

APPLICATIONS DEVELOPMENT 2A

WORKBOOK 2024

Table of Contents

Tutorial 1

Tutorial 2

Tutorial 3

Basic MVC ASP.NET applications with ASP controls

These applications listed below is uses the basic structure of an MVC application. These applications make use of the Model (properties and methods), View (ASP controls: input, output, buttons), Controller (calling of methods from the model).

All tutorials must have a full MVC application that includes the following:

- a) A full model code for this application.
- b) A full Controller code.
- c) A full code for the view layout

Tutorial 1

- a) What is ASP.NET MVC
- b) Explain the role of a Controller in an ASP.NET MVC application.
- c) What is the purpose of a View in an ASP.NET MVC application?
- d) What is Separation of Concerns in ASP.NET MVC?
- e) In your own words, elaborate what you understand about ASP.NET MVC naming convention.
- f) Briefly describe the ASP.NET MVC routing system.
- g) Where are routes registered in ASP.NET MVC Application?
- h) What is the purpose of a Razor View Engine in ASP.NET MVC application?
- i) In what situation would it be appropriate to use a ViewBag?
- j) Create an ASP.NET MVC5 application: Demonstrate how you would create a Controller class called *Employees* which contains at least three appropriate action methods. Furthermore, you must take advantage of a ViewBag object to pass appropriate employee information from your Controller to a relevant View.
- k) Demonstrate how you would create Views which are associated with your above defined action methods.

Tutorial 2

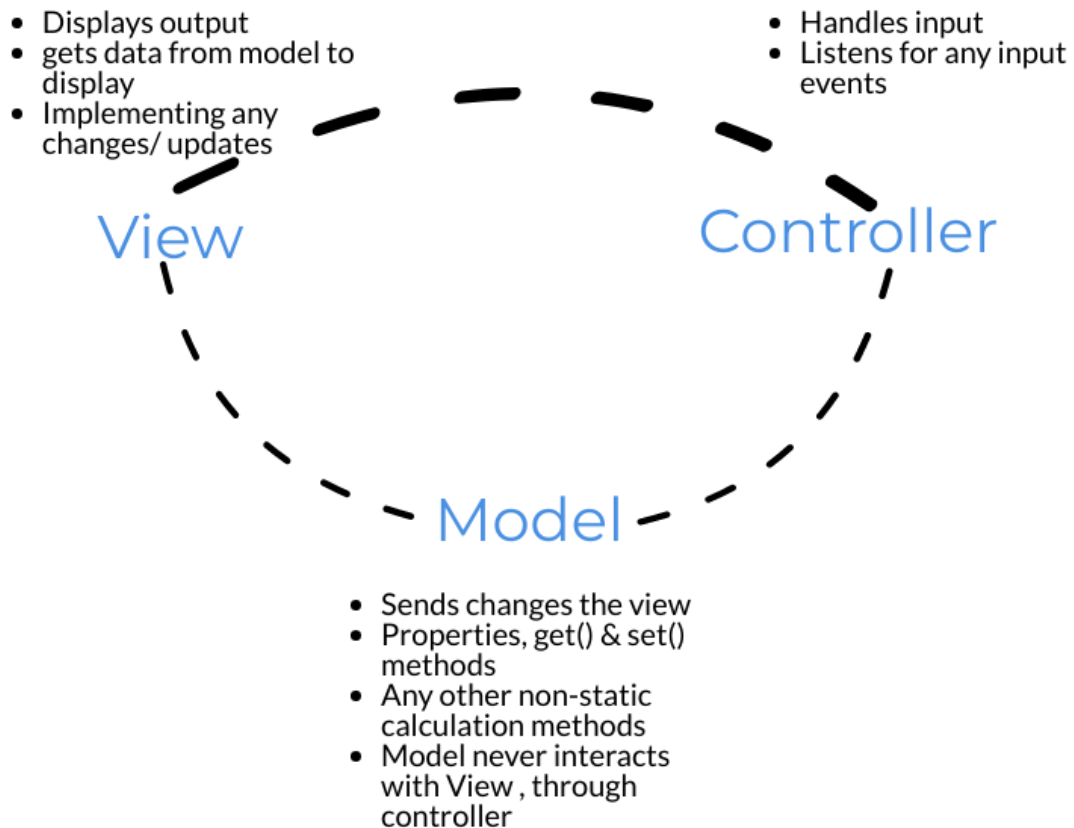
- a) What is the purpose of a View in an ASP.NET MVC application?
- b) Explain the role of a Controller in an ASP.NET MVC5/6 application.
- c) In your own words, elaborate what you understand about ASP.NET MVC naming convention.
- d) Briefly describe the ASP.NET MVC routing system.
- e) What is the purpose of a Razor View Engine in ASP.NET MVC5/6 application?, give some advantages.
- f) In what situation would it be appropriate to use a ViewBag?

Note:

In 2nd year we start designing our web applications using the MVC pattern. The MVC application is made up of the Model, which all of you would remember as the class with property structures and methods that return a value. In the model we create the property structures and the methods. Almost all our non-static methods within the model returns a value.

The next part in the MVC application is the controller, which we code to call the methods and properties from our model. You will see in the sample programs within this document that the controller is the most important part of any application as it controls the method calls and sends the results to the View. As seen below is the image explaining the MVC cycle and how each of the 3 components are crucial to the execution of the application.

Applications Development 2A



Apda201 - 2021

The code written in the model and controller is C#. The code we write to render a view that is formatted is HTML and RAZOR syntax. The HTML and RAZOR code can be researched using the www.w3schools.com web resource to learn more about the syntax and code for rendering a perfectly formatted view.

The most commonly used HTML used is

<div> ... </div> - div tag

<p> ... </p> - paragraph tag

 - line break

<h1>...</h1> - heading 1 tag

<input type= "Submit" value"Calculate"> - creates a button used for submission

RAZOR syntax used

@HTML.EditorFor(..) – creates an input field similar to the textbox and textarea.

@HTML.DisplayFor(...) – creates an output field similar to a label and listbox.

@HTML.RadioButtonFor(...) – creates radio buttons fields, where one has to be selected.

@HTML.CheckBoxFor(...) – creates a checkbox field, where you can choose to select the checkbox or not. This field can be null.

@HTML.DropDownListFor(...) – creates a dropdownlist of options to select from.

Tutorial 3

a) An “Action method” must always be public. True/False? Support your answer.

b) Label and state the significance for the underlined code below:

[HttpGet]

public ActionResult Create()

{

// create a student record

return View();

}

Complete the table below and provide a C# code that suits this request. Demonstrate the corresponding controller, action and Id. URL	Controller	Action	Id
http://localhost/categories/fruits/3			
http://localhost/Product/items/2			

Create your asp.net MVC app.

1. Open Visual Studio, click on create a new project

2. The screenshot below shows the different project types. Please select C# on the language menu on top and select web under the Project type menu. The red circles show this.

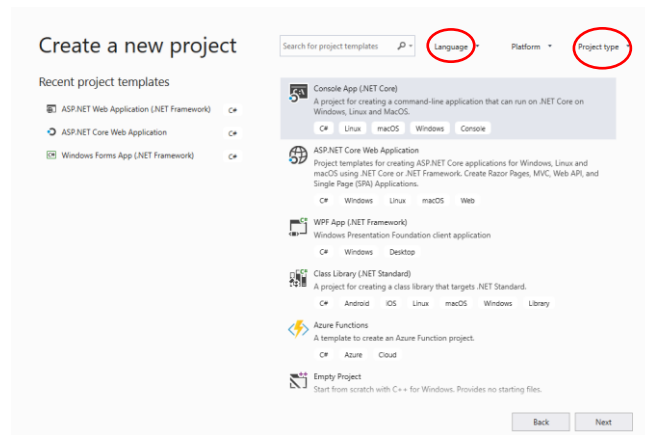


Fig2

3. This screenshot is the next screen. Please select the second option as seen in Fig3 and click next.

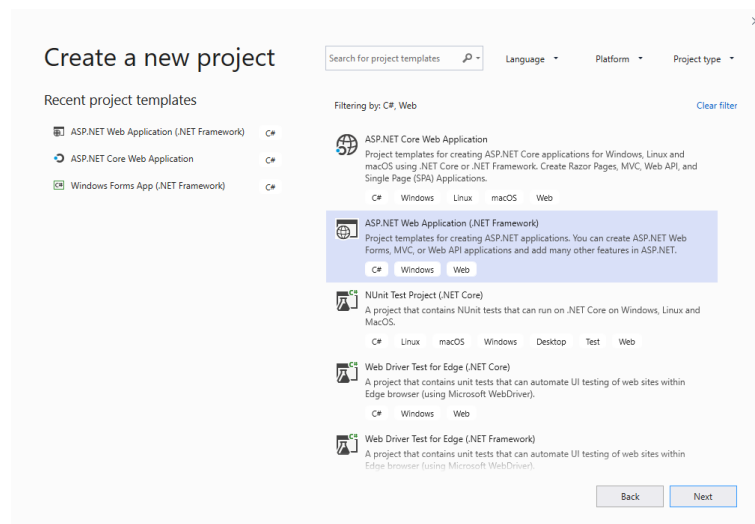


Fig3

4. The screenshot below is shown next. Change the Project name, location and click on create.

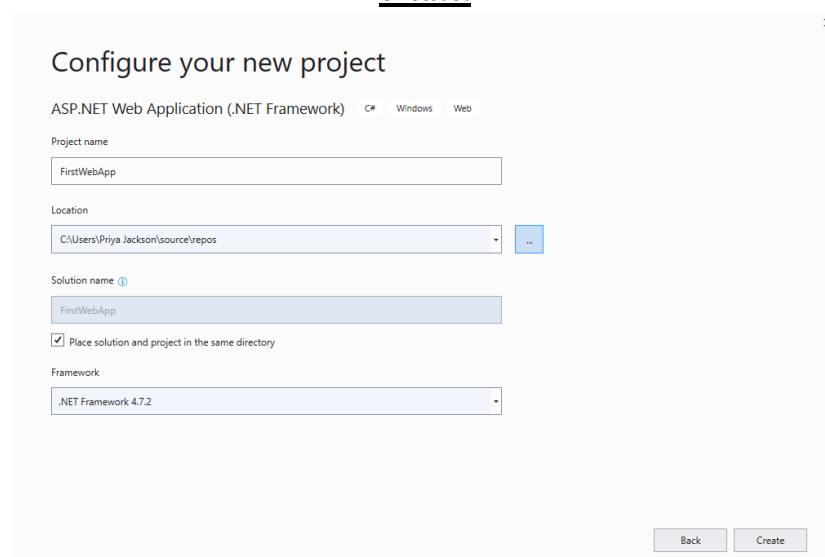


fig4

5. Select MVC type and click on create and your application is created with default controller called Home, which is located in the controller folder. There are 3 views created, which is located in the Views folder, Home .

Create a new ASP.NET Web Application

Empty
An empty project template for creating ASP.NET applications. This template does not have any content in it.

Web Forms
A project template for creating ASP.NET Web Forms applications. ASP.NET Web Forms lets you build dynamic websites using a familiar drag-and-drop, event-driven model. A design surface and hundreds of controls and components let you rapidly build sophisticated, powerful UI-driven sites with data access.

MVC
A project template for creating ASP.NET MVC applications. ASP.NET MVC allows you to build applications using the Model-View-Controller architecture. ASP.NET MVC includes many features that enable fast, test-driven development for creating applications that use the latest standards.

Web API
A project template for creating RESTful HTTP services that can reach a broad range of clients including browsers and mobile devices.

Single Page Application
A project template for creating rich client side JavaScript driven HTML5 applications using ASP.NET Web API. Single Page Applications provide a rich user experience which includes client-side interactions using HTML5, CSS3, and JavaScript.

Authentication
No Authentication
[Change](#)

Add folders & core references
☐ Web Forms
☒ MVC
☐ Web API

Advanced
☐ Docker support
(Requires [Docker Desktop](#))
☐ Also create a project for unit tests
☒ WebApplication1.Tests

[Back](#) [Create](#)

Fig5

First Web App to see the link between controller and view

Code in Controller-This code is already generated with every MVC application. No adding of any code.

```
namespace FirstWebApp.Controllers
{
    public class MainController : Controller
    {
        public ActionResult Index()
        {
            return View();
        }

        public ActionResult About()
        {
            ViewBag.Message = "Your application description page.";

            return View();
        }

        public ActionResult Contact()
        {
            ViewBag.Message = "Your contact page.";

            return View();
        }
    }
}
```

Go to the Views folder, Main folder and find Index.cshtml. Delete the code on this file. Then add the code below onto your Index.cshtml file and run the app. You will see that it displays the text written below.

```
@{
    ViewBag.Title = "Index Page";
}
<h2> Applications Development 2</h2>
```

<p>This module will look at the following content during this semester.</p>

<p>

Introduction to MVC

Using Razor syntax

Using HTML

</p>

Now add the following code to the Index.cshtml, then you see the function of an ActionLink html helper. The actionlink allows you to move from one view to the next.

<div>

Text shown on screen	Action method name	Controller Name
----------------------	--------------------	-----------------

@Html.ActionLink("About Page", "About", "Main")

@Html.ActionLink("Contact Page", "Contact", "Main")
</div>

Delete the code on the Contact.cshtml, which can be found in the same location as the Index.cshtml. Add the code below to your Contact.cshtml.

@{
 ViewBag.Title = "Contact";
}
<h2>@ViewBag.Title.</h2>
<h3>@ViewBag.Message</h3>

<address>
 Ritson Rd Campus

 Berea, KZN

 <abbr title="Phone">P:</abbr>
 4000
</address>
<p>
 @Html.ActionLink("Back to Home", "Index")
</p>

Add this code to the About.cshtml, which can be found where the Index.cshtml file is located.

@{
 ViewBag.Title = "About";
}
<h2>@ViewBag.Title.</h2>
<h3>@ViewBag.Message</h3>
<h3>Applications Development 2 is a very challenging module and requires students to critically solve a problem</h3>
<div>
 @Html.ActionLink("Back to Index", "Index")
</div>

Run the above application in visual studio and test the movement from Index View to Contact or About View.

MVC Program with code for the Model, View and Controller –Follow steps above to create new MVC application.

Example 1- Program calculates the product of two numbers and displays the answer.

The program inputs two numbers and calculates the product of the two numbers and outputs the product.

- Create a model with all the properties and a method to calculate the product of these two numbers.
- Create a controller to call all the properties and methods and return the results to the view.
- Write the code to render the view in Fig 1

Application name Home About Contact

DisplayProduct

Enter Number 1:

Enter Number 2:

Product is: 120

© 2022 - My ASP.NET Application

Fig6

Go to the solution explorer and right click on Models folder. Select Add, class and a window comes up like Fig7 below. Type the name “Numbers” for your class.

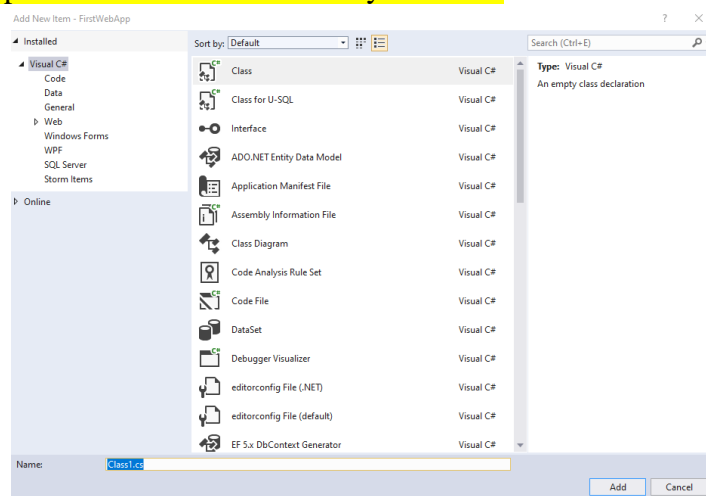


Fig7

Add the c# code listed in the model below and save your work.

//Code for the Model

```
public class Numbers
```

```
{ //Model code
```

```
    //Properties
```

```
    public int Num1 { get; set; }
```

```
    public int Num2 { get; set; }
```

```
    public int ProductAns { get; set; }
```

```
    //Methods
```

```
    public int calcProduct()
```

```
{
```



```

        return (Num1 * Num2);
    }
}
//End of Model code

```

Now, go the controller's folder in the solutions explorer located on the right hand side of the application screen. If you cannot find it, go to View menu and click on solutions explorer. Click on HomeController file and then type the code in red into your controller file.

```

//Code for the Controller
public class HomeController : Controller
{
    .....//The Index, About and Contact action methods
    public ActionResult DisplayProduct()
    {
        return View();
    }
    [HttpPost]
    public ActionResult DisplayProduct(Numbers tnumber)
    {
        tnumber.Num1 = tnumber.Num1;
        tnumber.Num2 = tnumber.Num2;
        tnumber.ProductAns = tnumber.calcProduct();
        return View(tnumber);
    }
}

```

Right click on the action method DisplayProduct which is located in the Home Controller. Select add View and click on create. The name and other details are already pre filled in.

A view with the name DisplayProduct.cshtml will be created under the Views folder and Home folder. Use the code below and add it to the view called DisplayProduct.cshtmlCode for the View

```

@model WebAppMultiply.Models.Numbers
@{
    ViewBag.Title = "DisplayProduct";
}

<h2>DisplayProduct</h2>

    @using (Html.BeginForm("DisplayProduct", "Home", FormMethod.Post))
    {
        <div>Enter Number 1: @Html.EditorFor(n1=>n1.Num1)</div>
        <div>Enter Number 2: @Html.EditorFor(n2=>n2.Num2)</div>
        <div>Answer is: @Html.DisplayFor(prod=>prod.ProductAns)</div>

        <div><input type="Submit" value="Calculate" /></div>
    }

```

Go to the AppStart folder and open the RouteConfig file. Change the action name to DisplayProduct and run your application. This file has the URL for your application

Tutorial 4

In this question, you are required to write a C# MVC application whose purpose is to calculate the total amount to be paid by a customer after discount. The discount system used by this program is described by Table 1. The input to the program is the unit price per item (before discount) and the quantity of each item purchased by the customer.

Total Amount Before Discount	Discount Rate or Percentage
Less than 200	No discount
Between R200 and R300	2%
Higher than R300	3%

Table 1. Discount system

Write the C# code to accomplish the following sequence of tasks.

- create a model with the appropriate properties.
- create a method to calculate the total amount to be paid by the customer before discount
- create a method to calculate the total discount amount.
- create a method to calculate the total amount to be paid by the customer after discount has been included.
- You are required to write C# code for this controller in order to demonstrate how the controller facilitates inputs from the view.
- Write the HTML and RAZOR lines to create the view for this app.

Write an MVC program to display a message depending on the value of a given number. Request a number from the user. Check the number – if the number is.... Less than zero, display the message “Number is a negative number”. Greater than or equal to zero but less than 100, display the message “Number is a positive number less than 100”. Greater than or equal to 100, display the message “Number is greater than or equal to 100”.

Tutorial 5 – Employee Wages

Write a program that will accept an employee’s name, number, hours worked and rate of pay. Calculate the employee’s wages using the hours worked and rate of pay. If the hours worked is less than or equal to 40, the employees works normal hours and get a basic wages. However if, the employee works more than 40 hours, then he get an overtime pay which is time and a half for every hour overtime hour. Calculate the overtime wages, normal wages and total wages as well as the overtime salary.

Create an ASP.NET MVC full application that will:

Input the employee name, employee gender (male/female), employee type (fulltime/parttime), hours worked and rate of pay. Calculate employee grosswage (overtime and normal pay), tax amount, further deduction and Net Wages.

For part time employees, tax must be 10% of gross and for full time employees the tax must be 15% of gross. Further deduction of 5% of gross must be deducted if the employee is female. Consider overtime hours if any, these hours are paid at time and a half of the rate.

Output the following:

- Employee Name

- Employee Gender
- Employee Type (Full Time or Part time)
- HoursWorked
- Normal Pay
- Over time pay
- Gross Pay
- Tax Amount
- Net Pay

Tutorial 6

A small car wash business offers seven (7) services to its customers as indicated by the table below. Calculate the total cost for the service with the VAT included.

Car Wash Number	Service Price
0 - Basic Wash	30
1 – Full Wash	40
2 - Full House	55
3 – Wash, Polish and clean	60

Table 1 Services offered by a car wash business

Tutorial 7

A Local TV shop specializes in delivering two types of TV sets to its customers: High definition TVs, and flat screen TVs. This shop would like you to write a C# MVC application to perform the following tasks every time that it has to make a delivery to its customers, assuming that one customer cannot have more than one TV in a given delivery:

- Input the TV set type, price of the tv, number of TV sets, delivery address, number of kilometres to deliver TV sets.
- Delivery Fee is R10 per kilometre.
- Calculate and display on the screen the amount to be paid by each customer.

Tutorial 8

The Shuttle Car Company has requested that you write a C# MVC application that accepts the name of the customer, the number of days the car is hired and the rental price per day. Calculate the amount due by a customer when the car is hired. The amount due is based on the rental price per day of the car and the number of days the car is hired. VAT must be included in the total amount due.

Write the C# code to accomplish the following sequence of tasks.

- a) create a model with the appropriate properties.
- b) create a method to calculate the total amount due by the customer before VAT has been considered.
- c) create a method to calculate the VAT amount
- d) create a method to calculate the total amount to be paid by the customer after VAT has been included.

- e) You are required to write C# code for this controller in order to demonstrate how the controller facilitates inputs from the view.
- f) Write the HTML and RAZOR lines to create the view for this app.

Example - RadioButtons & Checkboxes

Model Code

```
public class Wage
{
    public string empNum { get; set; }
    public string empType { get; set; }
    public int hoursWorked { get; set; }
    public double rate { get; set; }
    public double empWage { get; set; }
    public double empTax { get; set; }
    public double empNettWage { get; set; }
    public double empBonus { get; set; }
    public SkillsType empSkillsType { get; set; }

    //Calculate wages for employee working less than 40 hour. Calculate
    //wages for employee working overtime which is any number of hours over 40

    public double calcGrossWage()
    {
        double normalpay = 0, overtimepay=0;

        if (hoursWorked <= 40)
        {
            normalpay=(rate * hoursWorked);
        }
        else
        {
            normalpay = rate * 40;
            overtimepay= (hoursWorked - 40) * rate * 1.5;
        }

        return (normalpay + overtimepay);
    }

    //Calculate tax amount for employee based on their type of employment
    //If part-time, tax is 15%. If full time, tax is 40%

    public double calcEmpTax()
    { double taxAmt = 0;
      if(empType=="Part-time")
      {
          taxAmt = 0.15 * calcGrossWage();
      }
      else if(empType=="Full time")
```

```

        {
            taxAmt = 0.40 * calcGrossWage();
        }
        return (taxAmt);
    }

    public double calcNettWage()
    {
        return ((calcGrossWage()) - calcEmpTax());
    }

}

```

Code for the controller

```

public class EmployeeWageController:Controller
{
    public ActionResult DisplayWage()
    {
        return View();
    }
    [HttpPost]
    public ActionResult DisplayWage (Wage wg)
    {
        wg.empWage = wg.calcGrossWage();
        wg.empTax = wg.calcEmpTax();
        wg.empNettWage = wg.calcNettWage();

        return View(wg);
    }
}

```

Code for the View

```

@model WebAppWage.Models.Wage
@{
    ViewBag.Title = "DisplayWage";
}

```

```

<h2>DisplayWage</h2>

```

```

@using (Html.BeginForm("DisplayWage", "EmployeeWage", FormMethod.Post))
{<p>
    Emp Number: @Html.EditorFor(e => e.empNum)
    <br />
    Hours Worked: @Html.EditorFor(h => h.hoursWorked)
    <br /> Rate: @Html.EditorFor(r => r.rate)
    <br />Employment Type
    <br />@Html.RadioButtonFor(a => a.empType, "Part-time") Part-Time
    <br />@Html.RadioButtonFor(a => a.empType, "Full time") Full Time

    <br /> <input type="submit" value="Calc" />

```

```
<br /> Employee Wage: @Html.DisplayFor(w => w.empWage)
<br /> Employee Tax: @Html.DisplayFor(s => s.empTax)
<br /> Employee Nett Wage: @Html.DisplayFor(c => c.empNettWage)
```

</p>

}

Go to the AppStart folder and open the RouteConfig file. Change the action name to DisplayWage and controller name to EmployeeWage, run your application. This file has the URL for your application.

Tutorial 9

Parking Lot Management System – Part1

You are required to code a solution for a parking lot management system. The parking lot has three different parking types i.e. lockup garage, under cover, open bay and the parking cost is R20, R15 and 10 respectively. The parking service is charged per hour. These hours (parking duration) is the difference between the time-in and time-out.

Write an MVC web application to develop a solution for this management system. Output the following on a parking slip:

Vehicle Registration

Duration of the parking

Parking Cost

Parking Lot Management System – Part2

You are required to code a solution for a parking lot management system. The parking service is charged per hour. These hours (parking duration) is the difference between the time-in and time-out. The parking lot has three different parking types i.e. lockup garage, under cover, open bay and the parking cost for the first hour is R20, R15 and 10 respectively and a 50% discount is given for each hour thereafter for the respective parking cost.

Write an MVC web application to develop a solution for this management system. Output the following on a parking slip:

Vehicle Registration

Duration of the parking

Parking Cost

Tutorial 10

Over a year ago, the World Health Organization had declared a state of disaster across the globe. Currently many pharmaceutical companies have developed a vaccine that has proven to be effective when having the recommended two doses. However, it is not a cure to the corona virus. The vaccine has assisted many countries in minimizing the death rate across the world. In SA, our shipment of vaccines has arrived and they only need to be administered. The SA government requires a C# MVC application to be used at each venue that has been identified. Input the persons year of birth, id number, full name, cellphone number, address and income for a month. The application must calculate the age of each person to determine if they qualify to receive the vaccine immediately. If the person's age is 60 and over, then he/she will receive the first dose of the covid-19 vaccine. The date of the 2nd dose must be calculated exactly one calendar month after their 1st dose. There is a small fee that is charged

for administering the vaccine, which is three percent of your income. A total fee must be calculated using the fee and VAT. Display the fee, VAT and total fee to pay in cash or card.

Application name Home About Contact

Year of Birth 1960

Full Name Penlope Jackson

ID number 6004155236963

Cell Number 0741458560

Address 45 Winterton Road, Berea

Monthly Income 10000

Qualifying Status ☒

2nd Vaccine Dose Date 2021-6-29

Fee R 300

VAT R 45

Total Fee R 345

Submit

[Back to List](#)

Fig1 – Vaccine Tracking System

Given the C# code for the model called VTrackingModel showing all the necessary properties that suits the above scenario.

- calculate the age of the person. Determine if the person qualifies for the vaccine or not and return a status.
- calculate the date for the second dose of the vaccine (ie. 2021-7-2) based on the qualifying status of the person.
- calculate the total fee based on the income and the qualifying status of the person.
- calculate the VAT on the fee.
- calculate the total fee after including the VAT.
- The controller named VTracking is demonstrating how the controller facilitates inputs from the view, and their processing in line with the model.
- Write the code to render the view as shown in Fig1.

Dropdownlist example

Code for the model

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
```

```
namespace WebAppWage.Models
{
    public enum empGender
    {
        Select,
        Male,
        Female
    }
}
```

```
public enum SkillsType
```

```
{  
    Beginner,  
    Intermediate,  
    Expert  
}
```

```
public class Wage
```

```
{  
    public string empNum { get; set; }  
    public string empType { get; set; }  
    public empGender gender { get; set; }  
    public int hoursWorked { get; set; }  
    public double rate { get; set; }  
    public double empWage { get; set; }  
    public double empTax { get; set; }  
    public double empNettWage { get; set; }  
    public double empBonus { get; set; }  
    public SkillsType empSkillsType{get;set;}
```

```
//Calculate wages for employee working less than 40 hour. Calculate  
//wages for employee working overtime which is any number of hours over 40  
//calculate bonus based on the employee skills type  
//being beginner-R500, Intermediate- R1500 and Expert-5000
```

```
public double calcGrossWage()
```

```
{  
    double normalpay = 0, overtimepay=0;  
    if (hoursWorked <= 40)  
    {  
        normalpay=(rate * hoursWorked);  
    }  
    else  
    {  
        normalpay = rate * 40;  
        overtimepay= (hoursWorked - 40) * rate * 1.5;  
    }  
}
```

```
    return (normalpay + overtimepay);
```

```
}
```

```
//Calculate tax amount for employee based on their type of employment  
//If part-time, tax is 15%. If full time, tax is 40%
```

```
public double calcEmpTax()
```

```
{ double taxAmt = 0;  
    if(empType=="Part-time")  
    {  
        taxAmt = 0.15 * calcGrossWage();  
    }  
    else if(empType=="Full time")  
    {  
        taxAmt = 0.40 * calcGrossWage();
```



```

    }
    return (taxAmt);
}

public double calcBonus()
{ double empb = 0;
  if (empSkillsType == SkillsType.Beginner)
  {
    empb= 500;
  }
  else if (empSkillsType == SkillsType.Intermediate)
  {
    empb=1500;
  }
  else if (empSkillsType == SkillsType.Expert)
  {
    empb= 5000;
  }
  return empb;
}
public double calcNettWage()
{
  return ((calcGrossWage()+ calcBonus()) - calcEmpTax());
}

}
}

```

Code for the Controller

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using WebAppWage.Models;

namespace WebAppWage.Controllers
{
    public class EmployeeWageController : Controller
    {
        // GET: EmployeeWage
        public ActionResult DisplayWage()
        {
            return View();
        }
        [HttpPost]
        public ActionResult DisplayWage(Wage wg)
        {
            wg.empNum = wg.empNum;
            wg.empType = wg.empType;
            wg.gender = wg.gender;
            wg.hoursWorked = wg.hoursWorked;

```

```

        wg.rate = wg.rate;
        wg.empWage = wg.calcGrossWage();
        wg.empTax = wg.calcEmpTax();
        wg.empBonus = wg.calcBonus();
        wg.empNettWage = wg.calcNettWage();

        return View(wg);
    }
}

```

Code for the View

```

@model WebAppWage.Models.Wage
@{
    ViewBag.Title = "DisplayWage";
}

```

```
<h2>DisplayWage</h2>
```

```

@using (Html.BeginForm("DisplayWage", "EmployeeWage", FormMethod.Post))
{
    <p>
        Emp Number: @Html.EditorFor(e => e.empNum)
        <br />
        Hours Worked: @Html.EditorFor(h => h.hoursWorked)
        <br /> Rate: @Html.EditorFor(r => r.rate)
        <br /> Gender
        <br /> @Html.DropDownListFor(m => m.gender, new SelectList(Enum.GetValues
        (typeof(WebAppWage.Models.empGender))), "Select Item")

        <br /> Skills: @Html.DropDownListFor(d => d.empSkillsType, new
        SelectList(Enum.GetValues
        (typeof(WebAppWage.Models.SkillsType))), "Select")

        <br /> Employment Type
        <br /> @Html.RadioButtonFor(a => a.empType, "Part-time") Part-Time
        <br /> @Html.RadioButtonFor(a => a.empType, "Full time") Full Time

        <br /> <input type="submit" value="Calc" />
        <br /> Employee Wage: @Html.DisplayFor(w => w.empWage)
        <br /> Employee Tax: @Html.DisplayFor(s => s.empTax)
        <br /> Employee Bonus: @Html.DisplayFor(a => a.empBonus)
        <br /> Employee Nett Wage: @Html.DisplayFor(c => c.empNettWage)

    </p>
}

```

Go to the AppStart folder and open the RouteConfig file. Change the action name to DisplayWage and controller name to EmployeeWage, run your application. This file has the URL for your application.

Tutorial 11

Ibiza Pizza offers students the option to purchase a slice of a veg pizza [R11.99] or a chicken pizza [R14.99], depending on the amount that the student has available to spend. Currently, Ibiza Pizza uses paper-based system where they record their day-to-day sales. They also have a standalone calculator which they use to perform calculations for their sales. The Ibiza Pizza owner has approached an I.T company called AllThingsTech and requested them to design and develop a simple application that will assist them, Ibiza Pizza, to address their problem of sales miscalculations which has negatively impacted on their business profitability for the past few months.

Application name

Pizza Sales

Cash Tendered

Please enter your cash

Select pizza type :

Veg ☐

Chicken ☐

Pizza Slices in Quantity

Your quantity must be between 1-5

Calculate

Your amount Spent :R

Your change is :R

© 2019 - My ASP.NET Application

Ibiza Pizza prototype validations

Application name

Pizza Sales

Cash Tendered

Select pizza type :

Veg ☒

Chicken ☐

Pizza Slices in Quantity

Calculate

Your amount Spent :R 47,96

Your change is :R 52,04

© 2019 - My ASP.NET Application

Ibiza Pizza prototype inputs and output of the app

The programmer manager at AllThingsTech, where you work as a developer/programmer, has instructed you to use Visual Studio tool to develop this application. In the discussion that she had with you, she also mentioned that it will be a great idea if you could utilize the ASP.NET MVC5 Framework because of its suitability and its enormous benefits which includes separation of concerns. The examples shown above in Figure 1.1 and Figure 1.2 demonstrate a prototype of this application. You are required to use DataAnnotations to perform model validations.

Tutorial 12

Developing a simple web application using ASP.NET MVC Framework

This tutorial is based on Tutorial 14. The Ibiza Pizza business continues to grow and they no longer sell slices of pizza but rather they are now offering a full size pizza that comes in different sizes as indicate in Table 1 below. Pizza Size	Pizza Price [in Rands]
Extra Large	100.00
Large	75.50
Medium	50.50
Small	35.99

Application name

Pizza Sales

Cash Tendered

Select pizza type :

Veg ☐

Chicken ☐

Select pizza size :

*****Please Select Pizza Size***** ▼

Quantity

Want something to drink?:

Yes: ☐

Calculate

Your amount Spent :R

Your change is :R

© 2019 - My ASP.NET Application

Application name

Pizza Sales

Cash Tendered

Select pizza type :

Veg ☐

Chicken ☐

Select pizza size :

*****Please Select Pizza Size***** ▼

*****Please Select Pizza Size*****

Extra Large

Large

Medium

Small

Want something to drink?:

Yes: ☐

Calculate

Your amount Spent :R

Your change is :R

© 2019 - My ASP.NET Application

Figure 1.1: Ibiza Pizza extended prototype

Tutorial 13

The Best Budget Hotel has requested that you write a C# MVC application that will calculate and display the amount due by a guest. The amount due is based on the number of guests, the number of nights and the price per person per night. If a guest books for more than three nights, each guest pays half price for one of the nights.

Write the C# code to accomplish the following sequence of tasks.

- a) create a model with the appropriate properties.
- b) create a method to calculate the total amount due by the customer before VAT has been considered.
- c) create a method to calculate the VAT amount
- d) create a method to calculate the total amount to be paid by the customer after VAT has been included.
- e) You are required to write C# code for this controller in order to demonstrate how the controller facilitates inputs from the view.
- f) Write the HTML and RAZOR lines to create the view for this app.

Tutorial 14

Create a model to capture a lodge' name, city or location, room types (singles only OR singles and doubles) number of rooms and average price per room. Each lodge can further have luxurious features such as the ones tabled below:

Luxurious Feature	Cost Estimate
Any daily massages	50% of avg price per room
Fully packaged TV services	25% of avg price per room
Open line telephone services	75% of avg price per room

NB: Actual prices will be considered and calculated as the as part of actual fees of each lodge rooms are being booked for a particular number of days.

This model must further compute the expected profits in consideration of all prices involved, accordingly.

Demonstrate through coding techniques, how you would achieve fully functional MVC application, through the controller(s) and the view(s). Strive for user friendliness particularly in prompting for user input and presentation of user output. Please take note of the selections and/or

Tutorial 15

DUT IT Department wants to track number of years its employees have been, with the department. Input to the program is the employee's year of employment, calculate and output years of service. The system must categorised employees according to the categories in Table 1 below: You are required to write a C# MVC application to do the following:

YEARS	Type
<=5	Novice
6-10	Senior
11-20	Experienced
>20	Veteran

Table 1

Tutorial 16

Metered Taxi Application Solution

A metered Taxi application must be created to calculate the total fare a person pays for the number of kilometres travelled by taxi. Input the commuter name, pick up point, drop off point and number of kilometres travelled. For each kilometre travelled, the commuter pays a R15. Display the total due by the commuter.

Tutorial 17

Takeaway

Vees Takeaway is an informal takeaway situation outside the DUT campus.

They use a cash register to take customer's orders and provide them with a total owing on their purchase. However recently their customers have increased and they have decided to use a computer application to do the above. They offer set meals. The customer can choose only one type of food at any given time due to the prices being so low, but the quantity of the type of food is literally unlimited. Create a MVC based web application for Vees Takeaway to allow a user to key in the customer's order based on the criteria highlighted below and to determine the cost of the order for which they are required to pay.

Specifications

- The customer's name must be captured.
- Their food choice and their prices together with quantity must be entered ONLY ONE food choice can be made.
- Their TotalDue must be calculated
- Display an order number that's generated by the Random class. (any number between 1 and 1000)
- Display the customer's food choices, their cost and quantity
- Display all the monetary value outputs

Tutorial 18

A DHL customer hands over his or her good to the Durban branch, for couriering. A DHL employee then captures information about both customer & item details. The employee further checks whether that item can be delivered to that destination. If everything is okay, "way bill" is printed out. it contains

all the information to identify the delivery. Charges include package price where a kilogram of whatever parcel has a basic charge. A further charge is that of the distance between the DHL offices and the parcel destination; charged per kilometre. A parcel shipment must further show vehicle used to transport the parcel, and the driver involved. A vehicle can transport more than one parcel. It is also important to record initial mileage of parcels delivering vehicle. On return, should a vehicle's mileage be 100km more than the expected mileage, the driver must be flagged for manager's attention. In a case where a parcel weighs less 50kg and needs to be transported to more than a 100KM destination, a certain discount in percentages must be considered in terms of the package price. This discount increases by some percentages (e.g. 1.5% more) for every 50km from a 100km. No parcel can be transported for more than 1000km.

Tutorial 19

Musa and Gugu, a Durban couple obsessed with facial care products, subscribe to some of the well-recognised facial (cosmetics) brands such as Ponds, Gamier, Clinique and L'Oreal. They simply call their business Igugu Merciful Facials, shortened as IMF. As it is well known in the current age, facial care brands are now heterogeneously available to accommodate both men and women. Boys/men concerned with their facial care, therefore have Musa to advice and Gugu can take care of all the ladies before they can even consult her man. Similar to all business that deal with products, inventory and sales are the core of their thriving business. As a couple, they acknowledge other couples that come into IMF looking for the 'His and Hers' products; at special or discounted sales. They also recognise special days such as Valentine's, Women's, Mother's and Father's day; during which various specials and or discounts are considered across various products.

Tutorial 20

Musa and Gugu, a Durban couple obsessed with facial care products, subscribe to some of the well-recognised facial (cosmetics) brands such as Ponds, Gamier, Clinique and L'Oreal. They simply call their business Igugu Merciful Facials, shortened as IMF. As it is well known in the current age, facial care brands are now heterogeneously available to accommodate both men and women. Boys/men concerned with their facial care, therefore have Musa to advice and Gugu can take care of all the ladies before they can even consult her man. Similar to all business that deal with products, inventory and sales are the core of their thriving business. As a couple, they acknowledge other couples that come into IMF looking for the 'His and Hers' products; at special or discounted sales. They also recognise special days such as Valentine's, Women's, Mother's and Father's day; during which various specials and or discounts are considered across various products.

Suppose there is a model called IMFInventories that captures product names, brand, quantities, and unit price and mark-up percentage:

Tutorial 21

DUT Research TV Channel scenario

DUTRTV presenters attract and interview experts on various topics in the categories of technology, finance, politics and sports amongst others. Once the presenter locates an expert, an interview is scheduled and a recording of that interview is posted on the DUTRTV Facebook page, so as to meet the target audience, i.e. students, staff and the community at large. All presenters get a basic salary, however, the popularity of a video on the Facebook page, attracts extras income within at least a month, i.e. the number or likes and comments on the video for at least the first month it was posted, attracts

extra commission for the presenter. This is done so to compensate the presenter for coming up with an interesting topic and or combining a very interesting, influential and or knowledge contributing type of an interview. It will also be interesting to know top presenters, and videos in various categories over a period of time, so as to grant acknowledgements such as awards.

The possible models and their attributes that would be created in MVC application for the scenario above. Write your answer in the following format:

ModelName (Attribute 1, Attribute 2, etc.)

2. Assume there is ONLY one controller; write (code) appropriate controller methods to show the link between all of your models to the application's possible view

Tutorial 22

Silili Take Aways

Silili Take Aways offers take aways to the members of the public. The customer must come as a group and the minimum is 3 in a group. There are four different types of menus available e.g. Curry & Phutu, Curry & Rice Etc. Each menu type has its own cost (User input). The group of customer can choose extras such as bottom-less drinks or salads or both. These extras have their own costs (*you can decide on the cost*). Note that these extras are also based on the number of members in a group.

Write an application that will calculate and output *Basic Meal Cost*, *Cost for Extra*, and *Total Cost of the meal for the whole group*.

Tutorial 23

The Accounting and Auditing Firms Association (AAFA) employs consultants that are responsible for attracting and affiliating firms into the association, for the purposes of creating a network of accounting and auditing firms for client referrals. AAFA has the following consultant categories employed: Junior, Senior and Manager.

Each consultant, irrespective of category, earns a basic commission of R920.00, for every firm they attract and affiliate. However, a so called rank based commission is earned as tabulated in Table 1 below:

Table 1 Rank based commission

Consultant Code	Rank based commission
Junior	5% extra of the basic commission earned
Senior	15% extra of the basic commission earned
Manager	25% extra of the basic commission earned

The fullname, basic commission earned for all firms, rank based commission for all firms, the total commission is made up of the basic commission and the rank based commission.

Tutorial 24

Read the following case study carefully and then answer the questions that follow.

The Spa Reservations has requested you to develop a system, which calculates the total cost due for all group treatments. The company has three treatments currently. These are the morning spa, full day spa and night spa treatments. Furthermore, each reservation has a once-off additional cost of R200.00.

The user must enter their name, cellphone number, the number of people that are going to be coming in for the spa, spa package and treatment cost. Calculate the basic cost, additional cost, VAT cost and total cost due by a customer.

Tutorial 25

YouTube has a great Initiative that allows anyone to create great content videos to empower the world while earning some money.

The MVC ASP.Net application will input the name, surname, date of birth, ownership of PayPal account, Description of channel content, number of social media accounts, select the social media account with the highest number of viewers and the number of subscribers. The user earns an amount on the YouTube channel, which is 55% of the highest subscribers. An additional amount is earned on a PayPal account of 1% of their highest subscribers. The total number of social media accounts must be calculated and; if the user has 1 or 2 social accounts, then they earn an amount that is 10% of the number of social accounts and highest subscribers. If the user has 3 or 4 social media accounts, then the amount earned is quarter of the percentage, of the number of social accounts and highest subscribers. If the user has 5 or 6 subscribers, then the user earns an amount of 2 times the quarter of the percentage, of the number of social accounts and highest subscribers. Calculate the total before tax, the tax and the total after including tax.

The screenshot shows a web form with the following fields and values:

- Name: David
- Surname: Khosa
- Do you have a PayPal account?: Yes (dropdown menu)
- YouTube channel description: This channel has amazing
- Select all the social media accounts, you are registered for:
 - Facebook account: ☒
 - Twitter account: ☒
 - Instagram account: ☒
 - LinkedIn account: ☒
 - Pinterest account: ☒
 - Blogger account: ☒
- Highest subscribers from one of your social media accounts: 1000
- Number of social media accounts: 6
- PayPal earnings: 10
- YouTube earnings: 550
- Social media earnings: 3000

Write the C# code for the model called YouTubeModel and show all the necessary properties that suits the above scenario.

Method to calculate the amount earned on paypal account.

- a) Method to calculate amount earned on the Youtube channel.
- b) Method to calculate the number of social media accounts.
- c) Method to calculate the amount earned on the social media accounts.
- d) Method to calculate the total before tax.
- e) Method to calculate the total tax.
- f) Method to calculate the total after including tax.
- g) The controller named YoutubeController is demonstrating how the controller facilitates inputs from the view and their processing in line with the model.
- h) Write the code to render the view as shown in Fig1.

Unit Testing

Tutorial 26

- a) Explain what Test-Driven Development (TDD) is.
- b) What is the difference between Unit Testing and Integration Testing?
- c) List 3 Unit Testing frameworks that are available in .Net platform.
- d) Which Unit Testing framework is integrated in Visual Studio?
- e) What are the benefits of Unit Testing?
- f) In ASP.NET, what is the namespace that enables a developer to access unit testing C# attributes?
- g) Name 2 special C# attributes that a developer needs to include in a unit test class.
- h) What does the AAA pattern used when implementing unit tests stand for?
- i) Explain the function of each “A” of the AAA pattern.
- j) Write Unit tests for the following C# methods:

a. `public class Calculator`

```
{
public decimal multiplication( decimal num1, decimal num2)
{
decimal sum = num1 * num2;
return sum;
}
}
```

b. `public class Calculator`

```
{
public int num1 { get; set; }
public int num2 { get; set; }
public decimal addition(int num3)
{
decimal sum = num1 + num2 + num3;
return sum;
}
}
```

c. `public class PizzaSales`

```
{  
public string pizza_Type { get; set; }  
public double slice_cost { get; set; }  
public int quantity { get; set; }  
public double total_cost { get; set; }  
public double calcTotalCost()  
{  
if (pizza_Type == "Veg")  
{  
slice_cost = 11.99;  
total_cost = quantity * slice_cost;  
}  
else if (pizza_Type == "Chicken")  
{  
slice_cost = 14.99;  
total_cost = quantity * slice_cost;  
}  
return total_cost;  
}  
}
```

11. Write a C# code that demonstrates the use of Data Annotations for the following properties (Use a minimum of three for each property):

- a) `public int num1 { get; set; }`
- b) `public int num2 { get; set; }`
- c) `public int quantity { get; set; }`
- d) `public string pizza_Type { get; set; }`

12. Use any ASP.NET MVC application that you have completed (and add unit testing for any action method that exist in your controller.

Tutorial 27

The purpose of this tutorial is to assess the understanding of unit testing concept and logic reasoning, in relation to defining the methods of a domain class.

1. Given the table below, provide a C# code that could possibly appear in a model class.

ID	Event Name	Place	Start_Time	End_Time	No_of_guests	Cost	Event_Ref
1	Pens Down	DUT	23:00	04:00	15000	90000	D_Pens150

- a) Use unit testing to verify the method(s) defined in the model class.

- b) Assume there is a model class called Vehicle that exist. You are required to write a method that computes the price of the vehicle based on its name. If the vehicle name contains “s”, the price is R50 000, else if the vehicle name contains “t”, then the price is R90 000, otherwise the prices is R100 000. Write the controller and View code for this tutorial.
 - c) Use unit testing to verify the function/method defined in the Vehicle class.
-

Tutorial 28

The purpose of this tutorial is to assess the understanding of unit testing concept and logic reasoning, in relation to defining the methods of a domain class.

- a. Write the C# code for the model and show all the necessary properties that suits the above scenario.
- c. Write the c# code for the following methods to:
 - d. a method called CalcDiscount that takes in a price, as a parameter, and calculates a discount based on the information provided in the table below. Assume this method exists in a domain class named Products.
 - e. Write the code for the controller to demonstrate how the controller facilitates inputs from the view, their processing in line with the model.

Price	Discount
Below R1500	10%
Between R1500 and R5000	15%
Above R5000	20%

Table 1

- d) Implement unit testing for CalcDscount method.
-

Tutorial 29

Write a method called CalcDpMark that takes in three test marks, as parameters, and computes a student’s DP mark based on the best two test marks. Assume this method exists in a domain class called StudentRecord.

- a. Write the C# code for the model and show all the necessary properties that suits the above scenario.
 - c. Write the c# code for the CalcDPMark method.
 - d. Write the code for the controller to demonstrate how the controller facilitates inputs from the view, their processing in line with the model.
 - e. Write the code to render the view.
 - f. Implement unit testing for CalcDpMark method.
-

Tutorial 30

Write a method called TicketRef which returns a flight ticket reference number base on the flight name, passenger identity number, and class. For example, I (I.D number 2001205336086) may want to travel with Kululu.com, on business class; which means that my flight ticket reference number may look like Kul2001205336086Bus. Assume this method exists in a domain class called Tickets.

- Write the C# code for the model and show all the necessary properties that suits the above scenario.
- Write the c# code for the following methods to:
 - calculation of the total based on the kit type.
 - calculation of the VAT.
 - calculation of the total after including VAT.
- Write the code for the controller to demonstrate how the controller facilitates inputs from the view, their processing in line with the model.
- Write the code to render the view.
- Implement unit testing for TicketRef method.

Tutorial 31

A contractor has two types of employees, namely, full time and part time, who are work with him on various projects contracts that he attracted from the municipality. Due to the varying nature of the projects, it is hard for the contractor to set permanent basic wages, salaries etc. for either type of his employees. However, the contractor is certain about various payment attributes concerning his employees and this information is shown in the Table below:

Table 2.1 Payment attributes Employee Type	Set Basic Salary	Rate per Hours	Num_Ho urs	Tax%	UIF%	Pension Fund%	
Part Time	□ □		□ □		□ □		□
Full Time	□ □		□ □ □		□ □ □		□

This MVC application uses data annotations and scaffolding techniques and further incorporates unit testing for logical functionality purposes. Use Table 2.1 and and all the provided information, with sound decision making to answer the following questions

Write C# code that shows data annotation namespaces, prompts the user for, and forces them to capture the name of the employee.

Consider the type of employee; code all the necessary **methods** for the **ContractorEmployeesPay** Model to compute appropriate deductions from the relevant gross salary. Compute the gross salary where necessary. Furthermore, compute the net salary.

Write HTML code to present the Employee Types, Part Time and Full Time to the user in a form of giving the user a choice of selection.

Exam Revision Full SCENARIO 1

The T20 International cricket series was a huge success in previous years and therefore the cricket teams have decided to give their best to win this year. Many T20 matches are scheduled to be played at the Kingsmead stadium in Durban this month, where the tickets will be on sale. You are required to develop an ASP.Net MVC application to book tickets for a cricket match at Kingsmead stadium in Durban. Use Fig2 as a guide for the user interface. This online system will be an efficient initiative that will reduce the long queues each week, when the sales for the cricket matches is being processed. However, with the online system in place, spectators can view as well as buy the tickets immediately online. There are three different categories of seat sections that the spectator may choose to sit, See Table1. For those who are driving, the stadium has two different parking facilities i.e. VIP and General parking with a fee of R200 and R100 respectively, for the duration of the cricket match.

Table 1 below shows the seat section and its corresponding fee.

Seat Section	Fee (R) per ticket
Grass	50
Grandstand	90
Nederburg	150

Table 1

The prices for the scholars, adults and pensioners varies based on the following criteria: if you are a scholar, you pay half the fee listed above based on the seat section selected. If you are an adult, you pay full price as listed above, and lastly if you are a pensioner, you pay one quarter of the fee listed above.

- Write the C# code for the model called CricketBooking and show all the necessary properties
- that suits the above scenario.
- Write the c# code for the following methods to:
- calculate the cost based on the seat section.
- calculate the total cost based on seat section and the number of scholars.
- calculate the total cost based on seat section and the number of adults.
- calculate the total cost based on seat section and the number of pensioners.
- calculation of the parking cost based on the parking type.
- calculation of the total.
- calculation of the total cost after including VAT.
- The controller named Cricket is demonstrating how the controller facilitates inputs from the view, their processing in line with the model.
- Write the code to render the view as shown in Fig1.

T20Booking

Name:

Email Address:

Seat Section: ☐ Grass Area
☒ Grandstand
☐ Nederburg Stand

Number of Scholars

Number of Adults

Number of Pensioners

Parking Type ☒ VIP
☐ General

Total 447,5
Total After VAT 514,625

Fig1 – T20 Booking form GUI with test data

Exam Revision Full SCENARIO 3 - SPORTS SHOP

The interface below (Figure 1) allows a user to select her favourite sport, team and kits.

Online Shop

Favourite sport:
Favourite Team:

Kit Type

☒ Kit Set

☐ Individual Items

=====Summary=====

Favourite sport: Rugby

Favourite Team: Sharks

Kit Type: Kit Set

Kit Item:

- Jersey
- Track suit
- T-shirt
- Cap

Figure 1

- a. Write the C# code for the model and show all the necessary properties that suits the above scenario.
- c. Write the c# code for the following methods to:
 - d. calculation of the total based on the kit type.
 - e. calculation of the VAT.
 - f. calculation of the total after including VAT.
- g. Write the code for the controller to demonstrate how the controller facilitates inputs from the view, their processing in line with the model.
- f. Write the code to render the view as shown in Fig1.

Full SCENARIO 4

Casual Lounge App Hlabikhefu Casual Lounge [HCL] is a private resting area within the department of I.T DUT. In this lounge staff members are allowed to either rest, take a nap or have some refreshments. Each service has its own cost which is charged per hour, see Table 2.1 below. If a staff member wishes to enjoy refreshments or a nap, he / she will be charged an additional 50% of the resting service per hour (since refreshment as well as napping also involve resting). The IT department has two SECTIONS i.e. Software Development and Information Systems (IS), so a staff member can belong to either one. If a staff member from the software development section books for two hours or more, he/she get charged one hour less. For IS staff they need to book for four hours or more to get such discount. As an MVC developer you are asked to develop an ASP.NET MVC application that will work out the amount due when a staff member requests a service. Use Figure 1 and the guidelines below to develop a solution for Hlabikhefu lounge.

Service	Rate/Hour
Rest	R100.00
Refreshments	R120.00
Take a Nap	R150.00

Table 1 Service rate per hour

- Write a C# code for a complete class (or model) called Lounge.cs showing all the methods properties and necessary validation.
- Code the method that will determine chargeable hours.
- Code the method that will calculate basic service cost
- Code the test methods for chargeable hours. Code the test methods for service cost method.
- Code the view.
- Write the code that will link the model to this view.
- Write the full action method that will be executed when this view is submitted/posted.

Full SCENARIO 5

The International Rugby championship was a huge success in previous years and therefore the rugby teams have decided to give their best to win this year. Many of the matches are scheduled to be played at the Kings Park stadium in Durban this month, where the tickets will be on sale. You are required to develop an ASP.Net MVC application to book tickets for a rugby match at Kings Park stadium in Durban. Use Fig2 as a guide for the user interface. This online system will be an efficient initiative that will reduce the long queues for each season. With the online system in place, spectators can view and buy many tickets online. There are four different categories of seat tickets available that the spectator may choose to sit, See Table1. Category A is the closest stand tickets to the ground, while Category D is further away from the ground. For those who are driving to the stadium, there are 2 different parking facilities available i.e. undercover and open parking with a fee of R75 and R25 respectively, for the duration of the match.

Table 1 below shows the seat section and its corresponding fee.

Seat Ticket	Fee (R) per ticket
Cat A	200
Cat B	160

Cat C	120
Cat D	80

Table 1

The prices for the scholars, adults and pensioners varies based on the following criteria: if you are a scholar, you pay 3 quarters of the fee based on the seat ticket. If you are an adult, you pay the full price as listed above, and lastly if you are a pensioner, you pay two quarters of the fee listed above.

- Write the C# code for the model called RugbyBooking. Include data validations, enum methods and properties within the model.
- Write the C# code for the following methods to:
 - calculate the cost based on the seat ticket.
 - calculate the total cost based on seat section and the number of scholars.
 - calculate the total cost based on seat section and the number of adults.
 - calculate the total cost based on seat section and the number of pensioners.
- calculation of the parking cost based on the parking type.
- calculation of the total cost.
- calculation of the total cost after including VAT.
- The controller named Rugby is demonstrating how the controller facilitates inputs from the view their processing in line with the model.

Full SCENARIO 6

Build-a-Costume workshops are a global brand that both kids and parents love. These workshops give kids a sense of independence to create their own custom designed costume of their choice. Each costume is uniquely created for kids bears. Build-a-Costume requires an ASP.Net MVC application to do the following: select the type of bear from the list provided (Table 1), the material type (Table 2) to create the costume and also choose whether the costume should be stitched in-house or not. If a kid selects a Birthday Bear, then 5 meters of material will be required to create the costume, otherwise if a kid selects Tatty Teddy, then 4 meters of material is required. Lastly, if the kid selects Winnie Pooh, then 7 meters of material is needed. When groups of kids come to build-a-costume for their bear, a group discount of 2 quarters of the total per kid applies to all in the group. Lastly, a loyalty program allows kids to earn points for promotional and special offers. For every R25 spent, a kid earns one and a half loyalty points.

Bear Type	Cost(R) for bear
BirthdayBear	180
TattyTeddy	140
WinniePooh	200

Table 1

Material	Cost(R) per metre
Cotton	50
Nylon	25
Silk	100

Table 2

Write the C# code to creat the model, which includes data validations, properties and the use of any enum methods etc.

- A C# method to calculate and return the cost based on the BirthdayBear only.
- A C# method that calculates the stitching cost for the costume?
- A C# method to calculate and return the cost based on the material type and number of metres
- required for costume.

- e) A C# method to calculate the total due before any discounts are included?
- f) A C# method to calculate the discount amount based on the total amount due.
- g) A C# method to calculate the total amount due after discount has been included.
- h) A C# method that calculates the loyalty points earned.
- i) The controller named BearCostume is demonstrating how the controller facilitates inputs from
- j) the view, their processing in line with the model.
- k) Write the Razor syntax to render the view for this application.

Full SCENARIO 7 - All-in-One Event Planner Management System

All-in-One event planner is a company that organise and plan events. You are required to study and analyse Figure 1 below and answer the following questions:

The screenshot shows a web application titled "Event Planner". It contains several input fields and a "Submit" button. Below the inputs, it displays calculated costs for a specific event configuration.

Field	Value
Client Name	BN
Select Event Type -	Party <input type="radio"/> Wedding <input checked="" type="radio"/>
Enter Cost :	2500
Select Event Time -	Day Event <input type="checkbox"/> Night Event <input checked="" type="checkbox"/>
Duration in Days	2
Number of attendees	200
Submit	
Basic Event Cost :	5000
Attendees Cost :	500
Security Cost :	200
Total Event Cost :	5700

- a) You are required to write a C# model to demonstrate all the necessary properties, and ONLY a method to compute the event basic cost. Bear in mind that there are two main types of events i.e. Parties and Weddings and each event has its own basic cost (to be captured by the user) and are charged per day (Event Basic cost).
- b) There is an additional cost (attendees cost), this cost is based on the number of people attending the event. If the number of people are less than or equal 200, the additional cost is 10% of the event basic cost and if they are greater than 200, then the attendees cost is 15% of the event basic cost. You are required to write a C# method to calculate this attendees cost.
- c) There is another additional cost (security cost), this cost is based on whether the event is a day event or night event. The security cost for a day event is equal to the 10% of the attendees cost

and security cost for a night event is equal to 20% of the attendees cost, this cost is calculated per day. You are required to write a C# method to calculate this security cost.

- d) Write C# code for a method situated in your model to compute the overall event cost.
- e) The controller code that wires everything together. NB. Include only the necessary namespace(s) and ignore all the default ones.
- f) Provide the code that will produce the following layout of the view.

Event Planner

Client Name

Select Event Type - ☐ Party ☒ Wedding

Select Event Time - ☐ Day Event ☒ Night Event

Basic Event Cost : 5000

Total Event Cost : 5700

Fig 2 – A full View showing a running application for All-In-One Event Planner

Full SCENARIO 8

T&T adventure co. has decided to host another themed camping trip for kids in and around Durban area. Previously, T&T adventure co. has experienced low profit/gains, which was primarily caused by miscalculations due to their use of manual paper based system. T&T adventure co. are looking for an experienced programmer/developer, who will develop an application that will calculate the cost due to parents who wish to take their kids for camping.

You are required to develop an application that follows an ASP.NET framework in order to fluidly and easily achieve quality attributes such as maintainability, testability and robustness. Figure 2 shows the expected look and output of this application.

This T&T application must accept the following user inputs: name of parent and number of kids attending. Furthermore, the parent must select a camp type and activity type that their kid(s) will participate in. A standard fee of R150 is charged for each kid, which includes the venue and lunch or supper. There is an additional fee of R100 per kid who opted for the night camp, as this will require allocation of a camp area with a bed to sleep. Table 1 below shows each activity and its corresponding fee.

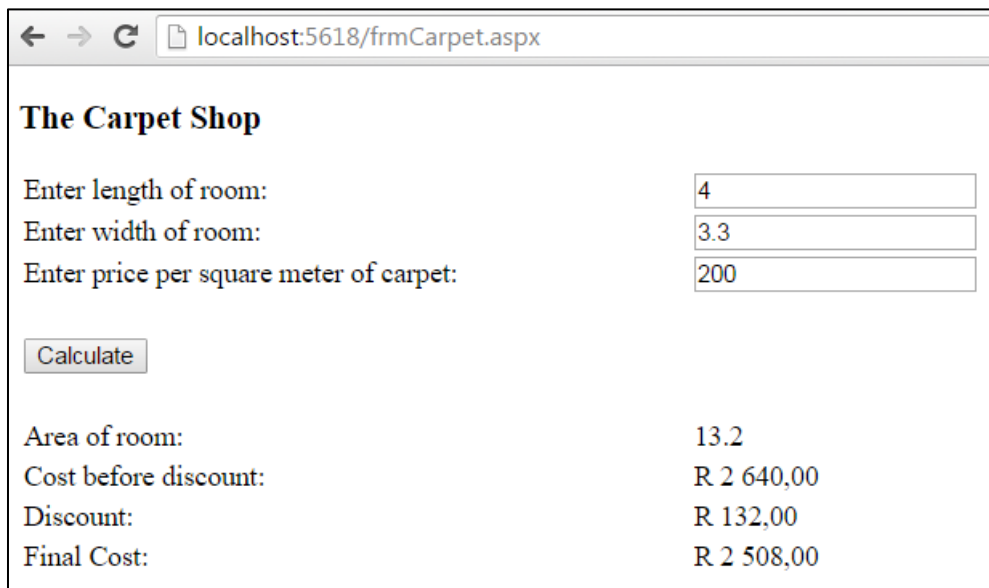
Activity	Fee (R)
Canoeing	50
Rafting	110
Hiking	120

Table 1: Activities with their corresponding fees

- Write a C# code for the model called Camp and show all the necessary properties and methods that suits the above scenario. Demonstrate all necessary namespaces where it is required. Use Figure 2 as a guide and make sure that methods enable for the following :
- Calculation of the standard cost.
- Calculation of the cost based on the camp type.
- Calculation of the cost based on the activity type.
- Calculation of the total cost.
- You are required to write a C# code for the controller named Camping in order to demonstrate how the controller facilitates inputs from the view, their processing in line with the model in 3.
- Write a RAZOR view syntax (code) for the following view.

Full SCENARIO 9

The Carpet Shop has requested that you create an application which will assist a customer to calculate and display the cost of a carpet for a room, given the length and width of the room and the price per square meter of carpet. Together with the management of The Carpet Shop, you have come up with the following design decisions:



The screenshot shows a web browser window with the address bar displaying 'localhost:5618/frmCarpet.aspx'. The page title is 'The Carpet Shop'. The form contains the following elements:

- Input fields for 'Enter length of room:', 'Enter width of room:', and 'Enter price per square meter of carpet:' with values 4, 3.3, and 200 respectively.
- A 'Calculate' button.
- Output fields for 'Area of room:', 'Cost before discount:', 'Discount:', and 'Final Cost:' with values 13.2, R 2 640,00, R 132,00, and R 2 508,00 respectively.

Figure 1

Write a C# code for the model called Camp and show all the necessary properties and methods that suits the above scenario. Demonstrate all necessary namespaces where it is required.

Description of methods:

- CalcArea(): uses the length and breadth to calculate the area of the room to be carpeted.
- CalcCost(): uses the method CalcArea() and the price per square meter of the carpet to calculate the basic cost of the carpet.
- CalcDisc(): uses the method CalcCost() to calculate the discount due to the customer. The customer gets a discount of 10% if the basic cost of the carpet is more than R2000.
- CalcFinal(): uses the methods CalcCost() and CalcDisc() to calculate the final amount due by the customer.
- You are required to write a C# code for the controller named Camping in order to demonstrate how the controller facilitates inputs from the view, their processing in line with the model in 3.
- Write a RAZOR view syntax (code) for the following view.

Full SCENARIO 10

Super Sleek Gym has requested that you create an application which will assist its members to calculate and display the monthly cost of a gym membership. The member will enter a name and select the type of membership requested (Gold, Silver or Bronze). The basic membership monthly fee is R300. However members get a discount on this basic fee based on their status (Table 1). There are also optional extra services that a member can request and the cost of this will be added to their monthly subscriptions (Table 2).

Status	Discount Percent
Gold	10
Silver	7
Bronze	5

Table 1

Extra Service	Cost per month
Personal Trainer	150
Baby Club	200

Table 2

Write a C# code for the model called Camp and show all the necessary properties and methods that suits the above scenario. Demonstrate all necessary namespaces where it is required.

Description of methods:

Description of methods:

- CalcDiscount(): uses the basic monthly membership fees and the discount percent (Table 1) to calculate the discount.
- CalculateFinal(): uses CalcDiscount() and the values of the extra services to calculate the final monthly membership fees due.
- You are required to write a C# code for the controller named Camping in order to demonstrate how the controller facilitates inputs from the view, their processing in line with the model.
- Write a RAZOR view syntax (code) for the following view.

Full SCENARIO 11 - Revision Higher Education App

The DUT is a higher education institution situated in Durban, with various campus in and around Durban and Pietermaritzburg. DUT has three different types of security clusters at its disposal i.e. body guards, strike force and gate keepers. Out of these clusters, DUT can decide to have one, two or all of the clusters on duty at any day. The securities are paid based on the number of hours they worked which can be 8, 10 or 12. These clusters are paid at different rate depending on the type of job they are doing e.g. body guard are paid at a higher rate followed by strike force and then gate keepers. If a cluster work overnight, they get paid an additional quarter of the entire income made on that day. Note that not all clusters can work overnight. DUT requires a security management system that will help them to manage and predict the costing of hiring these clusters.

Code the entire ASP.NET MVC application for this system, use the following as guidance:

- Carefully plan and design the view you would use for this application (design it on paper).
- Code a method that will calculate the total expense based on all the security clusters.
- Code a method that will determine the highest paid cluster.
- Code a method that will determine the second highest paid cluster.
- Given a budget of 1 million determine how many days can DUT be able to pay all the security clusters assuming that all clusters are working daily.
- Code a method that will determine how much will be needed to pay all the security cluster for the whole year.
- Code Test Cases for all the methods in this application.

NOTE: There are no costs / values given in this scenario and this is done deliberately. Methods can be easily coded without any values.

Full SCENARIO 12

A new recycling company called AG Recycling has opened in the Durban area. Recently, AG Recycling has received funds from a European investor to assist public schools both in urban and rural areas of KZN to earn money through recycling. The schools can recycle the following item types: paper, cans and ink cartridges/toner. These items will be collected and submitted to AG Recycling either on weekly or monthly basis.

Currently, AG Recycling has no system in place that keeps records of recycled items. They require a software developer to develop a web-based system that will assist each school in determining how much they will earn for their recycling initiatives.

The user must provide the necessary information as per Figure 2.1. If the user opts for the weekly collection, there is a charge of R30 for each collection; however, the companies recycling trucks collect recyclables monthly for free from all schools in KZN. Table 2.1 below shows the items that can be recycled together with their earnings in each area. A school must recycle a minimum of 20 items per type.

Item Type	Urban	Rural
Paper	90 cents (per kg)	2 times urban amount
Can	50 cents (per kg)	3 times urban amount
Ink cartridge/toner	R2 (per toner)	5 times urban amount

Table 1 Items

- a) Write a C# code for the model called SRecycle and show all the necessary namespaces, data annotation, properties and methods that suits the above scenario. Use Figure 1 as a guide and make sure that methods enable for the following :
 - a. Code the method that will calculate the earnings based on the types of items being recycled.
 - b. Code the method that will calculate the earnings based on the collection type.
 - c. Code the method that will calculate total earnings due.
- b) Write a C# code for the controller named SRecyclingController to demonstrate how the controller facilitates inputs from the view, their processing in line with the model.
- c) Write a RAZOR view syntax (code) for the following view as shown in Fig 1.
- d) Write the test methods for the following:
 - a. Test method for basic amount and amount based on the types of items cost.
 - b. Test method for school collection cost.

AG Recycling

School Name: MerryHill Primary

School Email: MerryHillPrimary@gmail.com

School Collection Type: Rural ☒ Urban ☐

School Collection Type: Weekly

Recycling Items

Cans ☒ 20

Paper ☒ 20

Ink Cartridge/Toner ☒ 20

Create

Type Cost	R 266
Collection Cost	R 30
Total Due	R 236

[Back to List](#)

Figure 1 – AG Recycling Data Annotations and View.

Full SCENARIO 13

Thuma Mina Foods (TMF) is a buffet and a “prokard” restaurant. A prokard is a reward programme where potential customers can subscribe to and be entitled to exclusive discounts when they dine in prokard-associated restaurants across the country. Customers that are part of the prokard programme can go to any participating buffet restaurant, such as TMF, to dine in or have a take-away. If two prokard-customers are dining in the buffet, they get a discount of half the price of the buffet cost. If they are five prokard-customers who are set to dine in, they receive a discount which is a quarter of the buffet cost and if they are more than five they get a 15% discount of buffet cost. Furthermore, there is a compulsory tip based on the number of customers that are dining, see Table 1 below. Moreover, there is an addition cost of 2%, of the total buffet cost, if a customer opts to dine in the restaurant and the same percentage discount is eligible for a customer that chooses to have a take-away. Non-prokard-customers can buy food and dine in or have a take away at normal buffet cost.

As a C# programmer, you are tasked to develop an ASP.NET MVC application that will work out amount due when a customer buys food from any Prokard associated restaurant.

Write a C# code to complete this application as shown in Figure 1 and copy and paste ONLY the code that shows the workings for the following.

- A method that calculates basic food cost.
- A method that calculates a tip due to waitrons.
- A method that calculates a discount due to the customer.
- A method that calculates a final cost due to a customer.
- A unit test method for question [1.2] above.
- A unit test method for question above.

- g) Code the controller for this application and show an action method that will be responsible for rendering the view to capture registration details and an action method that will serve the purpose of computing all necessary calculations and display them.
- h) Write the code for the View in Figure 1.

No of Customers	Tip
Equal to 2	50% of the original food cost
Greater than 2 but less or equal to 5	Quarter of the original food cost
Greater than 5	15% of the original food cost

Table 1 – Tips to be paid by Customers

TMF Restaurant

Paying Customer Name

Number of Customers

Prokard Member

☒

Seat In

☒

Buffet Cost

Basic Buffet Cost

R 375

Tip Amount

R 75

Disc / Seat Cost Amount

R 9

Final Cost

R 459

Figure 1 – TMF Restaurant APP in a running mode.

Full SCENARIO 14

A new recycling company called SR recycling has opened up in the Gauteng area. Recently, SR recycling has received funds from an Egyptian investor to help KwaZulu-Natal public hospitals both in urban and rural areas to earn money through recycling. Hospitals earn money if they collect disposable medical devices for recycling. The transporting team of SR recycling either on weekly or monthly basis collects these devices from each hospital. SR recycling collects from all participating hospitals free of charge on monthly basis. However, if an urban-based hospital or a rural-based hospital requires collection on a weekly base there is a penalty of 5% and 15% respectively.

No of Items	Urban Hospital	Rural
Below 99	Not applicable	Quarter of basic cash reward
Between 100-499	Not applicable	Two quarters of basic cash reward
Above 499	Not applicable	Three quarters of basic cash reward

Table 1

Write a C# code to complete this application as shown in Figure 1 and copy and paste ONLY the code that shows the workings for the following.

- A method that calculates basic cash reward.
- A method that calculates an additional cash reward.
- A method that calculates a penalty cost
- A method that calculates a final cash reward due to a hospital.
- A unit test method for question [1.2] above.
- A unit test method for question [1.3] above.
- Code the controller for this application and show an action method that will be responsible for rendering the view to capture registration details and an action method that will serve the purpose of computing all necessary calculations and display them.
- Code the view as shown in Figure 1.

Figure 1

Full SCENARIO 15

A new hotel Hluhluwe & Safaris offers exclusive and luxury accommodation in the KwaZulu-Natal area and an authentic bush wedding venue catering for up to 180 guests. You can celebrate your union with loved ones under the African sky and enjoy day trips around the area to experience game drives, pineapple tours, whale watching and turtle tours. Hluhluwe & Safaris hotel has enough space to enjoy, especially if you are travelling with children or as couple or even with friends. Their hotels have the most accommodating rooms with flamboyant interior design. The Hluhluwe & Safaris hotel booking system is responsible for making a decision of how many bedrooms a visitor(s) needs based on the number of people that will be visiting as shown in Table 1 below. The Hluhluwe & Safaris has a restrictions on the number of visitor(s), with 10 being the maximum, this includes children.

No of Bed Rooms	Visitors	
	Adults	Children's
1	Between 1 and 2	Less than Adults
2	Between 3 and 4	Less or equal to Adults
3	Between 3 and 4	Greater or equal to Adults
4	Above 4	Above 4

Table 1 – Room allocation based on the number of visitors

Hluhluwe & Safaris hotel rooms are available in two categories, namely exclusive rooms and luxury rooms. The room cost for each exclusive room is R1700 per person, per night and luxury room is R1500 per person, per night. However, their children are charged 25% less of the room cost. Hluhluwe & Safaris offers room service which is charged at 2% of the booking cost. It is noteworthy, that visitors that stay for more than 3 nights are eligible for a 10% discount of the overall booking cost.

- a) Write a C# code to complete this application as shown in Figure 1 and show the workings for the following:
 - a. A method that determines the number of bedrooms eligible for visitors.
 - b. A method that calculates the room cost based on the number of nights.
 - c. A method that calculates the booking cost for visitors.
 - d. A method that calculates the room service cost.
 - e. A method that calculates the final booking cost.
 - f. A unit test method for question [1.1] above.
 - g. A unit test method for question [1.2] above.
- b) Provide a C# code that shows an action method that will be responsible for
- c) rendering the view to capture the necessary hotel booking details and an action method that will serve the purpose of computing all necessary calculations and display them.

Hotel

Hluhluwe & Safaris Hotel Booking

Select Room Type
Exclusive ▼

Adults
5

Children
3

Nights
2

Room Service:
☒

Number of Rooms : 4
Room Cost R: 24650
Booking Cost R: 98600
Room Service Cost R: 1972
Final Booking Cost R: 100572

Create

Figure 1 – Hluhluwe & Safaris Hotel Booking System in a running state.

Full Scenario 16

The Durban Metropolitan Police services wants a system that will work out the parking violation penalties. The basic cost for parking violation is to be entered by the user. The other penalties are based on the following criteria and percentage costs of the basic cost, see Table 1 below. If the offence was committed in the city center an additional 10 % of the basic cost, get charged.

Vehicle Type	Parking Offence
Bakkie	20%
Sedan	15%
HatchBack	10%

Table 1

Parking Offence	
Double Parking	20%
Expired Ticket	10%

Table 2

Parking Violations

Tickets

Veh_Reg
ND 146643

Park_Offence
☒ Double Parking
☐ Expired Ticket

Select Vehicle Type
Sedan ▼

Basic_Cost
200

City
☒

Veh_Type_Cost
30

Area_Cost
20

Park_Off_Cost
40

Total_Cost
290

Create

Fig1

- a) Write the Model code together with the properties and methods for the scenario above. The system must calculate and output the following costs:
- b) Cost based on vehicle type (Sedan, Bakkie or Hatchback)
- c) Cost based on parking offense (Double parking or expired ticket)
- d) Cost based on parking area (City Center or not)
- e) Total violation cost.
- f) Write the code for the Controller and View as shown in Fig 1.
- g) Write the unit test for all the methods in the model: