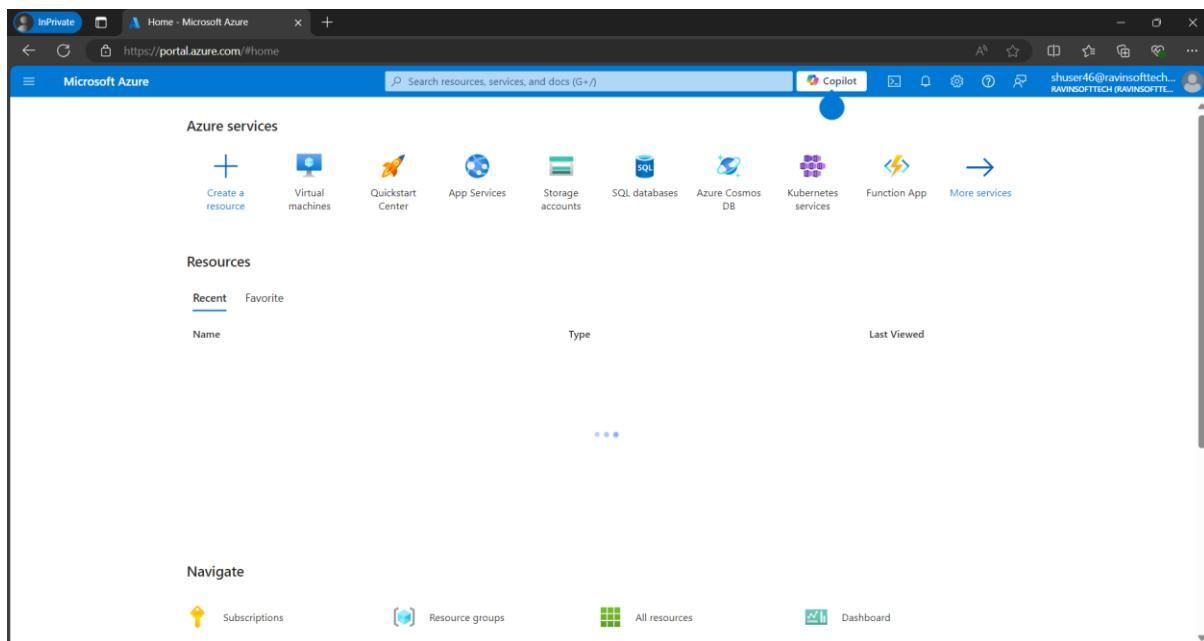


Name : Harsh Verma

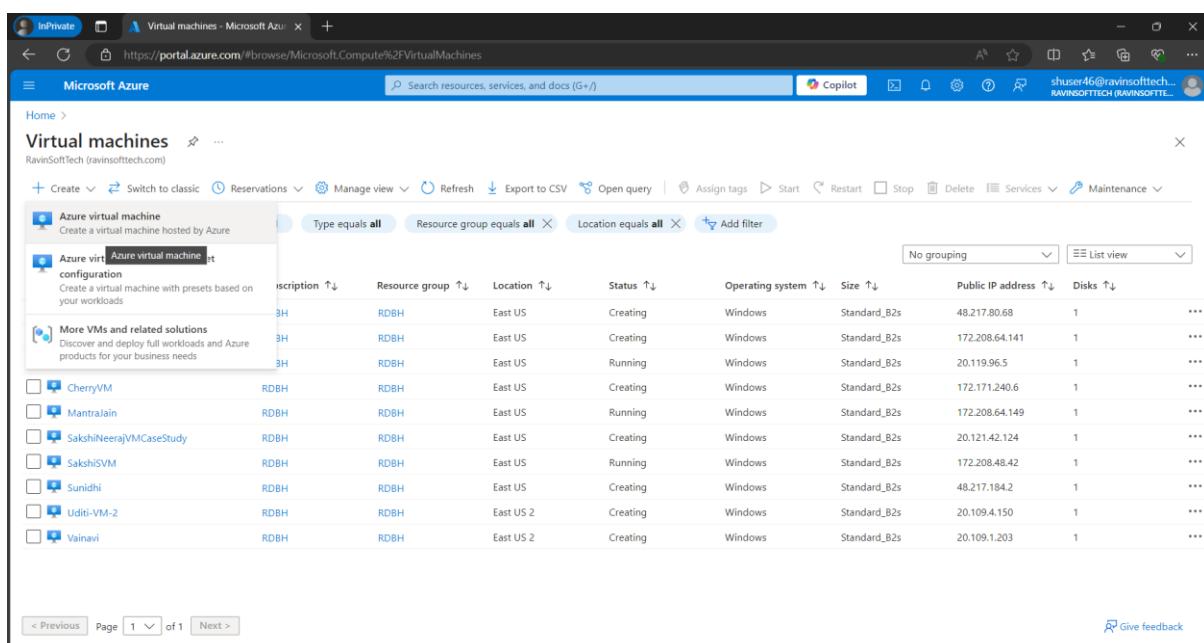
FreakyCoder0/UNext (github.com)

SDLC, Agile and Software Testing



The screenshot shows the Microsoft Azure portal homepage. At the top, there's a search bar and a Copilot button. Below the header, there's a section titled "Azure services" with icons for Create a resource, Virtual machines, Quickstart Center, App Services, Storage accounts, SQL databases, Azure Cosmos DB, Kubernetes services, Function App, and More services. Under "Resources", there are tabs for Recent and Favorite, followed by a table with columns for Name, Type, and Last Viewed. At the bottom, there's a "Navigate" section with links for Subscriptions, Resource groups, All resources, and Dashboard.

1. Open <https://portal.azure.com/> and login



The screenshot shows the Microsoft Azure portal Virtual machines page. The URL is https://portal.azure.com/#browse/Microsoft.Compute%2FVirtualMachines. The page has a header with various management options like Create, Refresh, Export to CSV, etc. Below the header is a search bar and filter options. The main area displays a table of virtual machines with columns for Description, Resource group, Location, Status, Operating system, Size, Public IP address, and Disks. The table includes entries for "Azure virtual machine" (status: Creating), "More VMs and related solutions" (status: Creating), and several other VMs named "CherryVM", "MantraJain", "SakshiNeerajVMCaseStudy", "SakshiSVM", "Sunidhi", "Uditi-VM-2", and "Vainavi". At the bottom, there are navigation buttons for < Previous, Page 1, Next >, and a Give feedback link.

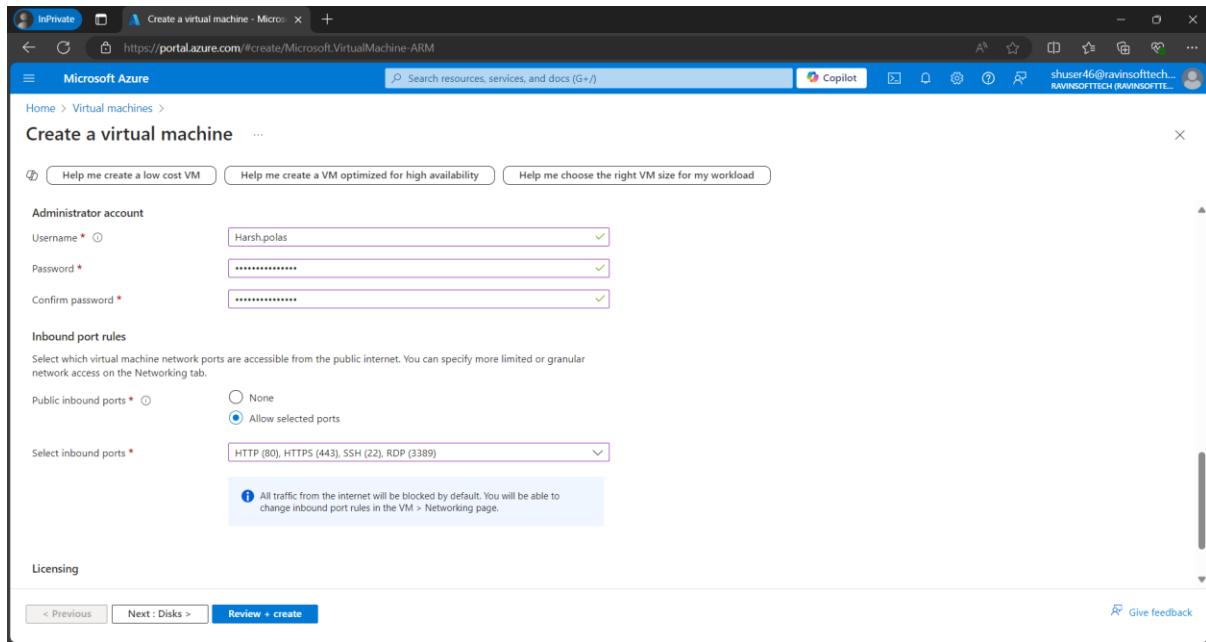
2. Create new VM.

The screenshot shows the 'Create a virtual machine' wizard in Microsoft Azure. The current step is 'Basics'. The 'Subscription' dropdown is set to 'RDBH'. The 'Resource group' dropdown is set to 'RDBH' and has a 'Create new' link below it. Under 'Instance details', the 'Virtual machine name' is 'HarshVM', 'Region' is '(US) East US', and 'Availability options' is 'Availability zone'. Under 'Zone options', the 'Self-selected zone' radio button is selected. At the bottom, there are buttons for '< Previous', 'Next : Disks >', and 'Review + create'.

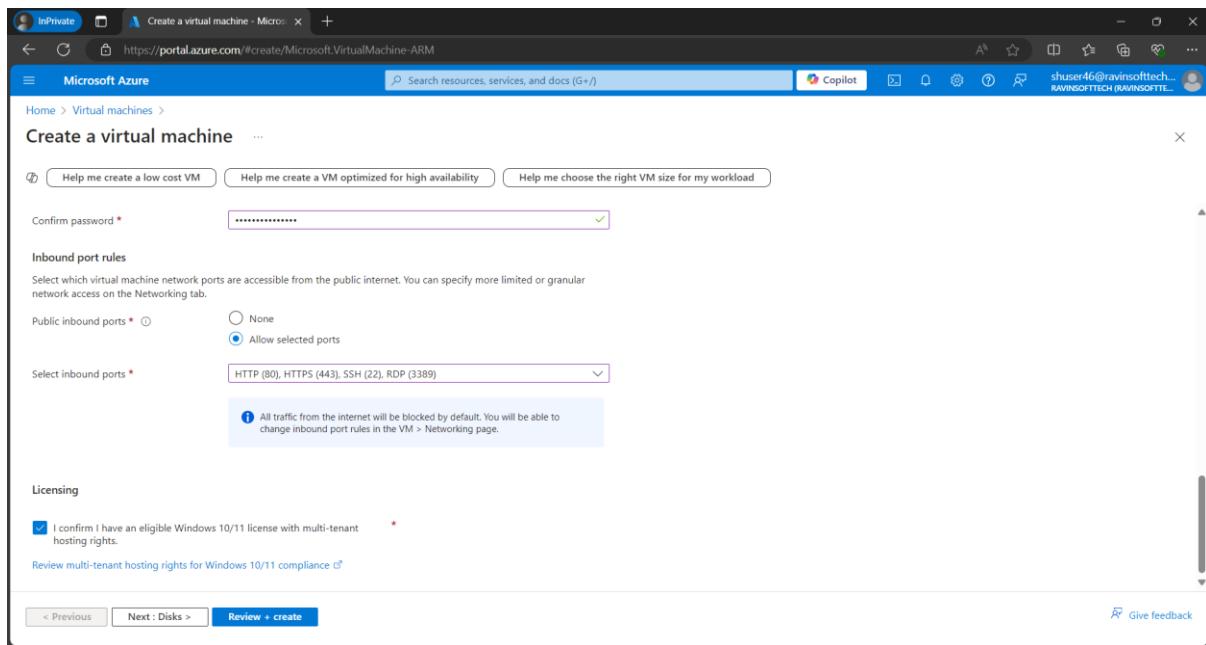
3. In resource group change it to “RDBH”.
4. Then add VM name
5. Change region

The screenshot shows the 'Create a virtual machine' wizard in Microsoft Azure. The current step is 'Image'. The 'Image' dropdown is set to 'Windows 10 Pro, version 22H2 - x64 Gen2'. A note says 'This image is compatible with additional security features. Click here to swap to the Trusted launch security type.' Under 'VM architecture', 'x64' is selected. A note says 'Arm64 is not supported with the selected image.' Under 'Size', 'Standard_B2s - 2 vcpus, 4 GiB memory (£2.526.46/month)' is selected. A note says 'Item(s) availability based on policy assignment(s) for the selected scope. Allowed virtual machine size SKUs (Policy details)'.

6. Then change Image to “Window 10 pro, version 22H2 – 64x Gen2”
7. Then change the size to standard.



8. Update the changes in administrator account.



9. Select all HTTP, HTTPS, SSH, RDP, then select I confirm and select next.

The screenshot shows the 'Create a virtual machine' wizard in progress, specifically Step 7: 'OS disk'. The page title is 'Create a virtual machine ...'. At the top, there are three buttons: 'Help me create a low cost VM', 'Help me create a VM optimized for high availability', and 'Help me choose the right VM size for my workload'. Below these, a note states: 'The size of the VM determines the type of storage you can use and the number of data disks allowed. Learn more'. A section titled 'VM disk encryption' notes: 'Azure disk storage encryption automatically encrypts your data stored on Azure managed disks (OS and data disks) at rest by default when persisting it to the cloud.' There is a note: 'Encryption at host is not registered for the selected subscription. Learn more about enabling this feature'. Under 'OS disk', 'OS disk size' is set to 'Image default (127 GB)' and 'OS disk type' is set to 'Standard HDD (locally-redundant storage)'. A note says: 'The selected VM size supports premium disks. We recommend Premium SSD for high IOPS workloads. Virtual machines with Premium SSD disks qualify for the 99.9% connectivity SLA.' There is a checked checkbox for 'Delete with VM'. Under 'Key management', 'Platform-managed key' is selected. Under 'Enable Ultra Disk compatibility', there is an unchecked checkbox. At the bottom, there are navigation buttons: '< Previous', 'Next : Networking >', 'Review + create', and 'Give feedback'.

10. Then make changes to standard.

The screenshot shows the 'Create a virtual machine' wizard in progress, specifically Step 8: 'Review + create'. The page title is 'Create a virtual machine ...'. At the top, there are three buttons: 'Help me create a low cost VM', 'Help me create a VM optimized for high availability', and 'Help me choose the right VM size for my workload'. Below these, a green bar indicates 'Validation passed'. The tabs at the top are 'Basics', 'Disks', 'Networking', 'Management', 'Monitoring', 'Advanced', 'Tags', and 'Review + create', with 'Review + create' being the active tab. Under 'Price', it shows '1 X Standard B2s by Microsoft' and 'Subscription credits apply' with a value of '3.4609 INR/hr'. There is a link for 'Pricing for other VM sizes'. Under 'TERMS', there is a note: 'By clicking "Create", I (a) agree to the legal terms and privacy statement(s) associated with the Marketplace offering(s) listed above; (b) authorize Microsoft to bill my current payment method for the fees associated with the offering(s), with the same billing frequency as my Azure subscription; and (c) agree that Microsoft may share my contact, usage and transactional information with the provider(s) of the offering(s) for support, billing and other transactional activities. Microsoft does not provide rights for third-party offerings. See the Azure Marketplace Terms for additional details.' A warning message at the bottom left says: '⚠ You have set RDP, SSH port(s) open to the internet. This is only recommended for testing. If you want to change this setting, go back to Basics tab.' At the bottom, there are navigation buttons: '< Previous', 'Next >', 'Create', 'Download a template for automation', and 'Give feedback'.

11. Then finally create and run the VM.

Microsoft Azure

CreateVm-MicrosoftWindowsDesktop.Windows-10-win10-20240906101426 | Overview

Deployment

Deployment is in progress

Deployment name: CreateVm-MicrosoftWindowsDesktop.Windows... Start time: 06/09/2024, 10:29:32

Subscription: RDBH Correlation ID: b8693396-d996-4c22-aec6-1dd3d408ab93

Resource group: RDBH

Deployment details

Resource	Type	Status	Operation details
No results.			

Give feedback

Tell us about your experience with deployment

Microsoft Defender for Cloud

Secure your apps and infrastructure

Go to Microsoft Defender for Cloud >

Free Microsoft tutorials

Start learning today >

Work with an expert

Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.

Find an Azure expert >

12. Wait for deployment.

Microsoft Azure

Home > CreateVm-MicrosoftWindowsDesktop.Windows-10-win10-20240906101426 | Overview

HarshVM

Virtual machine

Overview

Connect

Connect via Bastion

Resource group (move) : RDBH

Status : Running

Location : East US (Zone 1)

Subscription (move) : RDBH

Subscription ID : e8205d22-01fc-4dc5-9652-3078e8e1b0da

Availability zone : 1

Tags (edit) : Add tags

Operating system : Windows (Windows 10 Pro)

Size : Standard B2s (2 vcpus, 4 GiB memory)

Public IP address : 172.208.28.248

Virtual network/subnet : anshul/default

DNS name : Not configured

Health state : -

Time created : 06/09/2024, 04:59 UTC

Properties Monitoring Capabilities (8) Recommendations Tutorials

Virtual machine

Computer name	HarshVM
Operating system	Windows (Windows 10 Pro)
VM generation	V2
VM architecture	x64
Agent status	Ready
Agent version	2.7.41491.1139
Hibernation	Disabled

Networking

Public IP address	172.208.28.248 (Network interface harshvm727_z1)
Public IP address (IPv6)	-
Private IP address	10.1.0.5
Private IP address (IPv6)	-
Virtual network/subnet	anshul/default
DNS name	Configure

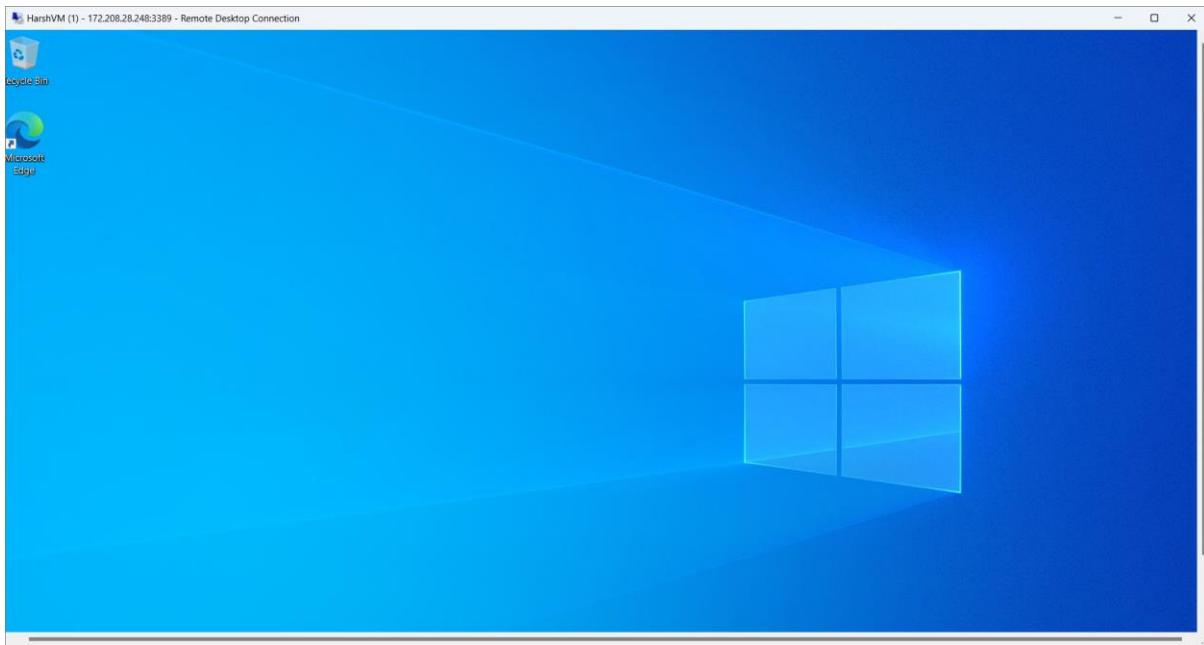
13. when deployed select connect.

The screenshot shows the Microsoft Azure portal interface. On the left, there's a navigation sidebar with various options like Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Connect, and more. Under the Connect section, 'Native RDP' is selected. The main content area displays connection details: Admin username (Harsh.polas.verma), Port (3389), and Just-in-time policy (Not configured for port 3389). Below this, a 'Most common' section shows the 'Native RDP' option again, along with 'Select' and 'Download RDP file' buttons.

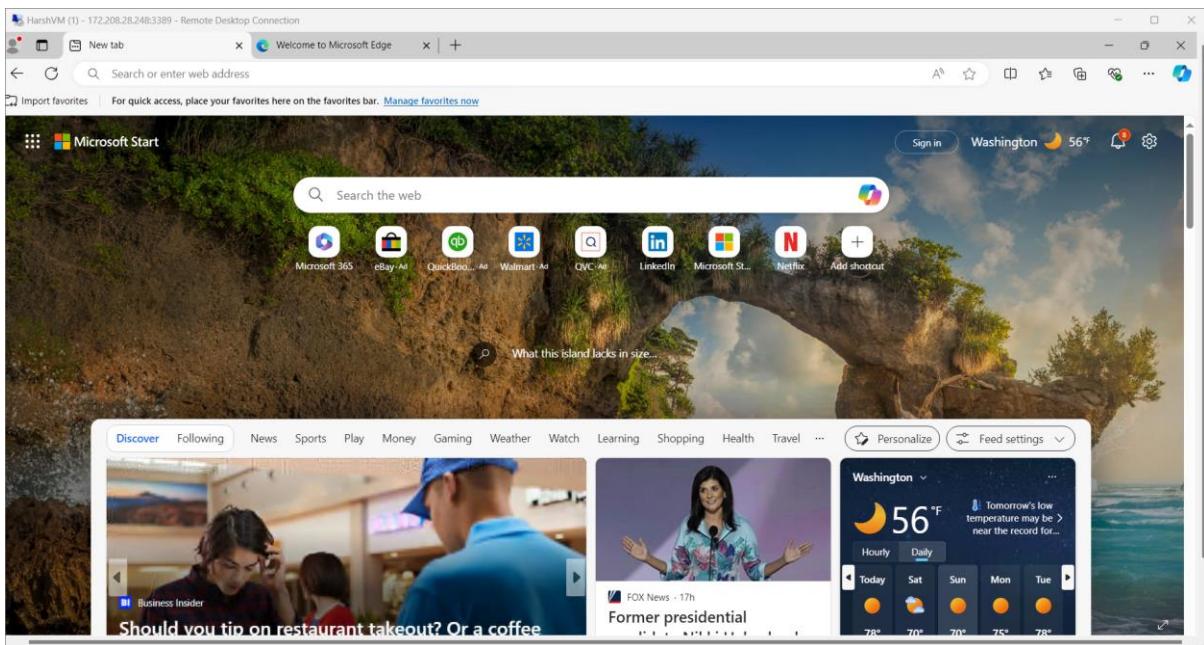
14. Then download the RDP file to run VM.

This screenshot continues from the previous one. It shows the same Azure portal interface with the 'Native RDP' connection selected. A 'Windows Security' dialog box is now open in the foreground, asking for 'Enter your credentials' to connect to the VM. The 'OK' button is highlighted in orange. In the top right corner of the browser window, a 'Downloads' notification is visible, showing a file named 'HarshVM (1).rdp'.

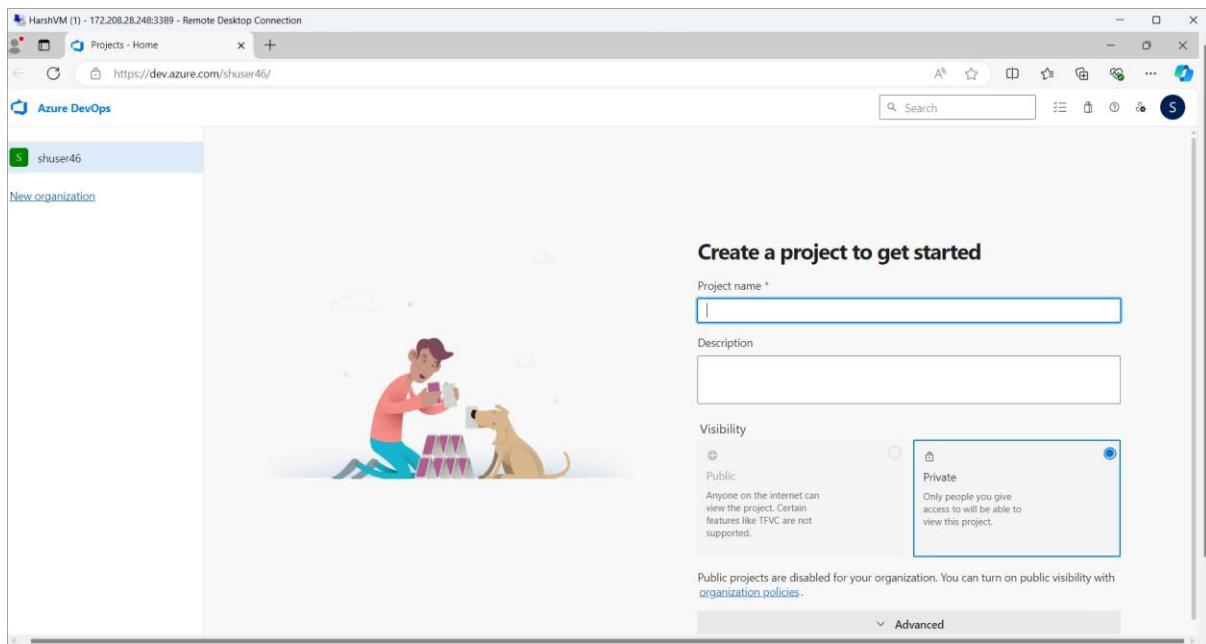
15. Now enter the password.



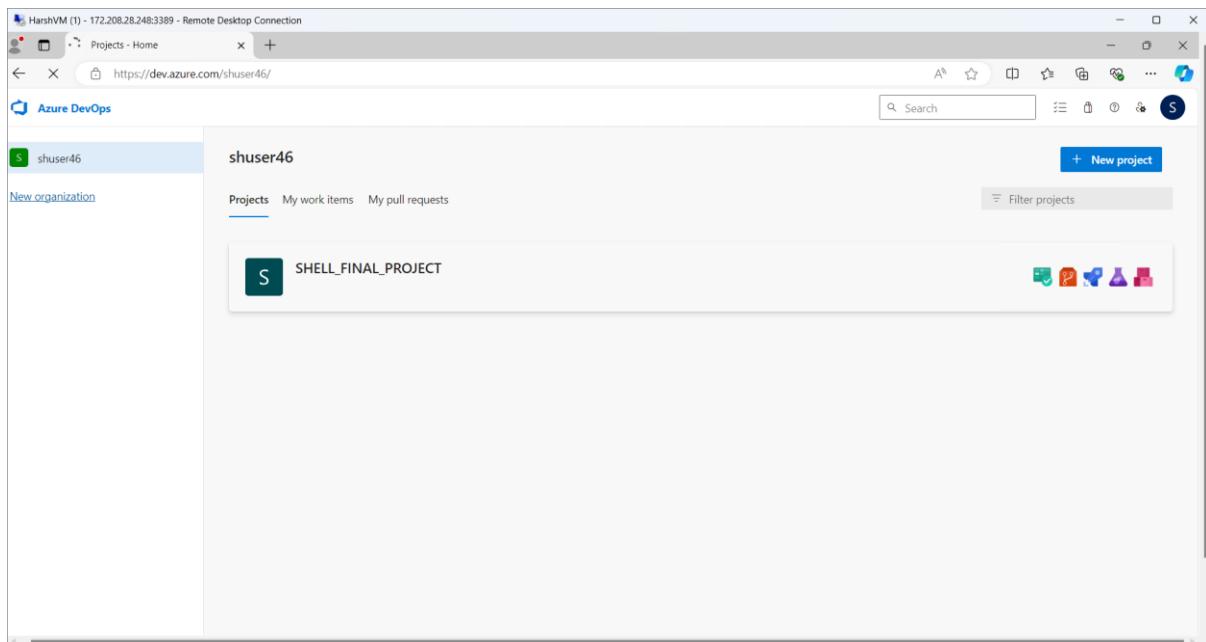
16. Successfully opened VM



17. Now open edge and search Azure DEVOPs



18. Now create new project.



19. Here is the final project for DevOPs

Website (OrangeHRM) :

"<https://opensourcedemo.orangehrmlive.com/web/index.php/auth/login>"

Objective: To test the concepts of SDLC, Agile, Software Testing, DBMS and CI/CD pipeline.

Task Instructions The below activity must be performed individually.

Follow the below instructions for the OrangeHRM website –

Test cases

File	Home	Insert	Page Layout	Formulas	Data	Review	View	Automate	Help	Comments	Share
D6	v	x	✓	fx	v	1. Navigate to the employee management page.					
1	A	B	C	D	E	F	G	H	I	J	K
Employee Management	1	Add New Employee	Test Steps	Test Data	Expected Result	Actual Result	Status	Status			
	2	Edit Employee Details	1. Navigate to the employee list page. 2. Select an employee to edit. 3. Update the employee details. 4. Click "Save."	New Department: IT	Employee details are updated successfully, and the changes are reflected in the employee list.	Employee details updated to IT department successfully.	Pass	Pass			
	3	Delete Employee Record	1. Navigate to the employee list. 2. Select an employee to delete. 3. Click "Delete" and confirm the action.	Employee ID: E54321	Employee record is deleted successfully.	Employee E54321 was deleted successfully.	Pass	Pass			
	4	Add Employee with Duplicate ID	1. Navigate to the employee management page. 2. Click "Add" button. 3. Enter details with an existing employee ID. 4. Click "Save."	E12345 (existing ID)	Error message indicating the employee ID already exists.	Error message: Employee ID already exists.	Pass	Pass			
	5	Save Employee Record with Missing Fields	1. Navigate to the employee management page. 2. Click "Add" button. 3. Leave required fields empty. 4. Click "Save."	Blank fields.	Error message indicating missing required fields.	Error message: Required fields are missing.	Pass	Pass			
7	USER STORY	Test Case ID	Test Case Description	Test Steps	Test Data	Expected Result	Actual Result	Status	Status		
Leave Request Management	1	Submit Leave Request	1. Navigate to the leave request page. 2. Enter valid leave details. 3. Click "Submit."	Leave Type: Vacation, Start Date: 2024-10-01, End Date: 2024-10-05	Leave request is submitted successfully with a confirmation message.	Leave request for Vacation from 2024-10-01 to 2024-10-05 submitted successfully.	Pass	Pass			
	2	View Leave History	1. Navigate to the leave history page. 2. Verify that past leave requests are listed.	Leave history data.	Leave history displays all previous leave requests correctly.	Leave history displayed previous requests accurately.	Pass	Pass			
	3	Cancel Leave Request	1. Navigate to the leave request history page. 2. Select a leave request to cancel. 3. Click "Cancel" and confirm the action.	Leave Request ID: L12345	Leave request is canceled successfully, and the status is updated.	Leave request L12345 canceled successfully.	Pass	Pass			
	4	Leave Request Submission with Empty Fields	1. Navigate to the leave request page. 2. Enter valid leave details. 3. Click "Submit."	Start Date: 2024-10-05, End Date: 2024-10-01	Error message indicating invalid date range.	Error message: End date cannot be before start date.	Pass	Pass			
13	USER STORY	Test Case ID	Test Case Description	Test Steps	Test Data	Expected Result	Actual Result	Status	Status		
Performance Review Management	1	Create Performance Review	1. Navigate to the performance review section. 2. Enter performance review details for an employee. 3. Click "Save."	Employee: Bob Johnson, Review: Excellent performance.	Performance review is created successfully with a confirmation message.	Performance review for Bob Johnson created successfully.	Pass	Pass			
	2	View Performance Review	1. Navigate to the performance review list. 2. Select a performance review to view.	Employee: Alice Smith	Performance review details for the selected employee are displayed correctly.	Performance review for Alice Smith displayed correctly.	Pass	Pass			
	3	Update Performance Review	1. Navigate to the performance review section. 2. Select a performance review to edit. 3. Modify review details. 4. Click "Save."	New Review: Needs improvement.	Performance review is updated successfully.	Performance review updated successfully.	Pass	Pass			
	4	Create Performance Review for Nonexistent Employee	1. Navigate to the performance review section. 2. Attempt to update a performance review with invalid data (e.g., nonexistent employee ID).	Nonexistent Employee	Error message indicating that the employee does not exist.	Error message: Employee does not exist.	Pass	Pass			

Test cases

File	Home	Insert	Page Layout	Formulas	Data	Review	View	Automate	Help	Comments	Share
D6	v	x	✓	fx	v	1. Navigate to the employee management page.					
7	A	B	C	D	E	F	G	H	I	J	K
Leave Request Management	1	Submit Leave Request	Test Steps	Test Data	Expected Result	Actual Result	Status	Status			
	2	View Leave History	1. Navigate to the leave history page. 2. Verify that past leave requests are listed.	Leave history data.	Leave history displays all previous leave requests correctly.	Leave history displayed previous requests accurately.	Pass	Pass			
	3	Cancel Leave Request	1. Navigate to the leave request history page. 2. Select a leave request to cancel. 3. Click "Cancel" and confirm the action.	Leave Request ID: L12345	Leave request is canceled successfully, and the status is updated.	Leave request L12345 canceled successfully.	Pass	Pass			
	4	Submit Leave Request with Invalid Dates	1. Navigate to the leave request page. 2. Enter an end date before the start date. 3. Click "Submit."	Start Date: 2024-10-05, End Date: 2024-10-01	Error message indicating invalid date range.	Error message: End date cannot be before start date.	Pass	Pass			
	5	Leave Request Submission with Empty Fields	1. Navigate to the leave request page. 2. Leave required fields empty. 3. Click "Submit."	Empty required fields.	Error message indicating that required fields are missing.	Error message: Required fields must be filled.	Pass	Pass			
13	USER STORY	Test Case ID	Test Case Description	Test Steps	Test Data	Expected Result	Actual Result	Status	Status		
Performance Review Management	1	Create Performance Review	1. Navigate to the performance review section. 2. Enter performance review details for an employee. 3. Click "Save."	Employee: Bob Johnson, Review: Excellent performance.	Performance review is created successfully with a confirmation message.	Performance review for Bob Johnson created successfully.	Pass	Pass			
	2	View Performance Review	1. Navigate to the performance review list. 2. Select a performance review to view.	Employee: Alice Smith	Performance review details for the selected employee are displayed correctly.	Performance review for Alice Smith displayed correctly.	Pass	Pass			
	3	Update Performance Review	1. Navigate to the performance review section. 2. Select a performance review to edit. 3. Modify review details. 4. Click "Save."	New Review: Needs improvement.	Performance review is updated successfully.	Performance review updated successfully.	Pass	Pass			
	4	Create Performance Review for Nonexistent Employee	1. Navigate to the performance review section. 2. Attempt to update a performance review with invalid data (e.g., nonexistent employee ID).	Nonexistent Employee	Error message indicating that the employee does not exist.	Error message: Employee does not exist.	Pass	Pass			

HarshVM (1) - 172.208.28.248:3389 - Remote Desktop Connection

Azure DevOps shuser46 / SHELL_FINAL_PROJECT / Boards / Work items

Search

SHELL_FINAL_PROJECT + Try the New Boards Hub for improved performance, accessibility, and new features. [Click here](#) to learn more.

Overview Boards Work items Boards Backlogs Sprints Queries Delivery Plans Analytics views Repos Pipelines Test Plans Artifacts Project settings

Recently updated | Back to Work Items

31 of 31 ↑ ↓

EPIC 1 Enhancing Core Functionalities of OrangeHRM

Unassigned 0 comments Add tag

State: New Area: SHELL_FINAL_PROJECT Reason: New Iteration: SHELL_FINAL_PROJECT Updated by shuser46: 2h ago

Links

+ Add link

Link ↑	State	Latest Update	Comment
Child (5)			
2 Employee Management	... ● New	Updated 2 hours ago	
8 Leave Request Management	● New	Updated 25 minutes ago	
14 Performance Review Management	● New	Updated 18 minutes ago	
15 Reporting and Analytics	● New	Updated 17 minutes ago	
16 System Security and Access Control	● New	Updated 16 minutes ago	

Type here to search

8:34 AM ENG 9/6/2024

HarshVM (1) - 172.208.28.248:3389 - Remote Desktop Connection

Azure DevOps shuser46 / SHELL_FINAL_PROJECT / Boards / Work items

Search

SHELL_FINAL_PROJECT + Try the New Boards Hub for improved performance, accessibility, and new features. [Click here](#) to learn more.

Overview Boards Work items Boards Backlogs Sprints Queries Delivery Plans Analytics views Repos Pipelines Test Plans Artifacts Project settings

Recently updated | Back to Work Items

30 of 31 ↑ ↓

USER STORY 2 Employee Management

Unassigned 0 comments Add tag

State: New Area: SHELL_FINAL_PROJECT Reason: New Iteration: SHELL_FINAL_PROJECT Updated by shuser46: 2h ago

Links

+ Add link

Link ↑	State	Latest Update	Comment
Child (5)			
6 Add Employee with Duplicate ID	... ● Design	Updated 29 minutes ago	
3 Add New Employee	● Design	Updated 33 minutes ago	
5 Delete Employee Record	● Design	Updated 30 minutes ago	
4 Save Employee Record with Missing Fields	● Design	Updated 31 minutes ago	
7 Save Employee Record with Missing Fields	● Design	Updated 28 minutes ago	
Parent			
1 Enhancing Core Functionalities of OrangeHRM	● New	Updated 2 hours ago	

Type here to search

8:34 AM ENG 9/6/2024

HarshVM (1) - 172.208.28.248:3389 - Remote Desktop Connection

Azure DevOps shuser46 / SHELL_FINAL_PROJECT / Boards / Work items

S SHELL_FINAL_PROJECT + Try the New Boards Hub for improved performance, accessibility, and new features. [Click here](#) to learn more.

Overview Recently updated | Back to Work Items 24 of 31 ↑ ↓

Boards

Work items

Boards

Backlogs

Sprints

Queries

Delivery Plans

Analytics views

Repos

Pipelines

Test Plans

Artifacts

Project settings

USER STORY 8

8 Leave Request Management

Unassigned 0 comments Add tag

Save Follow Refresh More

State: New Area: SHELL_FINAL_PROJECT
Reason: New Iteration: SHELL_FINAL_PROJECT Updated by shuser46: 25m ago

Details (6)

Links

+ Add link

	Link	State	Latest Update	Comment
+	11 Cancel Leave Request	Design	Updated 22 minutes ago	
+	13 Leave Request Submission with Empty Fields	Design	Updated 21 minutes ago	
+	12 Submit Leave Request	Design	Updated 23 minutes ago	
+	12 Submit Leave Request with Invalid Dates	Design	Updated 22 minutes ago	
+	10 View Leave History	Design	Updated 23 minutes ago	
+	1 Enhancing Core Functionalities of OrangeHRM	New	Updated 2 hours ago	

Type here to search

HarshVM (1) - 172.208.28.248:3389 - Remote Desktop Connection

Azure DevOps shuser46 / SHELL_FINAL_PROJECT / Boards / Work items

SHELL_FINAL_PROJECT + Try the New Boards Hub for improved performance, accessibility, and new features. [Click here](#) to learn more.

Overview Recently updated | Back to Work Items 18 of 31 ↑ ↓

Boards

Work items

Unassigned 0 comments Add tag

User Story 14 14 Performance Review Management

State: New Area: SHELL_FINAL_PROJECT

Reason: New Iteration: SHELL_FINAL_PROJECT Updated by shuser46: 19m ago

Details ⏪ (6) Links

Links + Add link

Link	State	Latest Update	Comment
Child (5)			
↳ 22 Create Performance Review	Design	Updated 14 minutes ago	
↳ 25 Create Performance Review for Nonexistent Employee	Design	Updated 13 minutes ago	
↳ 24 Update Performance Review	Design	Updated 13 minutes ago	
↳ 26 Update Performance Review with Invalid Data	Design	Updated 12 minutes ago	
↳ 23 View Performance Review	Design	Updated 13 minutes ago	
Parent	New	Updated 2 hours ago	
↳ 1 Enhancing Core Functionalities of OrangeHRM			

+ Add link

Link ↑ State Latest Update Comment

↳ Child (5)

↳ 22 Create Performance Review Design Updated 14 minutes ago

↳ 25 Create Performance Review for Nonexistent Employee Design Updated 13 minutes ago

↳ 24 Update Performance Review Design Updated 13 minutes ago

↳ 26 Update Performance Review with Invalid Data Design Updated 12 minutes ago

↳ 23 View Performance Review Design Updated 13 minutes ago

Parent New Updated 2 hours ago

↳ 1 Enhancing Core Functionalities of OrangeHRM

Project settings

Type here to search

8:35 AM 9/6/2024 ENG

The screenshot shows the Azure DevOps interface for the 'SHELL_FINAL_PROJECT' board. The left sidebar is collapsed, showing options like Overview, Boards, Work items, Boards, Backlogs, Sprints, Queries, Delivery Plans, and Analytics views. The main area displays a work item titled '15 Reporting and Analytics'. The work item details are as follows:

State	New	Area	SHELL_FINAL_PROJECT
Reason	New	Iteration	SHELL_FINAL_PROJECT

The 'Links' section shows five child items under 'Apply Filters to Report':

- 29 Apply Filters to Report (Design, Updated 4 minutes ago)
- 28 Export Report to Excel (Design, Updated 5 minutes ago)
- 31 Export Report When Export Function is Disabled (Design, Updated 4 minutes ago)
- 27 Generate Performance Report (Design, Updated 5 minutes ago)
- 30 Generate Report with No Filters (Design, Updated 4 minutes ago)

There is also one parent item listed:

- 1 Enhancing Core Functionalities of OrangeHRM (New, Updated 2 hours ago)

The status bar at the bottom indicates it's 8:35 AM on 9/6/2024.

The screenshot shows the Azure DevOps interface for the 'SHELL_FINAL_PROJECT' board. The left sidebar is collapsed, showing options like Overview, Boards, Work items, Boards, Backlogs, Sprints, Queries, Delivery Plans, and Analytics views. The main area displays a work item titled '16 System Security and Access Control'. The work item details are as follows:

State	New	Area	SHELL_FINAL_PROJECT
Reason	New	Iteration	SHELL_FINAL_PROJECT

The 'Links' section shows five child items under 'Access Admin Section Without Proper Permissions':

- 35 Access Admin Section Without Proper Permissions (Design, Updated 2 minutes ago)
- 32 Access Control for Admin User (Design, Updated 3 minutes ago)
- 36 Login with Incorrect Credentials (Design, Updated 2 minutes ago)
- 34 Password Change Functionality (Design, Updated 3 minutes ago)
- 33 Restrict Access for Non-Admin User (Design, Updated 3 minutes ago)

There is also one parent item listed:

- 1 Enhancing Core Functionalities of OrangeHRM (New, Updated 2 hours ago)

The status bar at the bottom indicates it's 8:35 AM on 9/6/2024.

USER STORY 1: Employee Management

As an HR administrator, I want to be able to add, edit, and delete employee records to manage the workforce efficiently.

SDLC Tasks:

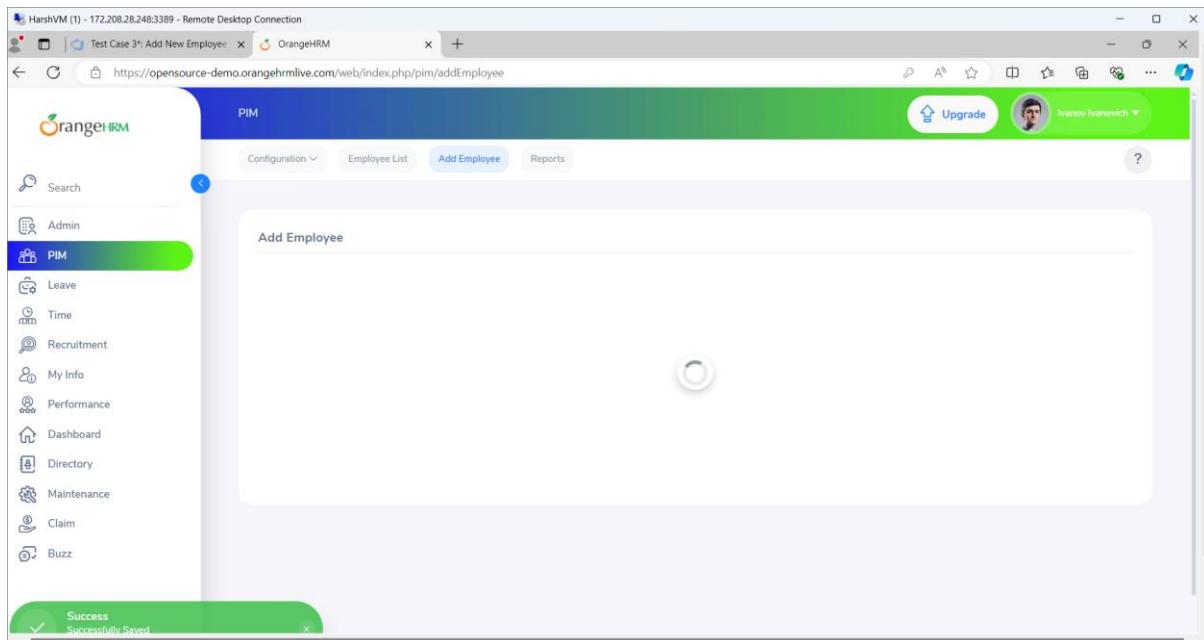
- Planning:** Define requirements for employee management features.
- Designing:** Create design specifications for adding, editing, and deleting employee records.
- Coding:** Implement the functionality to manage employee records.

- **Testing:** Verify that all functionalities work as intended.
- **Feedback:** Review and refine functionalities based on testing results.

Testing Phase - Test Cases:

Positive Test Case 1: Add New Employee

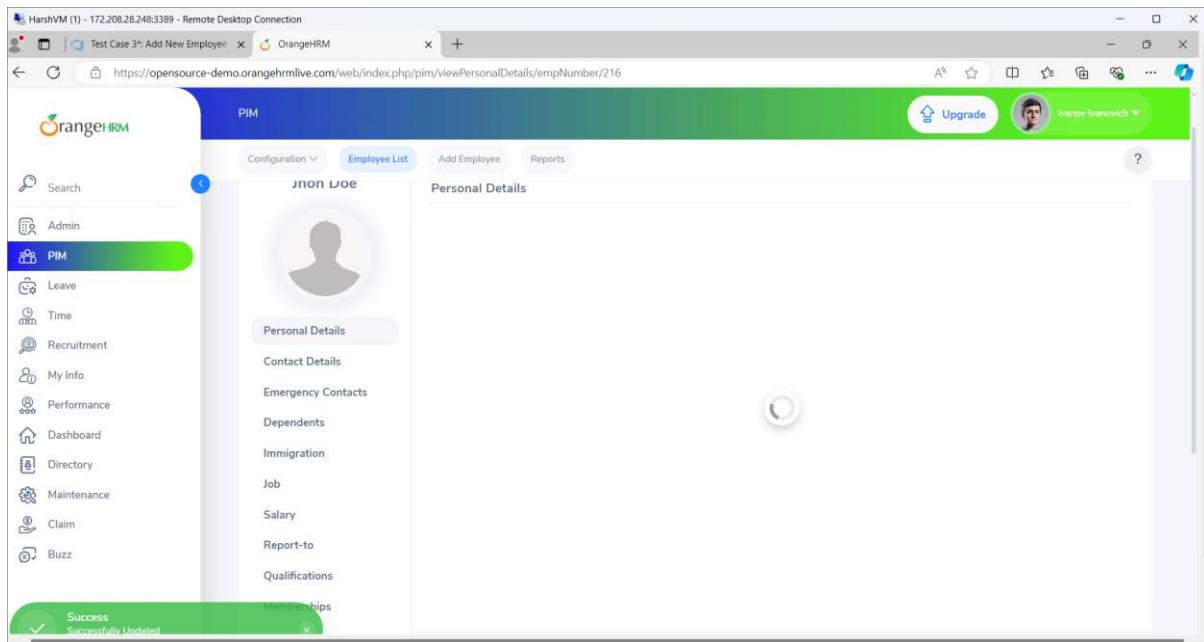
- **Test Steps:**
 1. Navigate to the employee management page.
 2. Click the “Add” button.
 3. Fill in valid employee details (e.g., name, ID, department).
 4. Click “Save.”
- **Test Data:** Name: John Doe, ID: E12345, Department: HR
- **Expected Result:** Employee is added successfully, and a confirmation message is displayed.
- **Actual Result:** Employee John Doe was added successfully.
- **Status:** Pass



Positive Test Case 2: Edit Employee Details

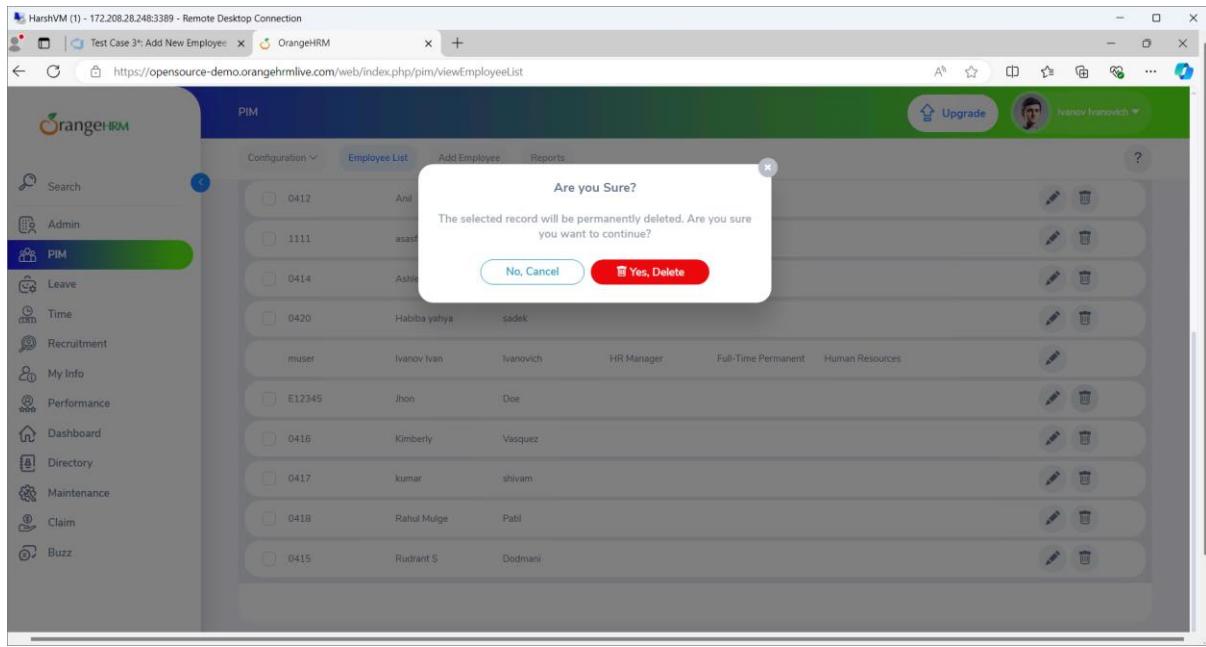
- **Test Steps:**
 1. Navigate to the employee list page.

2. Select an employee to edit.
 3. Update the employee details.
 4. Click “Save.”
- **Test Data:** New Department: IT
 - **Expected Result:** Employee details are updated successfully, and the changes are reflected in the employee list.
 - **Actual Result:** Employee details updated to IT department successfully.
 - **Status:** Pass



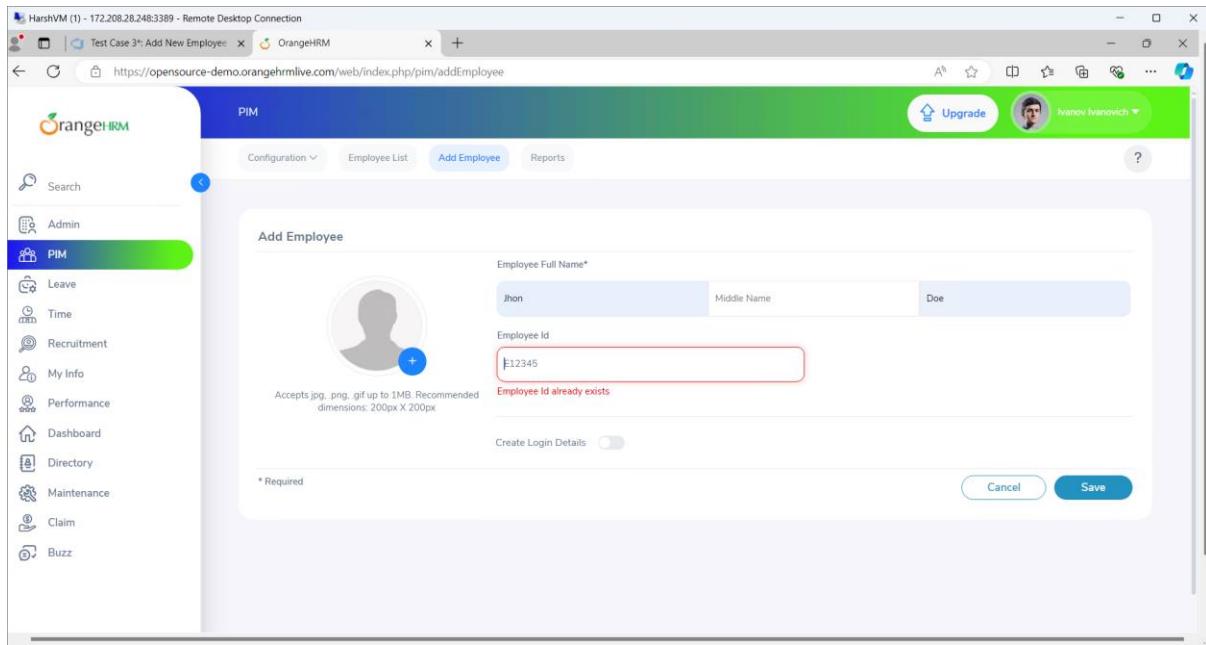
Positive Test Case 3: Delete Employee Record

- **Test Steps:**
 1. Navigate to the employee list.
 2. Select an employee to delete.
 3. Click “Delete” and confirm the action.
- **Test Data:** Employee ID: E54321
- **Expected Result:** Employee record is deleted successfully.
- **Actual Result:** [To be filled after testing] (e.g., "Employee E54321 was deleted successfully.")
- **Status:** [Pass/Fail] (e.g., "Pass")



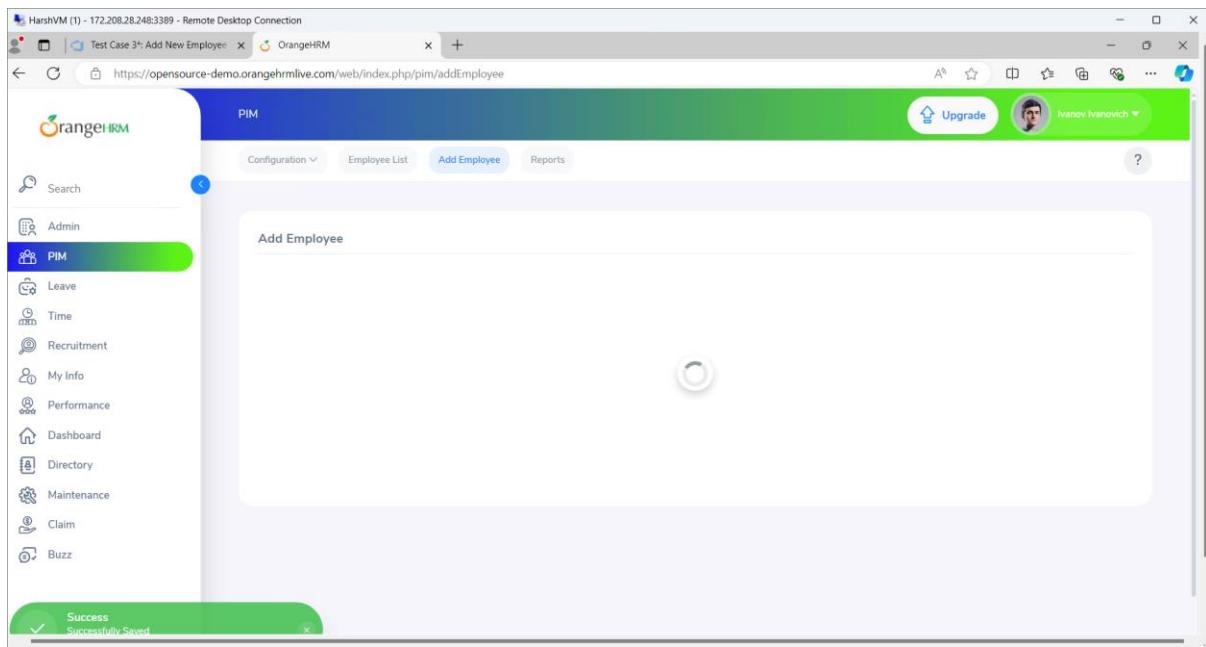
Negative Test Case 1: Add Duplicate Employee

- **Test Steps:**
 1. Navigate to the employee management page.
 2. Click the “Add” button.
 3. Enter details for an existing employee.
 4. Click “Save.”
- **Test Data:** Existing Employee ID: E12345
- **Expected Result:** Error message indicating that the employee already exists.
- **Actual Result:** Error message displayed: Employee already exists.
- **Status:** Pass



Negative Test Case 2: Save Employee with Invalid Data

- **Test Steps:**
 1. Navigate to the employee management page.
 2. Click the “Add” button.
 3. Enter invalid data (e.g., blank fields).
 4. Click “Save.”
- **Test Data:** Blank Employee ID
- **Expected Result:** Error message indicating that required fields are missing.
- **Actual Result:** NO error file
- **Status:** FAIL



Bug 1: No error while saving without employee id

Bug 2: Save button unresponsive when required fields are empty.

User Story 2: Leave Request Management

Description: As an employee, I need to request leave online and view my leave history to manage my time off effectively.

SDLC Tasks:

- **Planning:** Define requirements for leave request submission and history retrieval.
- **Designing:** Create designs for leave request forms and history pages.
- **Coding:** Implement leave request submission and history functionalities.
- **Testing:** Verify the functionality of leave requests and history.
- **Feedback:** Gather feedback from users and refine features.

Testing Phase - Test Cases:

1. Positive Test Case 1: Submit Leave Request

o Test Steps:

1. Navigate to the leave request page.

2. Enter valid leave details.
 3. Click “Submit.”
- **Test Data:** Leave Type: Vacation, Start Date: 2024-10-01, End Date: 2024-10-05
 - **Expected Result:** Leave request is submitted successfully with a confirmation message.
 - **Actual Result:** Leave request for Vacation from 2024-10-01 to 2024-10-05 submitted successfully.
 - **Status:** Pass

2. Positive Test Case 2: View Leave History

- **Test Steps:**
 1. Navigate to the leave history page.
 2. Verify that past leave requests are listed.
- **Test Data:** Leave history data.
- **Expected Result:** Leave history displays all previous leave requests correctly.
- **Actual Result:** [To be filled after testing] (e.g., "Leave history displayed previous requests accurately.")
- **Status:** [Pass/Fail] (e.g., "Pass")

3. Positive Test Case 3: Cancel Leave Request

- **Test Steps:**
 1. Navigate to the leave request history page.
 2. Select a leave request to cancel.
 3. Click “Cancel” and confirm the action.
- **Test Data:** Leave Request ID: L12345
- **Expected Result:** Leave request is canceled successfully, and the status is updated.
- **Actual Result:** [To be filled after testing] (e.g., "Leave request L12345 canceled successfully.")

- **Status:** [Pass/Fail] (e.g., "Pass")

4. Negative Test Case 1: Submit Leave Request with Invalid Dates

- **Test Steps:**

1. Navigate to the leave request page.
2. Enter an end date before the start date.
3. Click "Submit."

- **Test Data:** Start Date: 2024-10-05, End Date: 2024-10-01
- **Expected Result:** Error message indicating invalid date range.
- **Actual Result:** [To be filled after testing] (e.g., "Error message: End date cannot be before start date.")
- **Status:** [Pass/Fail] (e.g., "Pass")

5. Negative Test Case 2: Leave Request Submission with Empty Fields

- **Test Steps:**

1. Navigate to the leave request page.
2. Leave required fields empty.
3. Click "Submit."

- **Test Data:** Empty required fields.
- **Expected Result:** Error message indicating that required fields are missing.
- **Actual Result:** [To be filled after testing] (e.g., "Error message: Required fields must be filled.")
- **Status:** [Pass/Fail] (e.g., "Pass")

Bugs:

1. **Bug 1:** Error message for invalid dates is not displayed.
2. **Bug 2:** Leave request form submission fails without proper validation feedback.

User Story 3: Performance Review Management

Description: As a manager, I need to create and review performance evaluations for team members to assess their progress and provide feedback.

SDLC Tasks:

- **Planning:** Identify performance review requirements and evaluation criteria.
- **Designing:** Develop design for performance review forms and review interfaces.
- **Coding:** Implement performance review creation and viewing functionalities.
- **Testing:** Validate the performance review creation and retrieval process.
- **Feedback:** Gather feedback and make necessary adjustments.

Testing Phase - Test Cases:

1. Positive Test Case 1: Create Performance Review

- **Test Steps:**
 1. Navigate to the performance review section.
 2. Enter performance review details for an employee.
 3. Click “Save.”
- **Test Data:** Employee: Bob Johnson, Review: Excellent performance.
- **Expected Result:** Performance review is created successfully with a confirmation message.
- **Actual Result:** [To be filled after testing] (e.g., "Performance review for Bob Johnson created successfully.")
- **Status:** [Pass/Fail] (e.g., "Pass")

2. Positive Test Case 2: View Performance Review

- **Test Steps:**
 1. Navigate to the performance review list.
 2. Select a performance review to view.
- **Test Data:** Employee: Alice Smith
- **Expected Result:** Performance review details for the selected employee are displayed correctly.
- **Actual Result:** [To be filled after testing] (e.g., "Performance review for Alice Smith displayed correctly.")
- **Status:** [Pass/Fail] (e.g., "Pass")

3. Positive Test Case 3: Update Performance Review

- **Test Steps:**

1. Navigate to the performance review section.
 2. Select a performance review to edit.
 3. Modify review details.
 4. Click “Save.”
- **Test Data:** New Review: Needs improvement.
 - **Expected Result:** Performance review is updated successfully.
 - **Actual Result:** [To be filled after testing] (e.g., "Performance review updated successfully.")
 - **Status:** [Pass/Fail] (e.g., "Pass")

4. Negative Test Case 1: Create Performance Review for Nonexistent Employee

- **Test Steps:**
 1. Navigate to the performance review section.
 2. Enter details for a nonexistent employee.
 3. Click “Save.”
- **Test Data:** Employee: Nonexistent Employee
- **Expected Result:** Error message indicating that the employee does not exist.
- **Actual Result:** [To be filled after testing] (e.g., "Error message: Employee does not exist.")
- **Status:** [Pass/Fail] (e.g., "Pass")

5. Negative Test Case 2: Update Performance Review with Invalid Data

- **Test Steps:**
 1. Navigate to the performance review section.
 2. Attempt to update a performance review with invalid data (e.g., empty review field).
 3. Click “Save.”
- **Test Data:** Empty Review Field
- **Expected Result:** Error message indicating invalid data.

- **Actual Result:** [To be filled after testing] (e.g., "Error message: Review field cannot be empty.")
- **Status:** [Pass/Fail] (e.g., "Pass")

Bugs:

1. **Bug 1:** Error message for nonexistent employee not displayed.
 2. **Bug 2:** Performance review update fails without validation.
-

User Story 4: Reporting and Analytics

Description: As a report administrator, I need to generate and export reports to analyze employee performance, leave trends, and other key metrics.

SDLC Tasks:

- **Planning:** Define reporting requirements and data sources.
- **Designing:** Create report templates and export options.
- **Coding:** Develop report generation and export functionalities.
- **Testing:** Test report generation and export features.
- **Feedback:** Review feedback and refine reporting functionalities.

Testing Phase - Test Cases:

1. Positive Test Case 1: Generate Performance Report

- **Test Steps:**
 1. Navigate to the reporting section.
 2. Select "Performance Report" and apply filters.
 3. Click "Generate."
- **Test Data:** Report Type: Performance, Filters: Department: IT
- **Expected Result:** Performance report is generated successfully with applied filters.
- **Actual Result:** [To be filled after testing] (e.g., "Performance report generated successfully.")
- **Status:** [Pass/Fail] (e.g., "Pass")

2. Positive Test Case 2: Export Report to Excel

- **Test Steps:**
 1. Generate a report.
 2. Click “Export” and choose Excel format.
- **Test Data:** Report in Excel format.
- **Expected Result:** Report is exported successfully to Excel format.
- **Actual Result:** [To be filled after testing] (e.g., "Report exported to Excel format successfully.")
- **Status:** [Pass/Fail] (e.g., "Pass")

3. Positive Test Case 3: Apply Filters to Report

- **Test Steps:**
 1. Navigate to the reporting section.
 2. Generate a report and apply different filters.
- **Test Data:** Filters: Date Range: 2024-01-01 to 2024-09-30
- **Expected Result:** Report data is filtered according to the applied filters.
- **Actual Result:** [To be filled after testing] (e.g., "Report filtered successfully according to the specified date range.")
- **Status:** [Pass/Fail] (e.g., "Pass")

4. Negative Test Case 1: Generate Report with No Filters

- **Test Steps:**
 1. Navigate to the reporting section.
 2. Attempt to generate a report without applying any filters.
- **Test Data:** No filters applied.
- **Expected Result:** Error message indicating that filters are required.
- **Actual Result:** [To be filled after testing] (e.g., "Error message: Filters are required to generate the report.")
- **Status:** [Pass/Fail] (e.g., "Pass")

5. Negative Test Case 2: Export Report When Export Function is Disabled

- **Test Steps:**
 1. Generate a report.

2. Attempt to export the report while the export function is disabled.

- **Test Data:** Generated report.
- **Expected Result:** Error message indicating that the export function is disabled.
- **Actual Result:** [To be filled after testing] (e.g., "Error message: Export function is currently disabled.")
- **Status:** [Pass/Fail] (e.g., "Pass")

Bugs:

1. **Bug 1:** Export function for reports is not available despite being enabled.
 2. **Bug 2:** Report generation fails with incorrect filter application.
-

User Story 5: System Security and Access Control

Description: As a system administrator, I need to ensure that user access and security controls are correctly implemented to protect sensitive data.

SDLC Tasks:

- **Planning:** Define security requirements and access control mechanisms.
- **Designing:** Design security and access control features.
- **Coding:** Implement security controls and user access restrictions.
- **Testing:** Test security features and access control.
- **Feedback:** Collect feedback and make necessary adjustments.

Testing Phase - Test Cases:

1. Positive Test Case 1: Access Control for Admin User

- **Test Steps:**
 1. Log in as an Admin user.
 2. Navigate to admin-only sections.
- **Test Data:** Admin credentials.
- **Expected Result:** Admin user should have access to admin-only sections.

- **Actual Result:** [To be filled after testing] (e.g., "Admin user accessed all admin sections successfully.")
- **Status:** [Pass/Fail] (e.g., "Pass")

2. Positive Test Case 2: Restrict Access for Non-Admin User

- **Test Steps:**
 1. Log in as a non-admin user.
 2. Attempt to access admin-only sections.
- **Test Data:** Non-admin credentials.
- **Expected Result:** Non-admin user should be restricted from accessing admin-only sections.
- **Actual Result:** [To be filled after testing] (e.g., "Non-admin user was restricted from admin sections successfully.")
- **Status:** [Pass/Fail] (e.g., "Pass")

3. Positive Test Case 3: Password Change Functionality

- **Test Steps:**
 1. Log in as a user.
 2. Navigate to the profile settings.
 3. Change the password.
 4. Log out and log in with the new password.
- **Test Data:** New Password: NewPass123!
- **Expected Result:** Password is changed successfully and user can log in with the new password.
- **Actual Result:** [To be filled after testing] (e.g., "Password changed successfully and login with new password was successful.")
- **Status:** [Pass/Fail] (e.g., "Pass")

4. Negative Test Case 1: Access Admin Section Without Proper Permissions

- **Test Steps:**
 1. Log in as a user without admin permissions.
 2. Attempt to access admin-only sections.

- **Test Data:** Non-admin credentials.
- **Expected Result:** Access should be denied with an appropriate error message.
- **Actual Result:** [To be filled after testing] (e.g., "Access denied message displayed.")
- **Status:** [Pass/Fail] (e.g., "Pass")

5. Negative Test Case 2: Login with Incorrect Credentials

- **Test Steps:**
 1. Navigate to the login page.
 2. Enter incorrect credentials.
 3. Click "Login."
- **Test Data:** Username: invaliduser, Password: invalidpass
- **Expected Result:** Error message indicating invalid credentials.
- **Actual Result:** [To be filled after testing] (e.g., "Error message: Invalid credentials.")
- **Status:** [Pass/Fail] (e.g., "Pass")

Bugs:

1. **Bug 1:** Error message for invalid credentials is not displayed.
2. **Bug 2:** Restricted access to admin sections is bypassed due to permissions error.

5. Execute All the Test Cases Manually

Steps to Execute:

1. Log in to the application with the provided credentials.
2. Navigate to the relevant sections based on the test cases.
3. Perform the test steps as detailed in each test case.
4. Record the actual results and compare them with the expected results.
5. Mark each test case as pass or fail based on the results.

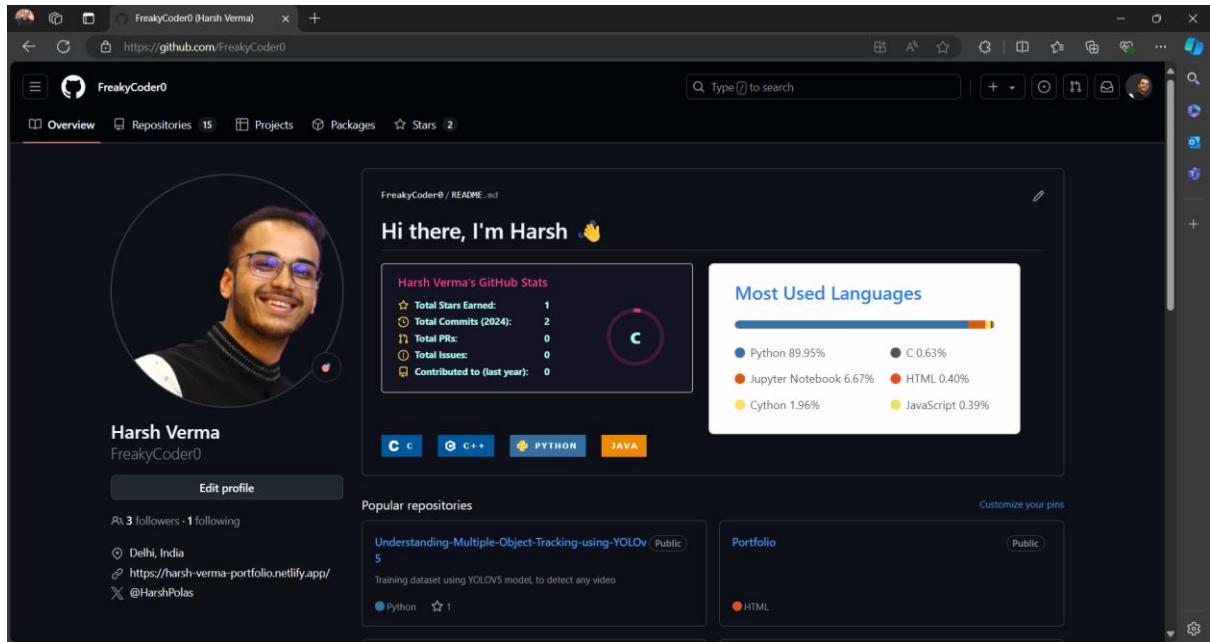
6. Create a Defect Report for Each Failed or Blocked Test Case

Example Defect Report:

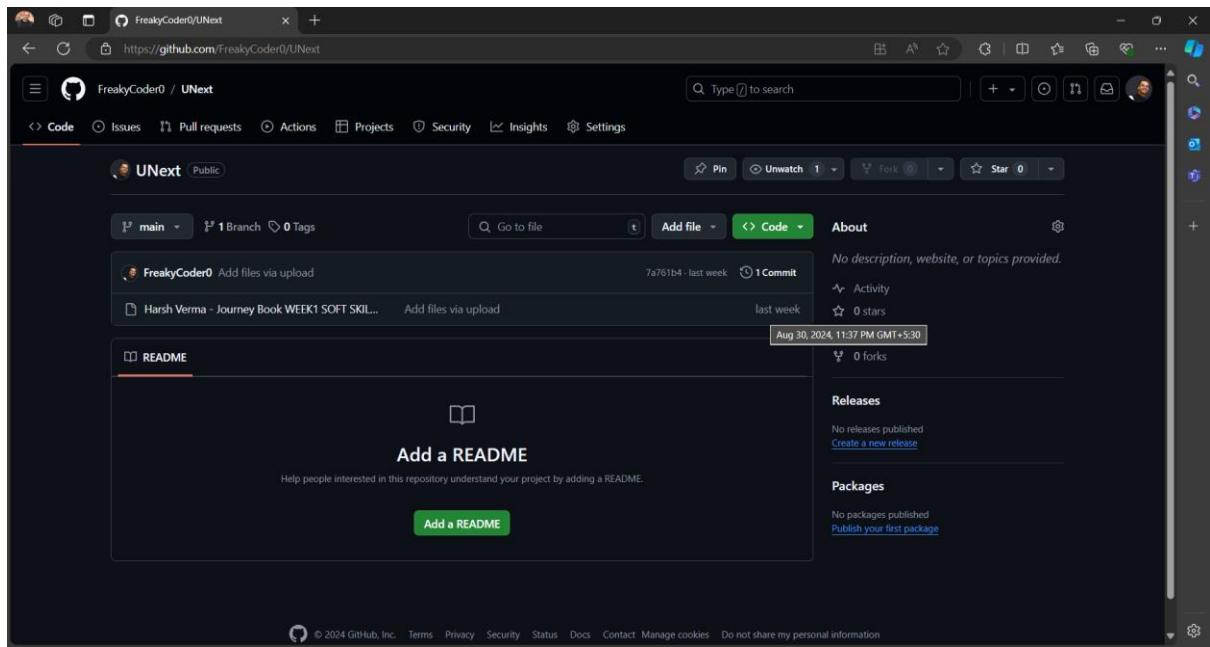
- **Defect ID:** DEF001
- **Summary:** Error message for duplicate ID not displayed.
- **Description:** When attempting to add an employee with an existing ID, no error message is shown.
- **Steps to Reproduce:**
 1. Navigate to the employee management page.
 2. Enter an existing ID.
 3. Click “Save.”
- **Expected Result:** Error message indicating the ID already exists.
- **Actual Result:** No error message displayed.
- **Severity:** Major
- **Status:** Open
- **Assigned To:** Development Team

This guide should help you effectively manage the sprint, user stories, and testing processes for the OrangeHRM application.

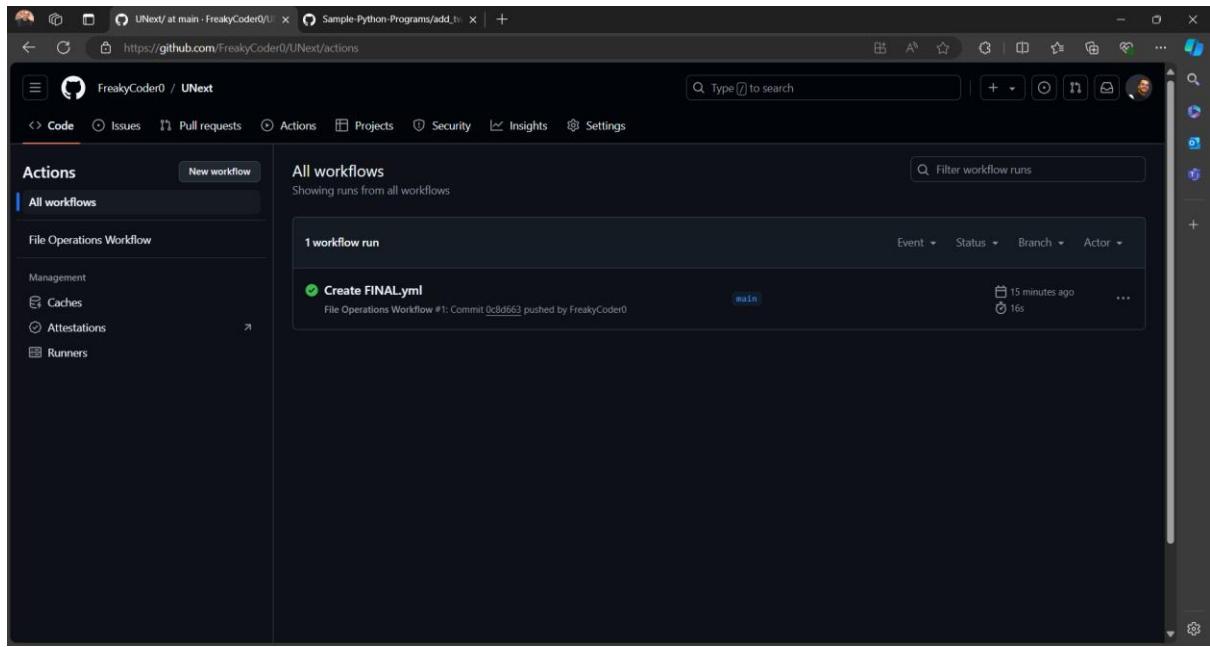
Implementing the CI/CD Pipeline using GitHub Actions



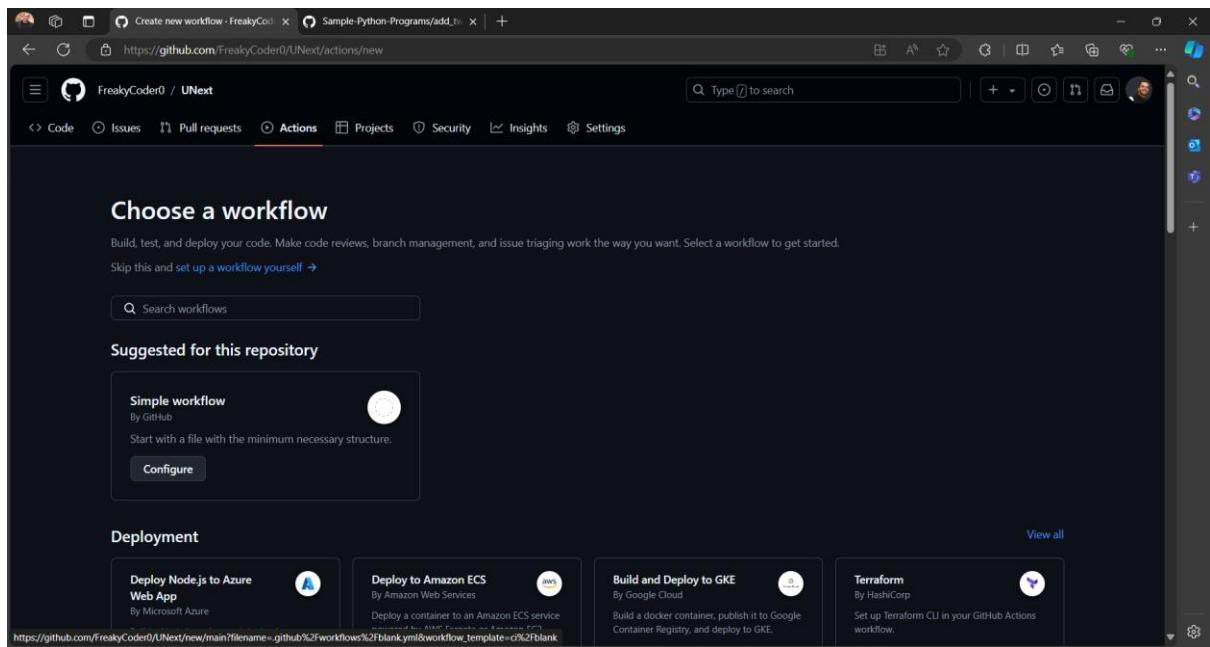
1. Open github



2. Created a repo name “UNext”



3. Now go to actions tab.



4. Choose new workflow

```

name: Sample Python Programs Workflow
on:
  push:
    branches:
      - main
  pull_request:
    branches:
      - main
  workflow_dispatch:
jobs:
  checkout_and_run:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout Repository
        uses: actions/checkout@v3
      - name: Download add_two_numbers.py
        run: |
          curl -O https://raw.githubusercontent.com/jissdeodates/Sample-Python-Programs/master/add_two_numbers.py

```

Control + Shift + M to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.
Control + Space to trigger autocomplete in most situations.

5. Now write the code for cloning and performing the action in the code of [add_two_numbers.py](#) “[Sample-Python-Programs/add_two_numbers.py at master · jissdeodates/Sample-Python-Programs \(github.com\)](#)”

```

# This program adds two numbers
num1 = 1.5
num2 = 6.3

# Add two numbers
sum = float(num1) + float(num2)

# The below code prints Hello, world!
print('Hello, world!')

# Display the sum
print('The sum of {} and {} is {}'.format(num1, num2, sum))

```

6. This is the file that I have cloned.

The screenshot shows a GitHub repository named 'FreakyCoder0/UNext' with a pull request. The code editor displays a workflow file named 'blank.yml' in the 'main' branch. The file contains the following YAML code:

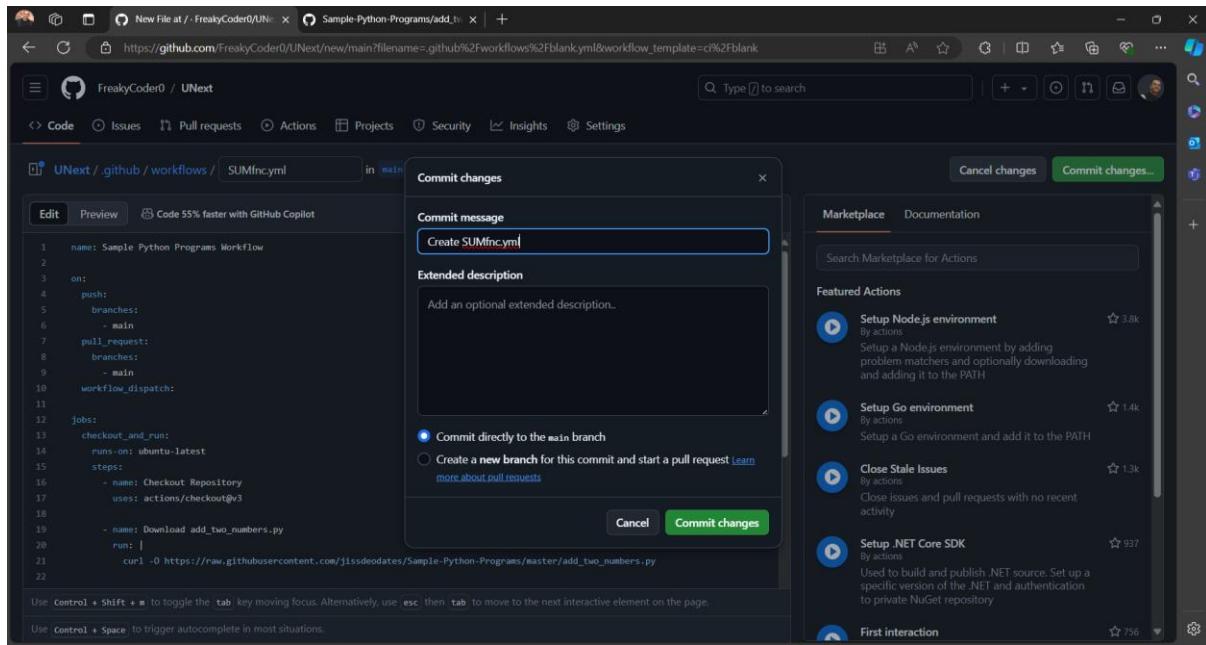
```
23 - name: Run add_two_numbers.py
24   run: |
25     python3 add_two_numbers.py
26
27 - name: Update add_two_numbers.py (Example Modification)
28   run: |
29     echo "# Updated by GitHub Actions" >> add_two_numbers.py
30     git config --global user.name "GitHub Actions"
31     git config --global user.email "actions@github.com"
32     git add add_two_numbers.py
33     git commit -m "Update add_two_numbers.py with new comment"
34     git push origin main
35
36 pull_results:
37   runs-on: ubuntu-latest
38   steps:
39     - name: Checkout Repository
40       uses: actions/checkout@v3
41
42     - name: Pull Results
43       run: |
44         git fetch origin main
45         git checkout main
46         git pull origin main
47         cat add_two_numbers.py
48
49 manual_dispatch:
50   runs-on: ubuntu-latest
51   steps:
52     - name: Manual Dispatch
53       run: echo "This job was manually triggered"
54
```

The GitHub Marketplace sidebar is open, showing featured actions such as 'Setup Node.js environment', 'Setup Go environment', 'Close Stale Issues', and 'Setup .NET Core SDK'. A tooltip indicates that a screenshot has been copied to the clipboard.

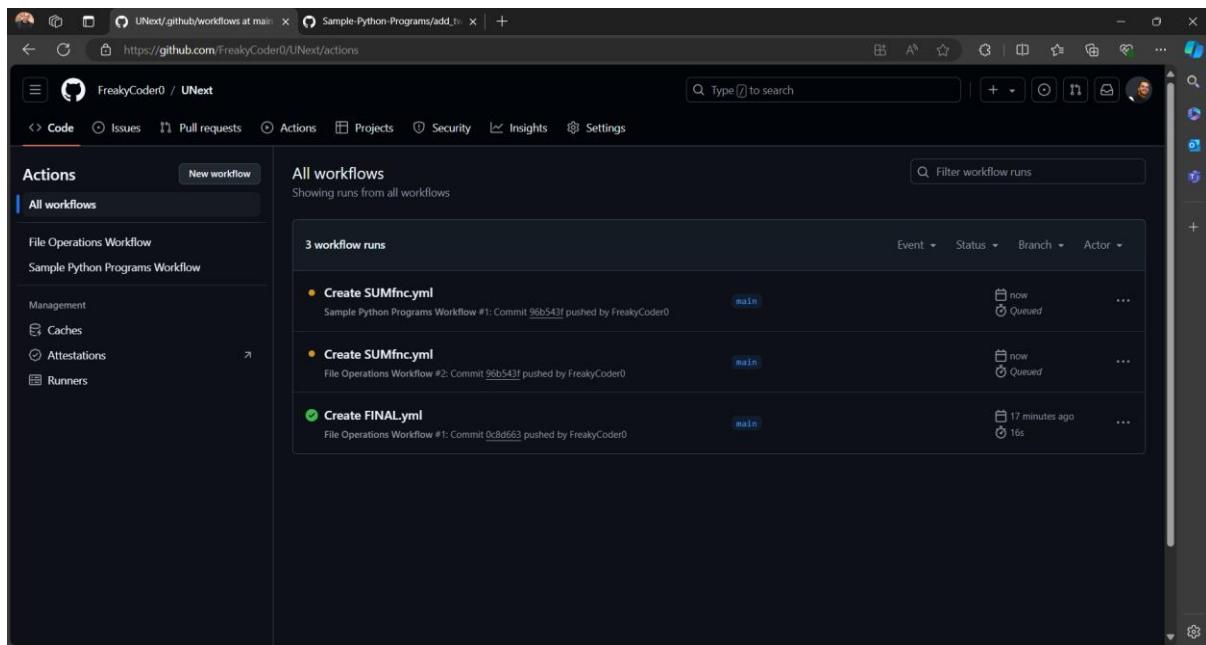
7. This is the continuity of the code.

The screenshot shows the same GitHub repository and pull request as the previous one. The code editor displays the same modified workflow file. The GitHub Marketplace sidebar is open, showing the same featured actions. A tooltip indicates that a screenshot has been copied to the clipboard.

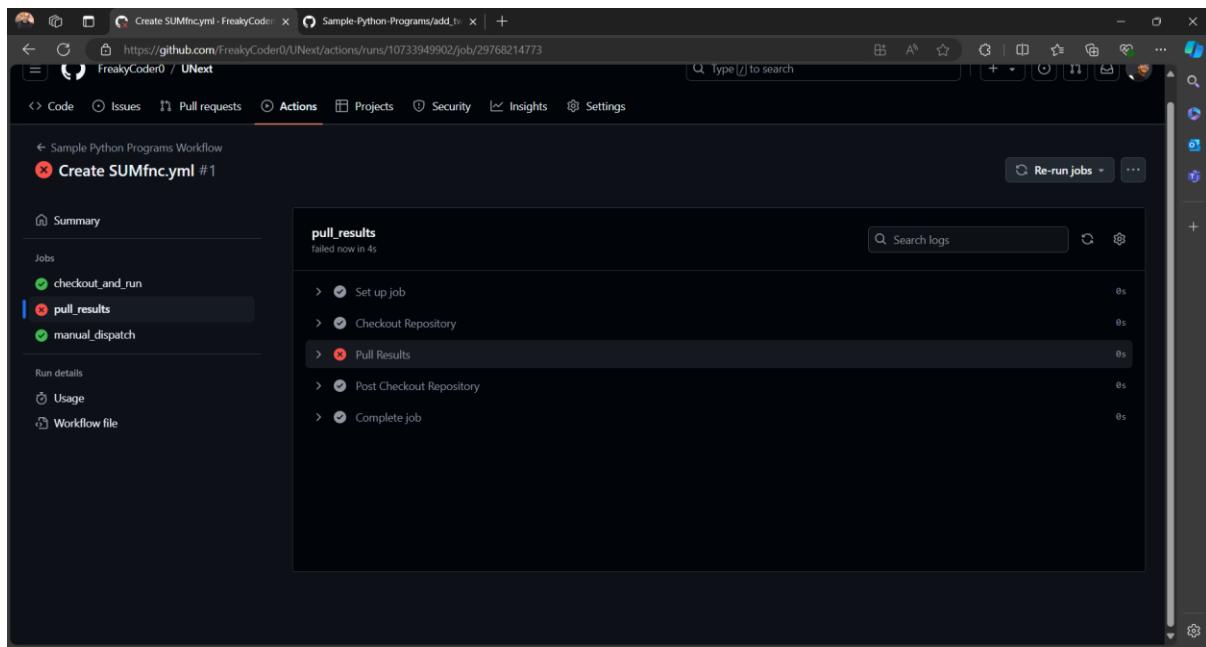
8. Continued.



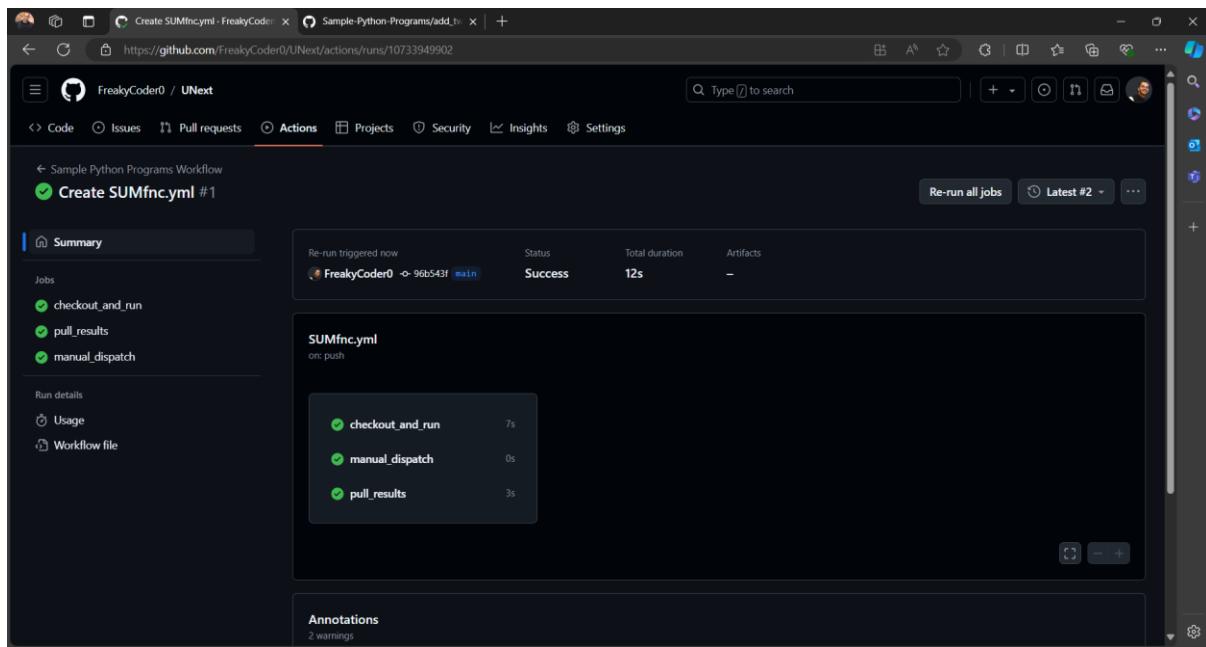
9. Commit changes.



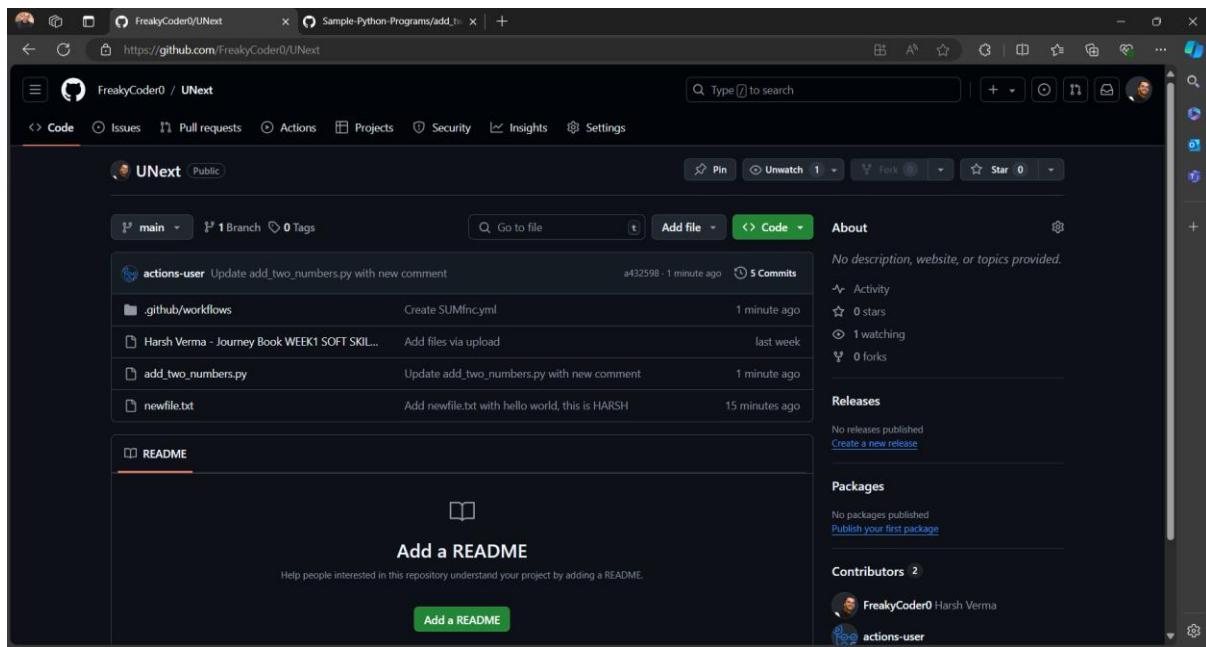
10. Waiting for committing the changes .



11. The pull request failed first.



12. Then performed the operation of action again to perform the addition.



13. Finally the action is performed successfully.