**Assignment: Desktop Application**



Session: 2021 – 2024

**Submitted by:**

Bisma Muhammad Ali 2021-CS-170

**Supervised by:**

Madam Maida

Department of Computer Science

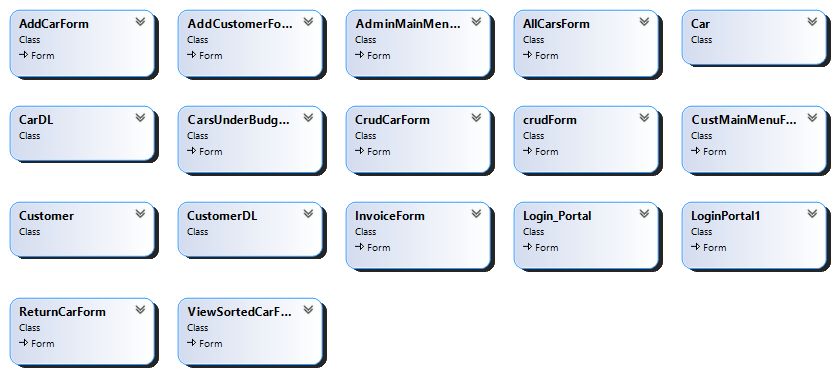
**University of Engineering and Technology**

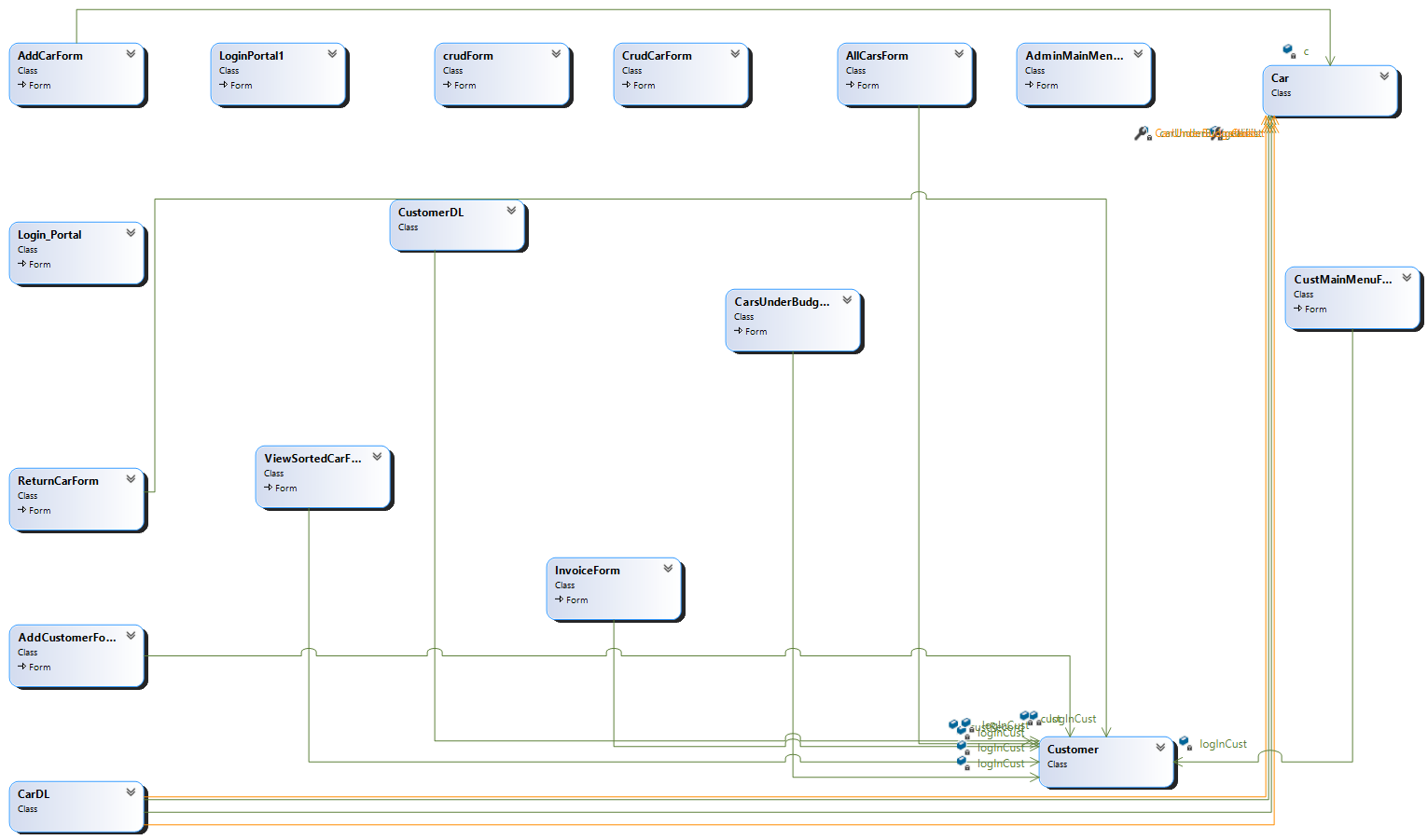
**Lahore Pakistan**

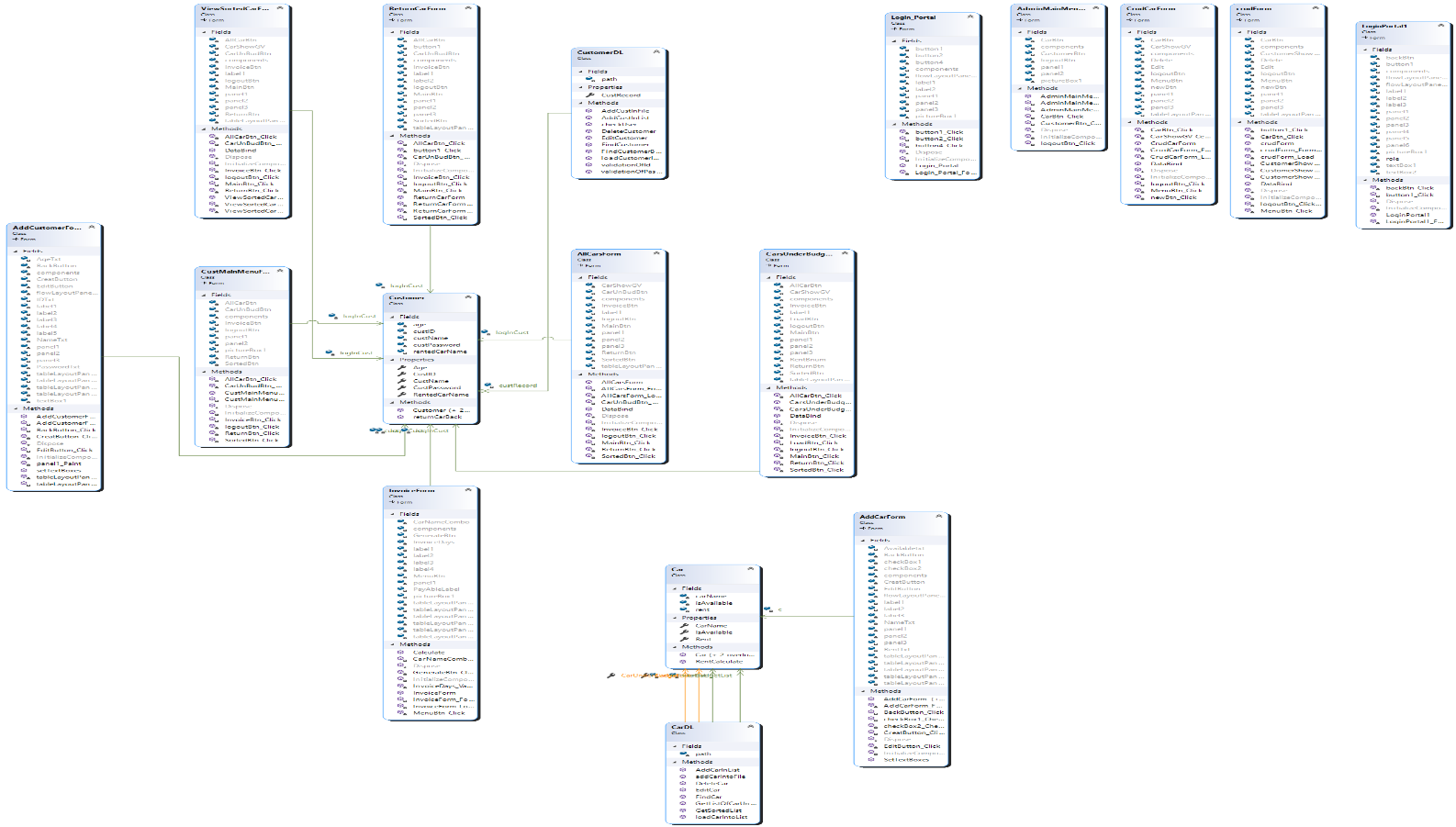
**Table Of Contents**

1. Domain Model …………………………………Pg No.3
2. Class Diagram…………………………………Pg No. 4
3. Sequence Diagram……………………………Pg No.5, 6
4. Wire Frames………………………………….…Pg No. 7- 12
5. Full Code………………………………….…Pg No. 13- 32

**Domain Model:**

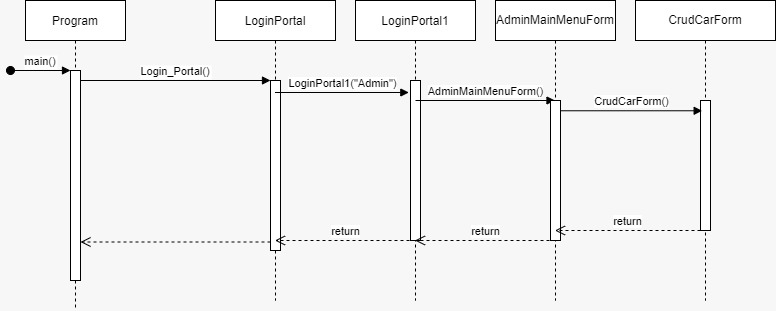
1. Domain Model with classes name.
2. Domain Model with relations and Constraints.



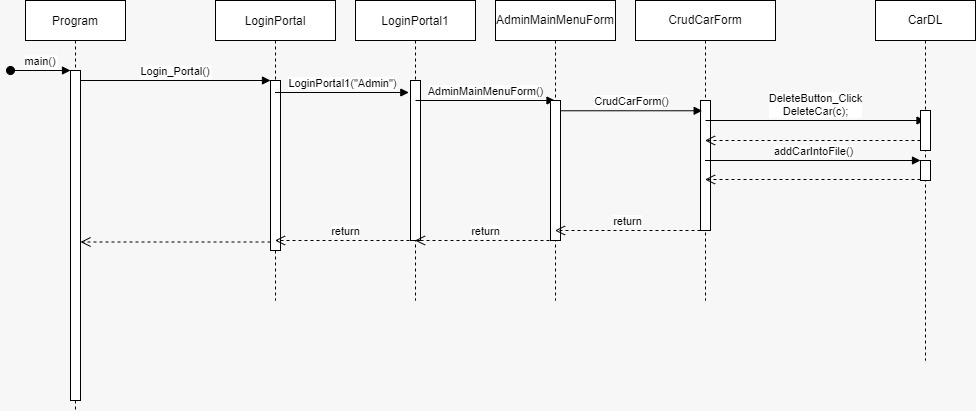
 **Class Diagram:**

**Sequence Diagrams:**

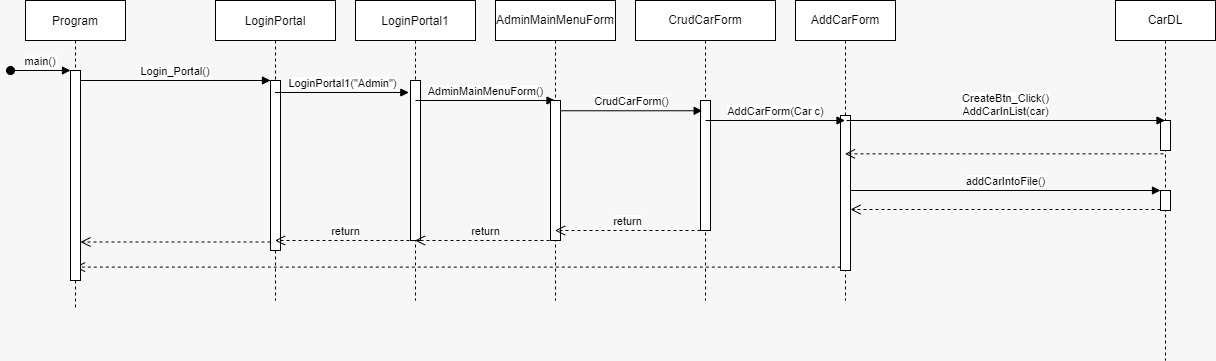
1. View Crud



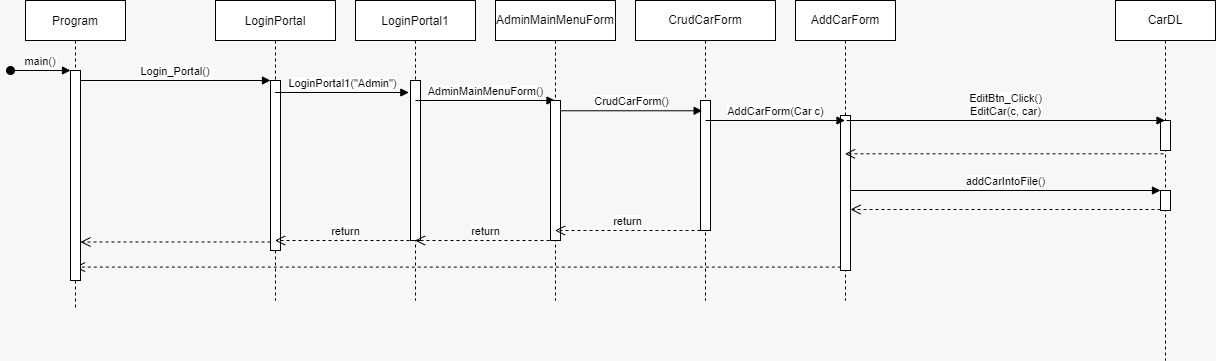
1. Delete Car



1. Add Car

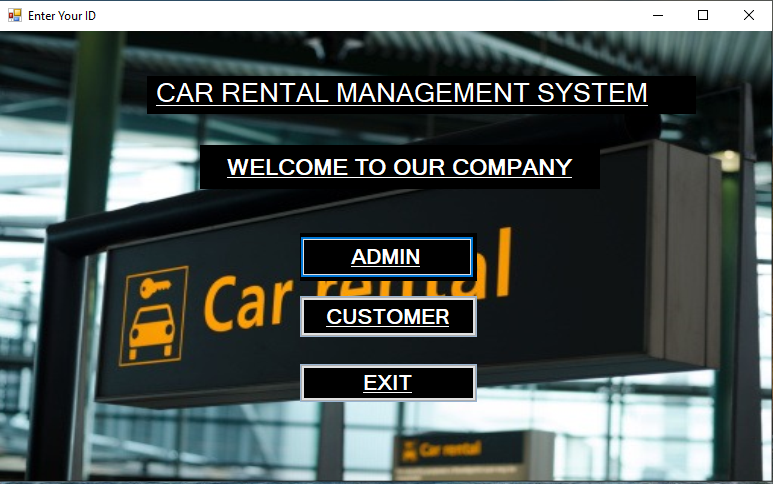


1. Edit Car

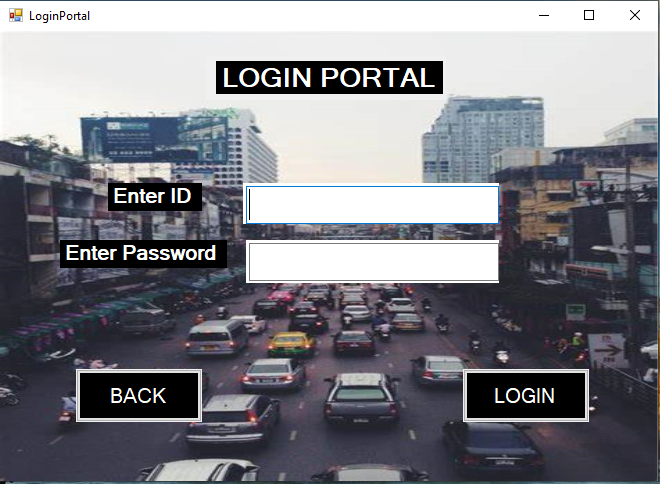


**Wire Frames:**

1. Status Form



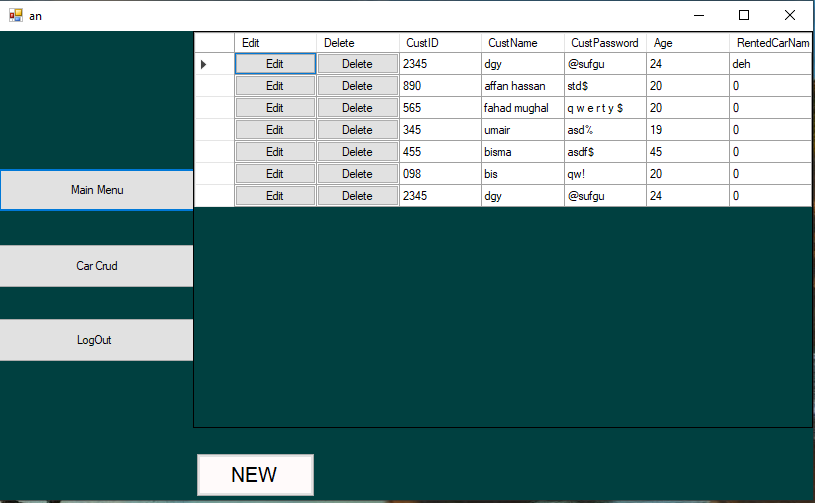
1. Login Portal



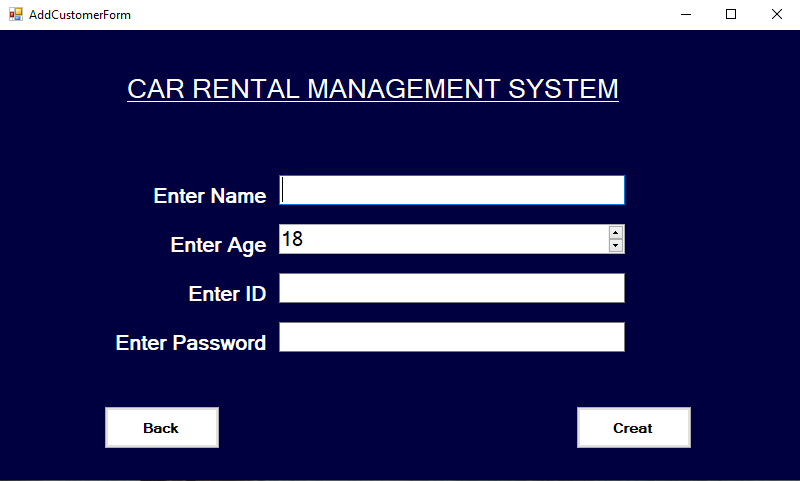
1. Admin’s Main Menu



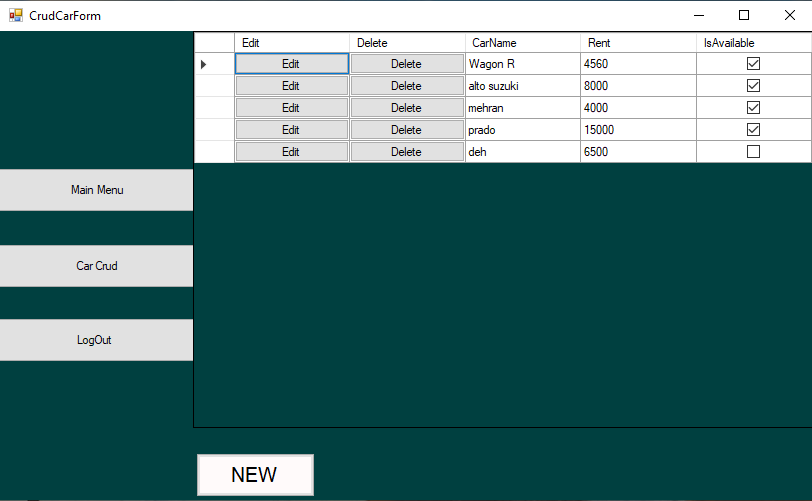
1. Customer’s CRUD



1. Add Customer Form



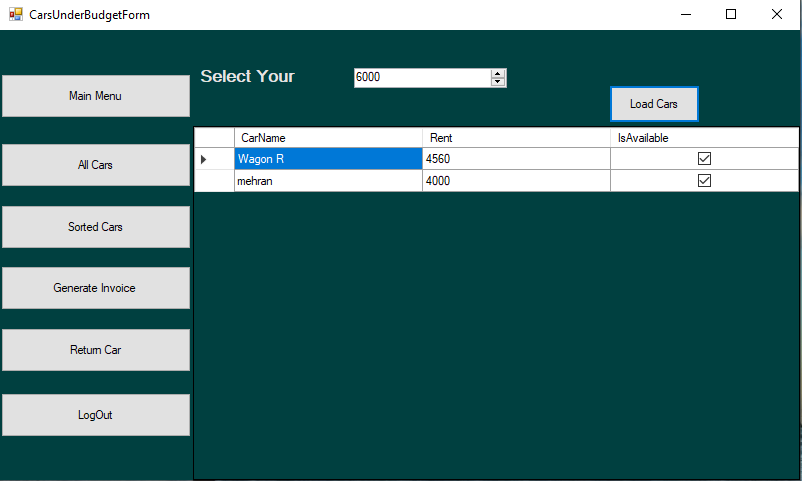
1. Car’s CRUD

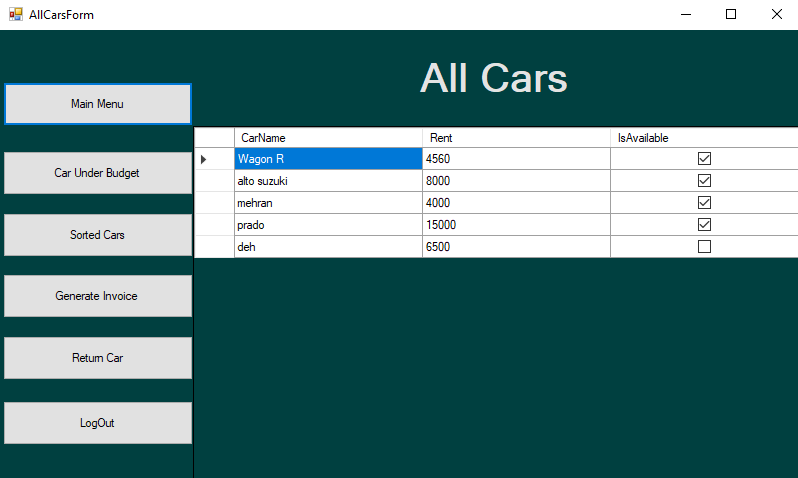
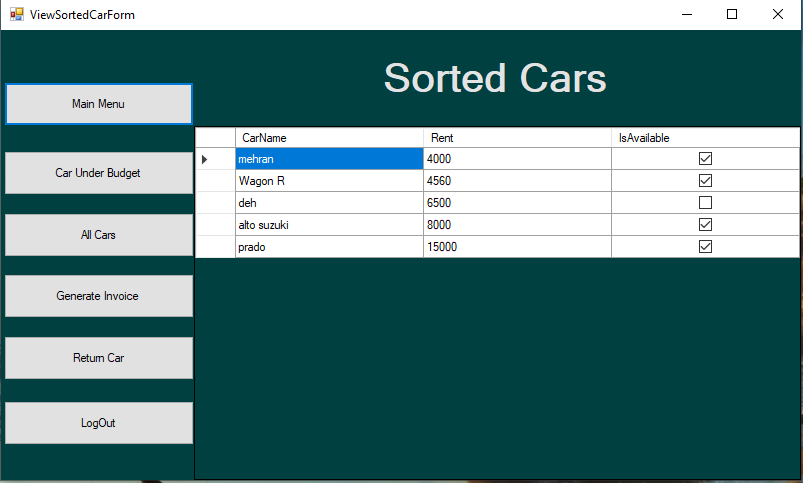
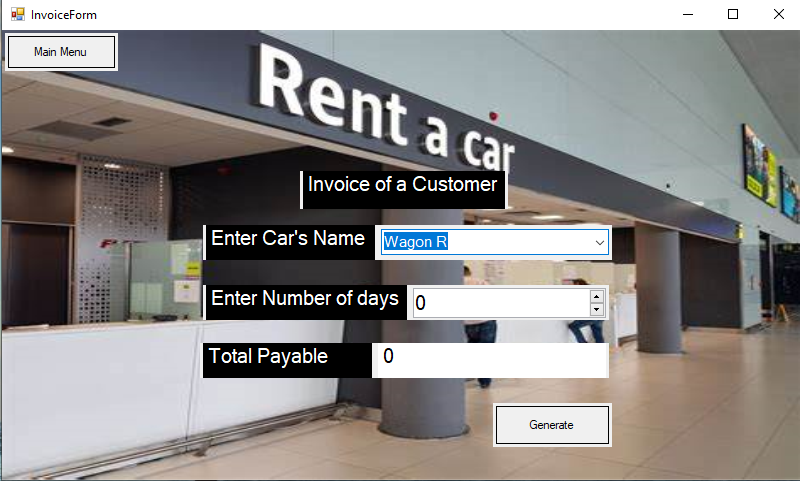
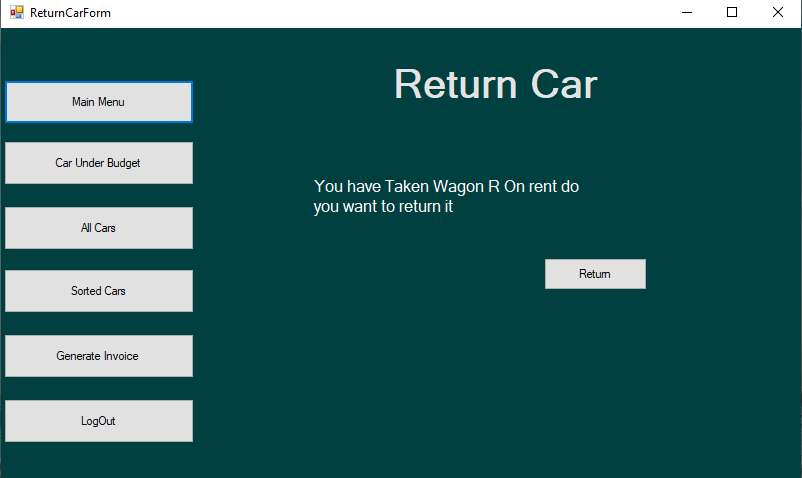


1. Customer’s Main Menu



1. Cars under budget



1. All cars Model
2. Sorted Cars
3. Generate Invoice
4. Return Car

**Full Code**

**BL Classes:**

1. **Customer**

// DEFAULT CONSTRUCTER

public Customer()

{

}

// PARAMETRIZED CONSTRUCTER

public Customer(string custID, string custName, string custPassword, int age)

{

this.CustID = custID;

this.CustName = custName;

this.CustPassword = custPassword;

this.Age = age;

RentedCarName = "0";

}

public Customer(string custID, string custPassword)

{

this.CustID = custID;

this.custPassword = custPassword;

}

// MEMBERS OF CLASS

private string custID;

private string custName;

private string custPassword;

private int age;

private string rentedCarName;

public string CustID { get => custID; set => custID = value; }

public string CustName { get => custName; set => custName = value; }

public string CustPassword { get => custPassword; set => custPassword = value; }

public int Age { get => age; set => age = value; }

public string RentedCarName { get => rentedCarName; set => rentedCarName = value; }

public void returnCarBack(List<Car> cars)

{

foreach (var c in cars)

{

if (rentedCarName == c.CarName)

{

c.IsAvailable = true;

rentedCarName = "0";

}

}

1. **Car**

// DEFAULT CONSTRUCTER

public Car()

{

}

// PARAMETRIZED CONSTRUCTER

public Car(string carName, int rent)

{

this.carName = carName;

this.rent = rent;

isAvailable = true;

}

public Car(string carName, int rent, bool isAvailable)

{

CarName = carName;

Rent = rent;

this.isAvailable = isAvailable;

}

private string carName;

private int rent;

private bool isAvailable;

//Getter Setters

public string CarName { get => carName; set => carName = value; }

public int Rent { get => rent; set => rent = value; }

public bool IsAvailable { get => isAvailable; set => isAvailable = value; }

public float RentCalculate(int day)

{

float rentalAmount;

rentalAmount = day \* rent;

if (day >= 5)

{

rentalAmount = rentalAmount - (rentalAmount \* 5 / 100);

}

return rentalAmount;

}

**DL Classes:**

1. **CustomerDL**

private static List<Customer> custRecord = new List<Customer>();

private static string path = "Customer.txt";

public static List<Customer> CustRecord { get => custRecord; set => custRecord = value; }

public static void EditCustomer(Customer c1, Customer c2)

{

foreach (Customer c in CustRecord)

{

if (c.CustID == c1.CustID)

{

c.CustName = c2.CustName;

c.Age = c2.Age;

c.CustID = c2.CustID;

c.CustPassword = c2.CustPassword;

}

}

}

public static void AddCustInList(Customer c)

{

CustRecord.Add(c);

}

public static bool DeleteCustomer(Customer c)

{

int idx = -1;

for (int i = 0; i < custRecord.Count; i++)

{

if (custRecord[i].CustID == c.CustID)

{

idx = i;

}

}

if (idx != -1)

{

custRecord.RemoveAt(idx);

return true;

}

return false;

}

public static Customer FindCustomerByName(string custName)

{

foreach (Customer cust in CustRecord)

{

if (custName == cust.CustName)

{

return cust;

}

}

return null;

}

public static Customer FindCustomer(Customer c)

{

foreach (Customer cust in CustRecord)

{

if (c.CustID == cust.CustID && c.CustPassword == cust.CustPassword)

{

return cust;

}

}

return null;

}

public static bool checkUser(string id, string password, ref int idx)

{

for (int i = 0; i < custRecord.Count; i++)

{

if (id == custRecord[i].CustID && password == custRecord[i].CustPassword)

{

idx = i;

return true;

}

}

idx = -1;

return false;

}

public static void AddCustInFile()

{

StreamWriter cust = new StreamWriter(path, false);

for (int i = 0; i < custRecord.Count; i++)

{

cust.WriteLine(custRecord[i].CustID + "," + custRecord[i].CustName + "," + custRecord[i].CustPassword + "," + CustRecord[i].Age + "," + custRecord[i].RentedCarName);

}

cust.Flush();

cust.Close();

}

public static void loadCustomerIntoList()

{

string word;

if (File.Exists(path))

{

StreamReader cust = new StreamReader(path);

while ((word = cust.ReadLine()) != null)

{

Customer c = new Customer();

string[] SplittedRecord = word.Split(',');

c.CustID = (SplittedRecord[0]);

c.CustName = SplittedRecord[1];

c.CustPassword = SplittedRecord[2];

c.Age = int.Parse(SplittedRecord[3]);

c.RentedCarName = SplittedRecord[4];

AddCustInList(c);

}

cust.Close();

}

}

public static bool validationOfId(string id)

{

foreach (Customer cust in custRecord)

{

if (id == cust.CustID)

{

return false;

}

}

for (int i = 0; i < id.Length; i++)

{

if ((id[i] >= 33 && id[i] <= 47) || (id[i] >= 58 && id[i] < 127))

{

return false;

}

}

return true;

}

public static bool validationOfPassword(string pass)

{

for (int x = 0; x < pass.Length; x++)

{

if (pass[x] == '@' || pass[x] == '!' || pass[x] == '#' || pass[x] == '%' || pass[x] == '^' || pass[x] == '&' || pass[x] == '\*' || pass[x] == '$')

return true;

}

return false;

}

1. **CarDL**

private static List<Car> carList = new List<Car>();

private static List<Car> carUnderBudgetList = new List<Car>();

private static string path = "Car.txt";

public static List<Car> CarList { get => carList; set => carList = value; }

public static List<Car> CarUnderBudgetList { get => carUnderBudgetList; set => carUnderBudgetList = value; }

public static List<Car> GetListOfCarUnderBudget(int rent)

{

carUnderBudgetList.Clear();

foreach (Car car in carList)

{

if (car.Rent <= rent)

{

carUnderBudgetList.Add(car);

}

}

return carUnderBudgetList;

}

public static List<Car> GetSortedList()

{

List<Car> sortedCarList = CarList.OrderBy(x => x.Rent).ToList();

return sortedCarList;

}

public static void AddCarInList(Car c)

{

carList.Add(c);

}

public static bool DeleteCar(Car c)

{

int idx = -1;

for (int i = 0; i < CarList.Count; i++)

{

if (carList[i].CarName == c.CarName)

{

idx = i;

}

}

if (idx != -1)

{

CarList.RemoveAt(idx);

return true;

}

return false;

}

public static void EditCar(Car c1, Car c2)

{

foreach (Car c in CarList)

{

if (c.CarName == c1.CarName)

{

c.CarName = c2.CarName;

c.Rent = c2.Rent;

c.IsAvailable = c2.IsAvailable;

}

}

}

public static Car FindCar(string CarName)

{

foreach (Car item in carList)

{

if (CarName == item.CarName)

{

return item;

}

}

return null;

}

public static void addCarIntoFile()

{

StreamWriter car = new StreamWriter(path, false);

for (int i = 0; i < carList.Count; i++)

{

car.WriteLine(carList[i].CarName + "," + carList[i].Rent + "," + carList[i].IsAvailable);

}

car.Flush();

car.Close();

}

public static void loadCarIntoList()

{

string word;

if (File.Exists(path))

{

StreamReader car = new StreamReader(path);

while ((word = car.ReadLine()) != null)

{

Car a = new Car();

string[] SplittedRecord = word.Split(',');

a.CarName = SplittedRecord[0];

a.Rent = int.Parse(SplittedRecord[1]);

a.IsAvailable = bool.Parse(SplittedRecord[2]);

AddCarInList(a);

}

car.Close();

}

**GUI Forms:**

1. **Admins form**

* **Add car form**

private Car c = null;

public AddCarForm(Car c)

{

InitializeComponent();

this.c = c;

CreatButton.Visible = false;

EditButton.Visible = true;

Availabletxt.Visible = true;

checkBox1.Visible = true;

checkBox2.Visible = true;

SetTextBoxes();

}

public AddCarForm()

{

InitializeComponent();

CreatButton.Visible = true;

EditButton.Visible = false;

Availabletxt.Visible = false;

checkBox1.Visible = false;

checkBox2.Visible = false;

}

public void SetTextBoxes()

{

NameTxt.Text = c.CarName;

RentTxt.Value = c.Rent;

if (c.IsAvailable)

{

checkBox1.Checked = true;

checkBox2.Checked = false;

}

else

{

checkBox1.Checked = false;

checkBox2.Checked = true;

}

}

private void EditButton\_Click(object sender, EventArgs e)

{

if (NameTxt.Text == "")

{

MessageBox.Show("Invalid Name");

}

else

{

string CarName = NameTxt.Text;

int Rent = int.Parse(RentTxt.Value.ToString());

bool IsAvailable;

if (checkBox1.Checked)

{

IsAvailable = true;

}

else

{

IsAvailable = false;

}

Car car = new Car(CarName, Rent, IsAvailable);

CarDL.EditCar(c, car);

CarDL.addCarIntoFile();

CrudCarForm carView = new CrudCarForm();

this.Hide();

carView.ShowDialog();

}

}

private void CreatButton\_Click(object sender, EventArgs e)

{

if (NameTxt.Text == "")

{

MessageBox.Show("Invalid Name");

}

else

{

string CarName = NameTxt.Text;

int Rent = int.Parse(RentTxt.Value.ToString());

Car car = new Car(CarName, Rent);

CarDL.AddCarInList(car);

CarDL.addCarIntoFile();

CrudCarForm carView = new CrudCarForm();

this.Hide();

carView.ShowDialog();

}

}

private void checkBox1\_CheckedChanged(object sender, EventArgs e)

{

if (checkBox1.Checked)

{

checkBox2.Checked = false;

}

else

{

checkBox2.Checked = true;

}

}

private void BackButton\_Click(object sender, EventArgs e)

{

CrudCarForm carView = new CrudCarForm();

this.Hide();

carView.ShowDialog();

}

private void AddCarForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

private void checkBox2\_CheckedChanged\_1(object sender, EventArgs e)

{

if (checkBox2.Checked)

{

checkBox1.Checked = false;

}

else

{

checkBox1.Checked = true;

}

}

* **Add Customer form**

private Customer cust = null;

public AddCustomerForm()

{

InitializeComponent();

EditButton.Visible = false;

CreatButton.Visible = true;

}

public AddCustomerForm(Customer cust)

{

InitializeComponent();

this.cust = cust;

setTextBoxes();

EditButton.Visible = true;

CreatButton.Visible = false;

}

public void setTextBoxes()

{

NameTxt.Text = cust.CustName;

IDTxt.Text = cust.CustID;

AgeTxt.Value = cust.Age;

PasswordTxt.Text = cust.CustPassword;

}

private void CreatButton\_Click(object sender, EventArgs e)

{

if (NameTxt.Text == "")

{

MessageBox.Show("Invalid Name");

}

else if (IDTxt.Text == "")

{

MessageBox.Show("Invalid ID");

}

else if (PasswordTxt.Text == "")

{

MessageBox.Show("Invalid Password");

}

else

{

string custName = NameTxt.Text;

string custID = IDTxt.Text;

int age = int.Parse(AgeTxt.Value.ToString());

if (CustomerDL.validationOfId(custID))

{

string custPassword = PasswordTxt.Text;

if (CustomerDL.validationOfPassword(custPassword))

{

Customer cust = new Customer(custID, custName, custPassword, age);

CustomerDL.AddCustInList(cust);

CustomerDL.AddCustInFile();

BackButton\_Click(sender, e);

}

else

{

MessageBox.Show("Password must have a special Character");

}

}

else

{

MessageBox.Show("Invalid ID");

}

}

}

private void BackButton\_Click(object sender, EventArgs e)

{

crudForm view = new crudForm();

this.Hide();

view.ShowDialog();

}

private void EditButton\_Click(object sender, EventArgs e)

{

if (NameTxt.Text == "")

{

MessageBox.Show("Invalid Name");

}

else if (IDTxt.Text == "")

{

MessageBox.Show("Invalid ID");

}

else if (PasswordTxt.Text == "")

{

MessageBox.Show("Invalid Password");

}

else

{

string custName = NameTxt.Text;

string custID = IDTxt.Text;

int age = int.Parse(AgeTxt.Value.ToString());

if (CustomerDL.validationOfId(custID))

{

string custPassword = PasswordTxt.Text;

if (CustomerDL.validationOfPassword(custPassword))

{

Customer c = new Customer(custID, custName, custPassword, age);

CustomerDL.EditCustomer(cust, c);

CustomerDL.AddCustInFile();

BackButton\_Click(sender, e);

}

else

{

MessageBox.Show("Password must have a special Character");

}

}

else

{

MessageBox.Show("Invalid ID");

}

}

}

private void AddCustomerForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

* **Add MainMenu form**

public AdminMainMenuForm()

{

InitializeComponent();

}

private void AdminMainMenuForm\_Load(object sender, EventArgs e)

{

}

private void logoutBtn\_Click(object sender, EventArgs e)

{

Login\_Portal logout = new Login\_Portal();

this.Hide();

logout.ShowDialog();

}

private void CustomerBtn\_Click(object sender, EventArgs e)

{

crudForm custView = new crudForm();

this.Hide();

custView.ShowDialog();

}

private void CarBtn\_Click(object sender, EventArgs e)

{

CrudCarForm carView = new CrudCarForm();

this.Hide();

carView.ShowDialog();

}

private void AdminMainMenuForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

* **crud form**

public crudForm()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

AddCustomerForm addCustomer = new AddCustomerForm();

this.Hide();

addCustomer.ShowDialog();

}

private void crudForm\_Load(object sender, EventArgs e)

{

DataBind();

}

private void CustomerShow\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

public void DataBind()

{

CustomerShowGV.DataSource = null;

CustomerShowGV.DataSource = CustomerDL.CustRecord;

// CustomerShowGV.Columns["custPassword"].Visible = false;

CustomerShowGV.Refresh();

}

private void CustomerShowGV\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

Customer cust = (Customer)CustomerShowGV.CurrentRow.DataBoundItem;

if (CustomerShowGV.Columns["Delete"].Index == e.ColumnIndex)

{

CustomerDL.DeleteCustomer(cust);

CustomerDL.AddCustInFile();

DataBind();

}

if (CustomerShowGV.Columns["Edit"].Index == e.ColumnIndex)

{

AddCustomerForm addCust = new AddCustomerForm(cust);

this.Hide();

addCust.ShowDialog();

}

}

private void logoutBtn\_Click\_1(object sender, EventArgs e)

{

Login\_Portal logout = new Login\_Portal();

this.Hide();

logout.ShowDialog();

}

private void CarBtn\_Click(object sender, EventArgs e)

{

CrudCarForm carView = new CrudCarForm();

this.Hide();

carView.ShowDialog();

}

private void MenuBtn\_Click(object sender, EventArgs e)

{

AdminMainMenuForm mainMenu = new AdminMainMenuForm();

this.Hide();

mainMenu.ShowDialog();

}

private void crudForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

* **car crud form**

public CrudCarForm()

{

InitializeComponent();

}

private void CrudCarForm\_Load(object sender, EventArgs e)

{

DataBind();

}

public void DataBind()

{

CarShowGV.DataSource = null;

CarShowGV.DataSource = CarDL.CarList;

CarShowGV.Refresh();

}

private void newBtn\_Click(object sender, EventArgs e)

{

AddCarForm carAdd = new AddCarForm();

this.Hide();

carAdd.ShowDialog();

}

private void CarShowGV\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

Car c = (Car)CarShowGV.CurrentRow.DataBoundItem;

if (CarShowGV.Columns["Delete"].Index == e.ColumnIndex)

{

CarDL.DeleteCar(c);

CarDL.addCarIntoFile();

DataBind();

}

if (CarShowGV.Columns["Edit"].Index == e.ColumnIndex)

{

AddCarForm carAdd = new AddCarForm(c);

this.Hide();

carAdd.ShowDialog();

}

}

private void logoutBtn\_Click(object sender, EventArgs e)

{

Login\_Portal logout = new Login\_Portal();

this.Hide();

logout.ShowDialog();

}

private void CarBtn\_Click(object sender, EventArgs e)

{

CrudCarForm carView = new CrudCarForm();

this.Hide();

carView.ShowDialog();

}

private void MenuBtn\_Click(object sender, EventArgs e)

{

AdminMainMenuForm mainMenu = new AdminMainMenuForm();

this.Hide();

mainMenu.ShowDialog();

}

private void CrudCarForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

1. **Customer’s form**

* **All car form**

private Customer logInCust = null;

public AllCarsForm(Customer logInCust)

{

InitializeComponent();

this.logInCust = logInCust;

}

private void AllCarsForm\_Load(object sender, EventArgs e)

{

DataBind();

}

public void DataBind()

{

CarShowGV.DataSource = null;

CarShowGV.DataSource = CarDL.CarList;

CarShowGV.Refresh();

}

private void logoutBtn\_Click(object sender, EventArgs e)

{

Login\_Portal logout = new Login\_Portal();

this.Hide();

logout.ShowDialog();

}

private void CarUnBudBtn\_Click(object sender, EventArgs e)

{

CarsUnderBudgetForm carsUnderBudget = new CarsUnderBudgetForm(logInCust);

this.Hide();

carsUnderBudget.ShowDialog();

}

private void SortedBtn\_Click(object sender, EventArgs e)

{

ViewSortedCarForm SortedCar = new ViewSortedCarForm(logInCust);

this.Hide();

SortedCar.ShowDialog();

}

private void InvoiceBtn\_Click(object sender, EventArgs e)

{

InvoiceForm invoice = new InvoiceForm(logInCust);

this.Hide();

invoice.ShowDialog();

}

private void ReturnBtn\_Click(object sender, EventArgs e)

{

ReturnCarForm returnC = new ReturnCarForm(logInCust);

this.Hide();

returnC.ShowDialog();

}

private void MainBtn\_Click(object sender, EventArgs e)

{

CustMainMenuForm mainPage = new CustMainMenuForm(logInCust);

this.Hide();

mainPage.ShowDialog();

}

private void AllCarsForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

* **Car under budget form**

private Customer logInCust = null;

public CarsUnderBudgetForm(Customer logInCust)

{

InitializeComponent();

CarShowGV.Visible = false;

this.logInCust = logInCust;

}

private void LoadBtn\_Click(object sender, EventArgs e)

{

CarShowGV.Visible = true;

DataBind();

}

public void DataBind()

{

CarShowGV.DataSource = null;

CarShowGV.DataSource = CarDL.GetListOfCarUnderBudget(int.Parse(RentBnum.Value.ToString()));

CarShowGV.Refresh();

}

private void AllCarBtn\_Click(object sender, EventArgs e)

{

AllCarsForm allCars = new AllCarsForm(logInCust);

this.Hide();

allCars.ShowDialog();

}

private void logoutBtn\_Click(object sender, EventArgs e)

{

Login\_Portal logout = new Login\_Portal();

this.Hide();

logout.ShowDialog();

}

private void SortedBtn\_Click(object sender, EventArgs e)

{

ViewSortedCarForm SortedCar = new ViewSortedCarForm(logInCust);

this.Hide();

SortedCar.ShowDialog();

}

private void InvoiceBtn\_Click(object sender, EventArgs e)

{

InvoiceForm invoice = new InvoiceForm(logInCust);

this.Hide();

invoice.ShowDialog();

}

private void ReturnBtn\_Click(object sender, EventArgs e)

{

ReturnCarForm returnC = new ReturnCarForm(logInCust);

this.Hide();

returnC.ShowDialog();

}

private void MainBtn\_Click(object sender, EventArgs e)

{

CustMainMenuForm mainPage = new CustMainMenuForm(logInCust);

this.Hide();

mainPage.ShowDialog();

}

private void CarsUnderBudgetForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

* **Customer’s main menu form**

private Customer logInCust = null;

public CustMainMenuForm(Customer logInCust)

{

InitializeComponent();

this.logInCust = logInCust;

}

private void CarUnBudBtn\_Click(object sender, EventArgs e)

{

CarsUnderBudgetForm carsUnderBudget = new CarsUnderBudgetForm(logInCust);

this.Hide();

carsUnderBudget.ShowDialog();

}

private void AllCarBtn\_Click(object sender, EventArgs e)

{

AllCarsForm allCars = new AllCarsForm(logInCust);

this.Hide();

allCars.ShowDialog();

}

private void SortedBtn\_Click(object sender, EventArgs e)

{

ViewSortedCarForm SortedCar = new ViewSortedCarForm(logInCust);

this.Hide();

SortedCar.ShowDialog();

}

private void InvoiceBtn\_Click(object sender, EventArgs e)

{

InvoiceForm invoice = new InvoiceForm(logInCust);

this.Hide();

invoice.ShowDialog();

}

private void ReturnBtn\_Click(object sender, EventArgs e)

{

ReturnCarForm returnC = new ReturnCarForm(logInCust);

this.Hide();

returnC.ShowDialog();

}

private void logoutBtn\_Click(object sender, EventArgs e)

{

Login\_Portal logout = new Login\_Portal();

this.Hide();

logout.ShowDialog();

}

private void CustMainMenuForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

* **Customer’s invoice form**

private Customer logInCust = null;

public InvoiceForm(Customer logInCust)

{

InitializeComponent();

this.logInCust = logInCust;

}

private void InvoiceForm\_Load(object sender, EventArgs e)

{

CarNameCombo.DataSource = CarDL.CarList;

CarNameCombo.DisplayMember = "CarName";

}

private void InvoiceForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

private void MenuBtn\_Click(object sender, EventArgs e)

{

CustMainMenuForm mainPage = new CustMainMenuForm(logInCust);

this.Hide();

mainPage.ShowDialog();

}

private void GenerateBtn\_Click(object sender, EventArgs e)

{

int days = int.Parse(InvoiceDays.Value.ToString());

if (days > 0)

{

if (logInCust.RentedCarName == "0")

{

string CarName = CarNameCombo.Text;

Car c = CarDL.FindCar(CarName);

if (c.IsAvailable)

{

c.IsAvailable = false;

logInCust.RentedCarName = CarName;

CarDL.addCarIntoFile();

CustomerDL.AddCustInFile();

CustMainMenuForm mainPage = new CustMainMenuForm(logInCust);

this.Hide();

mainPage.ShowDialog();

}

else

{

MessageBox.Show("Car Not Available");

}

}

else

{

MessageBox.Show("You already have a Car on Rent");

}

CarDL.addCarIntoFile();

CustomerDL.AddCustInFile();

}

}

private void InvoiceDays\_ValueChanged(object sender, EventArgs e)

{

Calculate();

}

public void Calculate()

{

string CarName = CarNameCombo.Text;

Car c = CarDL.FindCar(CarName);

int days = int.Parse(InvoiceDays.Value.ToString());

if (days != 0)

{

PayAbleLabel.Text = c.RentCalculate(days).ToString();

}

else

{

PayAbleLabel.Text = "0";

}

if (days >= 5)

{

MessageBox.Show("Days more than 5 or 5 will give you a discount of 5%");

}

}

private void CarNameCombo\_SelectedIndexChanged(object sender, EventArgs e)

{

Calculate();

}

* **Return car form**

private Customer logInCust = null;

public ReturnCarForm(Customer logInCust)

{

InitializeComponent();

this.logInCust = logInCust;

}

private void ReturnCarForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

private void MainBtn\_Click(object sender, EventArgs e)

{

CustMainMenuForm mainPage = new CustMainMenuForm(logInCust);

this.Hide();

mainPage.ShowDialog();

}

private void CarUnBudBtn\_Click(object sender, EventArgs e)

{

CarsUnderBudgetForm carsUnderBudget = new CarsUnderBudgetForm(logInCust);

this.Hide();

carsUnderBudget.ShowDialog();

}

private void AllCarBtn\_Click(object sender, EventArgs e)

{

AllCarsForm allCars = new AllCarsForm(logInCust);

this.Hide();

allCars.ShowDialog();

}

private void SortedBtn\_Click(object sender, EventArgs e)

{

ViewSortedCarForm SortedCar = new ViewSortedCarForm(logInCust);

this.Hide();

SortedCar.ShowDialog();

}

private void InvoiceBtn\_Click(object sender, EventArgs e)

{

InvoiceForm invoice = new InvoiceForm(logInCust);

this.Hide();

invoice.ShowDialog();

}

private void logoutBtn\_Click(object sender, EventArgs e)

{

Login\_Portal logout = new Login\_Portal();

this.Hide();

logout.ShowDialog();

}

private void ReturnCarForm\_Load(object sender, EventArgs e)

{

if (logInCust.RentedCarName != "0")

{

button1.Visible = true;

label2.Text = "You have Taken " + logInCust.RentedCarName + " On rent do you want to return it";

}

else

{

button1.Visible = false;

label2.Text = "No car have been taken by You";

}

}

private void button1\_Click(object sender, EventArgs e)

{

Car c = CarDL.FindCar(logInCust.RentedCarName);

c.IsAvailable = true;

logInCust.RentedCarName = "0";

CarDL.addCarIntoFile();

CustomerDL.AddCustInFile();

MessageBox.Show("Car Returned");

ReturnCarForm returnC = new ReturnCarForm(logInCust);

this.Hide();

returnC.ShowDialog();

}

* **View sort car form**

private Customer logInCust = null;

public ViewSortedCarForm(Customer logInCust)

{

InitializeComponent();

this.logInCust = logInCust;

}

private void ViewSortedCarForm\_Load(object sender, EventArgs e)

{

DataBind();

}

public void DataBind()

{

CarShowGV.DataSource = null;

CarShowGV.DataSource = CarDL.GetSortedList();

CarShowGV.Refresh();

}

private void logoutBtn\_Click(object sender, EventArgs e)

{

Login\_Portal logout = new Login\_Portal();

this.Hide();

logout.ShowDialog();

}

private void CarUnBudBtn\_Click(object sender, EventArgs e)

{

CarsUnderBudgetForm carsUnderBudget = new CarsUnderBudgetForm(logInCust);

this.Hide();

carsUnderBudget.ShowDialog();

}

private void AllCarBtn\_Click(object sender, EventArgs e)

{

AllCarsForm allCars = new AllCarsForm(logInCust);

this.Hide();

allCars.ShowDialog();

}

private void InvoiceBtn\_Click(object sender, EventArgs e)

{

InvoiceForm invoice = new InvoiceForm(logInCust);

this.Hide();

invoice.ShowDialog();

}

private void ReturnBtn\_Click(object sender, EventArgs e)

{

ReturnCarForm returnC = new ReturnCarForm(logInCust);

this.Hide();

returnC.ShowDialog();

}

private void MainBtn\_Click(object sender, EventArgs e)

{

CustMainMenuForm mainPage = new CustMainMenuForm(logInCust);

this.Hide();

mainPage.ShowDialog();

}

private void ViewSortedCarForm\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

1. **Login Portal Form**

private string role;

public LoginPortal1(string role)

{

InitializeComponent();

this.role = role;

}

private void button1\_Click(object sender, EventArgs e)

{

string ID = textBox1.Text;

string pass = textBox2.Text;

if (ID == "")

{

MessageBox.Show("Invalid ID");

}

else if (pass == "")

{

MessageBox.Show("Invalid Password");

}

else

{

if (role == "Admin")

{

if (ID == "123" && pass == "abc@")

{

AdminMainMenuForm mainMenu = new AdminMainMenuForm();

this.Hide();

mainMenu.ShowDialog();

}

else

{

MessageBox.Show("Invalid ID or Password");

}

}

else if (role == "Customer")

{

Customer cust = new Customer(ID, pass);

Customer logInCust = CustomerDL.FindCustomer(cust);

if (logInCust != null)

{

MessageBox.Show("Welcome");

CustMainMenuForm mainPage = new CustMainMenuForm(logInCust);

this.Hide();

mainPage.ShowDialog();

}

else

{

MessageBox.Show("Invalid Customer");

}

}

}

}

private void LoginPortal1\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

private void backBtn\_Click(object sender, EventArgs e)

{

Login\_Portal status = new Login\_Portal();

this.Hide();

status.ShowDialog();

}

1. **Status Form**

public Login\_Portal()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

LoginPortal1 login = new LoginPortal1("Admin");

this.Hide();

login.ShowDialog();

}

private void button2\_Click(object sender, EventArgs e)

{

LoginPortal1 loginC = new LoginPortal1("Customer");

this.Hide();

loginC.ShowDialog();

}

private void Login\_Portal\_FormClosing(object sender, FormClosingEventArgs e)

{

Application.Exit();

}

private void button4\_Click(object sender, EventArgs e)

{

MessageBox.Show("Thank you for using our Application");

Application.Exit();

}