

RAJALAKSHMI ENGINEERING COLLEGE

RAJALAKSHMI NAGAR, THANDALAM - 602 105



**RAJALAKSHMI
ENGINEERING COLLEGE**

**CS23432
SOFTWARE CONSTRUCTION**

Laboratory Record Note Book

Name :	Kiran Kumar B
Year / Branch / Section :	II/IT-'AD'
Register No.	231001093
Semester :	IV
Academic Year :	2024-2025



RAJALAKSHMI ENGINEERING COLLEGE (AUTONOMOUS)
RAJALAKSHMI NAGAR, THANDALAM- 602 105

BONAFIDE CERTIFICATE

NAME Kiran Kumar B REGISTER NO. 231001093

ACADEMIC YEAR 2024-25 SEMESTER- IV BRANCH: B. Tech Information

Technology [AD/AE]. This Certification is the Bonafide record of work done by the above student in the **CS23432- Software Construction** Laboratory during the year 2024-2025.

Signature of Faculty -in _ Charge

Submitted for the Practical Examination held on _____

Internal Examiner

External Examiner

LAB PLAN
CS23432-SOFTWARE CONSTRUCTION LAB

Ex No	Date	Topic	Page No	Sign
1	21/01/2025	Study of Azure DevOps		
2	28/01/2025	Problem Statement		
3	04/02/2025	Agile Planning		
4	18/02/2025	Create User stories with Acceptance Criteria		
5	25/02/2025	Designing Sequence Diagrams using Azure DevOps-WIKI		
6	04/03/2025	Designing Class Diagram using Azure DevOps-WIKI		
7	11/03/2025	Designing Use case Diagram using Azure DevOps-WIKI		
8	18/03/2025	Designing Activity Diagrams using Azure DevOps-WIKI		
9	25/03/2025	Designing Architecture Diagram Using Star UML		
10	01/04/2025	Design User Interface		
11	08/04/2025	Implementation – Design a Web Page based on Scrum Methodology		
12	15/04/2025	Testing-Test Plan, Test Case and Load Testing		

Course Outcomes (COs)

Course Name: Software Engineering

Course Code: CS23432

CO 1	Understand the software development process models.
CO 2	Determine the requirements to develop software
CO 3	Apply modeling and modeling languages to design software products
CO 4	Apply various testing techniques and to build a robust software products
CO 5	Manage Software Projects and to understand advanced engineering concepts

CO - PO – PSO matrices of course

PO/PSO CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CS23432.1	2	2	3	2	2	2	2	2	2	2	3	2	1	3	-
CS23432.2	2	3	1	2	2	1	-	1	1	1	2	-	1	2	-
CS23432.3	2	2	1	1	1	1	1	1	1	1	1	1	2	2	1
CS23432.4	2	2	3	2	2	2	1	0	2	2	2	1	1	2	1
CS23432.5	2	2	2	1	1	1	1	0	2	1	1	1	2	1	-
Average	2.0	2.2	2.0	1.6	1.6	1.4	1.3	1.3	1.6	1.4	1.8	1.3	1.4	2.0	1.0

Correlation levels 1, 2 or 3 are as defined below:

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High) No correlation: “-“

EXP NO: 1

AZURE DEVOPS ENVIRONMENT SETUP

Aim:

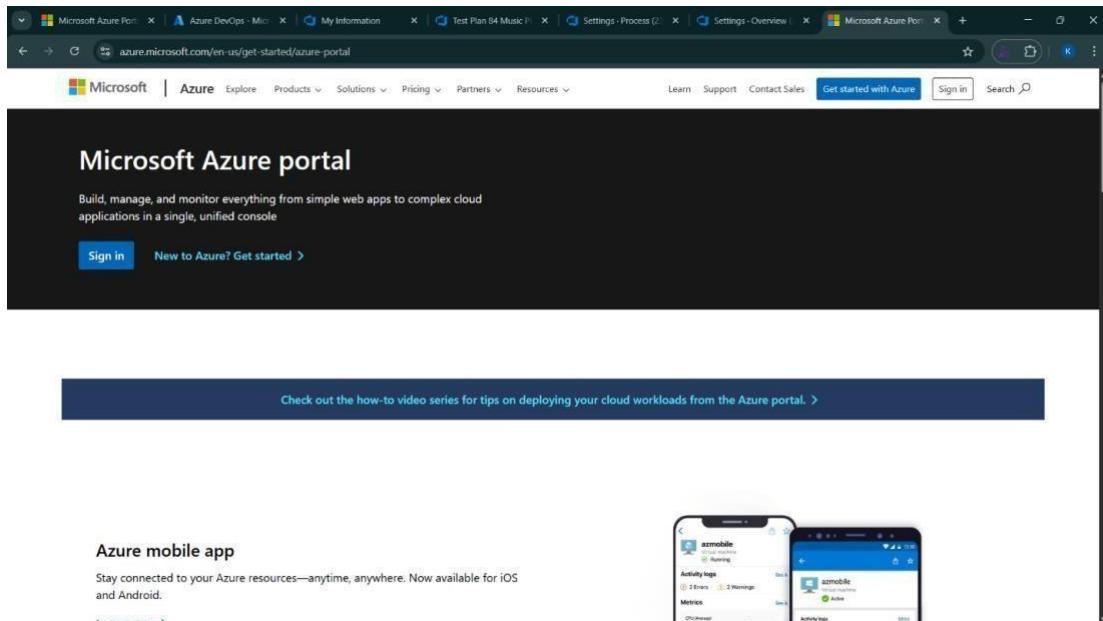
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

INSTALLATION

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



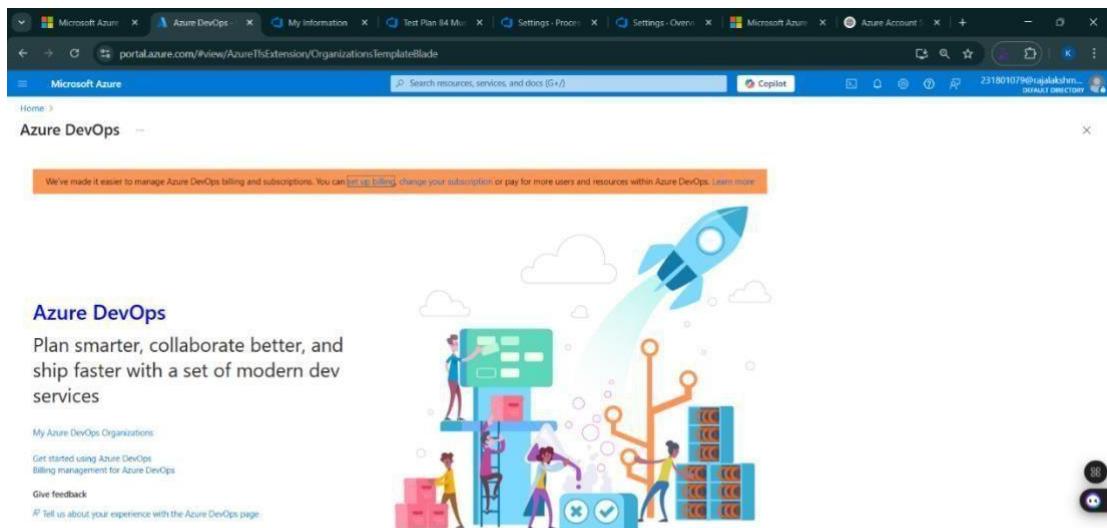
2. Azure home page

The screenshot shows the Microsoft Azure home page. At the top, there's a search bar and a Copilot button. Below the header, there are sections for 'Azure services' (with links to Create a resource, Azure DevOps organizations, Subscriptions, Dashboard hub, Resource groups, Azure Load Testing, Quickstart Center, Azure AI services, Kubernetes services, and More services), 'Resources' (listing recent and favorite resources like 'Music' and 'Music_playlist_Batch_Creator'), 'Navigate' (links to Subscriptions, Resource groups, All resources, and Dashboard), 'Tools' (links to Microsoft Learn, Azure Monitor, Microsoft Defender for Cloud, and Cost Management), and 'Useful links' (links to Azure mobile app and other Azure services). A sidebar on the right shows a user profile and navigation options.

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

The screenshot shows the Microsoft Azure home page with a search bar containing the text 'Azure DevOps'. A dropdown menu is open over the 'Services' section, showing a list of services including 'Azure Native New Relic Service', 'Managed DevOps Pools', 'Azure DevOps organizations', 'Azure Native Dynatrace Service', 'Static Web App', 'Rocky Linux 9', 'Build Agents for Azure DevOps', and 'InfluxDB Cloud (Official Version)'. The 'Azure DevOps organizations' option is highlighted. Other sections visible include 'Resources' (Recent and Favorite resources), 'Tools' (Microsoft Learn, Azure Monitor, Microsoft Defender for Cloud, and Cost Management), and 'Useful links' (Azure mobile app).

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.

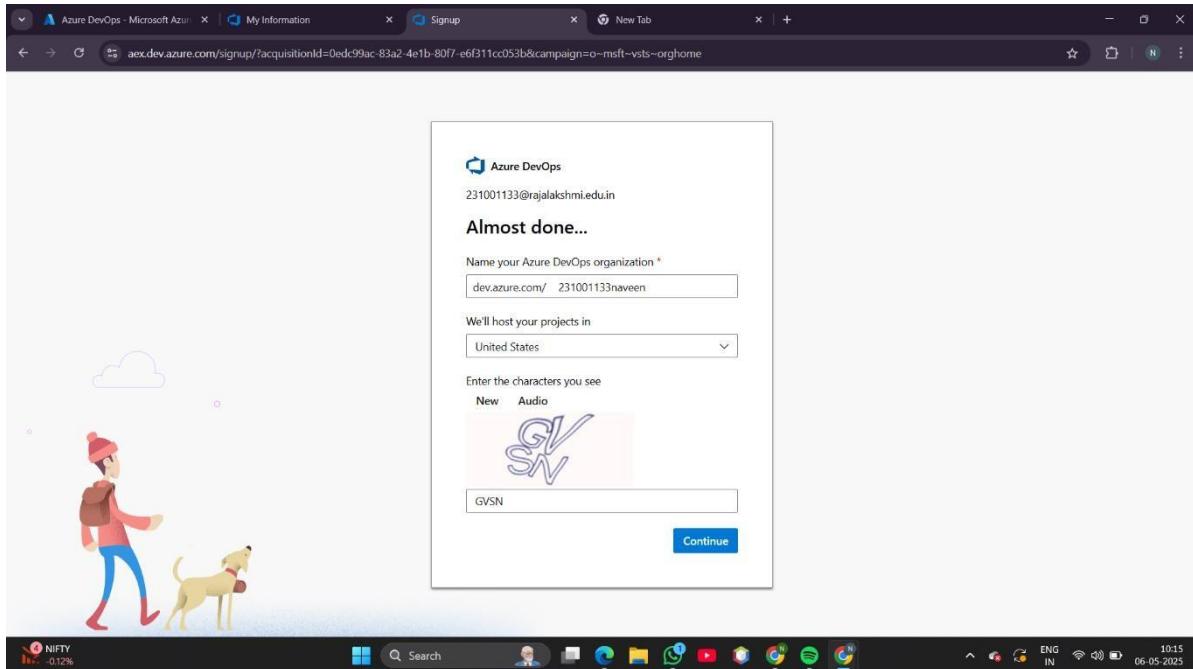
EXP NO: 2

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.

Create new project

X

Project name *

Music Playlist Batch Creator App

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

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.

The screenshot shows the Azure DevOps Organization Home page. At the top, there's a navigation bar with 'Microsoft' on the left and 'Naveen Devadass Sign out' on the right. Below the navigation bar, on the left, is a user profile section for 'Naveen Devadass' with a purple circular icon containing 'ND'. It includes fields for email (231001133@rajalakshmi.edu.in) and location (India). On the right, the main content area is titled 'Azure DevOps Organizations' and shows a tree view of organizations. Under 'dev.azure.com/231001133 (Owner)', there are three projects: 'Music Playlist Batch Creator', 'Boot_Tech(1)', and 'Music Playlist Batch Creator'. An 'Actions' button with 'Open in Visual Studio' is available. Below this, there are links to other organizations: 'dev.azure.com/BootTech (Owner)' and 'dev.azure.com/NAVEEN133 (Owner)'. A 'Create new organization' button is also present. At the bottom of the main content area, there's a 'Visual Studio Dev Essentials' section with a link to 'Use your benefits'.

4. Project dashboard

The screenshot shows the Azure DevOps Project Overview page for the 'Music Playlist Batch Creator' project. The URL in the browser is 'dev.azure.com/231001133/Music%20Playlist%20Batch%20Creator'. The page has a header with the project name 'Music Playlist Batch Creator' and buttons for 'Private' and 'Invite'. On the left, there's a sidebar with navigation links: 'Overview' (selected), 'Summary', 'Dashboards', 'Wiki', 'Boards', 'Repos', 'Pipelines', 'Test Plans', and 'Artifacts'. Below the sidebar, there's a 'Project settings' link. The main content area is divided into several sections: 'About this project' (with a 'Project Overview' section describing the system as a web-based application for creating, managing, and sharing personalized music playlists), 'Project stats' (showing '20 Work items' and '0 Work items' with a 'Period: Last 7 days' dropdown), 'Boards' (a board with 20 work items), and 'Members' (a list of three members with icons: ND, a person, and a mail icon). The bottom of the screen shows a Windows taskbar with various pinned icons and the date/time '06-05-2025 10:20'.

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

The screenshot shows the Azure DevOps interface for the 'Music Playlist Batch Creator' project. The left sidebar is collapsed, showing options like Overview, Boards, Work items, Boards, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The main area is titled 'Music Playlist Batch Creator Team' and shows a 'Backlog' view. At the top right are buttons for '+ New Work Item', 'View as Board', and 'Column Options'. Below is a table with columns: Order, Work Item Type, Title, State, Effort, Busin..., Value Area, and Tags. The backlog contains several items categorized as Epic and Feature, such as 'Playlist Creation and Customization', 'Song Management Within Playlists', and 'Song Recommendations'. All items are marked as 'New' in the State column and 'Business' in the Value Area column.

The screenshot shows the Microsoft sign-in page. At the top right is a 'Sign out' link. In the center, there's a large purple circular profile picture with the letters 'ND' in white. To the right of the picture, the name 'Naveen Devadass' is displayed in bold, along with the email address '231001133@rajalakshmi.edu.in'. Below the name is a blue button labeled 'My Microsoft account'. Further down are links for 'Switch directory' and a three-dot ellipsis. At the bottom left is a circular icon with a person and plus sign, and the text 'Sign in with a different account'.

Result:

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

EXP NO: 3

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

Aim:

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

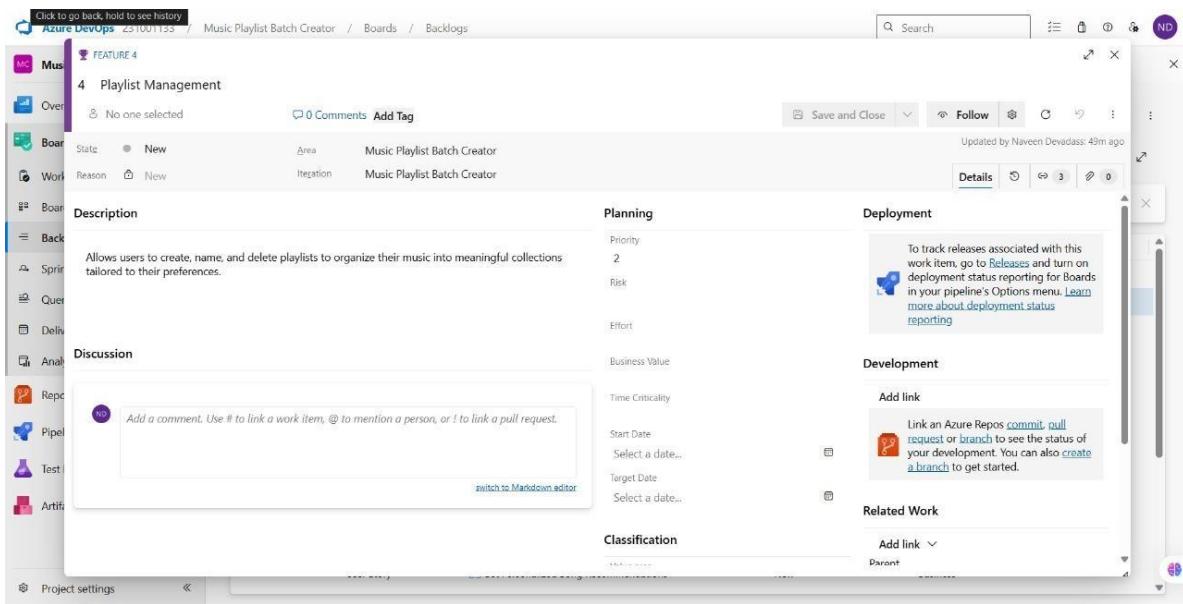
Create Epic, Features, User Stories, Task

The screenshot shows the Azure Boards Backlog view for the 'Music Playlist Batch Creator' team. The backlog is filtered by keyword and displays various work items. The columns include Order, Work Item Type, Title, State, Effort, Business Area, and Tags. The backlog items are categorized under different epics and features, such as 'Playlist Creation and Customization', 'Playlist Management', and 'Song Management Within Playlists'. Most items are marked as 'New'.

1. Fill in Epics

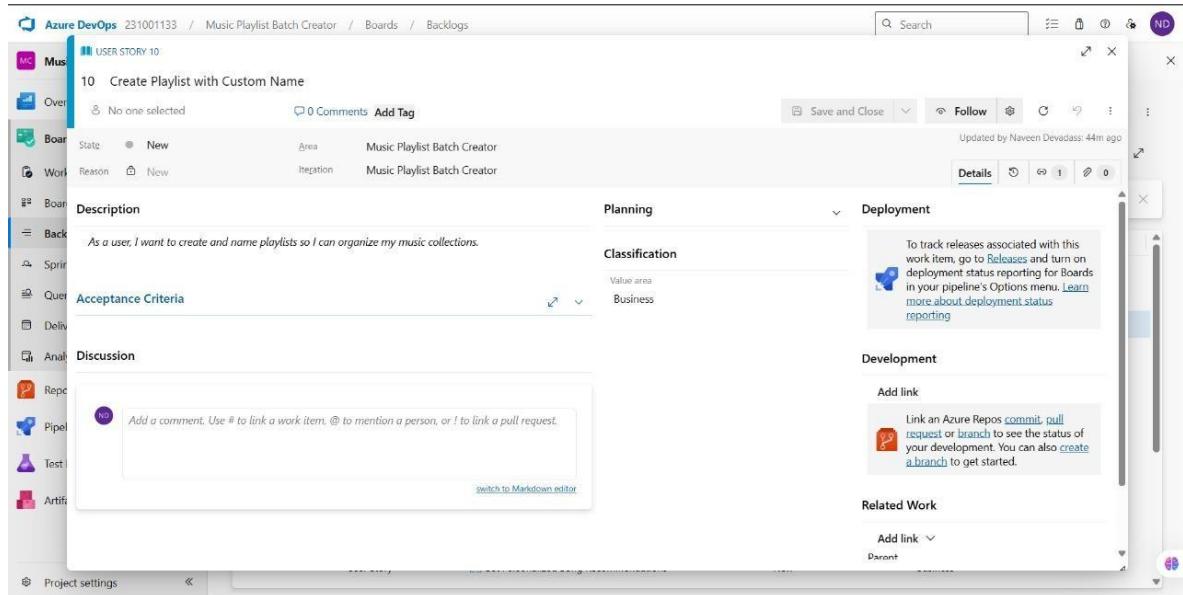
The screenshot shows the 'EPIC 1' creation dialog in Azure DevOps. The epic title is '1 Playlist Creation and Customization'. The dialog includes fields for State (New), Area (Music Playlist Batch Creator), Reason (New), Iteration (Music Playlist Batch Creator), and a Description section. The description text states: 'This epic focuses on enabling users to create, personalize, and manage their music playlists. It includes the core functionalities that allow users to build playlists from scratch, modify the song order, and maintain control over their music collections.' The dialog also includes sections for Planning, Deployment, Development, and Related Work.

2. Fill in Features



The screenshot shows the 'Music Playlist Batch Creator' project in Azure DevOps. A new feature titled '4 Playlist Management' is being created. The 'Description' field contains the text: 'Allows users to create, name, and delete playlists to organize their music into meaningful collections tailored to their preferences.' The 'Planning' section shows a priority of 2 and a risk level of 'Low'. The 'Deployment' section includes a note about tracking releases. The 'Development' section has a placeholder for linking an Azure Repos commit or pull request. The 'Classification' section indicates the value area is 'Business'.

3. Fill in User Story Details



The screenshot shows the 'Music Playlist Batch Creator' project in Azure DevOps. A new user story titled '10 Create Playlist with Custom Name' is being created. The 'Description' field contains the text: 'As a user, I want to create and name playlists so I can organize my music collections.' The 'Acceptance Criteria' section is currently empty. The 'Discussion' field has a placeholder for comments. The 'Planning' section shows a priority of 10 and a risk level of 'Low'. The 'Classification' section indicates the value area is 'Business'. The 'Deployment' section includes a note about tracking releases. The 'Development' section has a placeholder for linking an Azure Repos commit or pull request. The 'Related Work' section is empty.

Result:

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

EXP NO: 4

SPRINT PLANNING

Aim:

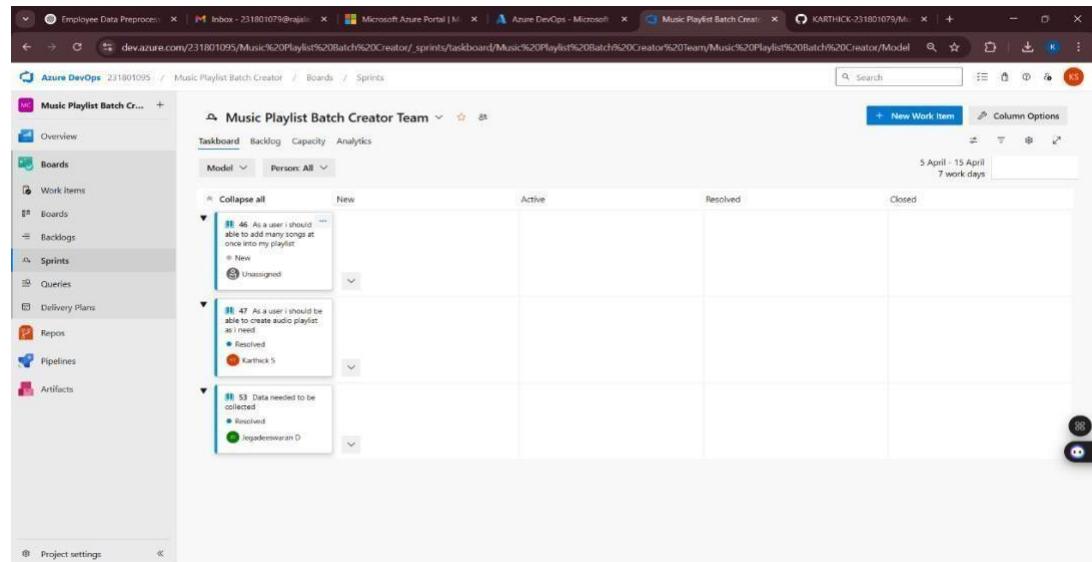
To assign user story to specific sprint for the Music Playlist Batch Creator Project.

Sprint Planning

Sprint 1

The screenshot shows the Azure DevOps Boards Sprints page for the 'Music Playlist Batch Creator' project. The left sidebar is collapsed, showing options like Overview, Boards, Backlogs, Sprints (selected), Queries, Delivery Plans, Repos, Pipelines, and Artifacts. The main area displays a backlog of user stories under the 'Sprint 1' tab. The backlog is organized into columns: New, Active, Resolved, and Closed. Two user stories are visible: '19 As a user I want to sign up and log in securely so that I can access my playlists' and '21 Implement NT- based authentication'. Both stories are marked as 'New' and are assigned to 'Karthikayee Senthil'. A summary at the top right indicates the sprint runs from '21 March - 4 April' with '4 work days remaining'.

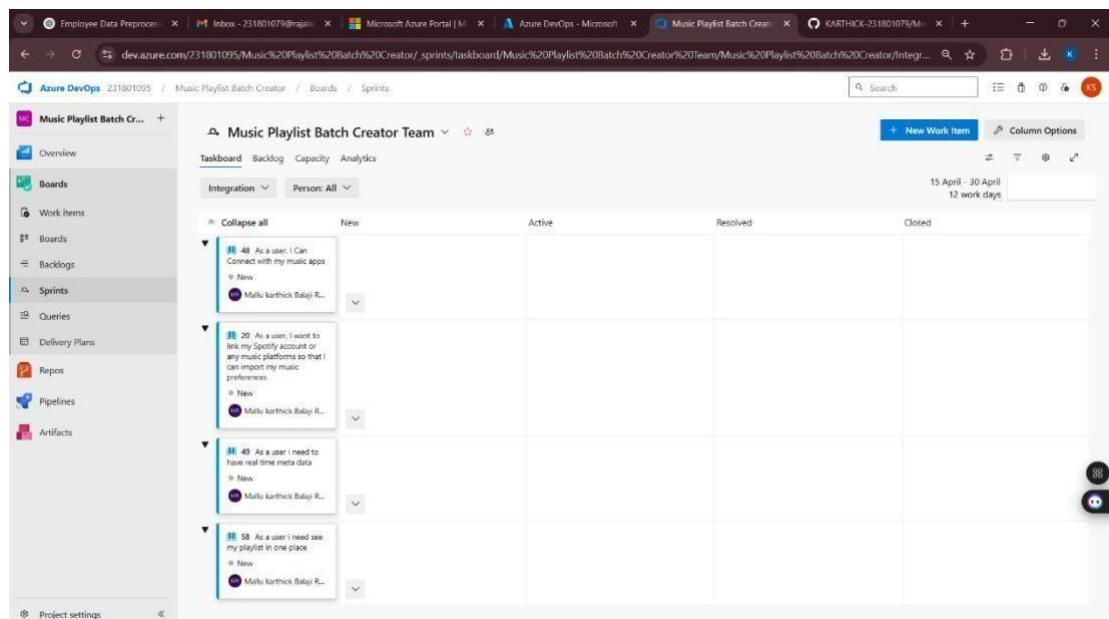
Sprint 2



The screenshot shows the Azure DevOps Taskboard for the 'Music Playlist Batch Creator Team' in Sprint 2. The board has columns for Model, Person: All, New, Active, Resolved, and Closed. There are three work items visible:

- Work item 46: As a user I should be able to add many songs at once into my playlist. Status: New, Assigned to Unassigned.
- Work item 47: As a user I should be able to create audio playlist as I need. Status: Resolved, Assigned to Karthick S.
- Work item 53: Data needed to be collected. Status: Resolved, Assigned to ragadeenwaran D.

Sprint 3



The screenshot shows the Azure DevOps Taskboard for the 'Music Playlist Batch Creator Team' in Sprint 3. The board has columns for Integration, Person: All, New, Active, Resolved, and Closed. There are four work items visible:

- Work item 48: As a user, I can Connect with my music apps. Status: New, Assigned to Mallu karthick Balaji R.
- Work item 20: As a user, I want to link my Spotify account or any other platforms so that I can import my music preferences. Status: New, Assigned to Mallu karthick Balaji R.
- Work item 49: As a user I need to have real time meta data. Status: New, Assigned to Mallu karthick Balaji R.
- Work item 58: As a user I need see my playList in one place. Status: New, Assigned to Mallu karthick Balaji R.

Sprint 4

The screenshot shows the Azure DevOps Taskboard for the 'Music Playlist Batch Creator Team'. The board is organized into columns: New, Active, Resolved, and Closed. There are four items in the 'New' column:

- S1 As a user i should be able to access a friendly and Modern UI
Status: New
Assigned to: Keerthra S.
- S2 As a user i should be able to rename record and change the playlist
Status: New
Assigned to: Unassigned
- S3 As a user i can able to customize my playlist
Status: New
Assigned to: Karthick S.
- S4 As a user i can able to add songs to my playlist
Status: New
Assigned to: Unassigned

The left sidebar shows the project navigation with 'Boards' selected. The top right corner displays the date range '29 April - 30 April' and '2 work days'.

Result:

The Sprints are created for the Music Playlist Batch Creator Project

EXP NO: 5

POKER ESTIMATION

Aim:

Create Poker Estimation for the user stories - Music Playlist Batch Creator Project.

Poker Estimation

The screenshot shows a Microsoft Azure DevOps work item details page for a User Story. The URL in the browser is https://dev.azure.com/231801095/Music%20Playlist%20Batch%20Creator/_backlogs/backlog/Music%20Playlist%20Batch%20Creator%20Team/Epics?workitem=47. The work item ID is 47, and the title is "As a user i should be able to create audio playlist as i need".

Details:

- Assignee:** Kirthick S
- Comments:** 0
- Add Tag:** (button)
- State:** Resolved
- Reason:** Code complete and unit tested
- Area:** Music Playlist Batch Creator
- Iteration:** Music Playlist Batch Creator\Model

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.

Planning:

- Story Points: 3
- Priority: 2
- Risk: (empty)

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: (button)
- Parent: (empty)
- Link to 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:

- 26 Auto-Playlist Creation Based on user preference (link)

Result:

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

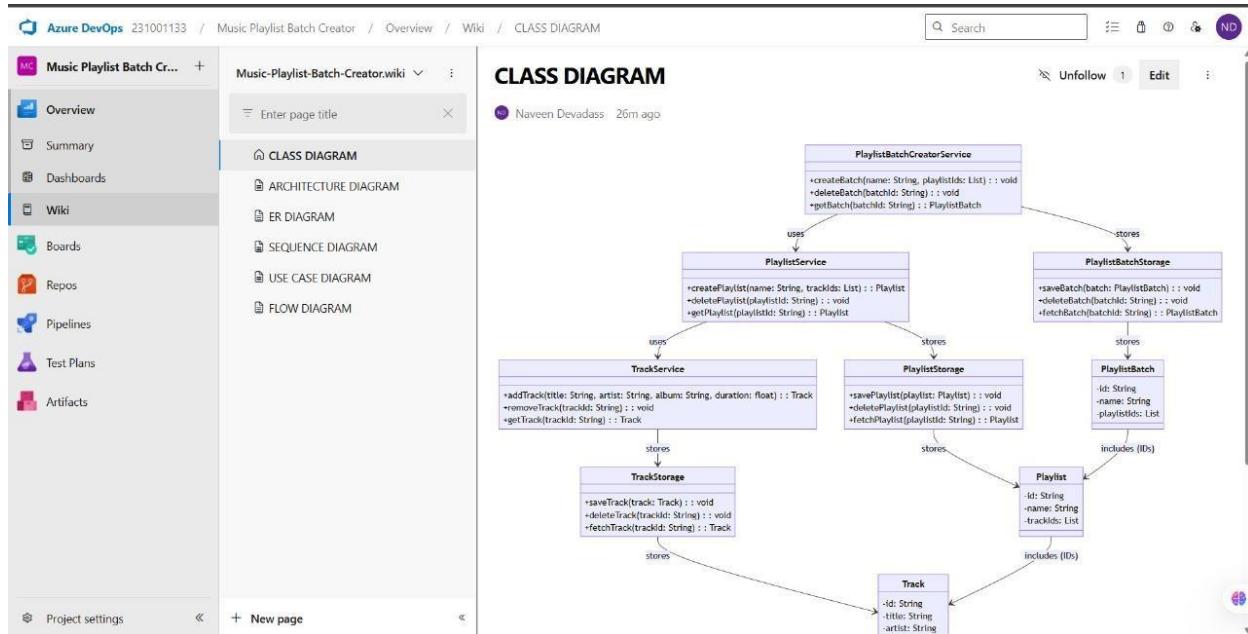
EXP NO: 6

DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE

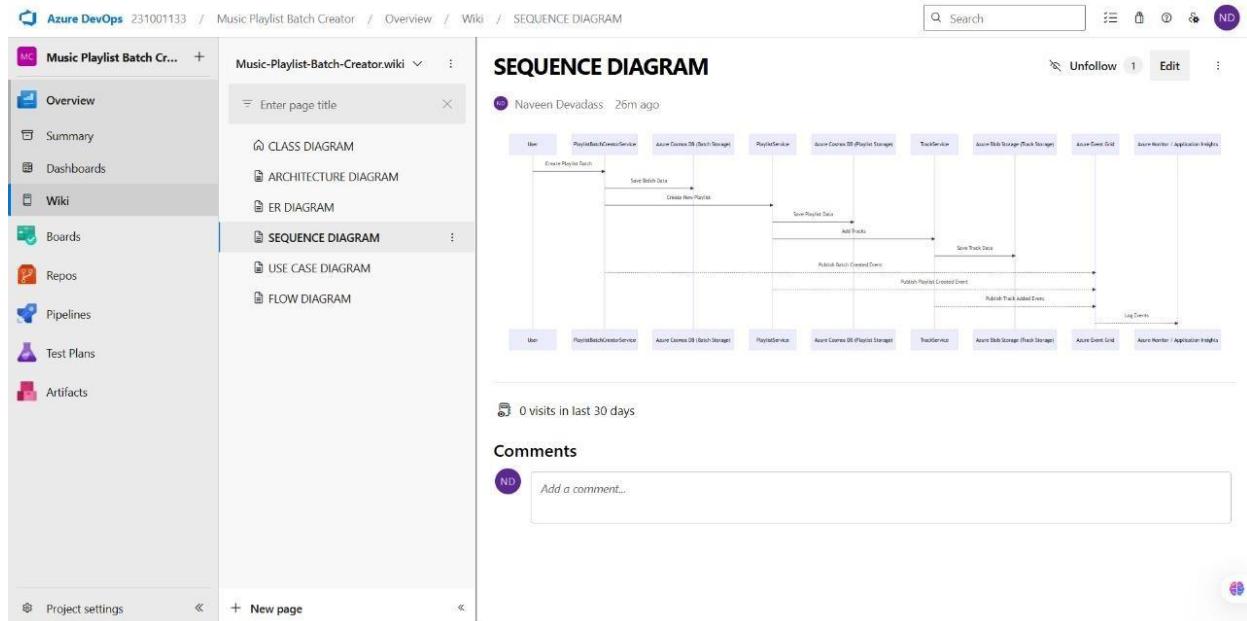
Aim:

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 Music Playlist Batch Creator.

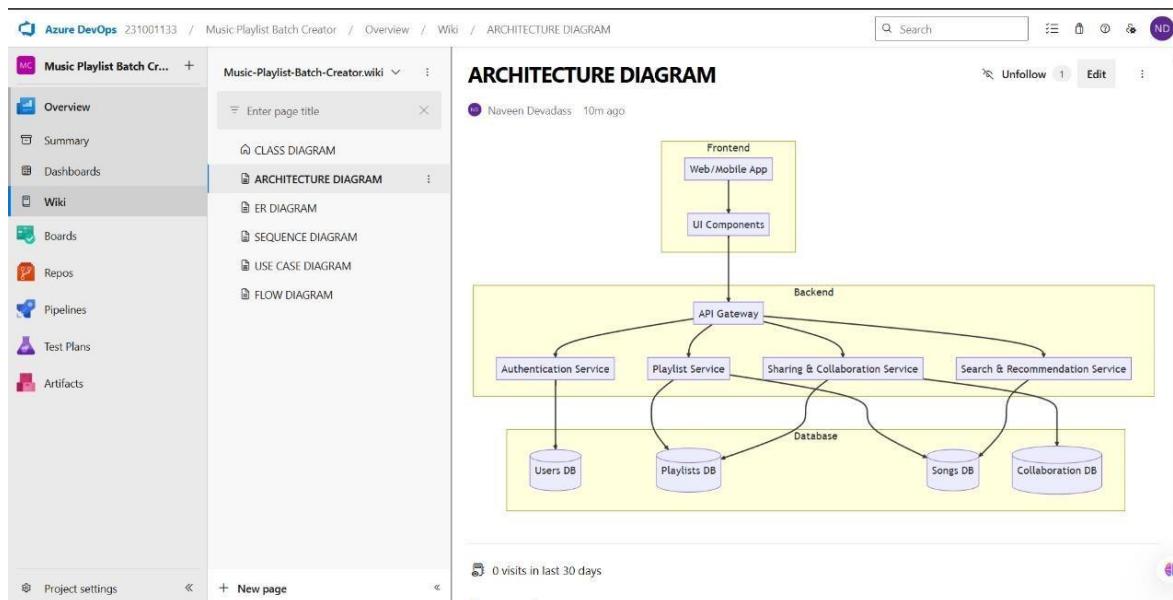
EXP NO: 7

DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE

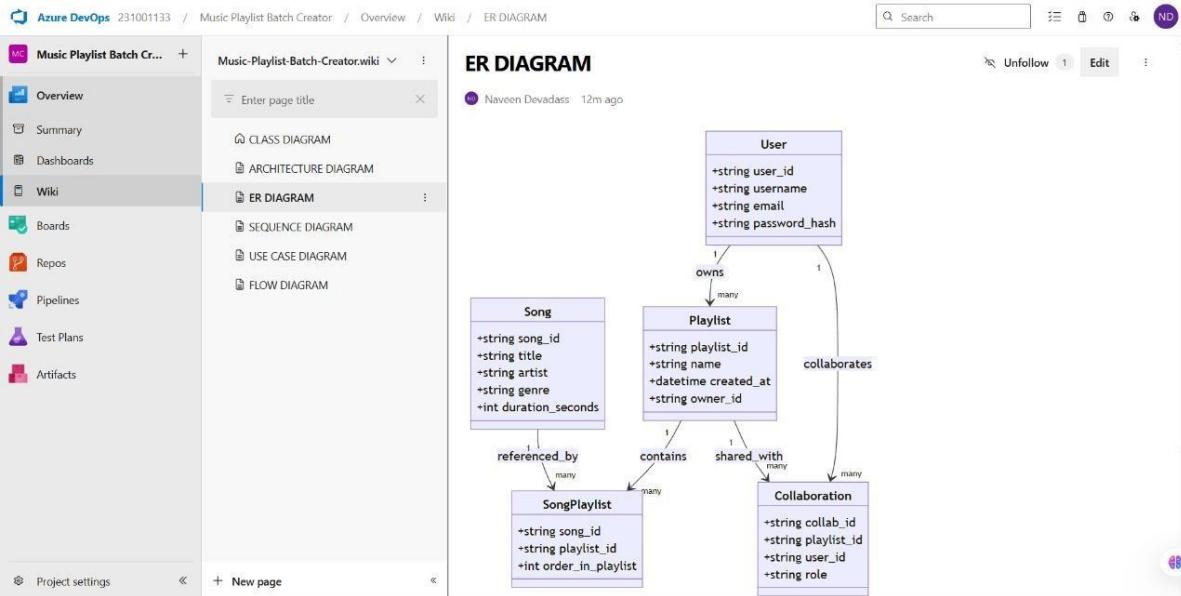
Aim:

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

New Test Plan

Name * Music Playlist Batch Creator - Test Plan

Area Path * Music Playlist Batch Creator

Iteration * Music Playlist Batch Creator/Integration

Create Cancel

2. Test suite

Test Case ID	Title	Order	Assigned To	Status
79	TC01 - Successful Sign Up	1	Karthikeyan Se... Design	
80	TC01 - Existing Email	2	Karthikeyan Se... Design	
81	TC01 - Existing Password	3	Karthikeyan Se... Design	
82	TC01 - Invalid Password	4	Karthikeyan Se... Design	

3. Test case

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

Music Playlist Batch Creator – 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:**
 - All 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

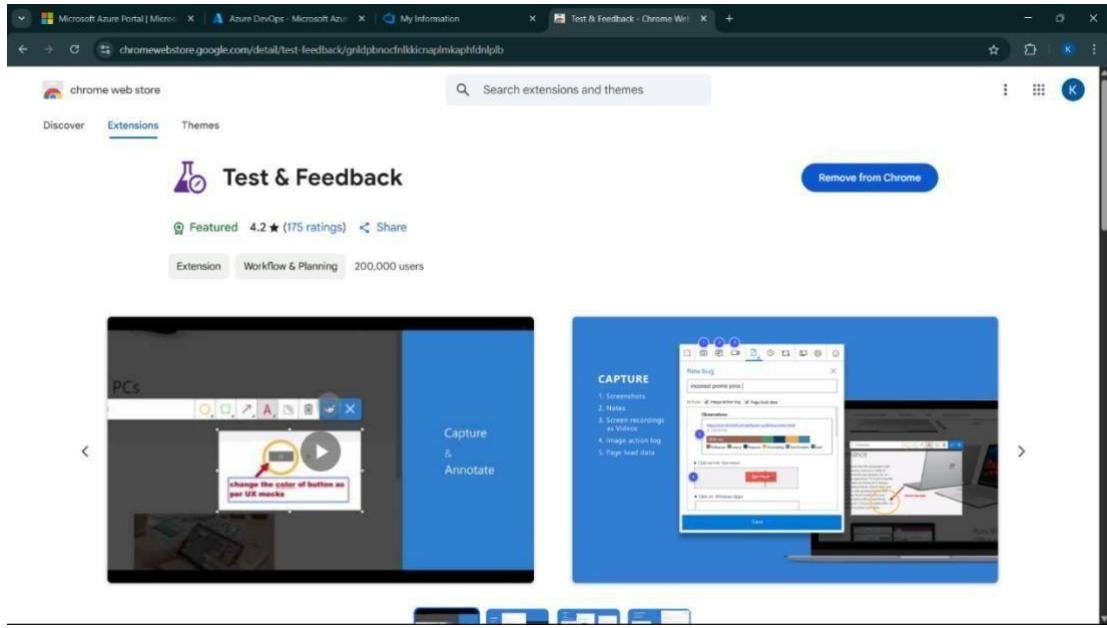
Test Cases

The screenshot shows a Microsoft Azure DevOps interface for a test plan. The URL in the browser is dev.azure.com/231801095/Music%20Playlist%20Batch%20Creator/_testPlans/define?planId=84&suitId=87. The main view displays a test case titled '77 - TC06 - Playlist Loading Failure' assigned to 'Karthick S'. The 'Steps' section contains two actions: 'Disconnect from internet' (Expected result: Network is offline) and 'Navigate to "My Playlists"' (Expected result: Error message "Unable to load playlists" is shown). The 'Status' section indicates a priority of 2 and an automation status of 'Not Automated'. The left sidebar shows navigation links for Music, Overview, Board, Repos, Pipelines, Test Plans, Programs, Parameters, Configuration, Runs, and Artifacts.

A screenshot of the Azure DevOps Test Plan interface. The left sidebar shows 'Music' under 'TEST CASES'. The main view displays a test case titled '75 - TC05 - View Playlist Page' for 'Karthick S'. The 'Steps' tab is selected, showing two steps: 'Log in successfully' (Expected result: User is redirected to dashboard) and 'Navigate to "My Playlists" section' (Expected result: All created playlists are displayed clearly). The 'Status' section indicates a priority of 2 and an automation status of 'Not Automated'. The top right has 'Save and Close', 'Follow', and other navigation buttons.

4. Installation of test

A screenshot of the Chrome Web Store page for the 'Test & Feedback' extension. The extension is listed as 'Featured' with a rating of 4.2 stars and 175 ratings, and 200,000 users. The 'Extensions' tab is selected. The page shows a preview of the extension's features: 'Capture & Annotate' (with a screenshot of a mobile device displaying a button with a red circle and the text 'change the color of button as per UX mocks') and 'CAPTURE' (with a screenshot of a desktop application showing a bug reporting interface with annotations).



Test and feedback

Showing it as an extension

The screenshot shows the Azure DevOps interface with a test plan for 'Music Playlist Batch Creator'. The 'Test Plans' section is active. A modal window titled 'Extensions' is open, displaying a list of extensions available for the project. The 'Test & Feedback' extension is listed among others like 'Copy Text from Picture', 'Dark Reader', 'Monica: ChatGPT AI Assistant', 'Selectext: Copy Text from V...', and 'Test & Feedback'.

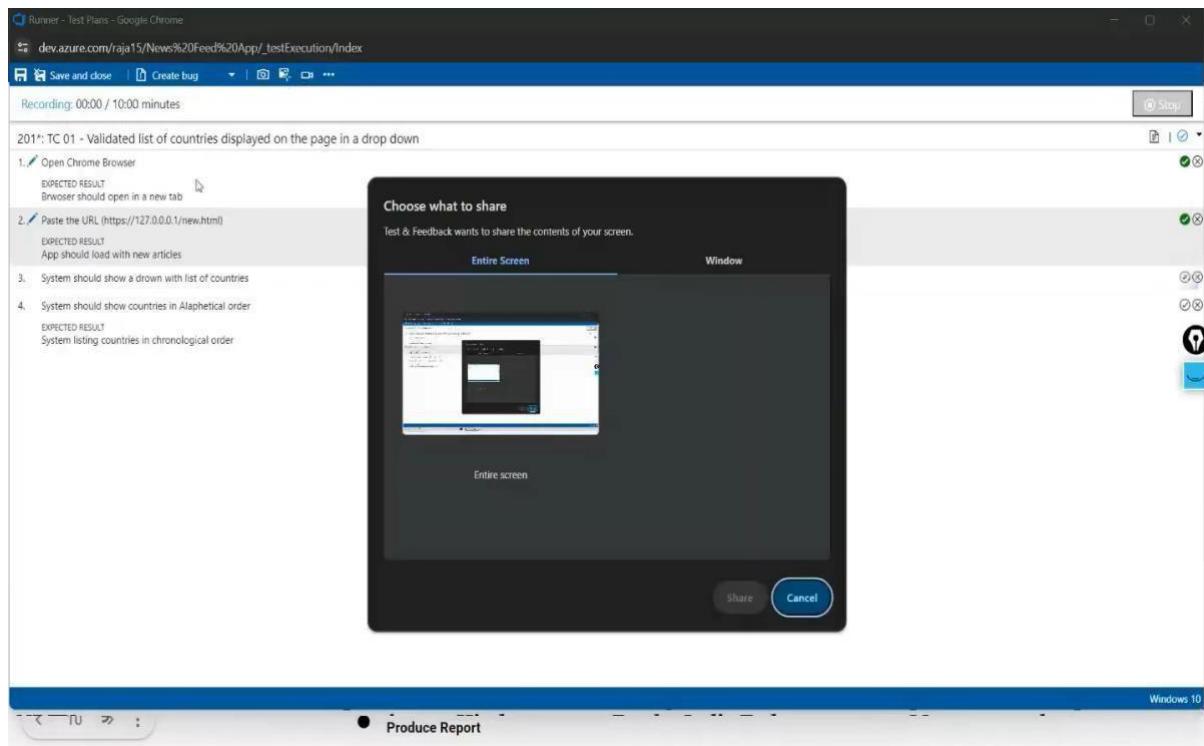
5. Running the test cases

The screenshot displays two windows side-by-side. The left window is the 'Test Plan 84 Music Playlist' page in Azure DevOps, showing a list of test suites under 'Music Playlist Batch Creator - T...'. The 'TS02 - View Playlists (ID: 87)' suite is selected. The right window is a browser session titled 'Runner - Test Plans - Google Chrome' showing a recorded test case named 'TC05 - View Playlist Page'. The test steps are:

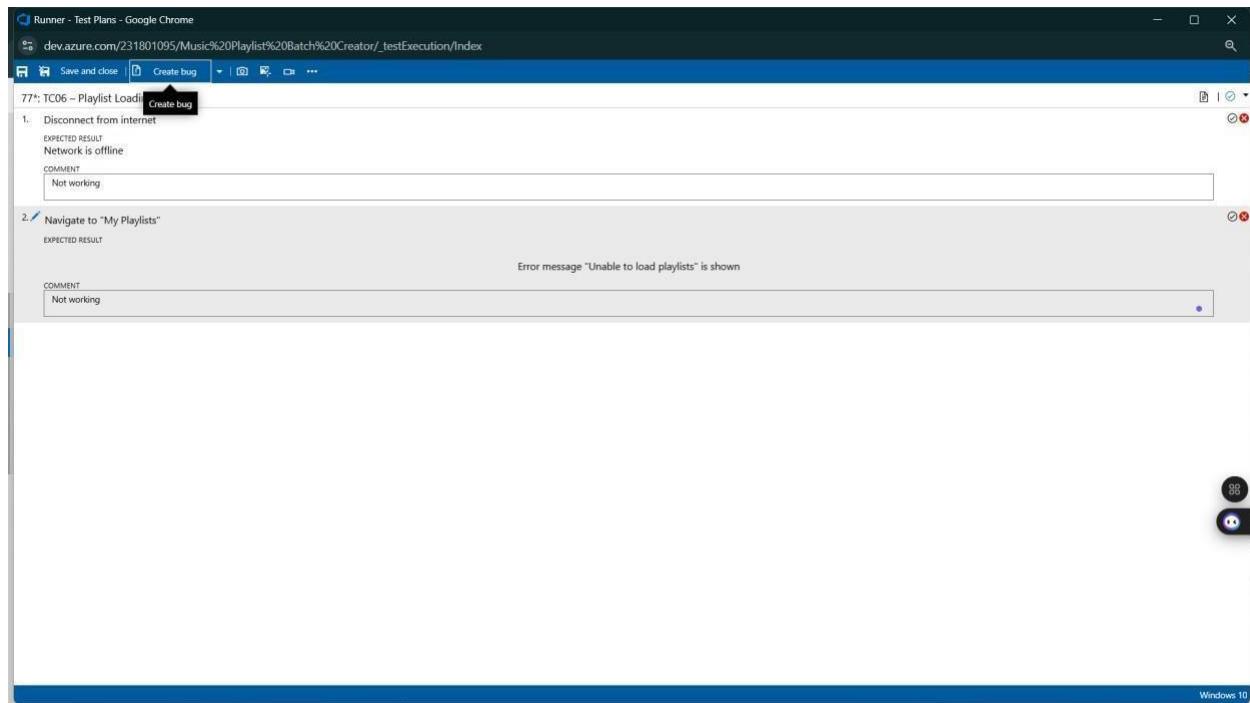
- Log in successfully
EXPECTED RESULT: User is redirected to dashboard
- Navigate to "My Playlists" section
EXPECTED RESULT: All created playlists are displayed clearly

A context menu is open over the second test step, listing options such as 'View execution history', 'Mark Outcome', 'Run', 'Reset test to active', 'Edit test case', 'Assign tester', and 'Run with options'.

6. Recording the test case



7. Creating the bug



8. Test case results

The screenshot shows the Azure DevOps interface for a test plan. On the left, the navigation bar includes 'Overview', 'Boards', 'Repos', 'Pipelines', 'Test Plans', 'Test plans' (selected), 'Progress report', 'Parameters', 'Configurations', 'Runs', and 'Artifacts'. The 'Test Plans' section shows a tree view with 'Music Playlist Batch Creator - T...' expanded, revealing 'TS01 - User Login (4)', 'TS02 - View Playlists (2)' (selected), 'TS03 - Real-Time Met...', 'TS04 - Playlist Editing (4)', and 'TS05 - Smart Playlist ...'. The main content area displays 'TS02 - View Playlists (ID: 87)' with tabs for 'Define', 'Execute' (selected), and 'Chart'. Under 'Test Points (2 items)', there are two entries: 'TC05 - View Playlist Page' (selected) and 'TC06 - Playlist Loading Failure'. A modal window titled 'TC05 - View Playlist Page' shows 'Test Case Results' with columns: Outcome, TimeStamp, Configuration, Run by, Tester, and Test Plan. The results are as follows:

Outcome	TimeStamp	Configuration	Run by	Tester	Test Plan
Passed	4m ago	Windows 10	Karthick S	Malu karthick Balaji ... Music	
Passed	12m ago	Windows 10	Karthick S	Malu karthick Balaji ... Music	
Not Applicable	12m ago	Windows 10	Karthick S	Malu karthick Balaji ... Music	
Passed	14m ago	Windows 10	Karthick S	Malu karthick Balaji ... Music	
Passed	Tuesday	Windows 10	Karthikeyan Senthil	Malu karthick Balaji ... Music	
Passed	Saturday	Windows 10	Malli karthick Balaji ...	Malu karthick Balaji ... Music	
Failed	Saturday	Windows 10	Malli karthick Balaji ...	Malu karthick Balaji ... Music	
Passed	Apr 11	Windows 10	Karthick S	Malu karthick Balaji ... Music	
Passed	Apr 11	Windows 10	Karthick S	Malu karthick Balaji ... Music	

[Open execution history for current test point](#)

9. Test report summary

The screenshot shows the Azure DevOps interface for work items. The navigation bar includes 'Overview', 'Boards', 'Work items' (selected), 'Backlogs', 'Sprints', 'Queries', 'Delivery Plans', 'Analytics views', 'Repos', 'Pipelines', 'Test Plans', 'Artifacts', and 'Project settings'. The main content area displays a work item for 'BUG 203: BG 01 - Countries Drop down Not Available on the page'. The work item details are as follows:

- Title:** BUG 203: BG 01 - Countries Drop down Not Available on the page
- Assignee:** rajesh prabhu
- Comments:** 0 Comments
- Tags:** Add Tag
- State:** New
- Reason:** New
- Iteration:** News Feed App
- Repro Step:** Active
- Result:** Passed
- Title:** Open Chrome Browser
- Expected Result:** Browser should open in a new tab
- Step no.:** 1.
- Result:** Passed
- Title:** Paste the URL (<https://127.0.0.1/new.html>)
- Expected Result:** App should load with new articles
- Step no.:** 2.
- Result:** Failed
- Title:** Section should show a dropdown with list of countries
- Actual Result:** Produce Report

Planning:

- Resolved Reason: [Details](#)
- Story Points: [Details](#)
- Priority: 2
- Severity: 3 - Medium
- Activity: [Details](#)

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:

- Assigning bug to the developer and changing state

Run 48 - TS02 - View Playlists (Manual) / TC06 - Playlist Loading Failure

92 TB01 - Playlist loading spinner keeps spinning indefinitely on poor network

Karthick S

Repro Steps:

- 1. Failed: Disconnect from internet. Expected Result: Network is offline. Comments: Page Not loading. Navigate to "My Playlists". Expected Result: Error message "Unable to load playlists" is shown.
- 2. Failed: Navigate to "My Playlists". Expected Result: Error message "Unable to load playlists" is shown.

Test Configuration: Windows 10

Planning: Resolved Reason: Bug filed on "TC06 - Playlist Loading Failure". Story Points: 2. Priority: 2. Severity: 3 - Medium. Activity: 18-04-2025 03:23.

Deployment: To track releases associated with this work item, go to Boards in your pipeline's Options menu. Learn more about deployment status reporting.

Development: 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. Titled By: 77 TC06 - Playlist Loading Failure. Updated: 10-04-2025. Design.

System Info:

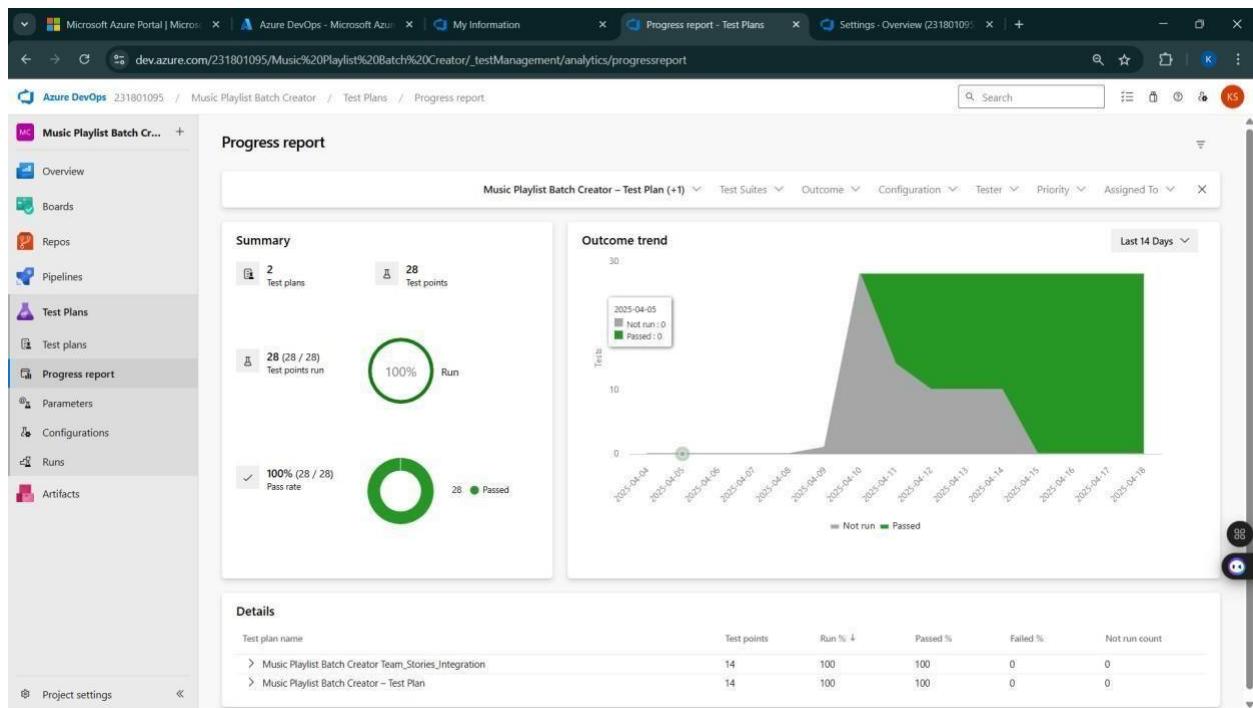
10. Progress report

Progress report

Summary: 1 Test plans, 14 Test points, 14 (14 / 14) Test points run, 100% Run, 100% Pass rate.

Outcome trend: Last 14 Days. Shows a sharp increase in test runs starting around April 10th.

Test plan name	Test points	Run %	Passed %	Failed %	Not run count
Music Playlist Batch Creator – Test Plan	14	100	100	0	0
TS01 - User Login	4	100	100	0	0
TS02 - View Playlists	2	100	100	0	0
TS03 - Real-Time Metadata	2	100	100	0	0
TS04 - Playlist Editing	4	100	100	0	0
TS05 - Smart Playlist Creation	2	100	100	0	0



11. Changing the test template

All processes

Name	Description	Team projects
Basic	This template is flexible for any process and great for teams getting started with Azure DevOps.	0
Agile	This template is flexible and will work great for most teams using Agile planning methods, including those practic...	0
Scrum	This template is for teams who follow the Scrum framework.	0
CMMI	This template is for more formal projects requiring a framework for process improvement and an auditable recor...	0

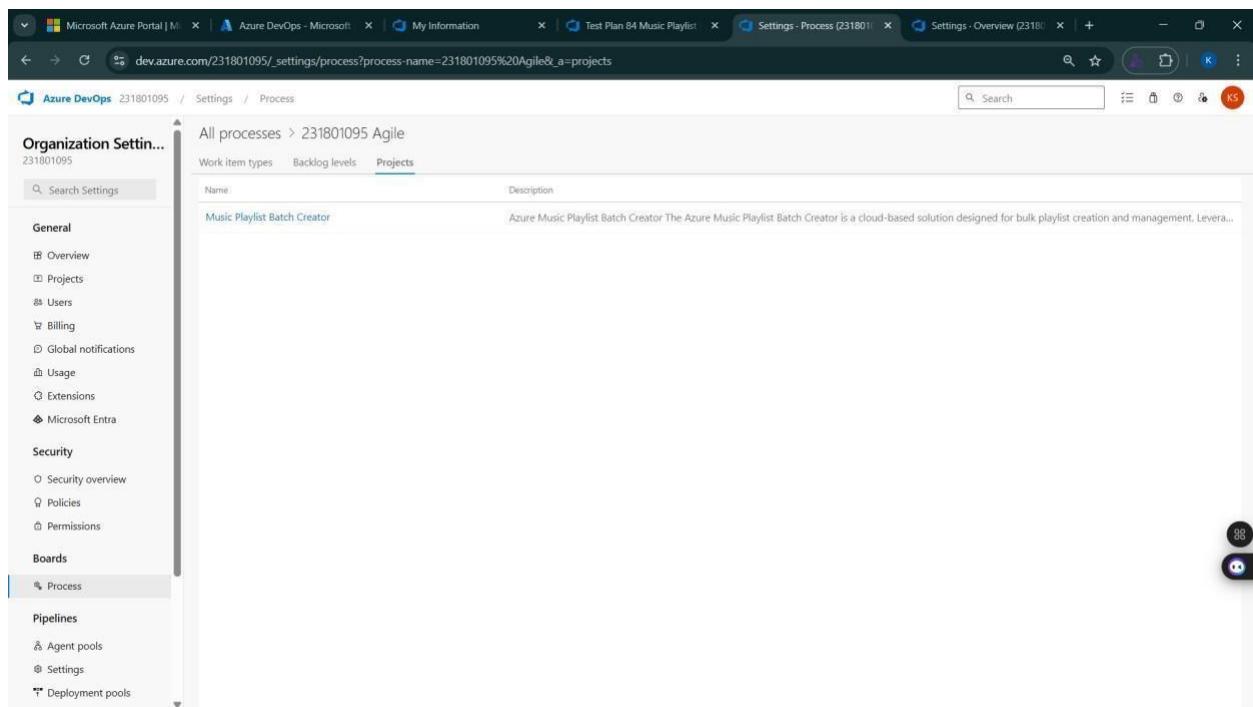
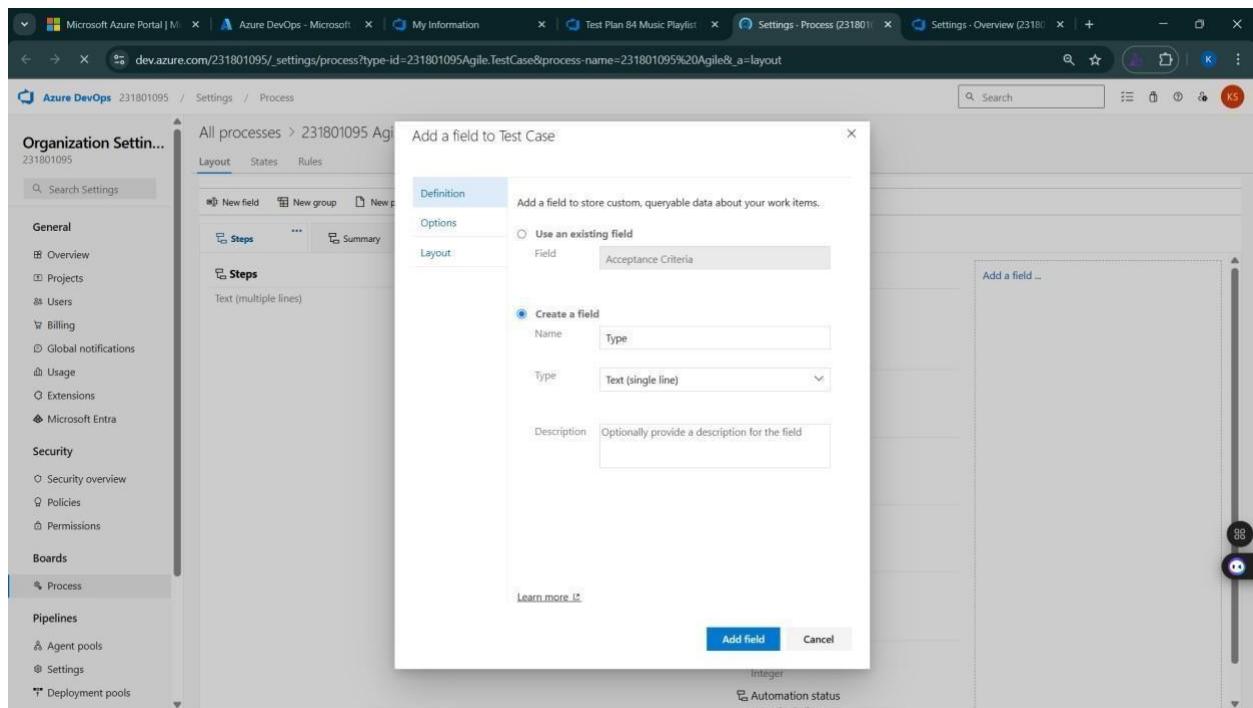
The screenshot shows the 'All processes' list in the Azure DevOps Settings - Process section. The 'Agile' template is selected, indicated by a highlighted row. The list includes:

Name	Description	Team projects
Basic	This template is flexible for any process and great for teams getting started with Azure DevOps.	0
Agile	... This template is flexible and will work great for most teams using Agile planning methods, including those pract...	0
Scrum	This template is for teams who follow the Scrum framework.	0
CMMI	... This template is for more formal projects requiring a framework for process improvement and an auditable reco...	0

The screenshot shows the 'All processes' list in the Azure DevOps Settings - Process section. A new entry, '231801095 Agile (default)', has been added under the Agile category. The list includes:

Name	Description	Team projects
Basic	This template is flexible for any process and great for teams getting started with Azure DevOps.	0
Agile	... This template is flexible and will work great for most teams using Agile planning methods, including those pract...	0
231801095 Agile (default)	... 1	1
Agile Plus	...	0
Scrum	This template is for teams who follow the Scrum framework.	0
CMMI	... This template is for more formal projects requiring a framework for process improvement and an auditable reco...	0

12. View the new test case template



The screenshot shows the Azure DevOps Settings - Process page. The URL in the address bar is dev.azure.com/231801095/_settings/process?type-id=231801095Agile.TestCase&process-name=231801095%20Agile&a=layout. The page title is "Settings - Process (231801095)". The left sidebar shows "Organization Settings" for project ID 231801095, with "Process" selected. The main content area displays a form for creating a new field. The field name is "Steps", which is a "Text (multiple lines)" type. Other fields shown include "Recent test results", "Deployment", "Development", "Related Work", and "Status". A search bar at the top right contains the text "Search".

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

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:

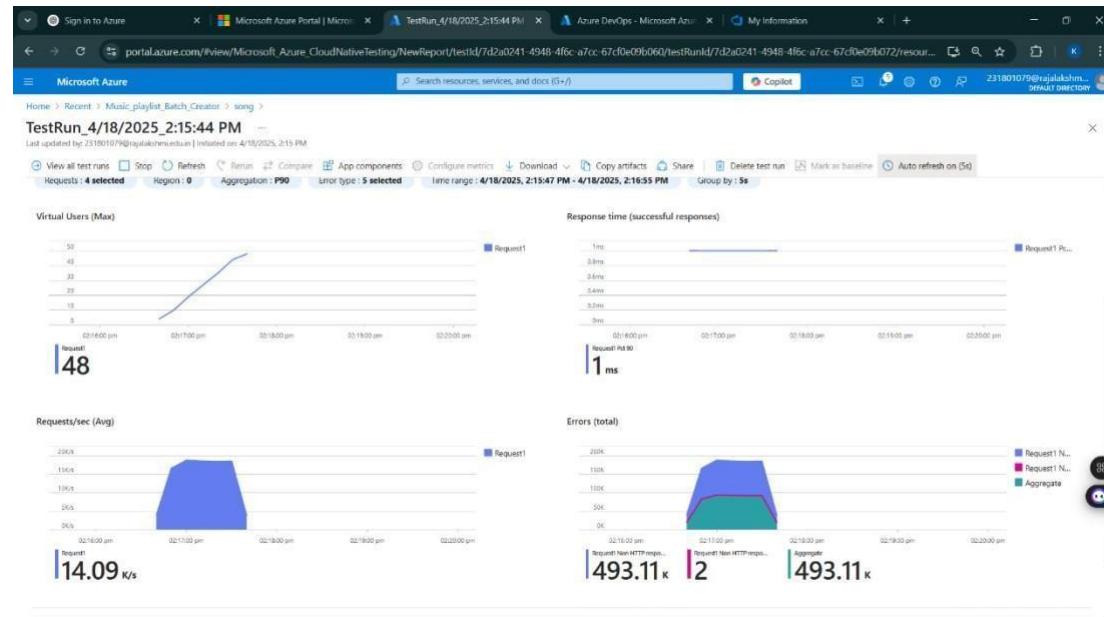
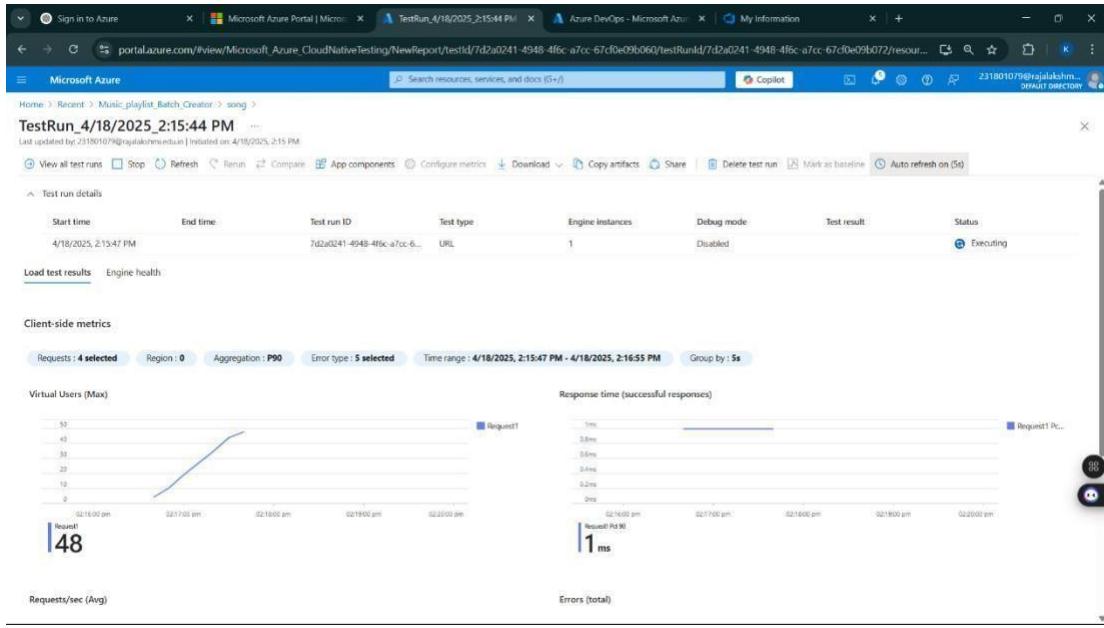
1. Sign in to Azure Portal
Go to <https://portal.azure.com> and log in.
2. Create the Resource Go to *Create a resource* → Search for “Azure Load Testing”.
 Select Azure Load Testing and click Create.
3. Fill in the Configuration Details Subscription: Choose your Azure subscription. Resource Group: Create 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.

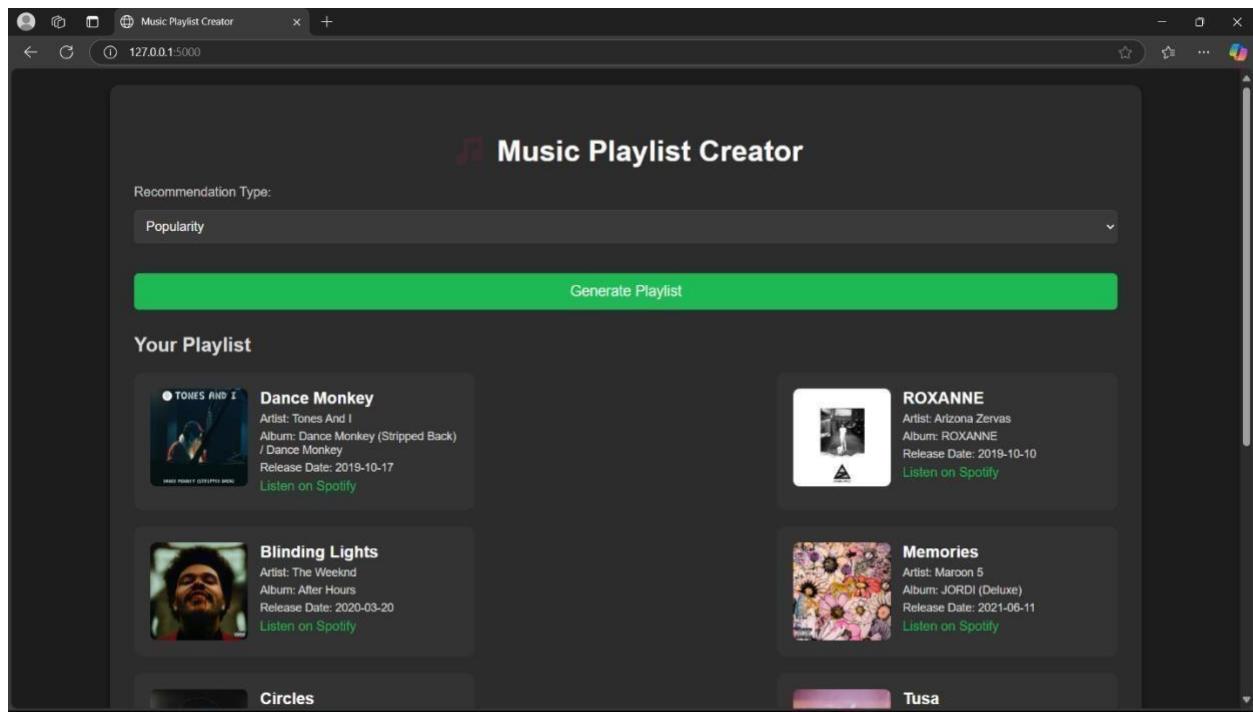
Steps to Create and Run a Load Test:

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





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 Music Playlist Batch Creator project.

GitHub Project Structure

The screenshot shows a GitHub repository page for 'KARTHICK-231801079/Music-Playlist-Batch-Creator'. The main area displays a list of commits, each with a file icon, the file name, a brief description, and the time of upload. The commits include various files like 'Diagram', 'Load Testing', 'Poker Estimation', 'Progress Report', 'Sprints', 'Testing-Test Plans and Test Cases', 'azure accountpic', 'static', 'templates', 'Backlog.pic.png', 'Music_Recommendation.py', 'app.py', and 'genre_similarity_mapping.pkl'. The right sidebar provides a 'Repo Summary' with a summary of the repository's activity, including 0 stars, 1 watching, and 0 forks. It also shows sections for 'About', 'Releases' (no releases), and 'Packages' (no packages published).

File	Description	Time
231801095 diagram	Add files via upload	2 weeks ago
Diagram 231801079	Add files via upload	2 weeks ago
Load Testing	Add files via upload	3 days ago
Poker Estimation	Add files via upload	2 weeks ago
Progress Report	Add files via upload	3 days ago
Sprints	Add files via upload	2 weeks ago
Testing-Test Plans and Test Cases	Add files via upload	3 days ago
azure accountpic	Add files via upload	2 months ago
static	Add files via upload	1 minute ago
templates	Add files via upload	1 minute ago
Backlog.pic.png	Rename Screenshot (227).png to Backlog.pic.png	2 months ago
Music_Recommendation.py	Add files via upload	now
app.py	Add files via upload	1 minute ago
genre_similarity_mapping.pkl	Add files via upload	1 minute ago

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.

EXP NO : 12

DATE : 12.04.2025

TESTING – TEST PLAN, TEST CASE, LOAD TESTING

AIM:

To design and manage structured test plans and test cases in Azure DevOps for validating core user stories through both happy path and error scenarios and evaluate the performance of the application's endpoint by creating and executing load tests using Azure Load Testing.

PROCEDURE:

TEST CASE DESIGN PROCEDURE

1. Understand Core Features of the Application

- Review requirement documents and user stories.
- Identify all main functionalities of the application.
- Ensure complete coverage of modules before test case creation.

2. Define User Interactions

- Determine common user behaviors based on application flow.
- Translate user actions into testable scenarios.
- Ensure each test case mimics a real user operation.

3. Design Happy Path Test Cases

- Create test cases for expected and correct user actions.
- Ensure each functionality works under normal conditions.
- Add these cases under the relevant Test Suite in Azure DevOps.

4. Design Error Path Test Cases

- Identify edge cases, invalid inputs, and system failures.
- Test how the system handles incorrect or unexpected behavior.
- Add these test cases to the same or a separate Test Suite in Azure DevOps.

5. Break Down Steps and Expected Results

- Write step-by-step instructions in the "Steps" section of the test case.
- Provide expected results for each action.

- Ensure clarity for both manual execution and automation mapping.

6. Use Clear Naming and IDs

- Name test cases clearly using a defined naming convention (e.g., TC01, TC02, etc.).
- Ensure titles reflect the purpose of the test case.
- Azure DevOps auto-generates test case IDs for tracking.

7. Separate Test Suites

- Group test cases based on functionality (e.g., Login, Playlist, Recommendations).
- Use Static, Requirement-based, or Query-based suites in Azure DevOps.
- Improves traceability and execution flow.

8. Prioritize and Review

- Mark test cases with priority (High, Medium, Low).
- Review test cases for completeness and correctness.
- Ensure alignment with associated user stories or features.

1. New test plan

The screenshot shows the Azure DevOps interface for managing test plans. On the left, there's a sidebar with project navigation: Overview, Boards, Repos, Pipelines, Test Plans (selected), Test plans, Progress report, Parameters, Configurations, Runs, and Artifacts. Below this is a 'Project settings' section. The main area displays a 'Test Suites' card for 'Demo.Day' (ID: 79). The card shows a summary: 'May 5 - May 12', '100% run, 100% passed', and a link to 'View report'. Below the card is a 'Test Cases (1 item)' table:

	Order	Test Case Id	Assigned To	State
<input type="checkbox"/> Title	1	00	Ska M	Design
<input type="checkbox"/> join Group and Post in Group				

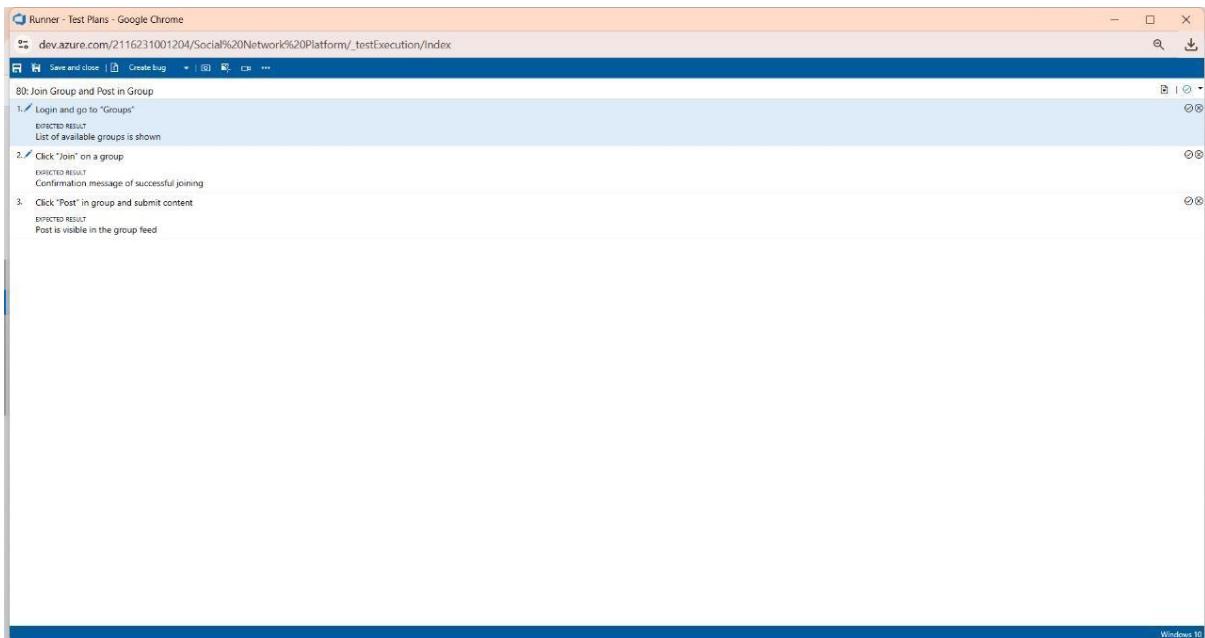
2. Test case

The screenshot shows a Microsoft Azure DevOps Test Case details page for 'TEST CASE BD'. The test case title is '80 Join Group and Post in Group'. The test case is in the 'Design' state and is associated with the 'Social Network Platform' area and iteration. It has 0 comments and no tags. The 'Steps' section contains three steps: 1. Login and go to 'Groups'. Expected result: List of available groups is shown. 2. Click 'Join' on a group. Expected result: Confirmation message of successful joining. 3. Click 'Post' in group and submit content. Expected result: Post is visible in the group feed. Below the steps, there are sections for 'Deployment', 'Development', 'Related Work', and 'Status'. The 'Deployment' section includes a note about tracking releases. The 'Development' section includes a note about linking to Azure Repos. The 'Related Work' section lists a work item titled '81 Post Not Displayed in Group Feed After Submission' with a status of 'Resolved'. The 'Status' section shows a priority level of 'Normal'.

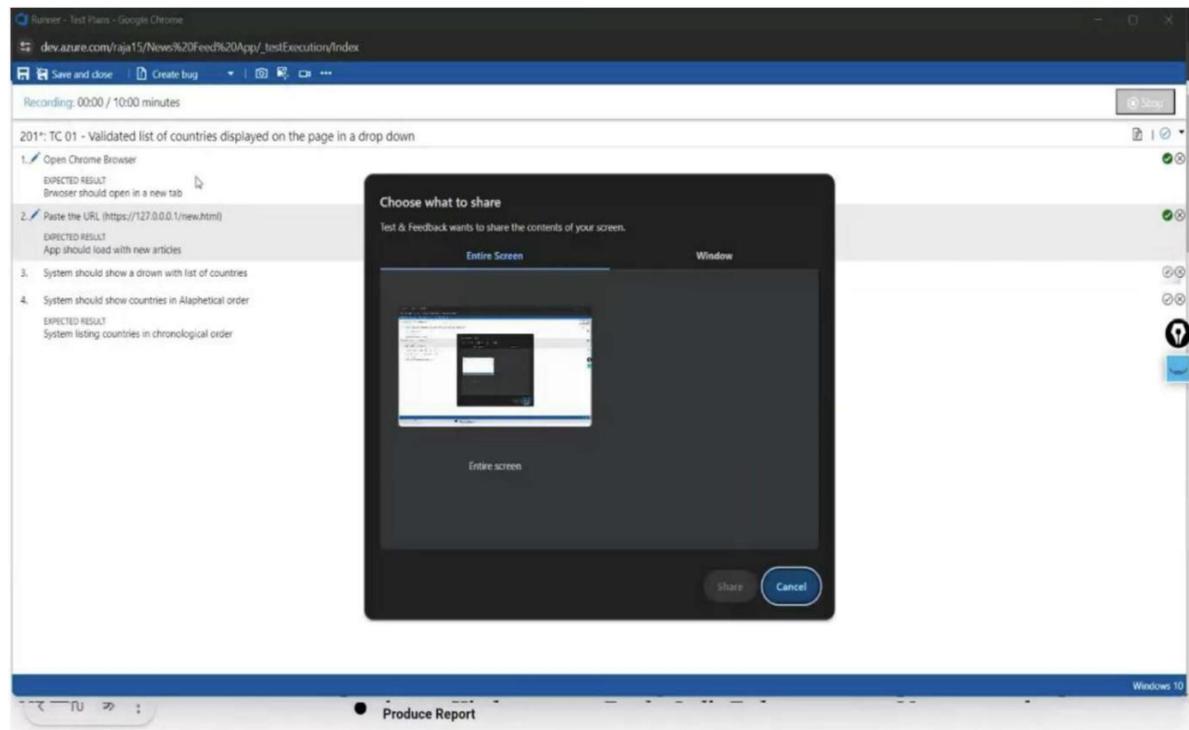
3. Installation of Test

The screenshot shows the Chrome Web Store page for the 'Test & Feedback' extension. The extension has a rating of 4.2 stars from 175 ratings and 200,000 users. It is categorized under 'Extension' and 'Workflow & Planning'. The page features two main screenshots: one showing a PC interface with a red annotation box and another showing a mobile device interface with a red annotation box. A large 'Add to Chrome' button is prominently displayed at the top right.

4. Running the Test Cases



5. Recording the Test Cases



6. Creating Bugs

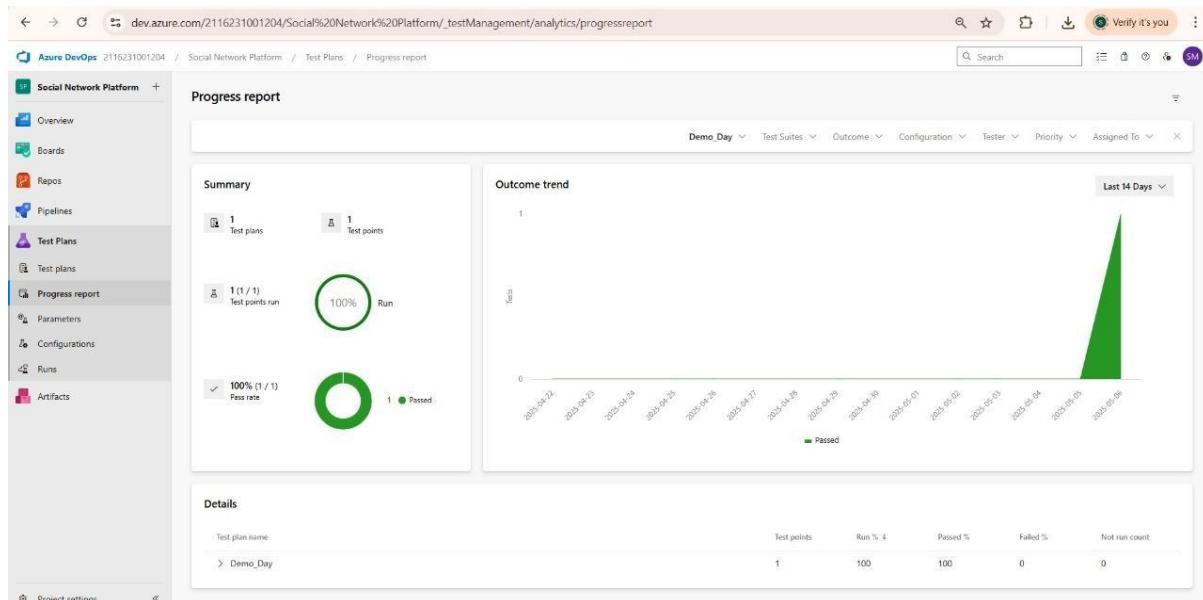
The screenshot shows the Azure DevOps interface for creating a work item. The left sidebar is titled "Social Network Platform" and includes sections for Overview, Boards, Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The main area displays a bug work item titled "81 Post Not Displayed in Group Feed After Submission" created by "Siva M". The work item details include State: New, Area: Social Network Platform, Reason: Not fixed, Iteration: Social Network Platform, and a repro step dated 5/6/2025 4:18 AM. The right side of the screen shows tabs for Planning, Deployment, Development, and Related Work, along with various status and link management options.

7. Test Case Results

The screenshot shows the "Test Case Results" section of the TCOS application. It displays a table of test results with columns: Outcome, TimeStamp, Configuration, Run by, Tester, and Test PI. The table contains 10 rows of data, mostly "Passed" outcomes, with one "Failed" outcome. The "Tester" column lists names like Karthick S, Mallu karthick Balaji, and Karthik Senthil. The "Test PI" column lists Music, Music, Music, Music, Music, Music, Music, Music, Music, and Music. At the bottom of the table, there is a link: "Open execution history for current test point".

Outcome	TimeStamp	Configuration	Run by	Tester	Test PI
Passed	4m ago	Windows 10	Karthick S	Mallu karthick Balaji	Music
Passed	12m ago	Windows 10	Karthick S	Mallu karthick Balaji	Music
Not Applicable	12m ago	Windows 10	Karthick S	Mallu karthick Balaji	Music
Passed	14m ago	Windows 10	Karthick S	Mallu karthick Balaji	Music
Passed	Tuesday	Windows 10	Karthik Senthil	Mallu karthick Balaji	Music
Passed	Saturday	Windows 10	Mallu karthick Balaji	Mallu karthick Balaji	Music
Failed	Saturday	Windows 10	Mallu karthick Balaji	Mallu karthick Balaji	Music
Passed	Apr 11	Windows 10	Karthick S	Mallu karthick Balaji	Music
Passed	Apr 11	Windows 10	Karthick S	Mallu karthick Balaji	Music

8. Progress Report



LOAD TESTING PROCEDURE :

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

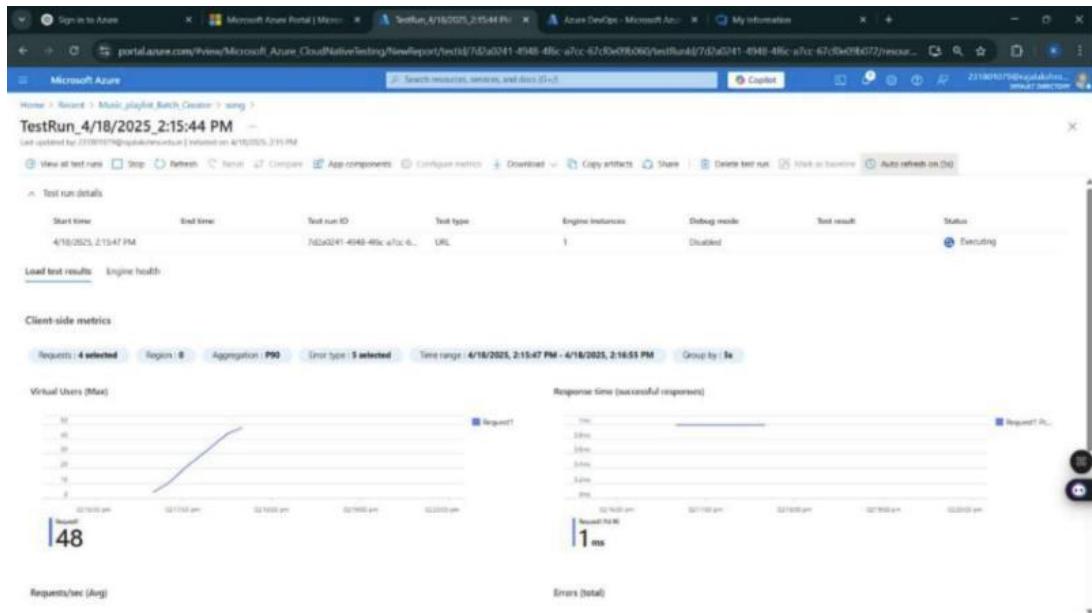
- Test Name: Provide a unique name.
- Description: (Optional) Add test purpose.
- Run After Creation: Keep checked.

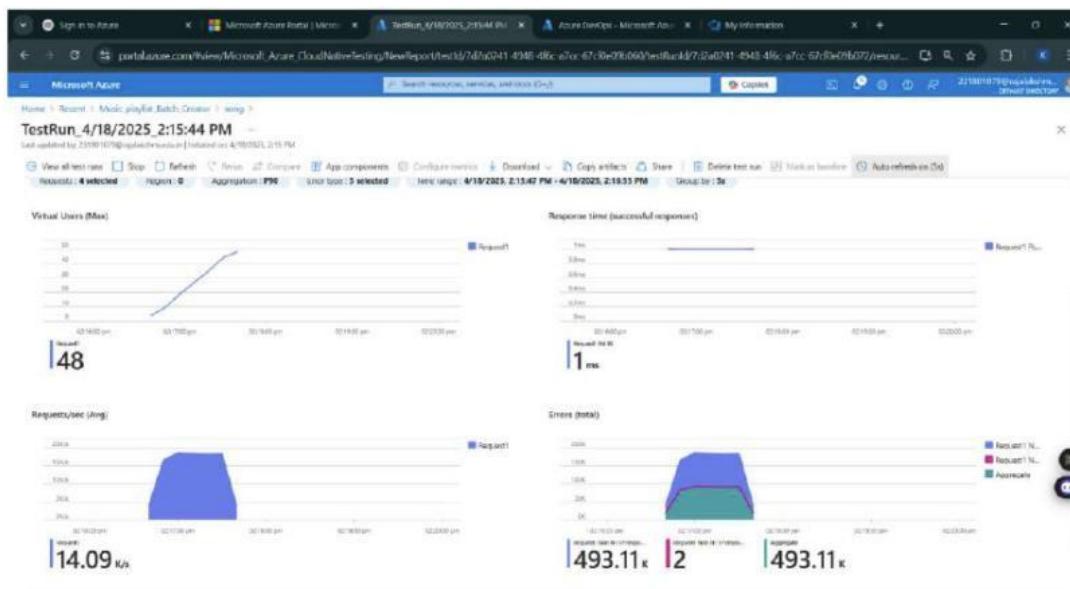
3. Load Settings

- Test URL: Enter the target endpoint (e.g., https://yourapi.com/products).

4. Click Review + Create — Create to start the test.

Load Testing





RESULT:

Test plans and test cases for selected user stories were created in Azure DevOps, covering both happy and error paths and an Azure Load Testing resource was also set up, and a load test was successfully run to evaluate the performance of the target endpoint.