**Source Code:**

1. Source Code of ASP.NET Core Project: <https://github.com/b00ma/Phase-4-Project.git>
2. Source Code of Selenium Testing:

Graphical user interface, text, application, email

Description automatically generated Packages to be installed!

using System;

using System.Collections.Generic;

using System.Text;

using OpenQA.Selenium;

using OpenQA.Selenium.Chrome;

using System.Threading;

namespace SeleniumTesting

{

class Program

{

static void Main(string[] args)

{

IWebDriver driver = new ChromeDriver(@"E:\Automation\Phase4\_Project");

string url = "http://localhost:62399/Pizzas";

driver.Navigate().GoToUrl(url);

driver.Manage().Window.Maximize();

Thread.Sleep(2000);

IWebElement element1 = driver.FindElement(By.Id("details"));

Thread.Sleep(2000);

element1.Click();

Thread.Sleep(1000);

IWebElement element2 = driver.FindElement(By.Id("order"));

Thread.Sleep(2000);

element2.Click();

Thread.Sleep(1000);

IWebElement element3 = driver.FindElement(By.Id("list"));

Thread.Sleep(2000);

element3.Click();

Thread.Sleep(10000);

driver.Quit();

}

}

}

1. Source Code for NUnit Testing:

Graphical user interface, application

Description automatically generatedPackages to be installed & library classes to be created!

using NUnit.Framework;

using DataAccessLogicLayer;

namespace NUnitTesting

{

public class Tests

{

[SetUp]

public void Setup()

{

}

[Test]

public void Test1()

{

//Assign or Arrange

int pid = 2001;

string ptype = "Peppy Paneer";

//Act

string acname = PizzaHelper.GetENameById(pid);

//Assert

Assert.AreEqual(ptype, acname);

Assert.Pass();

}

[Test]

public void Test2()

{

//Assign or Arrange

int pid = 4002;

string ptype = "Fresh Veggie";

//Act

string acname = PizzaHelper.GetENameById(pid);

//Assert

Assert.AreEqual(ptype, acname);

Assert.Pass();

}

[Test]

public void Test3()

{

//Assign or Arrange

int pid = 3001;

string ptype = "Cheese & Corn";

//Act

string acname = PizzaHelper.GetENameById(pid);

//Assert

Assert.AreEqual(ptype, acname);

Assert.Pass();

}

[Test]

public void Test4()

{

//Assign or Arrange

int pid = 5001;

string ptype = "Peppy Paneer";

//Act

string acname = PizzaHelper.GetENameById(pid);

//Assert

Assert.AreEqual(ptype, acname);

Assert.Pass();

}

[Test]

public void Test5()

{

//Assign or Arrange

int pid = 2002;

string ptype = "Peppy Paneer";

//Act

string acname = PizzaHelper.GetENameById(pid);

//Assert

Assert.AreEqual(ptype, acname);

Assert.Pass();

}

}

}

1. Jenkins:

Once Jenkins is connected to server,

Jenkins is available in localhost:8080 and give password : 6a21e9671d92481fa7a390060ec8c139

Setup: install suggested plugins

Admin1

Username: soppia

Password: soppia

Full Name: A Soppia

Email address: [soppia@gmail.com](mailto:soppia@gmail.com)

1. Deployment in Azure:

VM is created in Azure where Jenkins is installed and worked upon!

Create new app services and publish the ASP.NET project.

(Steps are shown in Screenshots)