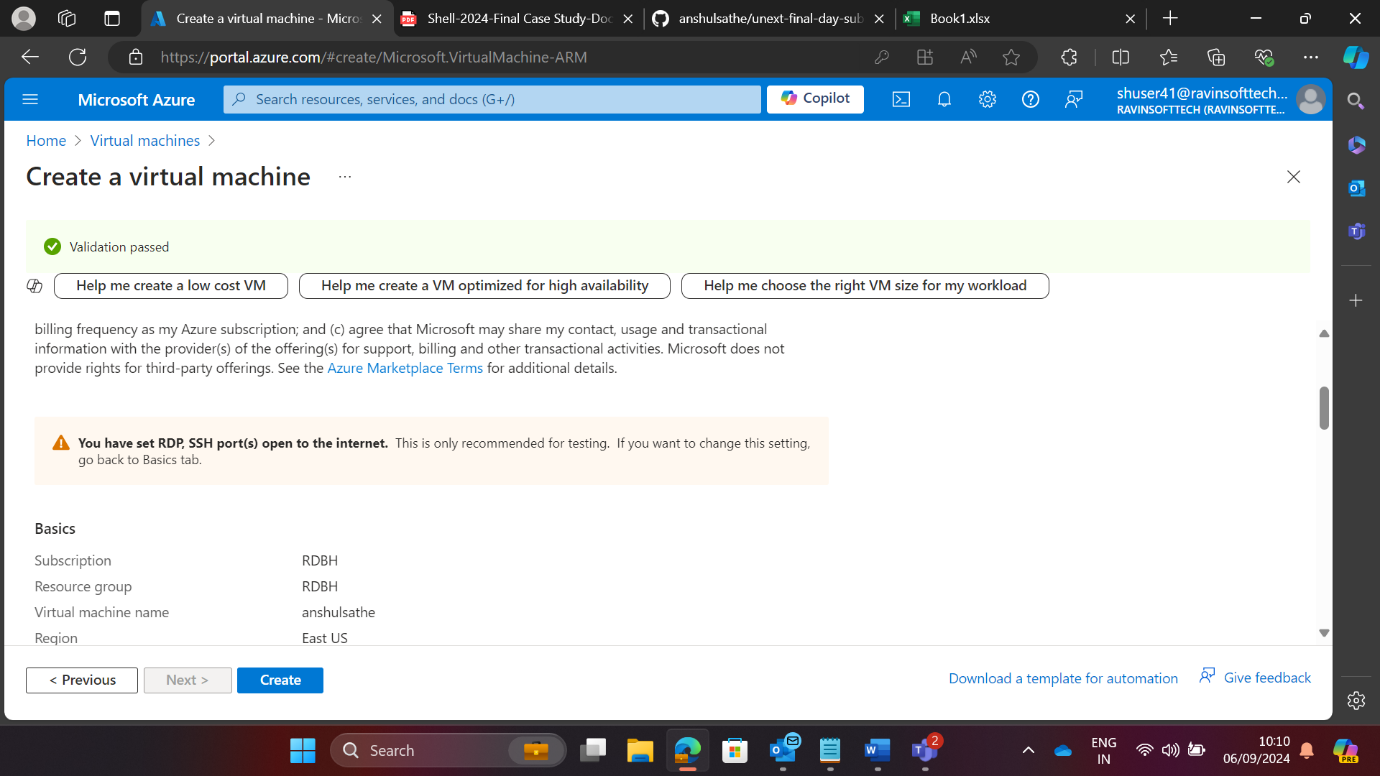
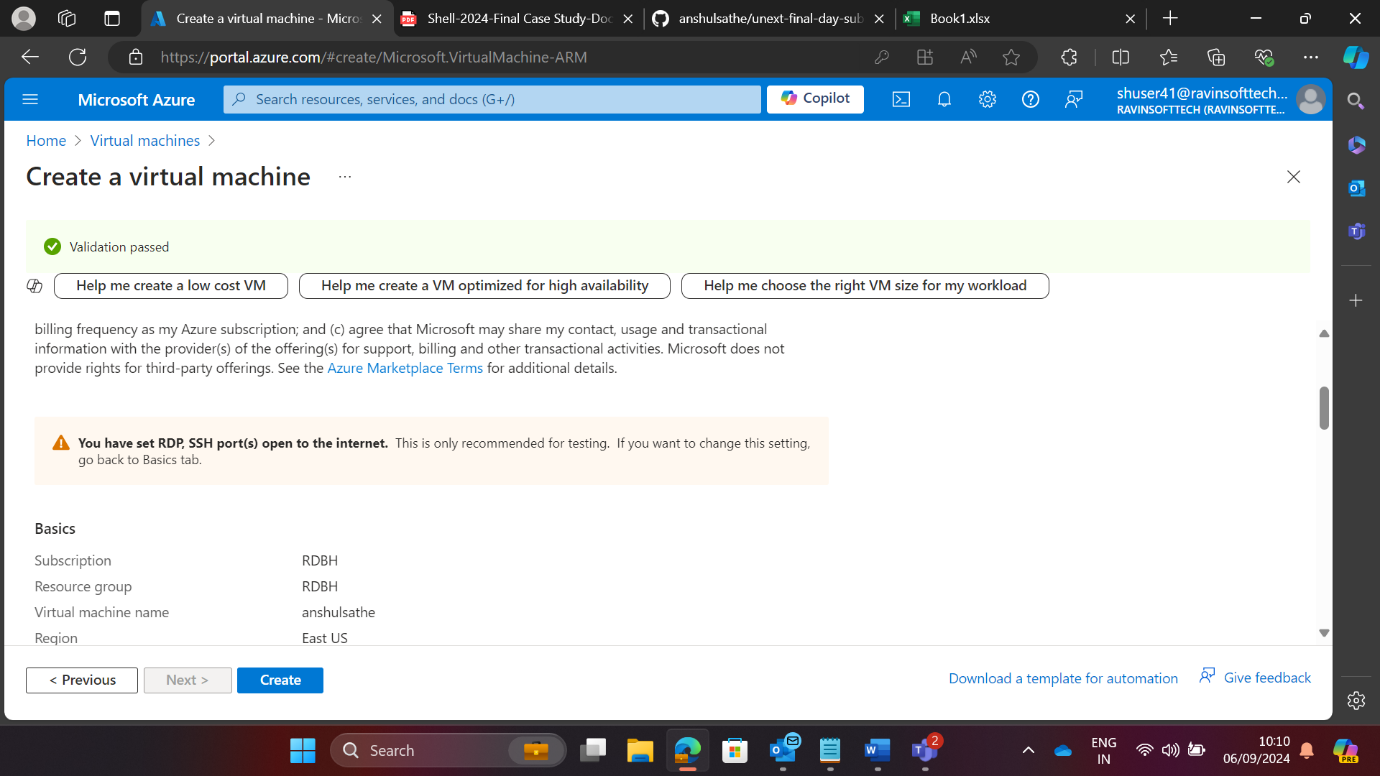
U-Next Final day Case-Study

Anshul Sathe

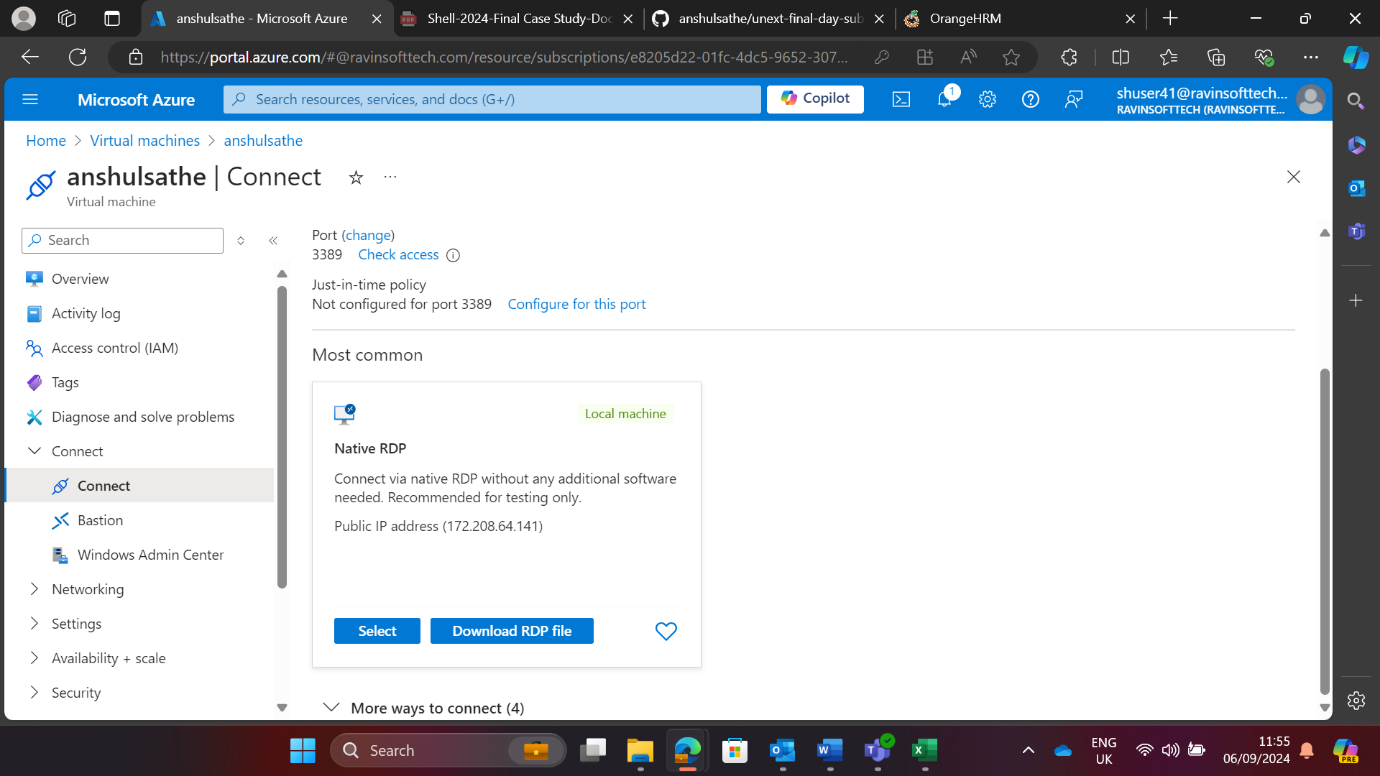
1. Open Azure DevOps.



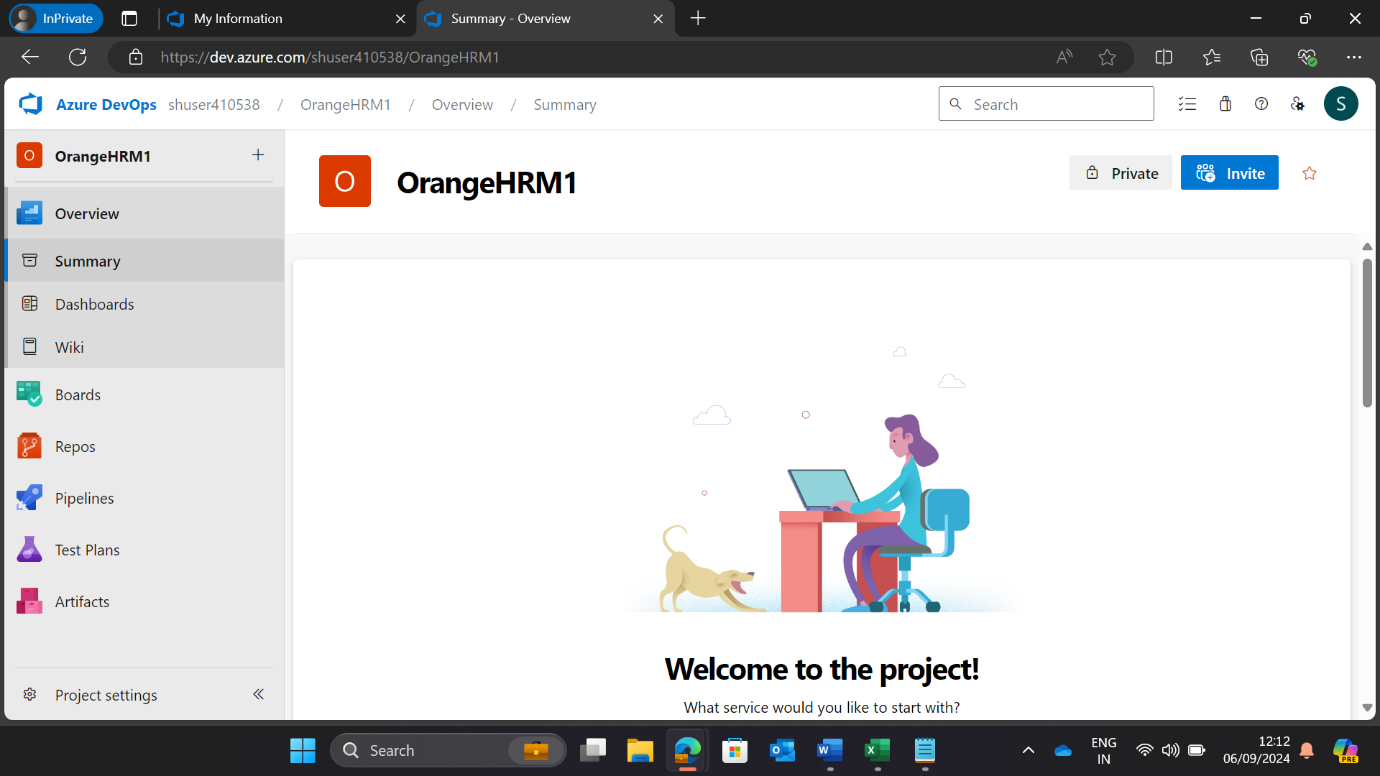
2. Create an Azure Virtual Machine (VM). You need to perform all the tasks within the VM.



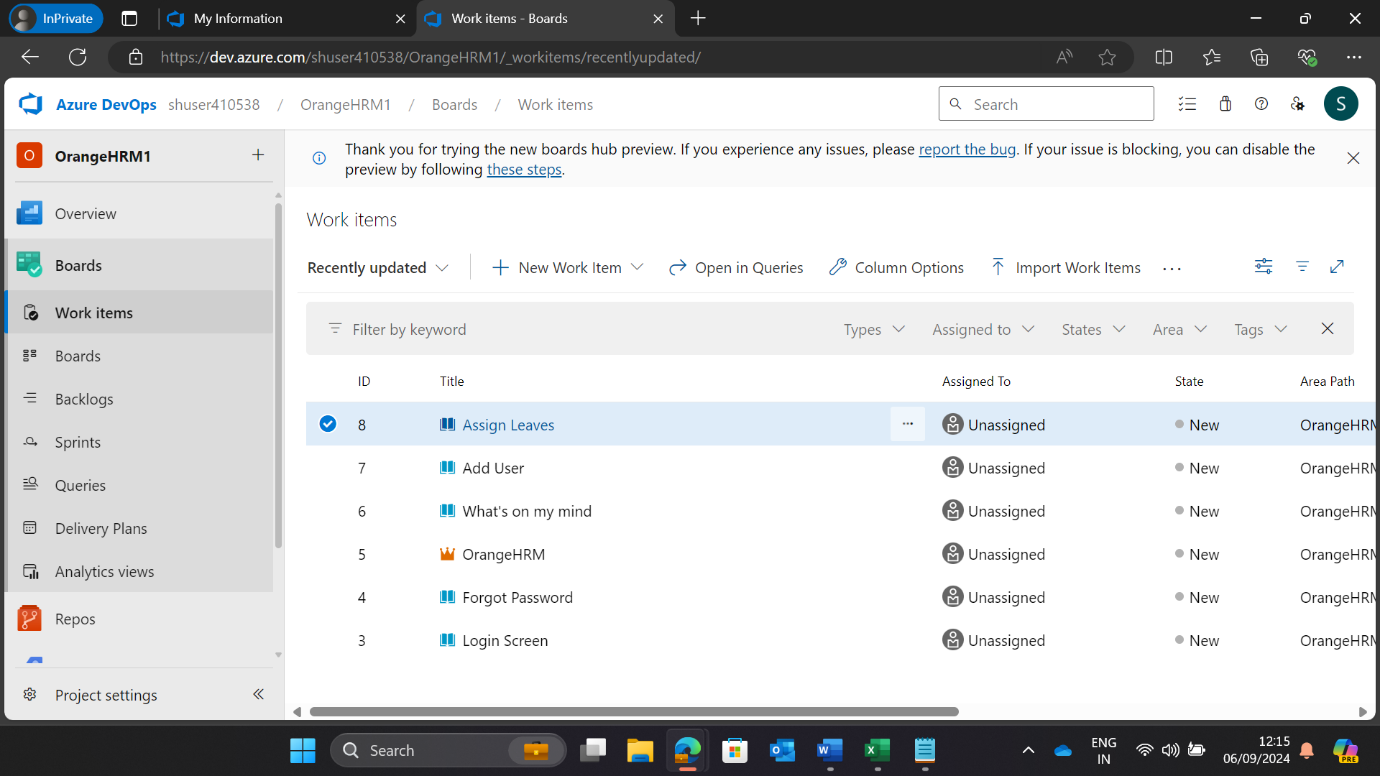
3. Login to the azure devops portal and navigate to the project.



4. Create a New Sprint. Set the Sprint goals and objectives, aligning them with the overall testing strategy for the website.



5. Create 5 User Stories in the Sprint for the Sprint Backlog for the given application.



6. Each user story should cover different aspects of the Software Development Life Cycle (SDLC) as it relates to the functionalities of the website.

A screenshot of a computer

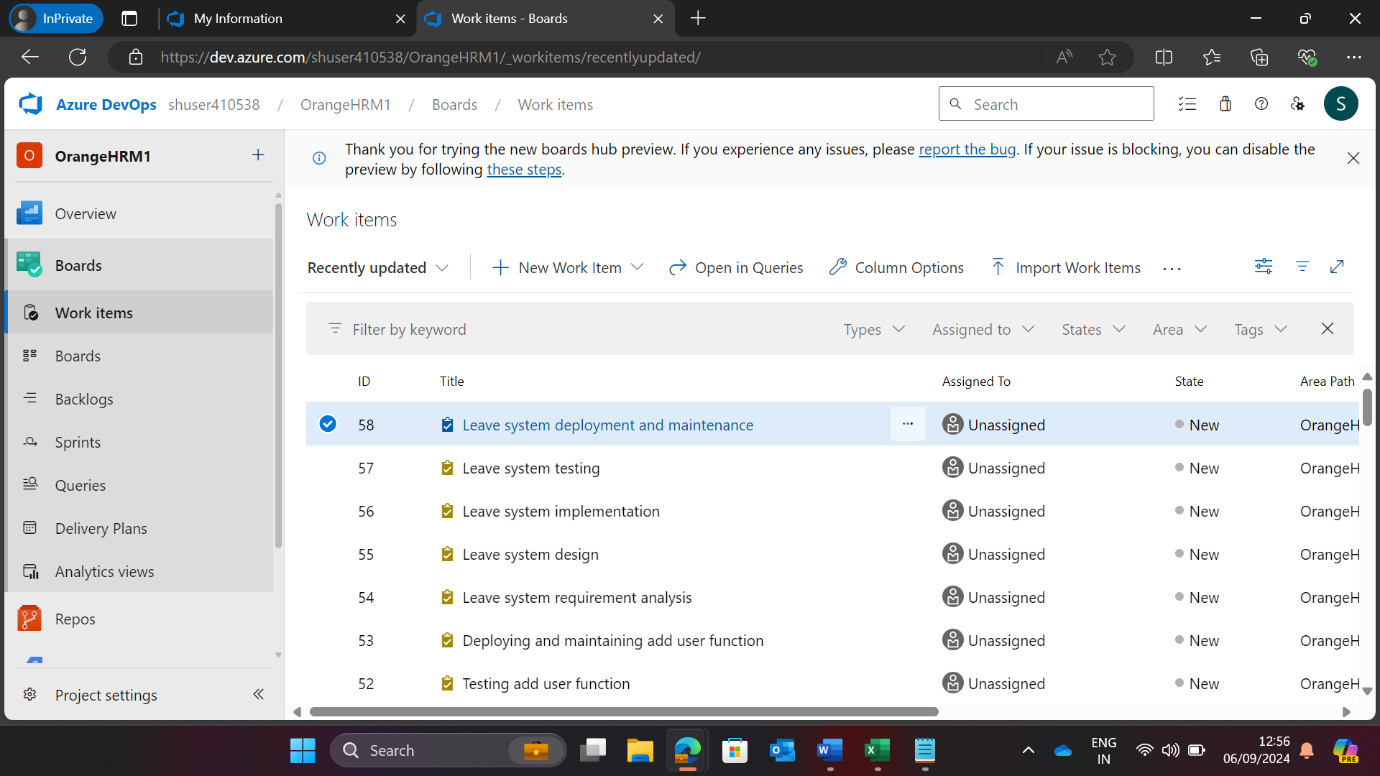
Description automatically generated

7. Create 25 test cases for the user stories created for the Sprint.

A screenshot of a computer

Description automatically generated

8. Define the tasks and sub tasks for each User Story.



9. Execute all the test cases manually (functional testing).

EXCEL SHEET IN GITHUB

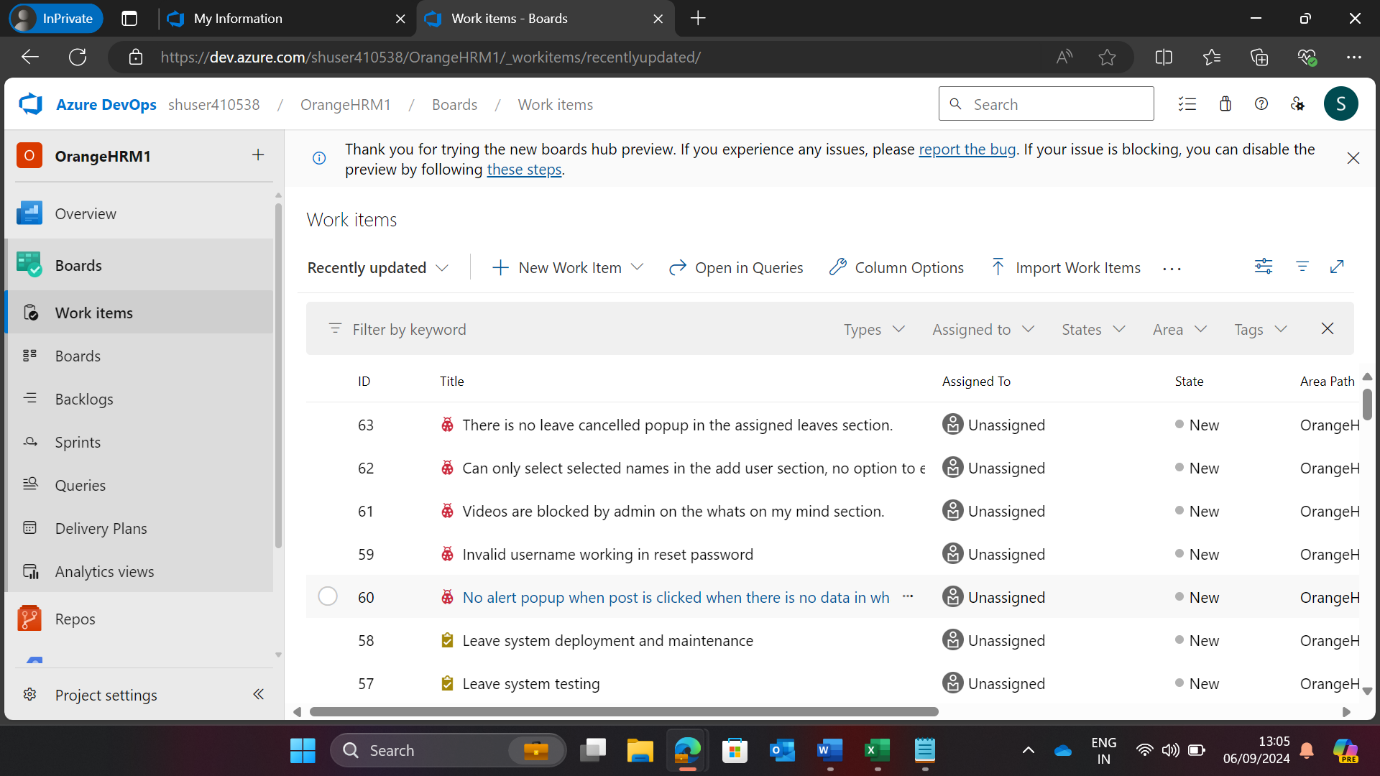
A screenshot of a computer

Description automatically generated

10. Create a Defect Report for each failed or blocked test case.

**A screenshot of a computer

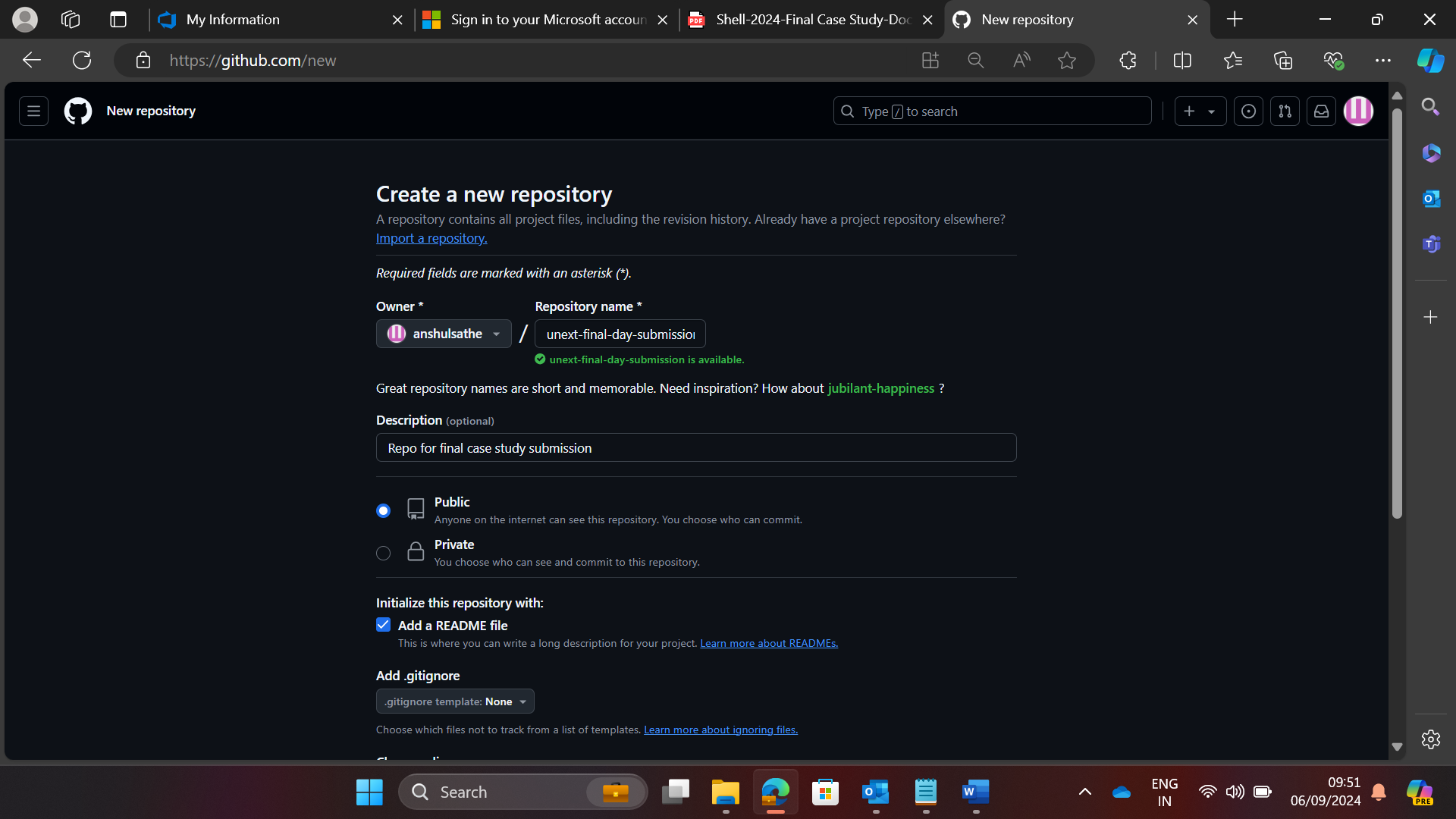
Description automatically generated**

****

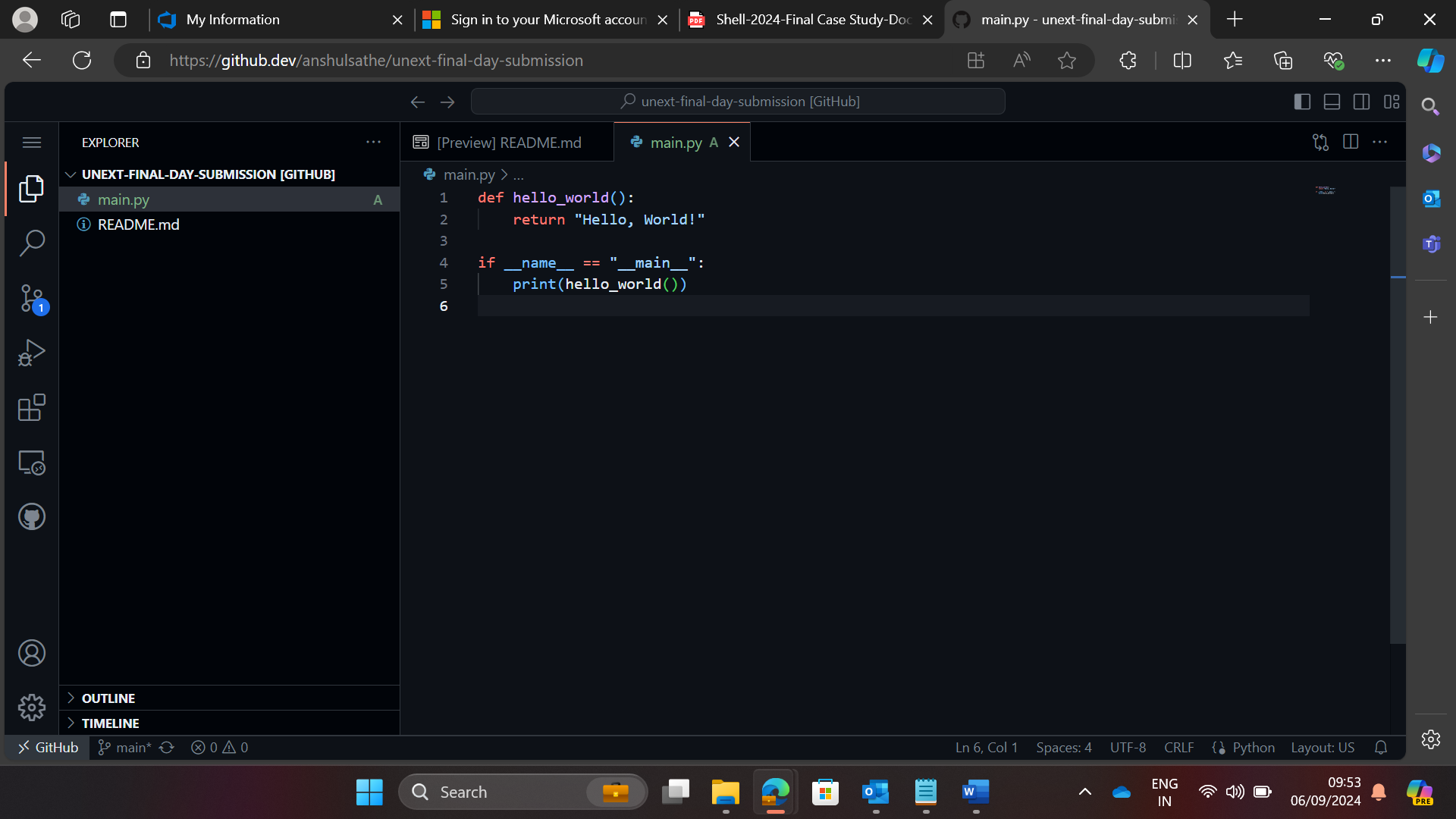
**Implementing the CI/CD Pipeline using GitHub Actions**

Repo Link: [anshulsathe/unext-final-day-submission: Repo for final case study submission (github.com)](https://github.com/anshulsathe/unext-final-day-submission)

11. Create a new repository on your GitHub account.

x`

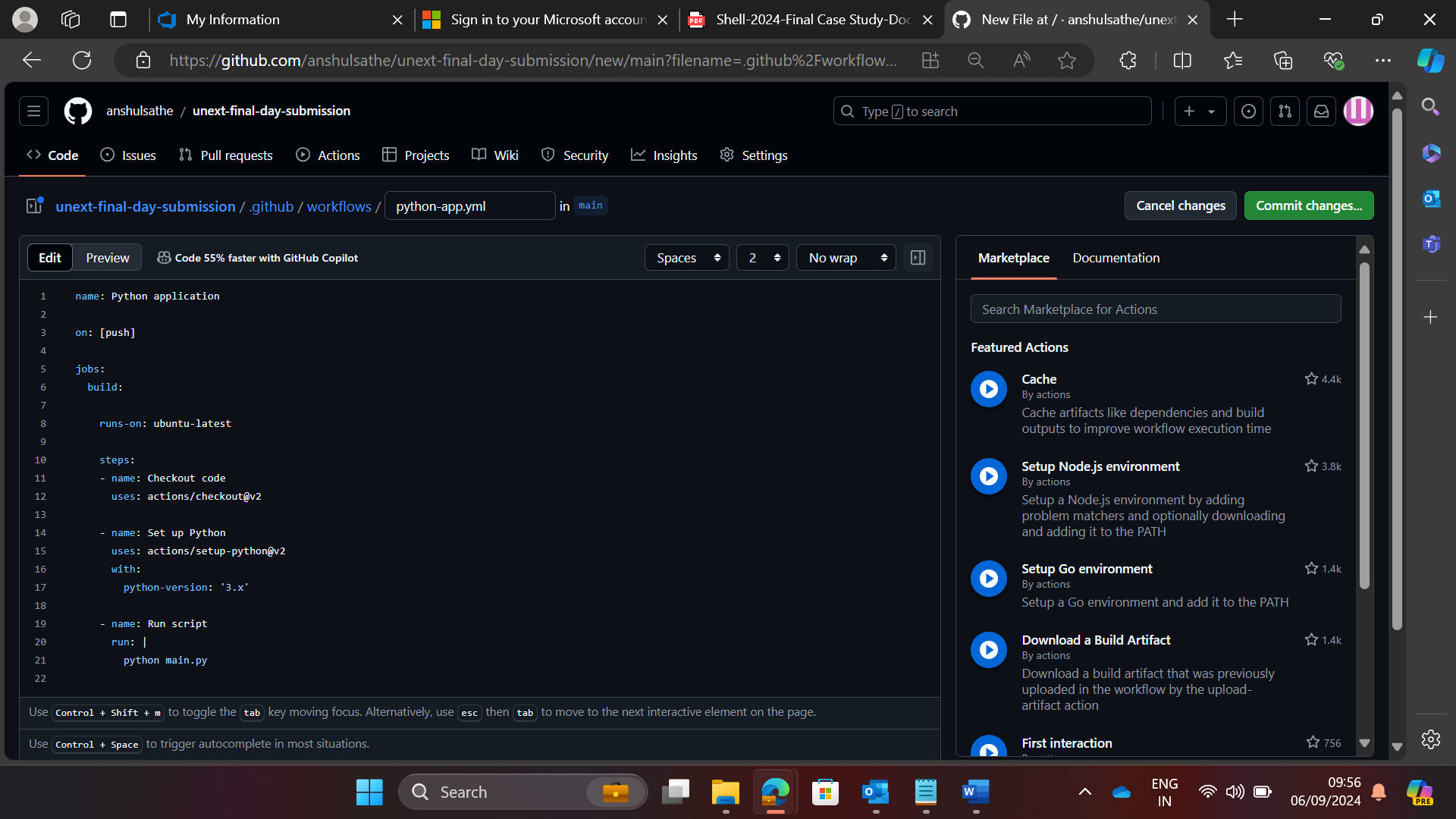
12. Clone the code base of the Java (any preferred language) project into your repository. The project’s codebase would be present in the VM.



A screenshot of a computer

Description automatically generated

13. Create a New Workflow on GitHub Actions for the Java/Maven code.



14. Monitor the created build.

A screenshot of a computer

Description automatically generated

15. Modify the code in your repository to recheck the Workflow and the Build. For example, you could modify a Java class or update the pom.xml file.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

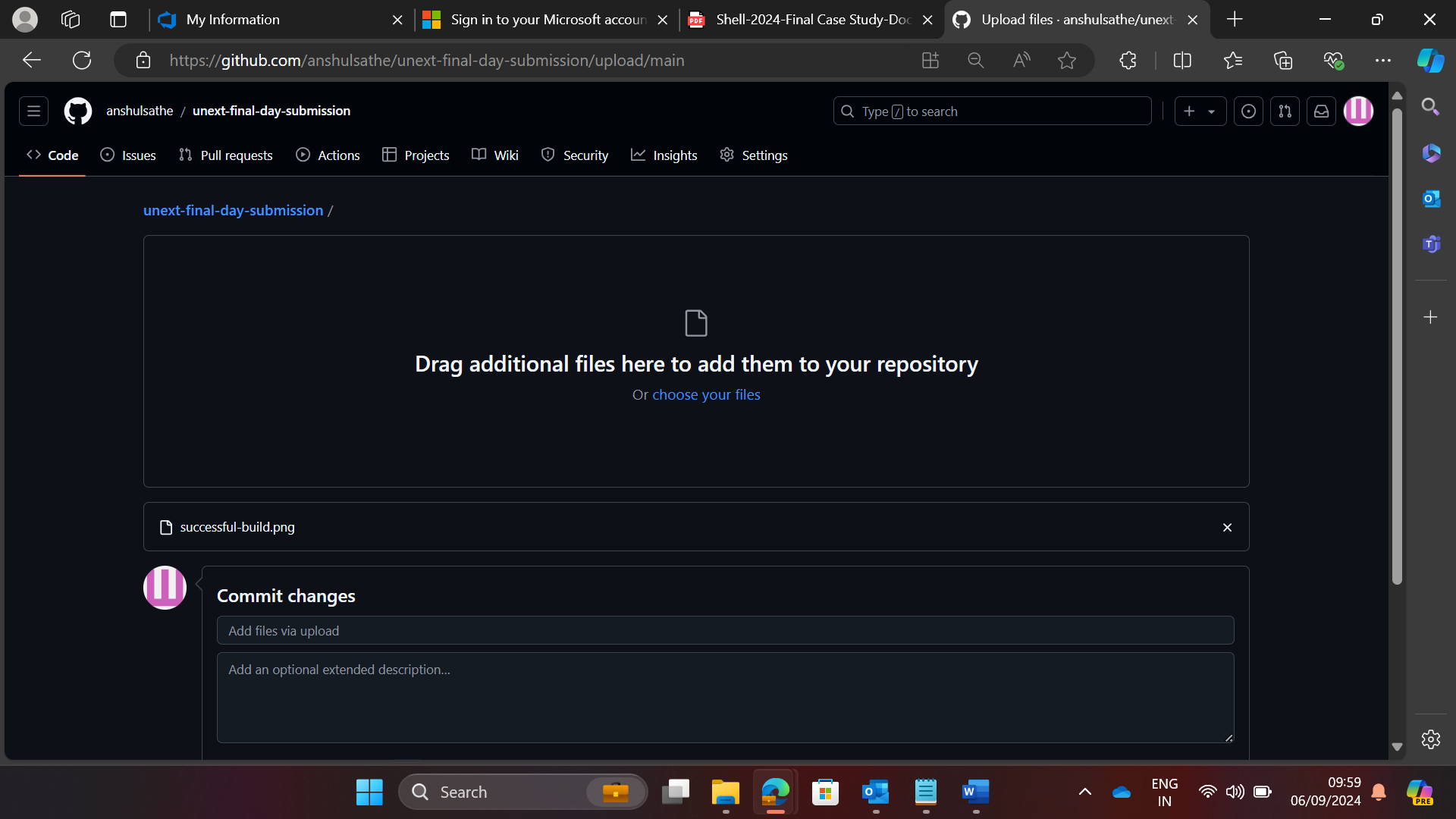
Description automatically generated

16. Commit it and monitor the build result

A screenshot of a computer

Description automatically generated

17. Update the screeshot of successful build and commit it in the repo.



A screenshot of a computer

Description automatically generated

18. In your repository, add the created test cases report and defect report.

