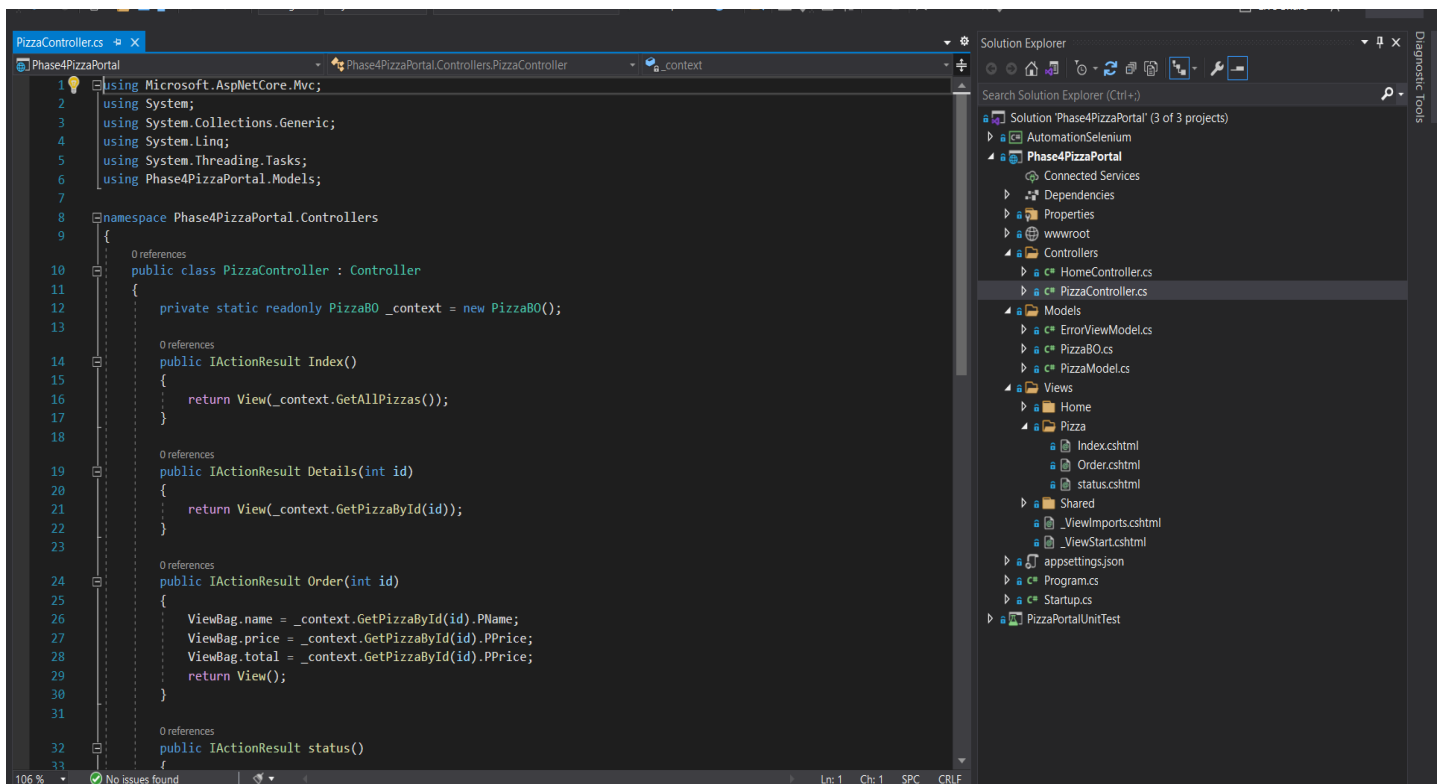


SCREENSHOTS

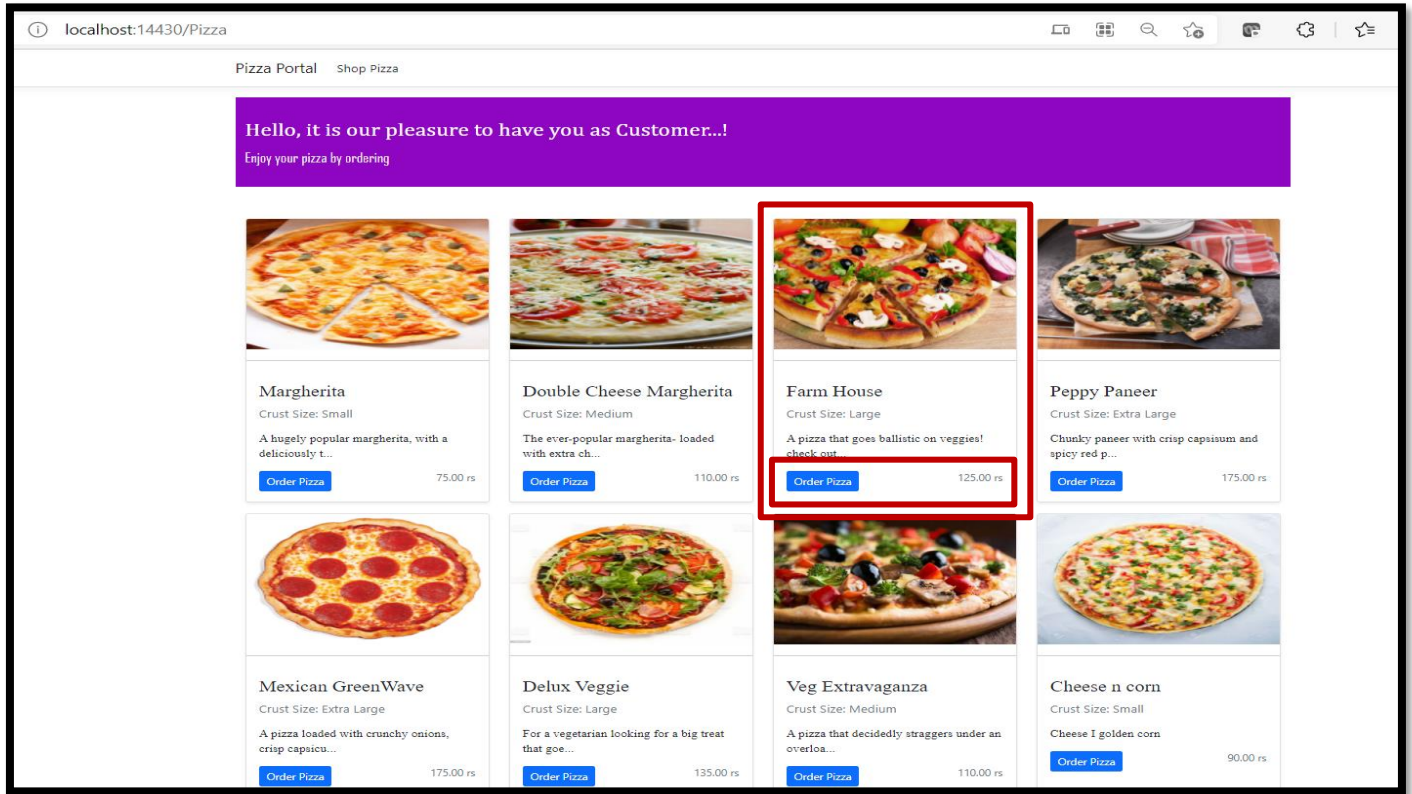
Submitted by
Ramya K T
Software Engineer
Dover India Private Limited

A. Pizza Portal Project

1. Below image shows the project view



2. Run the project, which opens the website in the localhost. Website shows all the pizzas available and select a pizza to order



3. This image shows the bill and checkout form. Enter all the details and click on the button

The screenshot shows a web browser at localhost:14430/Pizza/Order/3. The page is titled "Bill & Billing Details". It contains a form for billing information and a summary of the order. The "Billing address" section includes fields for First name (Ramya), Last name (K T), Email (ramya123@gmail.com), Address (ABCD, EFGH, India), and ZipCode (741258). The "Payment" section has radio buttons for Cash on Delivery (selected), Credit card, Debit card, and PayPal. The "Your Pizzas" section shows a summary of the order: Farm House, Qty: 1, Total (Rs) 125. The "Continue to checkout" button is highlighted with a red box.

Billing address

First name: Ramya
Last name: K T
Email: ramya123@gmail.com
Address: ABCD, EFGH, India
ZipCode: 741258

Payment

☒ Cash on Delivery
☐ Credit card
☐ Debit card
☐ PayPal

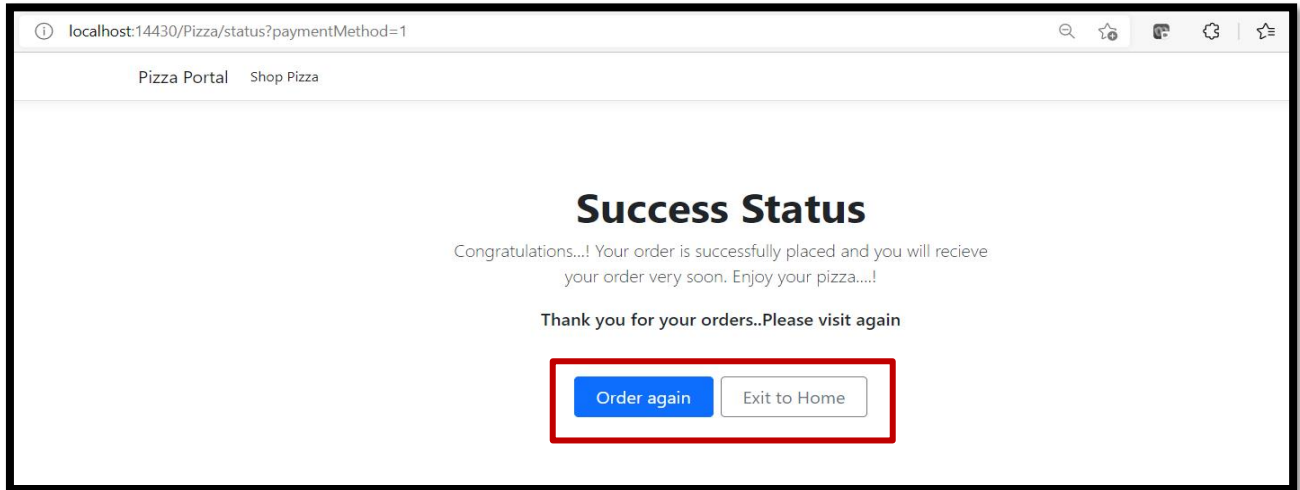
Your Pizzas

Pizza Name	Qty	Price
Farm House	1	125

Total (Rs) 125

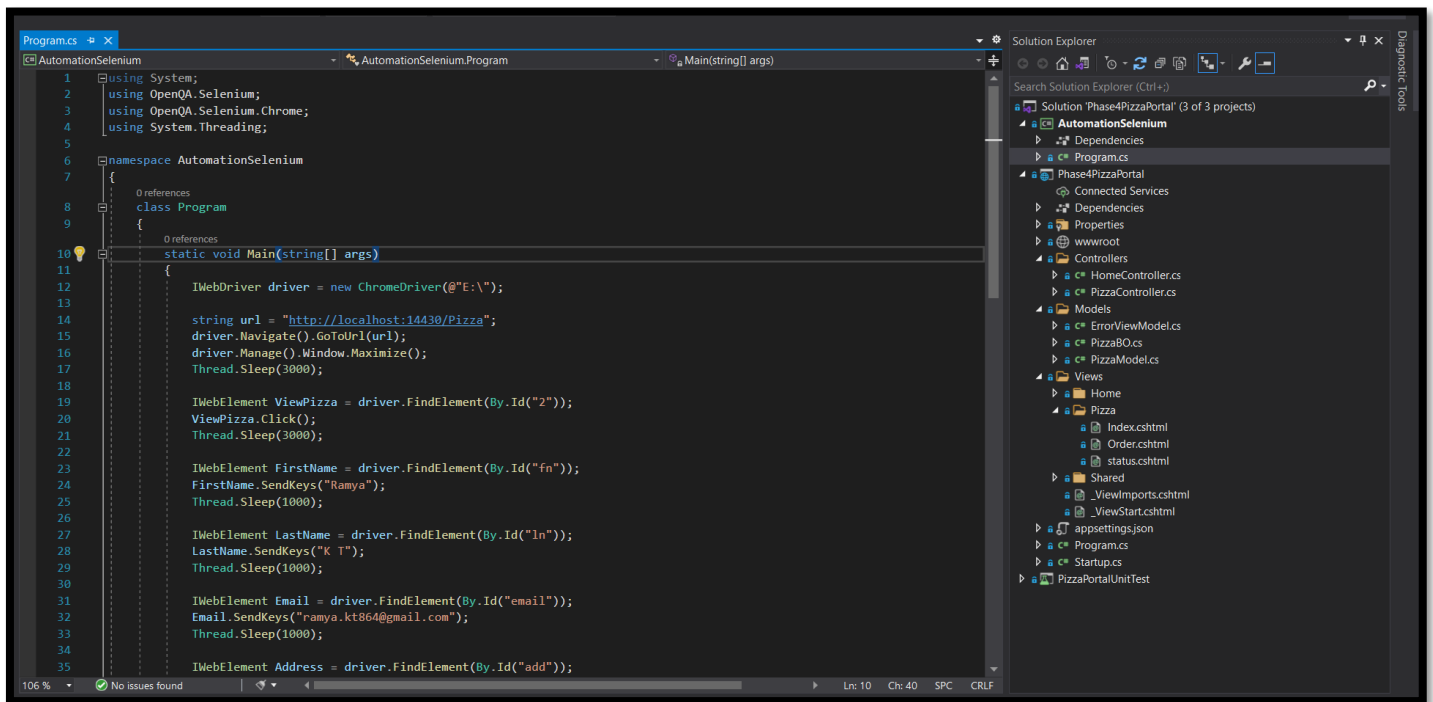
Continue to checkout

- The below image shows the status of order. Customer can either go to home page or can go to order another pizza

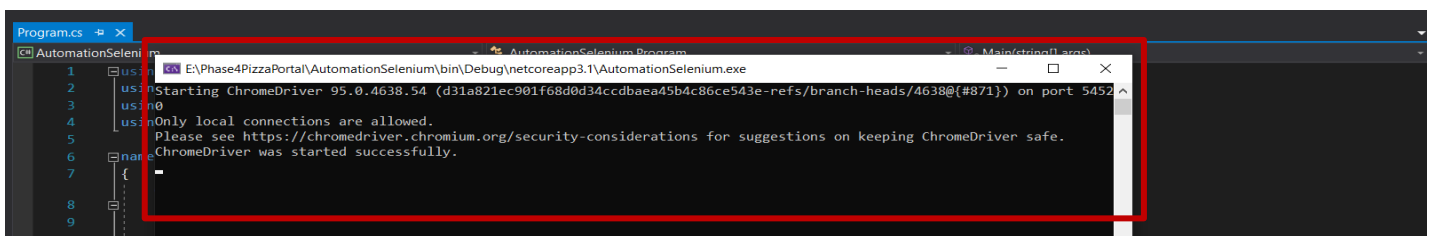


B. Project Automation using selenium

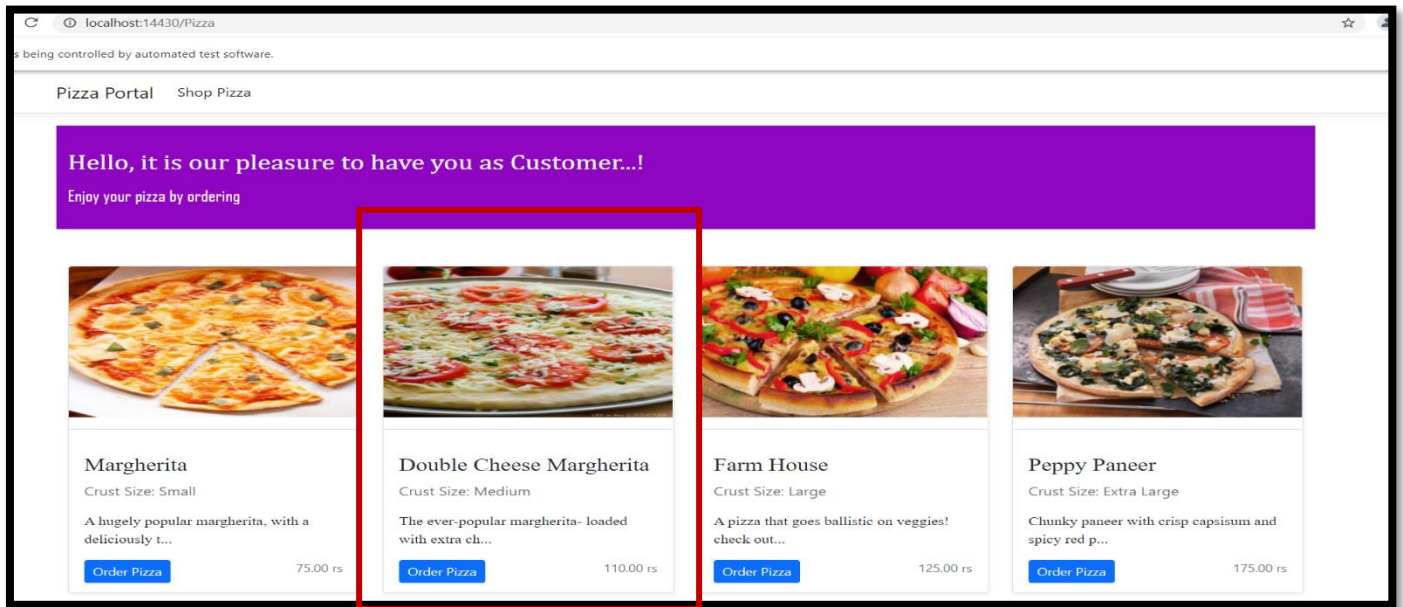
- Image shows the project code for automation. Complete code is presented through github link.



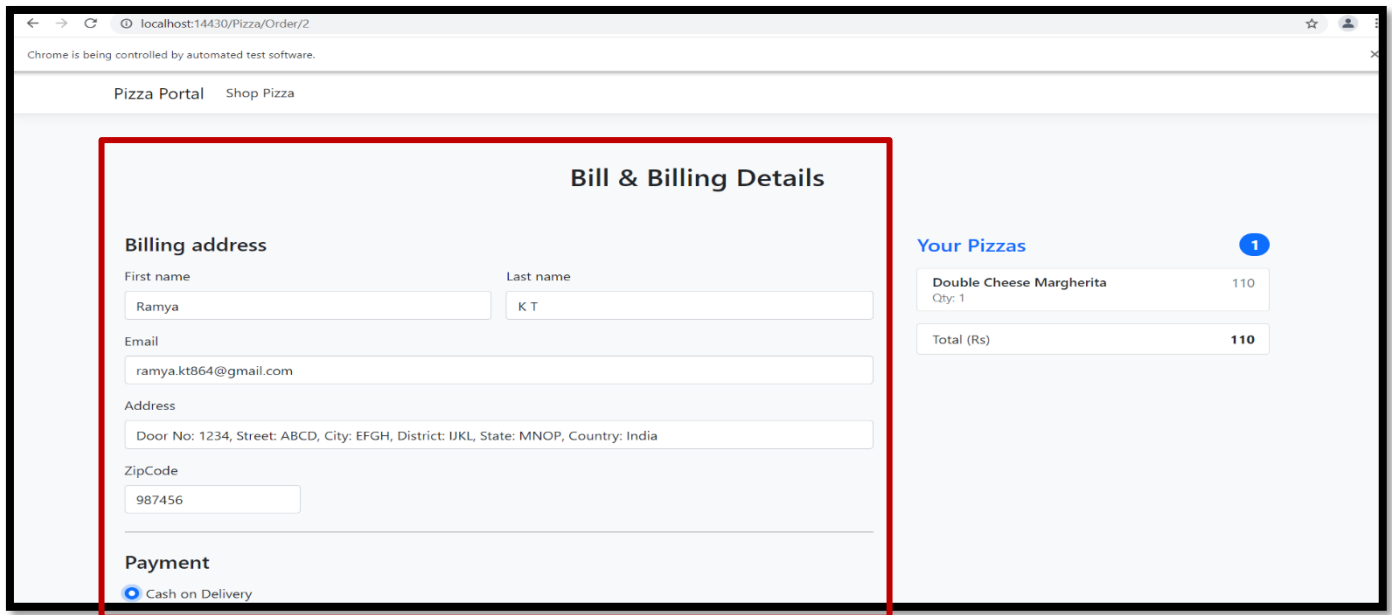
- Console opens when run button is clicked and go to url specified



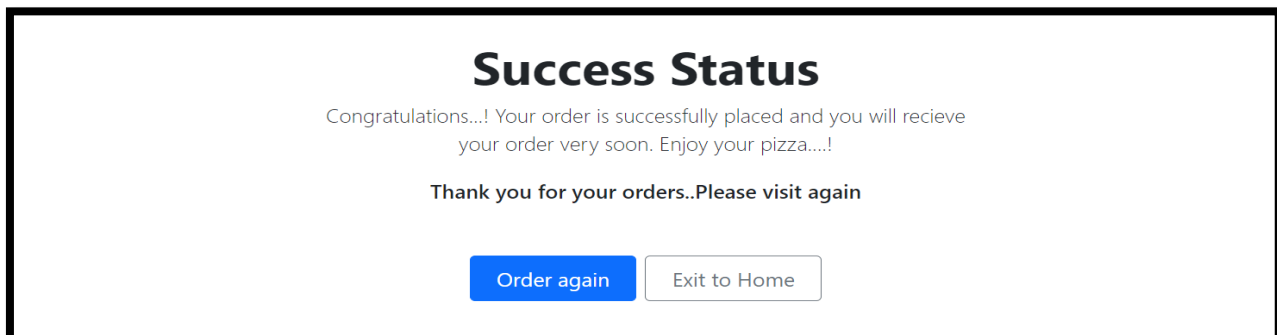
3. The windows get maximized and select pizza mentioned as per ID



4. The window takes through the billing page and inputs the value as given in the code



5. The window shows the order status page

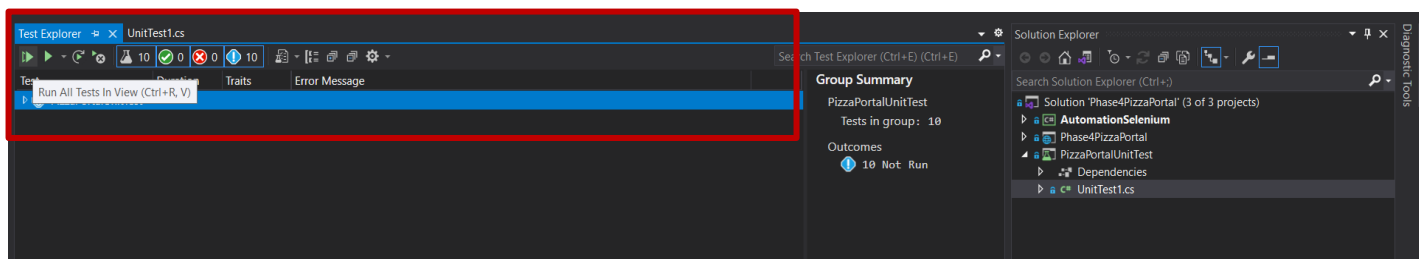


6. Console output after successful execution of automation

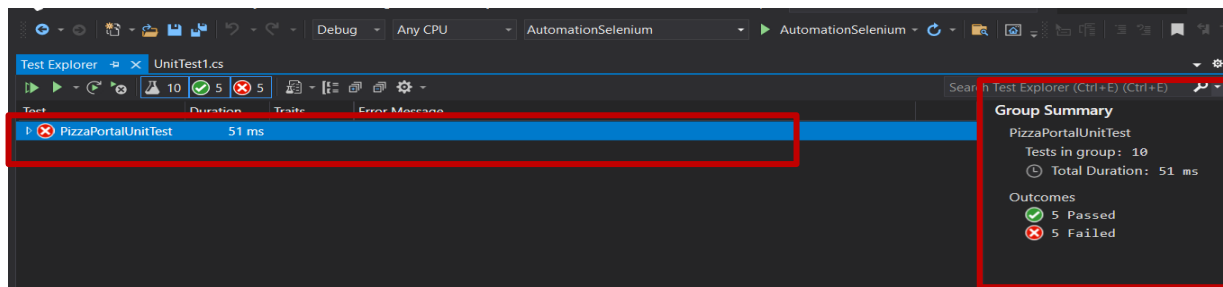
C. Project unit testing using NUnit

1. The image shows the Unit test project

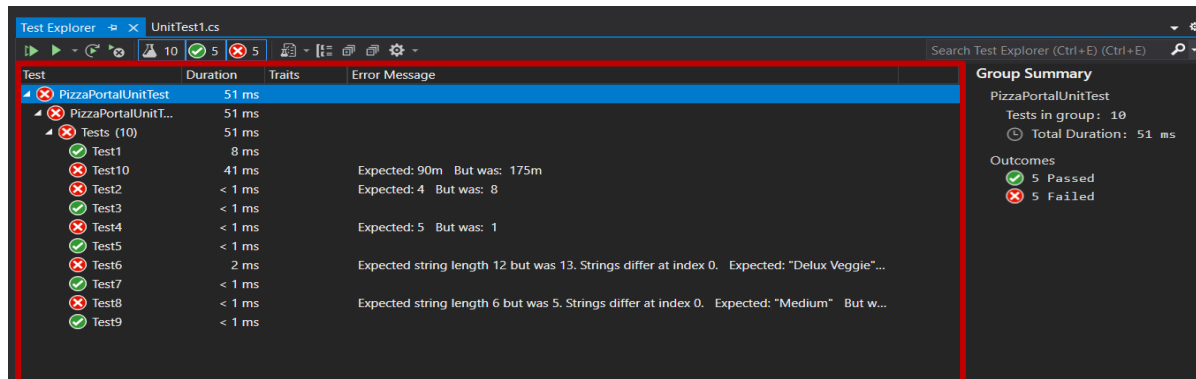
2. Test explorer is opened to run the test and click on run all tests



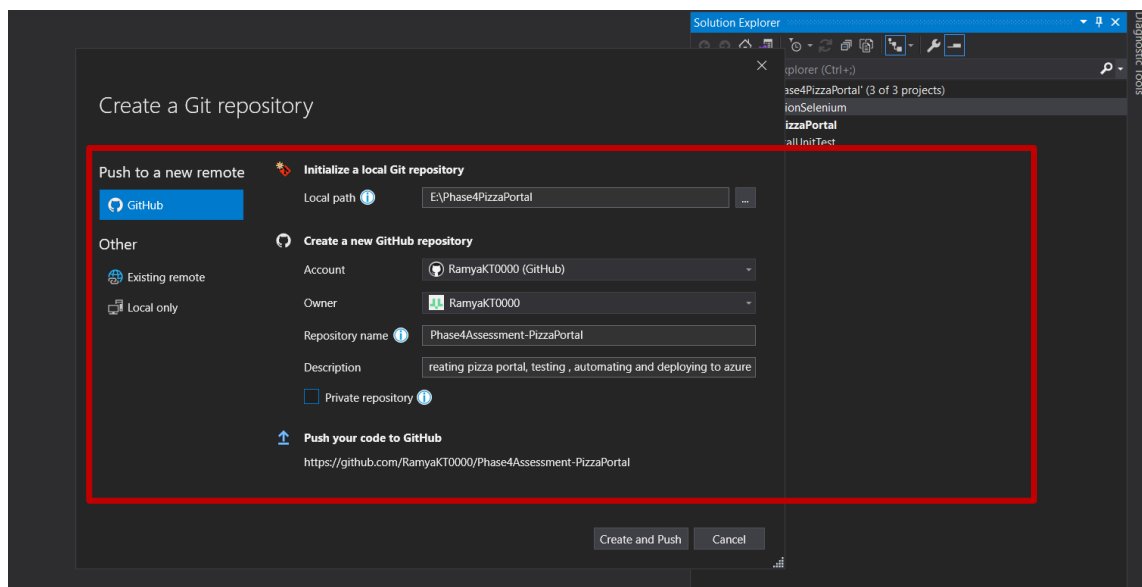
3. The image shows number of test cases passed and number of test cases failed



4. The image shows detailed view of test cases results



D. Push whole project to GitHub



E. Add a build in Jenkins

1. Create a resource group in Azure

Microsoft Azure

Search resources, services, and docs (G+/)

Home > Resource groups >

Create a resource group

Validation passed.

Basics Tags Review + create

Basics

Subscription Azure Pass - Sponsorship

Resource group phase4assessment-rg

Region Central US

Tags

None

Create < Previous Next > [Download a template for automation](#)

2. The below image shows the VM creation

Microsoft Azure

Search resources, services, and docs (G+/)

Home > Virtual machines >

Create a virtual machine

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * Azure Pass - Sponsorship

Resource group * phase4assessment-rg [Create new](#)

Instance details

Virtual machine name * vm-for-phase4 ✓

Region * (US) Central US

Availability options No infrastructure redundancy required

Security type Standard

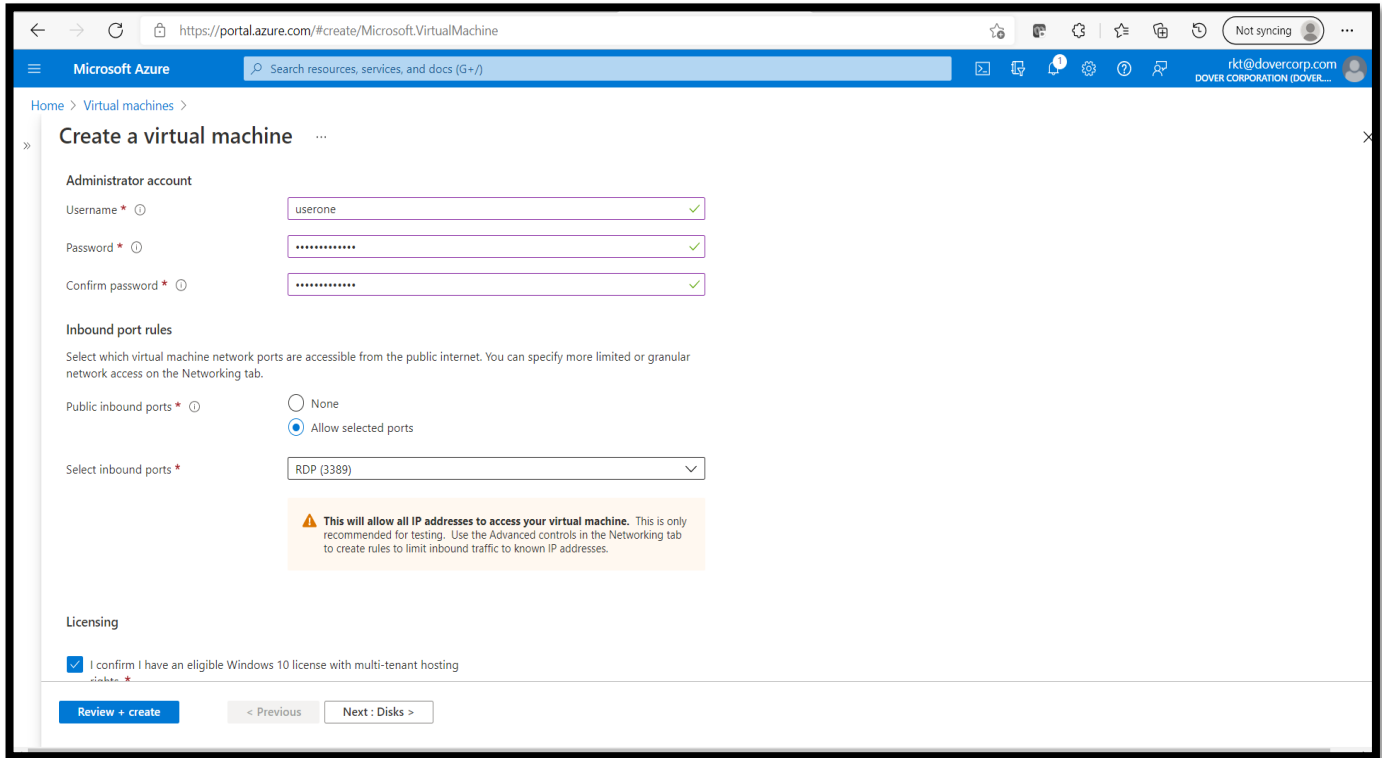
Image * Windows 10 Pro, Version 20H2 - Gen2 [See all images](#) [Configure VM generation](#)

Azure Spot instance ☐

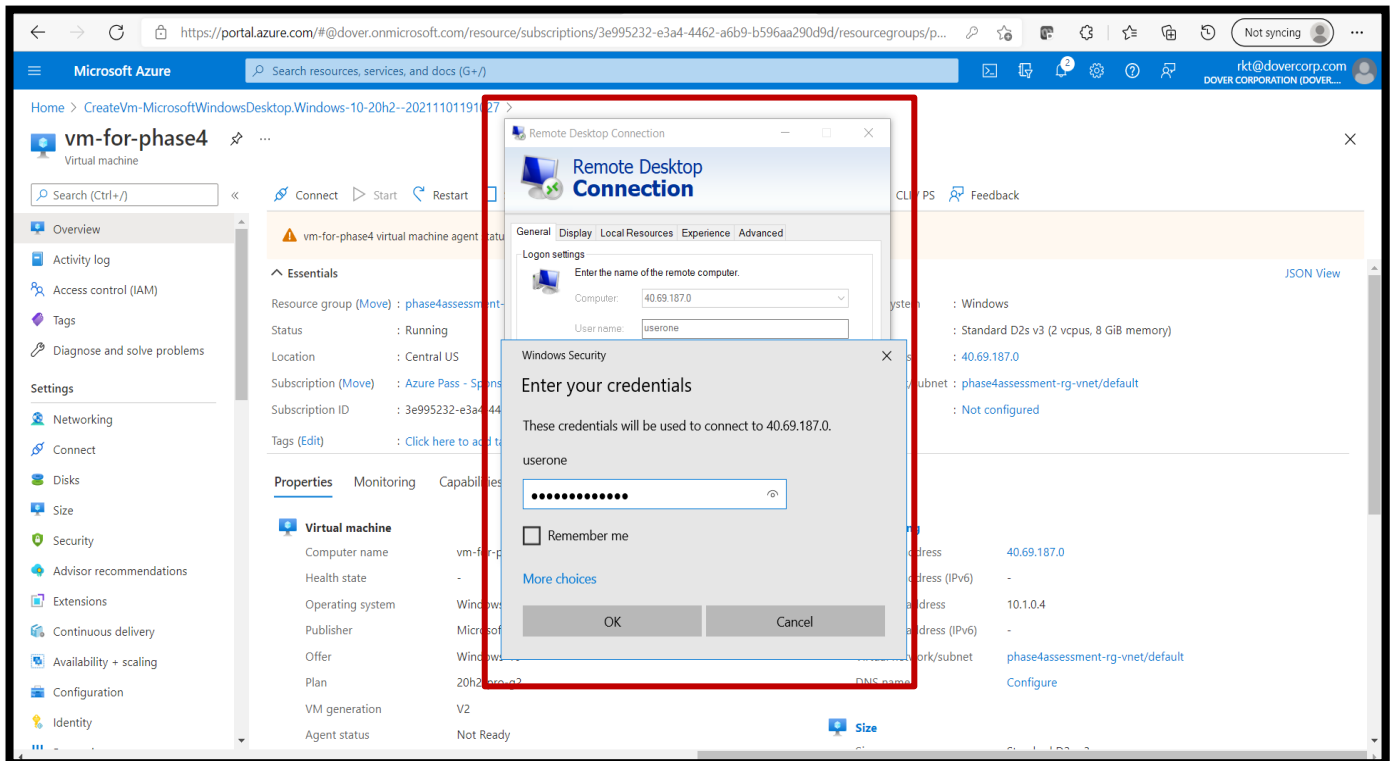
Size * Standard_DS2s_v3 - 2 vcpus, 8 GiB memory (₹5,785.23/month) [See all sizes](#)

Review + create < Previous Next: Disks >

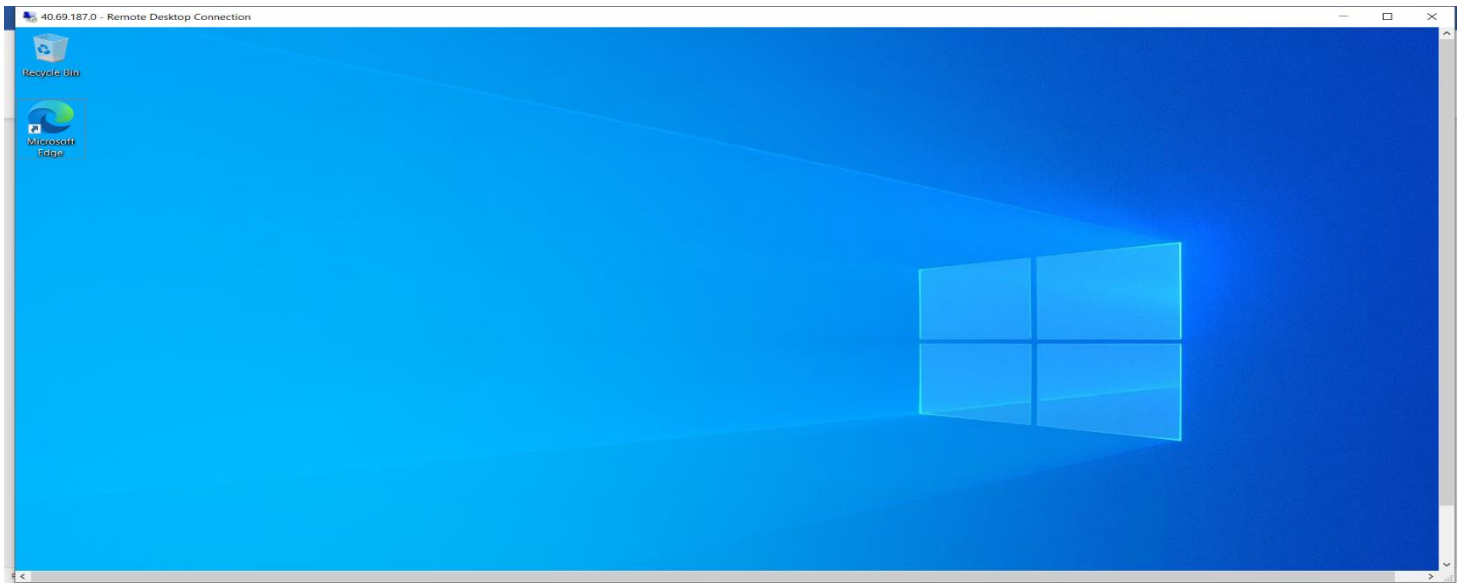
3. The below image gives details of VM



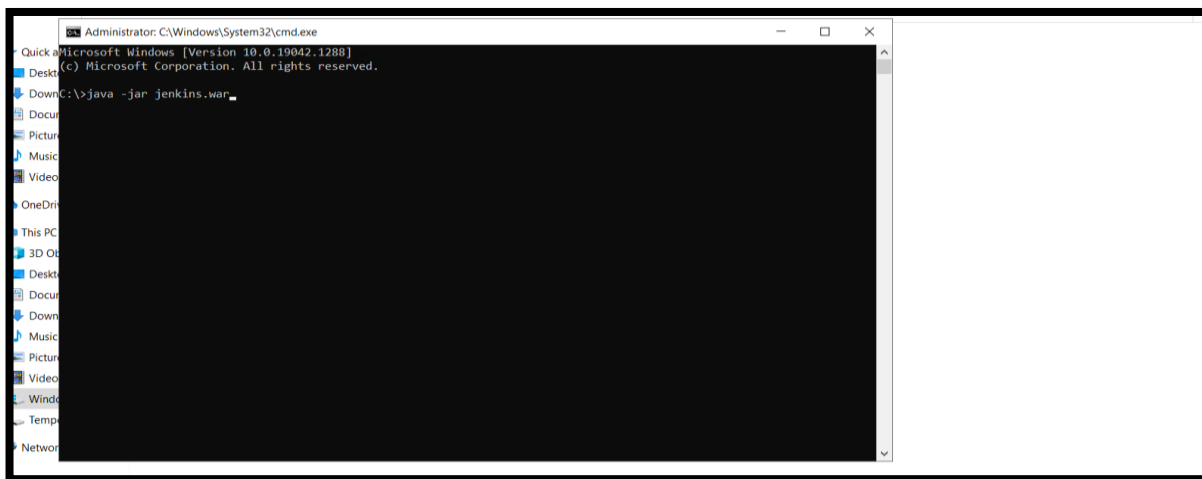
4. Login to the VM using remote desktop



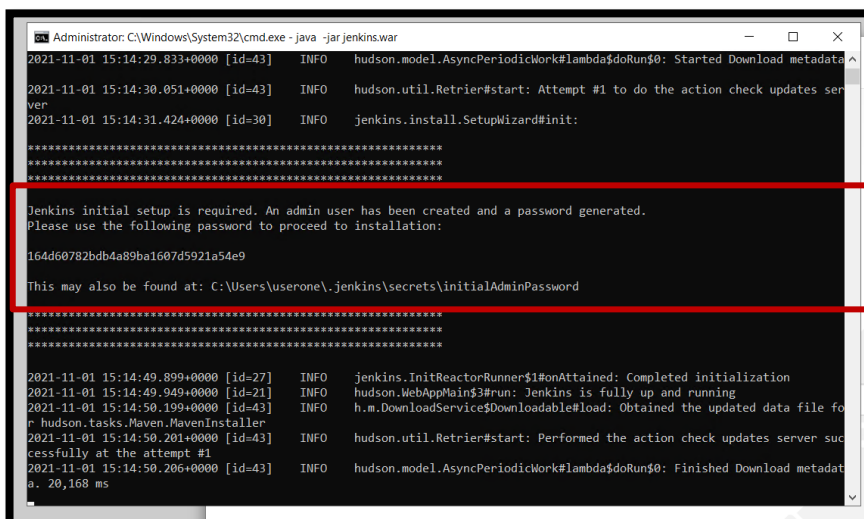
5. The below image shows the VM logged in



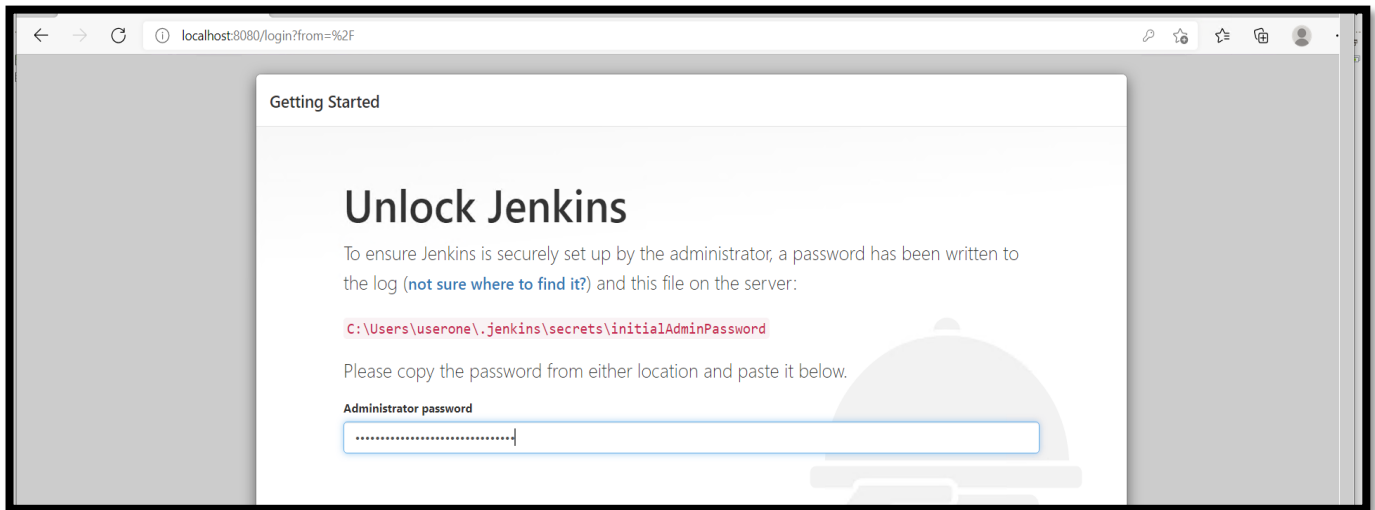
6. Install Jenkins, JDK, SDK, And in CLI give command to run Jenkins



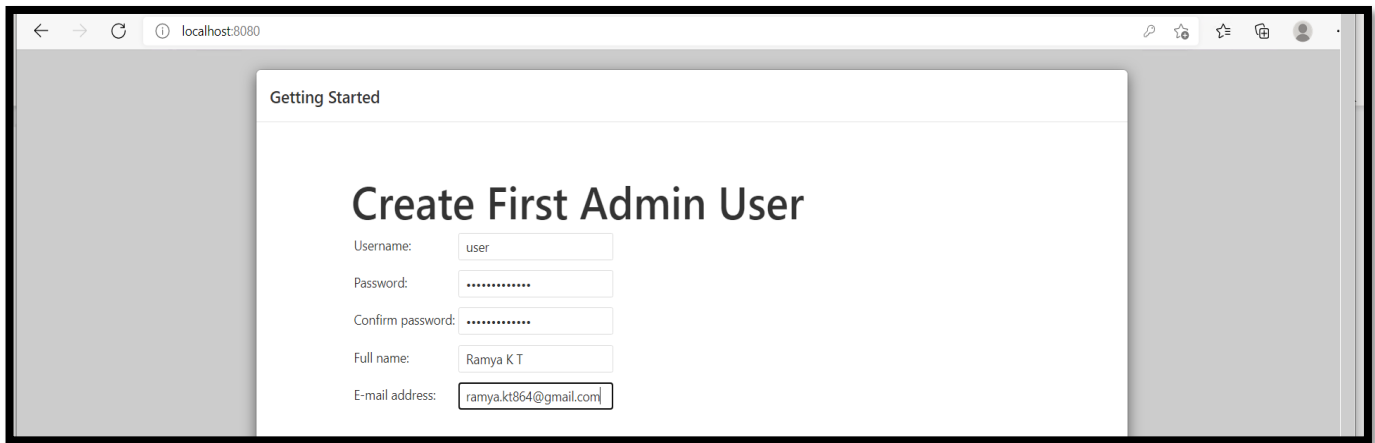
7. Copy the administrator password to validate



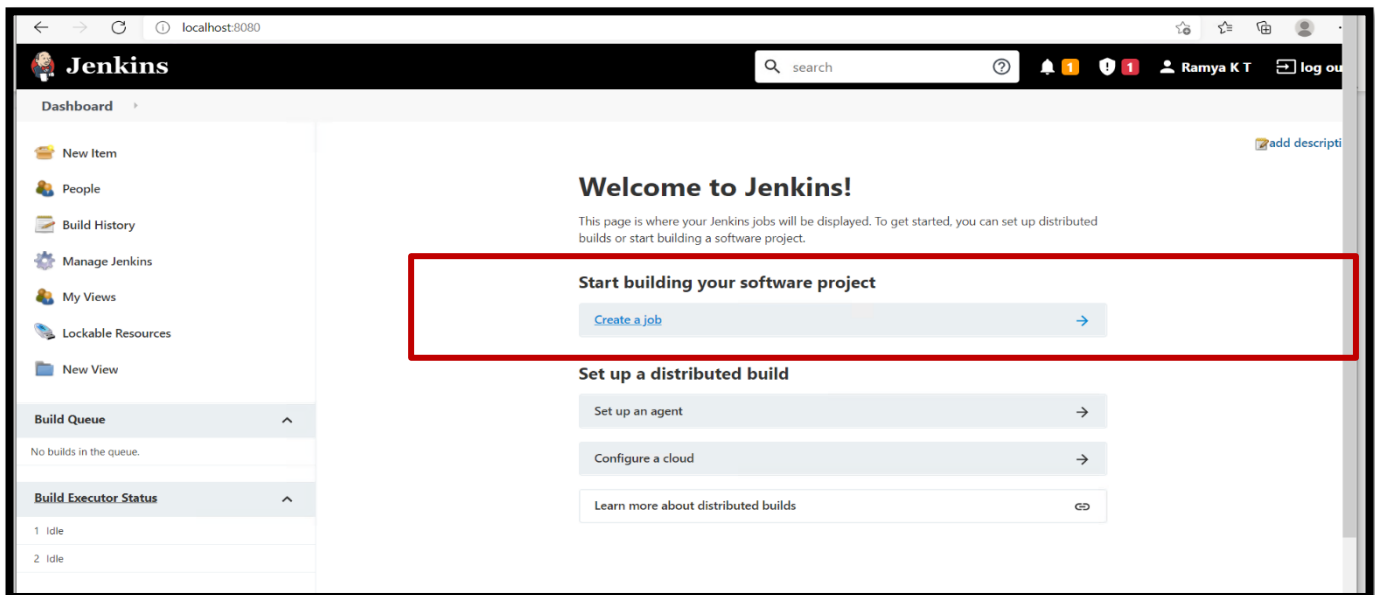
8. Run the localhost and paste the administrator password



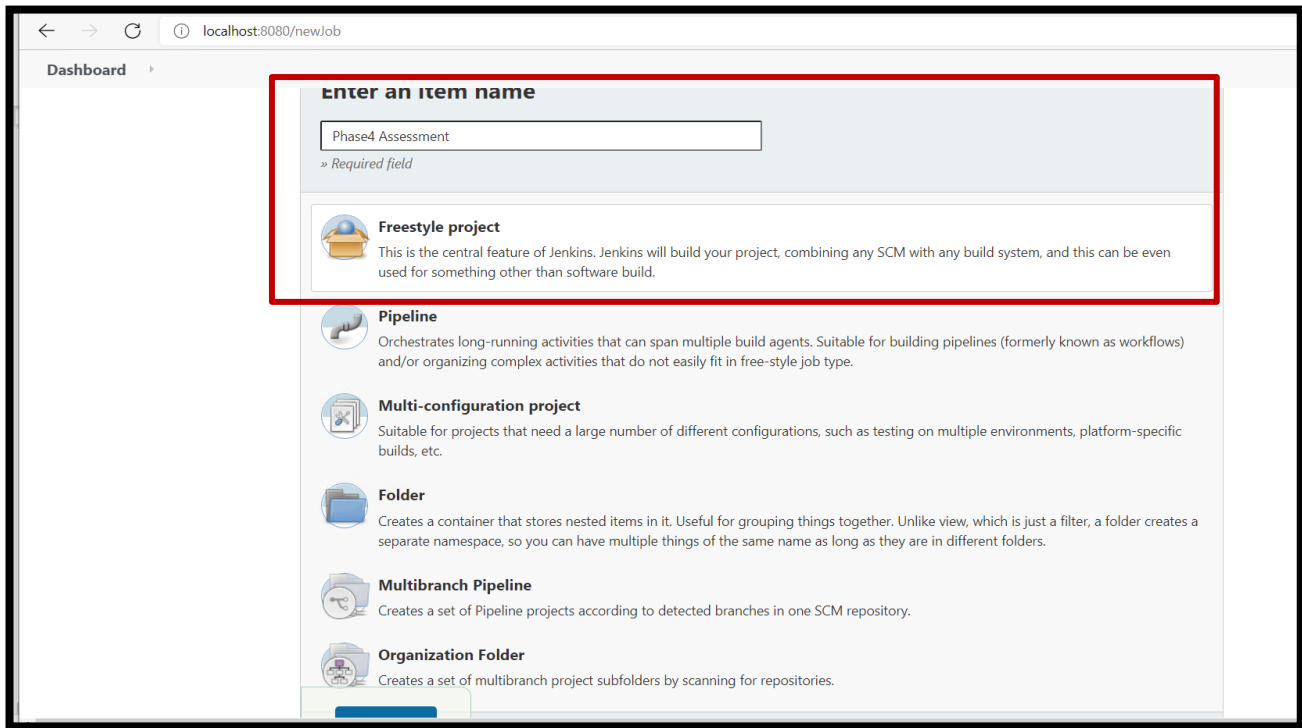
9. Create a new user



10. Select new job to create new job



11. Provide name for new job and select free style



The screenshot shows the Jenkins 'Enter an item name' dialog. A red box highlights the input field containing 'Phase4 Assessment' and the 'Freestyle project' option. Below the input field, a list of project types is shown: Freestyle project, Pipeline, Multi-configuration project, Folder, Multibranch Pipeline, and Organization Folder. Each option has a brief description.

localhost:8080/newJob

Dashboard

Enter an item name

Phase4 Assessment
» Required field

Freestyle project
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

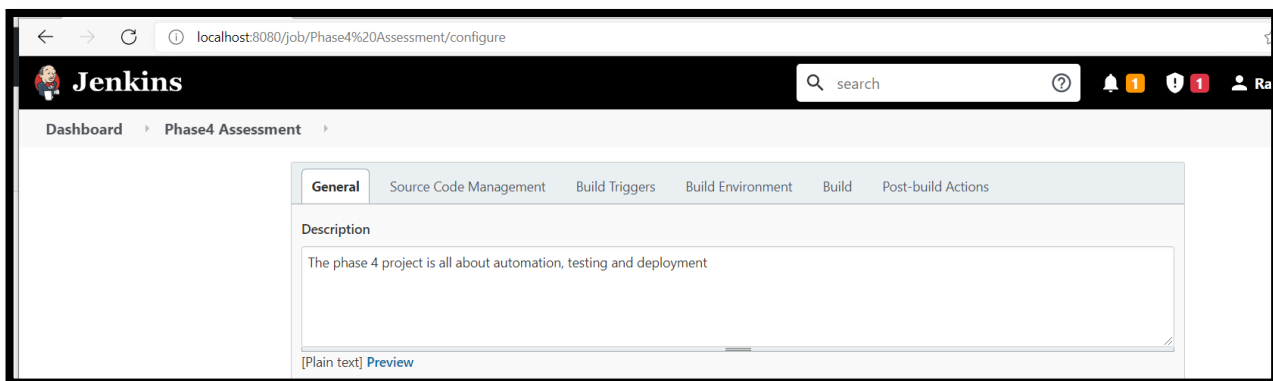
Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.

Organization Folder
Creates a set of multibranch project subfolders by scanning for repositories.

12. Add description for the project



The screenshot shows the Jenkins 'Configure' page for the 'Phase4 Assessment' job. The 'General' tab is selected, and the 'Description' field is filled with 'The phase 4 project is all about automation, testing and deployment'. The 'Source Code Management' tab is also visible.

localhost:8080/job/Phase4%20Assessment/configure

Jenkins

search

Dashboard > Phase4 Assessment

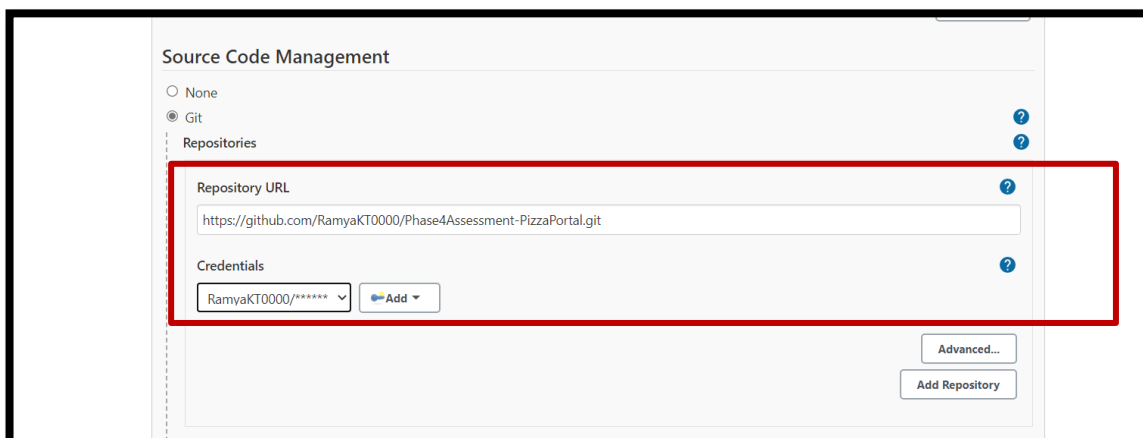
General Source Code Management Build Triggers Build Environment Build Post-build Actions

Description

The phase 4 project is all about automation, testing and deployment

[Plain text] [Preview](#)

13. Provide the github link to clone



The screenshot shows the 'Source Code Management' configuration page. A red box highlights the 'Repository URL' field, which contains 'https://github.com/RamyaKT0000/Phase4Assessment-PizzaPortal.git'. The 'Credentials' field is also visible, showing 'RamyaKT0000/*****' and an 'Add' button.

Source Code Management

☐ None
☒ Git

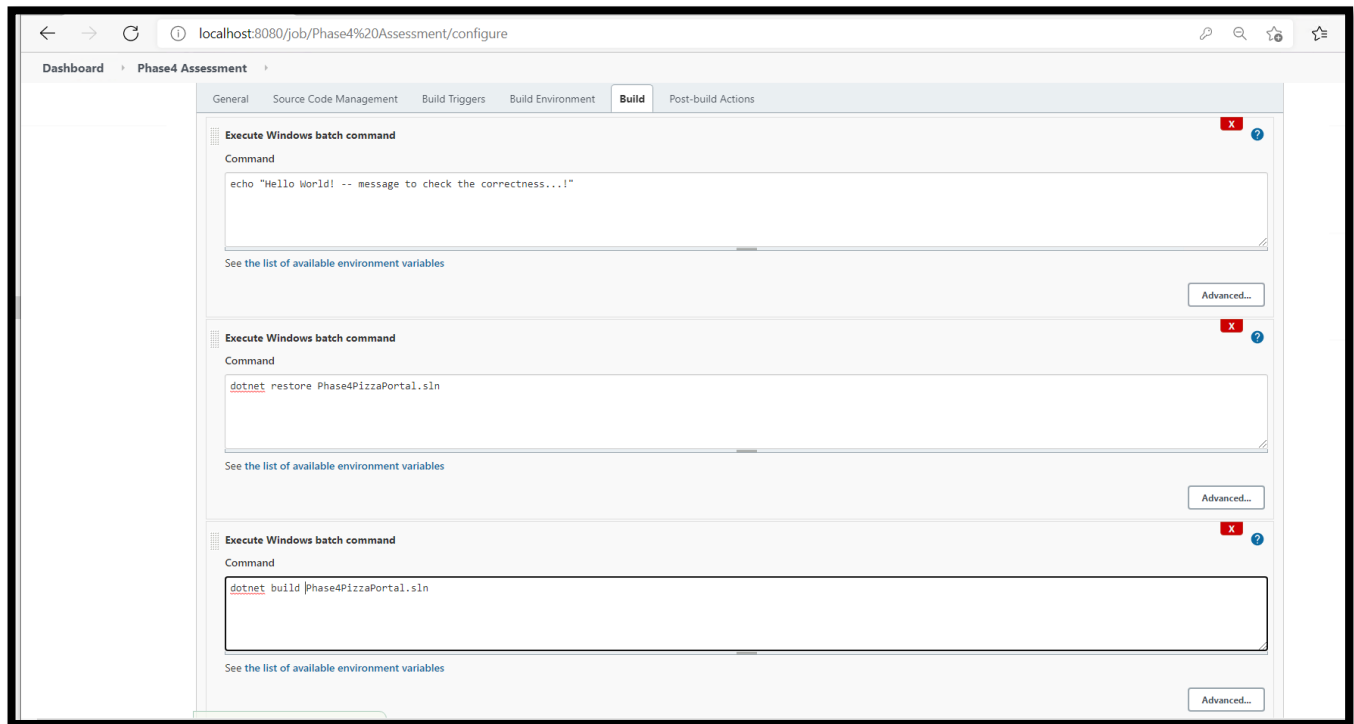
Repositories

Repository URL
https://github.com/RamyaKT0000/Phase4Assessment-PizzaPortal.git

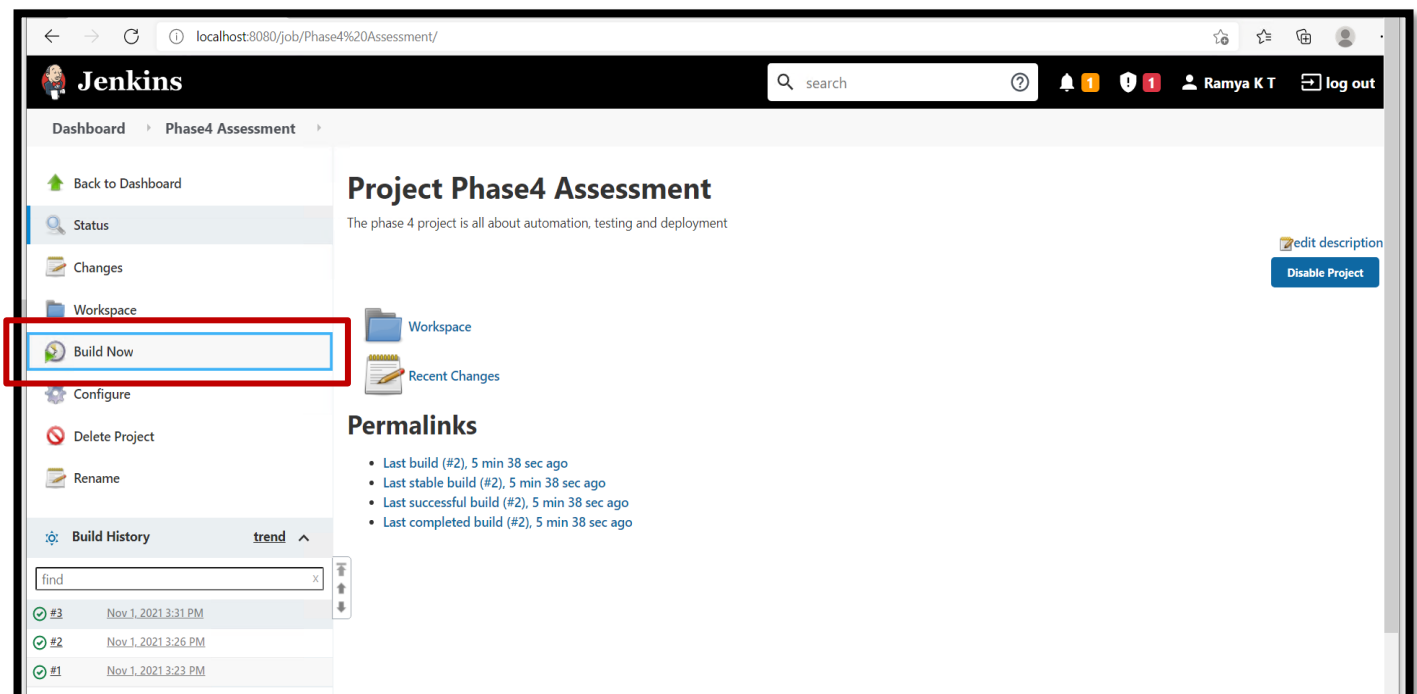
Credentials
RamyaKT0000/***** Add

Advanced...
Add Repository

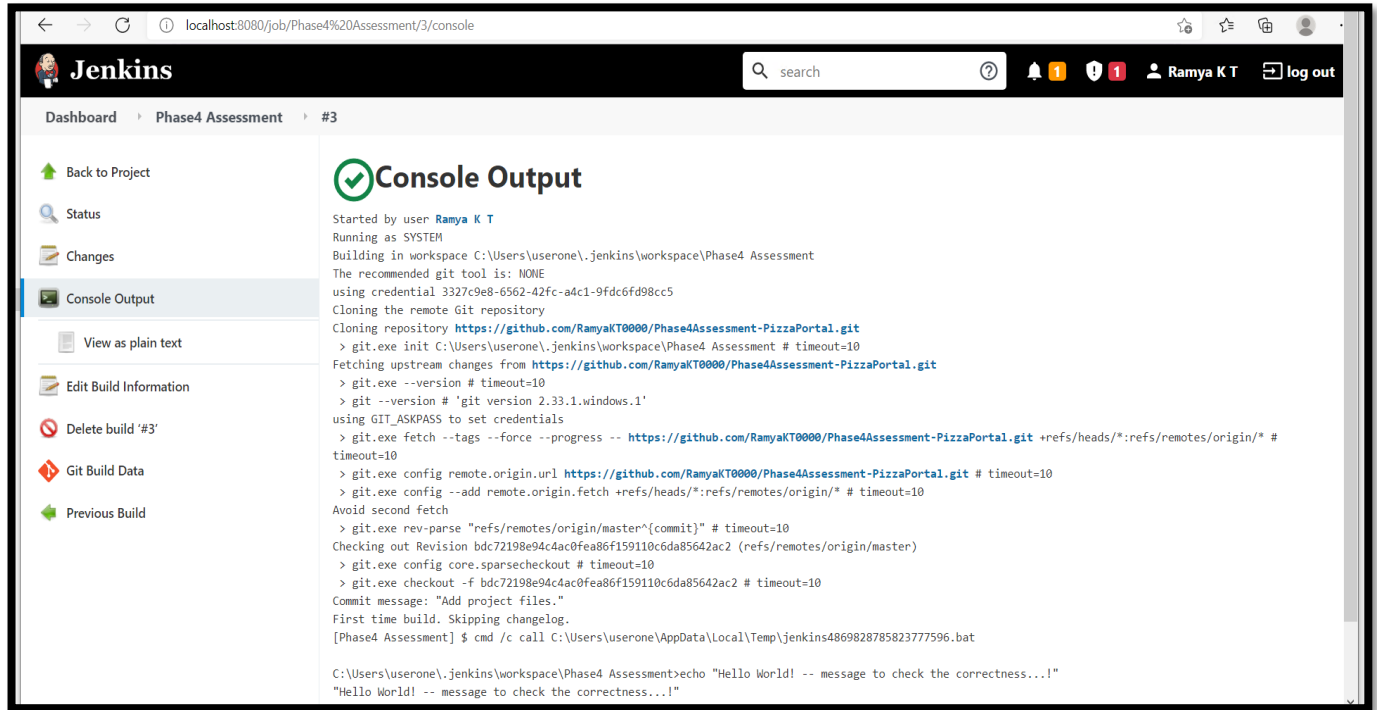
14. Provide the dotnet commands to build the project



15. Click on build now to build the project



16. The image shows the console output 1

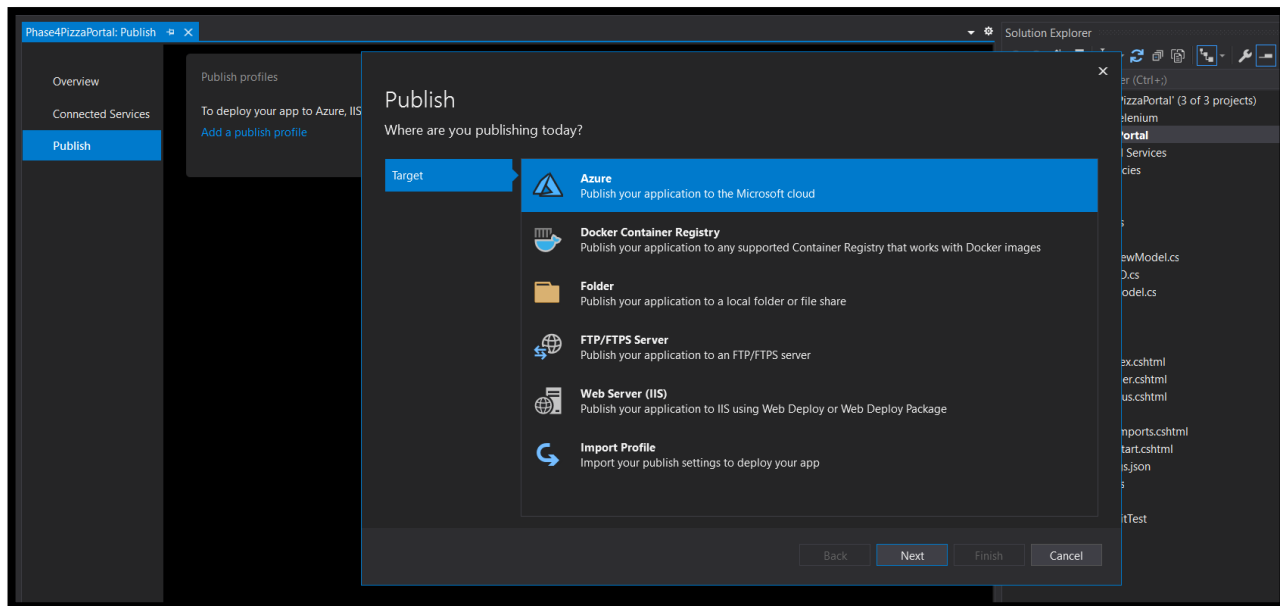


17. The image shows the Console output 2

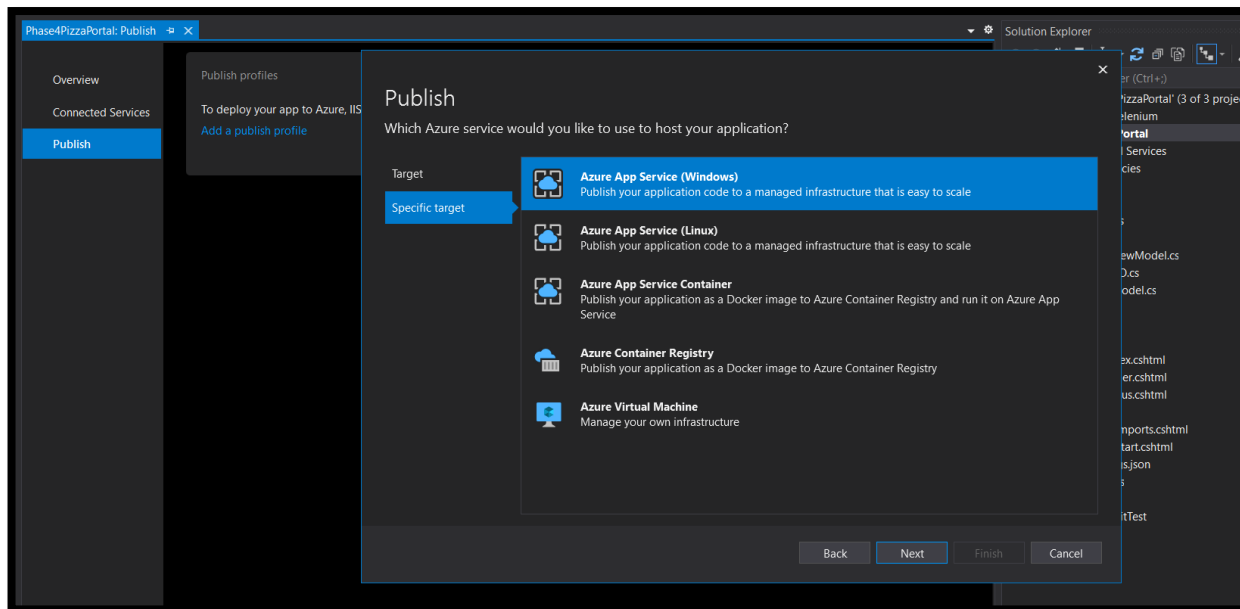


F. Publish and deploy website in Azure

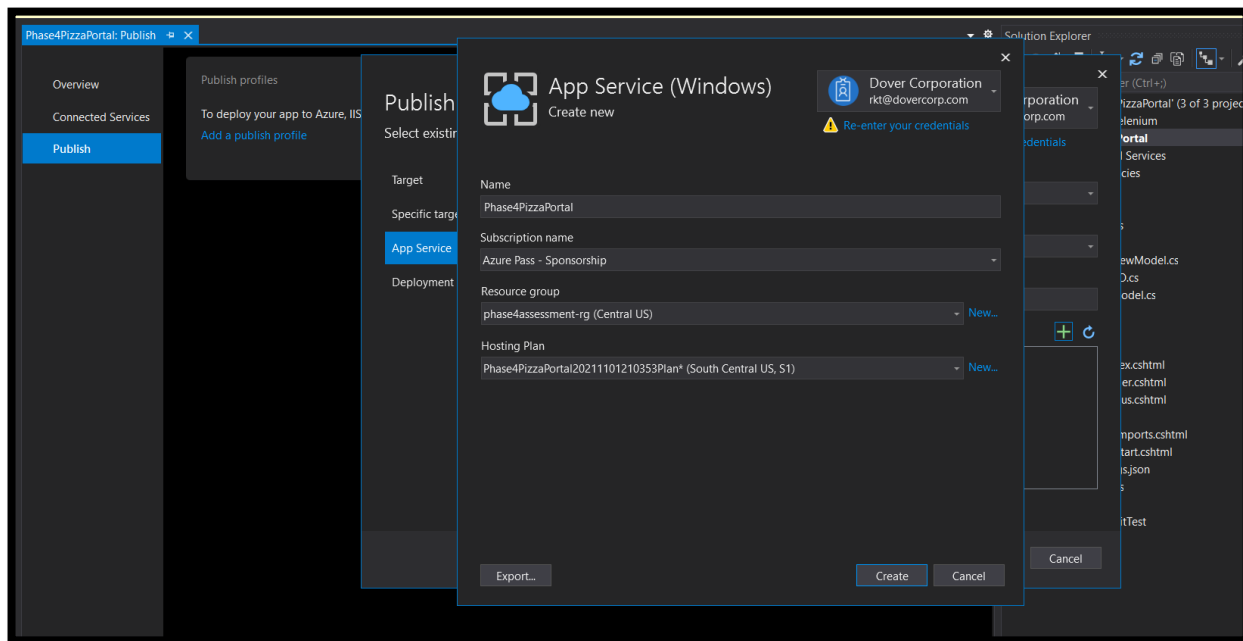
1. Right click on project and select publish. Choose azure



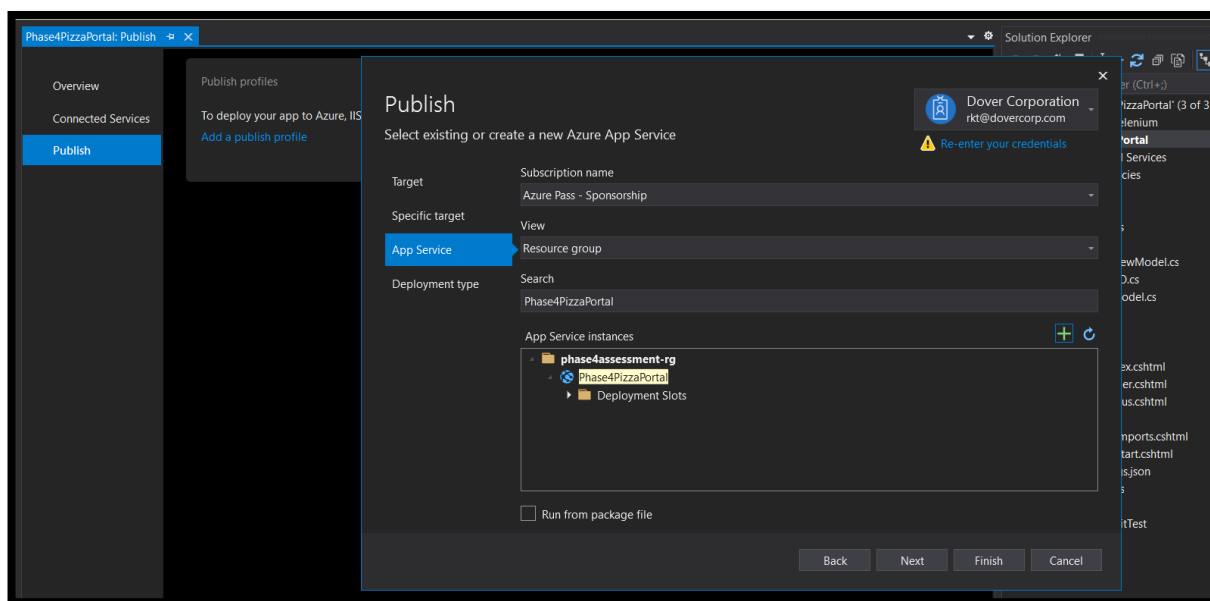
2. Choose as azure app service



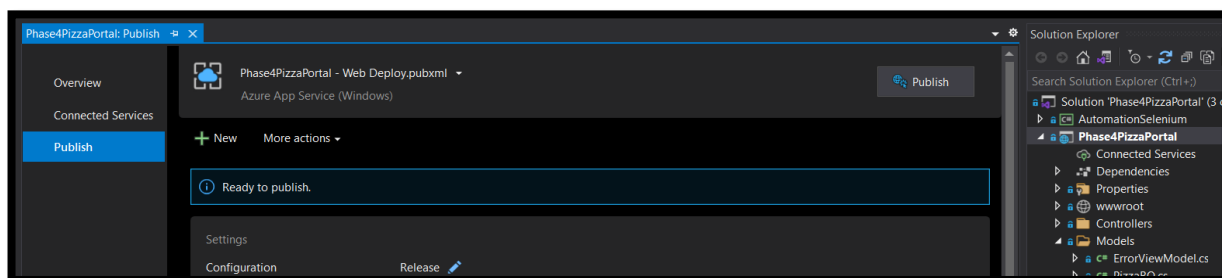
3. Add new app service



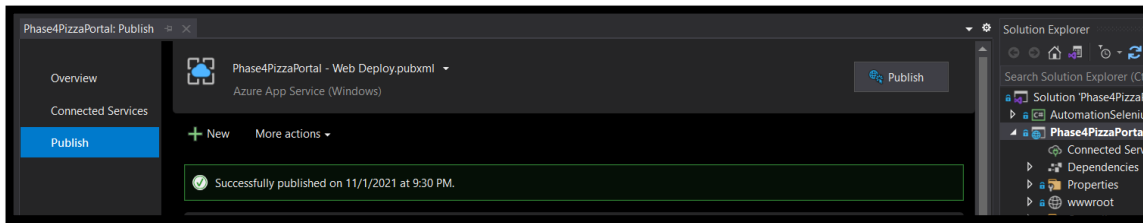
4. Click on finish button



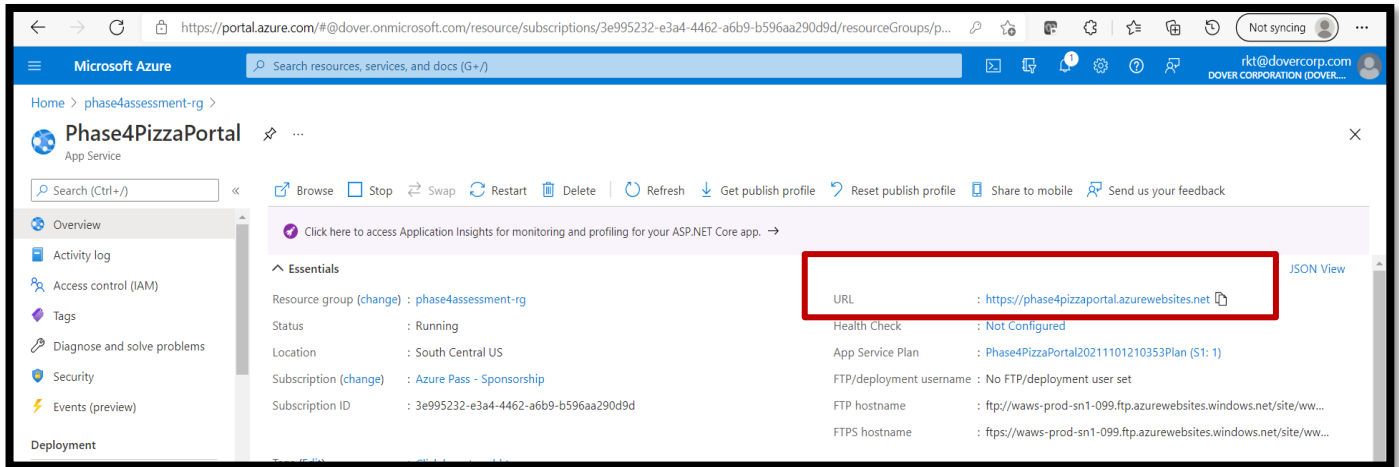
5. Click publish button to deploy website into azure site



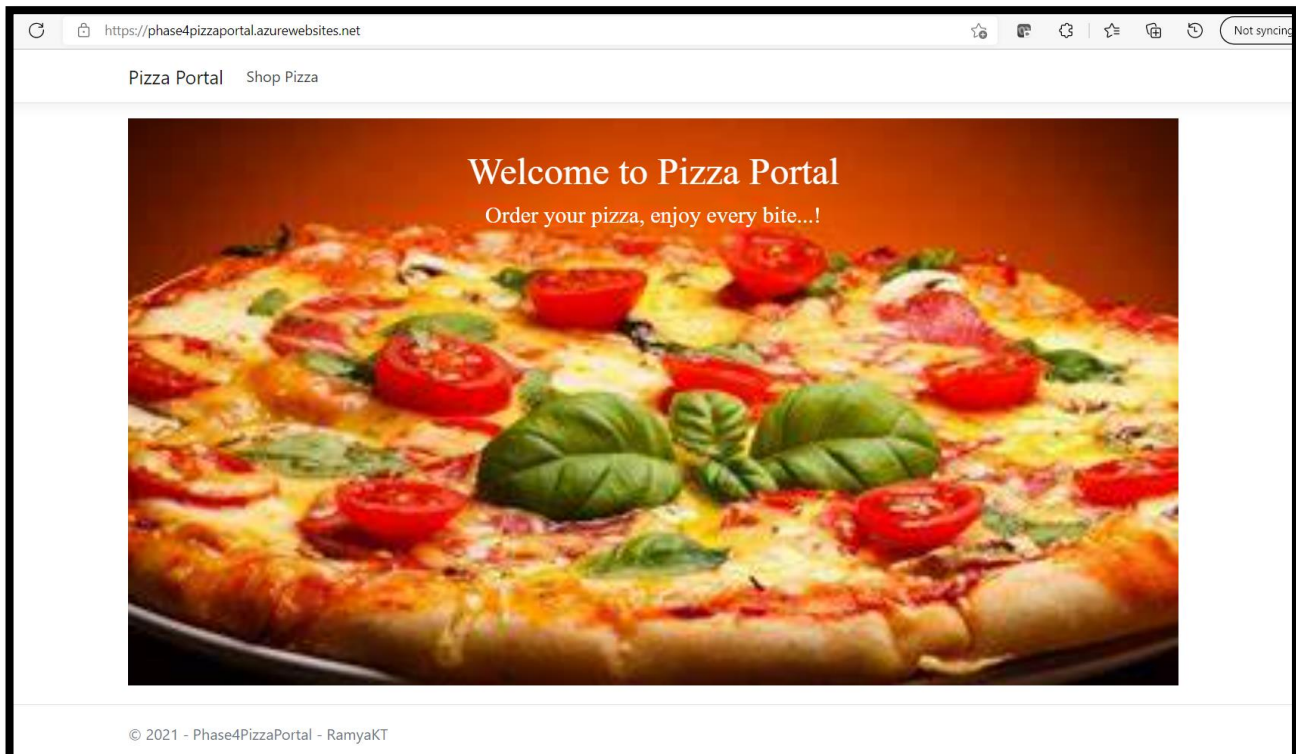
6. The image shows the successful deployment



7. Go to azure portal to see app service and click on website url



8. The image shows the home screen of project after hosting in Azure



9. The image shows the pizza portal screen of project after hosting in Azure

