

A
Project Report On
HOSPITAL MANAGEMENT SYSTEM

Submitted to
University of Mumbai

By
Mr. Aniket Sunil Dhulap

T.Y.B.Sc. (Computer Science)

Under Guidance Of
Miss. Dhanashri Shinde

Year Duration
2019-2020

Throught



DEPARTMENT OF COMPUTER SCIENCE
S. P. HEGSHETYE COLLEGE OF ARTS, COMMERCE AND
SCIENCE, RATNAGIRI.

(Affiliated to University of Mumbai)

RATNAGIRI, 415612

**S. P. HEGSHETYE COLLEGE OF ARTS, COMMERCE
AND SCIENCE, RATNAGIRI.**

(Affiliated to University of Mumbai)

**RATNAGIRI
MAHARASHTRA**

415612

BACHELOR OF COMPUTER SCIENCE



CERTIFICATE

This is to certify that **MR. ANIKET SUNIL DHULAP** bearing Examination Seat No _____ has successfully completed the project titled '**HOSPITAL MANAGEMENT SYSTEM**' for academic year 2019 - 2020 , in partial fulfilment of B.Sc. Degree in Computer Science as prescribed by the University of Mumbai.

Date :-

Place :-

Signature :-

Project Guide

HOD

Examiner

Principal

Acknowledgement

I would like to thank to “**Navnirman Shikshan Society**” who gave me the information of the entire workout of their present system.

I would especially thank **Miss.Dhanashri Shinde** and all teaching and non-teaching staff of Computer Science faculty who inspiring me in completion of project.

I am thankful to **Prof. Miss. Ruhina Mayer** madam Head of Department of computer science department for their timely help and guidance in completing this project successfully.

I would also like to express our deep regards and gratitude to our Principal **Mr. Sawant** sir for his support and **Mr. Bhagwan Mukadam** sir for facilities provided to us for the same.

Lastly, I would like to thank all those who directly and indirectly helped in completion of this project.

Thanking You

Mr.Aniket Sunil Dhulap.

**S. P. HEGSHETYE COLLEGE OF ARTS, COMMERCE
AND SCIENCE, RATNAGIRI.**

MAHARASHTRA

415612



DECLARATION

I here by declare that the project entitled, “**Hospital Management System**” done at **place where the project is done**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (COMPUTER SCIENCE)** to be submitted as final semester project as part of our curriculum.

Mr.Aniket Sunil Dhulap

Plagiarism Report



PLAGIARISM SCAN REPORT

Words	92	Date	October 04,2019
Characters	608	Exclude Url	

0%

Plagiarism

100%

Unique

0Plagiarized
Sentences**3**

Unique Sentences

Content Checked For Plagiarism

The Hospital Management System Project application provide the features like store the patient registration ,Admin detail,doctor details,nurse details,and make appointment in that.there is no need paper work,admin can handle the all system .they can manage doctors data ,they can manage patient data,also they can handle the payment and medicine in system.in that software doctor can see the appointment list,manage patient details.nurse can handle the patient and medicine data,if you are login from admin then you need a username or passsword also doctor and nurse have their login system in that software

Sources

Similarity

INDEX

Sr.No	Topics
1	Title-Introduction
2	Abstract
3	Requirement Analysis
4	Hardware Requirement
5	Software Requirement
6	System Analysis
7	Existing System
8	Proposed System
9	Feasibility Study
10	Objective
11	Drawback
12	Gantt Chart
13	System Design Detail
14	Use Case Diagram
15	E-R Diagram
16	System Implementation
17	Result
18	Tables
19	Screenshots
20	Conclusion

Title

“Hospital Management System”

Introduction:-

- Hospital Management System is offline management system which is used to management works in the hospital, I am explaining various functionality of hospital management system
- The project Hospital Management system includes registration of patients, storing their details into the system .The software has the facility to give a unique id for every patient and stores the details of every patient and the staff .
- The Hospital Management System can be entered using a username and password. It is accessible either by an administrator , doctor and nurse . Only they can add data into the database and user make appointment , see report and see doctors availability .The data can be retrieved easily. The interface is very user-friendly. The data are well protected for personal use and makes the data processing very fast.
- Hospital Management System is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to hospitals.
- It will process data speedily and accurately provide information when and where required.
- The Hospital Management System will be used to store data, produce reports and handle management inquiries.
- Therefore, it is hoped that the Hospital Management Information system would provide a flexible and reliable management for the Hospitality works. This program is carefully designed to work in a standalone system or in a multi user environment.

Modules:-

- Admin module
- User module
- Doctor module
- Nurse module

➤ Admin module:-

- manage department of hospitals, user, doctor, nurse, payments, medicines.
- watch appointment of doctors.
- watch transaction reports of patient payment.
- It is the most powerful user of the system.
- The admin can create and manage all others modules. Manage all the hospitals works.
- Admin can add the details (profile) of doctor and nurse.
- Manage patients informations.
- Manage medicines and payment .
- See report of Patient Data and Appointments.
-

➤ User module:-

- View appointment list and status with doctors.
- View doctor list.
- Register Appointment.

➤ Doctor module:-

- Manage patient. account opening and updating
- Create, manage appointment with patient
- Manage own profile

➤ Nurse module:-

- Nurse is an assistant of the doctor. The nurse is helps the doctors in performing the operations. She takes the diagnostic report and provide to the doctor.
- Manage patient. account opening and updating
- Manage own profile
- Manage export Medicines.

Abstract

- The purpose of the project entitled as “ **HOSPITAL MANAGEMENT SYSTEM** ” is to computerize the Front Office Management of Hospital to develop software which is user friendly simple, fast, and cost – effective. It deals with the collection of patient’s information etc. Traditionally, it was done manually.
- The main function of the system is register and store patient details and doctor details and retrieve these details as and when required, and also to manipulate these details meaningfully System input contains patient details, while system output is to get these details on to the screen.

Requirement Analysis

➤ Software Requirements

- Software :- Visual Studio 2010 - Visual Studio 2012
- Front End :- ASP.NET
- Back End :- Microsoft SQL server 2008
- Language :- C#
- Operating System :- Windows 7 to Windows 10

➤ Hardware Requirements

- RAM :- 2GB RAM
 - Processor :- i3
 - Hard Disk :- 50 GB
- **The above specified requirements are the minimum required to run the application**

System Analysis

➤ Existing System:-

- Hospitals currently use a manual system for the management and maintainance of critical information. The current system requires numerous paper forms, with data stores spread through out the hospital management infrastructure. Often information is incomplete or does not follow management standards. Forms are often lost in transit between departments requiring a comprehensive auditing process to ensure that no vital information is lost. Multiple copies of the same information exist in the hospital and may lead to inconsistencies in data in various data stores.

➤ Proposed System:

- The Hospital Management System is designed for any hospital to replace their existing manual paper based system. The new system is to control the information of patients. Room availability, staff and operating room schedules and patient invoices. These services are to be provided in an efficient, cost effective manner, with the goal of reducing the time and resources currently required for such tasks .
- The proposed system has enhanced features, which was not found inthe existing system. The salient features are
- Security for the data is done easily.
- Validation is done to enter correct data.
- Memory consumption is very less and the processing speedis fast.
- Data reports are presented in a neat format.

➤ Feasibility Study:

- The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Objective

Hospitals currently use a manual system for the management and maintenance of critical information. The current system requires numerous paper forms, with data stores spread throughout the hospital management infrastructure. Often information (on forms) is incomplete, or does not follow management standards. Forms are often lost in transit between departments requiring a comprehensive auditing process to ensure that no vital information is lost. Multiple copies of the same information exist in the hospital and may lead to inconsistencies in data in various data stores.

- **Computerization** –
 - All the details regarding hospitals, whether it is small or big, will be computerised.
- **Automated inventory** –
 - If the medicines are provided to the patients keep records of that.
- **No redundancy** –
 - For every test that is conducted of the patients, an automated report will be generated and will be available to the patients and his / her concerned doctor uniformly.
- **Keep the Records** –
 - It will be easier task for the management to keep the record of the patients for historical purpose.
- **Appointment** –
 - It will be easier for both the doctors and the patients to have the appointments. It is just two clicks away.

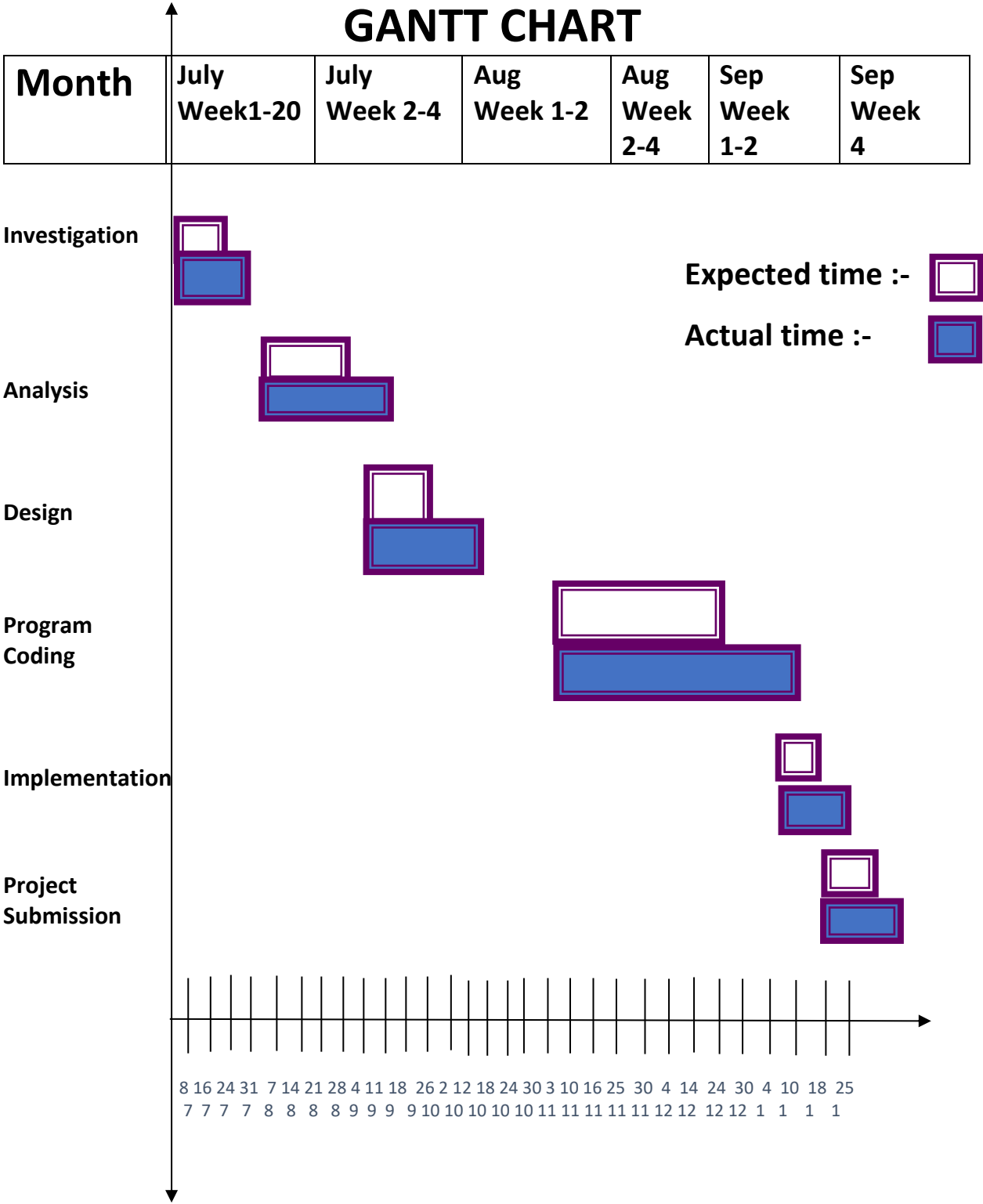
Key Features

- Multi user account system
- Monitoring the whole hospital system
- Management of all type of users' account
- Appointment Management
- View Appointments
- Medicine History
- Payments Management
- Appointment and patient Registration Report Management
- Internal Communication
- Responsive User Interfaces

Drawbacks

- The existing system is manual system.
- Needs to be converted into automated system.
- Risk of mismanagement of data.
- Less security.
- No proper coordination between different Applications and users.
- Accuracy not guaranteed.
- Reports are generated manually.
- Take lot of time.

Gantt Chart



System Design

➤ Reception:

The reception module handles various enquiries about the patient's admission and discharge details, and the patient's movements within the hospital. The system can also handle fixed-cost package deals for patients as well as Doctor Consultation and Scheduling, Doctor Consultancy Fees and Time Allocation.

- Doctor visit schedule
- Doctor Appointment Scheduling
- Enquiry of Patient
- Find History of Patient Enquired.
- View Reoprt
- Make Appointment

➤ Administration:

This module handles all the master entry details for the hospital requirement such as consultation detail, doctor specialization, consultancy fee, and service charges.

Employee

- All controls of system
- Doctor and nurse profile management
- Payment management
- Referral Doctor

➤ Registration:

This module helps in registering information about patients and handling both IPD and OPD patient's query. A unique ID is generated for each patient after registration. This helps in implementing customer relationship management and also maintains medical history of the patient.

Design

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

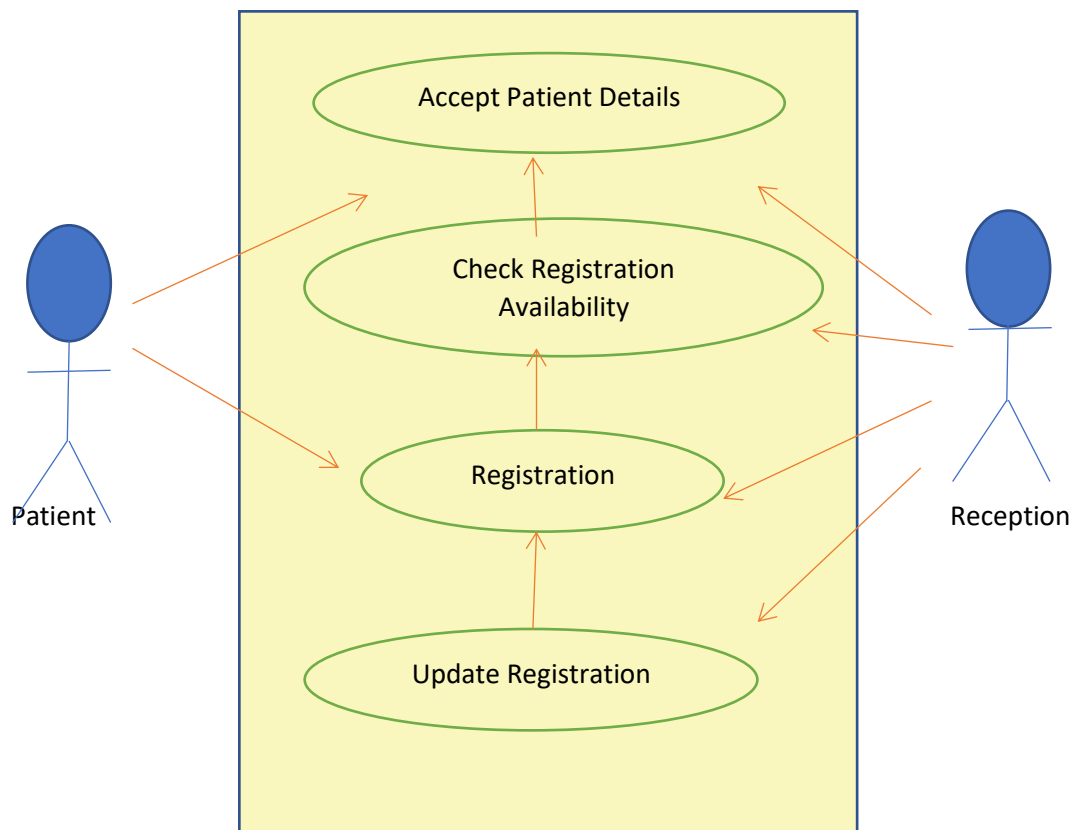
Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

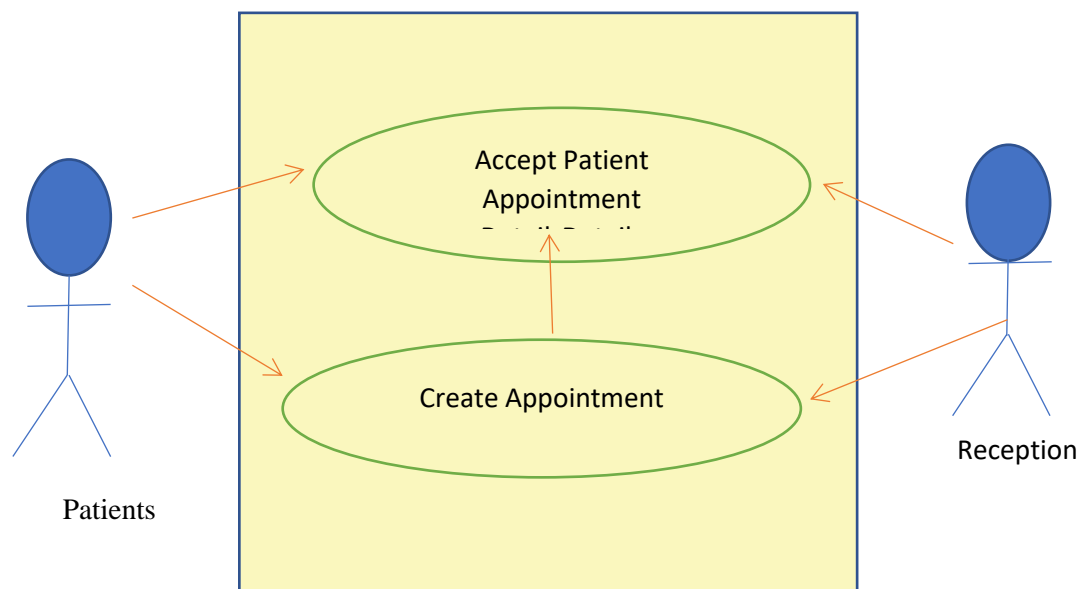
Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

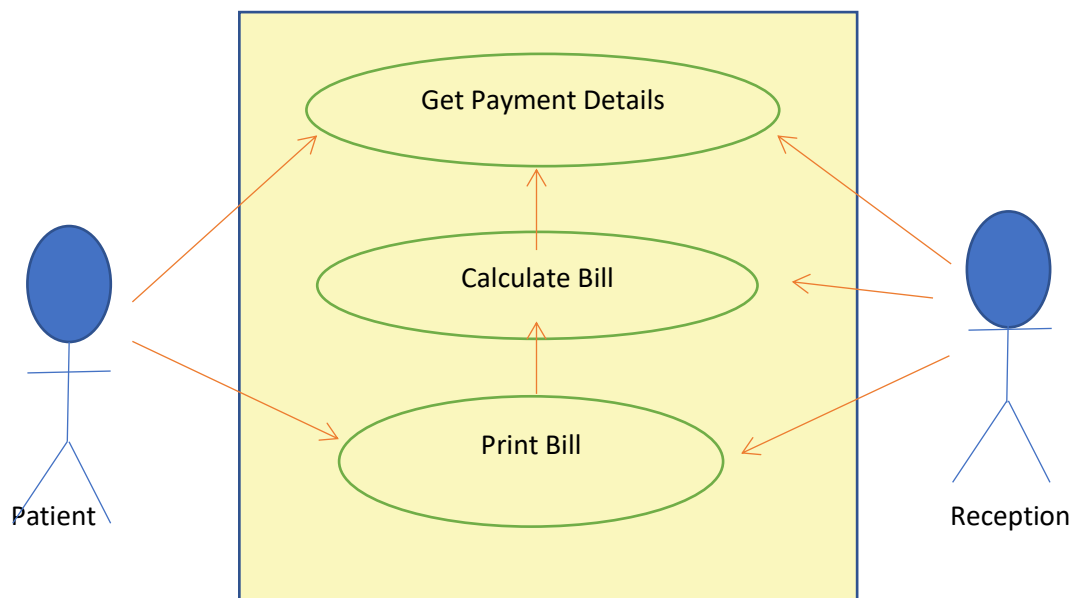
Use Case Diagram

Patient Registration:-

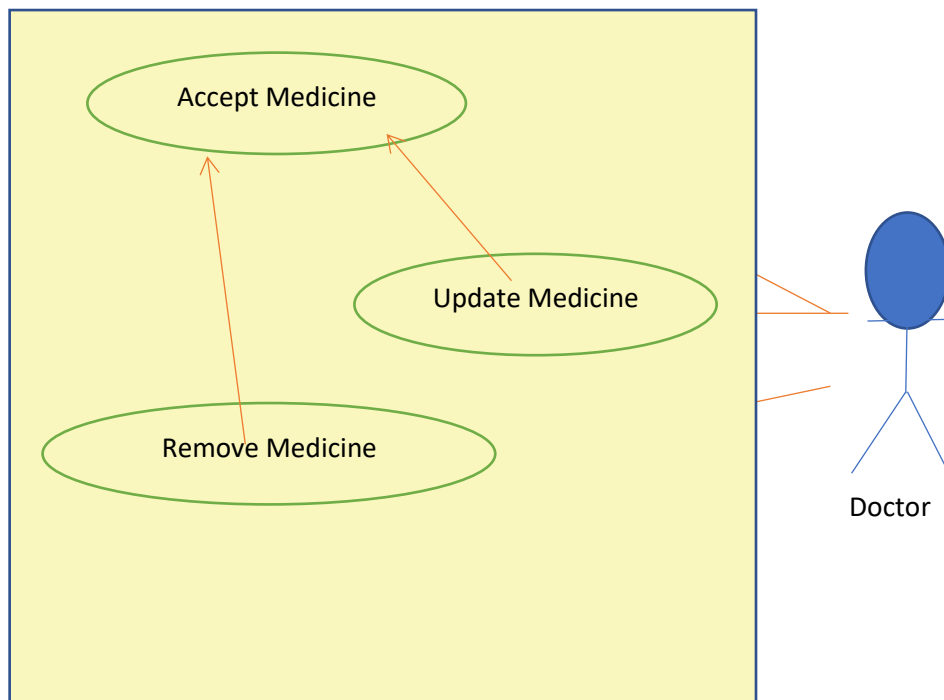


Patient Appointment:

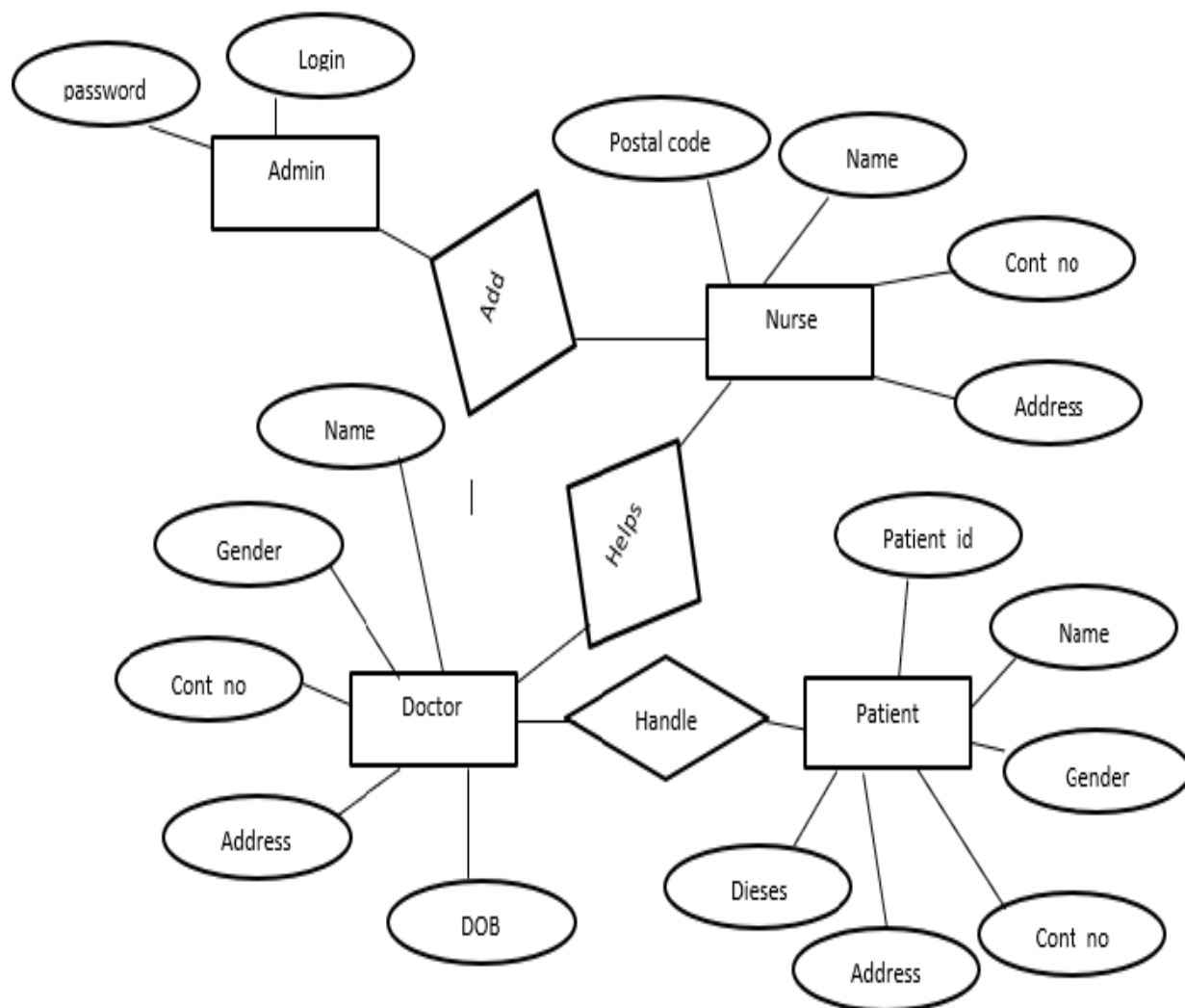


Bill Creation:

Medicine



E-R Diagram



System Implementation

Home Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Hospital_Management_System_Application
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnDoctor_Click(object sender, EventArgs e)
        {
            Form7 Doctor = new Form7();
            Doctor.Show();
        }

        private void btnNurse_Click(object sender, EventArgs e)
        {
            Form8 Nurse = new Form8();
            Nurse.Show();
        }

        private void btnAppointment_Click(object sender, EventArgs e)
        {
            Form2 Appointment = new Form2();
            Appointment.Show();
        }

        private void btnPatient_Click(object sender, EventArgs e)
        {
            Form_5_Patient = new Form_5_();
            Patient.Show();
        }

        private void btnAddDoc_Click(object sender, EventArgs e)
        {
            Form3 Doctor = new Form3();
            Doctor.Show();
        }

        private void btnExit_Click(object sender, EventArgs e)
        {
            DialogResult iExit;
            iExit = MessageBox.Show("Confirm if you want to exit", "Hospital Management System",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (iExit == DialogResult.Yes)

```

```

    {
        Application.Exit();
    }
}

private void btnClose_Click(object sender, EventArgs e)
{
    txtUsername.Clear();
    txtPassword.Clear();

    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the program", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.WindowState = FormWindowState.Minimized;
    }
}

private void btnAddNurse_Click(object sender, EventArgs e)
{
    Form4 Nurse = new Form4();
    Nurse.Show();
}

private void btnPayment_Click(object sender, EventArgs e)
{
    Form_6_ Payment = new Form_6_();
    Payment.Show();
}

private void btnContact_Click(object sender, EventArgs e)
{
    Form11 Medicine = new Form11();
    Medicine.Show();
}

private void btnAbout_Click(object sender, EventArgs e)
{
    Form12 About = new Form12();
    About.Show();
}

private void btnDoctorList_Click(object sender, EventArgs e)
{
    Form10 doctorlist = new Form10();
    doctorlist.Show();
}

private void btnLogOut_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Are you sure want to Logout From Admin Login ?", "Hospital Management
System", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        txtUsername.Clear();
        txtPassword.Clear();
        txtUsername.Focus();

        btnPayment.Enabled = false;
    }
}

```

```

        btnAddNurse.Enabled = false;
        btnAddDoc.Enabled = false;
        btnPatient.Enabled = false;
        btnExport.Enabled = false;
    }
}

private void btnLogin_Click(object sender, EventArgs e)
{
    if (txtUsername.Text == "Admin" && txtPassword.Text == "Pass")
    {
        btnPatient.Enabled = true;
        btnDoctor.Enabled = true;
        btnNurse.Enabled = true;
        btnPayment.Enabled = true;
        btnAddNurse.Enabled = true;
        btnAddDoc.Enabled = true;
        btnAppointment.Enabled = true;
        btnDoctorList.Enabled = true;
        btnAbout.Enabled = true;
        btnExport.Enabled = true;
        btnLogOut.Enabled = true;
        btnExport.Enabled = true;

        txtUsername.Clear();
        txtPassword.Clear();
    }
    else
    {
        MessageBox.Show("Please Enter correct Login details", "Hospital Management System");
        txtUsername.Clear();
        txtPassword.Clear();
        txtUsername.Focus();
    }
}

private void btnClear_Click(object sender, EventArgs e)
{
    txtUsername.Clear();
    txtPassword.Clear();
    txtUsername.Focus();

    btnPayment.Enabled = false;
    btnAddDoc.Enabled = false;
    btnAddNurse.Enabled = false;
    btnPatient.Enabled = false;
    btnLogOut.Enabled = false;
    btnExport.Enabled = false;
}

private void Form1_Load(object sender, EventArgs e)
{
    btnPatient.Enabled = false;
    btnDoctor.Enabled = true;
    btnNurse.Enabled = true;
    btnPayment.Enabled = false;
    btnAddNurse.Enabled = false;
    btnAddDoc.Enabled = false;
    btnAppointment.Enabled = true;
    btnDoctorList.Enabled = true;
    btnAbout.Enabled = true;
    btnExport.Enabled = true;
}

```



```
        btnLogOut.Enabled = false;
        btnExport.Enabled = false;
    }

    private void btnMinimise_Click(object sender, EventArgs e)
    {
        this.WindowState = FormWindowState.Minimized;
    }

    private void btnX_Click(object sender, EventArgs e)
    {
        DialogResult iExit;
        iExit = MessageBox.Show("Confirm if you want to exit", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (iExit == DialogResult.Yes)
        {
            Application.Exit();
        }
    }

    private void panel2_Paint(object sender, PaintEventArgs e)
    {
    }

    private void panel1_Paint(object sender, PaintEventArgs e)
    {
    }

    private void panel1_Paint_1(object sender, PaintEventArgs e)
    {
    }

    private void button1_Click(object sender, EventArgs e)
    {
    }

    private void label4_Click(object sender, EventArgs e)
    {
    }
}
}
```

Appointment Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Text.RegularExpressions;

namespace Hospital_Management_System_Application
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }

        private void appointmentBindingNavigatorSaveItem_Click(object sender, EventArgs e)
        {
            this.Validate();
            this.appointmentBindingSource.EndEdit();
            this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
        }

        private void Form2_Load(object sender, EventArgs e)
        {
            // TODO: This line of code loads data into the 'hospitalDataSet.Appointment' table. You can move, or remove
            it, as needed.
            this.appointmentTableAdapter.Fill(this.hospitalDataSet.Appointment);
        }

        private void btnAdd_Click(object sender, EventArgs e)
        {
            if (appointment_Ref_NoTextBox.Text == "")
            {
                MessageBox.Show("Enter the Appointment Ref No....");
            }
            else
            {
                if (first_NmaeTextBox.Text == "")
                {
                    MessageBox.Show("Enter The First Name");
                }
                else
                {
                    if (middle_NameTextBox.Text == "")
                    {
                        MessageBox.Show("Enter the Middle Name");
                    }
                    else
                    {
                        if (last_NameTextBox.Text == "")
                        {

```

```

    MessageBox.Show("Enter the Last Name");
}
else
{
    if (e_mail_AddressTextBox.Text == "")
    {
        MessageBox.Show("Enter Email Address");
    }
    else
    {
        if (genderComboBox.Text == "")
        {
            MessageBox.Show("Choose The Gender From box");
        }
        else
        {
            if (date_of_BirthDateTimePicker.Text == "")
            {
                MessageBox.Show("Select Date OF Birth");
            }

            else
            {
                if (area_CodeComboBox.Text == "")
                {
                    MessageBox.Show("Choose Area Code From Box");
                }
                else
                {
                    if (contact_NoTextBox.Text == "")
                    {
                        MessageBox.Show("Enter Contact Number");
                    }
                    else
                    {

                        if
                        (marital_StatusComboBox.Text == "")
                        {
                            MessageBox.Show("Choose The Marital Status From Boxx");
                        }
                        else
                        {
                            if (addressTextBox.Text == "")
                            {
                                MessageBox.Show("Enter Address");
                            }
                            else
                            {
                                if (postal_CodeTextBox.Text == "")
                                {
                                    MessageBox.Show("Enter Pincode of your Area");
                                }
                                else
                                {
                                    if (comboBoxCountry1.Text == "")
                                    {
                                        MessageBox.Show("Choose The Country From Box");
                                    }
                                    else
                                    {
                                        if (comboBoxstate1.Text == "")
                                        {

```

```

        MessageBox.Show("Choose the State From Box");
    }

    else
    {
        if (cityTextBox.Text == "")
        {
            MessageBox.Show("Enter City Name");
        }
        else
        {
            if (doctor_NameTextBox.Text == "")
            {
                MessageBox.Show("Enter Doctor Name");
            }
            else
            {
                if (doctor_Ref_NoTextBox.Text == "")
                {
                    MessageBox.Show("Enter Dr Ref Number ");
                }
                else
                {
                    if (date_TimeDateTimePicker.Text == "")
                    {
                        MessageBox.Show("Chooose the Date From Combobox");
                    }
                    else
                    {
                        this.appointmentBindingSource.AddNew();
                    }
                }
            }
        }
    }
}

private void btnDelete_Click(object sender, EventArgs e)
{
    this.appointmentBindingSource.RemoveCurrent();
}

```

```

private void btnNext_Click(object sender, EventArgs e)
{
    if (appointment_Ref_NoTextBox.Text == "" || first_NmaeTextBox.Text == "" || middle_NameTextBox.Text
    == "" || last_NameTextBox.Text == "" || e_mail_AddressTextBox.Text == "" || genderComboBox.Text == "" ||
    date_of_BirthDateTimePicker.Text == "" || area_CodeComboBox.Text == "" || marital_StatusComboBox.Text == "" ||
    contact_NoTextBox.Text == "" || addressTextBox.Text == "" || postal_CodeTextBox.Text == "" ||
    comboBoxCountry1.Text == "" || comboBoxstate1.Text == "" || cityTextBox.Text == "" || doctor_NameTextBox.Text
    == "" || doctor_Ref_NoTextBox.Text == "" || date_TimeDateTimePicker.Text == "")
    {
        MessageBox.Show("Plzzzzz Wait.....");
        this.appointmentBindingSource.RemoveCurrent();
    }
    else
    {
        this.appointmentBindingSource.MoveNext();
    }
}

private void btnPrevious_Click(object sender, EventArgs e)
{
    if (appointment_Ref_NoTextBox.Text == "" || first_NmaeTextBox.Text == "" || middle_NameTextBox.Text
    == "" || last_NameTextBox.Text == "" || e_mail_AddressTextBox.Text == "" || genderComboBox.Text == "" ||
    date_of_BirthDateTimePicker.Text == "" || area_CodeComboBox.Text == "" || marital_StatusComboBox.Text == "" ||
    contact_NoTextBox.Text == "" || addressTextBox.Text == "" || postal_CodeTextBox.Text == "" ||
    comboBoxCountry1.Text == "" || comboBoxstate1.Text == "" || cityTextBox.Text == "" || doctor_NameTextBox.Text
    == "" || doctor_Ref_NoTextBox.Text == "" || date_TimeDateTimePicker.Text == "")
    {
        MessageBox.Show("Plzzzzz Wait.....");
        this.appointmentBindingSource.RemoveCurrent();
    }
    else
    {
        this.appointmentBindingSource.MovePrevious();
    }
}

private void btnSave_Click(object sender, EventArgs e)
{
    if (appointment_Ref_NoTextBox.Text == "")
    {
        MessageBox.Show("Fill The All Details.....");
    }
    else
    {
        if (first_NmaeTextBox.Text == "")
        {
            MessageBox.Show("Enter The First Name");
        }
        else
        {
            if (middle_NameTextBox.Text == "")
            {
                MessageBox.Show("Enter the Middle Name");
            }
            else
            {
                if (last_NameTextBox.Text == "")
                {
                    MessageBox.Show("Enter the Last Name");
                }
                else
            }
        }
    }
}

```

```

{
    if (e_mail_AddressTextBox.Text == "")
    {
        MessageBox.Show("Enter Email Address");
    }
    else
    {
        if (genderComboBox.Text == "")
        {
            MessageBox.Show("Choose The Gender From box");
        }
        else
        {
            if (date_of_BirthDateTimePicker.Text == "")
            {
                MessageBox.Show("Select Date OF Birth");
            }

            else
            {
                if (area_CodeComboBox.Text == "")
                {
                    MessageBox.Show("Choose Area Code From Box");
                }
                else
                {
                    if (contact_NoTextBox.Text == "")
                    {
                        MessageBox.Show("Enter Contact Number");
                    }
                    else
                    {

                        if
                        (marital_StatusComboBox.Text == "")
                        {
                            MessageBox.Show("Choose The Marital Status From Boxx");
                        }
                        else
                        {
                            if (addressTextBox.Text == "")
                            {
                                MessageBox.Show("Enter Address");
                            }
                            else
                            {
                                if (postal_CodeTextBox.Text == "")
                                {
                                    MessageBox.Show("Enter Pincode of your Area");
                                }
                                else
                                {
                                    if (comboBoxCountry1.Text == "")
                                    {
                                        MessageBox.Show("Choose The Country From Box");
                                    }
                                    else
                                    {
                                        if (comboBoxstate1.Text == "")
                                        {
                                            MessageBox.Show("Choose the State From Box");
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```

else
{
    if (cityTextBox.Text == "")
    {
        MessageBox.Show("Enter City Name");
    }
    else
    {
        if (doctor_NameTextBox.Text == "")
        {
            MessageBox.Show("Enter Doctor Name");
        }
        else
        {
            if (doctor_Ref_NoTextBox.Text == "")
            {
                MessageBox.Show("Enter Dr Ref Number ");
            }
            else
            {
                if (date_TimeDateTimePicker.Text == "")
                {
                    MessageBox.Show("Chooose the Date From Combobox");
                }
                else
                {
                    this.Validate();
                    this.appointmentBindingSource.EndEdit();
                    this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
                }
            }
        }
    }
}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

}

private void btnClose_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the appointment page", "Hospital Management System", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

```

```

    }
}

private void btnExit_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to exit", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        Application.Exit();
    }
}

private void comboBoxCountry1_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void e_mail_AddressTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = "^([0-9a-zA-Z]([-\\.|\\w]*[0-9a-zA-Z])*@([0-9a-zA-Z]([-\\w]*[0-9a-zA-Z]\\.|)+[a-zA-Z]{2,9}))$";
    if (Regex.IsMatch(e_mail_AddressTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.e_mail_AddressTextBox, "Plzzz provide valid Mail address");
        return;
    }
}

private void first_NmaeTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(first_NmaeTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.first_NmaeTextBox, "Plzzz provide correct name");
        return;
    }
}

private void middle_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(middle_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.middle_NameTextBox, "Enter Correct Middle Name");
        return;
    }
}

```



```

private void last_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(last_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.last_NameTextBox, "Enter Correct Last Name");
        return;
    }
}

private void cityTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(cityTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.cityTextBox, "Enter Correct City Name");
        return;
    }
}

private void doctor_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(doctor_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.doctor_NameTextBox, "Enter Correct Doctor Name");
        return;
    }
}

private void contact_NoTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{3})?-? *\d{3}-? *-?\d{4}";
    if (Regex.IsMatch(contact_NoTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.contact_NoTextBox, "Enter Correct Contact Number");
        return;
    }
}

private void appointment_Ref_NoTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{0})?-? *\d{0}-? *-?\d{1}";
    if (Regex.IsMatch(appointment_Ref_NoTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
}

```

```

else
{
    errorHandler1.SetError(this.appointment_Ref_NoTextBox, "Enter sequentially and Correct Ref No");
    return;
}
}

private void postal_CodeTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{2})?-\d{2}-? *-\d{2}";
    if (Regex.IsMatch(postal_CodeTextBox.Text, pattern))
    {
        errorHandler1.Clear();
    }
    else
    {
        errorHandler1.SetError(this.postal_CodeTextBox, "Enter correct postal Code");
        return;
    }
}

private void doctor_Ref_NoTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{0})?-\d{0}-? *-\d{1}";
    if (Regex.IsMatch(doctor_Ref_NoTextBox.Text, pattern))
    {
        errorHandler1.Clear();
    }
    else
    {
        errorHandler1.SetError(this.doctor_Ref_NoTextBox, "Enter Correct Ref No");
        return;
    }
}

private void genderComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    if (genderComboBox.SelectedItem == null)
    {
        errorHandler1.SetError(genderComboBox, "Please Select the Items From the Combobox!");
    }
    else
    {
        errorHandler1.Clear();
    }
}

private void bindingNavigatorAddNewItem_Click(object sender, EventArgs e)
{
    if (appointment_Ref_NoTextBox.Text == "" || first_NmaeTextBox.Text == "" || middle_NameTextBox.Text == "" || last_NameTextBox.Text == "" || e_mail_AddressTextBox.Text == "" || genderComboBox.Text == "" || date_of_BirthDateTimePicker.Text == "" || area_CodeComboBox.Text == "" || marital_StatusComboBox.Text == "" || contact_NoTextBox.Text == "" || addressTextBox.Text == "" || postal_CodeTextBox.Text == "" || comboBoxCountry1.Text == "" || comboBoxstate1.Text == "" || cityTextBox.Text == "" || doctor_NameTextBox.Text == "" || doctor_Ref_NoTextBox.Text == "" || date_TimeDateTimePicker.Text == "")
    {
        MessageBox.Show("Plzzzzz Fill the All Details.....");
    }
    else
    {
        this.appointmentBindingSource.AddNew();
    }
}

```

```

    }

}

private void keypress(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KeyPr(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KeyPress(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KeyPress1(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KeyPress2(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP1(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP2(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP3(object sender, KeyPressEventArgs e)

```

```

    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }

private void KP4(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP5(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP7(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP6(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP8(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP9(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void date_TimeDateTimePicker_ValueChanged(object sender, EventArgs e)
{
}

private void date_of_BirthDateTimePicker_ValueChanged(object sender, EventArgs e)
{
}

```

```

    }

    private void area_CodeComboBox_SelectedIndexChanged(object sender, EventArgs e)
    {

    }

    private void marital_StatusComboBox_SelectedIndexChanged(object sender, EventArgs e)
    {
        string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
        if (Regex.IsMatch(marital_StatusComboBox.Text, pattern))
        {
            errorProvider1.Clear();
        }
        else
        {
            errorProvider1.SetError(this.marital_StatusComboBox, "Plzzz provide correct status.....");
            return;
        }
    }

    private void addressTextBox_TextChanged(object sender, EventArgs e)
    {

    }

    private void comboBoxstate1_SelectedIndexChanged(object sender, EventArgs e)
    {

    }

    private void KeyPress22(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void button1_Click(object sender, EventArgs e)
    {

    }

    private void btnMinimise_Click(object sender, EventArgs e)
    {
        this.WindowState = FormWindowState.Minimized;
    }

    private void btnX_Click(object sender, EventArgs e)
    {
        DialogResult iExit;
        iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (iExit == DialogResult.Yes)
        {
            this.Hide();
        }
    }
}

```

Doctor Profile Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Text.RegularExpressions;

namespace Hospital_Management_System_Application
{
    public partial class Form3 : Form
    {
        public Form3()
        {
            InitializeComponent();
        }

        private void doctorBindingNavigatorSaveItem_Click(object sender, EventArgs e)
        {
            this.Validate();
            this.doctorBindingSource.EndEdit();
            this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
        }

        private void Form3_Load(object sender, EventArgs e)
        {
            // TODO: This line of code loads data into the 'hospitalDataSet.Doctor' table. You can move, or remove it, as
            needed.
            this.doctorTableAdapter.Fill(this.hospitalDataSet.Doctor);
        }

        private void comboBoxCountry_SelectedIndexChanged(object sender, EventArgs e)
        {
        }

        private void btnAdd_Click(object sender, EventArgs e)
        {
            if (doctor_Ref_NoTextBox.Text == "")
            {
                MessageBox.Show("Enter the Doctor Ref No....");
            }
            else
            {
                if (first_NameTextBox.Text == "")
                {
                    MessageBox.Show("Enter The First Name");
                }
                else
                {
                    if (middle_NameTextBox.Text == "")
                    {
                        MessageBox.Show("Enter the Middle Name");
                    }
                }
            }
        }
    }
}

```

```

else
{
    if (last_NameTextBox.Text == "")
    {
        MessageBox.Show("Enter the Last Name");
    }
    else
    {
        if (e_mail_AddressTextBox.Text == "")
        {
            MessageBox.Show("Enter Email Address");
        }
        else
        {
            if (genderComboBox.Text == "")
            {
                MessageBox.Show("Choose The Gender From box");
            }
            else
            {
                if (dOBDateTimePicker.Text == "")
                {
                    MessageBox.Show("Select Date OF Birth");
                }

                else
                {
                    if (area_CodeComboBox.Text == "")
                    {
                        MessageBox.Show("Choose Area Code From Box");
                    }
                    else
                    {
                        if (contact_NoTextBox.Text == "")
                        {
                            MessageBox.Show("Enter Contact Number");
                        }
                        else
                        {
                            if
                                (marital_StatusComboBox.Text == "")
                            {
                                MessageBox.Show("Choose The Status From Boxx");
                            }
                            else
                            {
                                if (addressTextBox.Text == "")
                                {
                                    MessageBox.Show("Enter Address");
                                }
                                else
                                {
                                    if (postal_CodeTextBox.Text == "")
                                    {
                                        MessageBox.Show("Enter Pincode of your Area");
                                    }
                                    else
                                    {
                                        if (comboBoxCountry.Text == "")
                                        {
                                            MessageBox.Show("Choose The Country From Box");
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```
else
{
    if (comboBoxstate.Text == "")
    {
        MessageBox.Show("Choose the State From Box");
    }

    else
    {
        if (cityTextBox.Text == "")
        {
            MessageBox.Show("Enter City Name");
        }
        else
        {
            if(dateDateTimePicker.Text=="")
            {
                MessageBox.Show("Select Date.....From Boxx");
            }
            else
            {
                if (qualificationComboBox.Text == "")
                {
                    MessageBox.Show("Choose Qualification From Boxx");
                }
                else
                {
                    this.doctorBindingSource.AddNew();
                }
            }
        }
    }
}
}
```

```
private void btnDelete_Click(object sender, EventArgs e)
{
    this.doctorBindingSource.RemoveCurrent();
}
```

```
private void btnNext_Click(object sender, EventArgs e)
{
```



```

        if (doctor_Ref_NoTextBox.Text == "" || first_NameTextBox.Text == "" || middle_NameTextBox.Text == "" ||
last_NameTextBox.Text == "" || e_mail_AddressTextBox.Text == "" || genderComboBox.Text == "" ||
dOBDateTimePicker.Text == "" || area_CodeComboBox.Text == "" || marital_StatusComboBox.Text == "" ||
contact_NoTextBox.Text == "" || addressTextBox.Text == "" || postal_CodeTextBox.Text == "" ||
comboBoxCountry.Text == "" || comboBoxstate.Text == "" || cityTextBox.Text == "" || dateDateTimePicker.Text ==
"" || qualificationComboBox.Text == "")
        {
            MessageBox.Show("Plzzzzz Wait.....");
            this.doctorBindingSource.RemoveCurrent();
        }
        else
        {
            this.doctorBindingSource.MoveNext();
        }
    }

    private void btnPrevious_Click(object sender, EventArgs e)
    {
        if (doctor_Ref_NoTextBox.Text == "" || first_NameTextBox.Text == "" || middle_NameTextBox.Text == "" ||
last_NameTextBox.Text == "" || e_mail_AddressTextBox.Text == "" || genderComboBox.Text == "" ||
dOBDateTimePicker.Text == "" || area_CodeComboBox.Text == "" || marital_StatusComboBox.Text == "" ||
contact_NoTextBox.Text == "" || addressTextBox.Text == "" || postal_CodeTextBox.Text == "" ||
comboBoxCountry.Text == "" || comboBoxstate.Text == "" || cityTextBox.Text == "" || dateDateTimePicker.Text ==
"" || qualificationComboBox.Text == "")
        {
            MessageBox.Show("Plzzzzz Wait.....");
            this.doctorBindingSource.RemoveCurrent();
        }
        else
        {
            this.doctorBindingSource.MovePrevious();
        }
    }

    private void btnSave_Click(object sender, EventArgs e)
    {
        if (doctor_Ref_NoTextBox.Text == "")
        {
            MessageBox.Show("Enter the Doctor Ref No....");
        }
        else
        {
            if (first_NameTextBox.Text == "")
            {
                MessageBox.Show("Enter The First Name");
            }
            else
            {
                if (middle_NameTextBox.Text == "")
                {
                    MessageBox.Show("Enter the Middle Name");
                }
                else
                {
                    if (last_NameTextBox.Text == "")
                    {
                        MessageBox.Show("Enter the Last Name");
                    }
                    else
                    {
                        if (e_mail_AddressTextBox.Text == "")

```

```

{
    MessageBox.Show("Enter Email Address");
}
else
{
    if (genderComboBox.Text == "")
    {
        MessageBox.Show("Choose The Gender From box");
    }
    else
    {
        if (dOBDateTimePicker.Text == "")
        {
            MessageBox.Show("Select Date OF Birth");
        }

        else
        {
            if (area_CodeComboBox.Text == "")
            {
                MessageBox.Show("Choose Area Code From Box");
            }
            else
            {
                if (contact_NoTextBox.Text == "")
                {
                    MessageBox.Show("Enter Contact Number");
                }
                else
                {
                    if
                    (marital_StatusComboBox.Text == "")
                    {
                        MessageBox.Show("Choose The Status From Boxx");
                    }
                    else
                    {
                        if (addressTextBox.Text == "")
                        {
                            MessageBox.Show("Enter Address");
                        }
                        else
                        {
                            if (postal_CodeTextBox.Text == "")
                            {
                                MessageBox.Show("Enter Pincode of your Area");
                            }
                            else
                            {
                                if (comboBoxCountry.Text == "")
                                {
                                    MessageBox.Show("Choose The Country From Box");
                                }
                                else
                                {
                                    if (comboBoxstate.Text == "")
                                    {
                                        MessageBox.Show("Choose the State From Box");
                                    }

                                    else
                                    {

```

```

        if (cityTextBox.Text == "")
        {
            MessageBox.Show("Enter City Name");
        }
        else
        {
            if (dateDateTimePicker.Text == "")
            {
                MessageBox.Show("Select Date.....From Boxx");
            }
            else
            {
                if (qualificationComboBox.Text == "")
                {
                    MessageBox.Show("Choose Qualification From Boxx");
                }
                else
                {
                    this.Validate();
                    this.doctorBindingSource.EndEdit();
                    this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
                }
            }
        }
    }
}

private void btnClose_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the doctor page", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

private void btnExit_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to exit form application", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)

```

```

    {
        Application.Exit();
    }
}

private void countryComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void countryComboBox_SelectedIndexChanged_1(object sender, EventArgs e)
{
}

private void stateComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void doctor_Ref_NoTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{0})?-*\d{0}-? *-\d{1}";
    if (Regex.IsMatch(doctor_Ref_NoTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.doctor_Ref_NoTextBox, "Enter sequentially and Correct Ref No");
        return;
    }
}

private void bindingNavigatorAddNewItem_Click(object sender, EventArgs e)
{
    if (doctor_Ref_NoTextBox.Text == "" || first_NameTextBox.Text == "" || middle_NameTextBox.Text == "" ||
        last_NameTextBox.Text == "" || e_mail_AddressTextBox.Text == "" || genderComboBox.Text == "" ||
        dOBDateTimePicker.Text == "" || area_CodeComboBox.Text == "" || marital_StatusComboBox.Text == "" ||
        contact_NoTextBox.Text == "" || addressTextBox.Text == "" || postal_CodeTextBox.Text == "" ||
        comboBoxCountry.Text == "" || comboBoxstate.Text == "" || cityTextBox.Text == "" || dateDateTimePicker.Text ==
        "" || qualificationComboBox.Text == "")
    {
        MessageBox.Show("Plzzzzz Fill the All Details.....");
    }
    else
    {
        this.doctorBindingSource.AddNew();
    }
}

private void KP1(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP2(object sender, KeyPressEventArgs e)
{
}

```

```
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void KP3(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void KP4(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void KP5(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void KP6(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void KP7(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void KP9(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void KP10(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void KP11(object sender, KeyPressEventArgs e)
```

```

{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP12(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void first_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(first_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.first_NameTextBox, "Plzzz provide correct name");
        return;
    }
}

private void middle_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(middle_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.middle_NameTextBox, "Enter Correct Middle Name");
        return;
    }
}

private void last_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(last_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.last_NameTextBox, "Enter Correct Last Name");
        return;
    }
}

private void e_mail_AddressTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = "^[0-9a-zA-Z]([-\\.|\\w]*[0-9a-zA-Z])*@[0-9a-zA-Z]([-\\w]*[0-9a-zA-Z]\\.|)+[a-zA-Z]
Z]{2,9})$";
    if (Regex.IsMatch(e_mail_AddressTextBox.Text, pattern))
    {

```

```

        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.e_mail_AddressTextBox, "Plzzz provide valid Mail address");
        return;
    }
}

private void genderComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    if (genderComboBox.SelectedItem == null)
    {
        errorProvider1.SetError(genderComboBox, "Please Select the Items From the Combobox!");
    }
    else
    {
        errorProvider1.Clear();
    }
}

private void contact_NoTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{3})?-\d{3}-?\d{4}";
    if (Regex.IsMatch(contact_NoTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.contact_NoTextBox, "Enter Correct Contact Number");
        return;
    }
}

private void area_CodeComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void marital_StatusComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(marital_StatusComboBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.marital_StatusComboBox, "Plzzz provide the correct Status from boxx....");
        return;
    }
}

private void postal_CodeTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{2})?-\d{2}-?\d{2}";
    if (Regex.IsMatch(postal_CodeTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else

```

```

        {
            errorProvider1.SetError(this.postal_CodeTextBox, "Enter correct postal code");
            return;
        }
    }
private void cityTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(cityTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.cityTextBox, "Enter Correct City Name");
        return;
    }
}

private void qualificationComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    string pattern = @"\D+";
    if (Regex.IsMatch(qualificationComboBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.qualificationComboBox, "Plzzz provide the correct Qualification.....");
        return;
    }
}

private void addressTextBox_TextChanged(object sender, EventArgs e)
{
}

private void button1_Click(object sender, EventArgs e)
{
}

private void btnMinimise_Click(object sender, EventArgs e)
{
    this.WindowState = FormWindowState.Minimized;
}

private void btnX_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}
private void doctorDataGridView_CellContentClick(object sender, DataGridViewCellEventArgs e)
{
}
}

```


Nurse Profile Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Text.RegularExpressions;

namespace Hospital_Management_System_Application
{
    public partial class Form4 : Form
    {
        public Form4()
        {
            InitializeComponent();
        }

        private void nurseBindingNavigatorSaveItem_Click(object sender, EventArgs e)
        {
            this.Validate();
            this.nurseBindingSource.EndEdit();
            this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
        }

        private void Form4_Load(object sender, EventArgs e)
        {
            // TODO: This line of code loads data into the 'hospitalDataSet.Nurse' table. You can move, or remove it, as
            // needed.
            this.nurseTableAdapter.Fill(this.hospitalDataSet.Nurse);
        }

        private void btnAdd_Click(object sender, EventArgs e)
        {
            if (nurse_IDTextBox.Text == "")
            {
                MessageBox.Show("Enter the Nurse ID....");
            }
            else
            {
                if (first_NameTextBox.Text == "")
                {
                    MessageBox.Show("Enter The First Name");
                }
                else
                {
                    if (middle_NameTextBox.Text == "")
                    {
                        MessageBox.Show("Enter the Middle Name");
                    }
                    else
                    {
                        if (last_NameTextBox.Text == "")
                        {
                            MessageBox.Show("Enter the Last Name");
                        }
                    }
                }
            }
        }
    }
}

```

```

else
{
    if (e_mail_AddressTextBox.Text == "")
    {
        MessageBox.Show("Enter Email Address");
    }
    else
    {
        if (genderComboBox.Text == "")
        {
            MessageBox.Show("Choose The Gender From box");
        }
        else
        {
            if (dOBDateTimePicker.Text == "")
            {
                MessageBox.Show("Select Date OF Birth");
            }

            else
            {
                if (area_CodeComboBox.Text == "")
                {
                    MessageBox.Show("Choose Area Code From Box");
                }
                else
                {
                    if (contact_NoTextBox.Text == "")
                    {
                        MessageBox.Show("Enter Contact Number");
                    }
                    else
                    {

                        if
                        (marital_StatusComboBox.Text == "")
                        {
                            MessageBox.Show("Choose The Status Frem Boxx");
                        }
                        else
                        {
                            if (addressTextBox.Text == "")
                            {
                                MessageBox.Show("Enter Address");
                            }
                            else
                            {
                                if (postal_CodeTextBox.Text == "")
                                {
                                    MessageBox.Show("Enter Pincode of your Area");
                                }
                                else
                                {
                                    if (comboBoxCountry2.Text == "")
                                    {
                                        MessageBox.Show("Choose The Country From Box");
                                    }
                                    else
                                    {
                                        if (comboBoxstate2.Text == "")
                                        {
                                            MessageBox.Show("Choose the State From Box");
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

[illegible]

```

private void btnPrevious_Click(object sender, EventArgs e)
{
    if (nurse_IDTextBox.Text == "" || first_NameTextBox.Text == "" || middle_NameTextBox.Text == "" ||
last_NameTextBox.Text == "" || e_mail_AddressTextBox.Text == "" || genderComboBox.Text == "" ||
dOBDateTimePicker.Text == "" || area_CodeComboBox.Text == "" || marital_StatusComboBox.Text == "" ||
contact_NoTextBox.Text == "" || addressTextBox.Text == "" || postal_CodeTextBox.Text == "" ||
comboBoxCountry2.Text == "" || comboBoxstate2.Text == "" || cityTextBox.Text == "" || dateDateTimePicker.Text
== "")
    {
        MessageBox.Show("Plzzzzz Wait.....");
        this.nurseBindingSource.RemoveCurrent();
    }
    else
    {
        {
            this.nurseBindingSource.MovePrevious();
        }
    }
}

private void btnSave_Click(object sender, EventArgs e)
{
    if (nurse_IDTextBox.Text == "")
    {
        MessageBox.Show("Enter the Nurse ID....");
    }
    else
    {
        {
            if (first_NameTextBox.Text == "")
            {
                MessageBox.Show("Enter The First Name");
            }
            else
            {
                {
                    if (middle_NameTextBox.Text == "")
                    {
                        MessageBox.Show("Enter the Middle Name");
                    }
                    else
                    {
                        {
                            if (last_NameTextBox.Text == "")
                            {
                                MessageBox.Show("Enter the Last Name");
                            }
                            else
                            {
                                {
                                    if (e_mail_AddressTextBox.Text == "")
                                    {
                                        MessageBox.Show("Enter Email Address");
                                    }
                                    else
                                    {
                                        {
                                            if (genderComboBox.Text == "")
                                            {
                                                MessageBox.Show("Choose The Gender From box");
                                            }
                                            else
                                            {
                                                {
                                                    if (dOBDateTimePicker.Text == "")
                                                    {
                                                        MessageBox.Show("Select Date OF Birth");
                                                    }
                                                }
                                            }
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```

else
{
    if (area_CodeComboBox.Text == "")
    {
        MessageBox.Show("Choose Area Code From Box");
    }
    else
    {
        if (contact_NoTextBox.Text == "")
        {
            MessageBox.Show("Enter Contact Number");
        }
        else
        {
            if
            (marital_StatusComboBox.Text == "")
            {
                MessageBox.Show("Choose The Status From Boxx");
            }
            else
            {
                if (addressTextBox.Text == "")
                {
                    MessageBox.Show("Enter Address");
                }
                else
                {
                    if (postal_CodeTextBox.Text == "")
                    {
                        MessageBox.Show("Enter Pincode of your Area");
                    }
                    else
                    {
                        if (comboBoxCountry2.Text == "")
                        {
                            MessageBox.Show("Choose The Country From Box");
                        }
                        else
                        {
                            if (comboBoxstate2.Text == "")
                            {
                                MessageBox.Show("Choose the State From Box");
                            }

                            else
                            {
                                if (cityTextBox.Text == "")
                                {
                                    MessageBox.Show("Enter City Name");
                                }
                                else
                                {
                                    if(dateDateTimePicker.Text == "")
                                    {
                                        MessageBox.Show("Select Date.....From Boxx");
                                    }

                                    else
                                    {
                                        this.Validate();
                                        this.nurseBindingSource.EndEdit();
                                        this.tableAdapterManager.UpdateAll(this.hospitalDataSet);

```

```

    }
}

private void btnClose_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the nurse page", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

private void btnExit_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to exit the application....", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        Application.Exit();
    }
}

private void bindingNavigatorAddNewItem_Click(object sender, EventArgs e)
{
    if (nurse_IDTextBox.Text == "" || first_NameTextBox.Text == "" || middle_NameTextBox.Text == "" ||
last_NameTextBox.Text == "" || e_mail_AddressTextBox.Text == "" || genderComboBox.Text == "" ||
dOBDateTimePicker.Text == "" || area_CodeComboBox.Text == "" || marital_StatusComboBox.Text == "" ||
contact_NoTextBox.Text == "" || addressTextBox.Text == "" || postal_CodeTextBox.Text == "" ||
comboBoxCountry2.Text == "" || comboBoxstate2.Text == "" || cityTextBox.Text == "" || dateDateTimePicker.Text
== "")
    {
        MessageBox.Show("Plzzzzz Fill the All Details.....");
    }
    else
    {
        this.nurseBindingSource.AddNew();
    }
}

```

```

private void nurse_IDTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{0})?-? *\d{0}-? *-?\d{1}";
    if (Regex.IsMatch(nurse_IDTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.nurse_IDTextBox, "Enter sequentially and Correcct Nurse ID.....");
        return;
    }
}

private void first_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^([A-z-a-z][a-z-a-z][a-z-a-z]+)$";
    if (Regex.IsMatch(first_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.first_NameTextBox, "Plzzz provide correct name");
        return;
    }
}

private void middle_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^([A-z-a-z][a-z-a-z][a-z-a-z]+)$";
    if (Regex.IsMatch(middle_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.middle_NameTextBox, "Plzzz provide correct name");
        return;
    }
}

private void last_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^([A-z-a-z][a-z-a-z][a-z-a-z]+)$";
    if (Regex.IsMatch(last_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.last_NameTextBox, "Plzzz provide correct name");
        return;
    }
}

private void e_mail_AddressTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = "^[0-9a-zA-Z]([-\\.|\\w]*[0-9a-zA-Z])*@[0-9a-zA-Z]([-\\w]*[0-9a-zA-Z]\\.|)+[a-zA-Z]{2,9}$";
    if (Regex.IsMatch(e_mail_AddressTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
}

```

```

    }
    else
    {
        errorProvider1.SetError(this.e_mail_AddressTextBox, "Plzzz provide valid Mail address");
        return;
    }
}

private void genderComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    if (genderComboBox.SelectedItem == null)
    {
        errorProvider1.SetError(genderComboBox, "Please Select the Items From the Combobox!");
    }
    else
    {
        errorProvider1.Clear();
    }
}

private void contact_NoTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{3})?-\d{3}-?\d{4}";
    if (Regex.IsMatch(contact_NoTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.contact_NoTextBox, "Enter Correct Contact Number");
        return;
    }
}

private void marital_StatusComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(marital_StatusComboBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.marital_StatusComboBox, "Plzzz provide correct detail from box...");
        return;
    }
}

private void postal_CodeTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{2})?-\d{2}-?\d{2}";
    if (Regex.IsMatch(postal_CodeTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.postal_CodeTextBox, "Enter correct postal code.....");
        return;
    }
}

private void comboBoxCountry2_SelectedIndexChanged(object sender, EventArgs e)

```



```

{
}

private void comboBoxstate2_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void cityTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^([A-z-a-z][a-z-a-z][a-z-a-z]+)$";
    if (Regex.IsMatch(cityTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.cityTextBox, "Enter Correct City Name");
        return;
    }
}

private void area_CodeComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void addressTextBox_TextChanged(object sender, EventArgs e)
{
}

private void KP21(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP22(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP23(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP24(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP25(object sender, KeyPressEventArgs e)
{
}

```

```

        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    private void KP26(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }
    private void KP27(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }
    private void KP28(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }
    private void KP29(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }
    private void KP30(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }
    private void KP31(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }
    private void btnMinimise_Click(object sender, EventArgs e)
    {
        this.WindowState = FormWindowState.Minimized;
    }

    private void btnX_Click(object sender, EventArgs e)
    {
        DialogResult iExit;
        iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (iExit == DialogResult.Yes)
        {
            this.Hide();
        }
    }
}

```

Patient Registration Form Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Text.RegularExpressions;

namespace Hospital_Management_System_Application
{
    public partial class Form_5_ : Form
    {
        public Form_5_()
        {
            InitializeComponent();

            private void patientBindingNavigatorSaveItem_Click(object sender, EventArgs e)
            {
                this.Validate();
                this.patientBindingSource.EndEdit();
                this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
            }

            private void Form_5_Load(object sender, EventArgs e)
            {
                // TODO: This line of code loads data into the 'hospitalDataSet.Patient' table. You can move, or remove it, as
                needed.
                this.patientTableAdapter.Fill(this.hospitalDataSet.Patient);
            }

            private void bindingNavigatorAddNewItem_Click(object sender, EventArgs e)
            {
                if (patien_IDTextBox.Text == "" || first_NameTextBox.Text == "" || middle_NameTextBox.Text == "" ||
                last_NameTextBox.Text == "" || admin_DateDateTimePicker.Text == "" || e_mail_AddressTextBox.Text == "" ||
                genderComboBox.Text == "" || date_of_BirthDateTimePicker.Text == "" || area_CodeComboBox.Text == "" ||
                marital_StatusComboBox.Text == "" || contact_NoTextBox.Text == "" || addressTextBox.Text == "" ||
                postal_CodeTextBox.Text == "" || countryComboBox.Text == "" || countryComboBox.Text == "" || cityTextBox.Text
                == "" || discharge_DateDateTimePicker.Text == "")
                {
                    MessageBox.Show("Plzzzzz Fill the All Details.....");
                }
                else
                {
                    this.patientBindingSource.AddNew();
                }
            }

            private void btnAdd_Click(object sender, EventArgs e)
            {
                if (patien_IDTextBox.Text == "")
                {
                    MessageBox.Show("Enter the Patient ID....");
                }
                else

```

```

{
    if (first_NameTextBox.Text == "")
    {
        MessageBox.Show("Enter The First Name");
    }
    else
    {
        if (middle_NameTextBox.Text == "")
        {
            MessageBox.Show("Enter the Middle Name");
        }
        else
        {
            if (last_NameTextBox.Text == "")
            {
                MessageBox.Show("Enter the Last Name");
            }
            else
            {
                if (admin_DateDateTimePicker.Text == "")
                {
                    MessageBox.Show("Select Date From Combo Box....");
                }
                else
                {
                    if (e_mail_AddressTextBox.Text == "")
                    {
                        MessageBox.Show("Enter Email Address");
                    }
                    else
                    {
                        if (genderComboBox.Text == "")
                        {
                            MessageBox.Show("Choose The Gender From box");
                        }
                        else
                        {
                            if (date_of_BirthDateTimePicker.Text == "")
                            {
                                MessageBox.Show("Select Date OF Birth");
                            }
                            else
                            {
                                if (area_CodeComboBox.Text == "")
                                {
                                    MessageBox.Show("Choose Area Code From Box");
                                }
                                else
                                {
                                    if (contact_NoTextBox.Text == "")
                                    {
                                        MessageBox.Show("Enter Contact Number");
                                    }
                                    else
                                    {
                                        if
                                        (marital_StatusComboBox.Text == "")
                                        {
                                            MessageBox.Show("Choose The Status From Boxx");
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```

```
else
{
    if (addressTextBox.Text == "")
    {
        MessageBox.Show("Enter Address");
    }
    else
    {
        if (postal_CodeTextBox.Text == "")
        {
            MessageBox.Show("Enter Pincode of your Area");
        }
        else
        {
            if (countryComboBox.Text == "")
            {
                MessageBox.Show("Choose The Country From Box");
            }
            else
            {
                if (countryComboBox.Text == "")
                {
                    MessageBox.Show("Choose the State From Box");
                }

                else
                {
                    if (cityTextBox.Text == "")
                    {
                        MessageBox.Show("Enter City Name");
                    }
                    else
                    {
                        if (discharge_DateDateTimePicker.Text == "")
                        {
                            MessageBox.Show("Select Date.....From Boxx");
                        }

                        else
                        {
                            this.patientBindingSource.AddNew();
                        }
                    }
                }
            }
        }
    }
}
}
```

```

private void btnDelete_Click(object sender, EventArgs e)
{
    this.patientBindingSource.RemoveCurrent();
}

private void btnPrevious_Click(object sender, EventArgs e)
{
    if (patien_IDTextBox.Text == "" || first_NameTextBox.Text == "" || middle_NameTextBox.Text == "" ||
last_NameTextBox.Text == "" || admin_DateDateTimePicker.Text == "" || e_mail_AddressTextBox.Text == "" ||
genderComboBox.Text == "" || date_of_BirthDateTimePicker.Text == "" || area_CodeComboBox.Text == "" ||
marital_StatusComboBox.Text == "" || contact_NoTextBox.Text == "" || addressTextBox.Text == "" ||
postal_CodeTextBox.Text == "" || countryComboBox.Text == "" || countryComboBox.Text == "" || cityTextBox.Text
== "" || discharge_DateDateTimePicker.Text == "")
    {
        MessageBox.Show("Please Wait.....");
        this.patientBindingSource.RemoveCurrent();
    }
    else
    {
        this.patientBindingSource.MovePrevious();
    }
}

private void btnNext_Click(object sender, EventArgs e)
{
    if (patien_IDTextBox.Text == "" || first_NameTextBox.Text == "" || middle_NameTextBox.Text == "" ||
last_NameTextBox.Text == "" || admin_DateDateTimePicker.Text == "" || e_mail_AddressTextBox.Text == "" ||
genderComboBox.Text == "" || date_of_BirthDateTimePicker.Text == "" || area_CodeComboBox.Text == "" ||
marital_StatusComboBox.Text == "" || contact_NoTextBox.Text == "" || addressTextBox.Text == "" ||
postal_CodeTextBox.Text == "" || countryComboBox.Text == "" || countryComboBox.Text == "" || cityTextBox.Text
== "" || discharge_DateDateTimePicker.Text == "")
    {
        MessageBox.Show("Plzzzzz Wait.....");
        this.patientBindingSource.RemoveCurrent();
    }
    else
    {
        this.patientBindingSource.MoveNext();
    }
}

private void btnSave_Click(object sender, EventArgs e)
{
    if (patien_IDTextBox.Text == "")
    {
        MessageBox.Show("Enter the Patient ID....");
    }
    else
    {
        if (first_NameTextBox.Text == "")
        {
            MessageBox.Show("Enter The First Name");
        }
        else
        {
            if (middle_NameTextBox.Text == "")

```

```

{
    MessageBox.Show("Enter the Middle Name");
}
else
{
    if (last_NameTextBox.Text == "")
    {
        MessageBox.Show("Enter the Last Name");
    }
    else
    {
        if (admin_DateDateTimePicker.Text == "")
        {
            MessageBox.Show("Select Date From Combo Box....");
        }
        else
        {
            if (e_mail_AddressTextBox.Text == "")
            {
                MessageBox.Show("Enter Email Address");
            }
            else
            {
                if (genderComboBox.Text == "")
                {
                    MessageBox.Show("Choose The Gender From box");
                }
                else
                {
                    if (date_of_BirthDateTimePicker.Text == "")
                    {
                        MessageBox.Show("Select Date OF Birth");
                    }

                    else
                    {
                        if (area_CodeComboBox.Text == "")
                        {
                            MessageBox.Show("Choose Area Code From Box");
                        }
                        else
                        {
                            if (contact_NoTextBox.Text == "")
                            {
                                MessageBox.Show("Enter Contact Number");
                            }
                            else
                            {
                                if
                                (marital_StatusComboBox.Text == "")
                                {
                                    MessageBox.Show("Choose The Status From Boxx");
                                }
                                else
                                {
                                    if (addressTextBox.Text == "")
                                    {
                                        MessageBox.Show("Enter Address");
                                    }
                                    else
                                    {
                                        if (postal_CodeTextBox.Text == "")

```

```

        {
            MessageBox.Show("Enter Pincode of your Area");
        }
        else
        {
            if (countryComboBox.Text == "")
            {
                MessageBox.Show("Choose The Country From Box");
            }
            else
            {
                if (countryComboBox.Text == "")
                {
                    MessageBox.Show("Choose the State From Box");
                }

                else
                {
                    if (cityTextBox.Text == "")
                    {
                        MessageBox.Show("Enter City Name");
                    }
                    else
                    {
                        if (discharge_DateDateTimePicker.Text == "")
                        {
                            MessageBox.Show("Select Date.....From Boxx");
                        }

                        else
                        {
                            this.Validate();
                            this.patientBindingSource.EndEdit();
                            this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
                        }
                    }
                }
            }
        }
    }
}

private void btnClose_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the patients page", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)

```



```

    {
        this.Hide();
    }
}

private void btnExit_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to exit the application....", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        Application.Exit();
    }
}

private void patien_IDTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{0})?-? *\d{0}-? *-?\d{1}";
    if (Regex.IsMatch(patien_IDTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.patien_IDTextBox, "Enter enter sequentially and correct ID");
        return;
    }
}

private void first_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^([A-z-a-z][a-z-a-z][a-z-a-z]+)$";
    if (Regex.IsMatch(first_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.first_NameTextBox, "Plzzz provide correct name");
        return;
    }
}

private void middle_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^([A-z-a-z][a-z-a-z][a-z-a-z]+)$";
    if (Regex.IsMatch(middle_NameTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.middle_NameTextBox, "Plzzz provide correct name");
        return;
    }
}

private void last_NameTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^([A-z-a-z][a-z-a-z][a-z-a-z]+)$";
    if (Regex.IsMatch(last_NameTextBox.Text, pattern))
    {

```

```

        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.last_NameTextBox, "Plzzz provide correct name");
        return;
    }
}

private void e_mail_AddressTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = "^([0-9a-zA-Z]([-\\.|\\w]*[0-9a-zA-Z])*@([0-9a-zA-Z]([-\\w]*[0-9a-zA-Z]\\.|)+[a-zA-Z-
Z]{2,9}))$";
    if (Regex.IsMatch(e_mail_AddressTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.e_mail_AddressTextBox, "Plzzz provide valid Mail address");
        return;
    }
}

private void date_of_BirthDateTimePicker_ValueChanged(object sender, EventArgs e)
{
}

private void genderComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    if (genderComboBox.SelectedItem == null)
    {
        errorProvider1.SetError(genderComboBox, "Please Select the Items From the Combobox!");
    }
    else
    {
        errorProvider1.Clear();
    }
}

private void contact_NoTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{1})?-?*\d{1}-?*\d{2}";
    if (Regex.IsMatch(contact_NoTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.contact_NoTextBox, "Enter Ref Number....");
        return;
    }
}

private void marital_StatusComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    string pattern = @"^([A-z-a-z][a-z-a-z][a-z-a-z]+)$";
    if (Regex.IsMatch(marital_StatusComboBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {

```

```

        errorProvider1.SetError(this.marital_StatusComboBox, "Plzzz provide correct status from boxx....");
        return;
    }
}
private void postal_CodeTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{2})?-\d{2}-?\d{2}";
    if (Regex.IsMatch(postal_CodeTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.postal_CodeTextBox, "Enter Correct postal code");
        return;
    }
}
private void cityTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"^[A-z-a-z][a-z-a-z][a-z-a-z]+$";
    if (Regex.IsMatch(cityTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.cityTextBox, "Plzzz provide correct name");
        return;
    }
}
private void KK1(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void KKK1(object sender, KeyPressEventArgs e)
{
}
private void LL(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void KL(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void R1(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}
}

```

```

private void R2(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void R3(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void R4(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void R5(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void R6(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void R7(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}
private void button2_Click(object sender, EventArgs e)
{
    this.WindowState = FormWindowState.Minimized;
}

private void button1_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}
private void button3_Click(object sender, EventArgs e)
{
    this.patientBindingSource.MoveLast();
}
}

```

Payment Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Text.RegularExpressions;

namespace Hospital_Management_System_Application
{
    public partial class Form_6_ : Form
    {
        public Form_6_()
        {
            InitializeComponent();
        }

        private void paymentBindingNavigatorSaveItem_Click(object sender, EventArgs e)
        {
            this.Validate();
            this.paymentBindingSource.EndEdit();
            this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
        }

        private void Form_6__Load(object sender, EventArgs e)
        {
            // TODO: This line of code loads data into the 'hospitalDataSet.Payment' table. You can move, or remove it, as
            needed.
            this.paymentTableAdapter.Fill(this.hospitalDataSet.Payment);
        }

        private void btnMinimise_Click(object sender, EventArgs e)
        {
            this.WindowState = FormWindowState.Minimized;
        }

        private void btnX_Click(object sender, EventArgs e)
        {
            DialogResult iExit;
            iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (iExit == DialogResult.Yes)
            {
                this.Hide();
            }
        }

        private void btnAdd_Click(object sender, EventArgs e)
        {
            if (patient_IDTextBox.Text == "")
            {
                MessageBox.Show("Enter the Patient ID...");
            }
            else

```

```

{
    if (date_TimeDateTimePicker.Text == "")
    {
        MessageBox.Show("Select the Date From Box");
    }
    else
    {
        if (method_of_PaymentComboBox.Text == "")
        {
            MessageBox.Show("Select Option For Payment");
        }
        else
        {
            if (card_NumberTextBox.Text == "")
            {
                MessageBox.Show("Enter Last 4 Digit Of Payment Card");
            }
            else
            {
                if (pin_NumberTextBox.Text == "")
                {
                    MessageBox.Show("Enter Pin Number");
                }
                else
                {
                    if (ammount_PaidTextBox.Text == "")
                    {
                        MessageBox.Show("Enter total Ammount Of Payment");
                    }

                    else
                    {
                        this.paymentBindingSource.AddNew();
                    }
                }
            }
        }
    }
}

private void btnDelete_Click(object sender, EventArgs e)
{
    this.paymentBindingSource.RemoveCurrent();
}

private void btnNext_Click(object sender, EventArgs e)
{
    if (patient_IDTextBox.Text == "" || date_TimeDateTimePicker.Text == "" ||
method_of_PaymentComboBox.Text == "" || card_NumberTextBox.Text == "" || pin_NumberTextBox.Text == "" ||
ammount_PaidTextBox.Text == "")
    {
        MessageBox.Show("Plzzzzz Wait.....");
        this.paymentBindingSource.RemoveCurrent();
    }
    else
    {
        this.paymentBindingSource.MoveNext();
    }
}

```

```

private void btnPrevious_Click(object sender, EventArgs e)
{
    if (patient_IDTextBox.Text == "" || date_TimeDateTimePicker.Text == "" ||
method_of_PaymentComboBox.Text == "" || card_NumberTextBox.Text == "" || pin_NumberTextBox.Text == "" ||
ammount_PaidTextBox.Text == "")
    {
        MessageBox.Show("Plzzzzz Wait.....");
        this.paymentBindingSource.RemoveCurrent();
    }
    else
    {
        this.paymentBindingSource.MovePrevious();
    }
}

private void btnSave_Click(object sender, EventArgs e)
{
    if (patient_IDTextBox.Text == "")
    {
        MessageBox.Show("Enter the Patient ID....");
    }
    else
    {
        if (date_TimeDateTimePicker.Text == "")
        {
            MessageBox.Show("Select the Date From Box");
        }
        else
        {
            if (method_of_PaymentComboBox.Text == "")
            {
                MessageBox.Show("Select Option For Payment");
            }
            else
            {
                if (card_NumberTextBox.Text == "")
                {
                    MessageBox.Show("Enter Last 4 Digit Of Payment Card");
                }
                else
                {
                    if (pin_NumberTextBox.Text == "")
                    {
                        MessageBox.Show("Enter Pin Number");
                    }
                    else
                    {
                        if (ammount_PaidTextBox.Text == "")
                        {
                            MessageBox.Show("Enter total Ammount Of Payment");
                        }
                    }
                }
            }
        }
    }
}

```

```

    }
    }
}

private void btnClose_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the payment page", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

private void btnExit_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to exit the application", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        Application.Exit();
    }
}

private void patient_IDTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{0})?-? *\d{0}-? *-?\d{1}";
    if (Regex.IsMatch(patient_IDTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.patient_IDTextBox, "Enter correct patient ID");
        return;
    }
}

private void card_NumberTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{1})?-? *\d{1}-? *-?\d{2}";
    if (Regex.IsMatch(card_NumberTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.card_NumberTextBox, "Enter Last 4 digit....");
        return;
    }
}

private void pin_NumberTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{1})?-? *\d{1}-? *-?\d{2}";
    if (Regex.IsMatch(pin_NumberTextBox.Text, pattern))
    {
        errorProvider1.Clear();
    }
}

```



```

        else
        {
            errorHandler1.SetError(this.pin_NumberTextBox, "enter 4 digit Pin Number");
            return;
        }
    }

    private void ammount_PaidTextBox_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void method_of_PaymentComboBox_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void card_NumberTextBox_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void pin_NumberTextBox_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void patient_IDTextBox_KeyPress(object sender, KeyPressEventArgs e)
    {
        if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
        {
            e.Handled = true;
        }
    }

    private void date_TimeDateTimePicker_ValueChanged(object sender, EventArgs e)
    {
    }
}

```

Doctor Login Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Hospital_Management_System_Application
{
    public partial class Form7 : Form
    {
        public Form7()
        {
            InitializeComponent();
        }

        private void btnLoginDoc_Click(object sender, EventArgs e)
        {
            if (txtUsernameDoc.Text == "Doctor" && txtPasswordDoc.Text == "Pass")
            {
                btnAppointmentDoc.Enabled = true;
                btnPatientDoc.Enabled = true;
                btnAddDoctorDoc.Enabled = true;
                btnLogOutDoc.Enabled = true;

                txtUsernameDoc.Clear();
                txtPasswordDoc.Clear();
            }
            else
            {
                MessageBox.Show("Please Enter correct Login details", "Hospital Management System");
                txtUsernameDoc.Clear();
                txtPasswordDoc.Clear();
                txtUsernameDoc.Focus();
            }
        }

        private void btnClearDoc_Click(object sender, EventArgs e)
        {
            txtUsernameDoc.Clear();
            txtPasswordDoc.Clear();
            txtUsernameDoc.Focus();

            btnAppointmentDoc.Enabled = false;
            btnPatientDoc.Enabled = false;
            btnAddDoctorDoc.Enabled = false;
            btnLogOutDoc.Enabled = false;
        }

        private void btnLogOutDoc_Click(object sender, EventArgs e)
        {
            DialogResult iExit;
            iExit = MessageBox.Show("Are you sure want to Logout From Doctor Login ?", "Hospital Management System", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (iExit == DialogResult.Yes)
            {

```

```

        txtUsernameDoc.Clear();
        txtPasswordDoc.Clear();
        txtUsernameDoc.Focus();

        btnAppointmentDoc.Enabled = false;
        btnPatientDoc.Enabled = false;
        btnAddDoctorDoc.Enabled = false;
    }
}

private void panel1_Paint(object sender, PaintEventArgs e)
{
}

private void Form7_Load(object sender, EventArgs e)
{
    btnAppointmentDoc.Enabled = false;
    btnPatientDoc.Enabled = false;
    btnAddDoctorDoc.Enabled = false;
    btnLogOutDoc.Enabled = false;
}

private void btnAppointmentDoc_Click(object sender, EventArgs e)
{
    Form9 Appointment = new Form9();
    Appointment.Show();
}

private void btnPatientDoc_Click(object sender, EventArgs e)
{
    Form_5_Patient = new Form_5_();
    Patient.Show();
}

private void btnAddDoctorDoc_Click(object sender, EventArgs e)
{
    Form3 Doctor = new Form3();
    Doctor.Show();
}

private void button1_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the Doctor Login", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

private void btnMinimise_Click(object sender, EventArgs e)
{
    this.WindowState = FormWindowState.Minimized;
}

private void btnX_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

private void txtUsernameDoc_TextChanged(object sender, EventArgs e)
{
}
}
}

```

Nurse Login Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Hospital_Management_System_Application
{
    public partial class Form8 : Form
    {
        public Form8()
        {
            InitializeComponent();
        }

        private void btnLoginNur_Click(object sender, EventArgs e)
        {
            if (txtUsernameNur.Text == "Nurse" && txtPasswordNur.Text == "Pass")
            {
                btnAppointmentNur.Enabled = true;
                btnPatientNur.Enabled = true;
                btnAddNur.Enabled = true;
                btnLogOutNur.Enabled = true;

                txtUsernameNur.Clear();
                txtPasswordNur.Clear();
            }
            else
            {
                MessageBox.Show("Please Enter correct Login details", "Hospital Management System");
                txtUsernameNur.Clear();
                txtPasswordNur.Clear();
                txtUsernameNur.Focus();
            }
        }

        private void btnClearNur_Click(object sender, EventArgs e)
        {
            txtUsernameNur.Clear();
            txtPasswordNur.Clear();
            txtUsernameNur.Focus();

            btnAppointmentNur.Enabled = false;
            btnPatientNur.Enabled = false;
            btnAddNur.Enabled = false;
            btnLogOutNur.Enabled = false;
        }

        private void btnLogOutNur_Click(object sender, EventArgs e)
        {
            DialogResult iExit;
            iExit = MessageBox.Show("Are you sure want to Logout From Nurse Login ?", "Hospital Management System", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (iExit == DialogResult.Yes)

```

```

    {
        txtUsernameNur.Clear();
        txtPasswordNur.Clear();
        txtUsernameNur.Focus();

        btnAppointmentNur.Enabled = false;
        btnPatientNur.Enabled = false;
        btnAddNur.Enabled = false;
    }
}

private void Form8_Load(object sender, EventArgs e)
{
    btnAppointmentNur.Enabled = false;
    btnPatientNur.Enabled = false;
    btnAddNur.Enabled = false;
    btnLogOutNur.Enabled = false;
}

private void btnAddNur_Click(object sender, EventArgs e)
{
    Form4 Nurse = new Form4();
    Nurse.Show();
}

private void btnPatientNur_Click(object sender, EventArgs e)
{
    Form_5_Patient = new Form_5_();
    Patient.Show();
}

private void button1_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the Nurse Login", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

private void btnAppointmentNur_Click(object sender, EventArgs e)
{
    Form11 Medicine = new Form11();
    Medicine.Show();
}

private void btnMinimise_Click(object sender, EventArgs e)
{
    this.WindowState = FormWindowState.Minimized;
}

private void btnX_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}
}
}

```

Appointment List Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Hospital_Management_System_Application
{
    public partial class Form9 : Form
    {
        public Form9()
        {
            InitializeComponent();

            private void Form9_Load(object sender, EventArgs e)
            {
                // TODO: This line of code loads data into the 'hospitalDataSet.Appointment' table. You can move, or remove
                it, as needed.
                this.appointmentTableAdapter.Fill(this.hospitalDataSet.Appointment);

            }

            private void btnClose_Click(object sender, EventArgs e)
            {
                DialogResult iExit;
                iExit = MessageBox.Show("Confirm if you want to close the Appointment List", "Hospital Management
                System", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
                if (iExit == DialogResult.Yes)
                {
                    this.Hide();
                }
            }

            private void btnMinimise_Click(object sender, EventArgs e)
            {
                this.WindowState = FormWindowState.Minimized;
            }

            private void btnX_Click(object sender, EventArgs e)
            {
                DialogResult iExit;
                iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
                MessageBoxButtons.YesNo, MessageBoxIcon.Question);
                if (iExit == DialogResult.Yes)
                {
                    this.Hide();
                }
            }
        }
    }
}

```

Doctor List Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Hospital_Management_System_Application
{
    public partial class Form10 : Form
    {
        public Form10()
        {
            InitializeComponent();
        }

        private void Form10_Load(object sender, EventArgs e)
        {
            // TODO: This line of code loads data into the 'hospitalDataSet.Doctor' table. You can move, or remove it, as
            needed.
            this.doctorTableAdapter.Fill(this.hospitalDataSet.Doctor);
        }

        private void btnClose_Click(object sender, EventArgs e)
        {
            DialogResult iExit;
            iExit = MessageBox.Show("Confirm if you want to close the Doctor List", "Hospital Management System",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (iExit == DialogResult.Yes)
            {
                this.Hide();
            }
        }

        private void btnMinimise_Click(object sender, EventArgs e)
        {
            this.WindowState = FormWindowState.Minimized;
        }

        private void btnX_Click(object sender, EventArgs e)
        {
            DialogResult iExit;
            iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
            MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (iExit == DialogResult.Yes)
            {
                this.Hide();
            }
        }
    }
}

```

Medicine Page Code:-

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Text.RegularExpressions;

namespace Hospital_Management_System_Application
{
    public partial class Form11 : Form
    {
        public Form11()
        {
            InitializeComponent();
        }

        private void medicineBindingNavigatorSaveItem_Click(object sender, EventArgs e)
        {
            this.Validate();
            this.medicineBindingSource.EndEdit();
            this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
        }

        private void Form11_Load(object sender, EventArgs e)
        {
            // TODO: This line of code loads data into the 'hospitalDataSet.Medicine' table. You can move, or remove it,
            as needed.
            this.medicineTableAdapter.Fill(this.hospitalDataSet.Medicine);
        }

        private void btnAdd_Click(object sender, EventArgs e)
        {
            if (patient_IDTextBox.Text == "")
            {
                MessageBox.Show("Enter the Patient ID....");
            }
            else
            {
                if (date_TimeDateTimePicker.Text == "")
                {
                    MessageBox.Show("Select the Date From Box");
                }
                else
                {
                    if (medicineComboBox.Text == "")
                    {
                        MessageBox.Show("Select Option For Payment");
                    }
                    else
                    {
                        if (amountTextBox.Text == "")
                        {

```



```

        MessageBox.Show("Enter ammount ");
    }

    else
    {
        this.medicineBindingSource.AddNew();
    }
}

}

}

private void btnExit_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to exit the program", "Hospital Management System",
    MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        Application.Exit();
    }
}

private void btnDelete_Click(object sender, EventArgs e)
{
    this.medicineBindingSource.RemoveCurrent();
}

private void btnNext_Click(object sender, EventArgs e)
{
    if (patient_IDTextBox.Text == "" || date_TimeDateTimePicker.Text == "" || medicineComboBox.Text == "" ||
    amountTextBox.Text == "")
    {
        MessageBox.Show("Plzzzzz Wait.....");
        this.medicineBindingSource.RemoveCurrent();
    }
    else
    {
        this.medicineBindingSource.MoveNext();
    }
}

private void btnPrevious_Click(object sender, EventArgs e)
{
    if (patient_IDTextBox.Text == "" || date_TimeDateTimePicker.Text == "" || medicineComboBox.Text == "" ||
    amountTextBox.Text == "")
    {
        MessageBox.Show("Plzzzzz Wait.....");
        this.medicineBindingSource.RemoveCurrent();
    }
    else
    {
        this.medicineBindingSource.MovePrevious();
    }
}

private void btnSave_Click(object sender, EventArgs e)
{
    if (patient_IDTextBox.Text == "")
    {
        MessageBox.Show("Enter the Patient ID....");
    }
}

```

```

    }
    else
    {
        if (date_TimeDateTimePicker.Text == "")
        {
            MessageBox.Show("Select the Date From Box");
        }
        else
        {
            if (medicineComboBox.Text == "")
            {
                MessageBox.Show("fill the medicine box");
            }
            else
            {
                if (amountTextBox.Text == "")
                {
                    MessageBox.Show("Enter Ammount");
                }

                else
                {
                    this.Validate();
                    this.medicineBindingSource.EndEdit();
                    this.tableAdapterManager.UpdateAll(this.hospitalDataSet);
                }
            }
        }
    }
}

private void btnClose_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want to close the Medicine Section", "Hospital Management
System", MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

private void bindingNavigatorAddNewItem_Click(object sender, EventArgs e)
{
    if (patient_IDTextBox.Text == "" || date_TimeDateTimePicker.Text == "" || medicineComboBox.Text == "" ||
amountTextBox.Text == "")
    {
        MessageBox.Show("Plzzzzz Fill the All Details.....");
    }
    else
    {
        this.medicineBindingSource.AddNew();
    }
}

private void patient_IDTextBox_TextChanged(object sender, EventArgs e)
{
    string pattern = @"(?:\d{0})?-? *\d{0}-? *-?\d{1}";
    if (Regex.IsMatch(patient_IDTextBox.Text, pattern))
    {

```

```

        errorProvider1.Clear();
    }
    else
    {
        errorProvider1.SetError(this.patient_IDTextBox, "Enter Correct patien Id");
        return;
    }
}

private void KP60(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void KP80(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsLetter(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void Kp72(object sender, KeyPressEventArgs e)
{
    if (!char.IsControl(e.KeyChar) && !char.IsDigit(e.KeyChar))
    {
        e.Handled = true;
    }
}

private void medicineComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void btnMinimise_Click(object sender, EventArgs e)
{
    this.WindowState = FormWindowState.Minimized;
}

private void btnX_Click(object sender, EventArgs e)
{
    DialogResult iExit;
    iExit = MessageBox.Show("Confirm if you want close", "Hospital Management System",
        MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (iExit == DialogResult.Yes)
    {
        this.Hide();
    }
}

private void date_TimeLabel_Click(object sender, EventArgs e)
{
}

private void date_TimeDateTimePicker_ValueChanged(object sender, EventArgs e)
{
    date_TimeDateTimePicker.CustomFormat = "dd-MM-yyyy";
}

private void date_TimeDateTimePicker_KeyDown(object sender, KeyEventArgs e)
{
    if ((e.KeyCode == Keys.Back) || (e.KeyCode == Keys.Delete))
    {
        date_TimeDateTimePicker.CustomFormat = " ";
    }
}
}

```

Result

Event Tables:-

Event	Trigger	Source	Activity	Response	Destination
Patient Registration	Patient Arrival	Reception	Get Patient details or check Registration	Patient ID	Patient
Appointment Fixing	Patient Requests For Appointment	Reception	Get Appointment Detail	Appointment No and Time	Patient
Bill Creation	Consulting Completed	Reception	Create and Calculate Bill	Receipt	Patient
Medicine Export and Update	New Medicines	Doctor	Update Medicine Detail	Medicine Information	Doctor

Screenshots:-

Home Page Design:-

This screenshot shows the home page of the Hospital Management System. It features a login section in the top left with fields for 'Username' and 'Password', and buttons for 'Log In' and 'Clear'. In the top right, a small box states 'Developed By Aniket Dhulap ©2019-2020'. The main title 'Hospital Management System' is prominently displayed in the center. Below the title, there is a grid of buttons for various functions: 'Doctor', 'Doctor's List', 'Reports', 'Appointment', 'Nurse', 'Add Doctor', 'Patients', 'Export Medicine', 'Payment', and 'Add Nurse'. At the bottom, there are buttons for 'Exit', 'Log Out', and 'Close'. The background of the interface is a blurred image of medical equipment.

After Admin Login:-

This screenshot shows the same Hospital Management System interface as the previous one, but after an admin login. The 'Username' field now contains the text 'Admin' and the 'Password' field contains '****'. The 'Log In' button is highlighted, indicating it has been clicked. The rest of the interface, including the title, function buttons, and background, remains the same as in the previous screenshot.

Doctor Page:-



After Doctor Login:-



Appointment List:-

Appointment_Ref_No	First_Nmae	Middle_Name	Last_Name	E-mail_Address	Gender	Date_of_Birth	Area_Code
1	ram	sham	bargode	a@gmail.com	Male	04-09-2019	+91
2	raj	shaman	laman	a@gmail.com	Male	12-09-2019	+1
3	prathamesh	vijay	gothankar	gotya@gmail.com	Male	14-07-1995	+32
4	Aniket	Sunil	Dhulap	aniketdhulap@gmail.com	Male	03-03-1999	+91
5	Ranjan	panjan	manjan	manjan@gmail.com	Male	02-02-2000	+54
6	sas	ass	sddd	s@gmail.com	Male	02-02-2000	+1
7	gsasadas	sdasdda	sadsa	a@gmail.com	Female	31-01-1990	+61
8	jsadkscs	scscscdc	cscod	cdd@gmail.com	Female	07-02-2019	+45

You Can't
Enjoy Wealth...
If You're
Not In Good
Health...

Close

Patient Registration:-

Patient ID:	1	
First Name:	Aniket	
Middle Name:	Sunil	
Last Name:	Dhulap	
Admin Date:	05-06-2019	
E-mail Address:	aniket@gmail.com	
Gender:	Male	
Date of Birth:	29-12-2008	
Area Code:	+56	
Patient_Ref_ID	4335	
Marital Status:	Single	
Address:	upale	
Postal Code:	632565	
Country:	Canada	
State:	Chiba	
City:	ratnagiri	
Discharge Date:	01-10-2019	

Patient_ID	First_Name	Middle_Name	Last_Name	Admin_Date	E-mail_Address
1	Aniket	Sunil	Dhulap	05-06-2019 05:04	aniket@gmail.com
2	raj	sham	raman	03-06-2019 05:04	raj@gmail.com
3	Rajdas	pavan	chavan	05-06-2019 05:04	ram@gmail.com

Add Delete Previous Next Save Exit Close

Add Doctor Profile:-

Doctors

Doctor_Ref_No	First_Name	Middle_Name	Last_Name	E-mail_Address
1	Ram	Shankar	Chavan	ram@gmail.com
2	Sham	Ram	Chaugule	sham@gmail.com
3	ketan	vijay	modak	ketan@gmail.com
*				

All About Doctor.

Add Delete Next Previou Save Close Exit

Form Fields:

Doctor Ref No:

First Name:

Middle Name:

Last Name:

E-mail Address:

DOB:

Gender:

Area Code:

Contact No:

Marital Status:

Address:

Postal Code:

Country:

State:

City:

Date:

Qualification:

Doctor List:-

Doctor_Ref_No	First_Name	Middle_Name	Last_Name	E-mail_Address	Gender	DOB	Area_Code	Contact_No
1	Ram	Shankar	Chavan	ram@gmail.com	Male	10-03-2010	+91	9878899009
2	Sham	Ram	Chaugule	sham@gmail.com	Male	02-02-1994	+91	9898099089
3	ketan	vijay	modak	ketan@gmail.com	Male	07-07-1995	+54	7878789808
*								

You Can't Enjoy Wealth... If You're Not In Good Health...

Close

Report Page:-



Appointment Page:-

Make Appointment

We Care.. We Love...

Appointment Ref No:

First Name: ram

Middle Name: sham

Last Name: bargode

E-mail Address: a@gmail.com

Gender: Male

Date of Birth: 04-09-2019

Area Code: +91

Contact No: 7878787878

Marital Status: Single

Address: atPost Bargode

Postal Code: 454556

Country: Brazil

State: Maharashtra

City: ratnagiri

Doctor Name: aniket

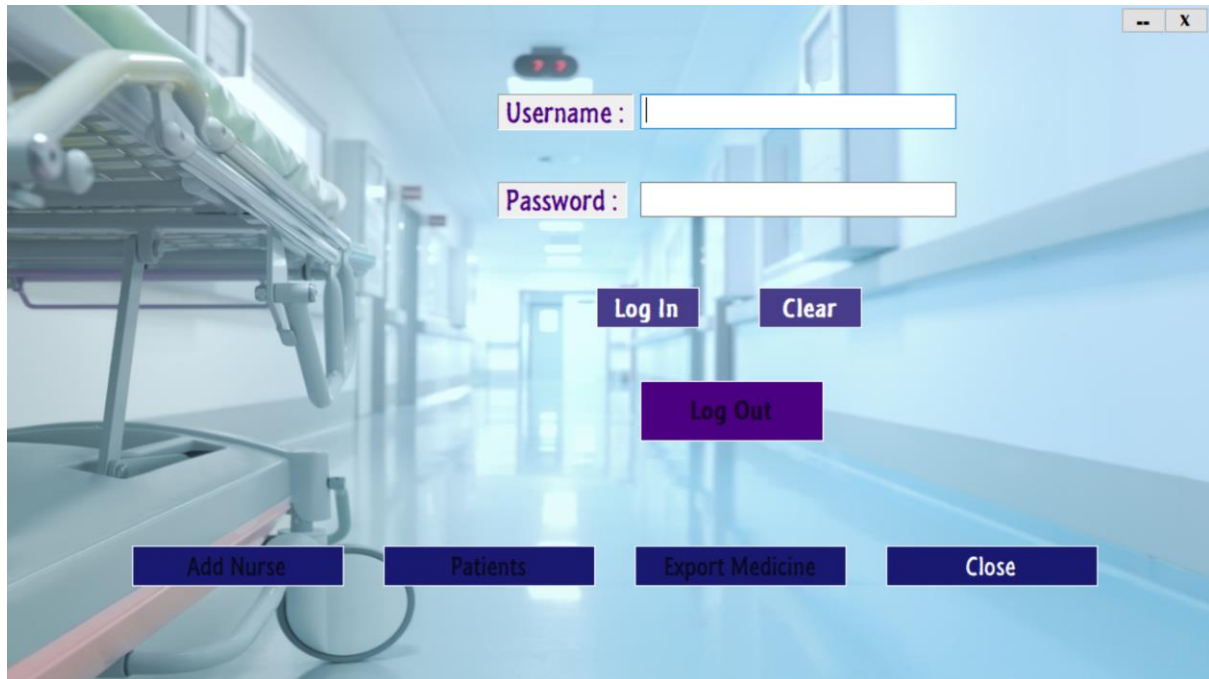
Doctor Ref No: 12

Date Time: 18-09-2019

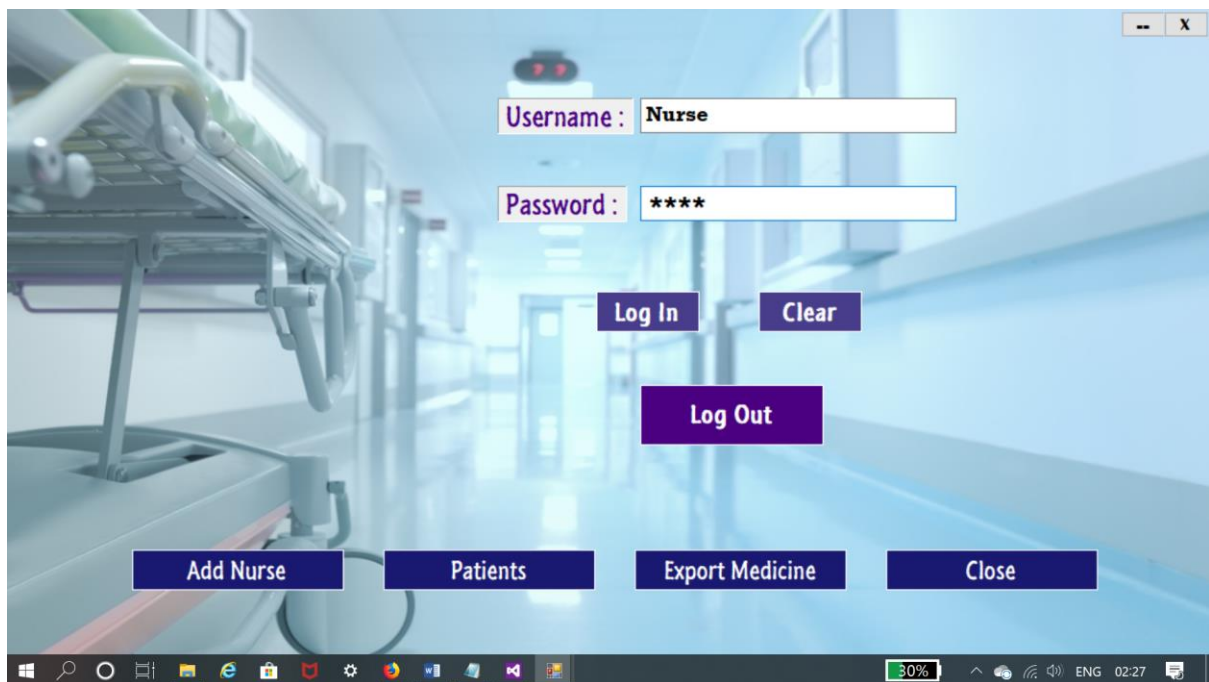
Appointment_Ref_No	First_Name	Middle_Name	Last_Name	E-mail_Address
1	ram	sham	bargode	a@gmail.com
2	raj	shaman	laman	a@gmail.com
3	prathamesh	vijay	gothankar	gotya@gmail.com
4	Aniket	Sunil	Dhulap	aniketdhulap@gmail.com
5	Ranjan	panjan	manjan	manjan@gmail.com
6	sas	ass	sddd	s@gmail.com
7	gsasadas	edasdda	sadea	a@gmail.com
8	jsadksdc	scscsdc	cscod	odd@gmail.com

Add Delete Next Previous Save Close Exit

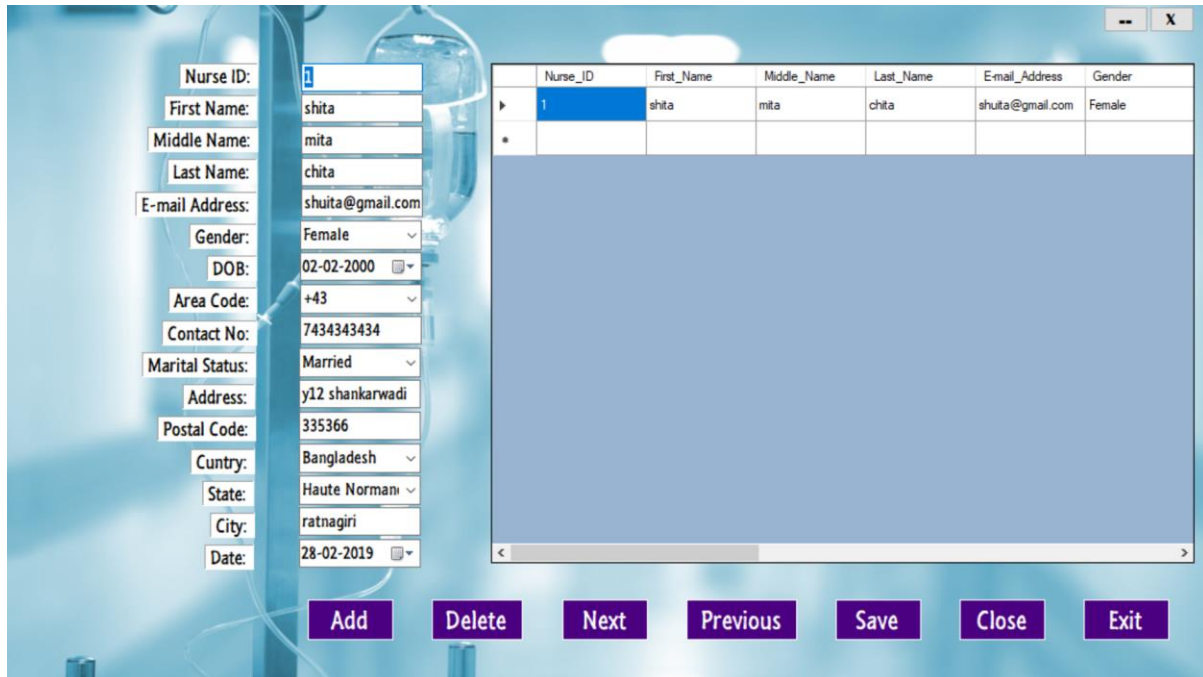
Nurse page:-



Nurse page After Login:-



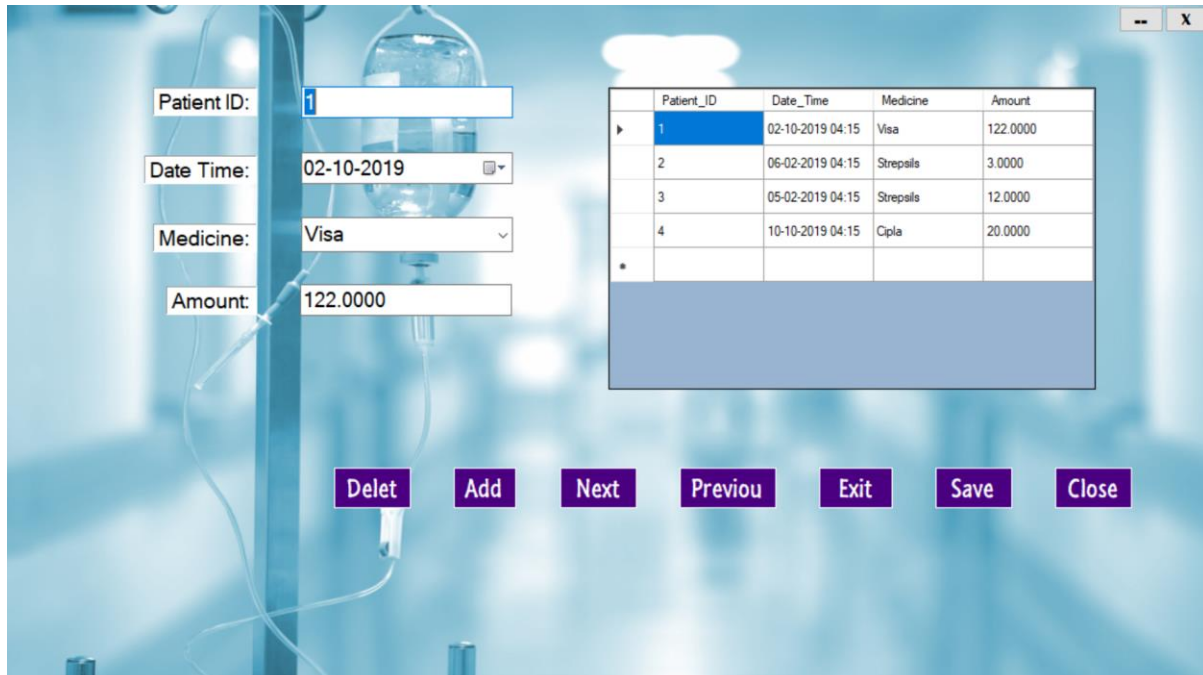
Nurse Profile Registration Form :-



The screenshot shows a Nurse Profile Registration Form with a light blue background featuring a medical drip. On the left, there are input fields for: Nurse ID (1), First Name (shita), Middle Name (mita), Last Name (chita), E-mail Address (shuita@gmail.com), Gender (Female), DOB (02-02-2000), Area Code (+43), Contact No (7434343434), Marital Status (Married), Address (y12 shankarwadi), Postal Code (335366), Country (Bangladesh), State (Haute Normandi), City (ratnagiri), and Date (28-02-2019). On the right, a table displays the entered data. At the bottom, there are buttons: Add, Delete, Next, Previous, Save, Close, and Exit.

Nurse_ID	First_Name	Middle_Name	Last_Name	E-mail_Address	Gender
1	shita	mita	chita	shuita@gmail.com	Female
*					

Export Medicine Form:-



The screenshot shows an Export Medicine Form with a light blue background featuring a medical drip. On the left, there are input fields for: Patient ID (1), Date Time (02-10-2019), Medicine (Visa), and Amount (122.0000). On the right, a table displays the entered data. At the bottom, there are buttons: Delet, Add, Next, Previou, Exit, Save, and Close.

Patient_ID	Date_Time	Medicine	Amount
1	02-10-2019 04:15	Visa	122.0000
2	06-02-2019 04:15	Strepsils	3.0000
3	05-02-2019 04:15	Strepsils	12.0000
4	10-10-2019 04:15	Cipla	20.0000
*			

Appointment Report Page:-

AptReport

SAP CRYSTAL REPORTS *

Main Report

Appointment Report 1

04-10-2019

AbtRefNo	FirstName	LastName	ContactNo	Dr.Name	Dr.RefNo	DateTime
1	ram	bargode	7878787878	aniket	12	18-09-2019 07
2	raj	laman	6767676767	ras	1	19-09-2019 07
3	gsgsg	sgsgsg	2535553533	babdbe	122	10-09-2019 07
4	Aniket	Dhulap	8787878787	Ashish	12	12-07-2019 07
5	Ranjan	manjan	2327323233	ramdase	34	12-09-2019 07
6	sas	sddd	2738126127	sgegd	7343	28-02-2019 07
7	gsasadas	sadsa	8443246234	dsfdfsdf	7323	05-06-2019 07

Current Page No.: 1 Total Page No.: 1 Zoom Factor: 100%

Patient Report Page:-

SAP CRYSTAL REPORTS *

Main Report

Patient Report 1

04-10-2019

Patient I	First Name	Last Name	Admin Date	Discharge Date
1	Aniket	Dhulap	05-06-2019 05:04:00	01-10-2019 05:04:00
2	raj	raman	03-06-2019 05:04:00	16-10-2019 05:04:00
3	Rajdas	chavan	05-06-2019 05:04:00	10-10-2019 05:04:00

Current Page No.: 1 Total Page No.: 1 Zoom Factor: 100%

Payment Page:-

**We Think....
About You.....**

Patient ID: 1
Date Time: 01-10-2019
Method of Payment: Master Card
Card Number: 1234
Pin Number: 7242
Ammount Paid: 332.0000

Patient_ID	Date_Time	Method_of_Paymer	Card_Number	Pin_Number	Ammount_Paid
1	01-10-2019 05:14	Master Card	1234	7242	332.0000
2	26-10-2019 05:14	Debit Card	2345	7878	676.0000
3	03-10-2019 05:14	Debit Card	6464	8474	89.0000
4	17-10-2019 05:14	Master Card	6464	6767	679.0000
5	18-10-2019 05:14	Debit Card	6767	8989	8576.0000
6	17-10-2019 05:14	visa	4342	4223	34234.0000
*					

Add Delet Next Previous Save Close Exit

Conclusion and Scope

System helps to store all the data about the hospital , patient , nurse , payment and doctors Details there is no need to do paper work . Data is going to be preserved carefully for longer period hence proper backup is required otherwise there is chance to losing data.

There are also chances that calculation error can occur which can cause huge problem.

Hospitals and healthcare centers have undergone a change for its betterment. The administrations of healthcare sector are opting IT solutions for the better management and patient care in their hospital campus. Have a look at some salient features of hospital management software.

Daily functions like patient registration, monitoring blood bank, managing admission and overall management of various departments can be easily performed with higher accuracy after the installation of [hospital software](#).

The modules of hospital management software are user-friendly and easy to access. It has a common user friendly interface having several modules. The officials can utilize these modules in their processes without any hassle and make the best possible use of hospital management system.

Since, every hospital has some or the other points of worth those vary in comparison with to its competitors. Hence, most of the IT companies give on-demand solutions or feature of customization. It further implicates that hospital information management software can be customized by specifying personal requirements of the campus.

This tool is a comprehensive solution that integrates all the departments by creating a common platform. In brief, hospital management system has all the modules that serve purpose of all the departments of healthcare institute. In fact, these modules have been competitively designed to make all the operations simplified.