

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>

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.

2. Azure home page

The screenshot shows the Microsoft Azure home page. At the top, there's a search bar with placeholder text "Search resources, services, and docs (G+J)". Below the search bar is a navigation bar with icons for Home, Notifications, Settings, and Help. The main content area is titled "Azure services" and features a grid of service icons: Create a resource, Azure DevOps organizations, Quickstart Center, Azure AI foundry, Kubernetes services, Virtual machines, App Services, Storage accounts, and SQL databases. A "More services" button is also present. Below this is a section titled "Resources" with tabs for "Recent" (which is selected) and "Favorite". It displays a table with columns for "Name", "Type", and "Last Viewed". A message "No resources have been viewed recently" is shown, along with a "View all resources" button. Further down, there's a "Navigate" section with links to Subscriptions, Resource groups, All resources, and Dashboard. Finally, a "Tools" section lists Microsoft Learn, Azure Monitor, Microsoft Defender for Cloud, and Cost Management.

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

The screenshot shows the Azure DevOps search results page. The search bar at the top contains the query "azure dev". Below the search bar, there are three tabs: "All" (selected), "Services (99+)", and "Resources". The "Services" tab is active, displaying a list of services under categories like Services, Marketplace, and Documentation. Services listed include Azure Device Registry, Azure DevOps organizations, Azure Database for MySQL servers, Education, Azure DevOps Auditing, Azure Devops Backup Tool, Self Hosted Runner for Azure DevOps, Install the Azure Developer CLI, What is Azure Dev/Test offer?, Set up Azure Deployment Environments - Azure Deployment Environments, What is Azure Deployment Environments? - Azure Deployment Environments, and Continue searching in Microsoft Entra ID. To the right of the search results, there's a large, colorful illustration of a rocket launching from a launch pad, with clouds and a sun in the background. At the bottom of the page, there are links for "My Azure DevOps Organizations", "Get started using Azure DevOps", "Billing management for Azure DevOps", "Give feedback", and "Tell us about your experience with the Azure DevOps page".

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.

The screenshot shows the Microsoft Azure DevOps Organization Home page. At the top, there's a blue header bar with the Microsoft Azure logo, a search bar, and user account information. Below the header, the page title is "Azure DevOps". A banner at the top of the main content area says, "We've made it easier to manage Azure DevOps billing and subscriptions. You can [set up billing](#), [change your subscription](#) or [pay for more users and resources](#) within Azure DevOps. Learn more". The main content features a large, colorful illustration of people working on various DevOps tasks like planning, building, and testing, with a rocket launching into the sky. To the left of the illustration, there's a section titled "Azure DevOps" with the subtext "Plan smarter, collaborate better, and ship faster with a set of modern dev services". Below this are links for "My Azure DevOps Organizations", "Get started using Azure DevOps", "Billing management for Azure DevOps", "Give feedback", and "Tell us about your experience with the Azure DevOps page".

Result:

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

EXP NO: 2

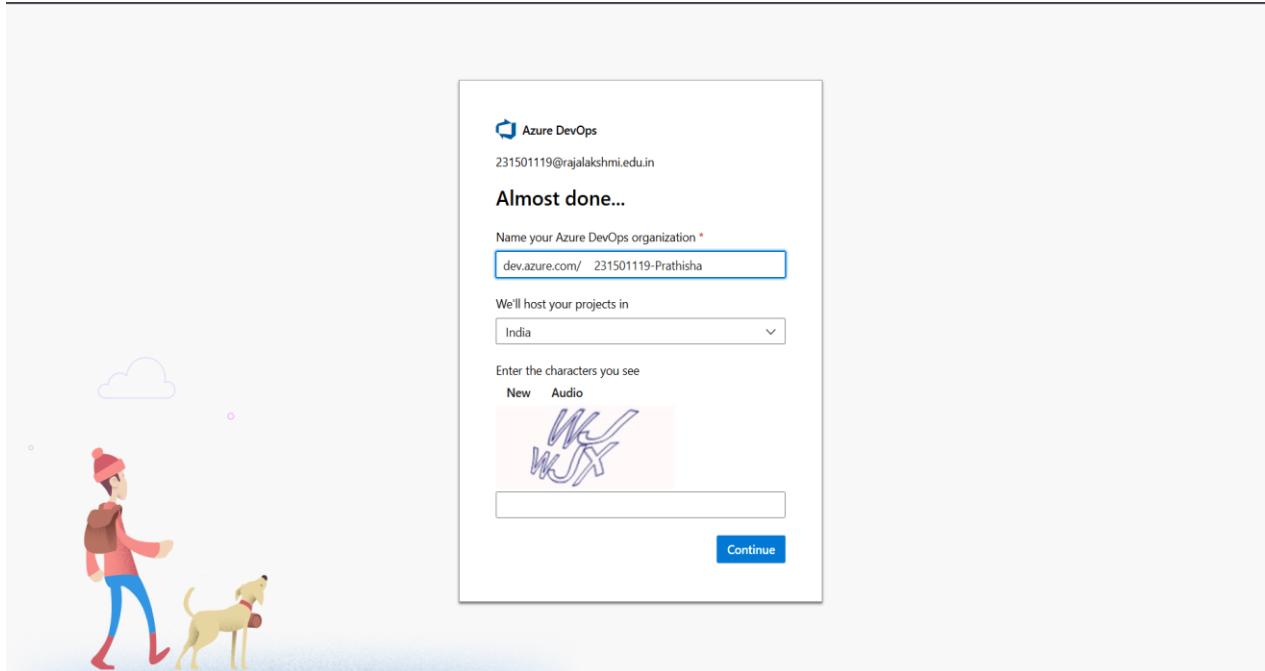
AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

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

a. 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.

b. On the organization's **Home page**, click on the **New Project** button.

c. 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).

d. 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. At the top, there's a blue header bar with the Microsoft logo and a sign-out link. Below the header is a purple circular profile picture with the letters 'PR'. To the right of the profile picture, the user's name 'Prathisha R' and email '23150119@rajalakshmi.edu.in' are displayed, along with an 'Edit profile' button. On the left, there's a sidebar for 'Visual Studio Dev Essentials' with a brief description and a 'Use your benefits' link. The main content area is titled 'Azure DevOps Organizations' and lists several organizations under 'Projects': 'dev.azure.com/2315011190742' (Owner), 'dev.azure.com/2315011191191' (Owner), 'dev.azure.com/23150111999999' (Owner), and 'dev.azure.com/2315011122' (Member). There are also 'Actions' like 'Open in Visual Studio'.

4. Project dashboard

The screenshot shows the Azure DevOps E-commerce project dashboard. The left sidebar includes options like Overview, Summary, Dashboards, Wiki, Boards, Repos, Pipelines, Test Plans, and Artifacts. The main content area has a title 'E-commerce' and a sub-section 'About this project'. It describes the E-commerce Product Uploader application and its features. To the right, there's a 'Project stats' section with metrics for Boards, Repos, Pipelines, and Members. The 'Boards' section shows 33 work items created and 0 completed. The 'Repos' section shows 0 pull requests opened and 11 commits by 1 author. The 'Pipelines' section shows 0% builds succeeded. The 'Members' section lists six team members with their initials: SS, PR, RC, and PG.

5. To manage user stories:

- 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 'Work items' page in the Azure DevOps interface for the 'Ecommerce' project. The left sidebar is collapsed, and the main area displays a table of work items. The columns include ID, Title, Assigned To, State, Area Path, and Tags. The first item in the list is selected, showing its details: ID 80, Title 'As a developer, I want to validate the file structure to ensure all ...', Assigned To 'Rino Calvin', State 'Resolved', Area Path 'Ecommerce'. Other items listed include 'As a store owner, I want to search products by name so that I can find ...', 'As a store owner, I want to manually add or edit a product in case I m ...', etc.

The screenshot shows the 'Backlogs' page in the Azure DevOps interface for the 'Ecommerce' project. The left sidebar is collapsed, and the main area displays a table of backlog items. The columns include Order, Work Item Type, Title, State, Story..., and Value Area. The backlog items are numbered 1 to 10, all categorized as 'User Story'. The first few items are: 'As a user, I want to filter products by price or description ke...', 'As a developer, I want to validate the file structure to ensur...', 'As a store owner, I want to upload a CSV/Excel file so that I ...'. To the right of the backlog, there is a 'Planning' section titled 'Ecommerce Team backlog' showing sprints: 'sprint3' (Planned Effort: 0, 5 working days), 'sprint4' (Planned Effort: 0, 5 working days), and 'sprint_1' (Planned Effort: 0, 5 working days). A '+ New Sprint' button is also visible.

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

EXP NO: 3

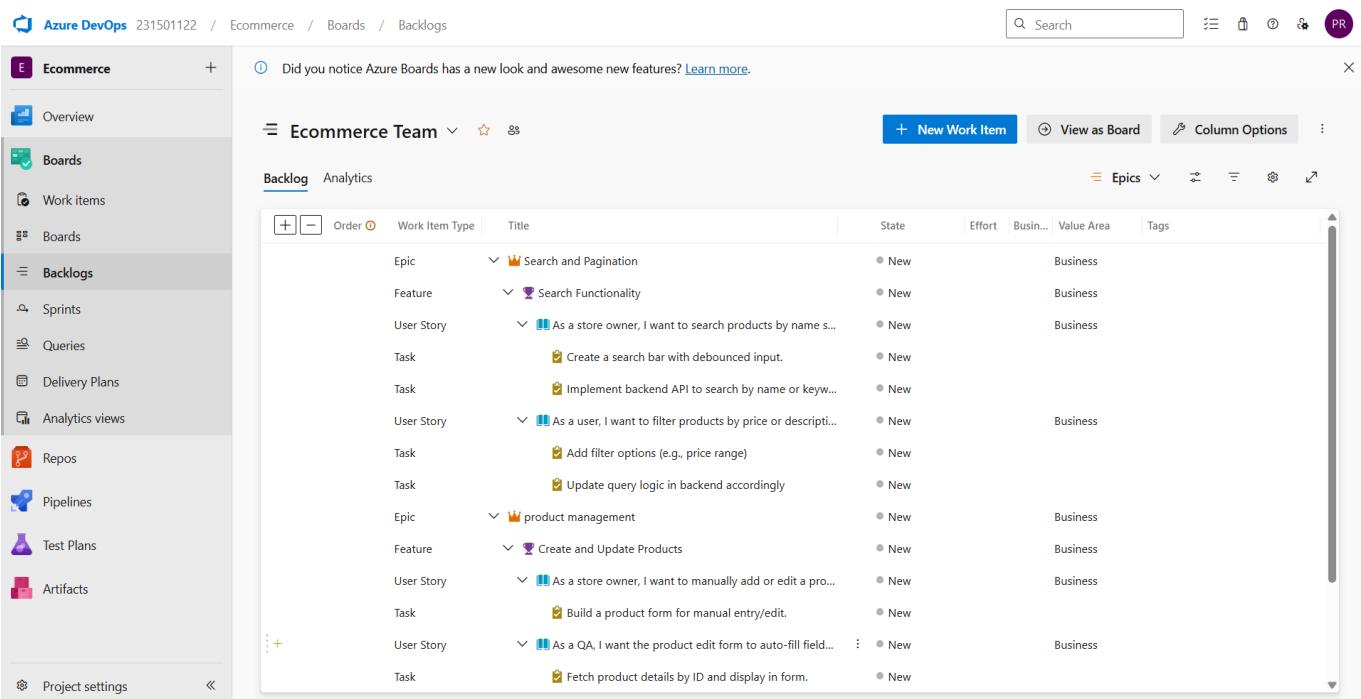
Date :

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

Aim:

To create epics, user stories, features, and tasks for the project, E - Commerce product uploader.

1.Create Epic, Features, User Stories, Task



The screenshot shows the Azure Boards interface for the 'Ecommerce' project. The left sidebar has 'Backlogs' selected. The main area displays a backlog of work items under the 'Ecommerce Team' board. The backlog is filtered to show 'Epics' and includes the following items:

Work Item Type	Title	State	Value Area
Epic	Search and Pagination	New	Business
Feature	Search Functionality	New	Business
User Story	As a store owner, I want to search products by name s...	New	Business
Task	Create a search bar with debounced input.	New	
Task	Implement backend API to search by name or keyw...	New	
User Story	As a user, I want to filter products by price or descripti...	New	Business
Task	Add filter options (e.g., price range)	New	
Task	Update query logic in backend accordingly	New	
Epic	product management	New	Business
Feature	Create and Update Products	New	Business
User Story	As a store owner, I want to manually add or edit a pro...	New	Business
Task	Build a product form for manual entry/edit.	New	
User Story	As a QA, I want the product edit form to auto-fill field...	New	Business
Task	Fetch product details by ID and display in form.	New	

2. Fill in Epics

The screenshot shows the Azure DevOps Boards backlog for the 'Ecommerce' team. The backlog is organized into columns: Order, Work Item Type, Title, State, Effort, Business Area, and Tags. The backlog items include:

Work Item Type	Title	State	Business Area
Epic	Search and Pagination	New	Business
Feature	Search Functionality	New	Business
User Story	As a store owner, I want to search products by name	New	Business
Task	Create a search bar with debounced input.	New	
Task	Implement backend API to search by name or keyw...	New	
User Story	As a user, I want to filter products by price or descripti...	New	Business
Task	Add filter options (e.g., price range)	New	
Task	Update query logic in backend accordingly	New	
Epic	product management	New	Business
Feature	Create and Update Products	New	Business
User Story	As a store owner, I want to manually add or edit a pro...	New	Business
Task	Build a product form for manual entry/edit.	New	
User Story	As a QA, I want the product edit form to auto-fill field...	New	
Task	Fetch product details by ID and display in form.	New	

3. Fill in Features

The screenshot shows the details of a feature work item titled 'upload and selection'. The work item has the following properties:

State	New	Area	Ecommerce
Reason	New	Iteration	Ecommerce\sprint3

The work item is divided into several sections:

- Description:** Click to add Description.
- Planning:** Priority: 2, Risk, Effort, Business Value, Time Criticality.
- 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.

A discussion box is present, with the placeholder text: 'Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request.' A link to 'switch to Markdown editor' is also visible.

4. Fill in User Stories

The screenshot shows the 'USER STORY 38' page in Azure DevOps. At the top, there is a title bar with the story ID '38' and the description: 'As a store owner, I want to search products by name so that I can find specific items quickly.' Below the title bar, there are several sections: 'Description' (with placeholder text 'Click to add Description.'), 'Acceptance Criteria' (with placeholder text 'Click to add Acceptance Criteria.'), 'Discussion' (with a comment input field and a 'switch to Markdown editor' link), 'Planning' (with fields for 'Story Points', 'Priority' set to '2', and 'Risk'), 'Classification' (with 'Value area' set to 'Business'), 'Deployment' (with a note about tracking releases via Azure DevOps Boards), 'Development' (with a note about linking to Azure Repos pull requests), and 'Related Work' (with an 'Add link' button). The bottom right corner shows a 'Details' button and some status indicators.

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

EXP NO: 4	SPRINT PLANNING
Date :	

Aim:

To assign a user story to a specific sprint for the project, E - Commerce product uploader.

SPRINT PLANNING

Sprint 1

The screenshot shows the Azure DevOps Boards interface for the Ecommerce project. The left sidebar navigation bar includes links for Overview, Boards, Work items, Boards, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The main area displays the 'Ecommerce Team' board under the 'Sprint 1' tab. The backlog is filtered by 'Person: All'. Two user stories are visible in the backlog:

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

The top right corner of the interface shows a search bar, a 'New Work Item' button, and a 'Column Options' dropdown. A status bar at the bottom indicates the URL as https://dev.azure.com/231501122/Ecommerce/_sprints/directory.

Sprint 2

Azure DevOps Boards interface for the E-commerce project. The left sidebar shows project navigation with 'Sprints' selected. The main board view for 'Ecommerce Team' shows a backlog of work items. A specific backlog item is expanded, showing details: '23 As a store owner, I want to upload a CSV/Excel file so that I can add multiple products at once.' Status: New, Unassigned. Another item below it is also New and Unassigned. The sprint dropdown shows 'Sprint2'.

Sprint 3

Azure DevOps Boards interface for the E-commerce project. The left sidebar shows project navigation with 'Sprints' selected. The main board view for 'Ecommerce Team' shows a backlog of work items. A specific backlog item is expanded, showing details: '80 As a developer, I want to validate the file structure to ensure all required columns are present.' Status: Resolved, Rino Calvin. Another item below it is also New and Unassigned. The sprint dropdown shows 'sprint3'.

Result: The Sprints are created for the project, E - Commerce product uploader.

EXP NO: 5

POKER ESTIMATION

Date :

Aim:

Create Poker Estimation for the user stories for the project, E - Commerce Product uploader.

Poker Estimation

The screenshot shows the Azure DevOps Boards interface for a project named 'Ecommerce'. On the left, there's a sidebar with various navigation options like Overview, Boards, Work items, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The 'Work items' option is selected. In the main area, a work item for 'USER STORY 80' is displayed. The description of the story is: 'As a developer, I want to validate the file structure to ensure all required columns are present.' The work item details include: State: Resolved, Reason: Code complete and unit tested, Area: Ecommerce, Iteration: Ecommerce\sprint3. Under the 'Description' tab, there's a note to 'Click to add Description.' Under 'Acceptance Criteria', there's a note to 'Click to add Acceptance Criteria.'. Under 'Discussion', there's a placeholder for comments: 'Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request.' To the right of the work item, there are sections for 'Planning' (Story Points, Priority, Risk), 'Classification' (Value area: Business), and 'Deployment' (with a note about tracking releases). There's also a 'Development' section with a note about linking to Azure Repos. At the bottom, there's a 'Related Work' section.

Result:

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

EXP NO: 6

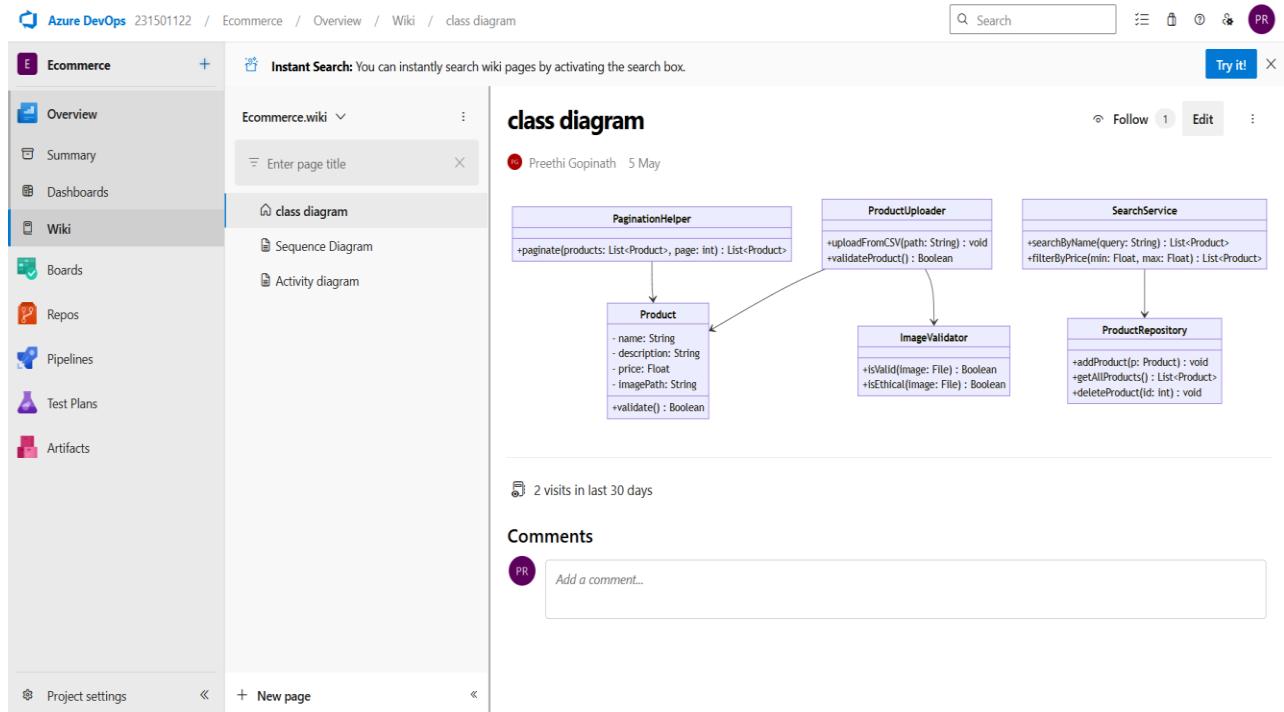
DESIGNING CLASS DIAGRAM AND SEQUENCE DIAGRAM

Date :

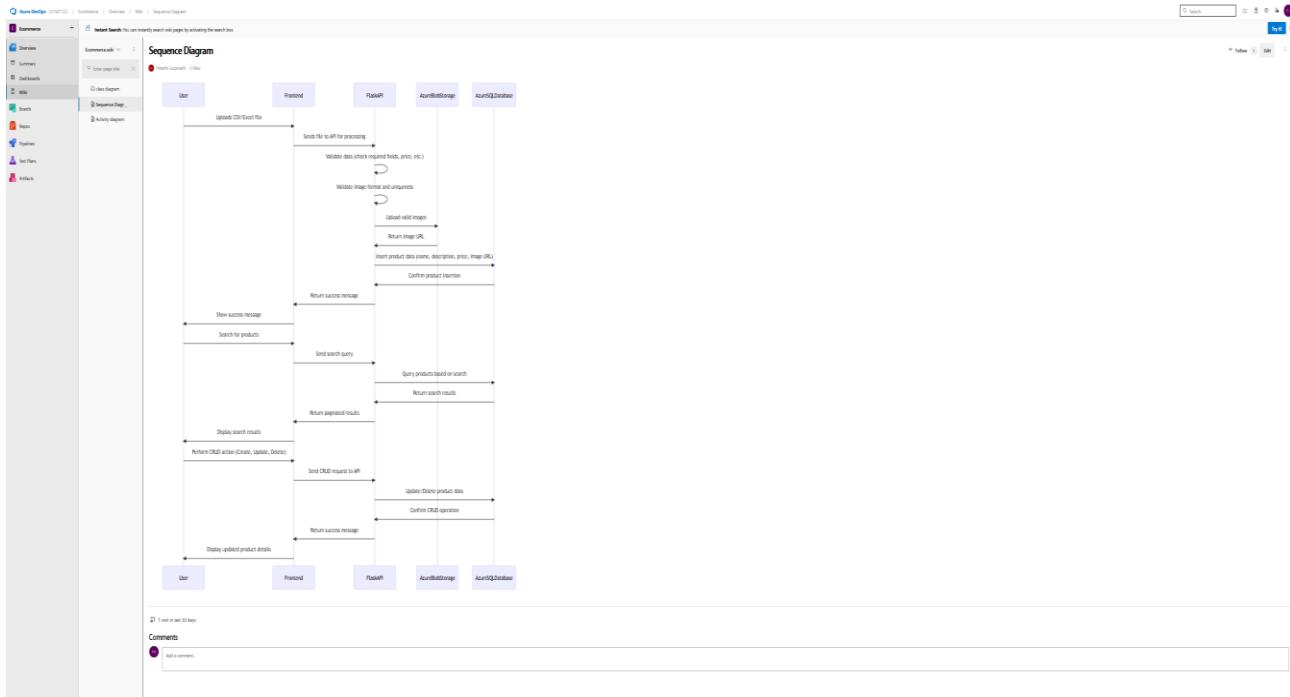
Aim:

To design a Class Diagram and Sequence Diagram for the project, E - Commerce uploader.

6A. Class Diagram



6B. Sequence Diagram



Result: The Class and Sequence Diagrams are designed successfully for the project, E - Commerce product uploader.

EXP NO: 7

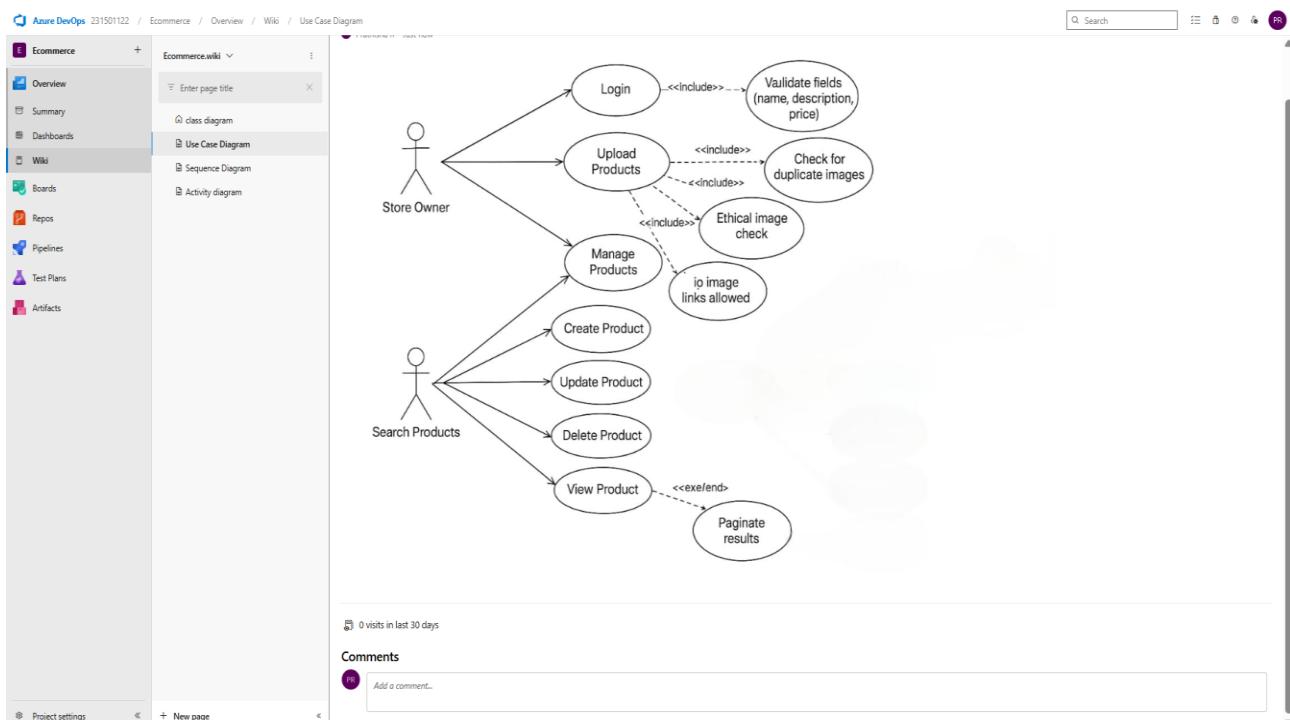
DESIGNING USE CASE DIAGRAM AND ACTIVITY DIAGRAM

Date :

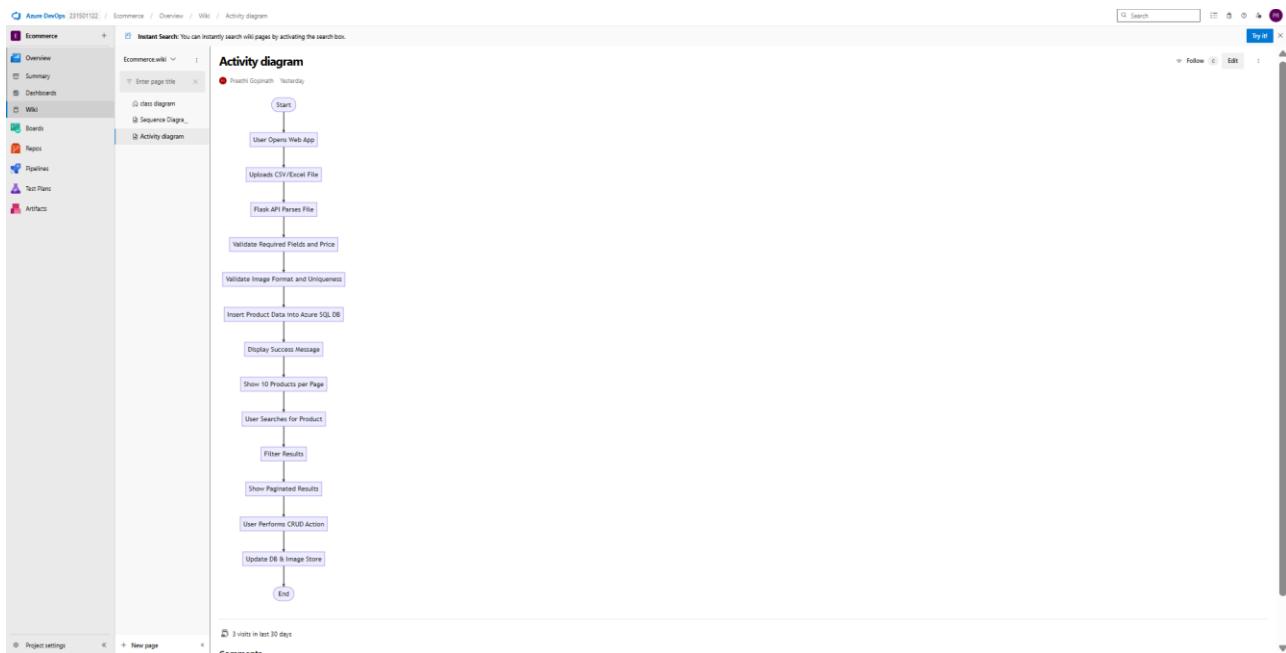
Aim:

To design a Use Case Diagram and an Activity Diagram for the project, E - Commerce product uploader.

7A. Use Case Diagram



7B. Activity Diagram



Result: The Use Case and Activity Diagrams are designed successfully for the project, E - Commerce 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

Azure DevOps 231501122 / Ecommerce / Test Plans

New Test Plan

Name *: login

Area Path *: Ecommerce

Iteration *: Ecommerce

Create Cancel

2. Test suite

Azure DevOps 231501122 / Ecommerce / Test Plans / login

Test Suites

Test Cases (1 item)

test case (2)

Define Execute Chart

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 displays two separate test plan interfaces side-by-side.

Top Test Plan (TS02- Logout Functionality):

- Title:** TS02- Logout Functionality (ID: 47)
- Status:** May 17 - May 24, 100% run, 100% passed. [View report](#)
- Test Suites:** BatchDataAnalysis (Current), TS02- Logout Functionality (2)
- Test Cases:** (2 items)

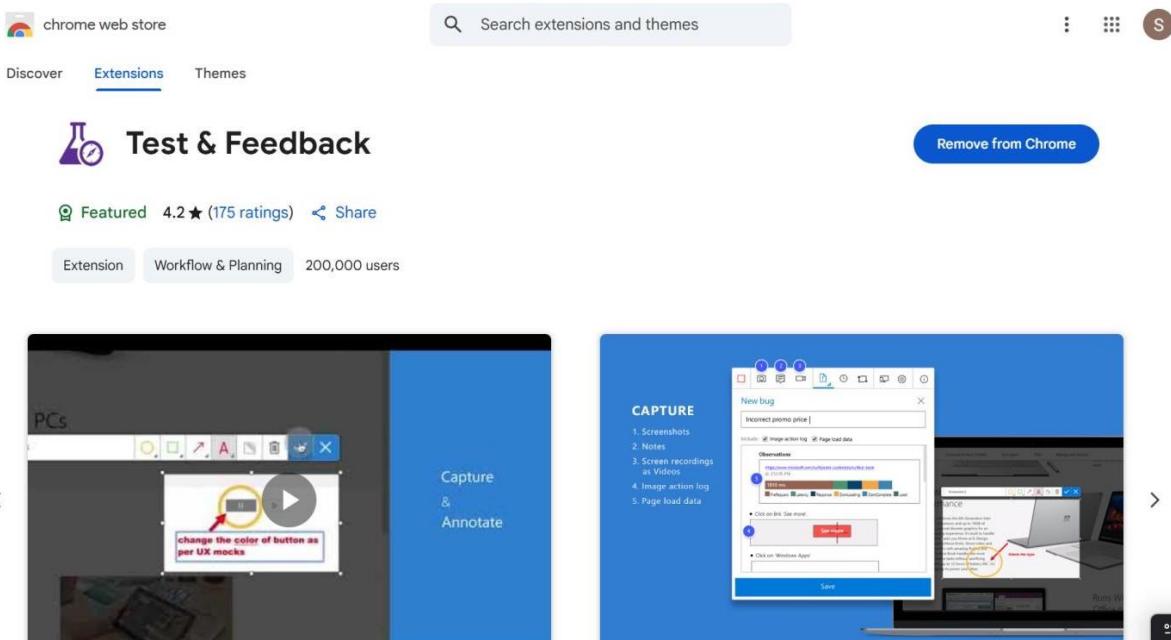
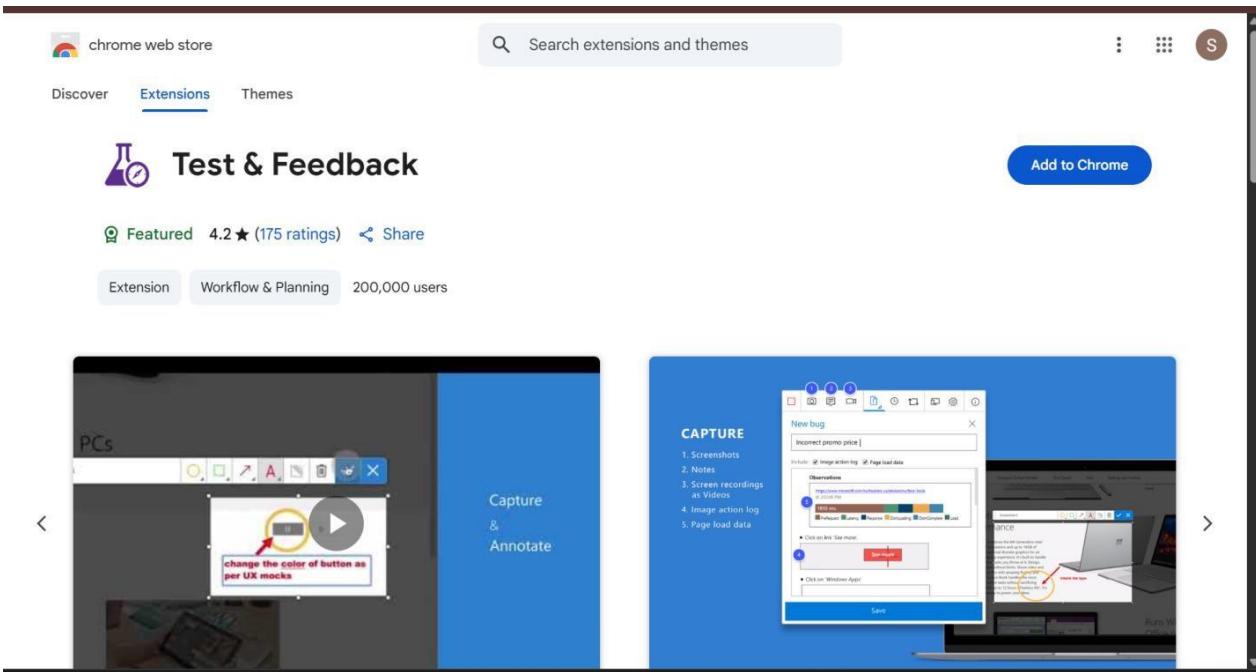
Title	Order	Test Case Id	Assigned To	State
TC03- Successfull logout and redirect	1	49	NIKSHITHA H	Design
TC04- Access protected page after logout	2	50	NIKSHITHA H	Design

Bottom Test Plan (TS01- User Authentication):

- Title:** TS01- User Authentication (ID: 54)
- Status:** May 17 - May 24
- Test Suites:** batch data analysis (Current), TS01- User Authentication (2), TS02- Logout Functionality (2)
- Test Points:** (2 items)

Title	Outcome	Order	Test Case Id	Configuration	Tester
TC01 – Successful Login	Passed	1	57	Windows 10	Shri Dharsini
TC02 – Prevent Login with Empty Fields	Failed	2	58	Windows 10	Shri Dharsini

4. Installation of test



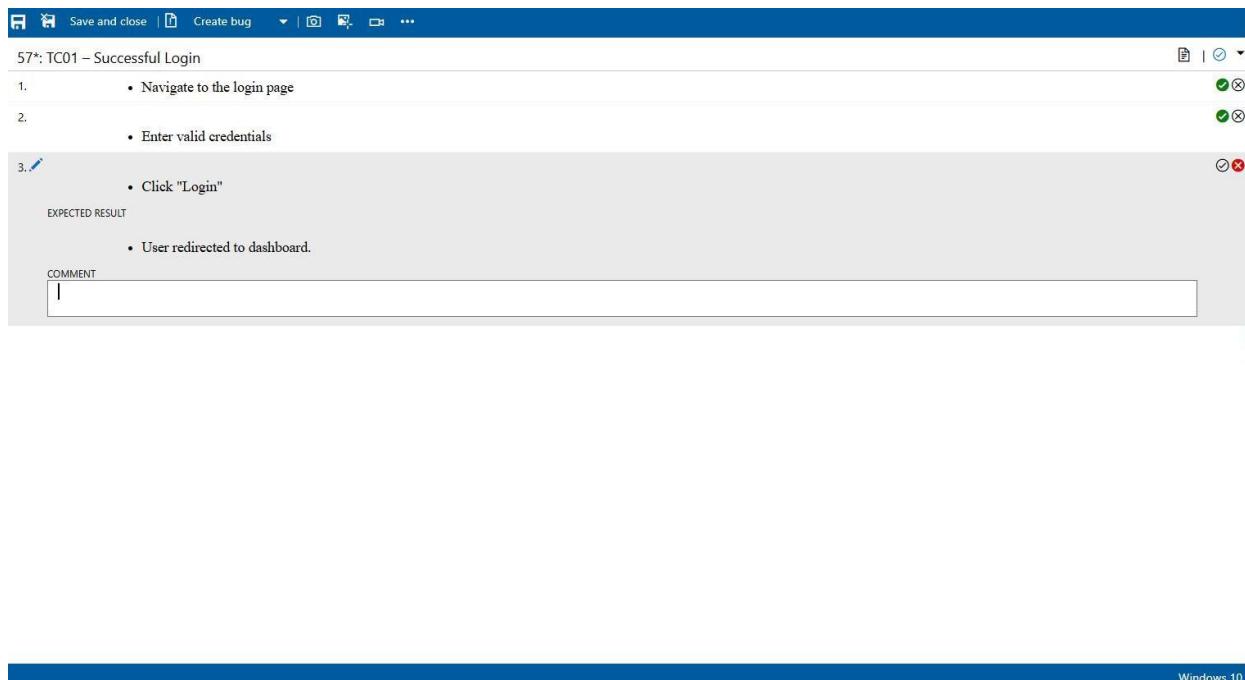
Test and feedback

Showing it as an extension

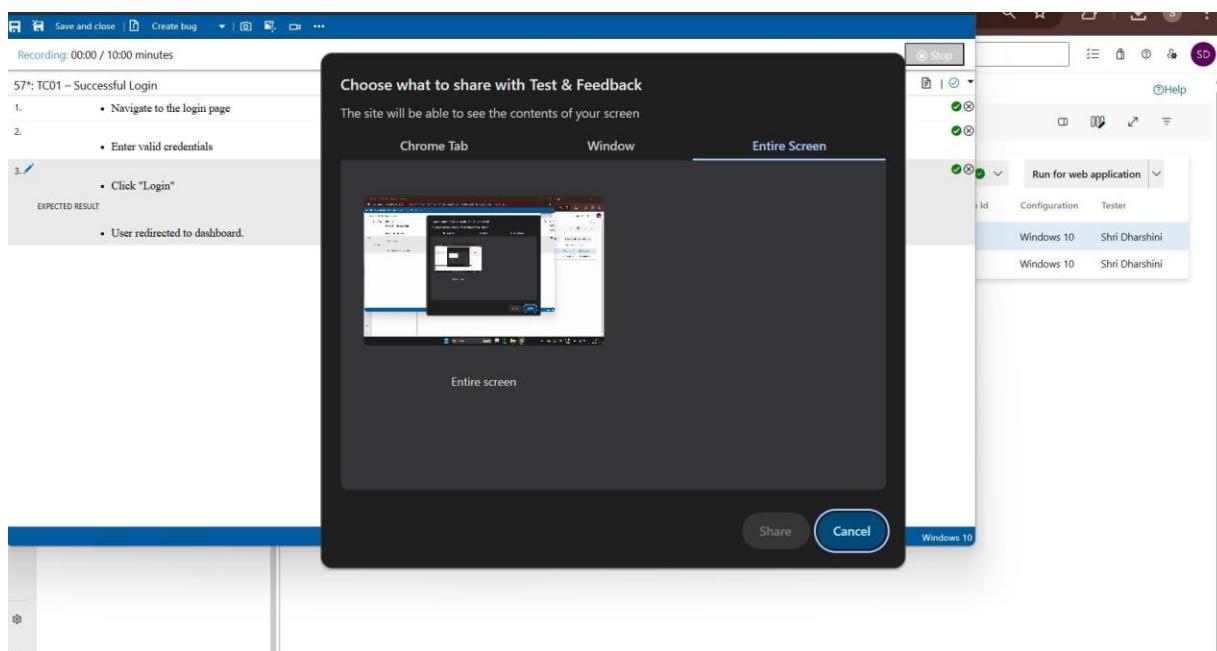
The screenshot shows the Microsoft Test Manager interface. On the left, there's a navigation pane with icons for different test management tasks. The main area displays a 'Test Suites' list under 'batch data analysis'. One suite, 'TS01- User Authentication (2)', is selected. The 'Execute' tab is active, showing 'Test Points (2 items)'. Two test points are listed: 'TC01 – Successful Login' (Passed) and 'TC02 – Prevent Login with Empty Fields' (Failed). A modal window titled 'Extensions' is overlaid on the screen. It has a 'Full access' section stating 'These extensions can see and change information on this site.' It lists 'Test & Feedback' as an extension with a 'Manage extensions' button. At the bottom right of the modal, it says 'Windows 10 Shri Dharshini'. The top right corner of the main window shows the user profile 'Dharshini' and the email '153@rajalakshmi.edu.in'.

5. Running the test cases

This screenshot shows the same Microsoft Test Manager interface as the previous one, but with a context menu open over the 'TC01 – Successful Login' row in the 'Test Points' list. The menu options are: 'View execution history', 'Mark Outcome', 'Run', 'Reset test to active', 'Edit test case', 'Assign tester', and 'View test result'. A sub-menu for 'Run' is expanded, showing 'Run for web application', 'Run for desktop application', and 'Run with options'. The rest of the interface remains consistent with the first screenshot, showing the 'Test Suites' list and the results of the two test points.



6. Recording the test case



7. Creating the bug

The screenshot shows a bug creation interface with the following details:

TC02 – Prevent Login with Empty Fields

Steps:

1. • Navigate to the login page.
2. • Leave username and/or password fields empty.
3. • Click on "Login".

EXPECTED RESULT:

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

The screenshot shows a bug detail view for "TC02 – Prevent Login with Empty Fields".

Repro Steps:

Step	Result	Title
1.	Passed	• Navigate to the login page.
2.	Passed	• Leave username and/or password fields empty.
3.	Failed	

Planning:

- Resolved Reason:
- Story Points:
- Priority: 2
- Severity: 3 - Medium
- Activity:

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.

Repro Steps

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

Step no.	Result	Title
1.	Passed	<ul style="list-style-type: none"> • Navigate to the login page.
2.	Passed	<ul style="list-style-type: none"> • Leave username and/or password fields empty.
3.	Failed	<ul style="list-style-type: none"> • Click on "Login". <p>Expected Result</p> <ul style="list-style-type: none"> • Validation error message is shown prompting user to fill required fields.

Planning

Resolved Reason
Story Points
Priority 2
Severity 3 - Medium
Activity

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
Remaining
Completed

Related Work

Add link

Add an existing work item as a parent

Tested By

58 TC02 – Prevent Login with Empty Fields

8. Test case results

nikshithaharikrishnan2005 / Batch Data Analysis and Vis... / Test Plans / batch data analysis

TS01- User Authentication (ID: 54)

Test Suites

- batch data analysis (May 17 - May 24, 50% run, 50% passed. View report)
 - TS01- User Authentication (2)
 - TS02- Logout Functionality (2)

Test Points (2 items)

Title
<input checked="" type="checkbox"/> TC01 – Successful Login
<input type="checkbox"/> TC02 – Prevent Login with Empty Fields

TC01 – Successful Login

Test Case Results

Outcome	TimeSta...	Configuration	Run by	Tester	Test
Passed	16m ago	Windows 10	Shri Dharshini	Shri Dharshini	batch
Failed	17m ago	Windows 10	Shri Dharshini	Shri Dharshini	batch
Passed	28m ago	Windows 10	Shri Dharshini	Shri Dharshini	batch

Open execution history for current test point

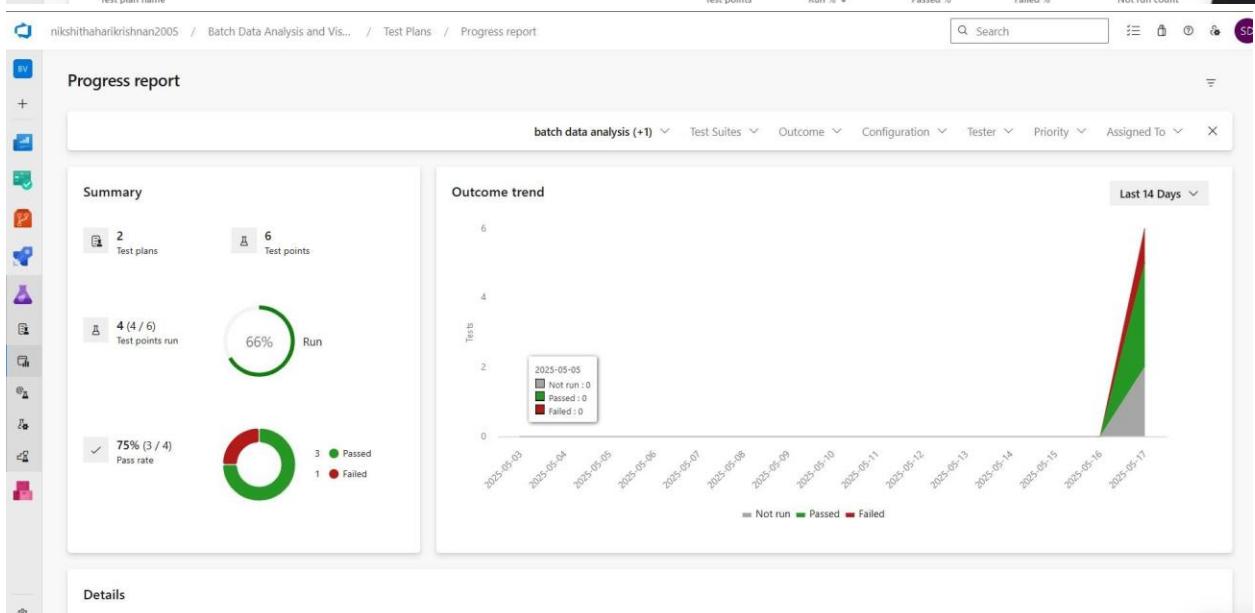
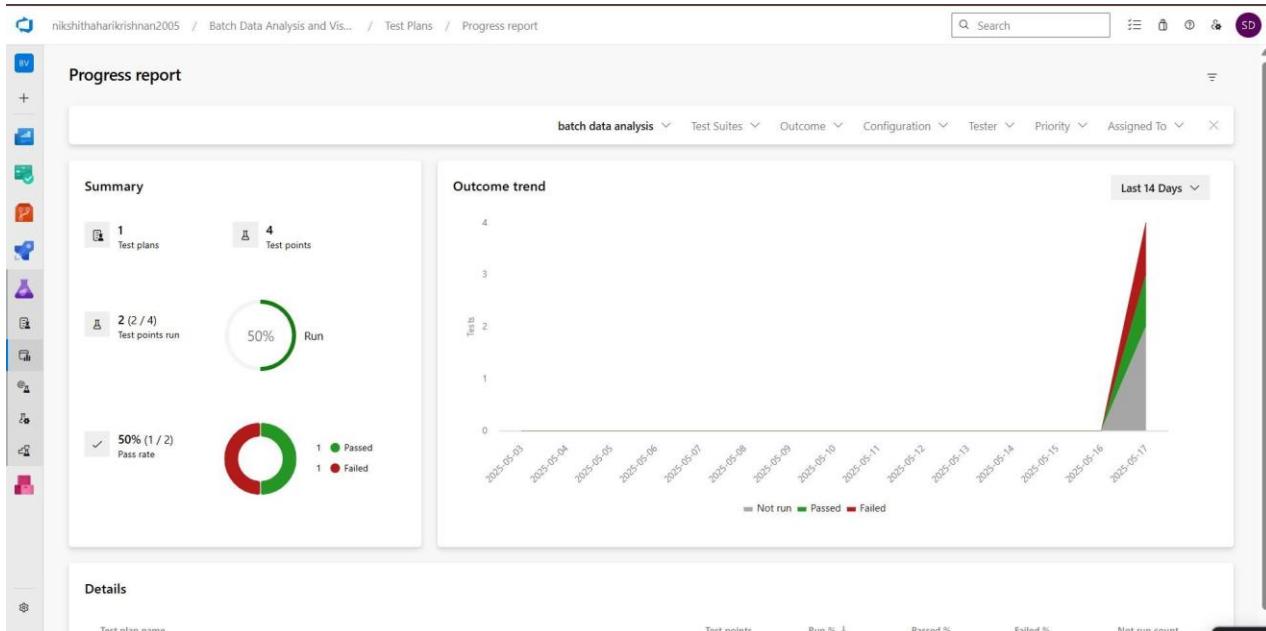
9. Test report summary

The screenshot shows the Azure Boards Work Item Detail page for a bug titled "BUG 60: not logging in due to system error". The work item is currently in the "New" state. The "Repro Steps" section contains three steps: 1. Passed (Navigate to the login page), 2. Passed (Leave username and/or password fields empty), and 3. Failed (Click on "Login"). The "Expected Result" is a validation error message prompting the user to fill required fields. The "Planning" section shows the bug was filed on "17/05/2025 10:51" with the title "Bug filed on 'TC02 – Prevent Login with Empty Fields'". The "Deployment" section includes a note about tracking releases and a link to Azure Repos. The "Development" section has an "Add link" button. The "Related Work" section has buttons for "Add link" and "Add an existing work item as a parent". The "Effort (Hours)" section shows original and remaining estimates.

- Assigning bug to the developer and changing state

The screenshot shows the same Azure Boards Work Item Detail page for BUG 60, but now the developer "Shri Dharshini" is listed under the assignee. The rest of the details remain the same, including the repro steps, expected result, planning information, deployment status, development links, and related work sections.

10. Progress report



11. Changing the test template

The screenshot shows the Azure DevOps Settings page under the Process section. On the left, there's a sidebar with General, Security, and Boards sections. The main area displays a list of process templates under 'All processes'. The 'Agile (default)' template is selected, and its details are shown: 'This template is flexible for any process and great for teams getting started with Azure DevOps.' Below it are other templates like 'Basic', 'Agile plus', 'BatchDataAnalysis', 'Scrum', and 'CMMI'.

12. View the new test case template

The screenshot shows the 'Add a field to Test Case' dialog box. It has tabs for Definition, Options, and Layout. Under Definition, the 'Create a field' option is selected with 'Name: text' and 'Type: Text (single line)'. The 'Layout' tab shows a 'Steps' section with 'Text (multiple lines)' and a 'New field' button. The background shows the 'All processes > Agile' section of the Azure DevOps settings.

Azure DevOps 231501153 / Settings / Process

Organization Settings 231501153

All processes > BATCH DATA ANALYSIS > Test Case

Layout States Rules

New field New group New page Get extensions

Steps Summary Associated Aut...

Steps Text (multiple lines)

Recent test results Recent test case results

Custom text Text (single line) ...

Deployment Deployments

Development Links

Related Work Links

Status

Priority Integer

Automation status Text (single line)

This screenshot shows the 'Process' settings for the 'BATCH DATA ANALYSIS' template. It displays a list of fields: 'Steps' (Text (multiple lines)), 'Recent test results' (Recent test case results), 'Custom' (text, Text (single line)), 'Deployment' (Deployments), 'Development' (Links), 'Related Work' (Links), and 'Status' (Priority Integer, Automation status). A note at the top states: 'System processes cannot be customized. To add customization create an inherited process.'

Azure DevOps 231501153 / Settings / Process

Organization Settings 231501153

All processes > Agile

Work item types Backlog levels Projects

Name	Description
Batch data analysis and visualization	About this project This project is a web-based application designed for batch data analysis and visualization, hosted on Microsoft Azure. It ena...
Digital lending library application	
digital library	
SHRI DHARSHINI	

This screenshot shows the 'Process' settings for the 'Agile' template. It displays a table of work item types: 'Batch data analysis and visualization', 'Digital lending library application', 'digital library', and 'SHRI DHARSHINI'. A note at the top states: 'System processes cannot be customized. To add customization create an inherited process.'

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. On the left, there's a sidebar with icons for Overview, Boards, Repos, Pipelines (which is selected), Environments, Library, Test Plans, and Artifacts. Below the sidebar, there are links for Project settings and a back arrow. The main area displays a pipeline run titled '#20250515.1 • Set up CI with Azure Pipelines'. A message indicates that this run is being retained as one of 3 recent runs by pipeline. There are tabs for Summary and Code Coverage. The Summary section includes a 'Pull request by' summary, repository information (azure-pipelines-2, commit 29a77a3f), time started and elapsed (Just now, 20s), related work items (0), and tests and coverage (3 commits). It also has a 'Get started' button. Below this is a 'Jobs' table with one entry: 'Job' (Status: Success, Duration: 12s). At the top right, there are buttons for 'Run new' and 'View retention leases'.

Result:

Successfully demonstrated pipelines in azure devops

EXP NO: 10

Date :

GITHUB: PROJECT STRUCTURE & NAMING CONVENTION NS

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 E - Commerce Product uploader project.

GitHub Project Structure

The screenshot shows a GitHub repository page for 'product-uploader'. The repository is public and was created by user '231501119'. It has 1 branch and 0 tags. The main branch contains four files: 'index.html', 'login.html', and 'uploader.html', all updated yesterday. There is also a 'README' file. The repository has 5 commits, 1 watch, and 0 forks. It has no releases or packages published. Deployment history shows one deployment to 'github-pages' yesterday, with 4 more deployments listed.

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.