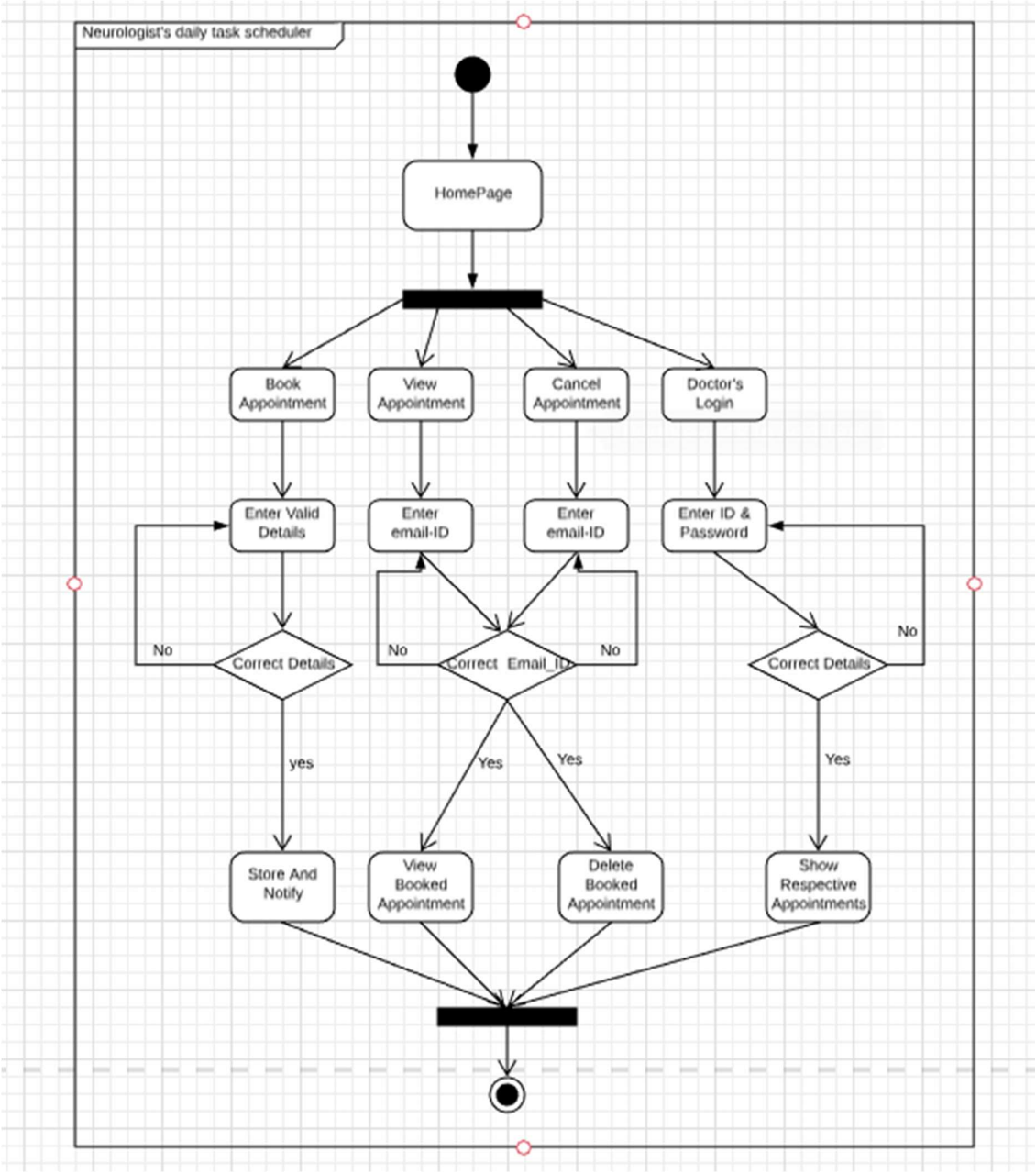

Fig.3.3 Activity Diagram

3.4 USE CASE DIAGRAM:

The user can book, view & cancel appointment, to book appointment you need to enter appointment details, it will be stored in database, it must have valid email id to view or cancel & the email id will be stored in database.

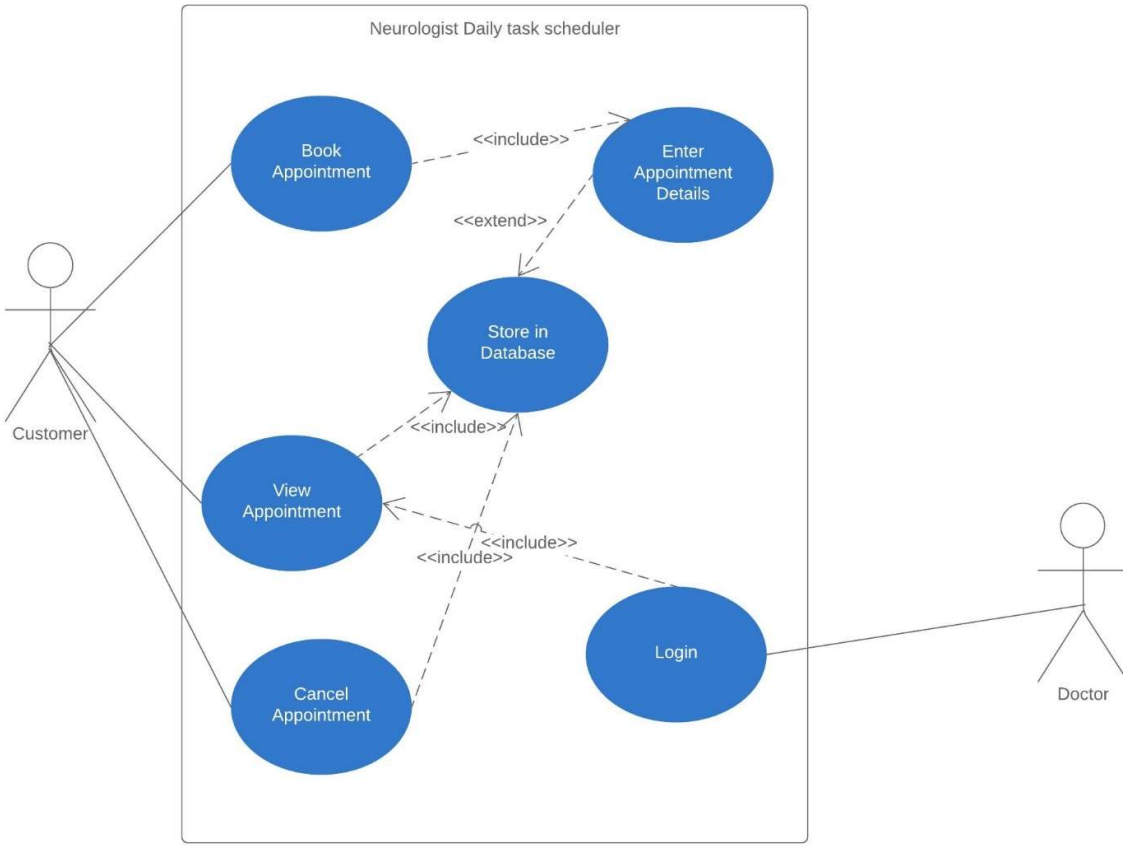

Fig.3.4 Use Case Diagram

3.4.1 USE CASE DESCRIPTION

Table 3.1: Use Case description for booking an appointment

Use Case Name: Appointment booking		**ID :1**	**Importance Level:** High
Primary Actor: User			
Description: User should be able to book appointment			
Type: External			
Trigger: Confirmation of booking			
Major Input: Valid information		**Major Output:** Text message	
Description	**Source**	**Description**	**Destination**
Storage of information	Patients	Confirmation by sending text	Next Page

Table 3.2: Use Case description for viewing the appointment

Use Case Name: Appointment Viewing		**ID :2**	**Importance Level:** High
Primary Actor: Doctor			
Description: User should be able to view booked appointment			
Type: External			
Trigger: View list of appointments			
Major Input: Valid login		**Major Output:** List of appointments	
Description	**Source**	**Description**	**Destination**
Enter registered login id and password	Doctor	Confirmation by allowing access to stored data	Next Page

3.5 ARCHITECTURAL DIAGRAM:

It shows the architecture of the software.

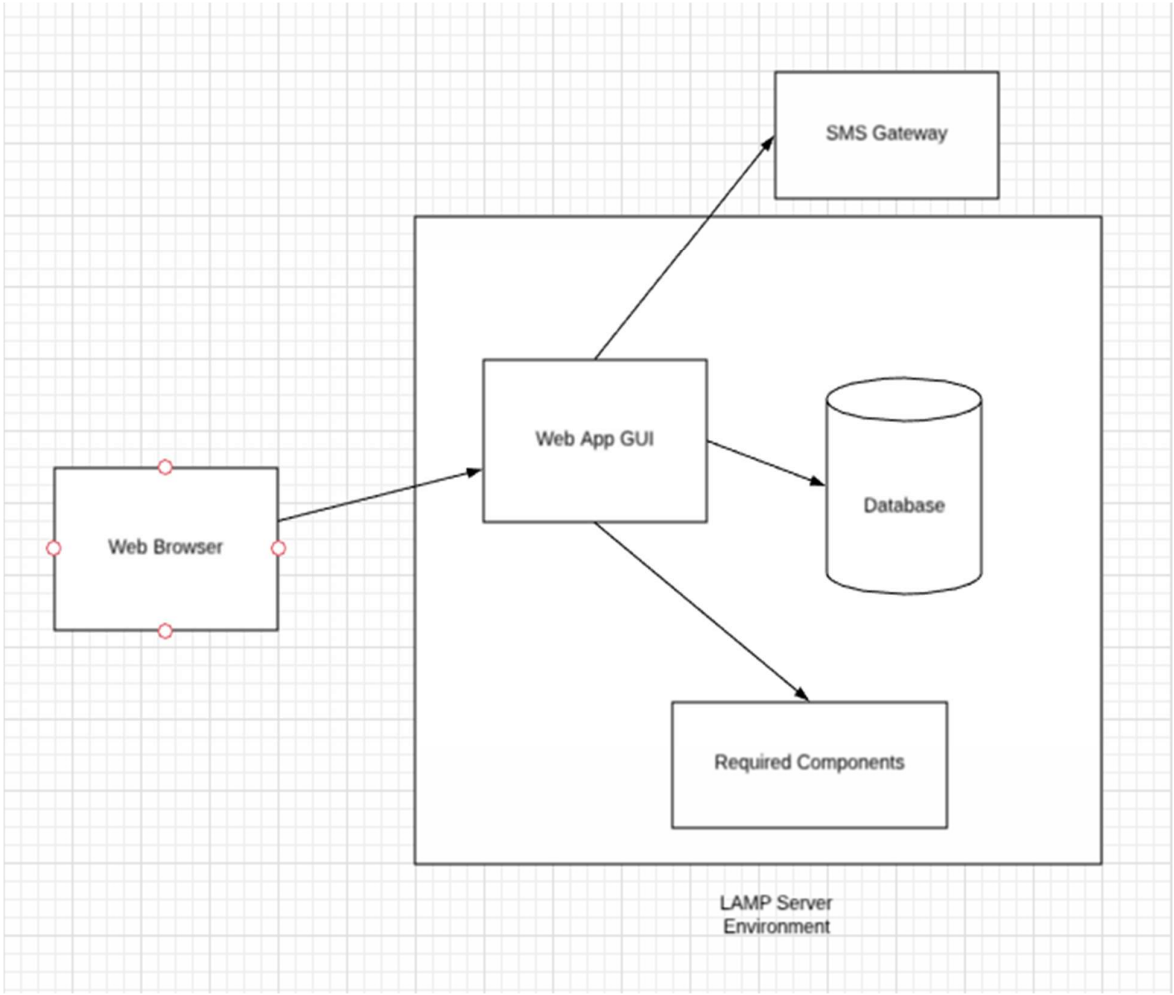

Fig.3.5 Architectural Diagram

3.6 E-R DIAGRAM:

It is diagram to show entity relation between database.

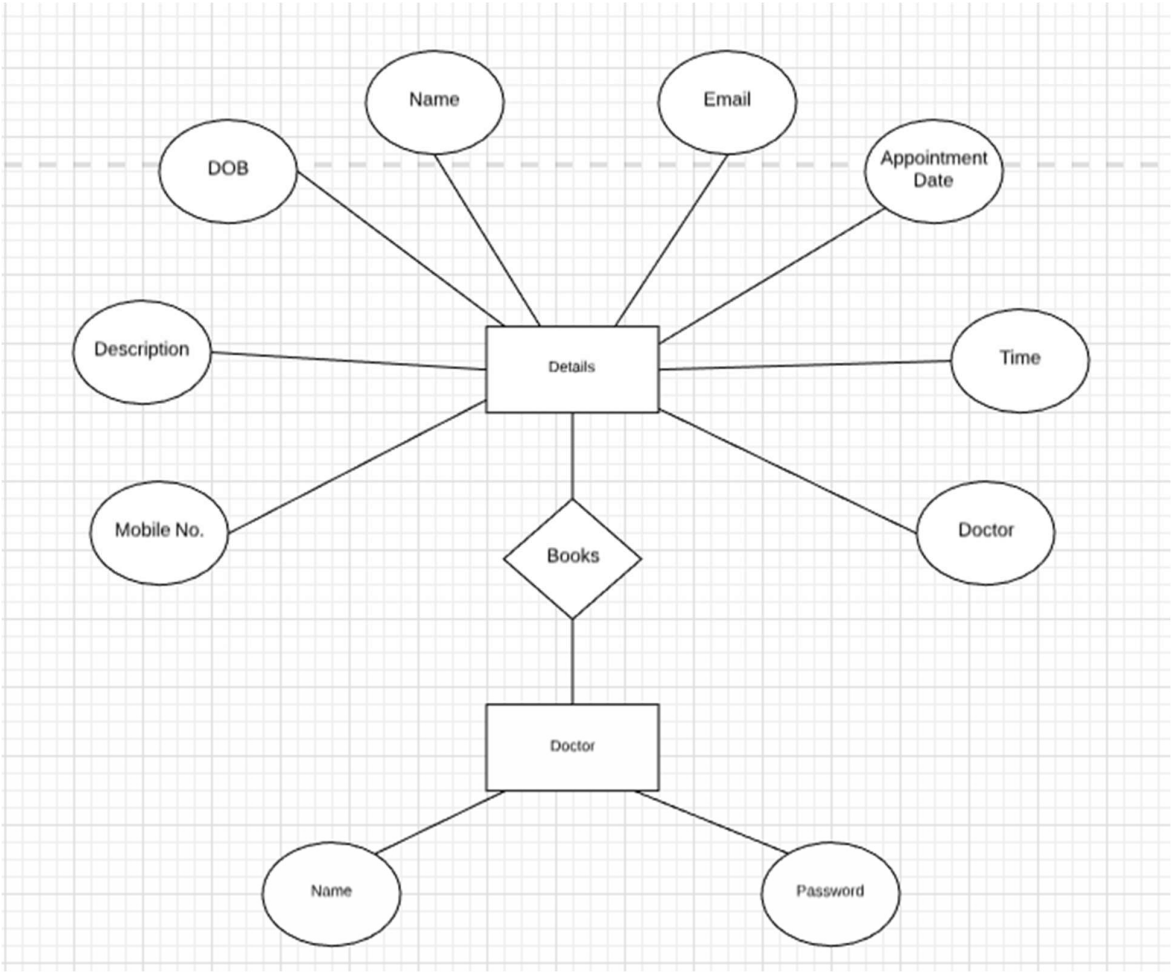

Fig.3.6 E-R Diagram

3.7 GUI Design:

3.7.1 HOMEPAGE:

Its shows the four option of book appointment, view appointment, cancel appointment & doctors log in.

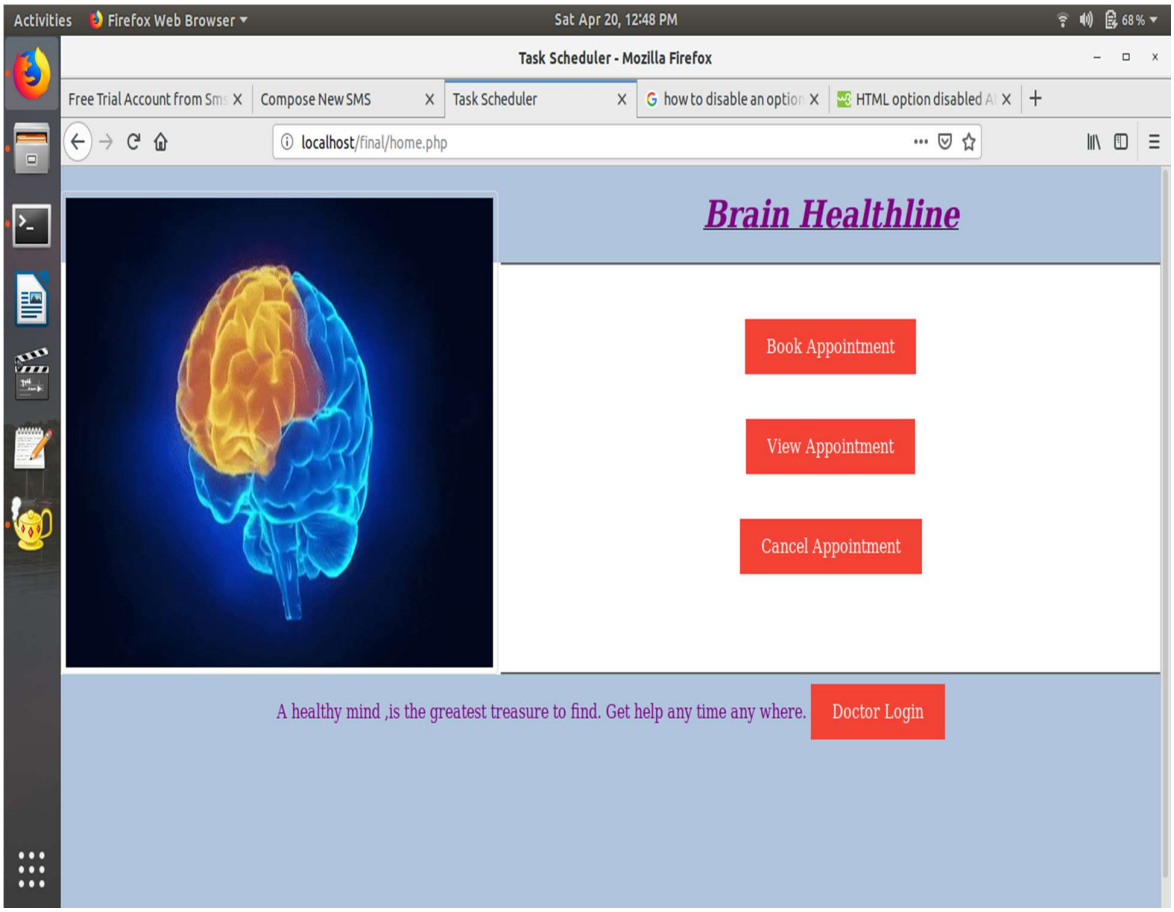

Fig: 3.7 Homepage

3.7.2 APPOINTMENT PAGE:

It is a page where personal details needs to fill to book appointment

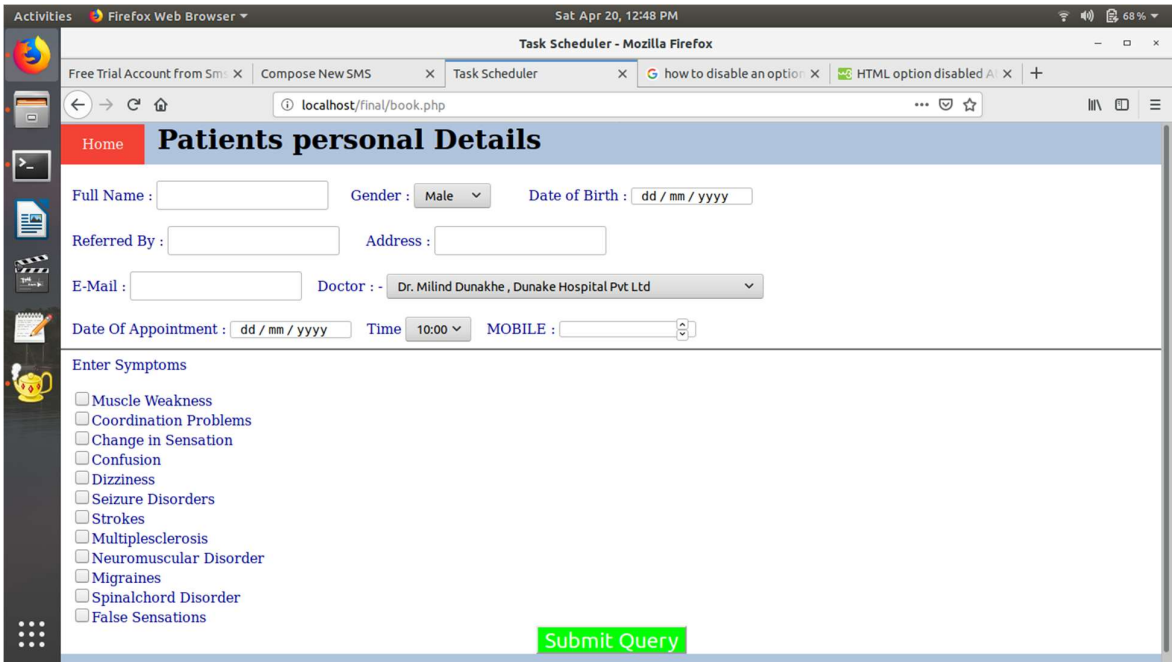

Fig: 3.8 Appointment Page

3.7.3 VIEW PAGE:

Here you can view your appointment details.

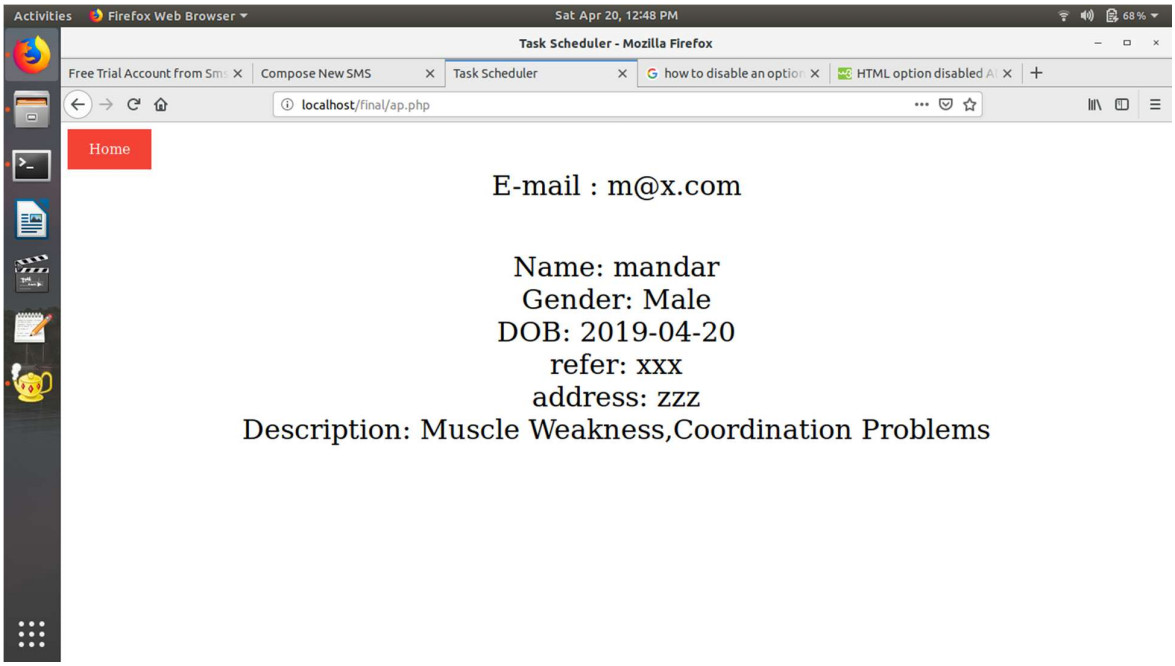

Fig: 3.9 View Page

3.7.4 CANCEL PAGE:

Here you can cancel your appointment

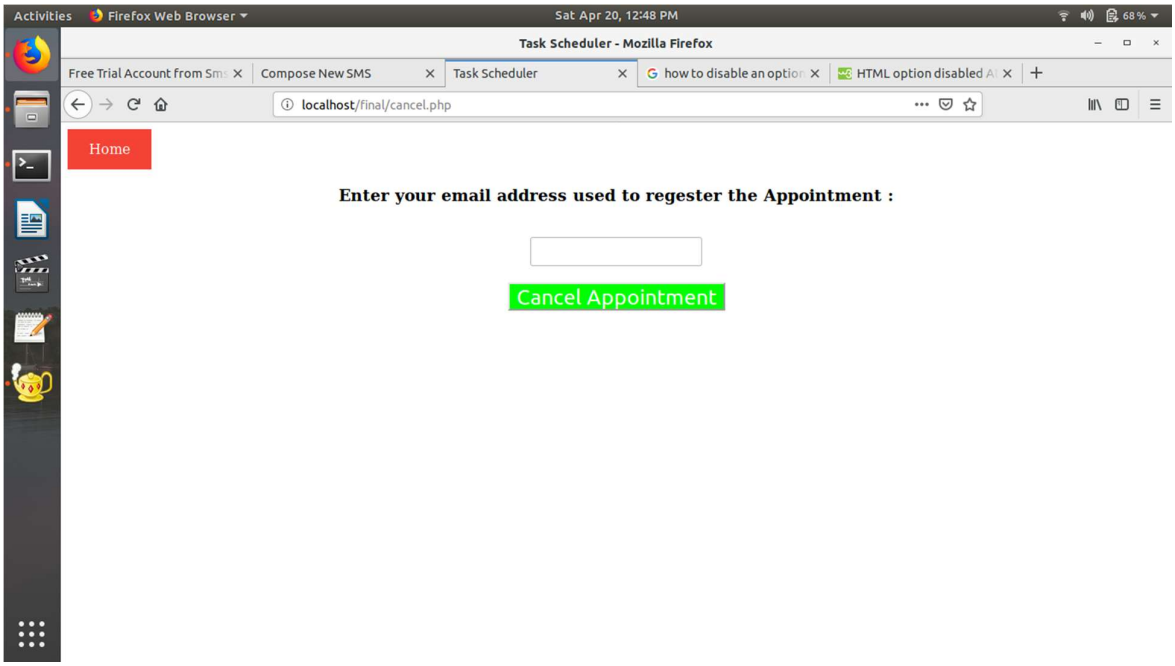

Fig: 3.10 Cancel Page

Chapter 4

Implementation

```
<!--  A simple Neurologist Daily Scheduler Design.It is a simple automation for Appointments -
->
<html>
<head>
        <title> Task Scheduler </title>
</head>
<body bgcolor="#B0C4DE">
        <center><img src="nlogo.gif"   align="left"><u><i><h1 style="color:purple;">Brain
Healthline </h1></u></i>
        <div class="mad">
        <hr><br><br>
        <a href=book.php> Book Appointment </a><br><br><br>
        <a href=manage.php>View Appointment</a><br><br><br>
         <a href=cancel.php>Cancel Appointment</a><br><br><br><br><br>
         <hr>
         </div>
        <font color="purple"> A healthy mind ,is the greatest treasure to find. Get help any time
any where. </font>
        <a href="doctor.php">Doctor Login</a>
</body>
</html>
```

```
 <!--    This  code describes the booking process details of appointment -->
<html>
<head>
        <title> Task Scheduler </title>
</head>
<body bgcolor="#B0C4DE">
                <!--    Take the details of patients -->
```

```
<form  method="post" name="myform" onsubmit="return valid()"
action="result.php">  <a href="home.php"> Home </a>   <font
size="6"><b>Patients personal Details</font></b>
<div class="mad">
<font color="darkblue"> <br>  Full Name : <input type="text" size="15"
name="name" required>    Gender : <select name="gen"> <option>Male</option>
<option>Female</option></select>    
  Date of Birth : <input type="date"  name="dob" size="20" ><br><br>
  Referred By : <input type="text" name="refer" size="15">     
Address : <input type="text" name="add" size="15" required><br><br>
  E-Mail : <input type="text" name="email" size="15" required>   Doctor : -
<select name="doc" id="doc">
<option>Dr. Milind Dunakhe  , Dunake Hospital Pvt Ltd</option>
<option>Dr. Jeevan Rajput ,  JJPlus Hospitals & NEURON International</option>
<option>Dr. Prafull P. Panse  , Kamalnayan Bajaj Hospital</option>
</select>  
<br><br>
  Date Of Appointment : <input type="date"  name="doa" id="doa" size="20">
 
Time <select name="tm">
<option>10:00</option> <option>10:30</option><option>11:00</option>
<option>11:30</option> <option>12:00</option><option>1:00</option>
<option>1:30</option><option>2:00</option>
</select>
 
MOBILE : <input type="number" name="mob" >   
<br>
<hr>
  Enter Symptoms <br><br>
  <input type="checkbox" name="check_list[]" value="Muscle
Weakness">Muscle Weakness <br>
  <input type="checkbox" name="check_list[]" value="Coordination
Problems">Coordination Problems<br>
  <input type="checkbox" name="check_list[]" value="Change in
Sensation">Change in Sensation<br>
```

```
          <input type="checkbox" name="check_list[]"
value="Confusion">Confusion<br>
          <input type="checkbox" name="check_list[]"
value="Dizziness">Dizziness<br>
          <input type="checkbox" name="check_list[]" value="Seizure
Disorders">Seizure Disorders<br>
          <input type="checkbox" name="check_list[]" value="Strokes">Strokes<br>
          <input type="checkbox" name="check_list[]"
value="Multiplesclerosis">Multiplesclerosis<br>
          <input type="checkbox" name="check_list[]" value="Neuromuscular
Disorder">Neuromuscular Disorder<br>
          <input type="checkbox" name="check_list[]"
value="Migraines">Migraines<br>
          <input type="checkbox" name="check_list[]" value="Spinalchord
Disorder">Spinalchord Disorder<br>
          <input type="checkbox" name="check_list[]" value="False Sensations">False
Sensations<br>
       <center><input type="submit" value="submit" style="font-
size:25;color:white;background-color:#00FF00;border:1"></center>
</form>
```

```
        <!--    Apply the validation-->
                <script>
    function valid()
    {
    var name = document.forms["myform"]["name"];
    if (name.value == "")
    {
  window.alert("Please enter name .");
  return false;
    }
    var dob = document.forms["myform"]["dob"];
    if (dob.value == "")
    {
  window.alert("Please enter Date of birth .");
  return false;
```

```
        }
    var add = document.forms["myform"]["add"];
        if (add.value == "")
        {
      window.alert("Please enter address .");
      return false;
        }
    var email = document.forms["myform"]["email"];
        if (email.value == "")
        {
      window.alert("Please enter email .");
      return false;
        }
        var doa = document.forms["myform"]["doa"];
        if (doa.value == "")
        {
      window.alert("Please enter Date of appointment .");
      return false;
        }
var mob = document.forms["myform"]["mob"];
                if (/^\d{10}$/.test(mob.value)) {
        } else {
    alert("Invalid number; must be ten digits");
    return false;
        }
        }
        </script>
</body>
</html>
```

```
 <!-- Connect to database mysql Create database table Dynamically-->
<?php
        $servername = "localhost";
        $username = "root";
        $password = "";
        $conn = new mysqli($servername, $username, $password);
```

```
        if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
        }
        //echo "Connected successfully";
        $sql = "CREATE DATABASE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Database created successfully";
        }
        else{
        //echo "Error creating database: " . $conn->error;
        }
        $sql = "USE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Using successfully";
        }
        else{
        //echo "Not using " . $conn->error;
        }
        $sql =" CREATE TABLE details(srno INT ,name VARCHAR(200) , gender
VARCHAR(100) ,Dob DATE,refer VARCHAR(100), address VARCHAR(100) ,
        Description varchar(1000) , email varchar(1000) , doa VARCHAR(10) , tm
VARCHAR(10) , doctor VARCHAR(100) , mobile VARCHAR(10) )";
        if ($conn->query($sql) === TRUE) {
        //echo "Table created successfully";
        }
        else{
        //echo "Error creating table: " . $conn->error;
        }
?>
```

```
<!-- Take the inputs and Store in database table Use SMS gateway to send Notification and
Detaild of the booked appointment  -->
<html>
<head>
        <title> Task Scheduler </title>
```

```
</head>
<body>
        <a href=home.php> Home </a>
</body>
</html>
<?php
        $servername = "localhost";
        $username = "root";
        $password = "";
        $conn = new mysqli($servername, $username, $password);
        if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
        }
        //echo "Connected successfully";
        $sql = "CREATE DATABASE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Database created successfully";
        }
        else{
        //echo "Error creating database: " . $conn->error;
        }
        $sql = "USE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Using successfully";
        }
        else{
        //echo "Not using " . $conn->error;
        }
        // take the details
        echo"<center><font size =6> Your Appointment Deatils";
        echo"<br>";
        $doap = $_POST["doa"];
        echo"<center><font size =6> Date of Appointment  : $doap </font>";
        $m = $_POST["mob"];
        echo"<center><font size =6> Mobile  : $m </font>";
$doct = $_POST["doc"];
```

```
        echo"<center><font size =6> Doctor  : $doct </font>";
        $t = $_POST["tm"];
        echo"<center><font size =6> Time  : $t </font>";
        $name = $_POST["name"];
        echo"<center><font size =6> Name  : $name </font>";
        echo"<br>";
        $gen = $_POST["gen"];
        echo"<center><font size =6> Gender  : $gen </font>";
        echo"<br>";
        $dob = $_POST["dob"];
        echo"<center><font size =6> Date of Birth  : $dob </font>";
        echo"<br>";
        $refer = $_POST["refer"];
        echo"<center><font size =6> Referred By : $refer </font>";
        echo"<br>";
        $add = $_POST["add"];
        echo"<center><font size =6> Address  : $add </font>";
        echo"<br>";
        $email =  $_POST["email"];
        echo"<center><font size =6> E-mail  : $email </font>";
        echo"<br>";
        echo"<br>";
        $checked_count = count($_POST['check_list']);
        echo "<center><font size =6>You have selected following ".$checked_count." option(s):
<br/>";
        // Loop to store and display values of individual checked checkbox.
        foreach($_POST['check_list'] as $selected) {
        echo "<p> <center><font size =6>".$selected ."</p>";
        }
        $check_list = implode(",",$_POST["check_list"]);
         //validate
        //input data in database
        $sql = "SELECT * FROM details WHERE doa='$doap' AND tm='$t' AND
doctor='$doct'" ;
        $result = $conn->query($sql);
  if ($result->num_rows > 0) {
```

```
                echo "<center><font size =8> Appointment already booked Choose Another
Time or Date or Doctor Your Appointment Is Dismissed";
        }
        else{
        $sql = "INSERT INTO details VALUES(1,'$name' , '$gen' , '$dob' , '$refer' , '$add' ,
'$check_list' , '$email' , '$doap' , '$t' ,'$doct' , '$m' )";
        if ($conn->query($sql) === TRUE) {
        //  echo "New record created successfully";
        }
        else {
        echo "Error: " . $sql . "<br>" . $conn->error;
        }
        echo"<center><font size =6> Your Appointment is Booked";
                // SMS gateway API for PHP
                        // Replace with your username
$user = "swayambhu";
// Replace with your API KEY (We have sent API KEY on activation email, also available on
panel)
$apikey = "iXEaFwuDVHM6xWHtKjSg";
// Replace if you have your own Sender ID, else donot change
$senderid  =  "MYTEXT";
// Replace with the destination mobile Number to which you want to send sms
$mobile  =  "$m";
// Replace with your Message content
$message   =  "Your Appointment is booked $name Appointment Date $doap Time  $t  $doct ";
$message = urlencode($message);
// For Plain Text, use "txt" ; for Unicode symbols or regional Languages like
hindi/tamil/kannada use "uni"
$type   =  "txt";
$ch =
curl_init("http://smshorizon.co.in/api/sendsms.php?user=".$user."&apikey=".$apikey."&mobile
=".$mobile."&senderid=".$senderid."&message=".$message."&type=".$type."");
   curl_setopt($ch, CURLOPT_HEADER, 0);
   curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
   $output = curl_exec($ch);
   curl_close($ch);  // Display MSGID of the successful sms push
```

```
echo $output;
}
?>
```

```
<!-- View Appointment -->
<html>
<head>
        <title> Task Scheduler </title>
</head>
<body >
        <a href=home.php> Home </a>
        <form method="post" name="myform" onsubmit="return valid()" action="ap.php" >
        <center>
        <h3>Enter your email address used to regester the Appointment :</h3><br> <input
type="text" name="mail" size="15"><br><br>

        <input type = "submit" value="View Appointment" style="font-
size:25;color:white;background-color:#00FF00;border:1">
        </form>
        <script>
        function valid()
        {
        var email = document.forms["myform"]["mail"];
        if (email.value == "")
        {
      window.alert("Please enter email .");
      return false;
        }
        }
        </script>
</body>
</html>
<?php
        $servername = "localhost";
        $username = "root";
        $password = "";
```

```
        $conn = new mysqli($servername, $username, $password);
        if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
        }
        //echo "Connected successfully";
        $sql = "CREATE DATABASE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Database created successfully";
        }
        else{
        //echo "Error creating database: " . $conn->error;
        }
        $sql = "USE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Using successfully";
        }
        else{
        //echo "Not using " . $conn->error;
        }
?>
```

```
 <!-- Validate and Show Details of appointment -->
<html>
<head>
        <title> Task Scheduler </title>
</head>
<body>
        <a href=home.php> Home </a>
</body>
</html>
<?php
        $servername = "localhost";
        $username = "root";
        $password = "";
        $conn = new mysqli($servername, $username, $password);
        if ($conn->connect_error) {
```

```
        die("Connection failed: " . $conn->connect_error);
        }
        //echo "Connected successfully";
        $sql = "CREATE DATABASE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Database created successfully";
        }
        else{
        //echo "Error creating database: " . $conn->error;
        }
        $sql = "USE Appointment";
        if ($conn->query($sql) === TRUE) {
         //echo "Using successfully";
        }
        else{
        //echo "Not using " . $conn->error;
        }
        $mail =  $_POST["mail"];
        echo"<center><font size =6> E-mail  : $mail </font>";
        $sql = "SELECT * FROM details WHERE email='$mail'";
        $result = $conn->query($sql);
        if ($result->num_rows > 0) {
// output data of each row
         while($row = $result->fetch_assoc()) {
         echo"<br><br><br><br>";
      echo  "<center><font size =6> Name: " . $row["name"] .   "<br>";
      echo  "<center><font size =6> Gender: " . $row["gender"] .  "<br>";
      echo  "<center><font size =6> DOB: " . $row["Dob"] .  "<br>";
       echo  "<center><font size =6> refer: " . $row["refer"] .  "<br>";
       echo  "<center><font size =6> address: " . $row["address"] .  "<br>";
      echo  "<center><font size =6> Description: " . $row["Description"] .  "<br>";
        }
        }
        else {
        echo "<center><font size =6> Entered email dosen't exist";
        }
```

```
        ?>
```

```
<!-- Page for Deleting Appointment -->
<html>
<head>
        <title> Task Scheduler </title>
</head>
<body>
        <a href=home.php> Home </a>
        <form method="post" name="myform" onsubmit="return valid()" action="apd.php" >
        <center>
        <h3>Enter your email address used to regester the Appointment :</h3><br> <input
type="text" name="mail" id="e" size="15"><br><br>
        <input type = "submit" value="Cancel Appointment" style="font-
size:25;color:white;background-color:#00FF00;border:1">
        </form>
        <script>
        function valid()
        {
        var email = document.forms["myform"]["mail"];
        if (email.value == "")
        {
      window.alert("Please enter email .");
      return false;
        }
        }
        </script>
</body>
</html>
<?php
        $servername = "localhost";
        $username = "root";
        $password = "";
        $conn = new mysqli($servername, $username, $password);
        if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
```

```
        }
        //echo "Connected successfully";
        $sql = "CREATE DATABASE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Database created successfully";
        }
        else{
        //echo "Error creating database: " . $conn->error;
        }
        $sql = "USE Appointment";
        if ($conn->query($sql) === TRUE) {
        //echo "Using successfully";
        }
        else{
        //echo "Not using " . $conn->error;
        }
?>
```

```
<!-- Page for Deleting Appointment -->
<html>
<head>
        <title> Task Scheduler </title>
</head>
<body>
        <a href=home.php> Home </a>
</body>
</html>
<?php
        $servername = "localhost";
        $username = "root";
        $password = "";
        conn = new mysqli($servername, $username, $password);
        if ($conn->connect_error) {
        die("Connection failed: " . $conn->connect_error);
        }
        //echo "Connected successfully";
```

```
$sql = "CREATE DATABASE Appointment";
if ($conn->query($sql) === TRUE) {
//echo "Database created successfully";
}
else{
//echo "Error creating database: " . $conn->error;
}
$sql = "USE Appointment";
if ($conn->query($sql) === TRUE) {
//echo "Using successfully";
}
else{
//echo "Not using " . $conn->error;
}
$mail =  $_POST["mail"];
echo"<br><br><font size =4> E-mail  : $mail </font>";
$sql = "DELETE  FROM details WHERE email='$mail'";
$result = $conn->query($sql);
echo "<center><font size =6> Record deleted successfully Appointment is Cancled
</font>";
?>
```

```
<!-- Doctor Login Page -->
<html>
<head>
<title> Task Scheduler </title>
</head>
<body>
<a href="home.php"> Home </a>
<form method="post" action="login.php" >
<font color="darkblue"> <br>
  Doctor : - <select name="log">
<option>Dr. Milind Dunakhe  , Dunake Hospital Pvt Ltd </option>
<option>Dr. Jeevan Rajput ,  JJPlus Hospitals & NEURON International</option>
<option>Dr. Prafull P. Panse  , Kamalnayan Bajaj Hospital</option>
</select>    <br><br>
```

```
   Password : <input type="password" size="15" name="pass" required>
<center><input type="submit" value="submit" style="font-
size:25;color:white;background-color:#00FF00;border:1"></center>
</form>
</body>
</html>
<?php
$servername = "localhost";
$username = "root";
$password = "";
$conn = new mysqli($servername, $username, $password);
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}
//echo "Connected successfully";
$sql = "CREATE DATABASE Appointment";
if ($conn->query($sql) === TRUE) {
//echo "Database created successfully";
}
else{
//echo "Error creating database: " . $conn->error;
}
$sql = "USE Appointment";
if ($conn->query($sql) === TRUE) {
//echo "Using successfully";
}
else{
//echo "Not using " . $conn->error;
}
?>
```

```
<!-- Validate Doctor login and Display appointment for each  -->
<?php
$servername = "localhost";
$username = "root";
$password = "";
```

```
	$conn = new mysqli($servername, $username, $password);
	if ($conn->connect_error) {
	die("Connection failed: " . $conn->connect_error);
	}
	//echo "Connected successfully";
	$sql = "CREATE DATABASE Appointment";
	if ($conn->query($sql) === TRUE) {
	//echo "Database created successfully";
	}
	else{
	//echo "Error creating database: " . $conn->error;
	}
	$sql = "USE Appointment";
	if ($conn->query($sql) === TRUE) {
	//echo "Using successfully";
	}
	else{
	//echo "Not using " . $conn->error;
	}
?>
<html>
<head>
	<title> Task Scheduler </title>
</head>
<body>
<a href="home.php"> Home </a>
<?php
	// take the details
	echo"<center><font size =6> Your Appointment Deatils";
	echo"<br>";
	$l = $_POST["log"];
	echo"$l";
		$p = $_POST["pass"];
	//echo"$p";
	$sql ="SELECT * FROM doctor WHERE name='Dr. Milind Dunakhe  , Dunake
Hospital Pvt Ltd' AND password='$p'";
```

```
        $result = $conn->query($sql)
        if ($result->num_rows > 0) {
   // output data of each row
        echo"<center><font size =6> Appointments";
?>
<center>
        <br>
        <font size="5"><a href="re.php">re-schedule</a></font>
        <br><br>
        <table border=2 >
        <tr>
                <th align="center">name</th>
                <th width="12" align="center">Appointment date</th><th align="center">
Discription</th><th width="12" align="center">Time
                </th>
                <th align="center">refer</th>
        </tr>
<?php
                $sql =" SELECT * FROM details WHERE doctor='Dr. Milind Dunakhe ,
Dunake Hospital Pvt Ltd';";
                $result = $conn->query($sql);
                if ($result->num_rows > 0) {
                        while($row = $result->fetch_assoc()) {
 ?>
 <tr width="10">
                <td align="center"><?php echo  $row['name']; ?></td>
                <td align="center"><?php echo  $row['doa'];?></td>
                <td align="justify"><?php echo  $row['Description']; ?></td>
                <td align="center"><?php echo  $row['tm']; ?></td>
                <td align="center"><?php echo  $row['refer']; ?></td>
</tr>
<?php
                }
                }
                else {
        echo "<center><font size =6> Entered record dosen't exist";
```

```
        }
        }
        else {
        //echo "<center><font size =6> Incorrect password";
        }
        $sql = "SELECT * FROM doctor WHERE name='Dr. Jeevan Rajput ,  JJPlus Hospitals
& NEURON International' AND password='$p';";
        $result = $conn->query($sql);
        if ($result->num_rows > 0) {
   // output data of each row
        echo"<center><font size =6> Appointments";
        $sql =" SELECT * FROM details WHERE doctor='Dr. Jeevan Rajput , JJPlus Hospitals
& NEURON International';";
                $result = $conn->query($sql);
                if ($result->num_rows > 0) {
?>
<center>
        <br>
        <font size="5"><a href="re1.php">re-schedule</a></font>
        <br><br>
        <table border=2 >
        <tr>
                <th align="center">name</th>
                <th width="12" align="center">Appointment date</th><th align="center">
Discription</th><th width="12" align="center">Time
                </th>
                <th align="center">refer</th>
        </tr>
<?php
                        while($row = $result->fetch_assoc()) {
?>
  <tr width="10">
                <td align="center"><?php echo  $row['name']; ?></td>
                <td align="center"><?php echo  $row['doa'];?></td>
                <td align="justify"><?php echo  $row['Description']; ?></td>
                <td align="center"><?php echo  $row['tm']; ?></td>
```

```
                <td align="center"><?php echo  $row['refer']; ?></td>
</tr>
<?php
                }
                }
                else {
        echo "<center><font size =6> Entered record dosen't exist";
        }
        }
        else {
        //echo "<center><font size =6> Incorrect Password";
        }
        $sql = "SELECT * FROM doctor WHERE name='Dr. Prafull P. Panse  , Kamalnayan
Bajaj Hospital' AND password='$p'";
        $result = $conn->query($sql);
        if ($result->num_rows > 0) {
   // output data of each row
        echo"<center><font size =6> Appointments";
        $sql ="SELECT * FROM details WHERE doctor='Dr. Prafull P. Panse , Kamalnayan
Bajaj Hospital          ';";
                $result = $conn->query($sql);
                if ($result->num_rows > 0) {
?>
<center>
        <br>
        <font size="5"><a href="re2.php">re-schedule</a></font>
        <br><br>
        <table border=2 >
        <tr>
                <th align="center">name</th>
                <th width="12" align="center">Appointment date</th><th align="center">
Discription</th><th width="12" align="center">Time
                </th>
                <th align="center">refer</th>
        </tr>
<?php
```

```
                        while($row = $result->fetch_assoc()) {
?>
 <tr width="10">
                <td align="center"><?php echo  $row['name']; ?></td>
                <td align="center"><?php echo  $row['doa'];?></td>
                <td align="justify"><?php echo  $row['Description']; ?></td>
                <td align="center"><?php echo  $row['tm']; ?></td>
                <td align="center"><?php echo  $row['refer']; ?></td>
</tr>
<?php
        }
                }
                else {
        echo "<center><font size =6> Entered record dosen't exist";
        }
        }
        else {
        //echo "<center><font size =6> Incorrect password";
        }
?>
</body>
</html>
```

```
<!-- Rescheduling for doctor Dr. Milind Dunakhe , Dunake Hospital Pvt Ltd
-->
<html>
<head>
        <title> Task Scheduler </title>
</head>
<body>
<a href="home.php"> Home </a>
</body>
</head>
<?php
        $servername = "localhost";
        $username = "root";
```

```
$password = "";
$conn = new mysqli($servername, $username, $password);
if ($conn->connect_error) {
die("Connection failed: " . $conn->connect_error);
}
//echo "Connected successfully";
$sql = "CREATE DATABASE Appointment";
if ($conn->query($sql) === TRUE) {
//echo "Database created successfully";
}
else{
//echo "Error creating database: " . $conn->error;
}
$sql = "USE Appointment";
if ($conn->query($sql) === TRUE) {
//echo "Using successfully";
}
else{
//echo "Not using " . $conn->error;
}
$sql = "SELECT mobile FROM details WHERE doctor='Dr. Milind Dunakhe , Dunake Hospital Pvt Ltd';";
$result = $conn->query($sql);
if ($result->num_rows > 0) {
                while($row = $result->fetch_assoc()) {
                        echo  "<center><font size =6> mobile: " . $row["mobile"] . "<br>";
                        //$m = mysql_query("SELECT mobile FROM details WHERE doctor='Dr. Milind Dunakhe , Dunake Hospital Pvt Ltd';");
                        //$mm = mysql_fetch_array($m);
                        }
        }
        else {
echo "<center><font size =6> Entered record dosen't exist";
}
// Replace with your username
```

```
$user = "swayambhu";
// Replace with your API KEY (We have sent API KEY on activation email, also available on
panel)
$apikey = "iXEaFwuDVHM6xWHtKjSg";
// Replace if you have your own Sender ID, else donot change
$senderid  =  "MYTEXT";
// Replace with the destination mobile Number to which you want to send sms
$mobile  =  "$mm";
//echo "$m";
// Replace with your Message content
$message   =  "Your Appointment is booked $name Appointment Date $doap Time  $t  $doct ";
$message = urlencode($message);
// For Plain Text, use "txt" ; for Unicode symbols or regional Languages like
hindi/tamil/kannada use "uni"
$type   =  "txt";
$ch =
curl_init("http://smshorizon.co.in/api/sendsms.php?user=".$user."&apikey=".$apikey."&mobile
=".$mobile."&senderid=".$senderid."&message=".$message."&type=".$type."");
   curl_setopt($ch, CURLOPT_HEADER, 0);
   curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
   $output = curl_exec($ch);
   curl_close($ch);
// Display MSGID of the successful sms push
//echo $output;
?>
```

Chapter 5

Testing

5.1 Introduction to Testing:

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test.[1] Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

Software testing can also be stated as the process of validating and verifying that a software program/application/product:

1. meets the business and technical requirements that guided its design and development;
2. Works as expected; and
3. Can be implemented with the same characteristics.

5.2 Verification:

Verification is the process of evaluating work-products of a development phase to determine whether they meet the specified requirements.

Verification ensures that the product is built according to the requirements and design specifications. It also answers to the question, Are we building the product right?

5.3 Validation:

Validation Testing ensures that the product actually meets the client's needs. It can also be defined as to demonstrate that the product fulfills its intended use when deployed on appropriate environment.

It answers to the question, Are we building the right product?

5.4 System Testing:

5.4.1 Unit Testing:

Unit testing is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed. A unit is the smallest testable part of any software. It usually has one or a few inputs and usually a single output. In procedural programming, a unit may be an individual program, function, procedure, etc. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class. (Some treat a module of an application as a unit. This is to be discouraged as there will probably be many individual units within that module.) Unit testing frameworks, drivers, stubs, and mock/ fake objects are used to assist in unit testing.

5.4.2 Test Cases:

Table: 5.1

Project Name	Neurologist's Daily Appointment Scheduler Design					
Module Name	Login					
Test Scenario ID	Test Scenario Description	Test Case Description	Test Steps	Expected Result	Actual Result	Status
TS_1	Verify the Login Functionality of Doctor Login Page	Enter a valid username and password	1.Enter Valid Username 2.Enter Valid Password 3.Click on Login Button	Successful Login	Login successfully	Pass
TS_2	Verify the Login Functionality of Doctor Login Page	Enter a valid username and invalid password	1.Enter Valid Username 2.Enter Invalid Password 3.Click on Login Button	A pop up message box to show an error"invalid username/password"	Login unsuccessfully	Fail
TS_3	Verify the Login Functionality of Doctor Login Page	Enter a invalid username and password	1.Enter Invalid Username 2.Enter Valid Password 3.Click on Login Button	A pop up message box to show an error"invalid username/password"	Login unsuccessfully	Fail

TS_4	Verify the Login Functionality of Doctor Login Page	Enter a invalid username and invalid password	1.Enter Invalid Username 2.Enter Invalid Password 3.Click on Login Button	A pop up message box to show an error" invalid username/password"	Login unsuccessfully	Fail

Table: 5.2

Project Name	Neurologist's Daily Appointment Scheduler Design					
Module Name	Home Page					
TS_5	Verify the Home Page	Click on "Book Appointment"	Step-1 Click on "Book Appointment" Button	"Patients all Details Form" should be validate.	"Patients all Details form" is validated.	Pass
TS_6	Verify the Home Page	Click on "Book Appointment"	Step-1 Fill all Details. Step-2Click on "Submit" Button	"Patients all Details Form" should be validate.	"Patients all Details form" is not validated.	Fail
TS_7	Verify "View Appointment" page	Click on "View Appointment"	Step-1 Click on "View Appointment" Button	Appointment Details should open.	Appointment Details is Displayed.	Pass
TS_8	Verify "View Appointment" page	Click on "View Appointment"	Step-1 Click on "View Appointment" Button	Appointment Details should open.	Appointment Details is not Displayed.	Fail
TS_9	Verify "Cancel Appointment" page	Click on "Cancel Appointment"	Step-1 Click on "Cancel Appointment" Button Step-2 Enter correct Email ID which is entered at the time of Booking the Appointment.	Appointment should be cancelled and again the user should not possible to view.	Appointment is Cancelled.	Pass

TS_10	Verify "Cancel Appointment" page	Click on "Cancel Appointment"	Step-1 Click on "Cancel Appointment" Button Step-2 Enter wrong Email ID which is entered at the time of Booking the Appointment.	Appointment should be cancelled and again the user should not possible to view.	Appointment is not cancelled and display popup message i.e. Enter correct Email ID.	Fail

Chapter 6

Deployment Observation

Observations:

Login successfully only when credentials are correct.

Messages are delivered only when email is valid.

Only valid information will be stored as per validations.

No two appointments should be assigned on same time and date.

Conclusion

By all requirement gathering and the investigation of the applicability, conclusion of this appointment scheduler enables you to automatically perform routine appointments of doctors on chosen computer. The appointment scheduler does this by monitoring whatever criteria you choose to initiate the appointments and executing the appointments when the criteria is met.

Neurologist Daily appointment Scheduler can be used to execute appointments such as starting an application, sending an email message, or showing a message box. Appointments can be scheduled to execute:

- When a serious seizures occurs to a patient.
- At a specific time.
- At a specific time on a daily schedule.
- At a specific time on a weekly schedule.
- At a specific time on a monthly schedule.
- At a specific time on a monthly day-of-week schedule.
- When the system enters an idle state.
- When the appointment is registered.
- When the system is booted.
- When a user logs on.

REFERENCES

[1] www.microsoft.com

[2] www.appointmentmanagementsoft.com

[3] www.windowsreport.com.

[4] www.wikipedia.com

[5] https://zapier.com/blog/best-appointment-scheduling-apps/

[6] https://en.wikipedia.org/wiki/Appointment_scheduling_software

[7]https://www.smshorizon.in/?gclid=Cj0KCQjwov3nBRDFARIsANgsdoE33Pg3QTu9URz0JGK7DIc7eO4hJyEJF31Ehq4sTv8D6avg-zcvNXwaAknEEALw_wcB