

# 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:**

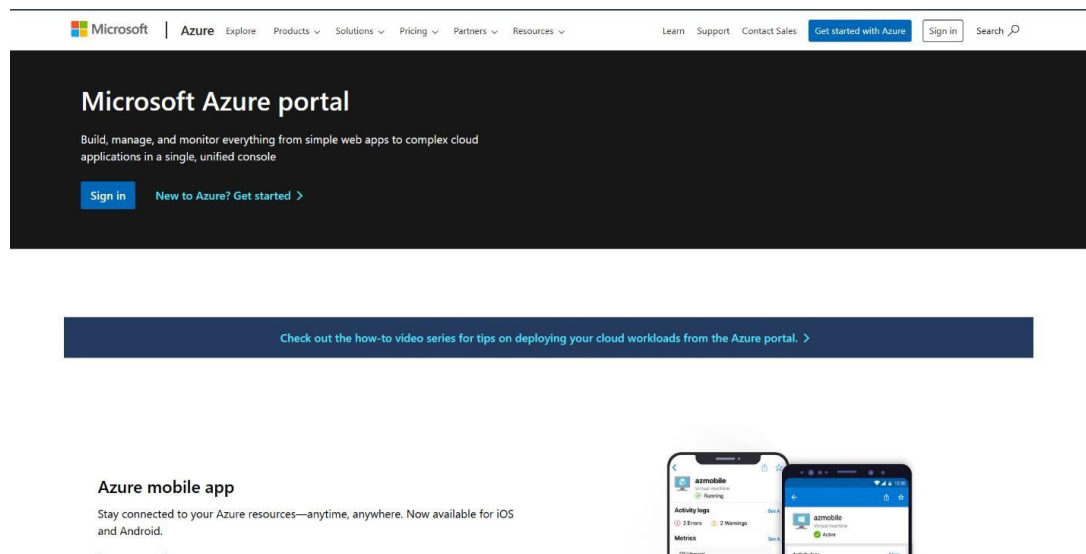
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

### **INSTALLATION**

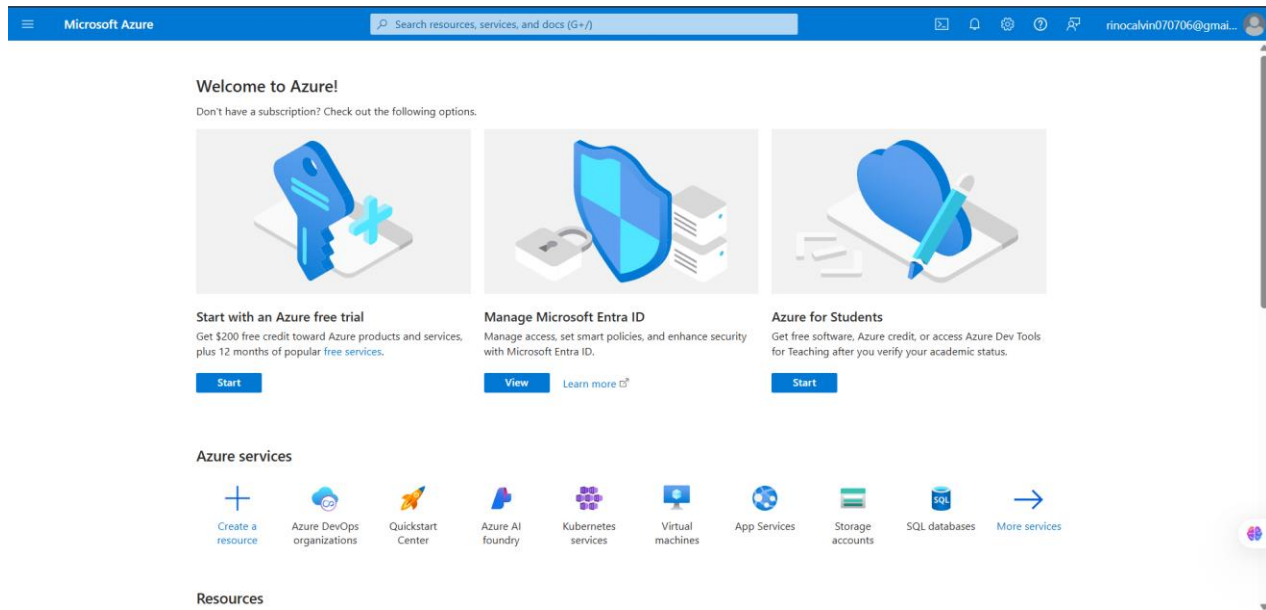
1. Open your web browser and go to the Azure website: <https://azure.microsoft.com/en-us/get-started/azure-portal>.

Sign in using your Microsoft account credentials.

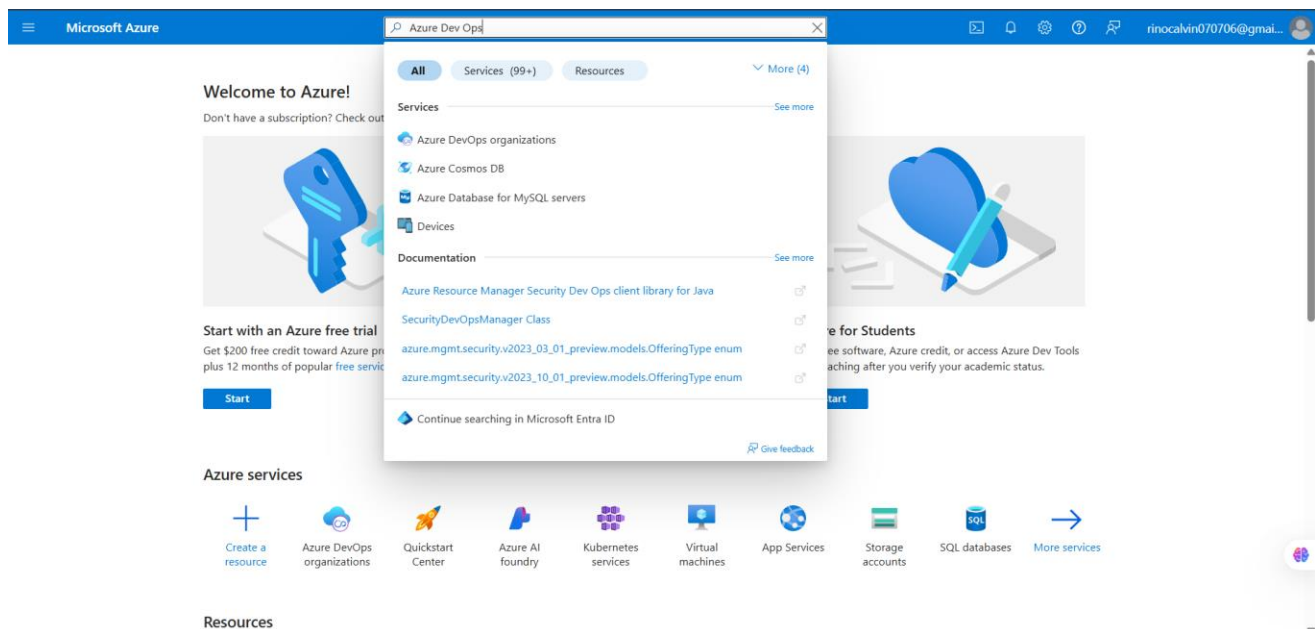
If you don't have a Microsoft account, you can create one here: <https://signup.live.com/?lic=1>



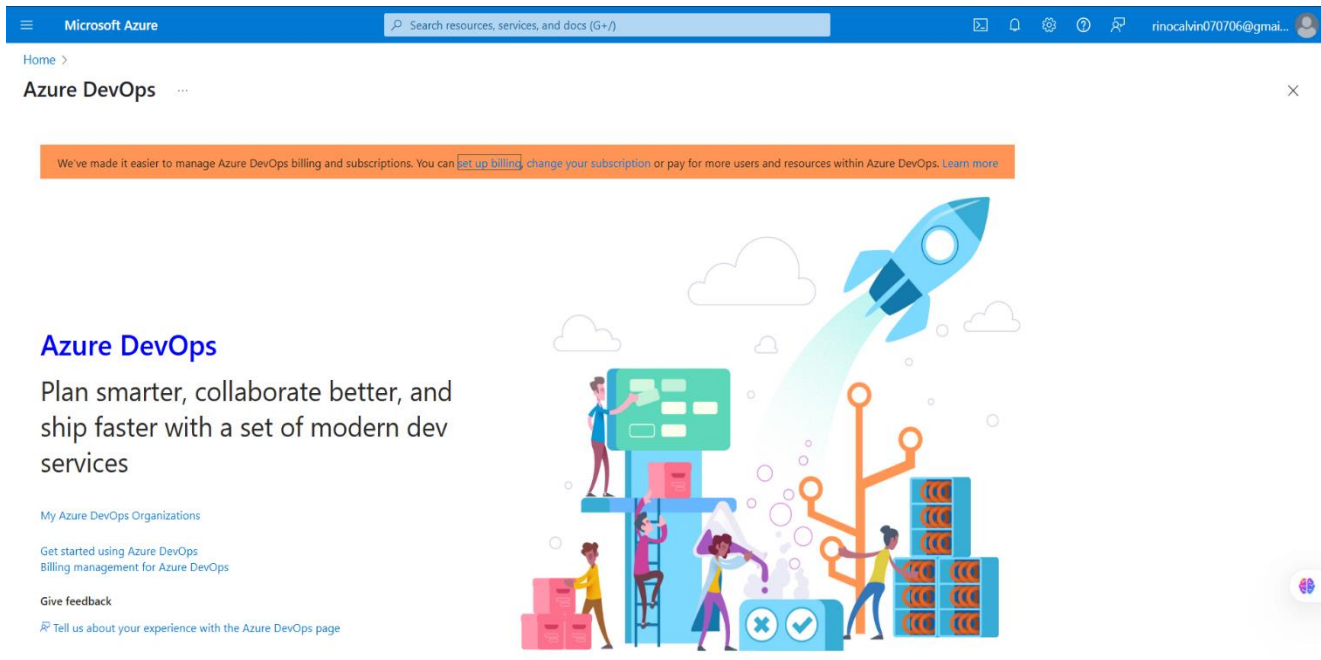
## 2. Azure home page



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



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

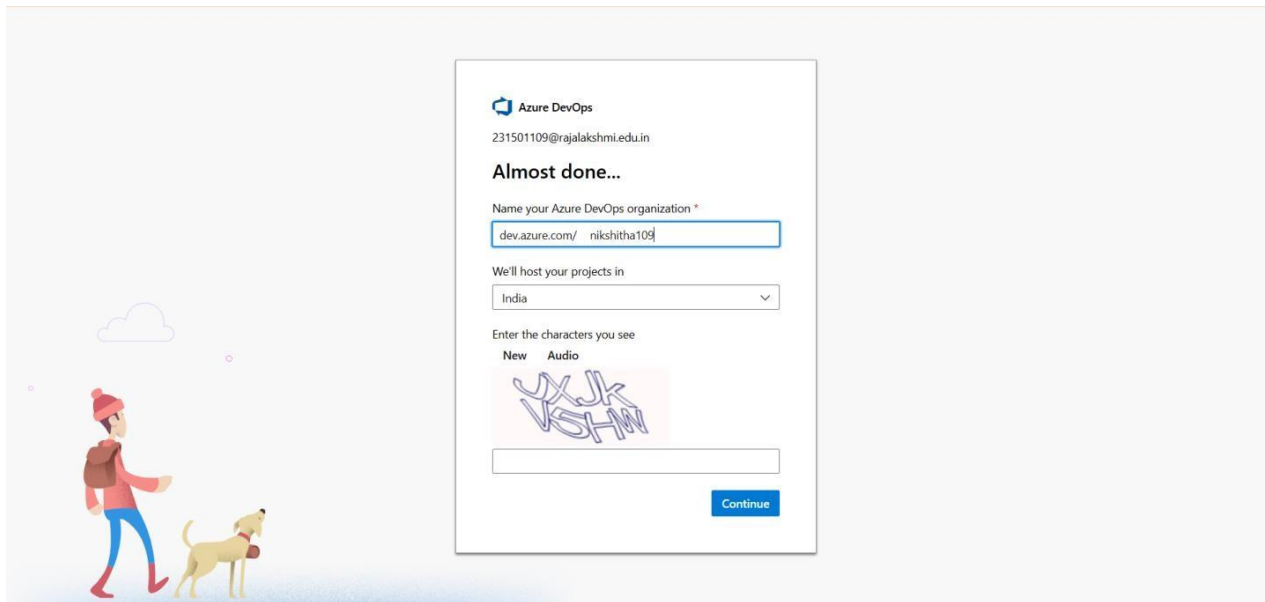
Date :

## AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

### 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

Project name \*

E-Commerce Product Uploader

The project 'E-Commerce Product Uploader' already exists.

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

3. 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.

2116231501135

6

CS23432

Microsoft

Rino Calvin Sign out

RC

Rino Calvin

rinoalvin070706@gmail.com

India

rinoalvin070706@gmail.com

Edit profile

Visual Studio Dev Essentials

Get everything you need to build and deploy your app on any platform.

Use your benefits

Azure DevOps Organizations

Create new organization

dev.azure.com/rinocalvin (Owner)

Projects

new project try

E-commerce Product uploader

atm TICKET REG

New project

Actions

Open in Visual Studio

dev.azure.com/rinocalvin0707 (Owner)

dev.azure.com/rinocalvin070706 (Owner)

dev.azure.com/rinocalvin0707060813 (Owner)

dev.azure.com/231501122 (Member)

## 4. Project dashboard

Azure DevOps

231501122 / Ecommerce / Overview / Summary

Search

Private Invite

Ecommerce

Overview

Summary

Dashboards

Wiki

Boards

Pipelines

Artifacts

Ecommerce

About this project

The E-commerce Product Uploader is a web-based application designed to help store owners efficiently manage bulk product uploads using CSV or Excel files. The system allows uploading essential product details such as name, description, price, and images, with built-in validation rules to ensure data accuracy, image integrity, and compliance with ethical standards.

The project incorporates robust backend and frontend components, supporting full CRUD operations, product search, and paginated views for enhanced usability. Images are locally verified to prevent duplication across products and ensure only appropriate content is uploaded—excluding external links or unethical visuals.

The application is integrated into Azure DevOps for streamlined development, testing, and deployment. Within Azure, detailed test plans, test suites, and test cases have been created to verify system behavior under both normal (happy path) and error (negative) conditions. This enables automated and manual test execution, bug tracking, and status monitoring for high-quality delivery.

This project serves as a complete example of how to manage a full-stack software system lifecycle using Azure DevOps, emphasizing collaborative development, test-driven delivery, and continuous quality assurance.

Project stats

Period: Last 7 days

Boards

29 Work items created

0 Work items completed

Repos

0 Pull requests opened

11 Commits by 1 authors

Pipelines

0% Builds succeeded

Members

6

## 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 top screenshot shows the Azure DevOps interface for the 'Ecommerce Team' project. The left sidebar contains navigation options: Overview, Boards, Work items, Boards, Backlogs, Sprints, Queries, Delivery Plans, Pipelines, and Artifacts. The main area displays the 'Backlog' view with a table of work items. The table has columns for Order, Work Item Type, Title, State, Effort, Business Area, Value Area, and Tags. The work items include tasks like 'Create a search bar with debounced input', 'Implement backend API to search by name or key...', 'Add filter options (e.g., price range)', and 'Update query logic in backend accordingly'. There are also user stories like 'As a user, I want to filter products by price or description...' and epics like 'product management' and 'bulk product upload'.

The bottom screenshot shows the same interface but with a simplified backlog view. The table now only displays three items, all of which are 'Epic' type work items: 'Search and Pagination', 'product management', and 'bulk product upload'. The 'New Work Item' button is visible at the top right of the main area.

**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, Ecommerce Product Uploader .

### 1.Create Epic, Features, User Stories, Task

The screenshot shows the Azure DevOps interface for the 'Ecommerce' project. The left sidebar contains navigation links: Overview, Boards, Work items, Boards, Backlogs (selected), Sprints, Queries, Delivery Plans, Pipelines, and Artifacts. The main area displays the 'Ecommerce Team' backlog. A notification at the top states: 'As a stakeholder, you can access the backlog, task and Kanban boards, work items and manage approvals for Releases. [Learn more](#)'. The backlog is titled 'Ecommerce Team' and includes a '+ New Work Item' button and 'View as Board' and 'Column Options' links. The 'Backlog' tab is active, showing a list of work items with columns for Order, Work Item Type, Title, State, Effort, Business Area, Value Area, and Tags. The work items are grouped by order, with items 2 and 3 expanded to show their sub-items.

Order	Work Item Type	Title	State	Effort	Busin...	Value Area	Tags
	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				
2	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				
3	Epic	bulk product upload	New		Business		
	Feature	CSV/Excel File Upload	New		Business		

## 2. Fill in Epics

Azure DevOps 231501122 / Ecommerce / Boards / Backlogs

Search

Ecommerce

Overview

Boards

Work items

Boards

Backlogs

Sprints

Queries

Delivery Plans

Pipelines

Artifacts

Project settings

As a stakeholder, you can access the backlog, task and Kanban boards, work items and manage approvals for Releases. [Learn more](#)

Ecommerce Team

New Work Item

View as Board

Column Options

Bulk Product Upload

Add to top

	Order	Work Item Type	Title	State	Effort	Busin...	Value Area	Tags
1		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 descri...	New			Business	
		Task	Add filter options (e.g., price range)	New				
		Task	Update query logic in backend accordingly	New				
2		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	

## 3. Fill in Features

NEW FEATURE \*

File upload and selection

No one selected

0 Comments

Add Tag

Save and Close

State: New

Area: Batch data analysis and visualization

Reason: New

Iteration: Batch data analysis and visualization\Sprint 1

Details

Description

Click to add Description.

Discussion

Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request.

switch to Markdown editor

Planning

Priority: 2

Risk

Effort

Business Value

Time Criticality

Start Date

Select a date...

Target Date

Select a date...

Classification

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

## 4. Fill in User Stories

USER STORY 3

3 As a user, I want to be able to upload multiple CSV files at once, so I can analyze them together.

No one selected0 CommentsAdd TagSave and CloseFollowDetails

StateNewAreaBatch data analysis and visualizationReasonNewIterationBatch data analysis and visualization\Sprint 1Updated by Shri Dharshini: Mar 27

Description

Click to add Description.

Acceptance Criteria

Click to add Acceptance Criteria.

Discussion

Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request.

switch to Markdown editor

Planning

Story Points

Priority2

Risk

Classification

Value area

Business

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

**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 displays the Azure DevOps Sprints interface for the 'Ecommerce Team'. The left sidebar shows the navigation menu with 'Sprints' selected. The main area shows the 'Taskboard' for 'sprint 1' with columns for 'New', 'Active', 'Resolved', and 'Closed'. Two user stories are visible in the 'New' column:

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

The interface also includes a search bar, a 'New Work Item' button, and a 'Column Options' menu. The sprint duration is shown as 'June 3 - June 9' with '5 work days'.

## Sprint 2

The screenshot shows the Azure DevOps interface for the 'Ecommerce' project. The left sidebar contains navigation links: Overview, Boards, Work items, Boards, Backlogs, Sprints (selected), Queries, Delivery Plans, Pipelines, and Artifacts. The main area displays the 'Sprints' board for 'Sprint2' (May 13 - May 19, 5 work days). The board is organized into columns: New, Active, Resolved, and Closed. There are four work items in the 'New' column, all assigned to 'Unassigned'. The work items are: 23 (As a store owner, I want to upload a CSV/Excel file so that I can add multiple products at once.), 24 (Implement file upload UI and backend endpoint.), 25 (Parse CSV/Excel file and extract product fields.), and 26 (As a developer, I want to validate the file structure to ensure all required columns are present.).

## Sprint 3

The screenshot shows the Azure DevOps interface for the 'Ecommerce' project. The left sidebar contains navigation links: Overview, Boards, Work items, Boards, Backlogs, Sprints (selected), Queries, Delivery Plans, Pipelines, and Artifacts. The main area displays the 'Sprints' board for 'sprint3' (May 20 - May 26, 4 work days remaining). The board is organized into columns: New, Active, Resolved, and Closed. There are two work items in the 'New' column, both assigned to 'Rino Calvin'. The work items are: 80 (As a developer, I want to validate the file structure to ensure all required columns are present.) and 77 (As a store owner, I want to search products by name so that I can find specific items quickly.).

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

**EXP NO: 5**

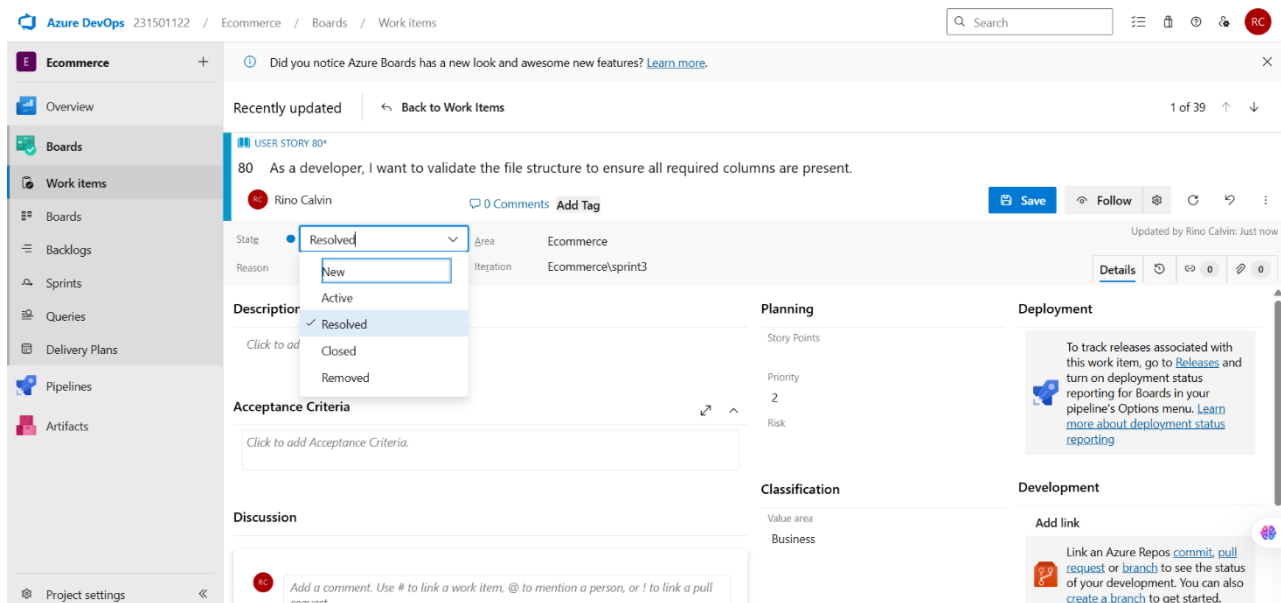
**Date :**

# POKER ESTIMATION

## Aim:

Create Poker Estimation for the user stories for the project, Batch Data Analysis and Visualization.

## Poker Estimation



## Result:

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

EXP NO: 6

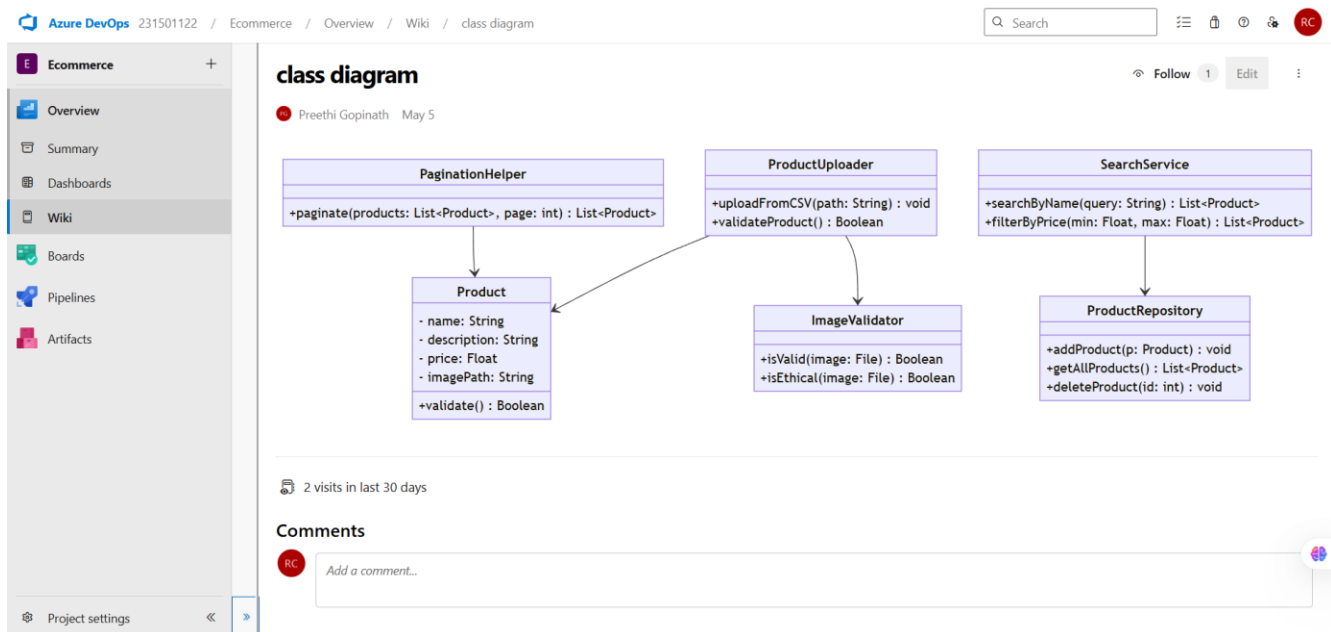
Date :

## DESIGNING CLASS DIAGRAM AND SEQUENCE DIAGRAM

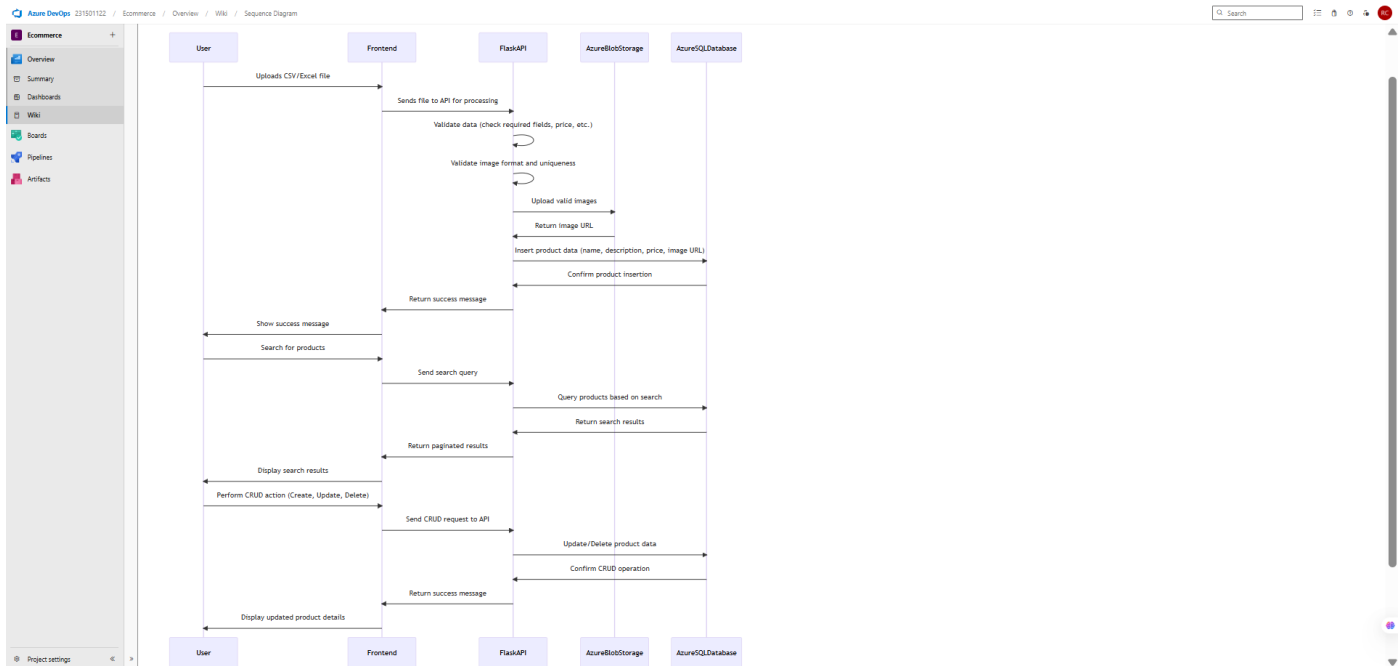
### Aim:

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

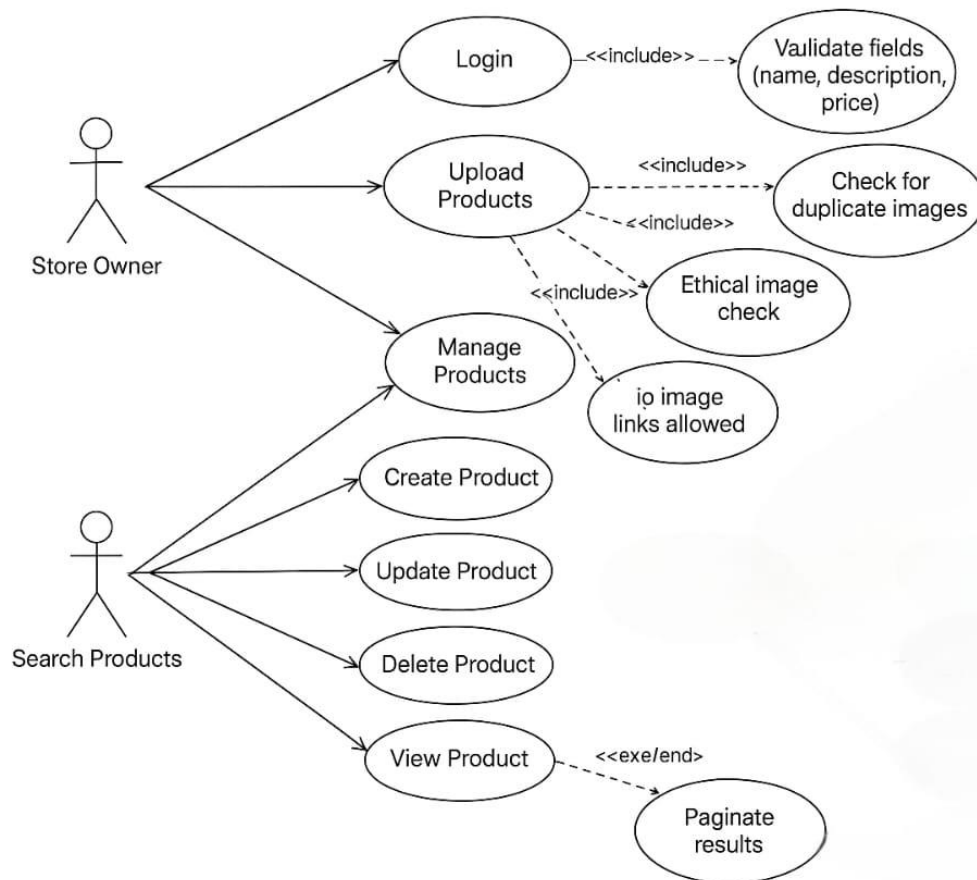
## DESIGNING USE CASE DIAGRAM AND ACTIVITY DIAGRAM

Date :

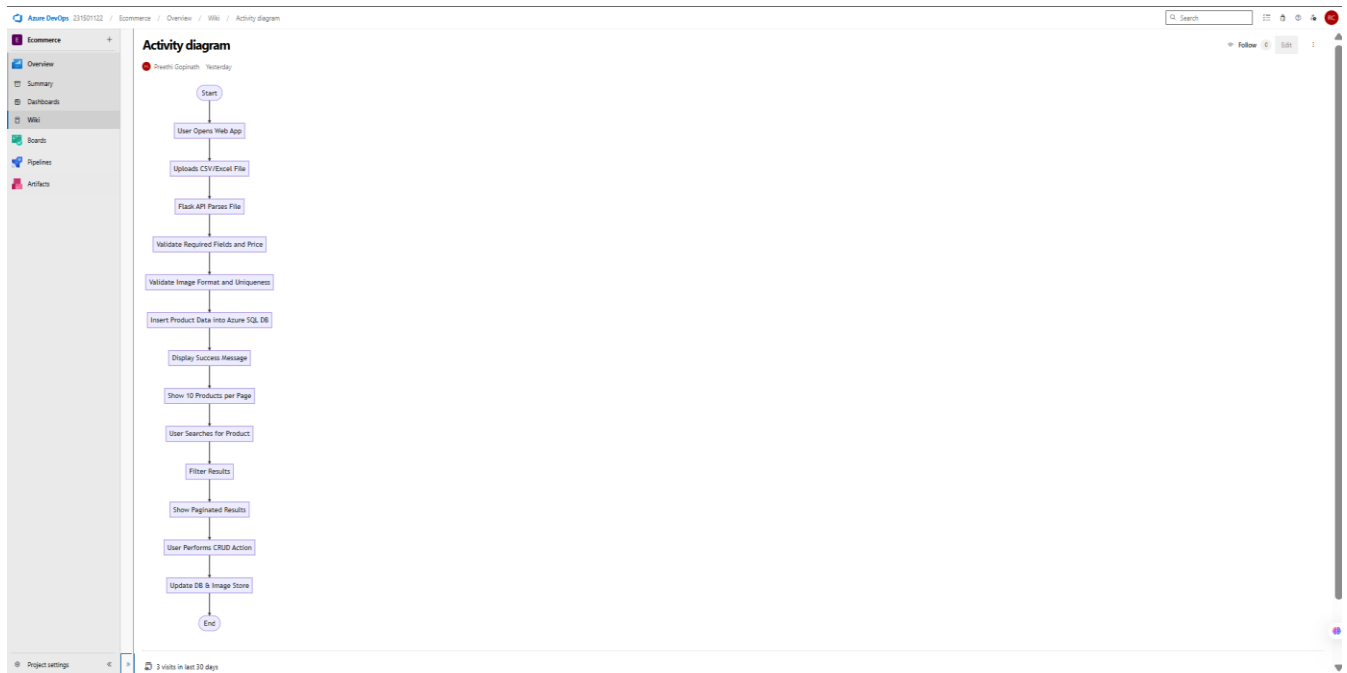
### Aim:

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 .

<b>EXP NO: 8</b>	<b>TESTING – TEST PLANS AND TEST CASES</b>
<b>Date :</b>	

**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

nikshithaharikrishnan2005 / Batch Data Analysis and Vis... / Test Plans

Search

New Test Plan

Name \*

login

Area Path \*

Batch Data Analysis and Visualization

Iteration \*

Batch Data Analysis and Visualization/Iteration 1

Create Cancel

## 2. Test suite

nikshithaharikrishnan2005 / Batch Data Analysis and Vis... / Test Plans / login page check

Search

login page check (ID: 23)

Define Execute Chart

Test Suites

Filter suites by name

login page check (2)

dashboard check (2)

index page check (2)

Test Cases (2 items)

Title	Order	Test Case Id
if i click login page it should redirect to dashboard page	3	26
if i don't enter my password then it should say error	4	27

## 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

nikshithaharikrishnan2005 / Batch Data Analysis and Vis... / Test Plans / BatchDataAnalysis

Search

BatchDataAnalysis

May 17 - May 24

Current

100% run, 100% passed. View report

Test Suites

Filter suites by name

BatchDataAnalysis

TS02- Logout Functionality (2)

TS02- Logout Functionality (ID: 47)

Define Execute Chart

Test Cases (2 items)

New Test Case

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

nikshithaharikrishnan2005 / Batch Data Analysis and Vis... / Test Plans / User Authentication

Search

batch data analysis

May 17 - May 24

Current

Test Suites

Filter suites by name

batch data analysis

TS01- User Authentication (2)

TS02- Logout Functionality (2)

TS01- User Authentication (ID: 54)

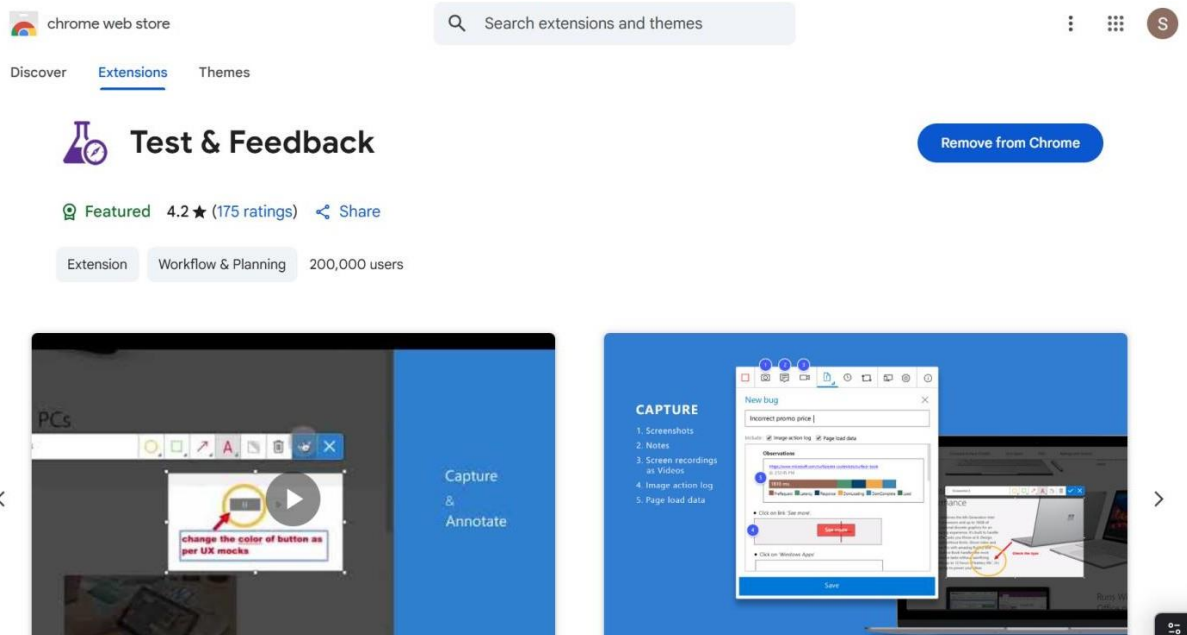
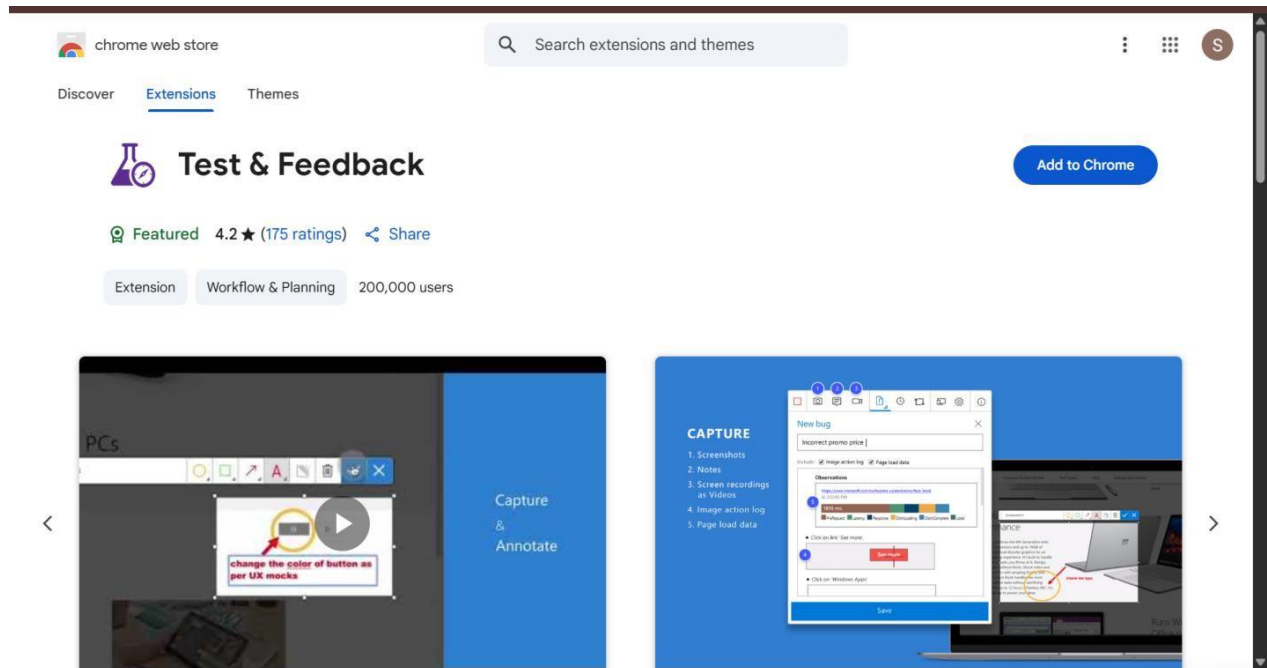
Define Execute Chart

Test Points (2 items)

Run for web application

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

## 4. Installation of test





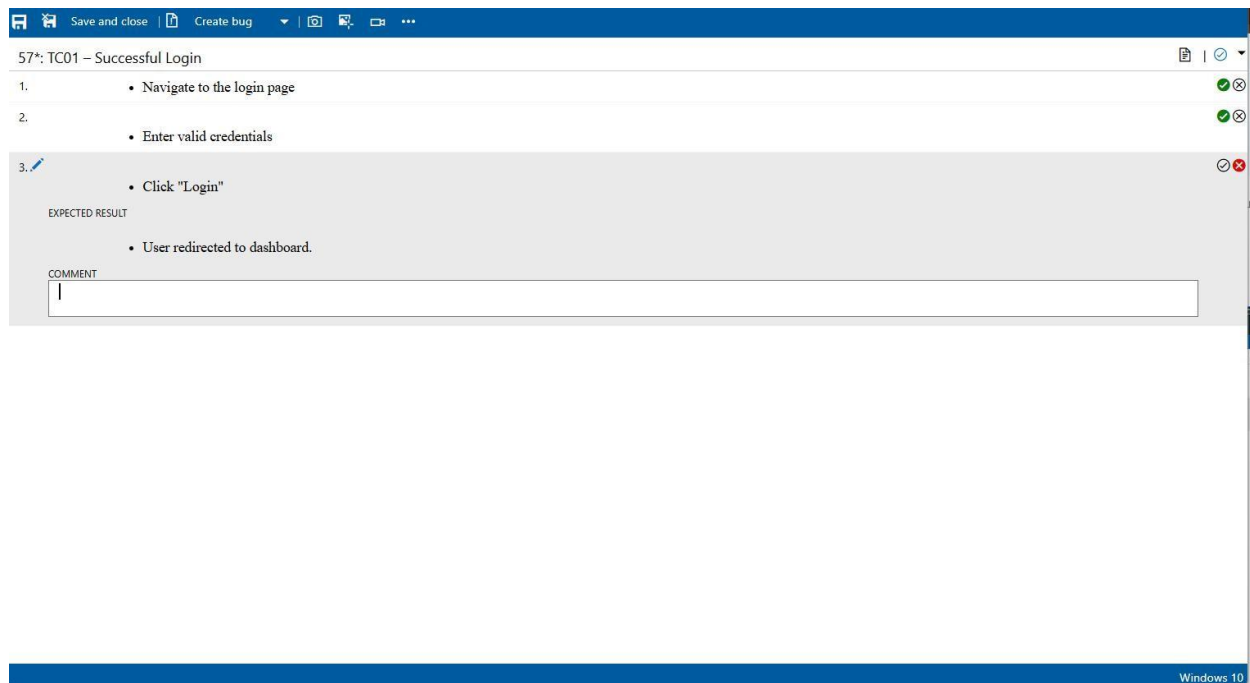
## Test and feedback

### Showing it as an extension

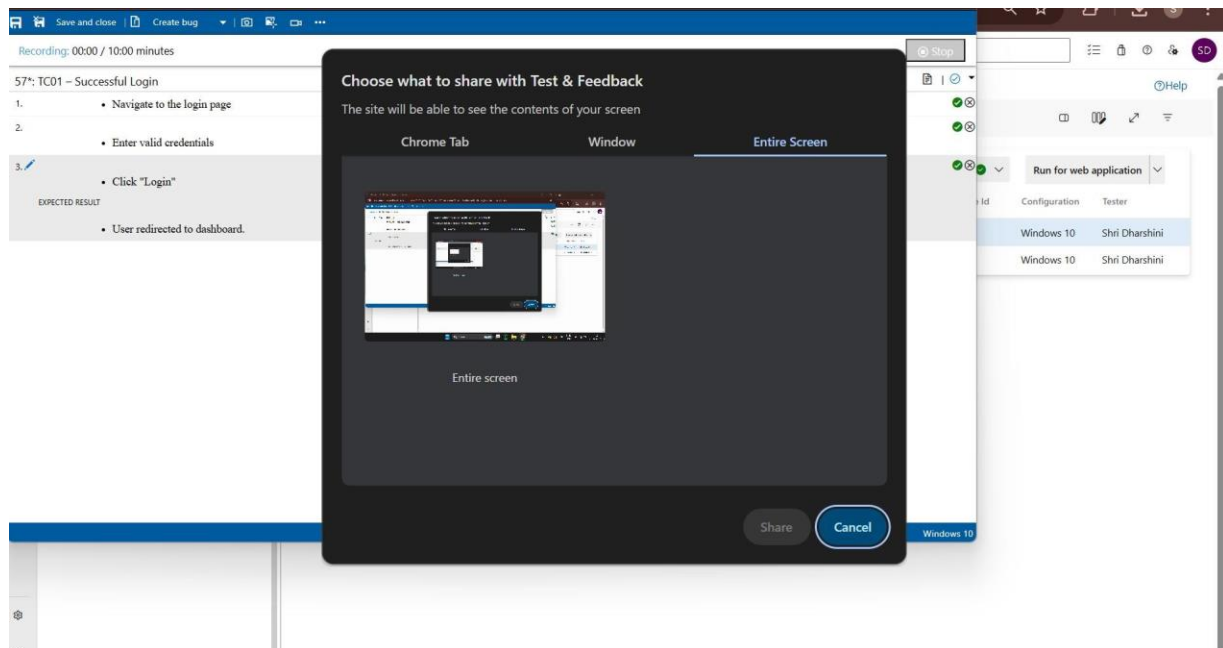
The screenshot shows the Visual Studio Code interface with the 'Test & Feedback' extension installed. The extension is visible in the 'Extensions' sidebar on the left. The main window displays the 'TS01- User Authentication (ID: 54)' test suite. The 'Test Points (2 items)' table shows two test cases: 'TC01 - Successful Login' (Passed) and 'TC02 - Prevent Login with Empty Fields' (Failed). The 'Outcome' column shows 'Passed' for TC01 and 'Failed' for TC02. The 'Order' column shows 1 for TC01 and 2 for TC02. The 'Test Case Id' column shows 57 for TC01 and 58 for TC02. The 'Configuration' column shows 'Windows 10' for both. The 'Tester' column shows 'Shri Dharshini' for both. The 'Test & Feedback' extension is highlighted in the sidebar, and its 'Manage extensions' button is visible in the top right corner.

## 5. Running the test cases

The screenshot shows the Visual Studio Code interface with the 'Test & Feedback' extension installed. The extension is visible in the 'Extensions' sidebar on the left. The main window displays the 'TS01- User Authentication (ID: 54)' test suite. The 'Test Points (2 items)' table shows two test cases: 'TC01 - Successful Login' (Passed) and 'TC02 - Prevent Login with Empty Fields' (Failed). The 'Outcome' column shows 'Passed' for TC01 and 'Failed' for TC02. The 'Order' column shows 1 for TC01 and 2 for TC02. The 'Test Case Id' column shows 57 for TC01 and 58 for TC02. The 'Configuration' column shows 'Windows 10' for both. The 'Tester' column shows 'Shri Dharshini' for both. The 'Run' button is visible in the top right corner, and the 'Run for web application' dropdown menu is open, showing options: 'Run for web application', 'Run for desktop application', and 'Run with options'.



## 6. Recording the test case



## 7. Creating the bug

Save and close | Create bug

58: TC02 – Prevent Login with Empty Fields

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.

Windows 10

Save and close | Create bug

58\*: TC02 – Prevent Login with Empty Fields

1. NEW BUG \*

login failed

2. Unassigned 0 comments Add tag Save & Close

3.

State: ● New Area: Batch Data Analysis and Visualization

Reason: New Iteration: Batch Data Analysis and Visualization

Details (2)

**Repro Steps**

17/05/2025 10:44 Bug filed on "TC02 – Prevent Login with Empty Fields"

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

**Planning**

Resolved Reason

Story Points

Priority: 2

Severity: 3 - Medium

Activity

**Effort (Hours)**

Original Estimate

**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.

Windows 10

BUG 60

60 not logging in due to system error

No one selected0 CommentsAdd Tag

Save and CloseFollow

Updated by Shri Dharshini: Just now

Details30

StateNew

AreaBatch Data Analysis and Visualization

ReasonNew

IterationBatch 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	
2.	Passed	<ul style="list-style-type: none"> <li>Navigate to the login page.</li> </ul>
3.	Failed	<ul style="list-style-type: none"> <li>Leave username and/or password fields empty.</li> <li>Click on "Login".</li> </ul>

Expected Result

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

Planning

Resolved Reason

Story Points

Priority2

Severity3 - Medium

Activity

Effort (Hours)

Original Estimate

Remaining

Completed

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

S8 TC02 – Prevent Login with Empty Fields

## 8. Test case results

batch data analysis

May 17 - May 24

50% run, 50% passed. View report

Test Suites

Filter suites by name

batch data analysis

TS01- User Authentication (2)

TS02- Logout Functionality (2)

TS01- User Authentication (ID: 54)

Define Execute Chart

Test Points (2 items)

Title

☒ TC01 – Successful Login
 ☐ 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 data analysis
Failed	17m ago	Windows 10	Shri Dharshini	Shri Dharshini	batch data analysis
Passed	28m ago	Windows 10	Shri Dharshini	Shri Dharshini	batch data analysis

Open execution history for current test point

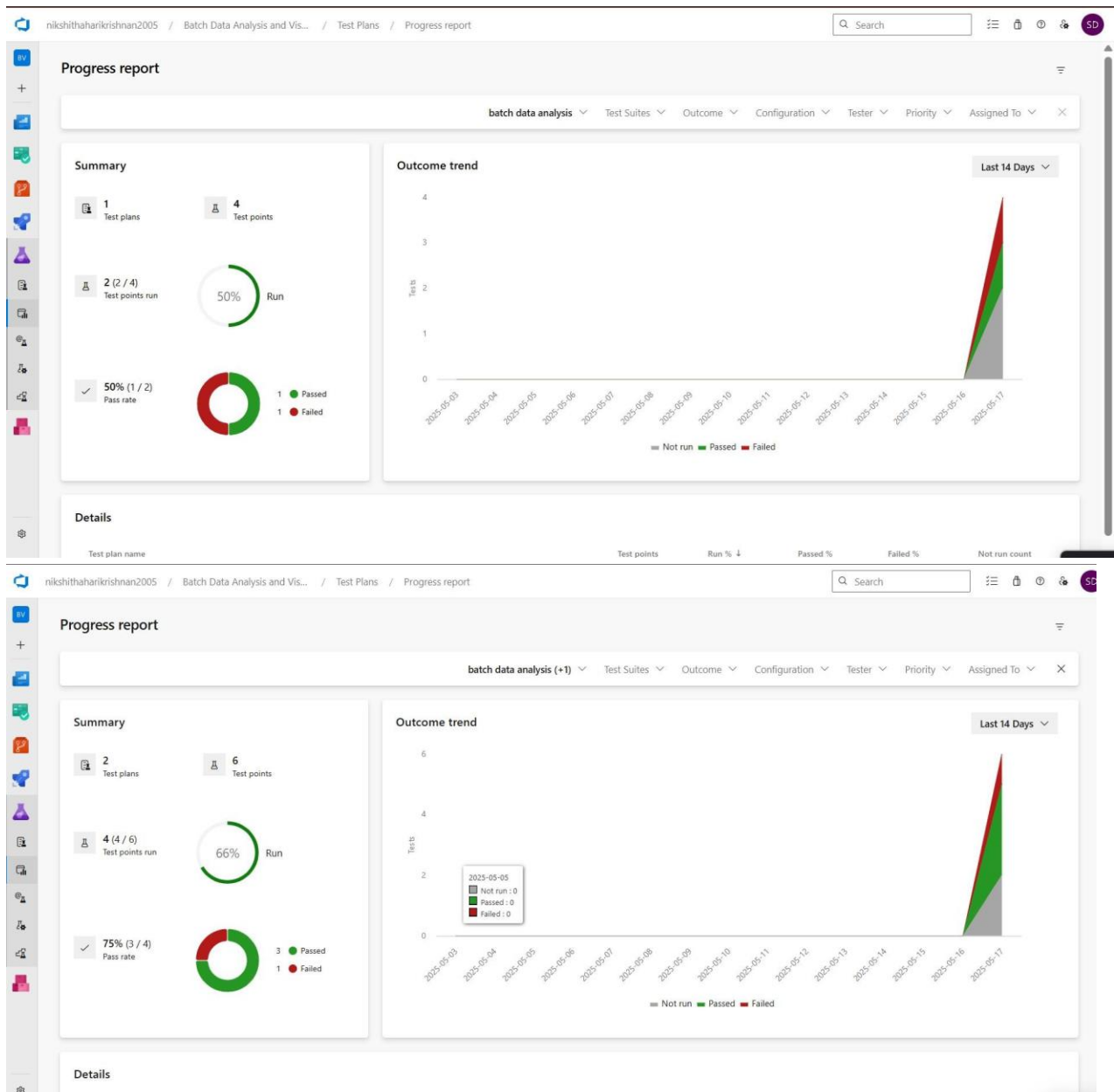
## 9. Test report summary

The screenshot displays an Azure Boards work item interface. At the top, a navigation bar shows the user 'nikshithaharikrishnan2005' and the path 'Batch Data Analysis and Vis... / Boards / Work items'. A search bar and a notification banner are also present. The main header area includes 'Recently updated' and 'Back to Work Items' links, along with a '1 of 47' indicator. The work item itself is titled '60 not logging in due to system error' and is categorized as a 'BUG'. It has '0 Comments' and an 'Add Tag' button. The 'Status' is 'New', and the 'Area' and 'Reason' are both 'Batch Data Analysis and Visualization'. The 'Repro Steps' section lists three steps: 1. 'Passed', 2. 'Passed', and 3. 'Failed'. The 'Expected Result' is 'Validation error message is shown prompting user to fill required fields.' The 'Planning' section shows 'Resolved Reason', 'Story Points', 'Priority: 2', 'Severity: 3 - Medium', and 'Activity'. The 'Effort (Hours)' section shows 'Original Estimate' and 'Remaining'. The 'Deployment' section includes a link to 'Releases' and a 'Development' section with an 'Add link' button. The 'Related Work' section also has an 'Add link' button and a link to 'Add an existing work item as a parent'.

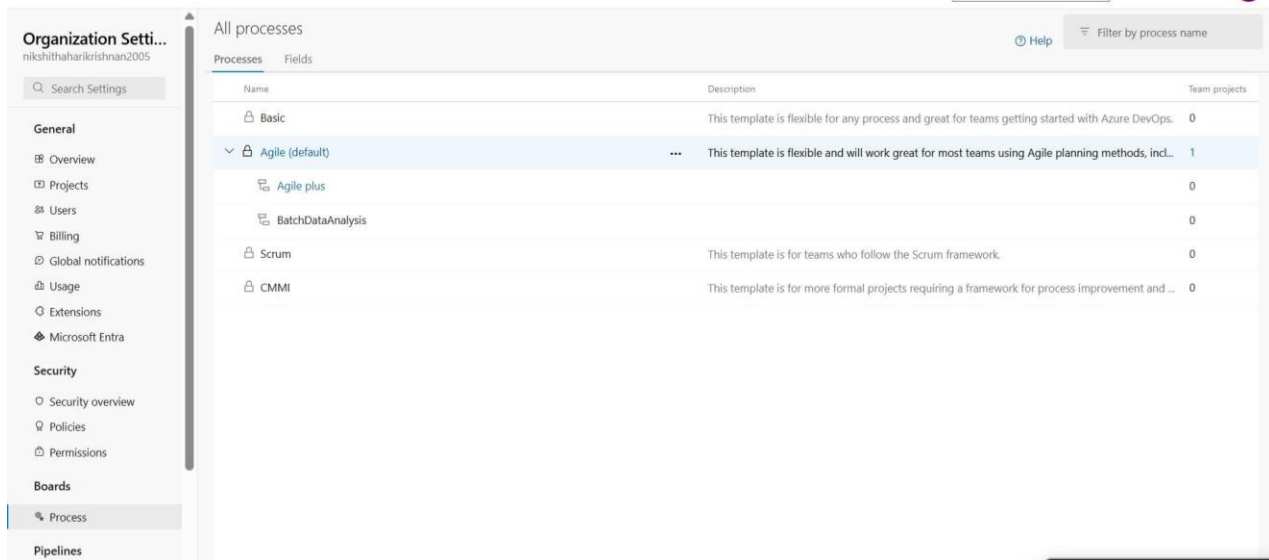
- Assigning bug to the developer and changing state

This screenshot is identical to the one above, but the 'Status' of the work item is now 'New' instead of 'BUG'. The 'Repro Steps' section remains the same, with three steps: 1. 'Passed', 2. 'Passed', and 3. 'Failed'. The 'Expected Result' is 'Validation error message is shown prompting user to fill required fields.' The 'Planning' section shows 'Resolved Reason', 'Story Points', 'Priority: 2', 'Severity: 3 - Medium', and 'Activity'. The 'Effort (Hours)' section shows 'Original Estimate' and 'Remaining'. The 'Deployment' section includes a link to 'Releases' and a 'Development' section with an 'Add link' button. The 'Related Work' section also has an 'Add link' button and a link to 'Add an existing work item as a parent'.

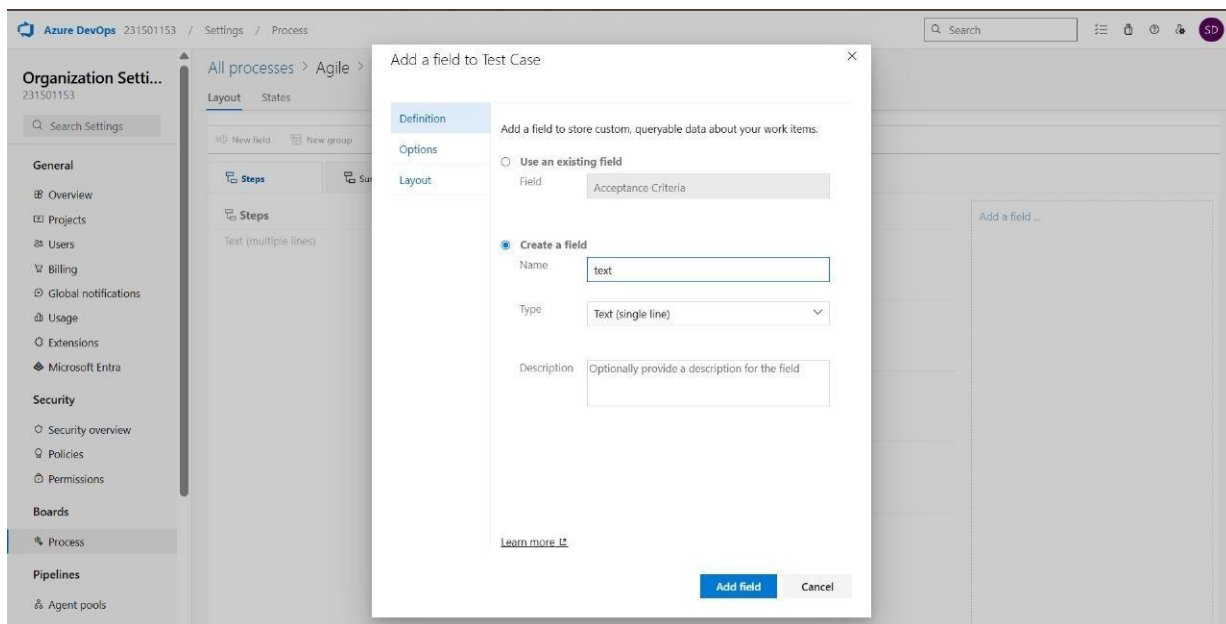
## 10. Progress report

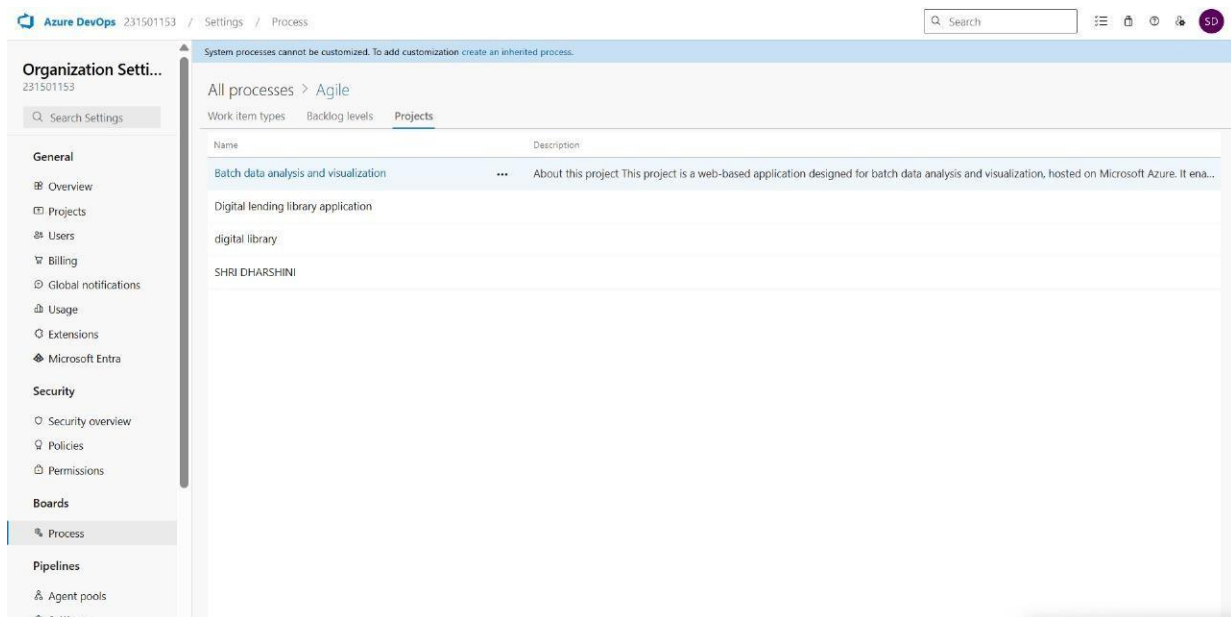
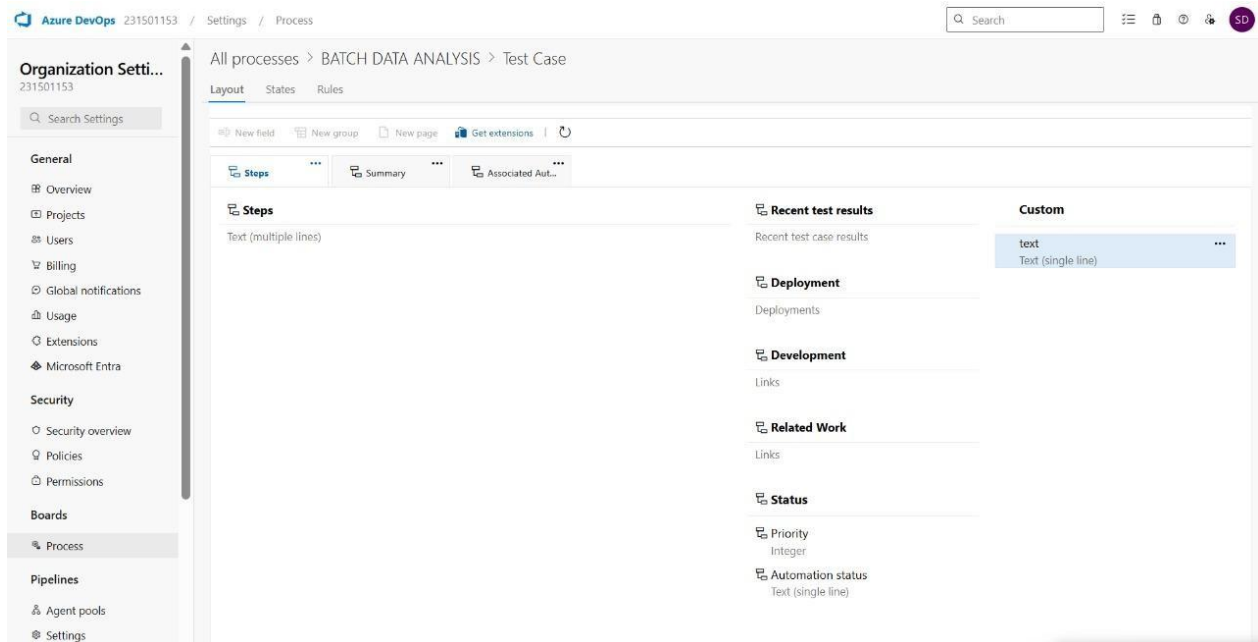


## 11. Changing the test template



## 12. View the new test case template





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



<b>EXP NO: 9</b>	<b>CI/CD PIPELINES IN AZURE</b>
<b>Date:</b>	

**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](https://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

Azure DevOps

231501122

Ecommerce

Pipelines

Ecommerce (4)

20250520.1

Search

PR

Ecommerce

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Library

Test Plans

Artifacts

Project settings

#20250520.1 • Set up CI with Azure Pipelines

Run new

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

Summary

Code Coverage

Individual CI by Prathisha R

3 commits

Repository and version

Time started and elapsed

Related

Tests and coverage

Ecommerce

Today at 7:00 PM

0 work items

Get started

main

93ebaf40

20s

0 artifacts

Jobs

Name

Status

Duration

Job

Success

12s

Jobs

Name

Status

Duration

## Result:

Successfully demonstrated pipelines in azure devops

2116231501135

35

CS23432

**EXP NO: 10**

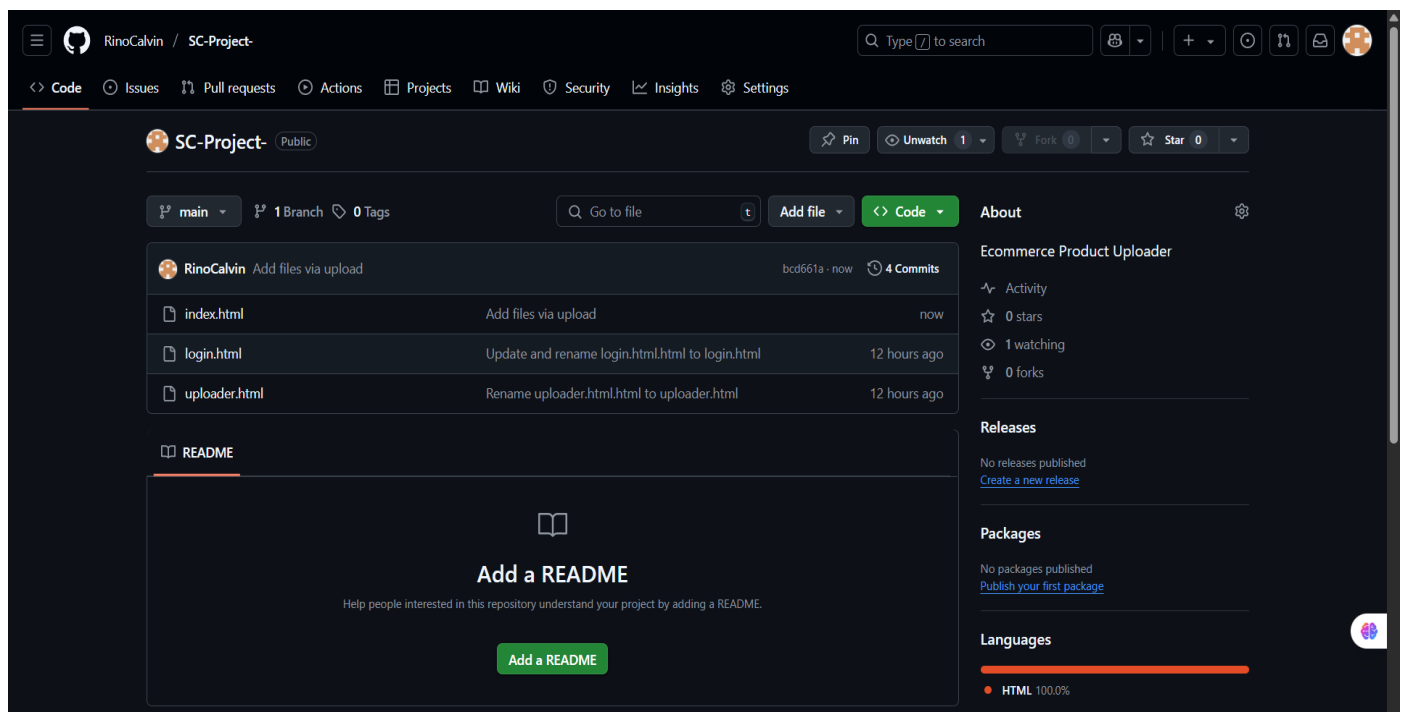
**Date :**

## **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 project.

### **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.