

EXP NO: 1	<b>AZURE DEVOPS ENVIRONMENT SETUP</b>
-----------	---------------------------------------

### Aim:

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

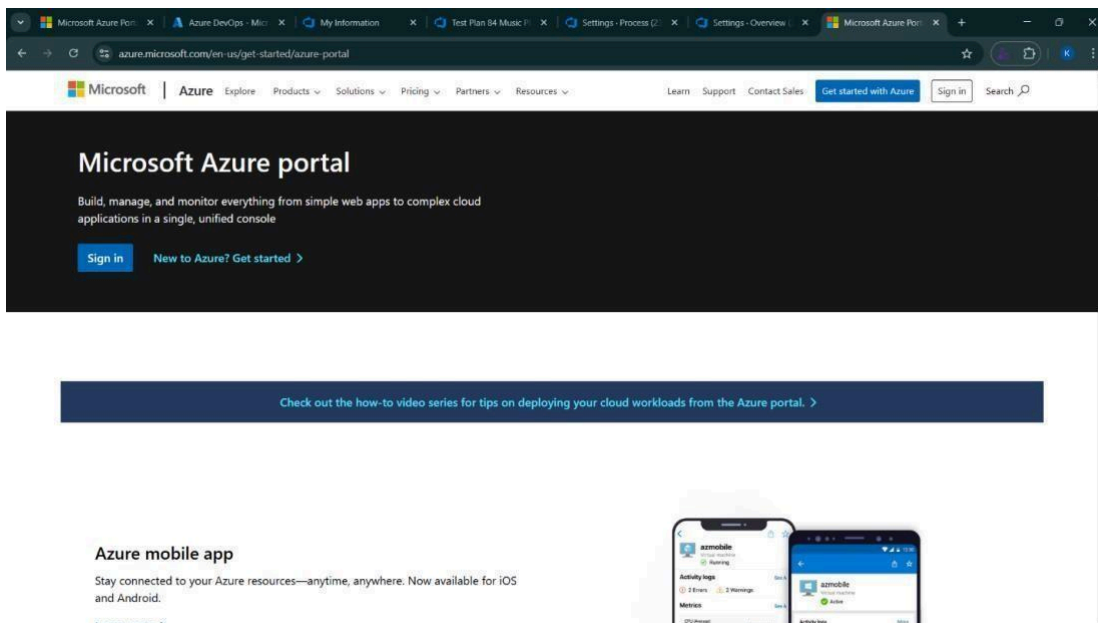
### INSTALLATION

1. Open your web browser and go to the Azure website:

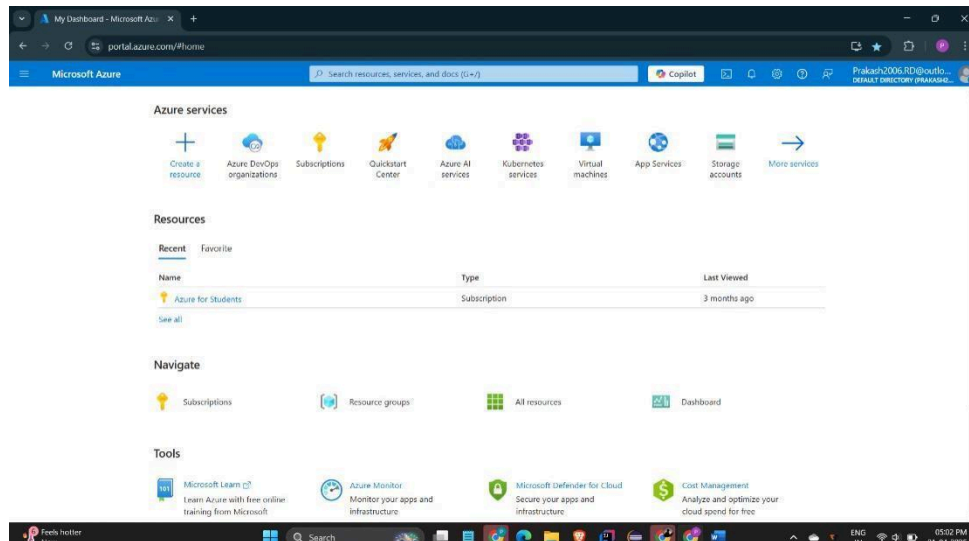
<https://azure.microsoft.com/en-us/getstarted/azureportal>.

Sign in using your Microsoft account credentials.

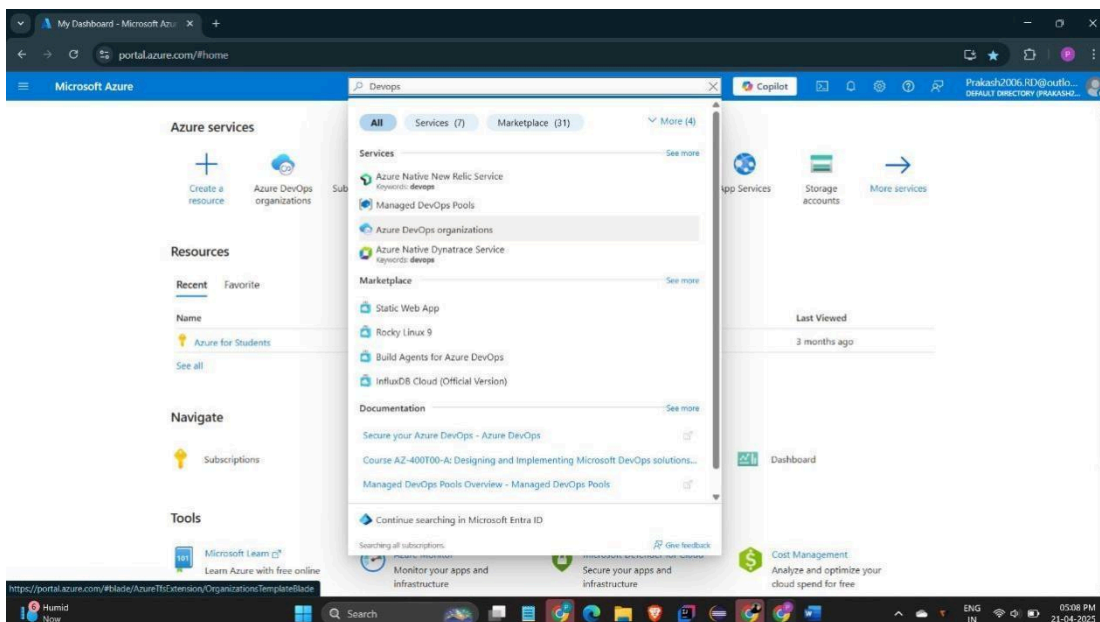
If you don't have a Microsoft account, you can create one here: <https://signup.live.com/?lic=1>



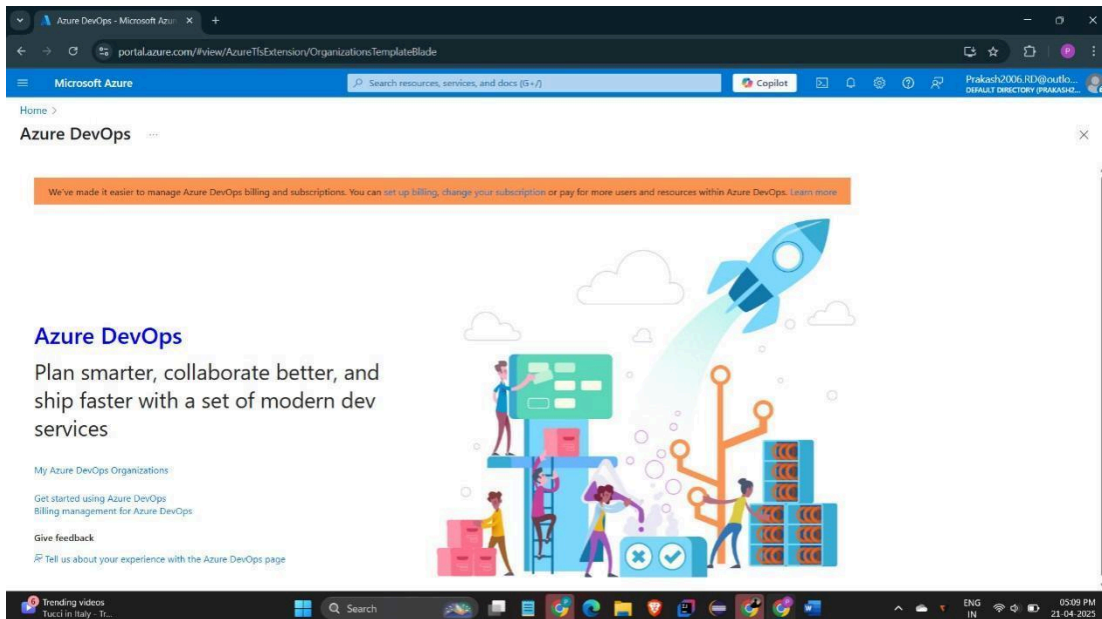
## 2. Azure home page



3. Open 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.

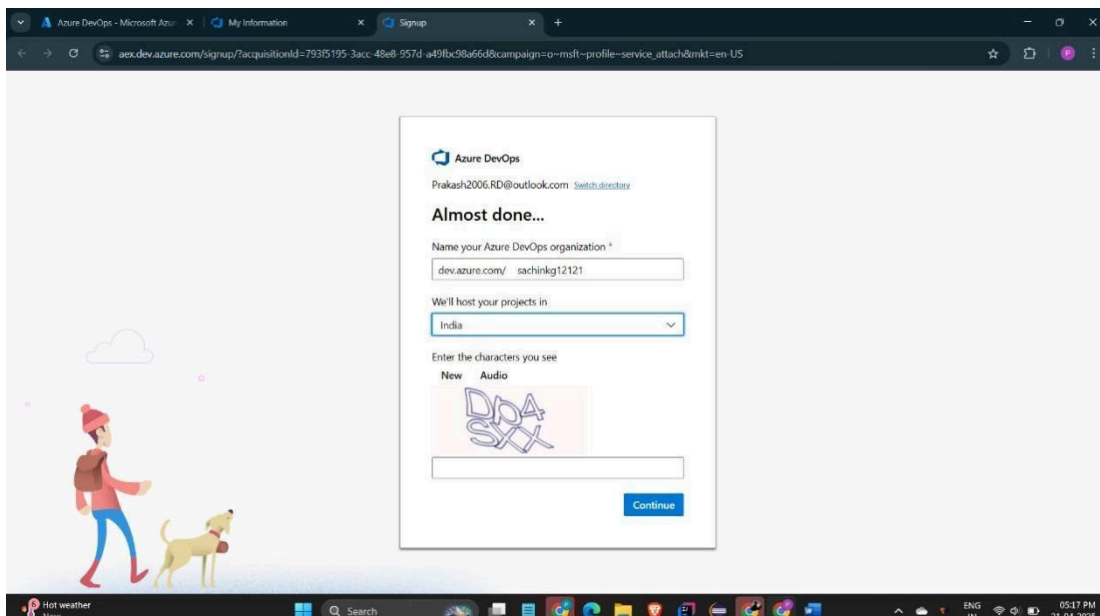
EXP NO: 2

# AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

## Aim:

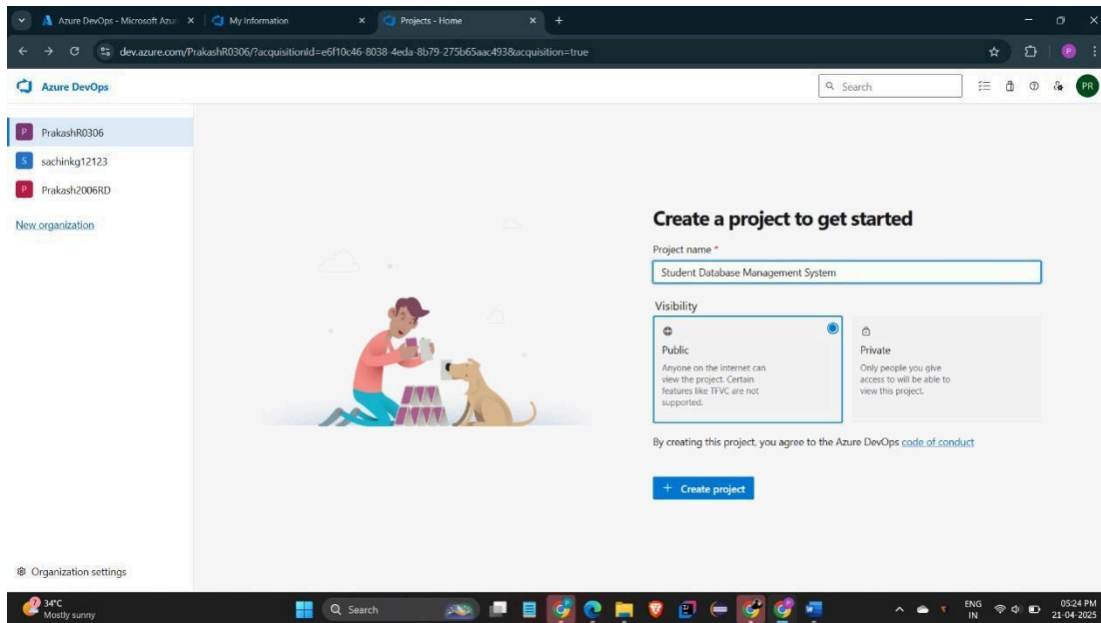
To set up an Azure DevOps project for efficient collaboration and agile work management.

## 1.Create An Azure Account

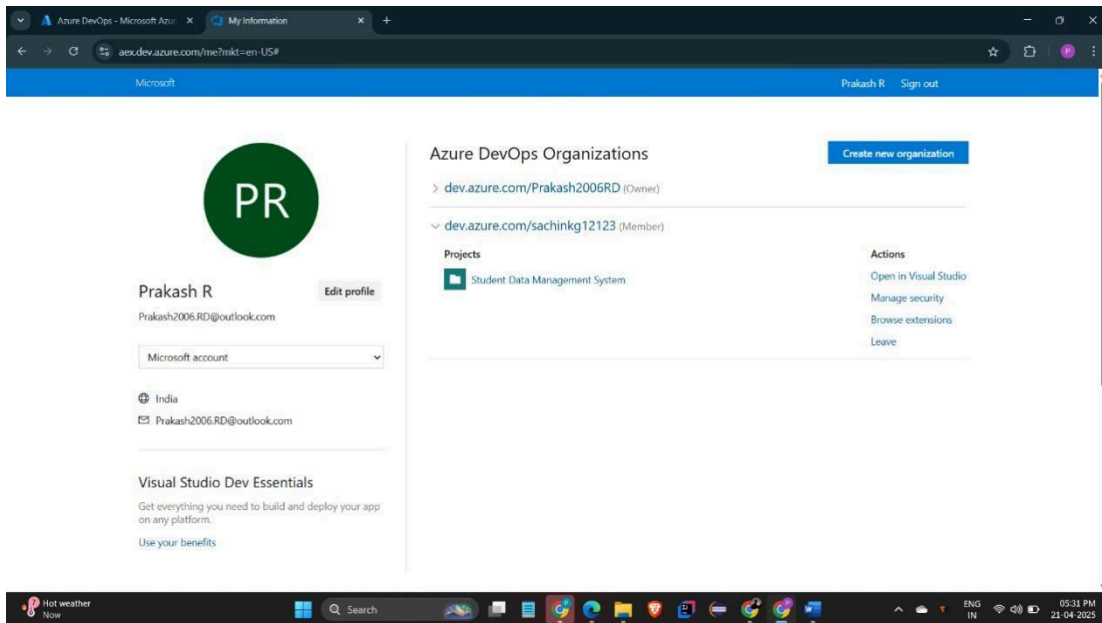


## 2.Create the First Project in Your Organization

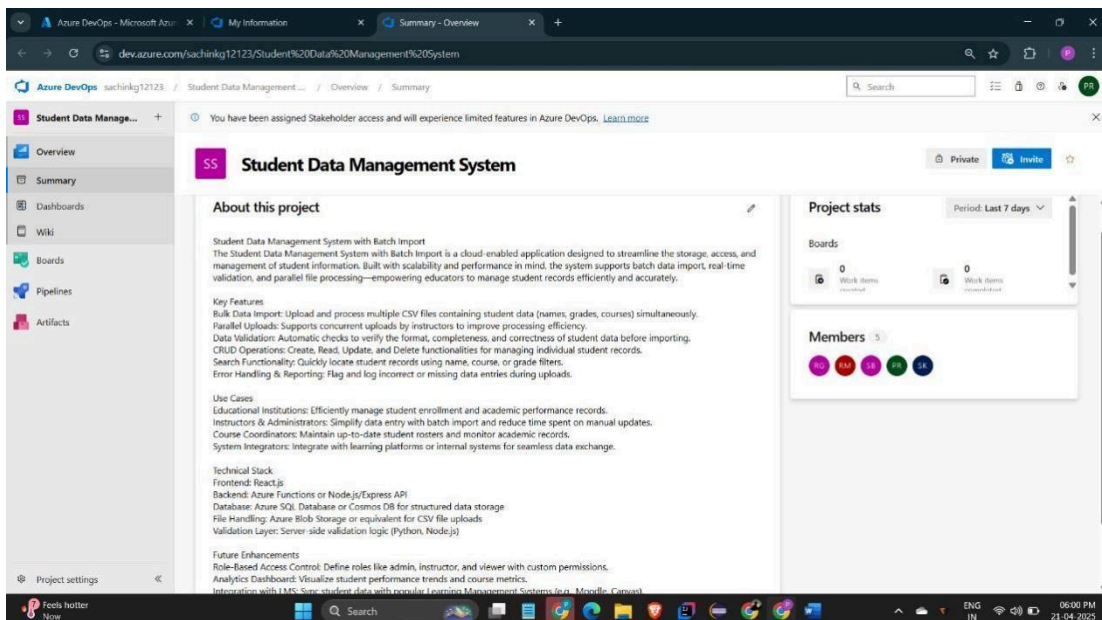
- After the organization is set up, you'll need to create your first **project**. This is where you'll begin to manage code, pipelines, work items, and more.
- On the organization's **Home page**, click on the **New Project** button.
- Enter the project name, description, and visibility options:
  -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



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.



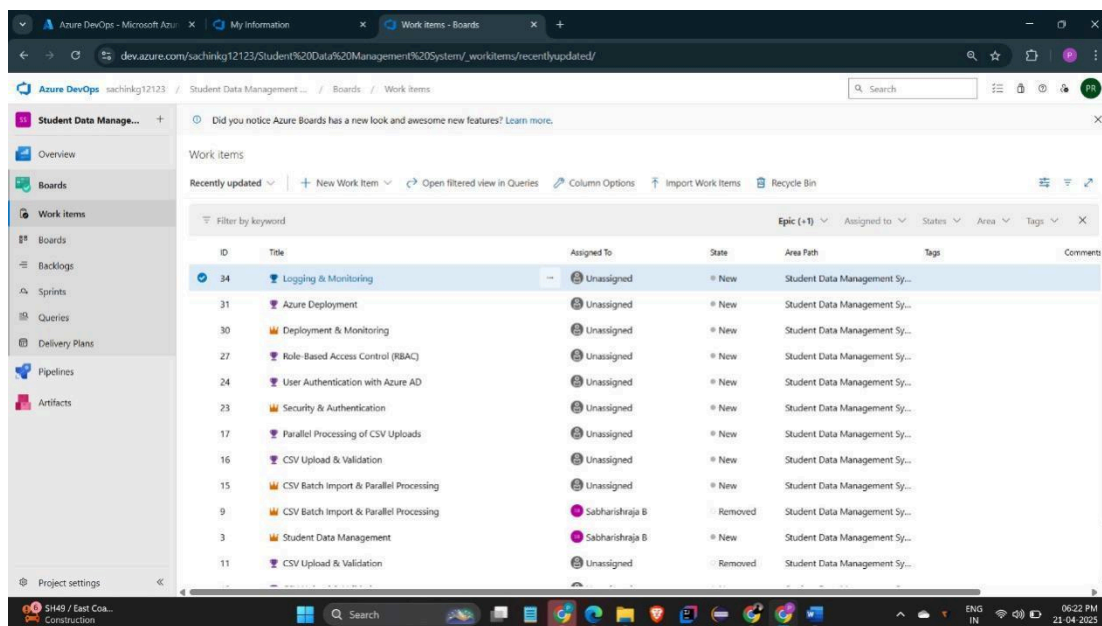
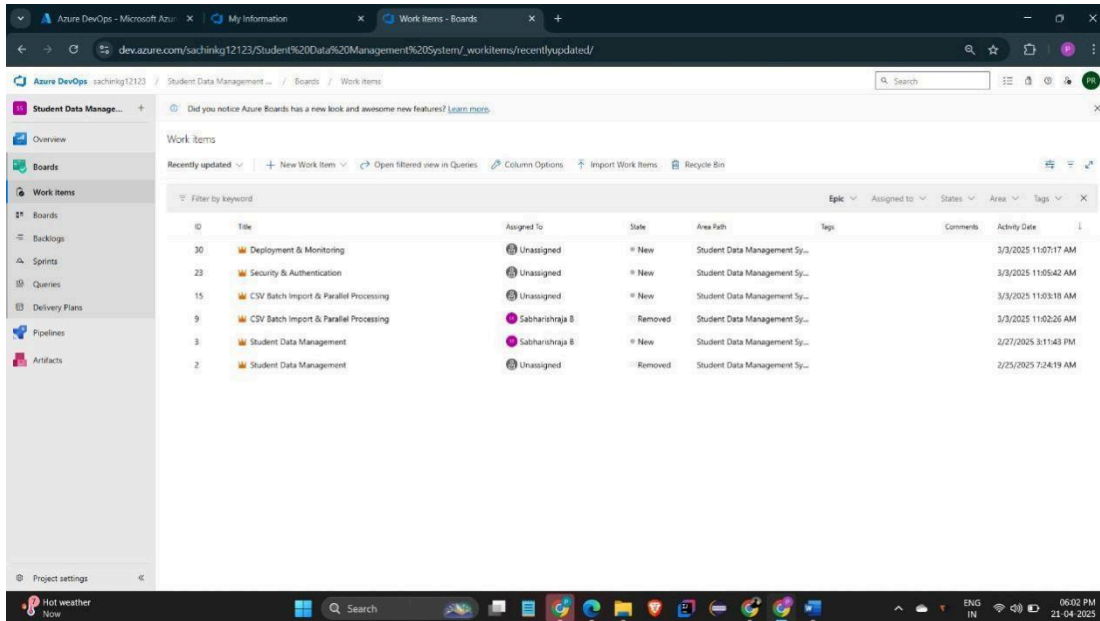
#### 4. Project dashboard



## 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.



**Result:**

2116231801144

CS23432



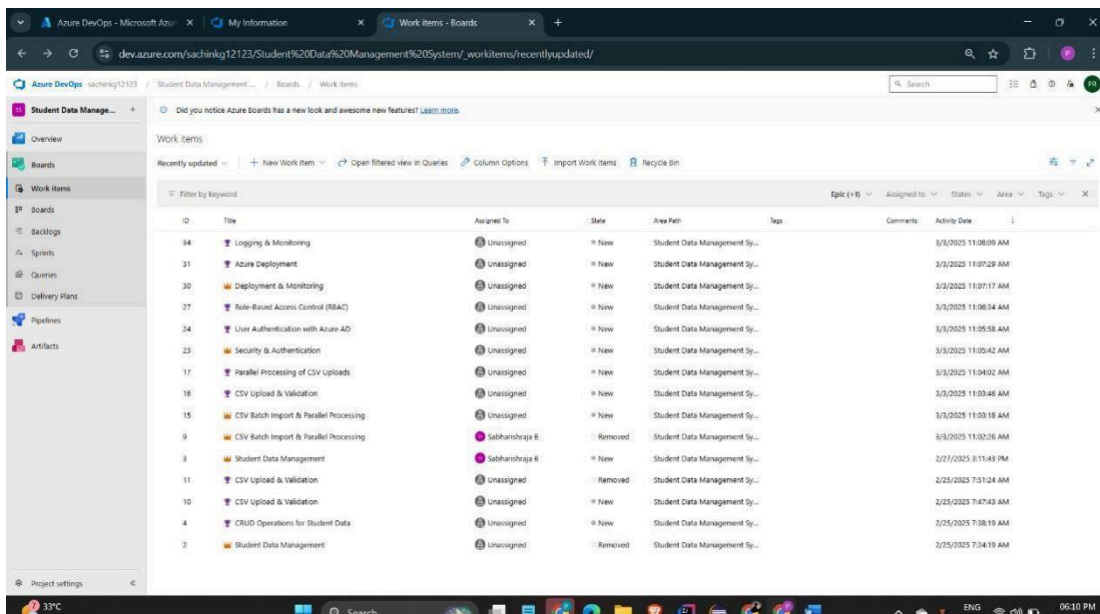
Successfully created an Azure DevOps project with user story management and agile workflow setup.

<b>EXP NO: 3</b>	<h2>SETTING UP EPICS, FEATURES, AND USER STORIES FOR PROJECT PLANNING</h2>
------------------	--

**Aim:**

To learn about how to create epics, user story, features, backlogs for your assigned project.

**Create Epic, Features, User Stories, Task**



The screenshot displays the Azure DevOps 'Work Items' board for a project named 'Student Data Management'. The interface includes a left-hand navigation pane with options like Overview, Boards, Work Items, Backlogs, Sprints, Queries, Delivery Plans, Pipelines, and Artifacts. The main area shows a table of work items, which are currently filtered by keyword. The table columns include ID, Title, Assigned To, State, Area Path, Tags, Comments, and Activity Date. The work items listed are tasks related to logging, deployment, security, and data processing, all currently in an 'Unassigned' state.

ID	Title	Assigned To	State	Area Path	Tags	Comments	Activity Date
34	Logging & Monitoring	Unassigned	New	Student Data Management Sy...			3/3/2025 11:08:09 AM
31	Azure Deployment	Unassigned	New	Student Data Management Sy...			3/3/2025 11:07:29 AM
30	Deployment & Monitoring	Unassigned	New	Student Data Management Sy...			3/3/2025 11:07:17 AM
27	Role-Based Access Control (RBAC)	Unassigned	New	Student Data Management Sy...			3/3/2025 11:06:34 AM
24	User Authentication with Azure AD	Unassigned	New	Student Data Management Sy...			3/3/2025 11:05:58 AM
23	Security & Authentication	Unassigned	New	Student Data Management Sy...			3/3/2025 11:05:42 AM
17	Parallel Processing of CSV Uploads	Unassigned	New	Student Data Management Sy...			3/3/2025 11:04:02 AM
16	CSV Upload & Validation	Unassigned	New	Student Data Management Sy...			3/3/2025 11:03:46 AM
15	CSV Batch Import & Parallel Processing	Unassigned	New	Student Data Management Sy...			3/3/2025 11:03:18 AM
9	CSV Batch Import & Parallel Processing	Subharishya B.	Removed	Student Data Management Sy...			3/3/2025 11:02:26 AM
8	Student Data Management	Subharishya B.	New	Student Data Management Sy...			2/25/2025 8:11:43 PM
11	CSV Upload & Validation	Unassigned	Removed	Student Data Management Sy...			2/25/2025 7:51:24 AM
10	CSV Upload & Validation	Unassigned	New	Student Data Management Sy...			2/25/2025 7:47:43 AM
4	CRUD Operations for Student Data	Unassigned	New	Student Data Management Sy...			2/25/2025 7:38:19 AM
2	Student Data Management	Unassigned	Removed	Student Data Management Sy...			2/25/2025 7:34:19 AM

## 1.Fill in Epics

The screenshot shows the Azure DevOps web interface. The left sidebar contains navigation links: Overview, Boards, Work Items, Backlogs, Sprints, Queries, Delivery Plans, Pipelines, and Artifacts. The main content area displays 'Epic 30: Deployment & Monitoring' under the 'Student Data Management System' area. The 'Description' field is empty with a placeholder 'Click to add Description'. The 'Discussion' section has a text input field and a 'switch to Markdown editor' link. The 'Planning' section includes fields for Priority (2), Risk, Effort, Business Value, Time Criticality, Start Date, and Target Date. The 'Deployment' section has a text area for deployment status reporting. The 'Development' section has a text area for development status reporting. The 'Related Work' section shows a list of related work items.

## 2.Fill in Features

The screenshot shows the Azure DevOps web interface. The left sidebar contains navigation links: Overview, Boards, Work Items, Backlogs, Sprints, Queries, Delivery Plans, Pipelines, and Artifacts. The main content area displays 'Feature 34: Logging & Monitoring' under the 'Student Data Management System' area. The 'Description' field is empty with a placeholder 'Click to add Description'. The 'Discussion' section has a text input field and a 'switch to Markdown editor' link. The 'Planning' section includes fields for Priority (2), Risk, Effort, Business Value, Time Criticality, Start Date, and Target Date. The 'Deployment' section has a text area for deployment status reporting. The 'Development' section has a text area for development status reporting. The 'Related Work' section shows a list of related work items, including '30: Deployment & Monitoring' and '35: As a system admin, I want to monitor API logs so that I can...'. A Microsoft sign-in overlay is visible in the top right corner.

### 3.Fill in User Story Details

The screenshot shows the Azure DevOps interface for creating a new user story. The browser address bar shows the URL: `dev.azure.com/sachinkg12123/Student%20Data%20Management%20System/_workitems/create/User%20Story`. The left sidebar contains navigation options: Overview, Boards, Work Items, Backlogs, Sprints, Queries, Delivery Plans, Pipelines, and Artifacts. The main content area is titled 'Work Items' and shows a 'NEW USER STORY' form. The form has the following fields and values:

- Title:** As an instructor, I want to upload a CSV file containing student details, So that I can import all student records into the system efficiently without entering them one by one.
- State:** New
- Reason:** New
- Area:** Student Data Management System
- Iteration:** Student Data Management System
- Description:** Click to add Description.
- Acceptance Criteria:** Click to add Acceptance Criteria.
- Discussion:** Click to add Discussion.
- Planning:** Story Points, Priority: 2, Risk.
- Classification:** Value area.
- 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.

The form also includes a 'Save' button and a 'Maximize Discussion section' button. The bottom of the screen shows the Windows taskbar with the time 06:17 PM on 21-04-2025.

**Result:**

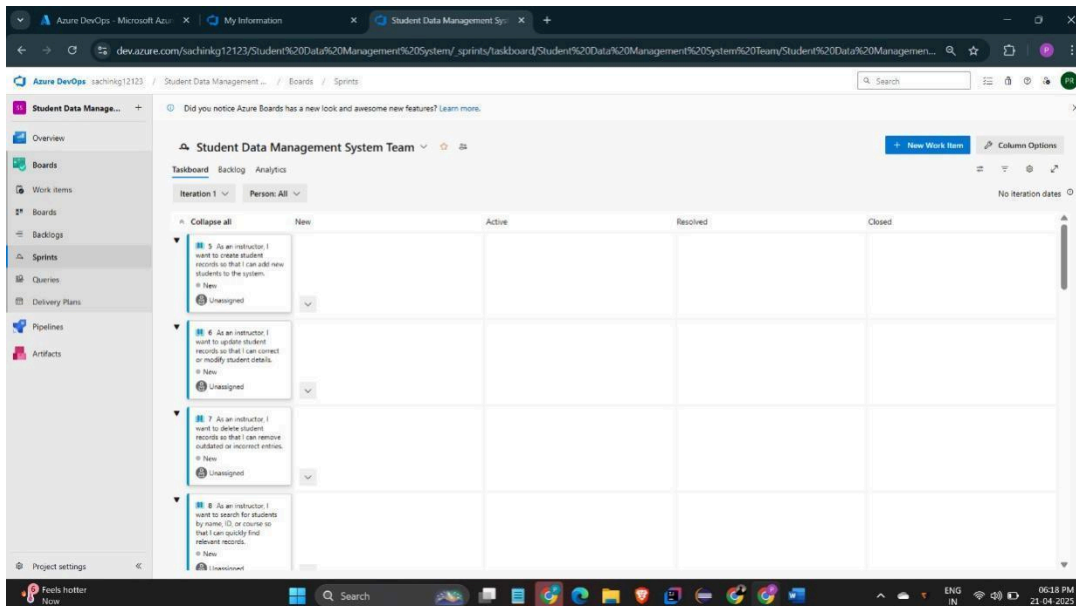
Thus, the creation of epics, features, user story and task has been created successfully.

<b>EXP NO: 4</b>	<b>SPRINT PLANNING</b>
------------------	------------------------

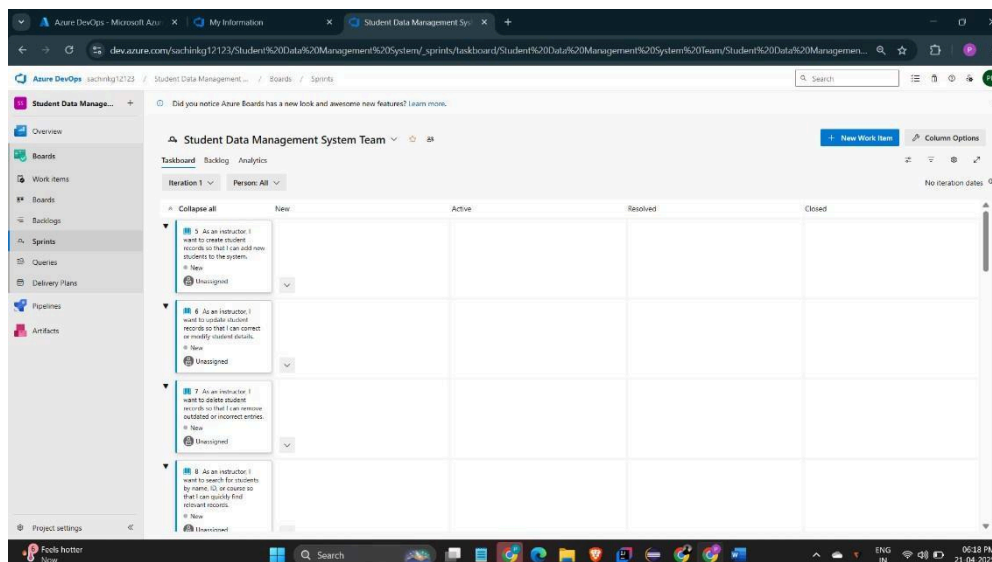
**Aim:**

To assign user story to specific sprint for the Student Database Management System Project.

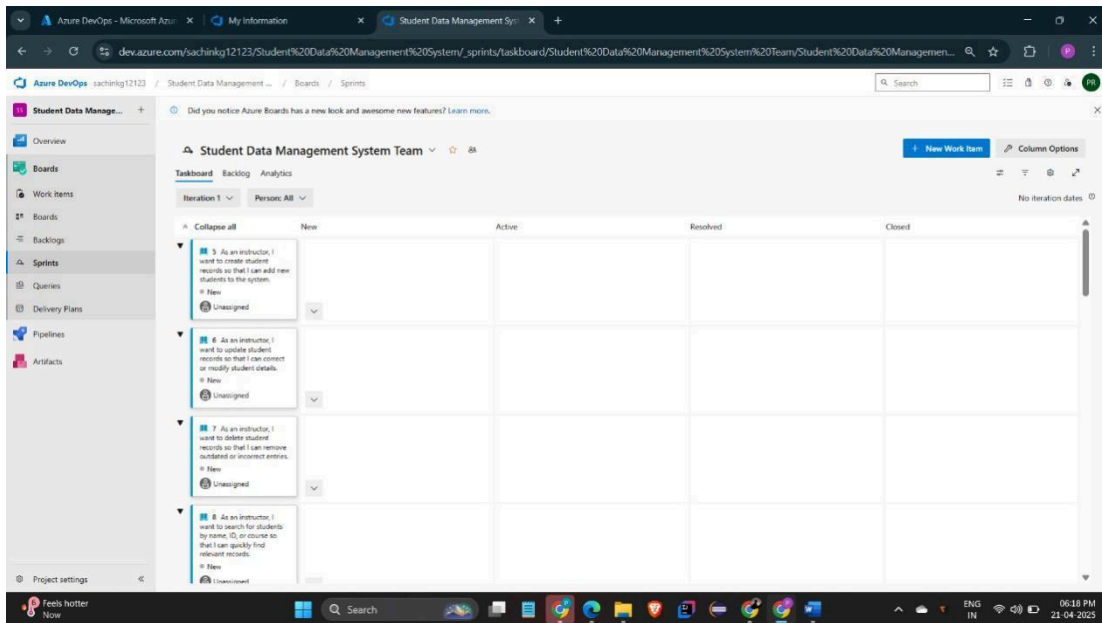
**Sprint Planning****Sprint 1**



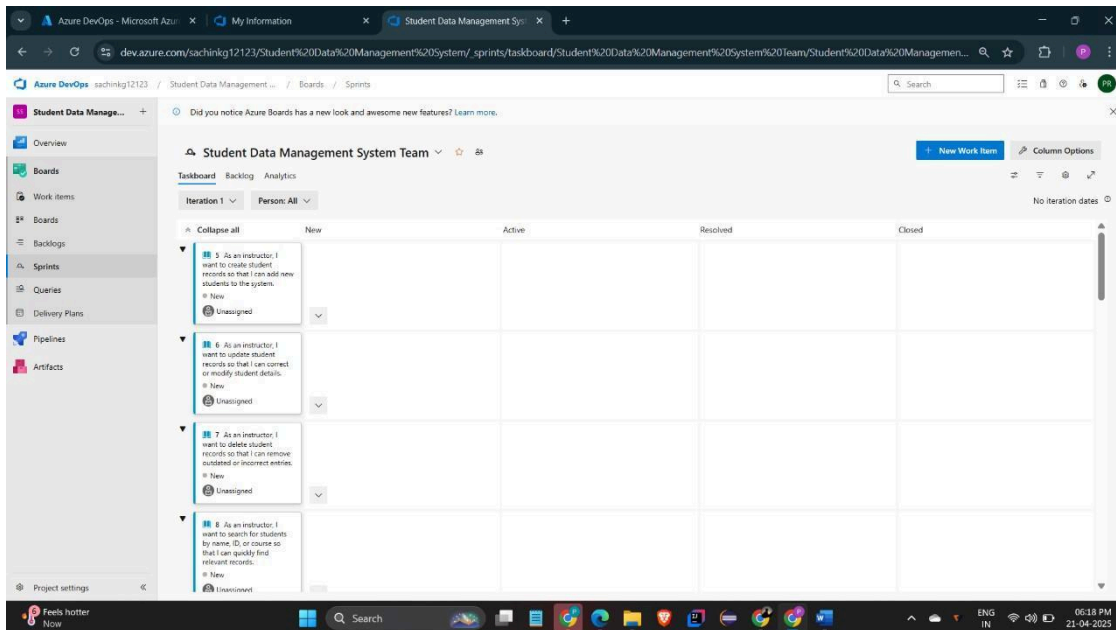
## Sprint 2



## Sprint 3



## Sprint 4





**Result:**

The Sprints are created for the Student Database Management System Project.

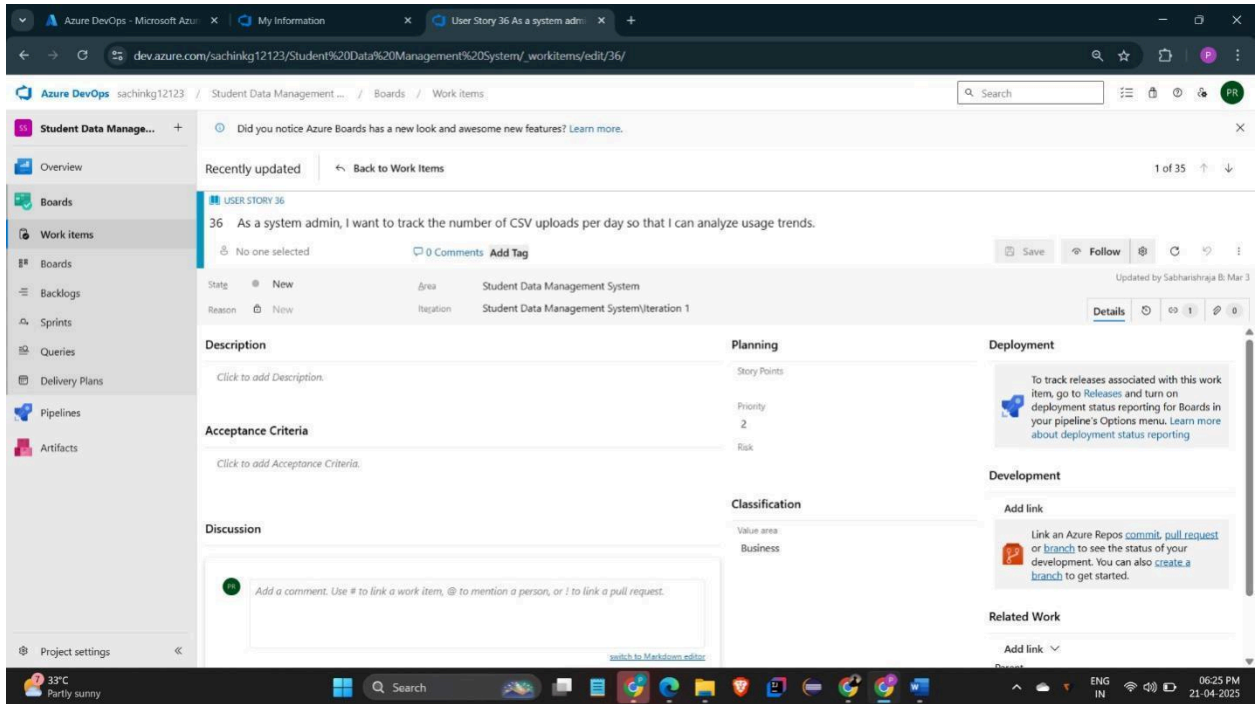
<b>EXP NO: 5</b>	<b>POKER ESTIMATION</b>
------------------	-------------------------

**Aim:**

Create Poker Estimation for the user stories - Student Database Management System Project.

**Poker Estimation**





### Result:

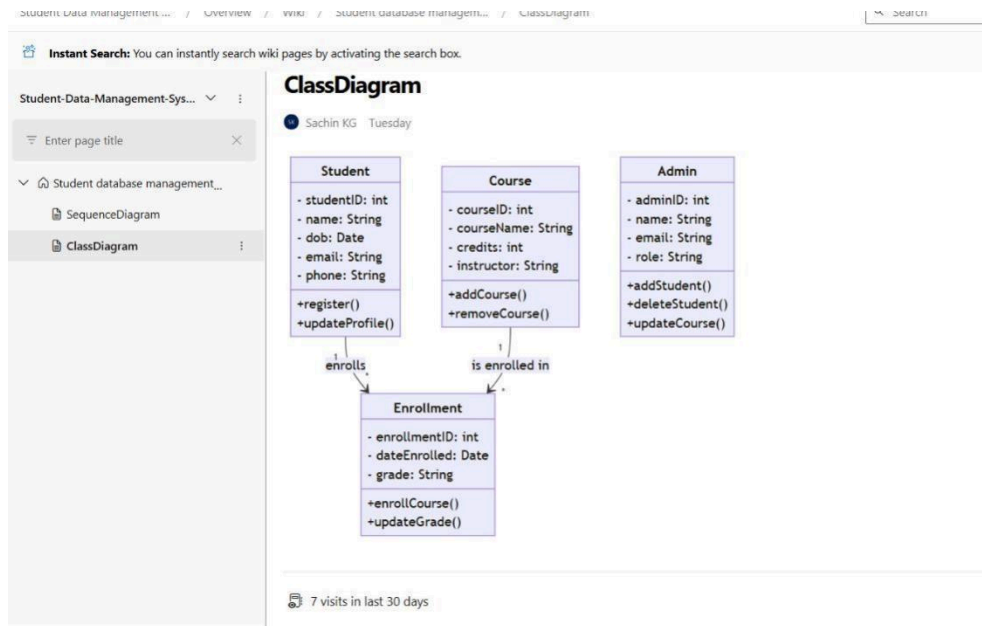
The Estimation/Story Points is created for the project using Poker Estimation.

EXP NO: 6	<h2 style="text-align: center;">DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE</h2>
-----------	---

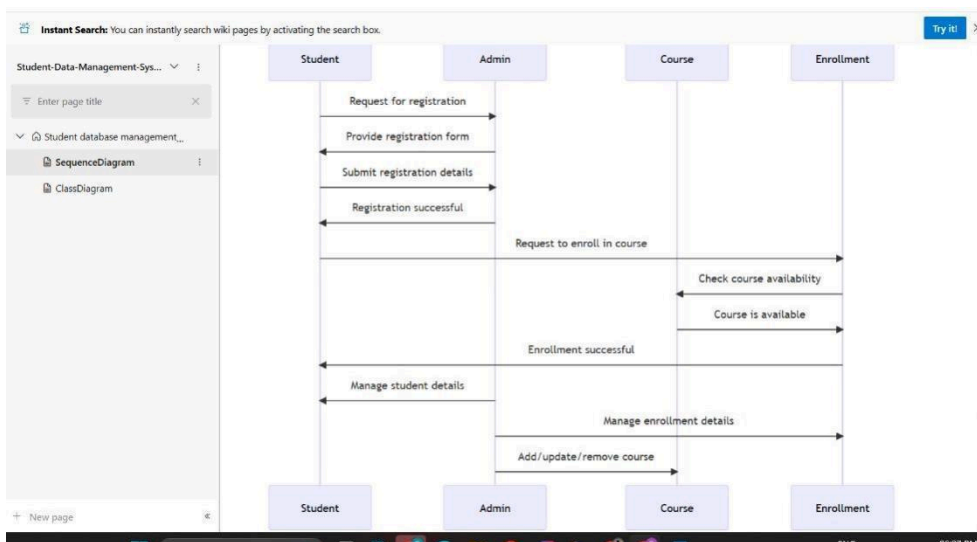
### Aim:

To Design a Class Diagram and Sequence Diagram for the given Project. 6A.

### Class Diagram



## 6B. Sequence Diagram



**Result:**

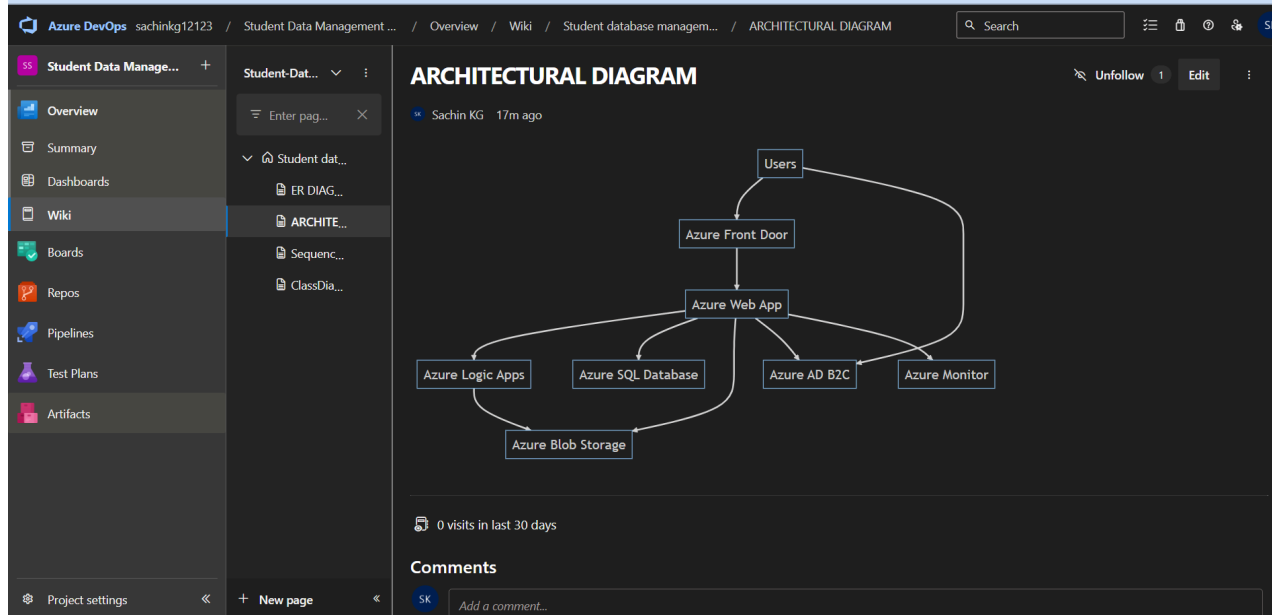
The Class Diagram and Sequence Diagram is designed Successfully for the Student Database Management System.

<b>EXP NO: 7</b>	<b>DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE</b>
------------------	--

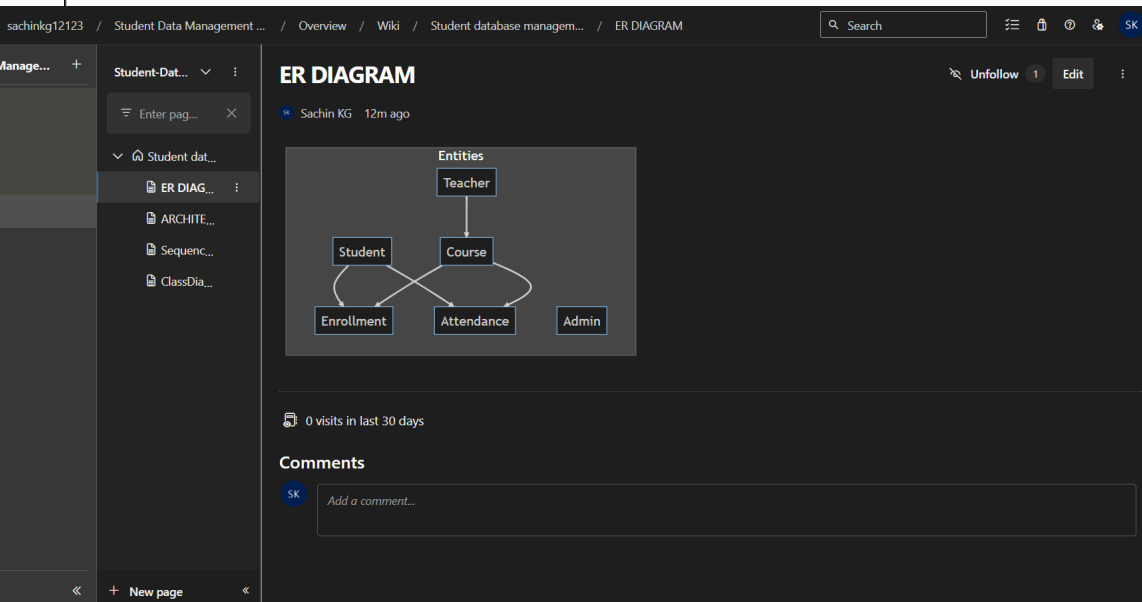
**Aim:**

To Design an Architectural Diagram and ER Diagram for the given Project. 7A.

## Architectural Diagram



## 7B.ER Diagram



### Result:

The Architecture Diagram and ER Diagram is designed Successfully for the Music Playlist Batch Creator

EXP NO: 8

TESTING – TEST PLANS AND TEST CASES



## Aim:

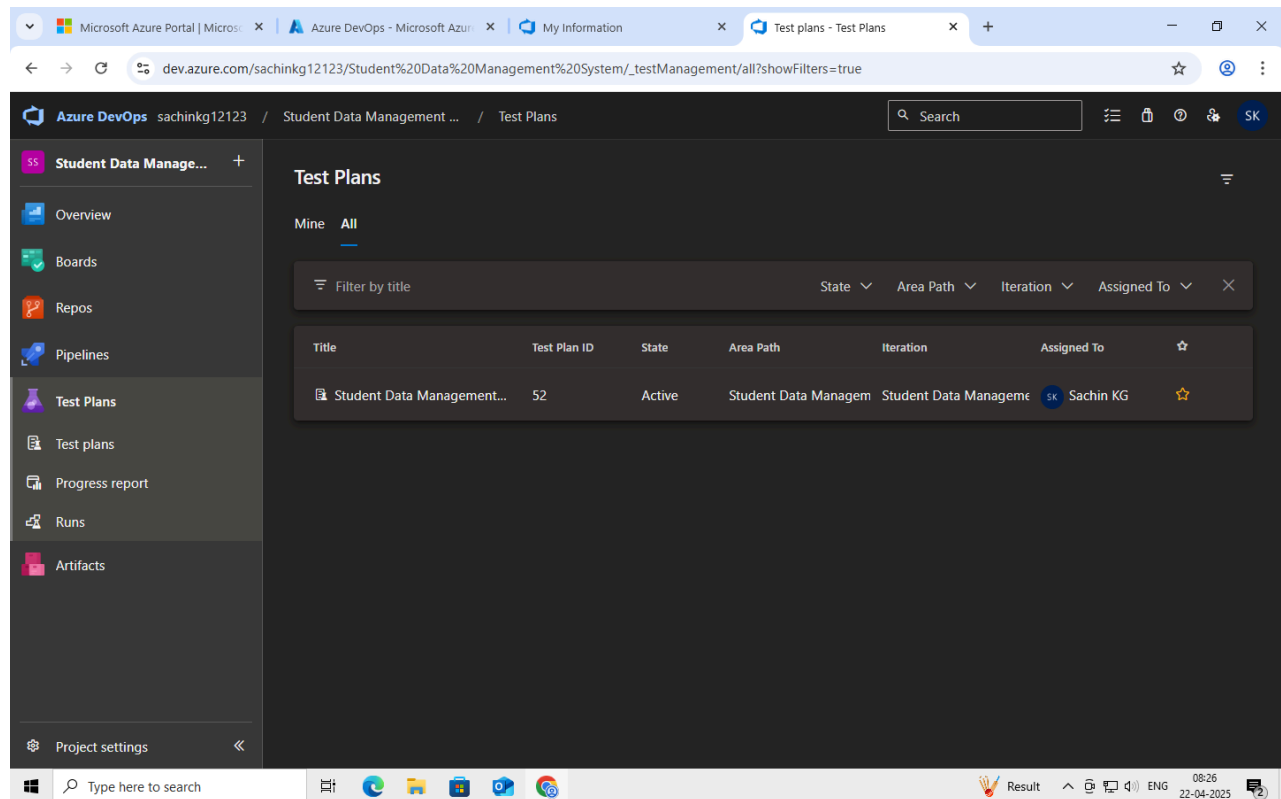
Test Plans and Test Case and write two test cases for at least five user stories showcasing the happy path and error scenarios in azure DevOps platform.

### Test Planning and Test Case Test Case Design Procedure

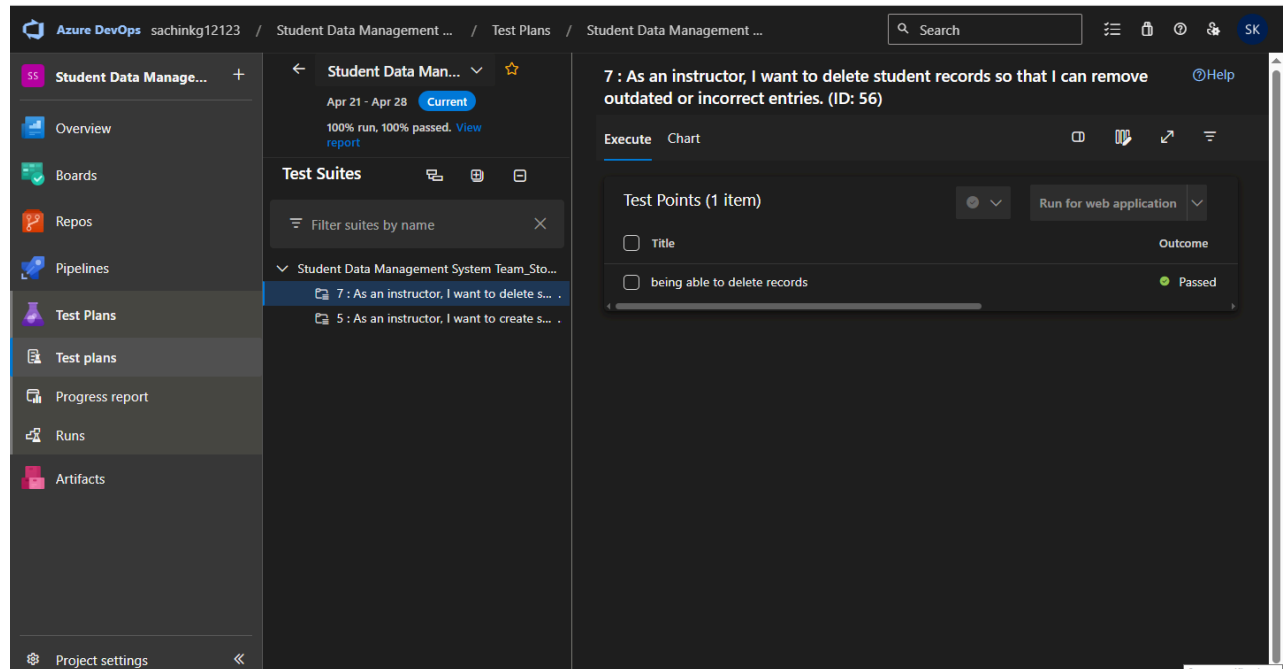
1. **Understand Core Features of the Application** ○ User Signup & Login ○ Viewing and Managing Playlists ○ Fetching Real-time Metadata ○ Editing playlists (rename, reorder, record)
  - Creating smart audio playlists based on categories (mood, genre, artist, etc.)
2. **Define User Interactions** ○ Each test case simulates a real user behaviour (e.g., logging in, renaming a playlist, adding a song).
3. **Design Happy Path Test Cases** ○ Focused on validating that all features function as expected under normal conditions.
  - Example: User logs in successfully, adds item to playlist, or creates a category-based playlist.
4. **Design Error Path Test Cases** ○ Simulate negative or unexpected scenarios to test robustness and error handling.
  - Example: Login fails with invalid credentials, save fails when offline, no recommendations found.
5. **Break Down Steps and Expected Results** ○ Each test case contains step-by-step actions and a corresponding expected outcome.
  - Ensures clarity for both testers and automation scripts.
6. **Use Clear Naming and IDs** ○ Test cases are named clearly (e.g., TC01 – Successful Login, TC10 – Save Playlist Fails).
  - Helps in quick identification and linking to user stories or features.
7. **Separate Test Suites** ○ Grouped test cases based on functionality (e.g., Login, Playlist Editing, Recommendation System).

- Improves organization and test execution flow in Azure DevOps.
- 8. Prioritize and Review**
- Critical user actions are marked high-priority.
  - Reviewed for completeness and traceability against feature requirements.

## 1.New test plan



## 2.Test suite



### 3. Test case

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

Student Database Management System – Test Plans

#### USER STORIES

- As a user, I want to sign up and log in securely so that I can access my playlists (ID: 79).
- As a user, I need to see my playlist in one place (ID: 76).
- As a user, I should be able to create an audio playlist as needed (ID: 73).
- As a user, I should be able to rename, record, and change the playlist (ID: 68).
- As a user, I need to have real-time metadata (ID: 65).

#### Test Suites

##### Test Suit: TS01 - User Login (ID: 86)

##### 1. TC01 – Successful Sign Up ○ Action:

- Go to the Sign-Up page.
- Enter valid name, email, and password.
- Click "Sign Up".

##### ○ Expected Results:



- Sign-Up form is displayed.
- Fields accept values without error.
- Account is created, and the user is redirected to the dashboard.

○ **Type:** Happy Path

## **2. TC02 – Secure Login**

○ **Action:**

- Go to the Login page.
- Enter valid email and password.
- Click on "Login".

○ **Expected Results:**

- Login form is displayed.
- Fields accept data without error.
- User is logged in and redirected to the dashboard.

○ **Type:** Happy Path

## **3. TC03 – Sign Up with Existing Email**

○ **Action:**

- Go to the Sign-Up page.
- Enter a name and an already registered email.
- Click on "Sign Up".

○ **Expected Results:**

- Fields accept data.
- Error message "Email already registered" is displayed.

○ **Type:** Error Path

## **4. TC04 – Login with Wrong Password**

○ **Action:**

- Go to the Login page.

- Enter valid email and incorrect password.
- Click on "Login".
- **Expected Results:**
  - Input is accepted.
  - Error message "Invalid username or password" is shown.
- **Type:** Error Path

## **Test Suit: TS02 - View Playlists (ID: 87)**

### **1. TC05 – View Playlist Page**

- **Action:**
  - Log in successfully.
  - Navigate to "My Playlists" section.
- **Expected Results:**
  - SAll created playlists are displayed clearly.
- **Type:** Happy Path

### **2. TC06 – Playlist Loading Failure**

- **Action:**
  - Disconnect from the internet.
  - Navigate to "My Playlists".
- **Expected Results:**
  - Network is offline.
  - Error message "Unable to load playlists" is shown.
- **Type:** Error Path

## **Test Suit: TS03 - Real-Time Metadata (ID: 88)**

### **1. TC07 – Real-Time Metadata Display**

#### **Action:**

- Play a song.

- Observe the metadata panel.

- **Expected Results:**

- Metadata (title, artist, album, duration) is displayed and updates in real time.
- **Type:**

Happy Path

2. TC08 – Metadata Not Updating ○

**Action:**

- Play a different song.
- Observe the metadata panel.

- **Expected Results:**

- Metadata remains static or shows default/fallback message.

- **Type:** Error Path

**Test Suit: TS04 - Playlist Editing (ID: 89)**

1. TC09 – Rename Playlist Successfully ○

**Action:**

- Navigate to "My Playlists".
- Click "Rename" next to a playlist.
- Enter a new name and click "Save".

- **Expected Results:**

- Playlist name updates successfully.

- **Type:** Happy Path

2. TC10 – Rename with Blank Name ○

**Action:**

- Click "Rename" on a playlist.
- Leave the field blank.
- Click "Save".

- **Expected Results:**

- Error message "Playlist name cannot be empty" is shown.

- **Type:** Error Path

### 3. TC11 – Change Playlist Order ○

**Action:**

- Open a playlist.
- Drag and drop songs to reorder.
- Click "Save".

- **Expected Results:**

- Playlist order is updated and saved.

- **Type:** Happy Path

### 4. TC12 – Change Playlist Order Fails ○

**Action:**

- Login and go to “My Playlists”.
- Select a playlist.
- Go offline or simulate server error. ▪ Reorder songs and click “Save Order”.

- **Expected Results:**

- Error message: "Failed to update order. Please check your connection".

- **Type:** Error Path

## Test Suit: TS05 - Smart Playlist Creation (ID: 90)

### 1. TC13 – Generate Playlist Based on Various Categories ○

**Action:**

- Login with valid credentials.
- Click on "Generate Playlist".
- Select categories.
- Click "Generate Playlist".

- **Expected Results:**

- Playlist is generated based on selected mood and categories.

- **Type:** Happy Path

## 2. TC14 – Fail to Generate Playlist Due to Missing Category Selection or Invalid Input ○

### Action:

- Login with valid credentials.
- Click on "Generate Playlist".
- Select categories.
- Click "Generate Playlist".

### ○ Expected Results:

- Error message: "Please select at least one valid category" or "No recommendations found for the selected filters".

### ○ Type: Error Path

## Test Cases

The screenshot displays the Azure DevOps web interface. The top navigation bar shows the user 'sachinkg12123' and the current path: 'Student Data Management ... / Test Plans / Student Data Management ...'. A search bar and various utility icons are also present.

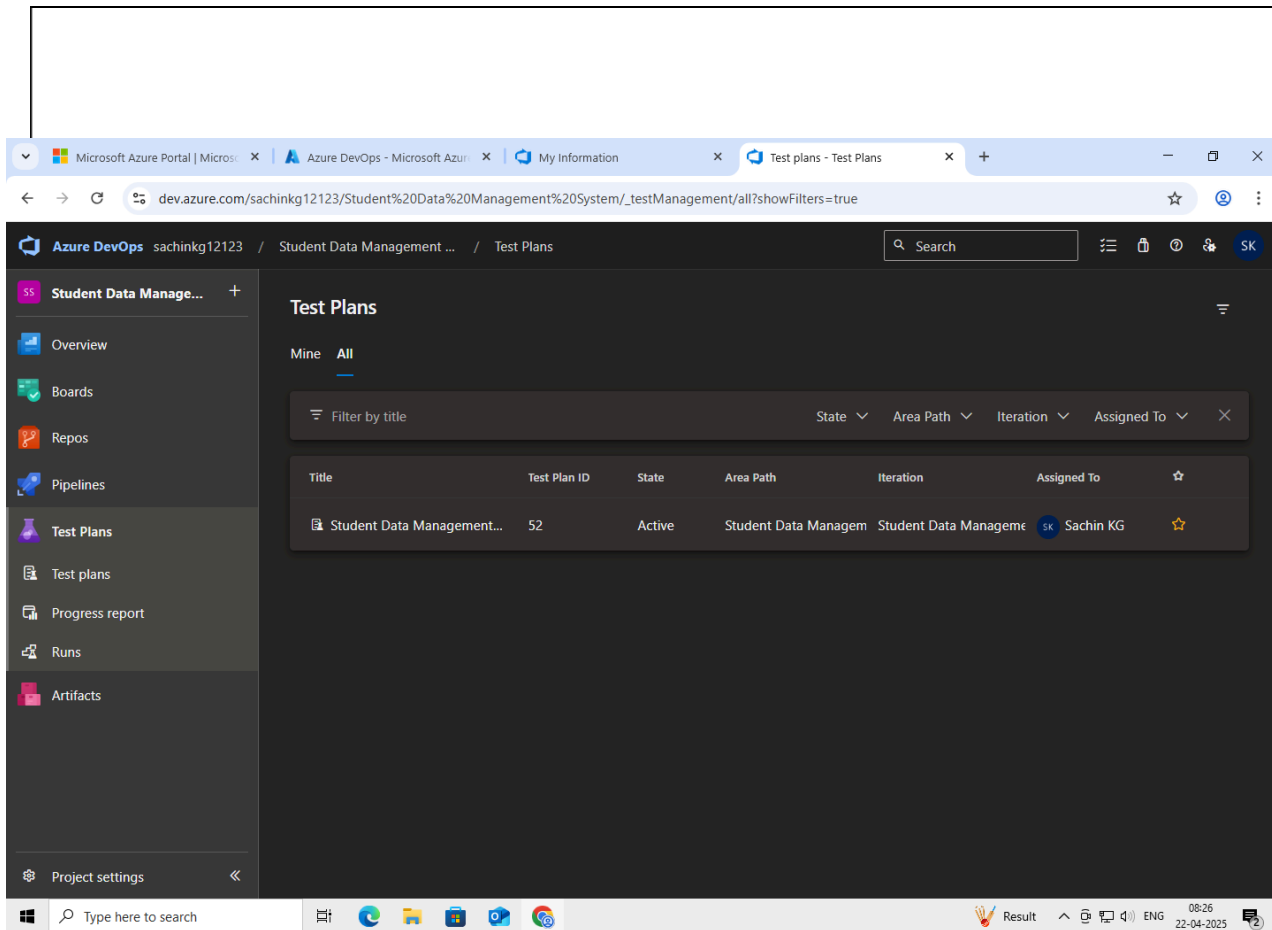
The main content area is titled 'TEST CASE 51' and 'being able to update records'. It shows the test case was created by 'Sachin KG' and has '0 Comments'. The 'State' is 'Design' and the 'Area' is 'Student Data Management System'. The 'Reason' is 'New' and the 'Iteration' is 'Student Data Management System\Iteration...'. The test case was 'Updated by Sachin KG: Yesterday'.

Below the metadata, there are tabs for 'Steps', 'Summary', and 'Associated Automation'. The 'Steps' tab is active, showing a table with the following content:

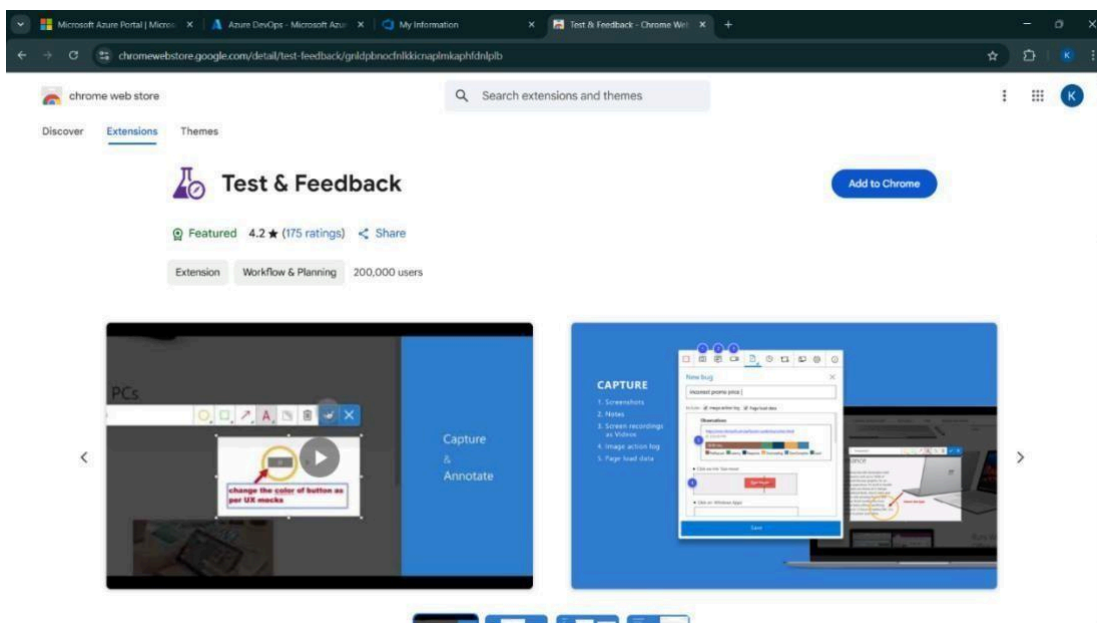
Steps	Action	Expected result
1.	able to view details	
2.	updating the new entries	
3.	display entire details	

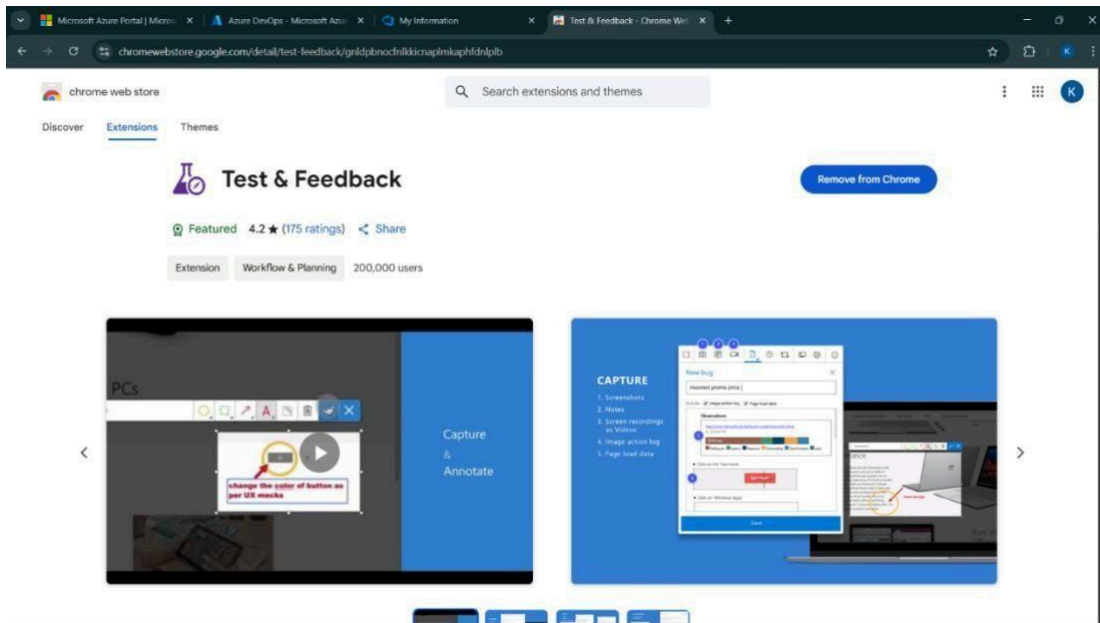
Below the table, there is a link to 'Click or type here to add a step' and a section for 'Parameter values'.

On the right side of the interface, there are sections for 'Deployment' and 'Development'. The 'Deployment' section contains a link to 'Releases' and a link to 'Learn more about deployment status reporting'. The 'Development' section contains a link to 'Add link' and a link to '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.'.



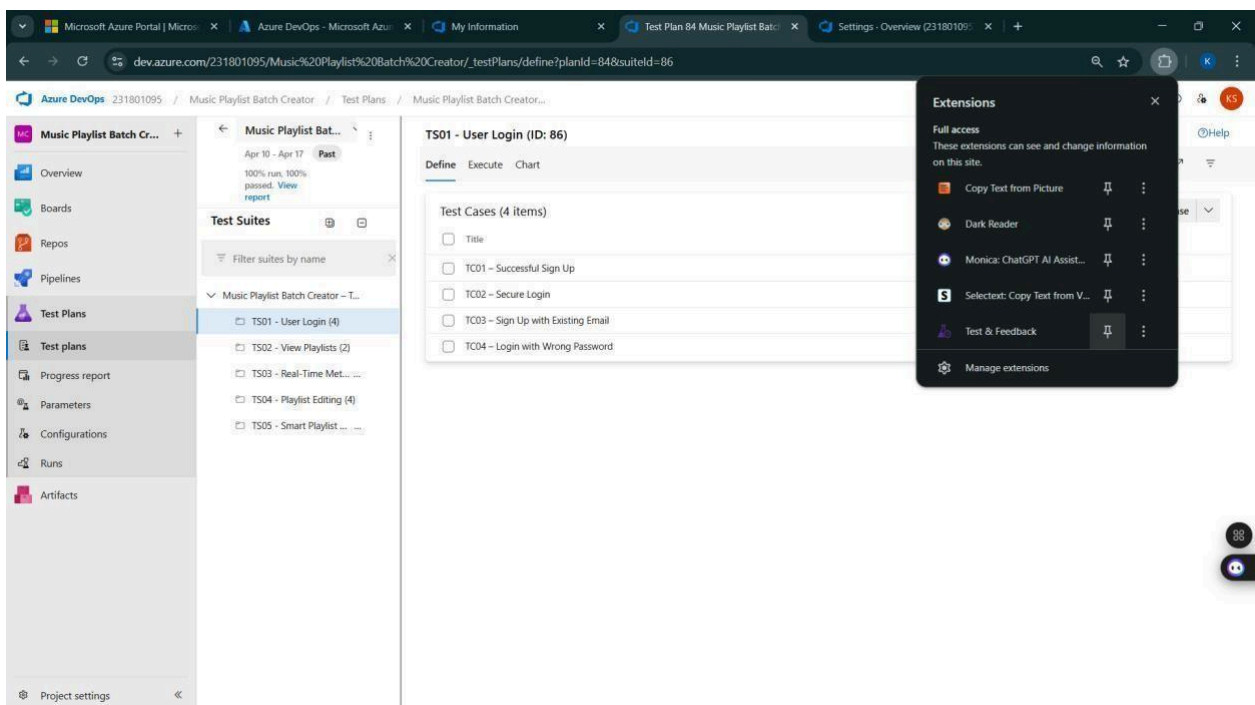
#### 4.Installation of test



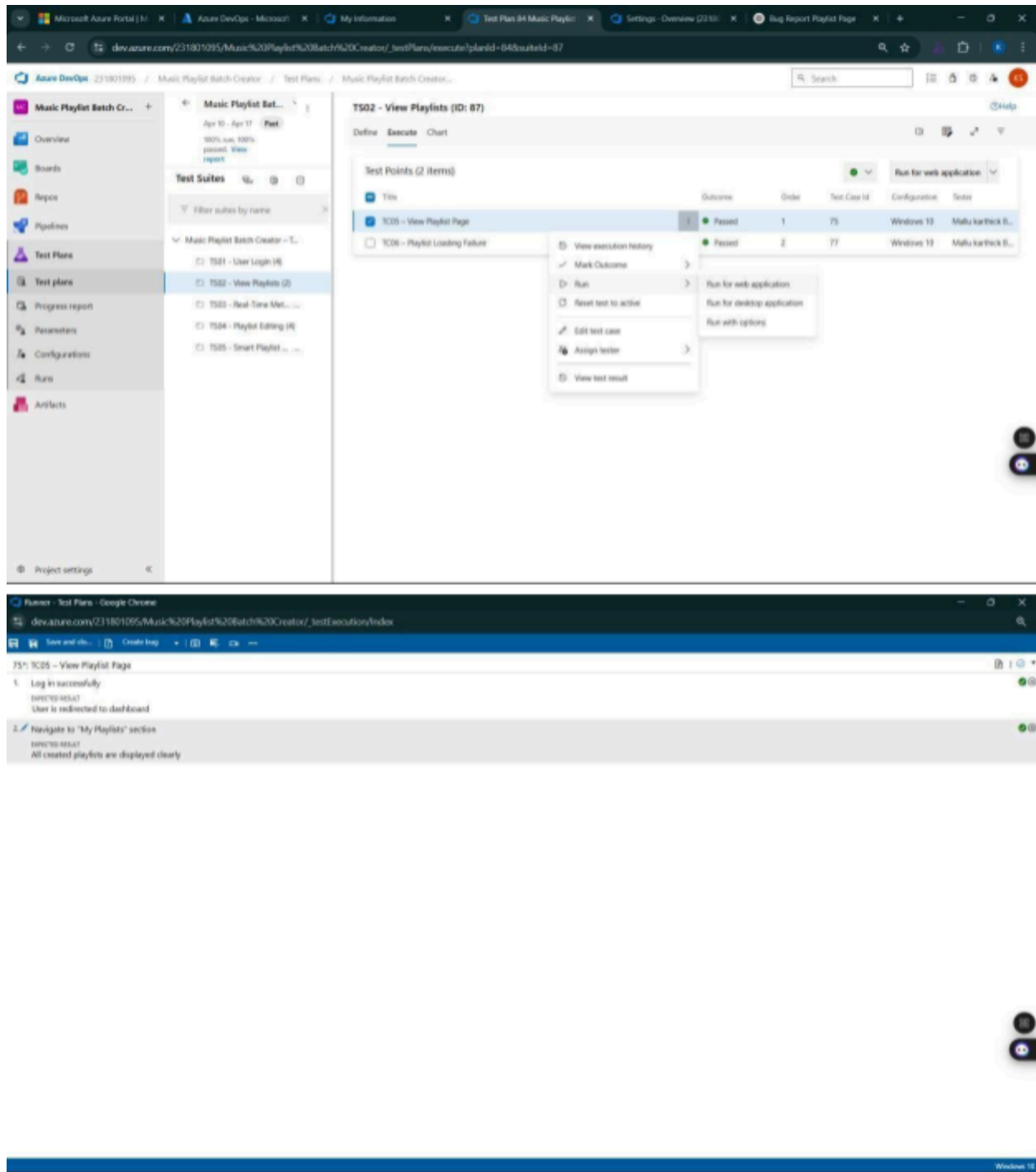


Test and feedback

Showing it as an extension

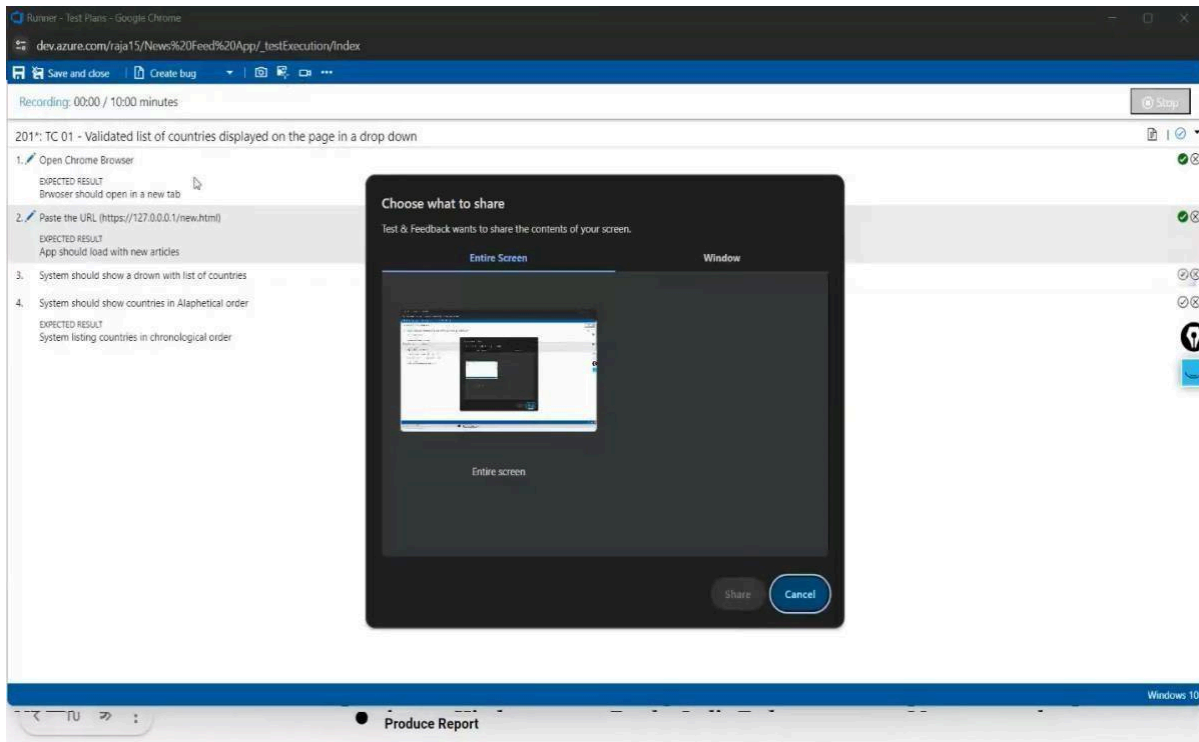


## 5. Running the test cases



## 6. Recording the test case





## 7. Creating the bug

Azure DevOps | sachinkg12123 | Student Data Management ... | Boards | Work items

66 records overload

State: New | Area: Student Data Management System | Reason: New | Iteration: Student Data Management System\Iteration 1

Repro Steps

Step no.	Result	Title
1.	None	able to view student details
2.	None	modify student details
3.	None	display records

Test Configuration: Windows 10

Planning

Resolved Reason

Story Points

Priority: 2

Severity: 3 - Medium

Activity

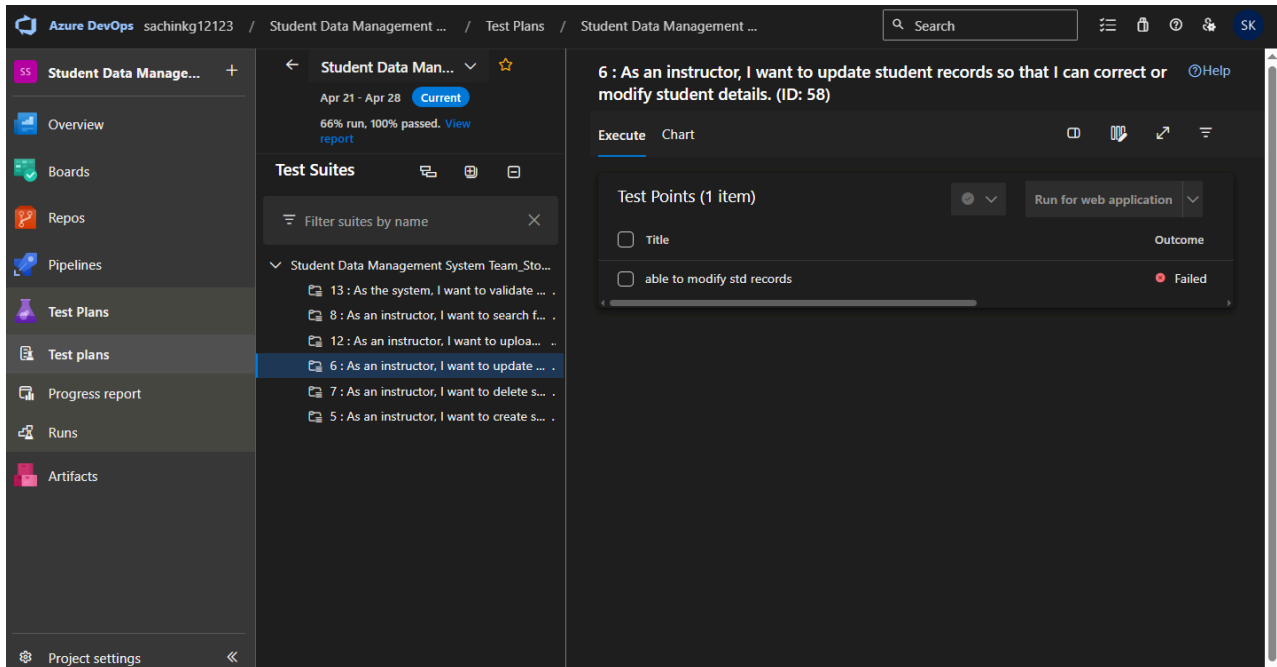
Deployment

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](#)

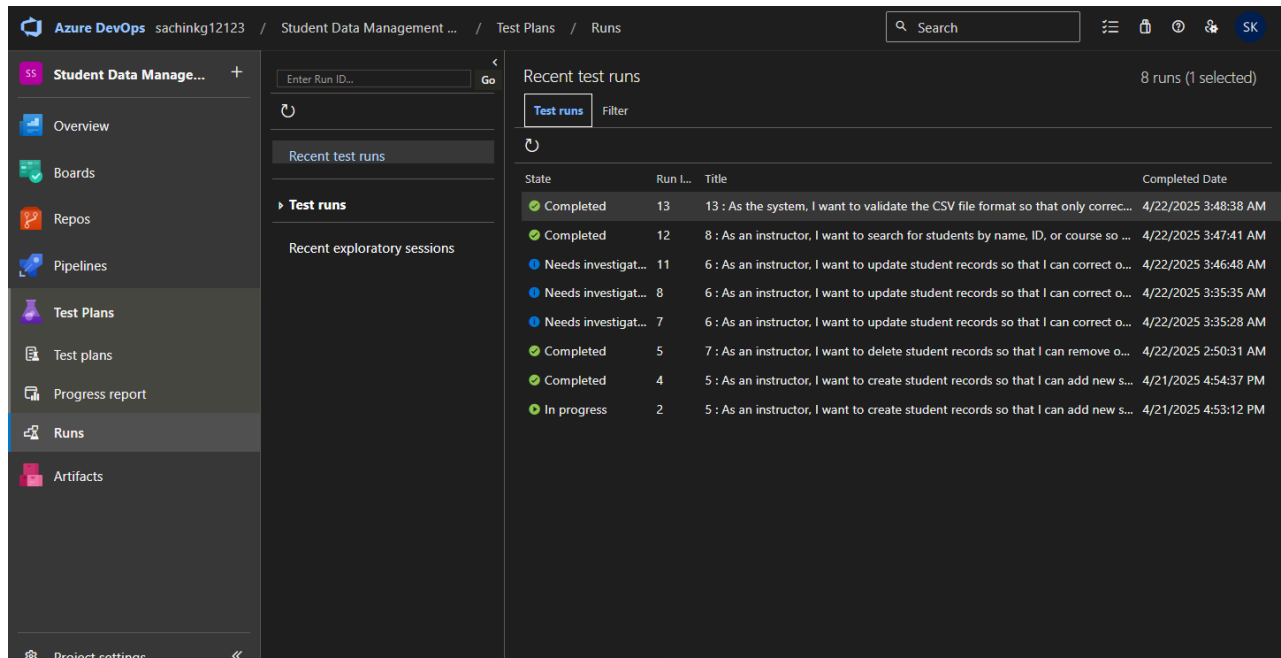
Development

Add link

Link an Azure Repos [commit](#), [pull request](#) or [branch](#) to see the status



## 8. Test case results



## 9. Test report summary

The screenshot shows the Azure DevOps interface for a work item. The left sidebar contains navigation links: Overview, Boards, Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Project settings. The main area displays a bug report titled '66 records overload' with the following details:

- State:** New
- Area:** Student Data Management System
- Reason:** New
- Iteration:** Student Data Management System\Iteration 1
- Repro Steps:**

Step no.	Result	Title
1.	None	able to view student details
2.	None	modify student details
3.	None	display records
- Test Configuration:** Windows 10
- 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

## 10. Progress report

The screenshot shows the Azure DevOps Test Plans Progress report page. The left sidebar contains navigation links: Overview, Boards, Repos, Pipelines, Test Plans, Test plans, Progress report, Runs, and Artifacts. The main area displays the progress report for 'Student Data Management System Team\_Stories\_It...'. The report includes a summary of test plans and points, and an outcome trend chart.

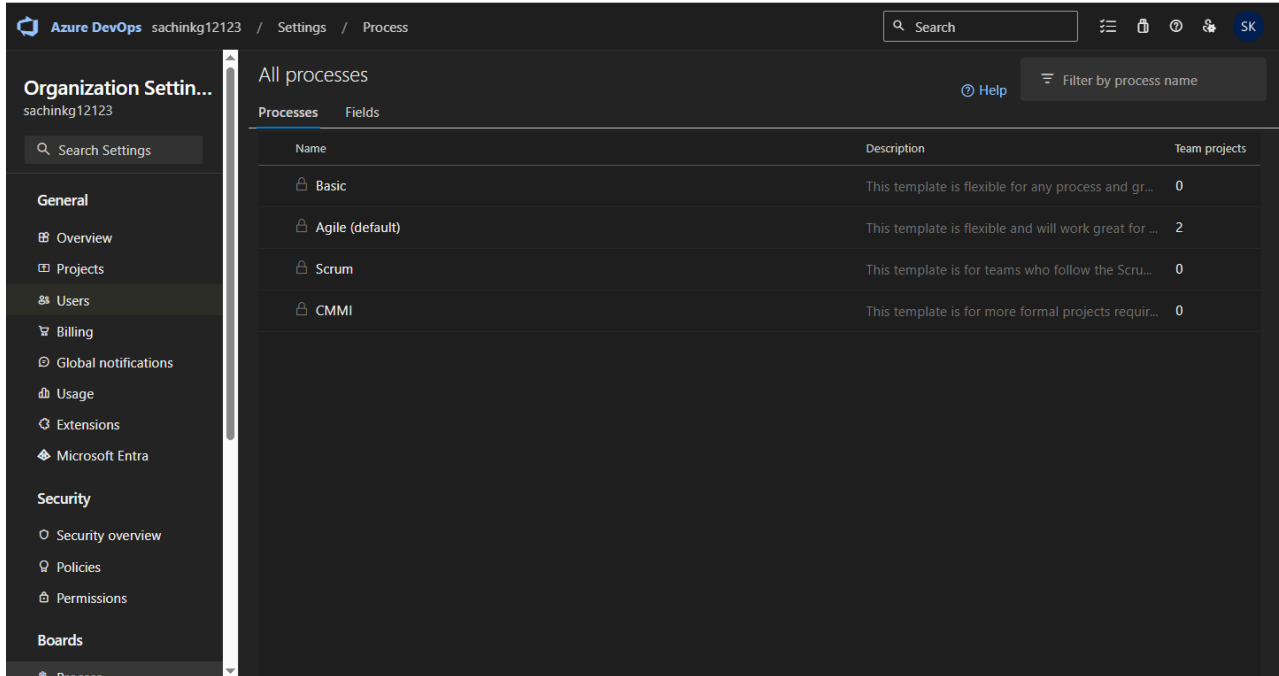
**Summary**

- 1 Test plans
- 6 Test points
- 5 (5 / 6) Test points run
- 83% Run
- 80% (4 / 5) Pass rate
- 4 Passed
- 1 Failed

**Outcome trend**

The chart shows the number of tests run over time, categorized by Not run (grey), Passed (green), and Failed (red). The x-axis represents dates from 2023-04-08 to 2023-04-22. The y-axis represents the number of tests, ranging from 0 to 6. The legend indicates: Not run (grey), Passed (green), Failed (red).

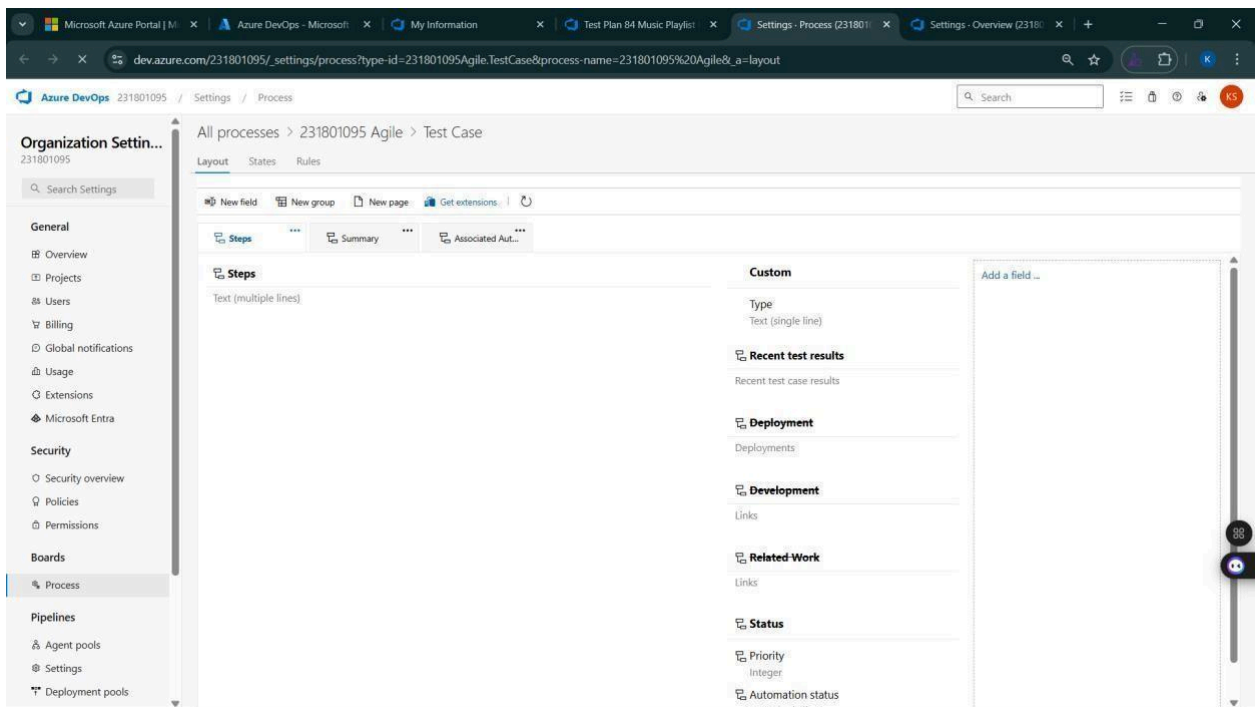
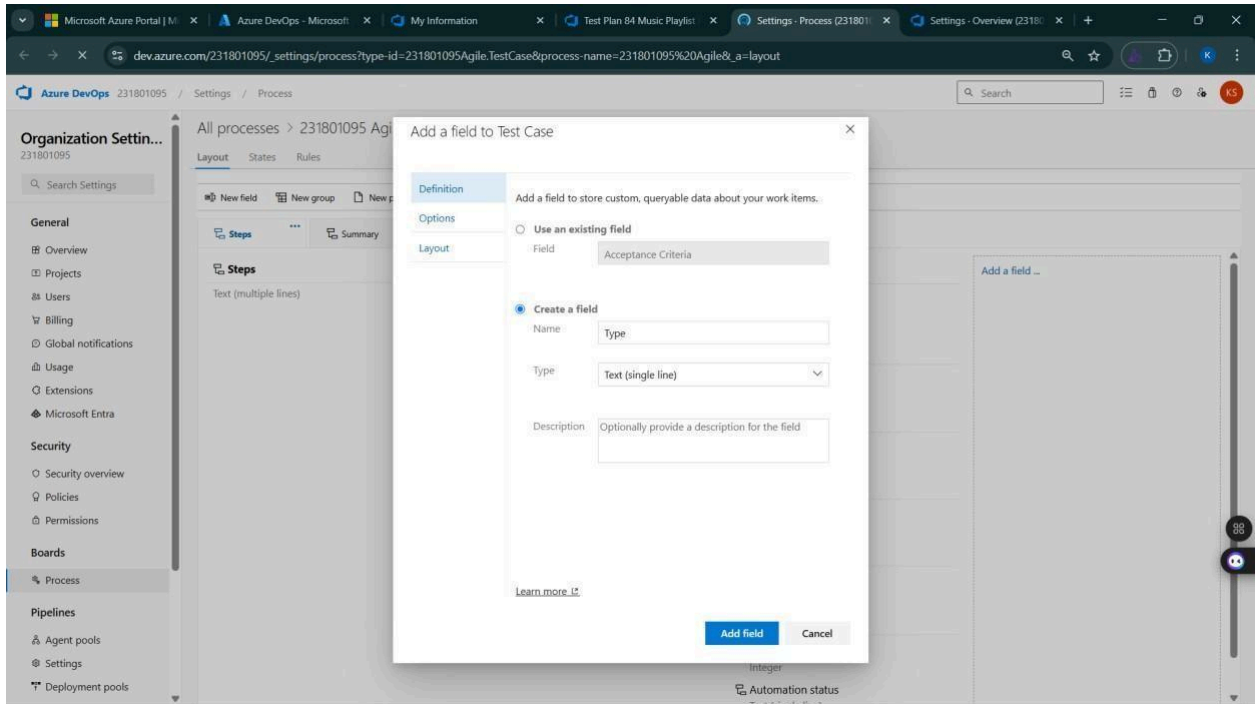
## 11.Changing the test template



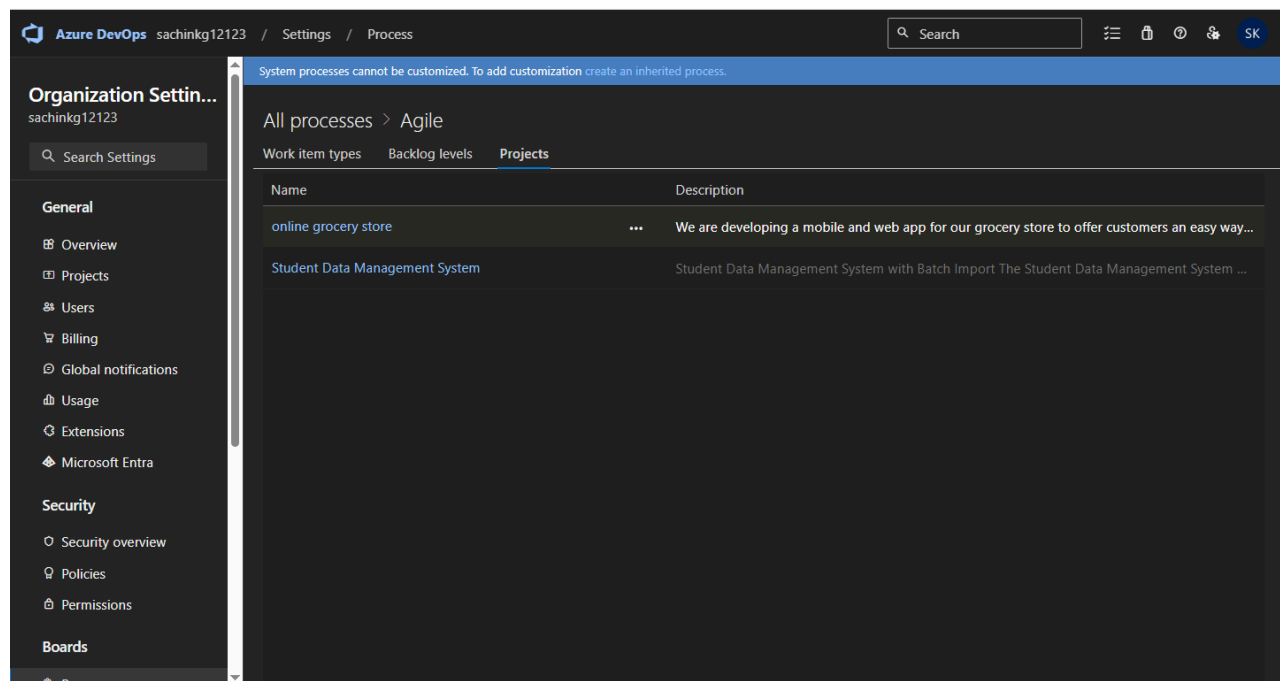
The screenshot shows the Azure DevOps interface for the 'All processes' page. The left sidebar contains the 'Organization Settings' menu with options like General, Overview, Projects, Users, Billing, Global notifications, Usage, Extensions, Microsoft Entra, Security, and Boards. The main content area displays a table of process templates. The table has columns for Name, Description, and Team projects. The templates listed are Basic, Agile (default), Scrum, and CMMI. The 'Agile (default)' template is highlighted, indicating it is the selected template.

Name	Description	Team projects
Basic	This template is flexible for any process and gr...	0
Agile (default)	This template is flexible and will work great for ...	2
Scrum	This template is for teams who follow the Scru...	0
CMMI	This template is for more formal projects requir...	0

## 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

EXP NO: 9

**LOAD TESTING AND PERFORMANCE TESTING****Aim:**

To create an Azure Load Testing resource and run a load test to evaluate the performance of a target endpoint.

**Load Testing****Steps to Create an Azure Load Testing Resource:**

Before you run your first test, you need to create the Azure Load Testing resource:

1. Sign in to Azure Portal  
Go to <https://portal.azure.com> and log in.
2. Create the Resource ○ Go to *Create a resource* → Search for “Azure Load Testing”.
  - Select Azure Load Testing and click Create.
3. Fill in the Configuration Details ○ Subscription: Choose your Azure subscription. ○ Resource Group: Create new or select an existing one. ○ Name: Provide a unique name (no special characters).
  - Location: Choose the region for hosting the resource.
4. (Optional) Configure tags for categorization and billing.
5. Click Review + Create, then Create.
6. Once deployment is complete, click Go to resource.

**Steps to Create and Run a Load Test:**

Once your resource is ready:

1. Go to your Azure Load Testing resource and click Add HTTP requests > Create.
2. Basics Tab ○ Test Name: Provide a unique name.
  - Description: (Optional) Add test purpose.
  - Run After Creation: Keep checked.
3. Load Settings ○ Test URL: Enter the target endpoint (e.g., <https://yourapi.com/products>).
4. Click Review + Create → Create to start the test.

## Load Testing :

portal.azure.com/#@ralakmuthugmail.onmicrosoft.com/resource/subscriptions/99af76e2-8bda-472c-aa5a-4f0c1e55a84d/resourcegroups/Stu...

Microsoft Azure Search resources, services, and docs (G+)

Home > Microsoft.CloudNativeTesting1745293013778 | Overview >

### StudentDataMangement

Azure Load Testing

Search Refresh Delete resource Manage budget Give feedback

Overview

- Activity log
- Access control (IAM)
- Tags
- Resource visualizer
- Tests
- Settings
- Monitoring
- Automation
- Help

Install the new VS Code extension to create Locust test scripts using GitHub Copilot and run load tests at scale on Azure Load Testing. [Get extension](#)

Essentials

Resource group (move) [StudentDataMangementLoadTest](#)

Status ---

Location Central India

Subscription (move) [Azure for Students](#)

Subscription ID 99af76e2-8bda-472c-aa5a-4f0c1e55a84d

VUH usage (current month) [0 / No limit](#)

JSON View

Get started Test runs Tutorials

### Load test your application and infrastructure

<https://portal.azure.com/#@ralakmuthugmail.onmicrosoft.com/resource/subscriptions/99af76e2-8bda-472c-aa5a-4f0c1e55a84d/resourceGroups/StudentDataMangementLoadTest>

Microsoft Azure Search resources, services, and docs (G+)

Home > Microsoft.CloudNativeTesting1745293013778 | Overview > StudentDataMangement >

### TestRun\_4/22/2025\_9:48:46 AM

Last updated by: ralakmuthu@gmail.com | Initiated on: 4/22/2025, 9:48 AM

View all test runs Stop Refresh Rerun Compare App components Configure metrics Download Copy artifacts Share

#### Requests/sec (Avg)

1,000/s  
800/s  
600/s  
400/s  
200/s  
0/s

Request1

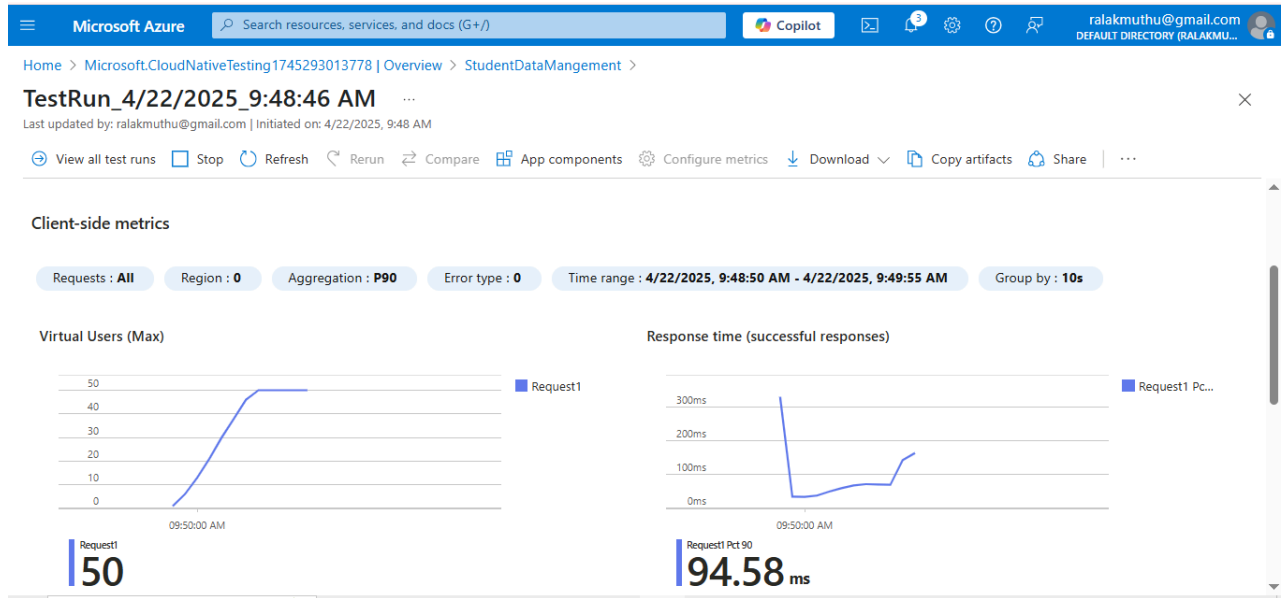
09:50:00 AM

Request1 **643.9 /s**

#### Errors (total)

#### Server-side metrics





### Result:

Successfully created the Azure Load Testing resource and executed a load test to assess the performance of the specified endpoint.

EXP NO:10

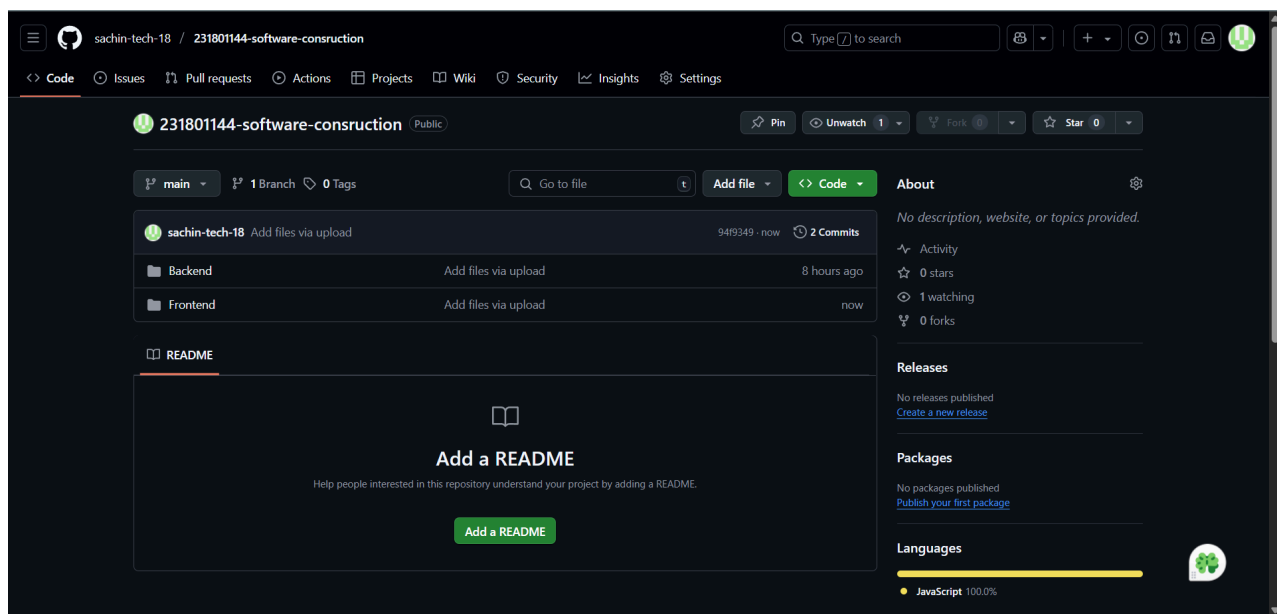
## GITHUB: PROJECT STRUCTURE & NAMING

### 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 Student Database Management System 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.