

INDEX

S.No.	Date	Title
1.	23/1/25	Azure Devops Environment Setup.
2.	30/1/25	Azure Devops Project Setup and User Story Management.
3.	06/2/25	Setting Up Epics, Features, And User Stories for Project Planning.
4.	13/2/25	Sprint Planning.
5.	20/2/25	Poker Estimation.
6.	27/2/25	Designing Class Diagram and Sequence Diagram.
7.	06/3/25	Designing Use Case Diagram and Activity Diagram.
8.	20/3/25	Testing – Test Plans and Test Cases.
9.	27/3/25	Load Testing and Pipelines.
10.	03/4/25	GitHub: Project Structure & Naming Conventions.

EXP NO: 1

AZURE DEVOPS ENVIRONMENT SETUP

Date :

Aim:

To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

INSTALLATION

1. Open your web browser and go to the Azure website: <https://azure.microsoft.com/en-us/get-started/azure-portal>.

Sign in using your Microsoft account credentials.

If you don't have a Microsoft account, you can create one here: <https://signup.live.com/?lic=1>

The screenshot shows the Microsoft Azure portal homepage. At the top, there's a navigation bar with links for Microsoft, Azure, Update, Products, Solutions, Pricing, Partners, Resources, Learn, Support, Contact Sales, and a 'Sign in' button. Below the navigation bar, the title 'Microsoft Azure portal' is displayed, followed by the tagline 'Build, manage, and monitor everything from simple web apps to complex cloud applications in a single, unified console.' There are two buttons: 'Sign in' and 'New to Azure? Get started >'. A dark blue banner at the bottom of the main content area contains the text 'Check out the how-to video series for tips on deploying your cloud workloads from the Azure portal. >'.

Azure mobile app
Stay connected to your Azure resources—anytime, anywhere. Now available for iOS and Android.

The mobile app interface is shown on two devices. The left device (iOS) displays a dashboard with sections for Analytics, Metrics, and Metrics. The right device (Android) shows a similar dashboard with sections for Analytics, Metrics, and Metrics.

2. Azure home page

Welcome to Azure!

Don't have a subscription? Check out the following options.

Start with an Azure free trial
Get \$200 free credit toward Azure products and services, plus 12 months of popular free services.

Manage Microsoft Entra ID
Manage access, set smart policies, and enhance security with Microsoft Entra ID.

Azure for Students
Get free software, Azure credit, or access Azure Dev Tools for teaching after you verify your academic status.

Azure services

Resources

Search bar: azure dev

3. Open a DevOps environment in the Azure platform by typing **Azure DevOps Organizations** in the search bar.

Search results for "azure dev":

- All Services (99+)
- Resources
- More (4)

Services

- Azure Device Registry
- Azure DevOps organizations**
- Azure Database for MySQL servers
- Education

Marketplace

- Build Agents for Azure DevOps
- Azure DevOps Auditing
- Azure DevOps Backup Tool
- Self Hosted Runner for Azure DevOps

Documentation

- Install the Azure Developer CLI
- What is Azure DevTest Labs?
- Set up Azure Deployment Environments - Azure Deployment Environments

Continue searching in Microsoft Entra ID

Get started

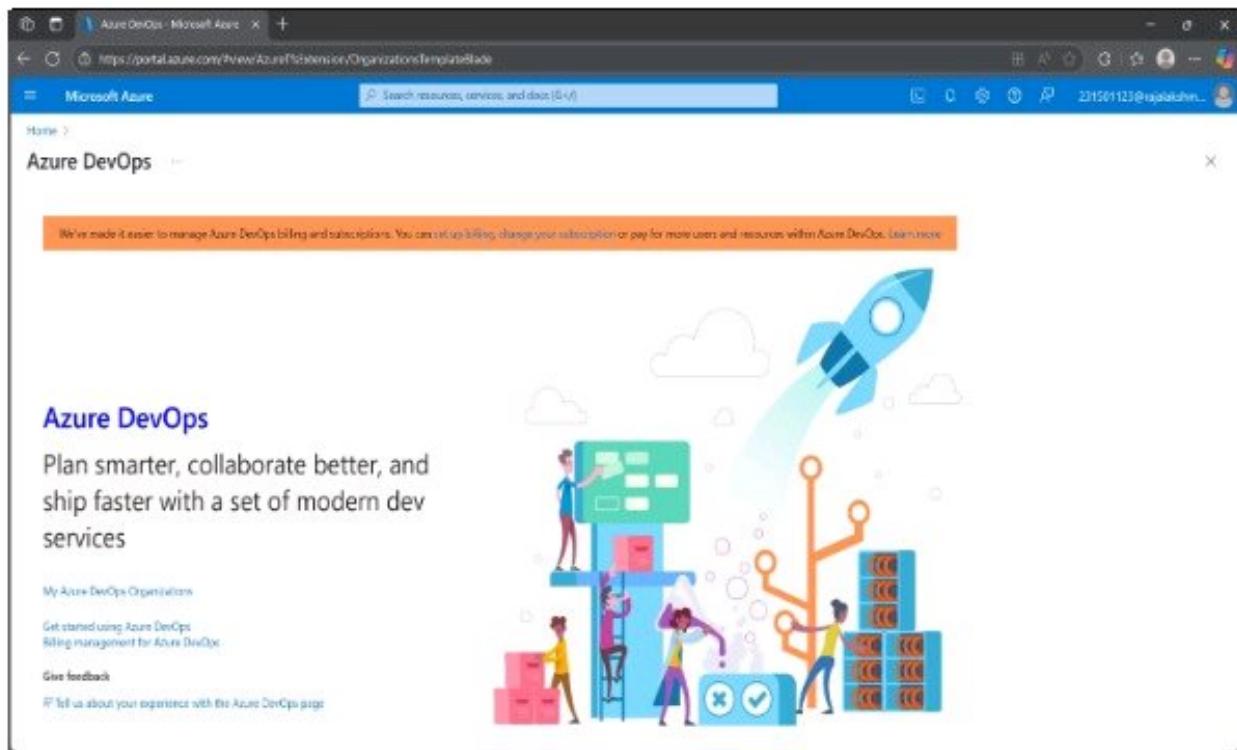
Dashboard

Cost Management

Azure mobile app

Download on the App Store Get it on Google Play

4. Click on the ***My Azure DevOps Organization*** link and create an organization and you should be taken to the Azure DevOps Organization Home page.



Result:

Successfully accessed the Azure DevOps environment and created a new organization through the Azure portal.

EXP NO: 2

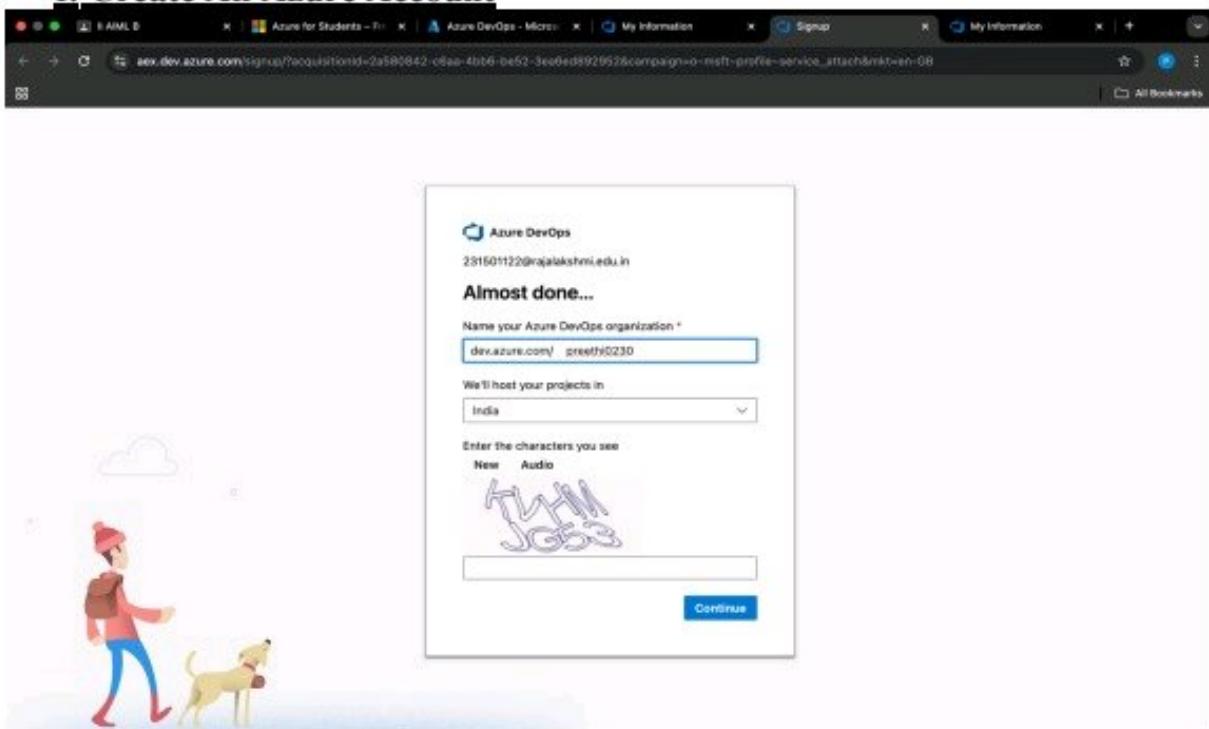
AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

Date :

Aim:

To set up an Azure DevOps project for efficient collaboration and agile work management.

1. Create An Azure Account



2. Create the First Project in Your Organization

- After the organization is set up, you'll need to create your first **project**. This is where you'll begin to manage code, pipelines, work items, and more.
- On the organization's **Home page**, click on the **New Project** button.
- Enter the project name, description, and visibility options:

Name: Choose a name for the project (e.g., **LMS**).

Description: Optionally, add a description to provide more context about the project.

Visibility: Choose whether you want the project to be **Private** (accessible only to those invited) or **Public** (accessible to anyone).

- Once you've filled out the details, click **Create** to set up your first project.

Create new project

X

Project name *

Batch Data Analysis and Visualizations

Description

Visibility



Public

Anyone on the internet can view the project. Certain features like TFVC are not supported.



Private

Only people you give access to will be able to view this project.



Public projects are disabled for your organization. You can turn on public visibility with [organization policies](#).

Advanced

Version control

Git

Work item process

Agile

Cancel

Create

- Once logged in, ensure you are in the correct organization. If you're part of multiple organizations, you can switch between them from the top left corner (next to your user profile). Click on the Organization name, and you should be taken to the Azure DevOps Organization Home page.

The screenshot shows the Azure DevOps Organizations dashboard. On the left, there's a user profile card for 'Princess Darlene Karonah' with a green circular icon containing 'PK'. Below the card are contact details: email (231501123@rajalakshmi.edu.in), location (India), and phone number (98 231501123@rajalakshmi.edu.in). A section for 'Visual Studio Dev Essentials' is present. To the right, the 'Azure DevOps Organizations' interface lists several organizations: dev.azure.com/231501123 (Owner) with projects ATM and Attendance management; dev.azure.com/2315011230074 (Owner); dev.azure.com/2315011230439 (Owner); and dev.azure.com/231501122 (Member). There's a 'Create new organization' button at the top right.

4. Project dashboard

The screenshot shows the Azure DevOps E-commerce project dashboard. The left sidebar includes links for Overview, Summary, Dashboards, WUs, Boards, Repos, Pipelines, Test Plans, and Artifacts, with 'Summary' currently selected. The main area features a large purple header with the project name 'E-commerce'. Below it, there are two main sections: 'About this project' and 'Project stats'. The 'About this project' section contains a detailed description of the E-commerce Product Uploader application, mentioning its web-based nature, bulk product upload support, and validation rules. It also highlights the robust backend and frontend components, full CRUD operations, product search, and paginated views. The 'Project stats' section displays various metrics over the last 7 days: 42 work items created, 0 work items completed, 0 pull requests opened, 11 commits by 11 authors, and a 'Members' section showing a grid of team member icons.

5. To manage user stories:

- a. From the **left-hand navigation menu**, click on **Boards**. This will take you to the main **Boards** page, where you can manage work items, backlogs, and sprints.

b. On the **work items** page, you'll see the option to **Add a work item** at the top. Alternatively, you can find a + button or **Add New Work Item** depending on the view you're in. From the **Add a work item** dropdown, select **User Story**. This will open a form to enter details for the new User Story.

The screenshot shows the Azure DevOps interface for the Ecommerce project. The left sidebar has 'Work items' selected. The main area is titled 'Work items' and shows a list of user stories. The first user story, 'Try Upload Without Entering Product Name or Price', is highlighted. The columns in the table are ID, Title, Assigned To, State, Area Path, and Tags. Other visible user stories include 'login failed', 'As a store owner, I want to upload a CSV/Excel file so that I can add it', 'As a developer, I want to validate the file structure to ensure all require', 'As a user i want to upload multiple csv files at once so I can analyze th', 'file upload and selection', 'data visualisation', 'test case (2)', and 'upload and selection'.

ID	Title	Assigned To	State	Area Path	Tags
64	Try Upload Without Entering Product Name or Price	Preethi Gopinath	Design	Ecommerce	
90	login failed	Unassigned	New	Ecommerce	
89	login failed	Unassigned	New	Ecommerce	
23	As a store owner, I want to upload a CSV/Excel file so that I can add it	Preethi Gopinath	Resolved	Ecommerce	
78	As a developer, I want to validate the file structure to ensure all require	Preethi Gopinath	New	Ecommerce	
26	As a developer, I want to validate the file structure to ensure all requir	Preethi Gopinath	New	Ecommerce	
88	As a user i want to upload multiple csv files at once so I can analyze th	Preethi Gopinath	New	Ecommerce	
87	file upload and selection	Unassigned	New	Ecommerce	
86	data visualisation	Unassigned	New	Ecommerce	
85	test case (2)	Prathisha R	Design	Ecommerce	
82	upload and selection	Unassigned	New	Ecommerce	

Result: Successfully created an Azure DevOps project with user story management and agile workflow setup.

EXP NO: 3

Date :

SETTING UP EPICS, FEATURES, AND USER STORIES FOR PROJECT PLANNING

Aim:

To create epics, user stories, and tasks for the project, Ecommerce Product Uploader.

1.Create Epic, Features, User Stories, Task

The screenshot shows the Azure DevOps Backlog board for the 'Ecommerce' project. The backlog is organized into three sprints:

- Sprint 1 (5/20/2025 - 5/26/2025):** Contains one Epic (data-visualization) which includes two User Stories: "As a user I want to upload multiple files at once so..." and "As a store owner, I want to search products by name so...".
- Sprint 2 (5/27/2025 - 6/2/2025):** Contains one Epic (product-management) which includes two User Stories: "As a user, I want to filter products by price or description so..." and "As a store owner, I want to manually add or edit a product so...".
- Sprint 3 (6/3/2025 - 6/9/2025):** Contains one Epic (Search-and-Pagination) which includes two User Stories: "As a user I want to implement backend API to search by name or key..." and "As a store owner, I want to update query logic in backend accordingly".

The backlog table has columns for Order, Work Item Type, Title, State, Effort, Business Area, and Tags. A sidebar on the right shows the planning view with sprints and their details.

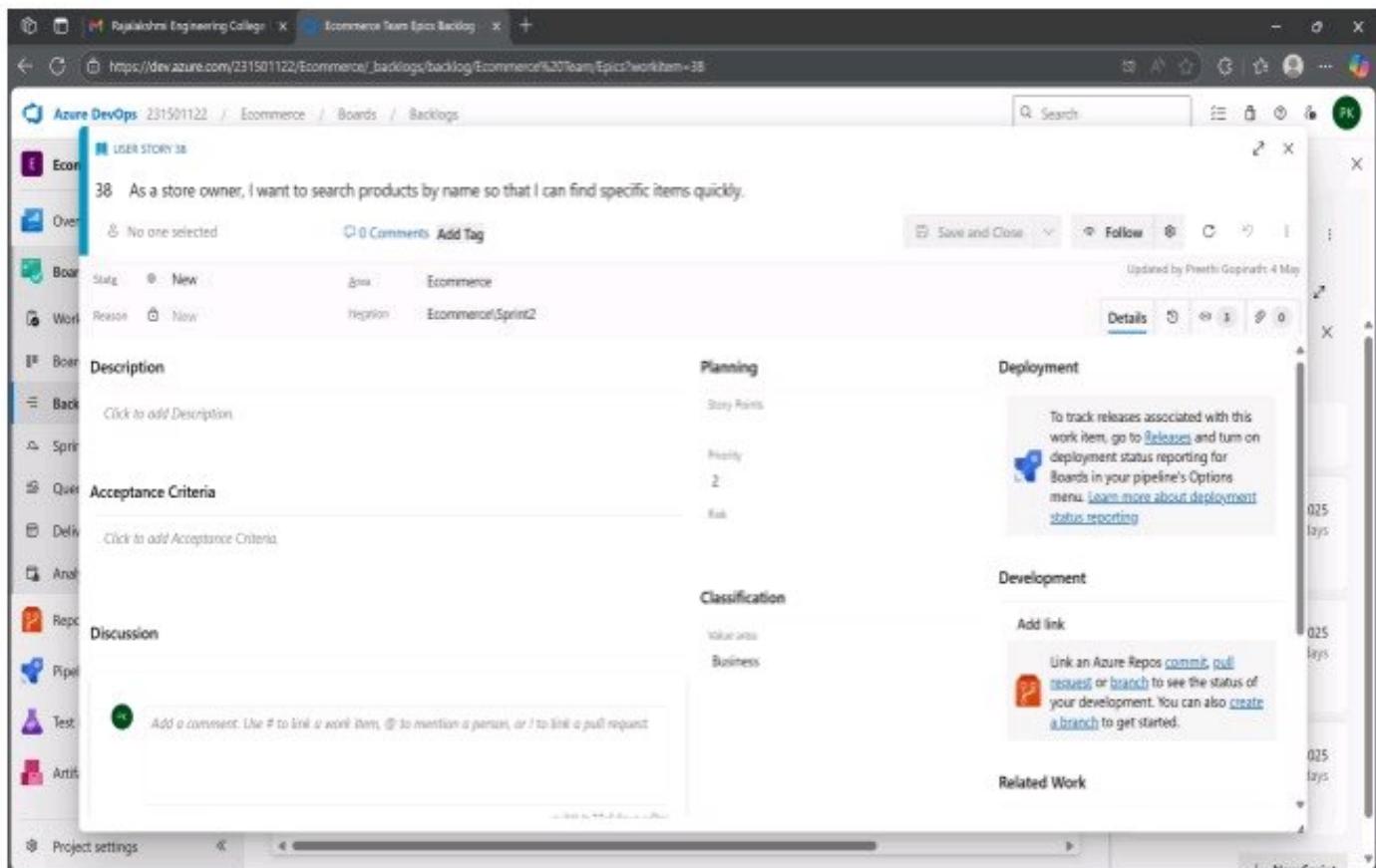
2. Fill in Epics

The screenshot shows the Azure DevOps Backlog page for the Ecommerce team. The backlog is displayed in a grid format with columns for Order, Work item type, Title, State, Prior., Business Area, and Tags. The backlog items are categorized into epics, user stories, and tasks. A 'Planning' sidebar on the right shows three sprints: Sprint 1 (5/26/2025 - 5/30/2025), Sprint 2 (5/27/2025 - 6/2/2025), and Sprint 3 (6/5/2025 - 6/9/2025).

3. Fill in Features

The screenshot shows the Azure DevOps Work Item details page for a feature work item titled "Search Functionality". The work item has a priority of 2 and a risk of 1. The deployment status is set to "To track releases associated with this work item, go to Releases and turn on deployment status reporting for Boards in your pipeline's Options menu." The development section includes a link to an Azure Repos repository and a related work section.

4. Fill in User Stories



The screenshot shows the Azure DevOps interface for a User Story titled "USER STORY 38". The story description is: "38 As a store owner, I want to search products by name so that I can find specific items quickly." It has 0 comments and 0 tags. The story is set to "New" status, assigned to "Ecommerce", and is part of the "Ecommerce/Sprint2" iteration. The "Planning" section shows Story Points as 2 and Priority as 2. The "Deployment" section indicates the story is associated with a release. The "Development" section shows a link to an Azure Repos pull request. The "Classification" section categorizes the story under "Business". A sidebar on the left lists various project management sections like Backlog, Sprint, Queries, Acceptance Criteria, Definitions, Analytics, Reports, Pipelines, Test, and Artifacts. A bottom bar includes Project settings and navigation controls.

Result: Thus, epics, features, user stories, and tasks have been created successfully.

EXP NO: 4

Date :

SPRINT PLANNING

Aim:

To assign a user story to a specific sprint for the project, Ecommerce Product Uploader.

SPRINT PLANNING

Sprint 1

The screenshot shows the Azure DevOps Taskboard for the 'Ecommerce' project under the 'Ecommerce Team' sprint. The taskboard displays two user stories assigned to the sprint:

User Story ID	Description	Status	Assigned To
79	As a user, I want to filter products by price or description keywords.	New	Rino Calvin
78	As a store owner, I want to manually add or edit a product in case I miss it during bulk upload.	New	Praethi Gopinath

The taskboard interface includes a sidebar with navigation links like Overview, Boards, Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The top right features a search bar and column options. The bottom right shows the sprint duration: 3 June - 9 June, 5 work days.

Sprint 2

The screenshot shows the Azure DevOps interface for the Ecommerce project. The left sidebar is collapsed, showing the 'Sprints' section is selected. The main area displays the Taskboard for the 'Ecommerce Team' in 'Sprint 2'. The backlog is organized into four columns: New, Active, Resolved, and Closed. There are five work items listed:

- WI 23: As a store owner, I want to upload a CSV/Excel file so that I can add multiple products at once. Status: New, Assigned to Preethi Gopinath.
- WI 24: Implement file upload UI and backend endpoint. Status: New, Unassigned.
- WI 25: Parse CSV/Excel file and extract product fields. Status: New, Unassigned.
- WI 26: As a developer, I want to validate the file structure to ensure all required columns are present. Status: New, Assigned to Preethi Gopinath.
- WI 27: Check headers for required fields (name, description, price, image). Status: New, Unassigned.
- WI 28: Return user-friendly error messages on. Status: New, Unassigned.

Sprint 3

The screenshot shows the Azure DevOps interface for the Ecommerce project. The left sidebar is collapsed, showing the 'Sprints' section is selected. The main area displays the Taskboard for the 'Ecommerce Team' in 'Sprint 3'. The backlog is organized into four columns: New, Active, Resolved, and Closed. There are three work items listed:

- WI 60: As a developer, I want to validate the file structure to ensure all required columns are present. Status: Resolved, Assigned to Rina Calvin.
- WI 77: As a store owner, I want to search products by name so that I can find specific items quickly. Status: New, Assigned to Rina Calvin.
- WI 69: As a user I want to upload multiple csv files at once so I can analyze them together. Status: New, Assigned to Preethi Gopinath.

Result: The Sprints are created for the project, Ecommerce Product Uploader.

EXP NO: 5

POKER ESTIMATION

Date :

Aim:

Create Poker Estimation for the user stories for the project, Ecommerce Product Uploader.

Poker Estimation

The screenshot shows the Azure DevOps interface for a work item titled "User Story 23 As a store owner, I want to upload a CSV/Excel file so that I can add multiple products at once." The work item is categorized under "E-commerce" and has a status of "Resolved". It was last updated by Preethi Gopinath. The "Planning" section indicates a priority of 2 and a story point value of 1. The "Deployment" section provides instructions on tracking releases. The "Development" section includes a link to an Azure Repos repository. The "Acceptance Criteria" and "Discussion" sections are currently empty.

Result:

The Estimation/Story Points is created for the project using Poker Estimation.

EXP NO: 6

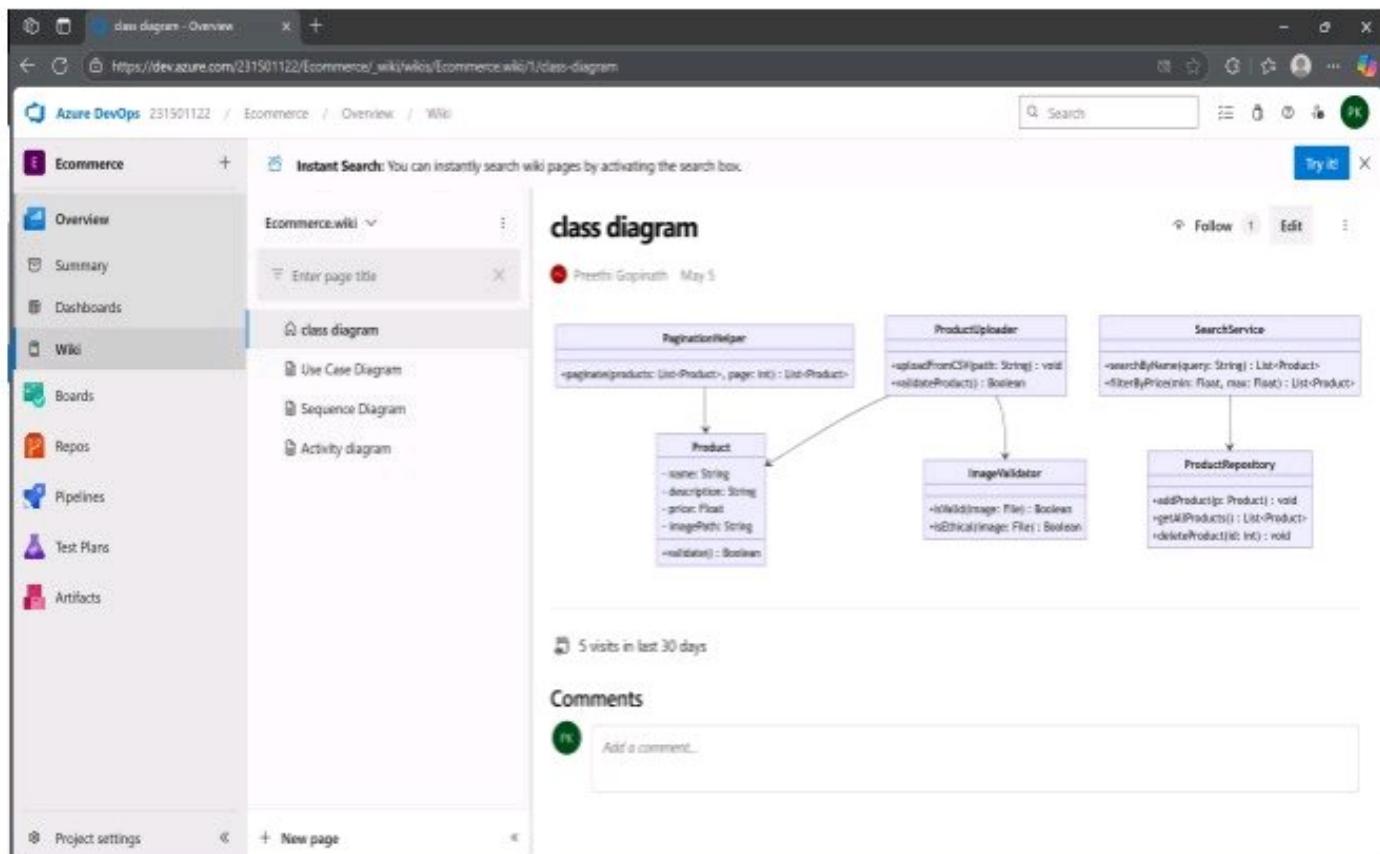
DESIGNING CLASS DIAGRAM AND SEQUENCE DIAGRAM

Date :

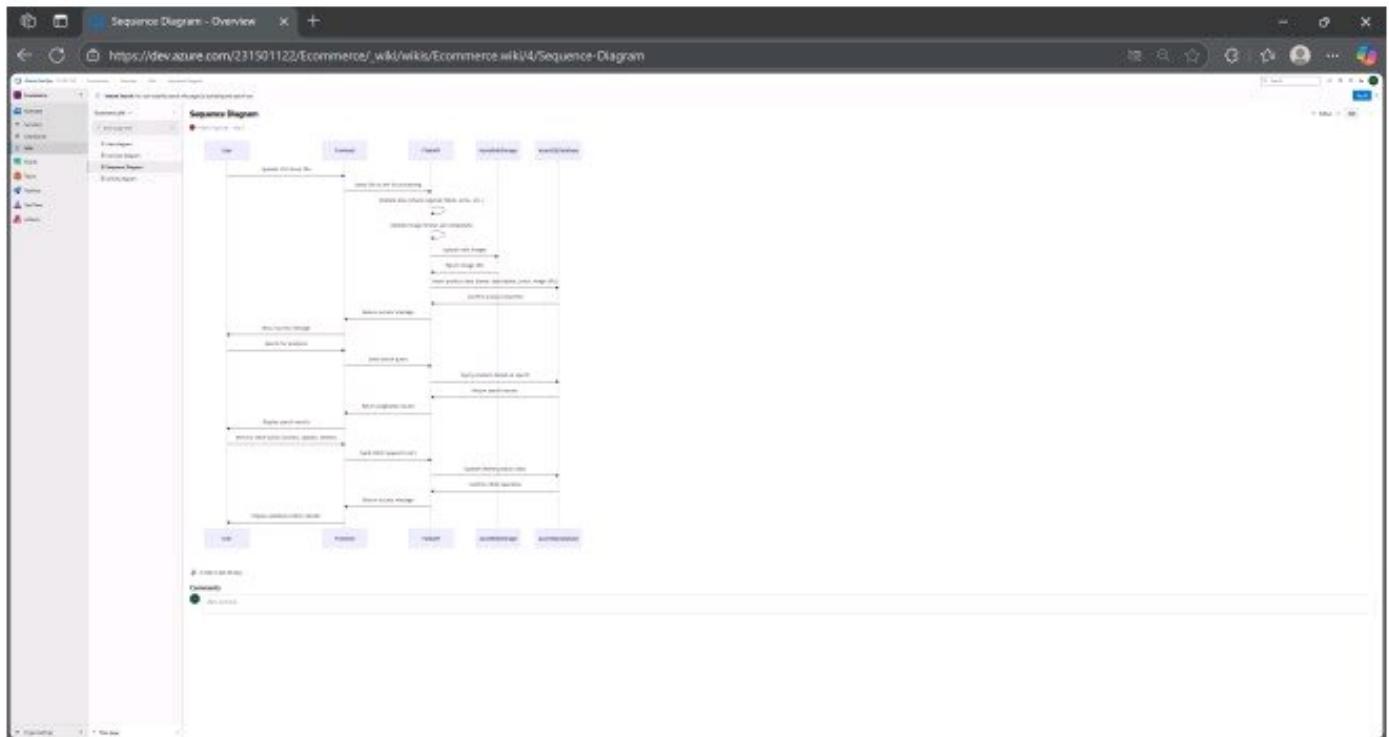
Aim:

To design a Class Diagram and Sequence Diagram for the project, Ecommerce Product Uploader.

6A. Class Diagram



6B. Sequence Diagram



Result: The Class and Sequence Diagrams are designed successfully for the project, Ecommerce Product Uploader.

EXP NO: 7

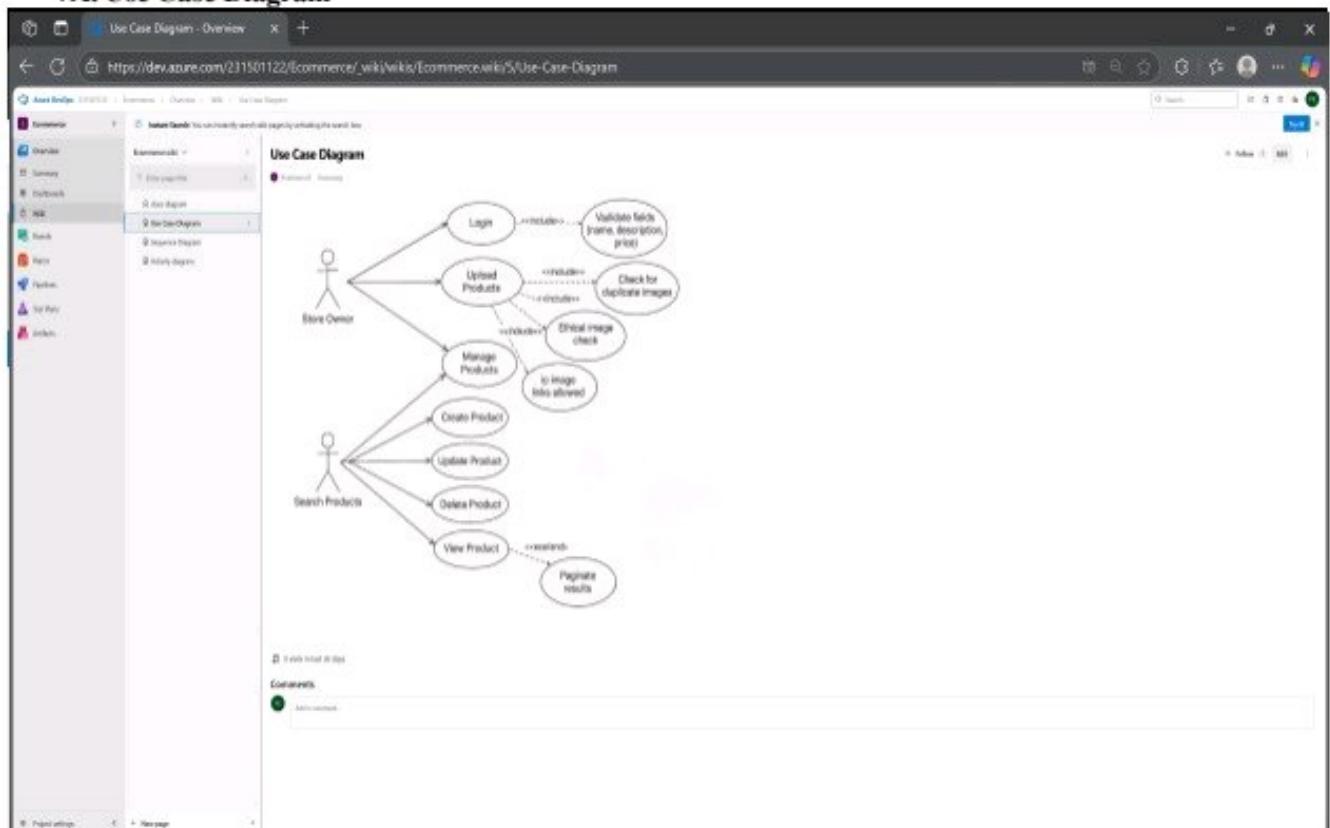
Date :

DESIGNING USE CASE DIAGRAM AND ACTIVITY DIAGRAM

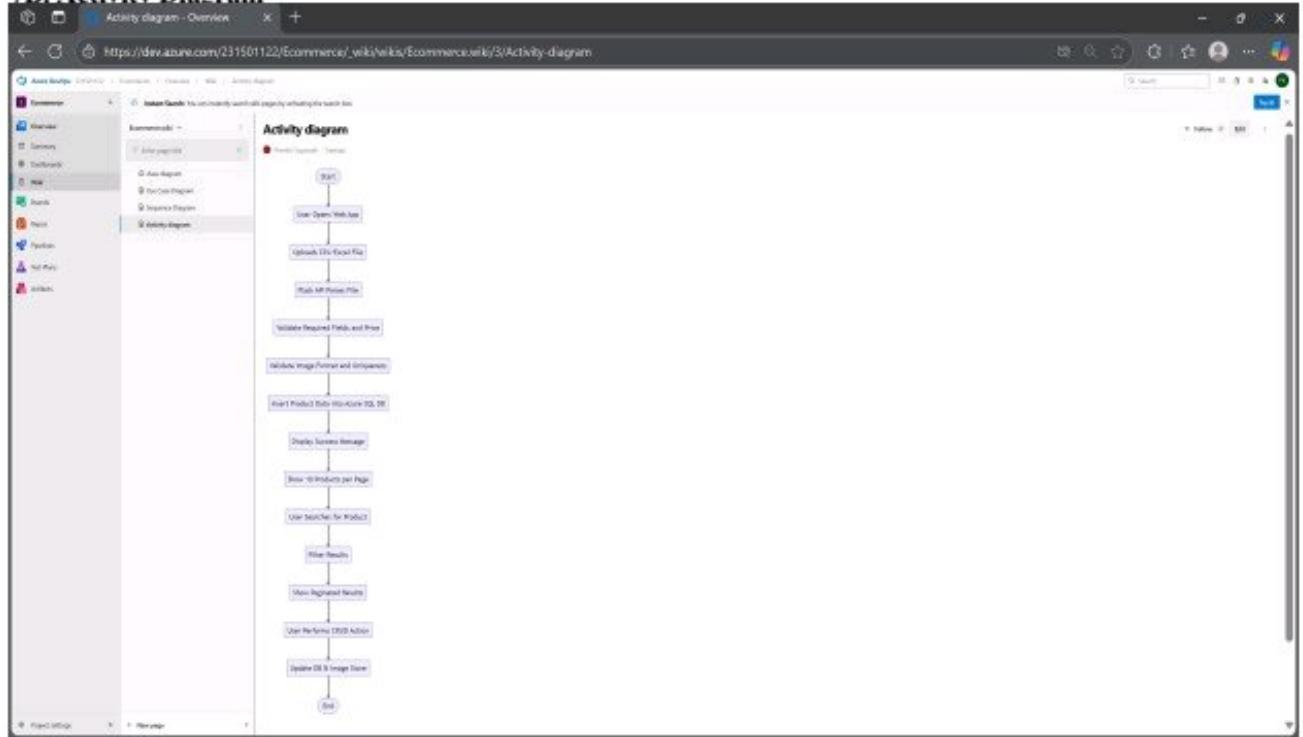
Aim:

To design a Use Case Diagram and an Activity Diagram for the project, Ecommerce Product Uploader.

7A. Use Case Diagram



7B. Activity Diagram



Result: The Use Case and Activity Diagrams are designed successfully for the project, Ecommerce Product Uploader.

EXP NO: 8	TESTING – TEST PLANS AND TEST CASES
Date :	

Aim:

Test Plans and Test Case and write two test cases for at least five user stories showcasing the happy path and error scenarios in azure DevOps platform.

Test Planning and Test Case

Test Case Design Procedure

1. Understand Core Features of the Application

- 1. User Authentication
- 2. Uploading and Managing Batch Data Files
- 3. Running Batch Analysis Jobs
- 4. Viewing Interactive Visualizations and Charts
- 5. Exporting Analysis Results

2. Define User Interactions

- Simulate real scenarios (e.g., upload dataset, trigger job, download result).

3. Design Happy Path Test Cases

- Validate all main functions work properly (e.g., successful login, upload, and visualization).

4. Design Error Path Test Cases

- Simulate unexpected or invalid user behavior (e.g., upload fails, unsupported file, job timeout).

5. Break Down Steps and Expected Results

- Each test case includes step-by-step actions and expected outcomes.

6. Use Clear Naming and IDs

- Example: TC01 – Successful File Upload, TC08 – Visualization Fails.

7. Separate Test Suites

- Suites grouped by modules (Login, File Upload, Job Execution, Visualization, Export).

8. Prioritize and Review

- Critical test cases marked as High Priority.

- Mapped to user stories in Azure DevOps.

1. New test plan

New Test Plan

Name *: performance tests

Area Path *: Ecommerce

Iteration *: Ecommerce

Create Cancel

2. Test suite

Test Plan 05 Performance Tests

Bulk Upload Performance (ID: 67)

Test Point	Outcome	Order	Test Case Id	Configuration	Tester
Upload 500 Products in CSV and Measure Time	Active	1	68	Windows 10	Preeti G...

3. Test case

Give two test cases for at least five user stories showcasing the happy path and error scenarios in azure DevOps platform.

USER STORIES

- As a user, I want to log in using my username and password so that I can access my account.
- As a user, I should not be able to submit the login form with empty fields so that I can provide the required data.
- As a user, I want to log out when I click the logout button so that I can end my session securely.
- As a user, I want to be redirected to the login page after logging out so that I know my session has ended and I can log in again if needed.
- As a user, I want to be able to upload multiple CSV files at once, so I can analyze them together.

Test Suites

Test Suite: TS01 - User Authentication (ID: 54)

1. TC01 – Successful Login

- **Action:**
 - Navigate to the login page
 - Enter valid credentials
 - Click "Login"
- **Expected Results:**
 - User redirected to dashboard.
- **Type:** Happy Path

2. TC02 – Prevent Login with Empty Fields

- **Action:**
 - Navigate to the login page.
 - Leave username and/or password fields empty.
 - Click on "Login".
- **Expected Results:**
 - Validation error message is shown prompting user to fill required fields.
- **Type:** Error Path
-

Test Suite: TS02 - Logout Functionality (ID: 47)

1. TC03 – Successful Logout and Redirect

- **Action:**
 - Log in successfully.
 - Click the "Logout" button.
- **Expected Results:**

- User session ends.
- User is redirected to the login page.
- **Type:** Happy Path

2. TC04– Access Protected Page After Logout

- **Action:**
 - Logout.
 - Attempt to navigate back to a protected page (e.g., dashboard) via browser back button or URL.
- **Expected Results:**
 - User is redirected to the login page and denied access.
- **Type:** Error Path

Test Suite: TS03 - CSV Upload Functionality (ID: 88)

1. TC05 – Upload Multiple Valid CSV Files

- **Action:**
 - Log in successfully
 - Navigate to the CSV upload section
 - Select multiple valid .csv files
 - Click "Upload"
- **Expected Results:**
 - All files are uploaded successfully.
 - Files are listed and ready for analysis.
- **Type:** Happy Path

2. TC06 – Upload Attempt Without Selecting Files

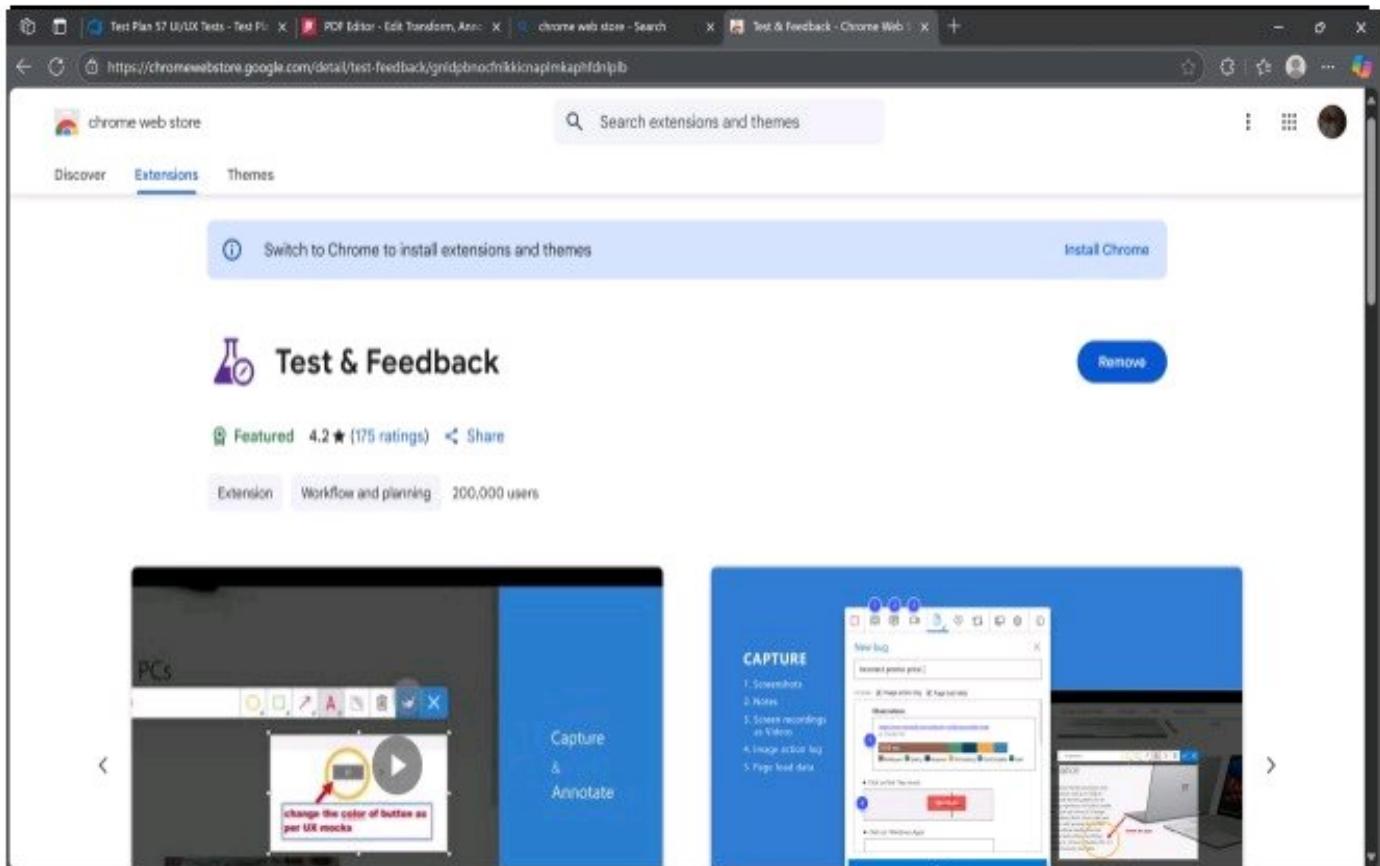
- **Action:**
 - Navigate to the CSV upload section
 - Click "Upload" without selecting any files.
- **Expected Results:**
 - Validation message prompting the user to select at least one file.
- **Type:** Error Path

Test Cases

The screenshot shows the Azure DevOps interface for a project named "Ecommerce". The left sidebar is open, showing options like Overview, Boards, Repos, Pipelines, Test Plans, Test plans, Progress report, Parameters, Configurations, Runs, and Artifacts. Under "Test Plans", "Test plans" is selected. The main area displays a "UI/UX Tests" section for a "Mobile Responsiveness" test plan (ID: 59). The "Execute" tab is active, showing one test point: "Check Product Upload Page on Mobile Devices". The status is "Active", Order is 1, Test Case Id is 60, Configuration is "Windows 10", and Tester is "Preethi G". A "Run for web application" button is also present.

This screenshot shows the same "Ecommerce" project in Azure DevOps. The left sidebar is identical to the first screenshot. The main area displays an "Input Validation" section for a "Missing Required Fields" test plan (ID: 63). The "Execute" tab is active, showing one test case: "Try Upload Without Entering Product Name or Price". The status is "Active", Order is 1, Test Case Id is 64, and Tester is "Preethi G". A "New Test Case" button is visible.

4. Installation of test



Test and feedback

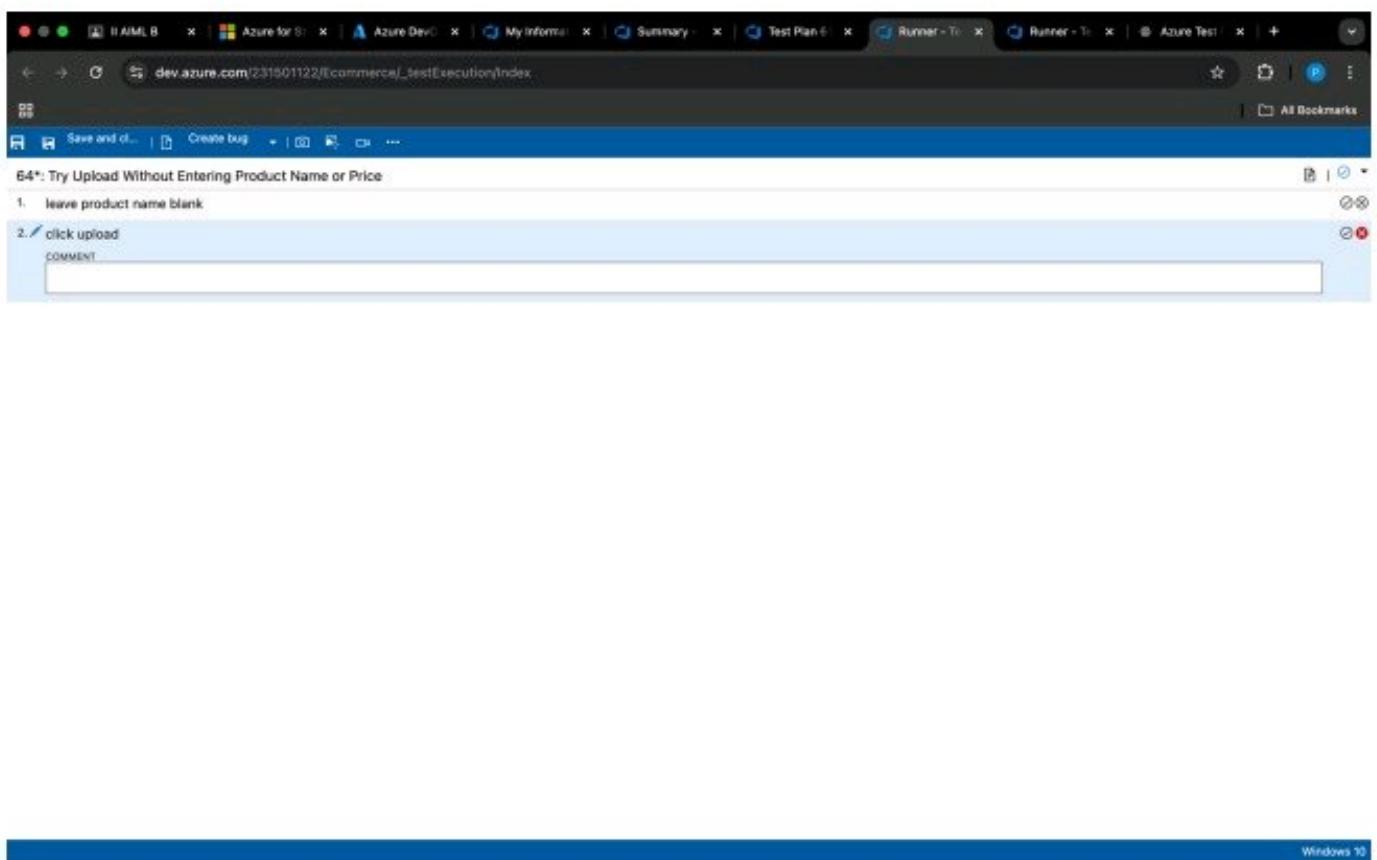
Showing it as an extension

The screenshot shows the Azure DevOps Test Plan interface for an 'Ecommerce' project. On the left, the navigation bar includes 'Overview', 'Boards', 'Repos', 'Pipelines', 'Test Plans' (selected), 'Test plans', 'Progress report', 'Parameters', 'Configurations', 'Runs', and 'Artifacts'. The main area displays a 'Database Sync Check (ID: 7)' test plan with a 'Test Suites' section containing a single suite named 'Database Sync Check (0)'. The 'Execute' tab is selected, showing a table with one test point: 'Check Product Entry in DB After Upload' (Status: Passed). A context menu is open over this test point, with 'Test & Feedback' highlighted. Other options in the menu include 'GoFullPage - Full Page Screen Capt...', 'Manage extensions', and 'Get extensions for Microsoft Edge'.

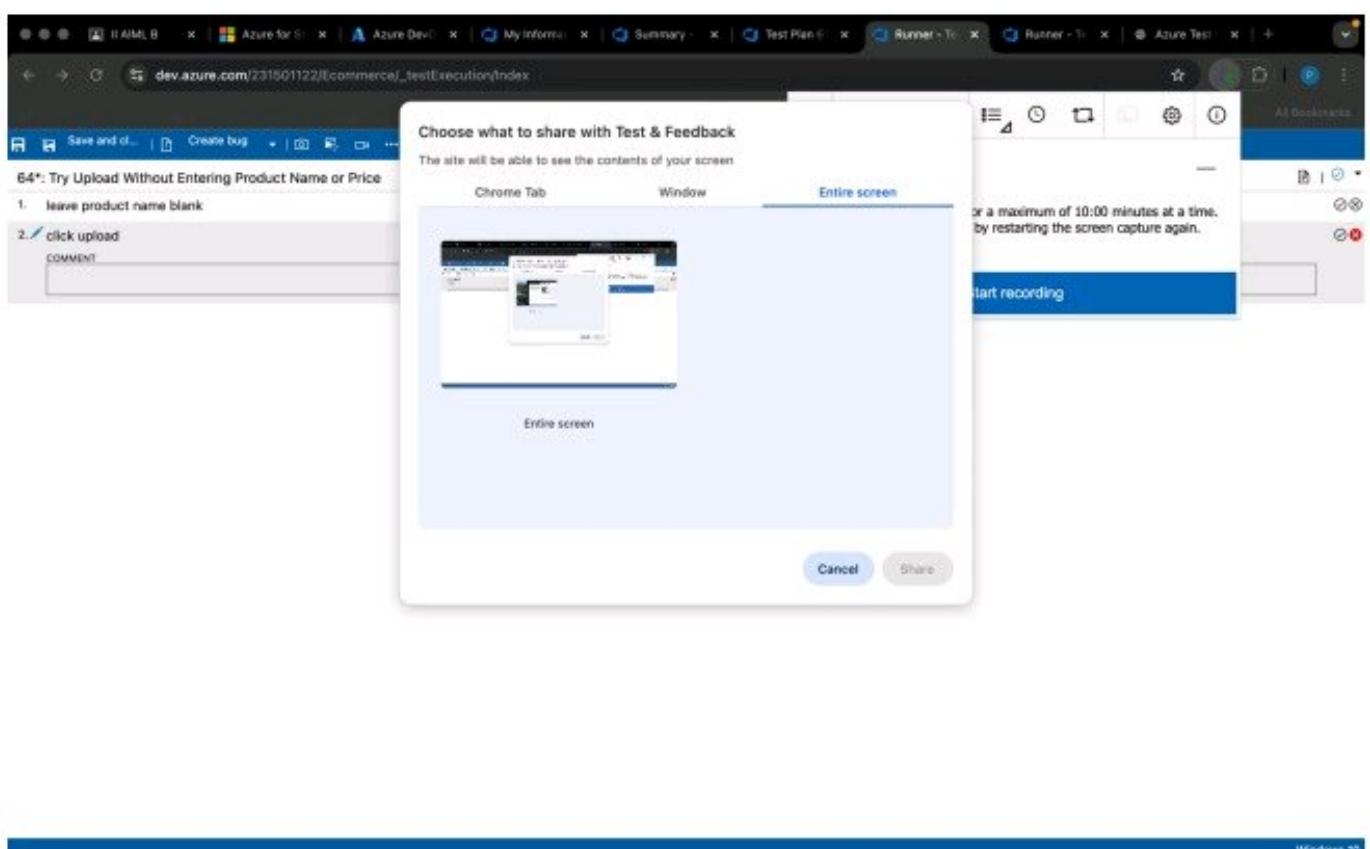
5. Running the test cases

A screenshot of the Azure DevOps Test Plan interface. The left sidebar shows the project navigation with 'Ecommerce' selected. Under 'Test Suites', 'Integration Tests' is expanded, and 'Database Sync Check (1)' is selected. In the main area, the title 'Database Sync Check (ID: 71)' is displayed with tabs for 'Define', 'Execute', and 'Chart'. The 'Execute' tab is active, showing 'Test Points (1 item)'. A single test point titled 'Check Product Entry in DB After Upload' is listed with an outcome of 'Passed'. A context menu is open over this test point, listing options: 'View execution history', 'Mark Outcome', 'Run', 'Reset test to active', 'Edit test case', 'Assign tester', and 'View test result'. At the top of the browser window, the URL is https://dev.azure.com/231501122/Ecommerce/_testPlans/execute?planId=69&suiteId=71.

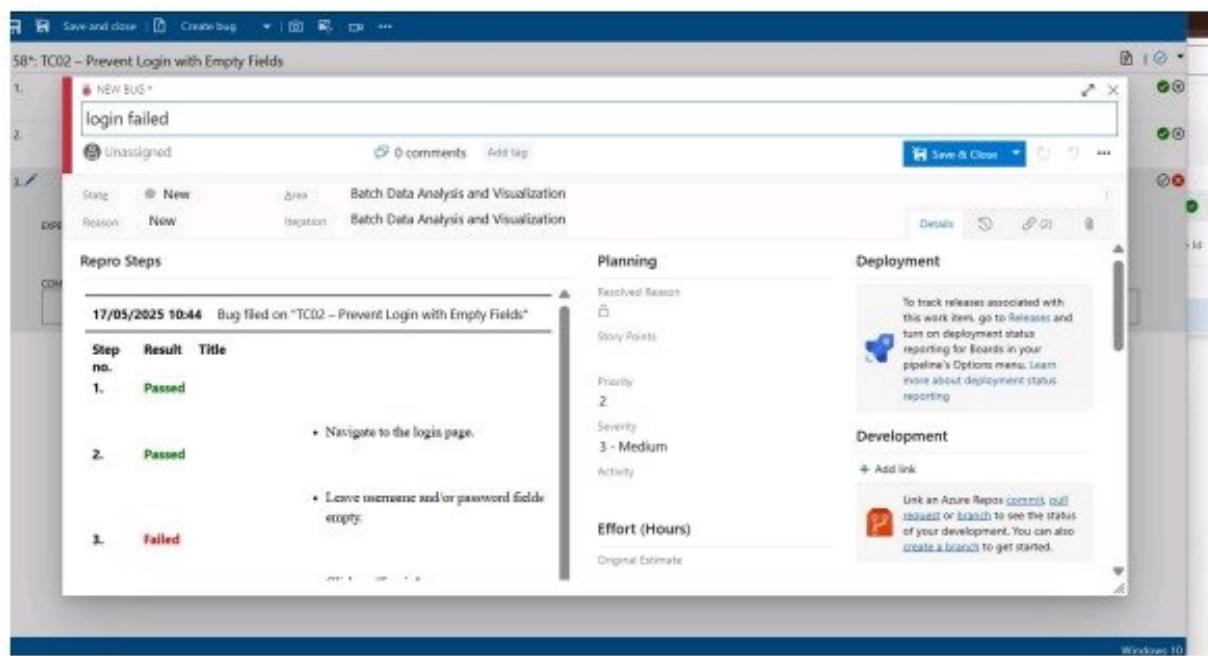
A screenshot of the Azure DevOps Test Plan interface, similar to the one above but with a different view. The left sidebar shows the project navigation with 'Ecommerce' selected. Under 'Test Suites', 'Integration Tests' is expanded, and 'Database Sync Check (1)' is selected. In the main area, the title 'Database Sync Check (ID: 71)' is displayed with tabs for 'Define', 'Execute', and 'Chart'. The 'Execute' tab is active, showing 'Test Points (1 item)'. A single test point titled 'Check Product Entry in DB After Upload' is listed with an outcome of 'Active'. A context menu is open over this test point, listing options: 'Run for web application', 'Run for desktop application', and 'Run with options'. At the top of the browser window, the URL is https://dev.azure.com/231501122/Ecommerce/_testPlans/execute?planId=69&suiteId=71.



6. Recording the test case



7. Creating the bug



dev.azure.com/2116231501122/Ecommerce/_testExecution/Index

64*: Try Upload Without Entering Product Name or Price

1. leave product name blank
2. click upload

NEW BUG *

Title: login failed

Status: Unassigned **Comments:** 0 comments **Add tag:**

Reason: New **Iteration:** Ecommerce/sprint3

Repro Steps:

22/05/2025 04:19 Bug filed on "Try Upload Without Entering Product Name or Price"

Step no.	Result	Title
1.	None	leave product name blank
2.	Failed	click upload

Test Configuration: Windows 10

System Info:

Browser - Name	Google Chrome 136
Browser - Language	en-GB
Browser - Height	776

Planning:

Resolved Reason: None
Story Points: 1
Priority: 2
Severity: 3 - Medium
Activity: None

Deployment:

To track releases associated with this work item, go to Releases and turn on deployment status reporting for Boards in your pipeline's Options menu. Learn more about deployment status reporting.

Development:

+ Add link

Link an Azure Repos commit, pull request or branch to see the status of your development. You can also create a branch to get started.

Effort (Hours):

Original Estimate: None
Remaining: None
Completed: None

Related Work:

+ Add link

Add an existing work item as a parent

Tested By: Srinivas Dharshini (Last seen)

Bug #60

60 not logging in due to system error

No one selected 0 Comments Add Tag

Status: New **Area:** Batch Data Analysis and Visualization
Reason: New **Iteration:** Batch Data Analysis and Visualization

Repro Steps:

17/05/2025 10:51 Bug filed on "TC02 – Prevent Login with Empty Fields"

Step no.	Result	Title
1.	Passed	• Navigate to the login page.
2.	Passed	• Leave username and/or password fields empty.
3.	Failed	• Click on "Login".

Expected Result:

- Validation error message is shown prompting user to fill required fields.

Planning:

Resolved Reason: None
Story Points: 1
Priority: 2
Severity: 3 - Medium
Activity: None

Deployment:

To track releases associated with this work item, go to Releases and turn on deployment status reporting for Boards in your pipeline's Options menu. Learn more about deployment status reporting.

Development:

Add link

Link an Azure Repos commit, pull request or branch to see the status of your development. You can also create a branch to get started.

Effort (Hours):

Original Estimate: None
Remaining: None
Completed: None

Related Work:

Add link

Add an existing work item as a parent

Tested By: Srinivas Dharshini (Last seen)

8. Test case results

The screenshot shows the Microsoft Test Manager interface. On the left, the 'Test Suites' tree view is open, showing a 'batch data analysis' suite with two test cases: 'TS01 - User Authentication (2)' and 'TS02 - Logout Functionality (0)'. The 'TS01' node is expanded. In the center, the details for 'TS01 - User Authentication (ID: 54)' are displayed, with the 'Execute' tab selected. Under 'Test Points (2 Items)', 'TC01 - Successful Login' is highlighted. A modal window titled 'TC01 - Successful Login' shows the 'Test Case Results' table:

Outcome	TimeSta...	Configuratio...	Run by	Tester	Last
Passed	16m ago	Windows 10	Shri Dhanshni	Shri Dhanshni	last
Pending	17m ago	Windows 10	Shri Dhanshni	Shri Dhanshni	last
Passed	28m ago	Windows 10	Shri Dhanshni	Shri Dhanshni	last

At the bottom of the modal, there is a link: 'Open execution history for current test point'.

9. Test report summary

Bug #60: not logging in due to system error

Repro Steps:

- 1. Passed: Navigate to the login page.
- 2. Passed: Leave username and/or password fields empty.
- 3. Failed: Click on 'Login'.

Expected Result:

- Validation error message is shown prompting user to fill required fields.

Planning:

- Priority: 2
- Sentiment: 3 - Medium
- Activity: 0

Deployment:

To track releases associated with this work item, go to Releases and turn on deployment status reporting for Boards in your pipeline's Options menu. Learn more about deployment status reporting.

Development:

Add link

Link an Azure Repos commit, pull request, or branch to see the status of your development. You can also create a branch to get started.

Related Work:

Add link

Add an existing work item as a parent

Tested By: [Redacted]

- Assigning bug to the developer and changing state

Bug #60: not logging in due to system error

Assignee: Shri Dhashtini

Repro Steps:

- 1. Passed: Navigate to the login page.
- 2. Passed: Leave username and/or password fields empty.
- 3. Failed: Click on 'Login'.

Expected Result:

- Validation error message is shown prompting user to fill required fields.

Planning:

- Priority: 2
- Sentiment: 3 - Medium
- Activity: 0

Deployment:

To track releases associated with this work item, go to Releases and turn on deployment status reporting for Boards in your pipeline's Options menu. Learn more about deployment status reporting.

Development:

Add link

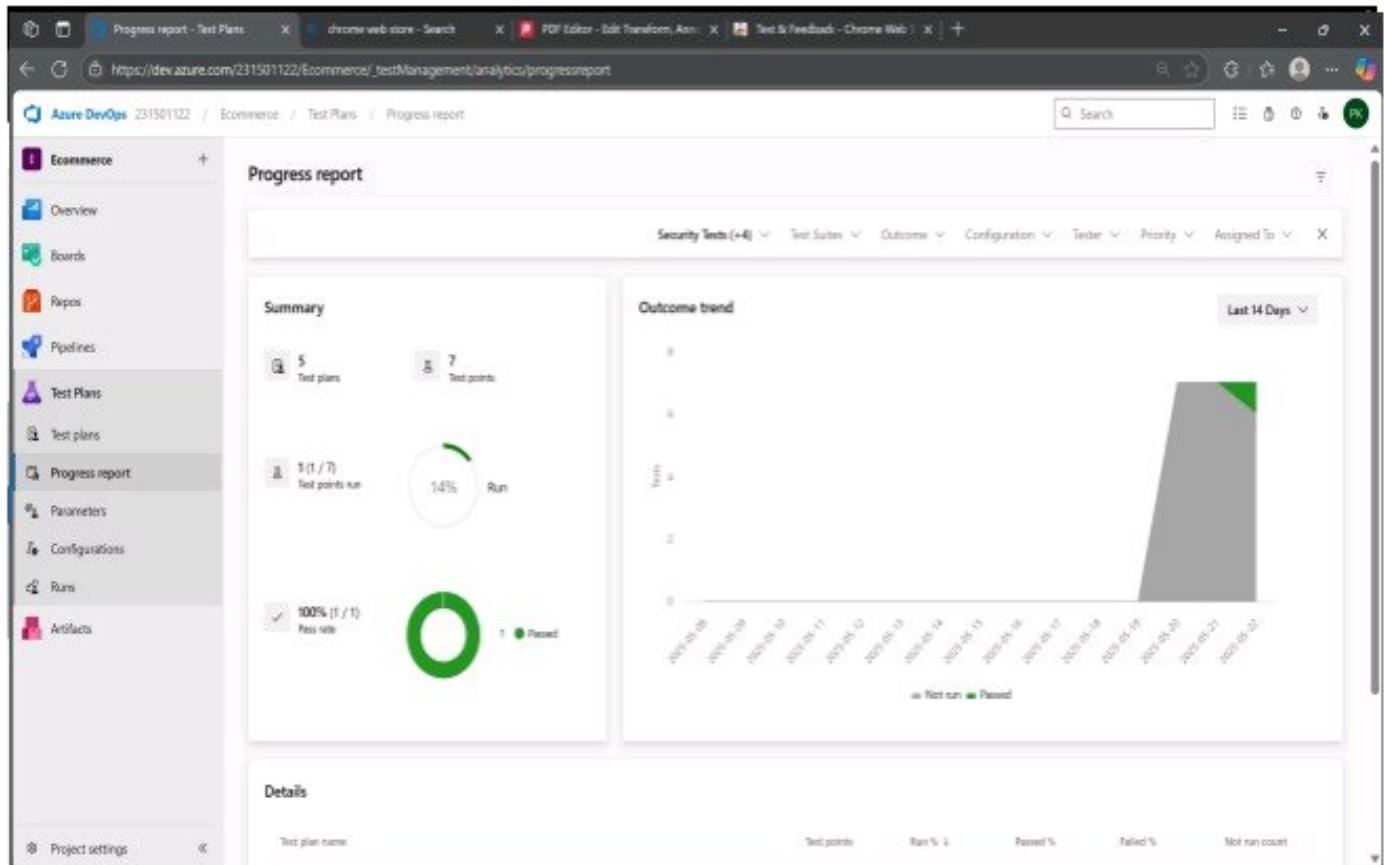
Link an Azure Repos commit, pull request, or branch to see the status of your development. You can also create a branch to get started.

Related Work:

Add link

Add an existing work item as a parent

10. Progress report



11. Changing the test template

The screenshot shows the 'All processes' section of the Azure DevOps settings. The 'Agile (default)' template is selected, indicated by a blue highlight. The 'Basic' template is also listed. Other templates like 'Scrum' and 'OMNI' are visible but not selected.

12. View the new test case template

The screenshot shows the 'Add a field to Test Case' dialog box. A new field named 'Test' is being created, defined as a 'Text (single line)' type. The 'Acceptance Criteria' field is also visible.

Azure DevOps Settings - Organization Settings (2116231501123) / Settings / Process

All processes > BATCH DATA ANALYSIS > Test Case

Layout States Rules

New field New group New page Get extensions ...

Steps Summary Automated Job...

Steps Text (multiple lines)

Recent test results Recent test case results

Deployment Deployments

Development Links

Related Work Links

Status Priority (integer)

Automation status Test (single line)

Custom

Test Text (single line)

Azure DevOps Settings - Organization Settings (2116231501123) / Settings / Process

All processes > Agile

Work item type: Tracking level: Projects

Name Description

Batch data analysis and visualization

Digital lending library application

digital library

SHREDDERSHRE

Result:The test plans and test cases for the user stories is created in Azure DevOps with Happy Path and Error Path.

EXP NO: 9	CI/CD PIPELINES IN AZURE
Date:	

Aim:

To create and demonstrate an Azure DevOps pipeline for automating application builds, tests, and deployment.

PROCEDURE:**Steps to Create and implement pipelines in Azure:**

1. Sign in to Azure DevOps and Navigate to Your Project

Log in to dev.azure.com, select your organization, and open the project where your Student Management System code resides.

2. Connect a Code Repository (Azure Repos or GitHub)

Ensure your application code is stored in a Git-based repository such as Azure Repos or GitHub. This will be the source for triggering builds and deployments in your pipeline.

3. Create a New Pipeline

Go to the Pipelines section on the left panel and click “Create Pipeline”.

Choose your source (e.g., Azure Repos Git or GitHub), and then select the repository containing your project code.

4. Choose the Pipeline Configuration

You can select either the YAML-based pipeline (recommended for version control and automation) or the Classic Editor for a GUI-based setup.

If using YAML, Azure DevOps will suggest a template or allow you to define your own.

5. Define Build Stage (CI - Continuous Integration) from YAML file

6. Install dependencies (e.g., npm install, dotnet restore)

7. Build the application (dotnet build, npm run build)

8. Run unit tests (dotnet test, npm test)

9. Publish build artifacts to be used in the release stage

10. Save and Run the Pipeline for the First Time

Save the YAML or build definition and click “Run”.

Azure will fetch the latest code and execute the defined build and test stages.

11. Configure Continuous Deployment (CD)

Navigate to the Releases tab under Pipelines and click “New

Release Pipeline”. Add an Artifact (from the build stage) and create

a new Stage (e.g., Development, Production).

12. Configure the CD stage with deployment tasks such as deploying to Azure

App Service, running database migrations or scripts, and restarting services

using the Azure App Service Deploy task linked to your subscription and app details.

13. Set Triggers and Approvals

Enable continuous deployment trigger so the release pipeline runs automatically after a successful build.

For production environments, configure pre-deployment approvals to ensure manual verification before release.

14. Monitor Pipelines and Manage Logs

View all pipeline runs under the Runs section.

Check logs for build/test/deploy stages to debug any errors.

You can also integrate email alerts or Microsoft Teams notifications for build failures.

15. Review and Maintain Pipelines

Regularly update your pipeline tasks or YAML configurations as your application grows. Ensure pipeline runs are clean and artifacts are stored securely.

Integrate quality gates and code coverage policies to maintain code quality.

Pipeline

The screenshot shows the Azure DevOps Pipeline interface for the Ecommerce project. The pipeline run is titled '#20250520.1 • Set up CI with Azure Pipelines' and is associated with the Ecommerce repository. The run was triggered by Prathisha R and started at 7:00 PM today. It completed successfully in 12 minutes. The interface includes a summary of the run, a list of jobs, and links for code coverage and getting started.

Azure DevOps 231501123 / Ecommerce / Pipelines / Ecommerce (R) / 20250520.1

Run new

Ecommerce

Overview

Boards

Repos

Pipelines

Environments

Library

Test Plans

Artifacts

#20250520.1 • Set up CI with Azure Pipelines

Ecommerce [4]

This run will be cleaned up after 1 month based on your project settings.

Summary Code Coverage

Individual CI by Prathisha R

Repository and version:

- Ecommerce
- IP main · 9lebal40

Time started and elapsed:

- Today at 7:00 PM
- 12m

Related:

- 0 work items
- 0 artifacts

Jobs

Name	Status	Duration
Job	Success	12m

Jobs

Code coverage

Get started

Result:

Successfully demonstrated pipelines in azure devops

EXP NO: 10

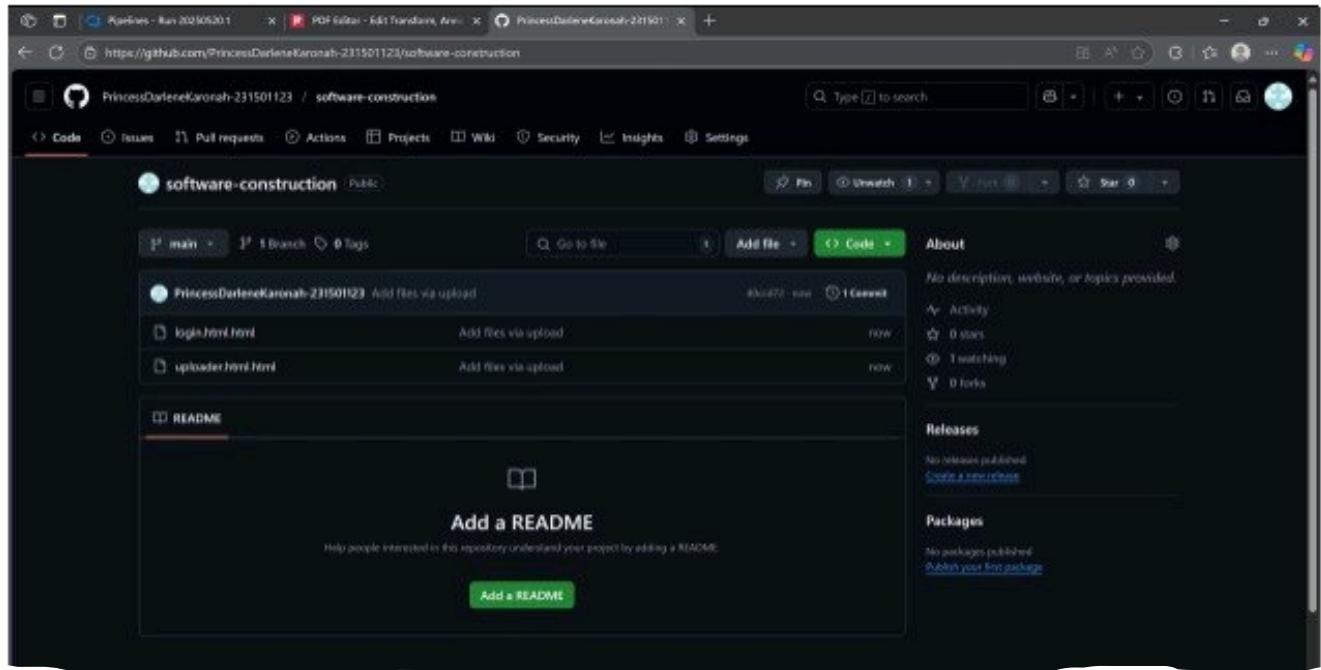
Date :

GITHUB: PROJECT STRUCTURE & NAMING CONVENTIONS

Aim:

To provide a clear and organized view of the project's folder structure and file naming conventions, helping contributors and users easily understand, navigate, and extend the Ecommerce Product uploader.

GitHub Project Structure



Result:

The GitHub repository clearly displays the organized project structure and consistent naming conventions, making it easy for users and contributors to understand and navigate the codebase.