**Name:** Shobhit Agrawal

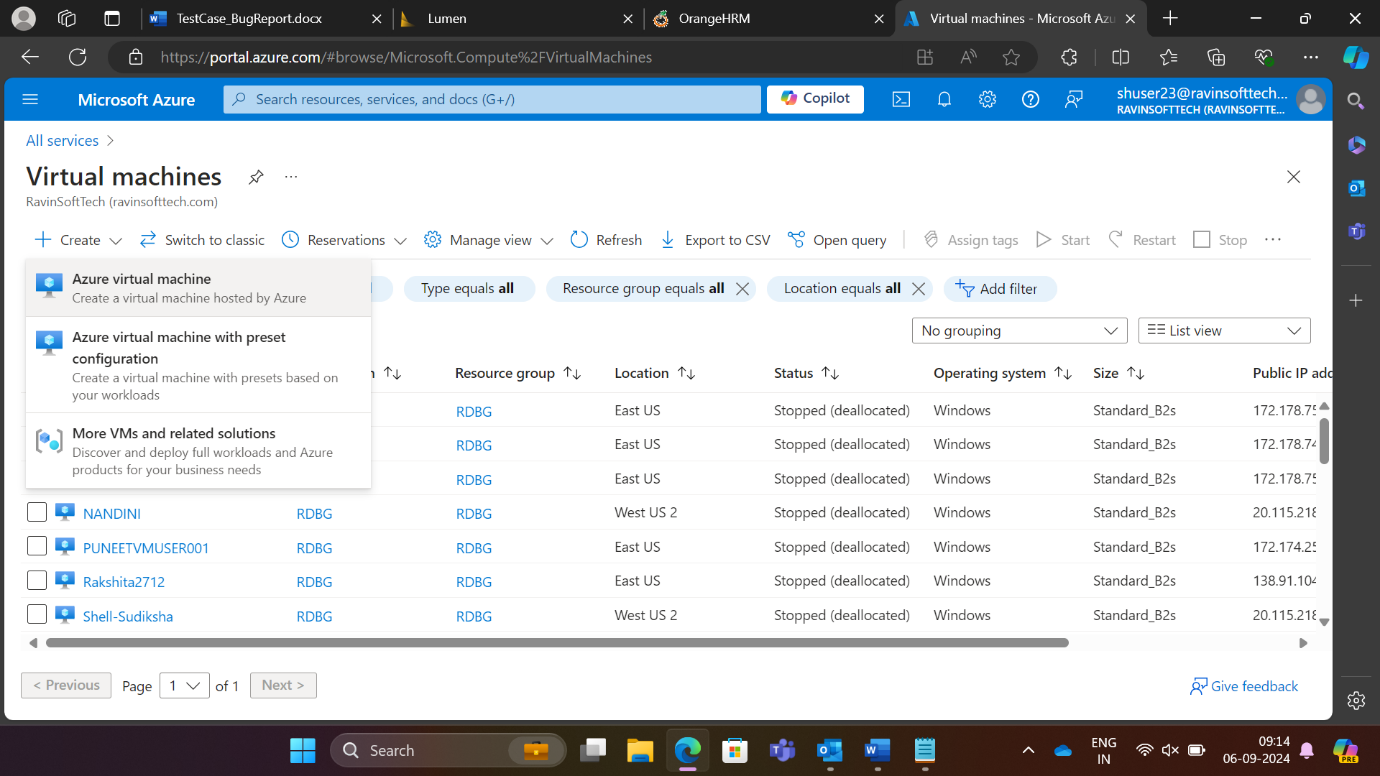
**Submitted to:** Mohammad Sohel Abbasi Sir

**Date:** 06/09/2024

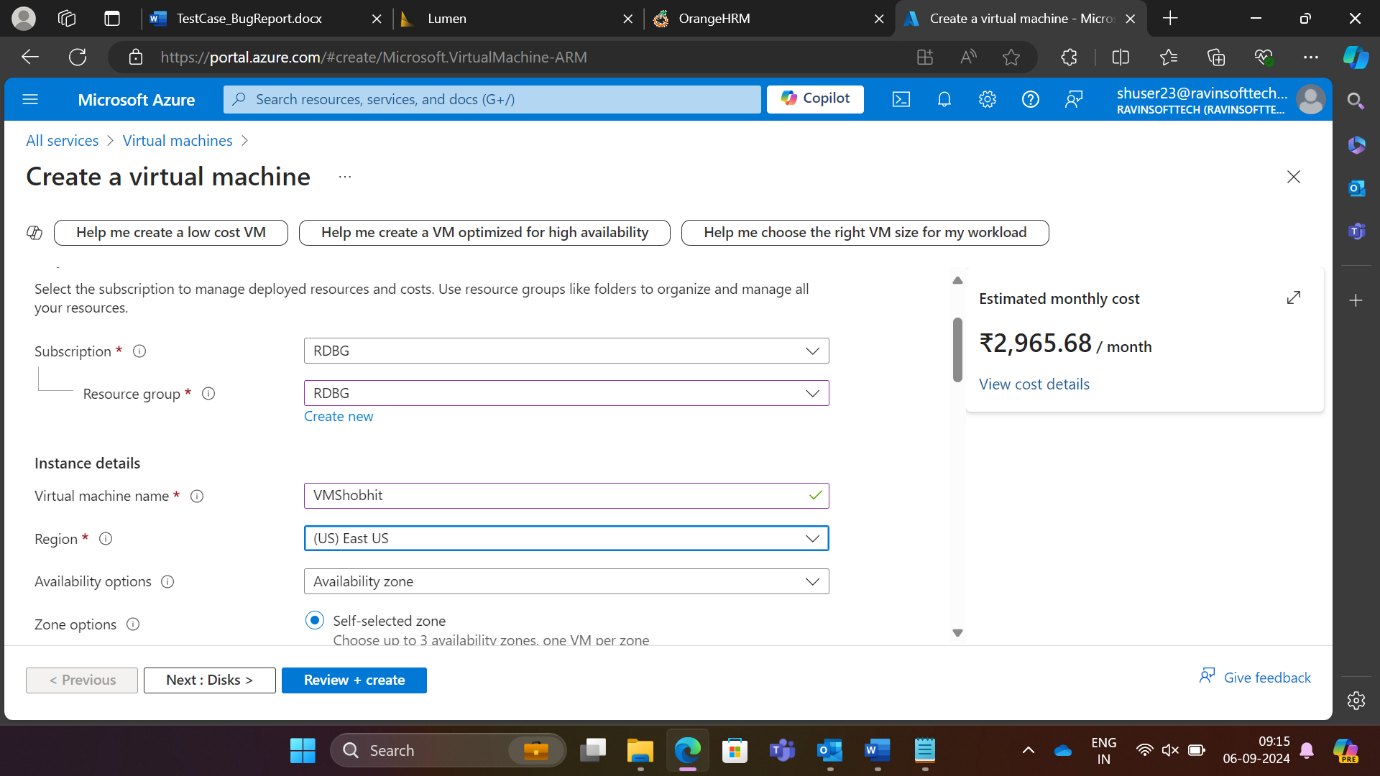
**Task:** Implementing the CI/CD Pipeline using GitHub Actions

**Creating an Azure Virtual Machine**

Step 1: Click on Create 🡪 Click on Azure Virtual Machine



Step 2: Select the resource group as RDBG and give a unique name (VMShobhit) with Region (US) East US



Step 3: Select Security type as Standard and Image as Windows 10 pro version 22H2 x64 gen 2 and VM architecture as x64

A screenshot of a computer

Description automatically generated

Step 4: Give username and password (as we received in the mail)

A screenshot of a computer

Description automatically generated

Step 5: In Select inbound ports, select all the available ports HTTP (80), HTTPS (443), SSH (22), RDP (3389) and check the Licensing checkbox

A screenshot of a computer

Description automatically generated

Step 6: Go to Next : Disks and select OS disk type as Standard HDD (Standard HDD (locally-redundant storage))

A screenshot of a computer

Description automatically generated

Step 7: Go to Next: Networking and create a unique Virtual network (ShobhitVM-net)



Step 8: Check the checkbox Delete public IP and NIC when VM is deleted

A screenshot of a computer

Description automatically generated

Step 9: Then go to Review+Create and once the validation is complete, click on Create

A computer screen with a message

Description automatically generated

Step 10: The deployment will be in progress

A screenshot of a computer

Description automatically generated

Step 11: Deployment successful

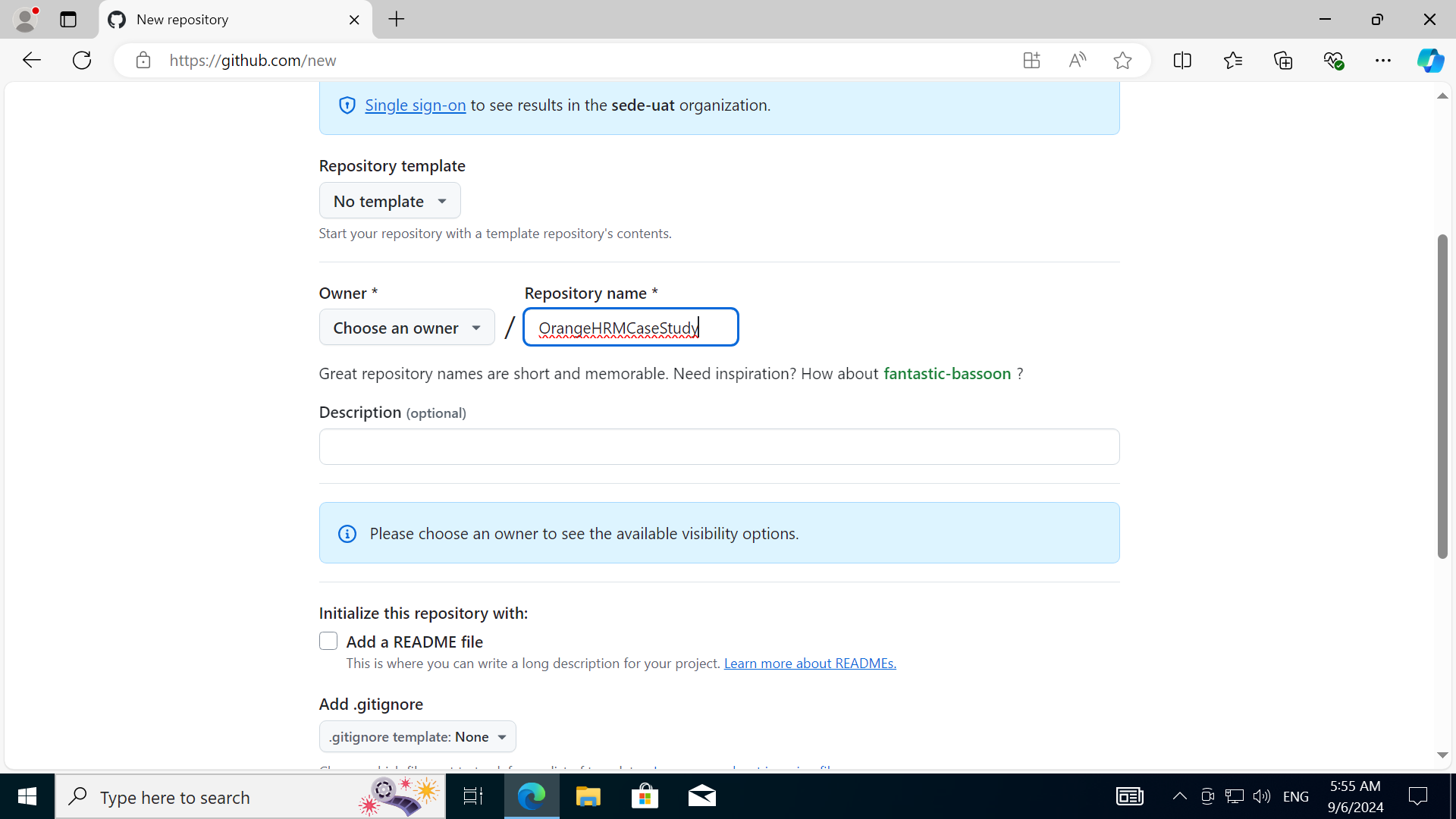
A screenshot of a computer

Description automatically generated

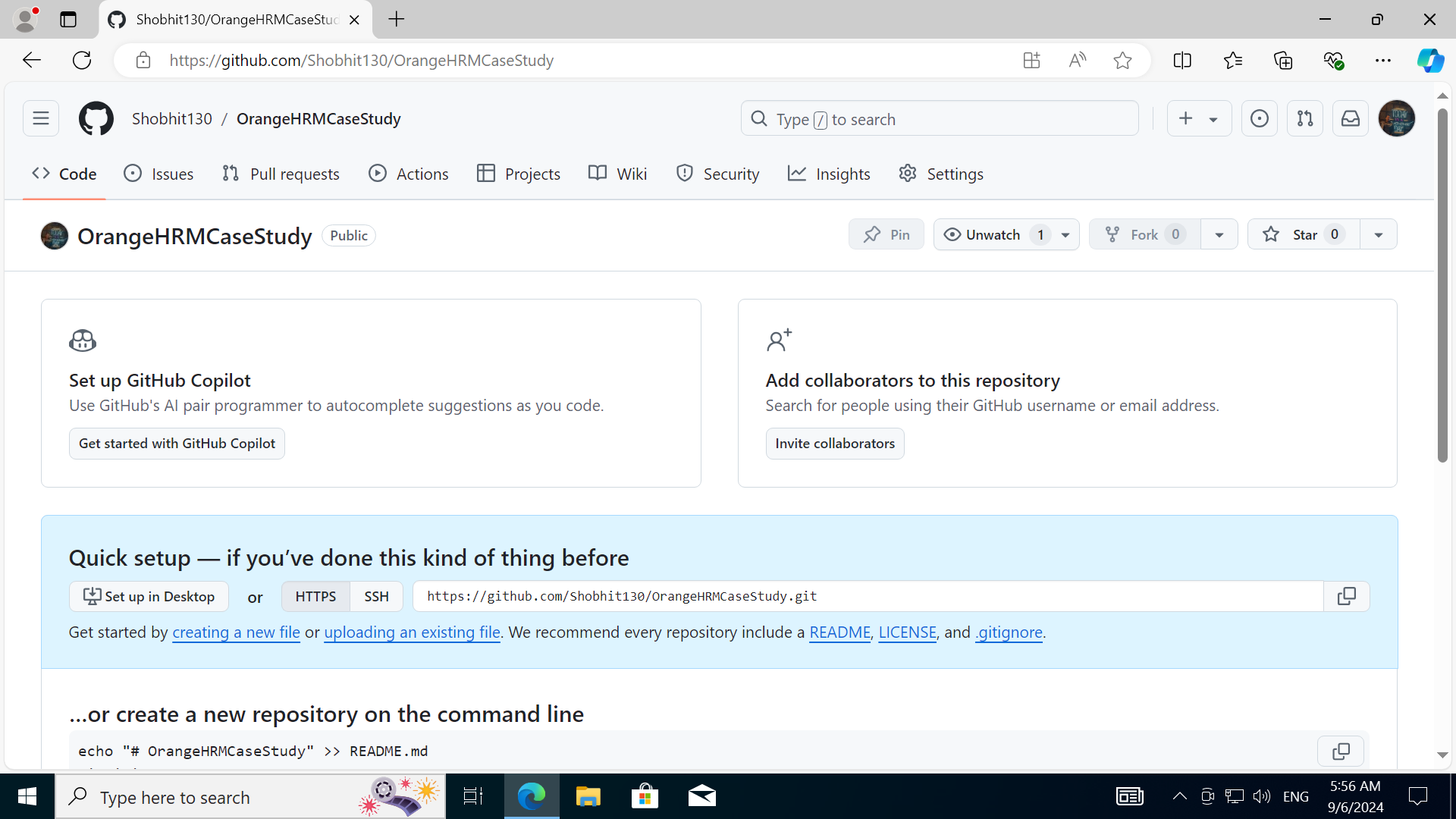
**GitHub Action Workflow**

After the VM creation, we make a GitHub Repository

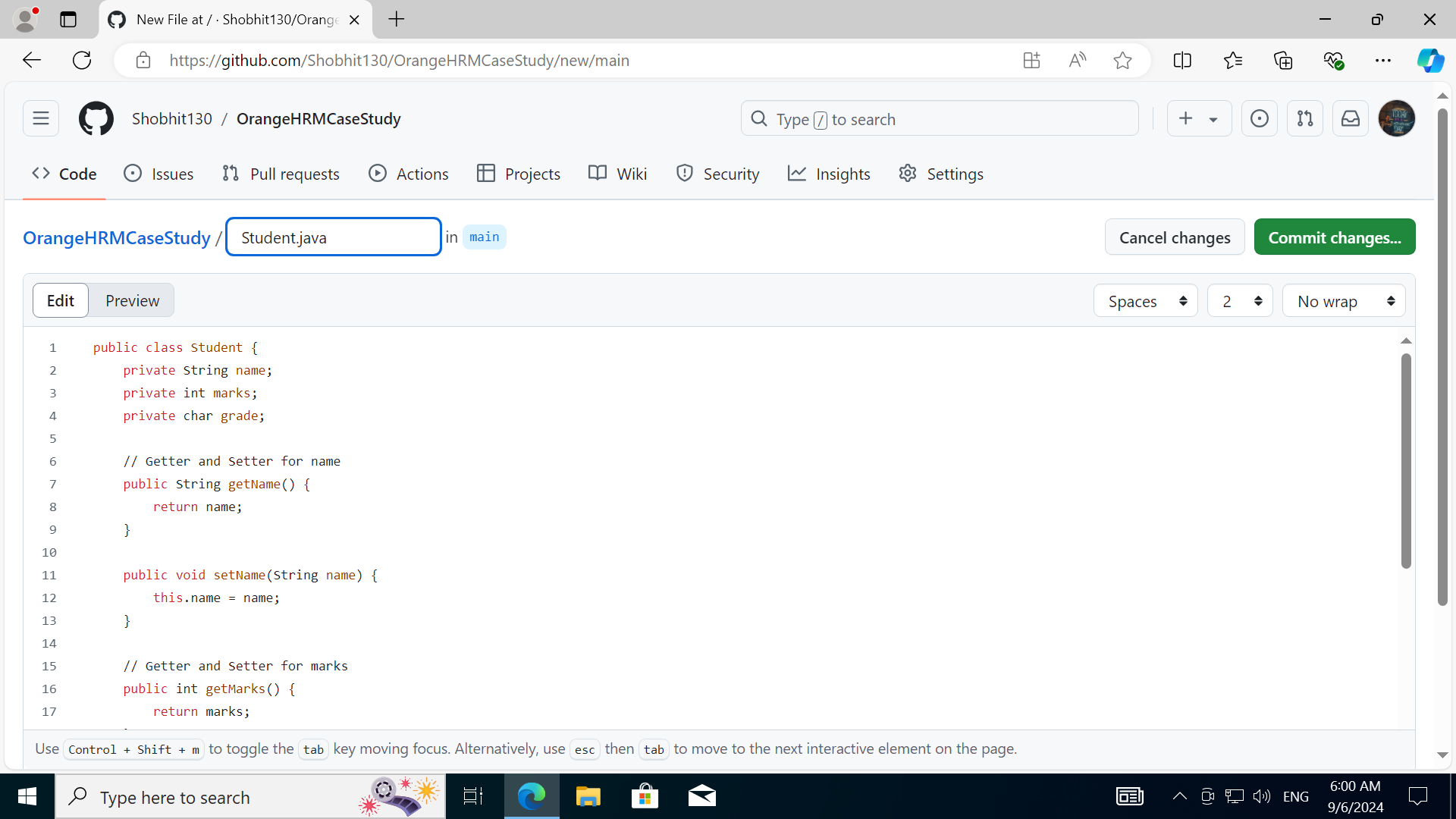
Step 1: Created a GitHub Repository “OrangeHRMCaseStudy”



Step 2: The repository



Step 3: Created a Java code “Student.java”



Step 4: The java code file in repository

A screenshot of a computer

Description automatically generated

Step 5: Create a Java with Maven workflow in GitHub actions

A screenshot of a computer

Description automatically generated

Step 6: A maven.yml file is created as shown below, we commented out the last 2 lines

A screenshot of a computer

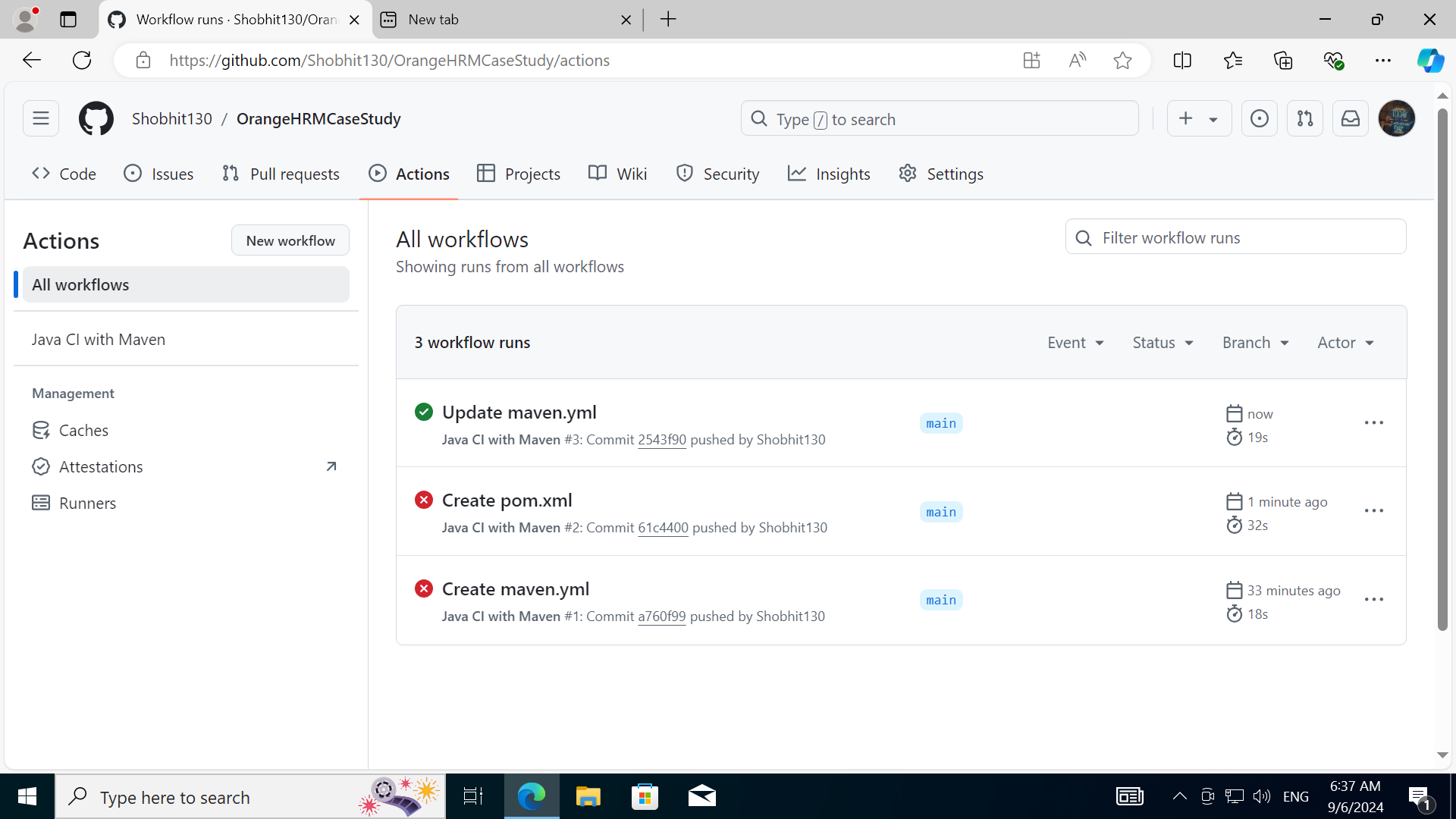
Description automatically generated

Step 7: We created a pom.xml file

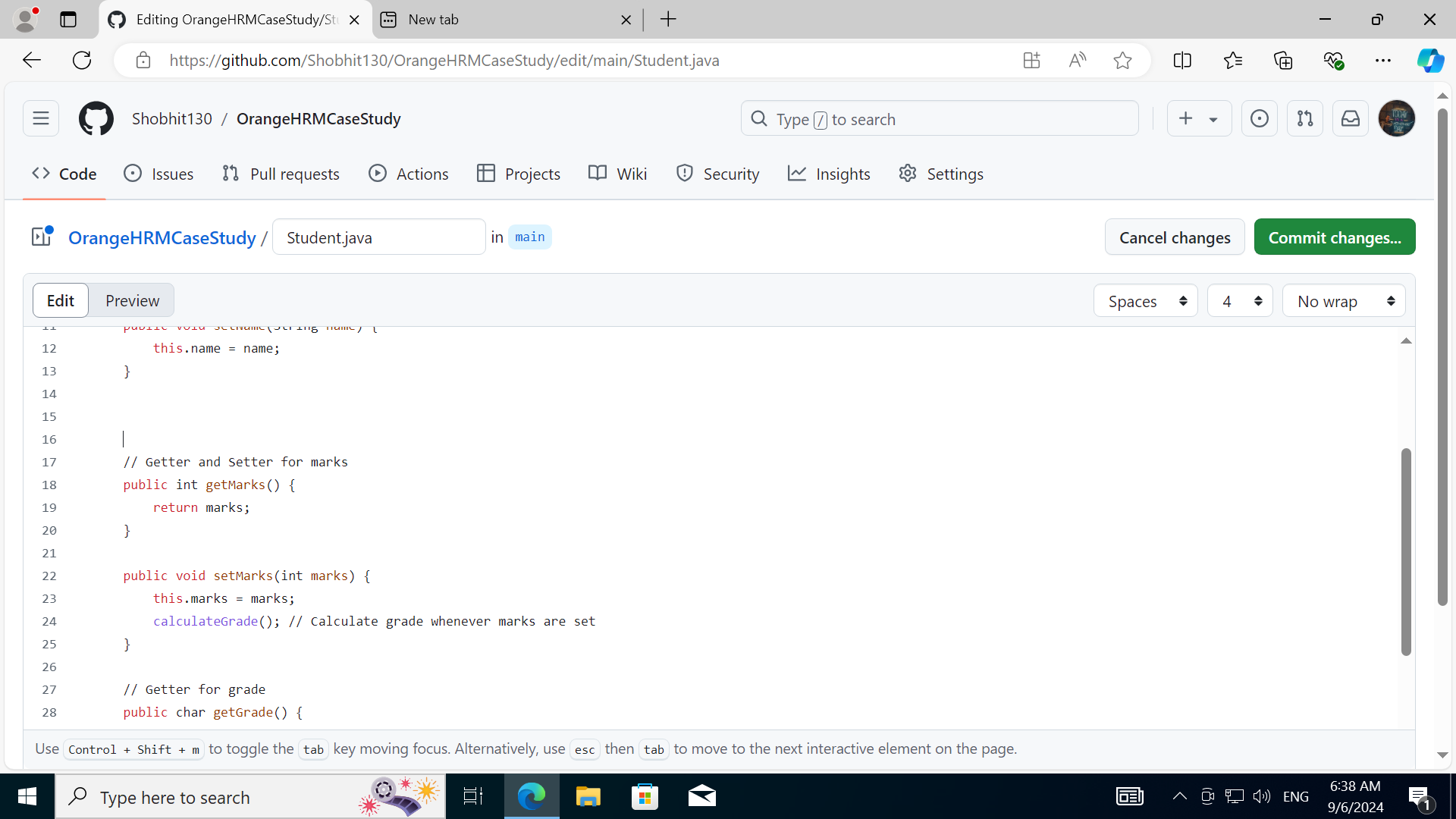
A screenshot of a computer

Description automatically generated

Step 8: We can see the initial build or workflow as the maven.xml file was updated previously (2 lines were commented out)



Step 9: Changes in the code file after the initial build  
Changes : Gave some enters in the code file



Removed the highlighted comment in blue as shown below

A screenshot of a computer

Description automatically generated

Step 10: Now we can see the build is going on after the code updation

A screenshot of a computer

Description automatically generated

Step 11: We can see that the Update Student.java is successful

A screenshot of a computer

Description automatically generated