



RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE LAB MANUAL

CS23432 – Software Construction

(REGULATION 2023)

RAJALAKSHMI ENGINEERING COLLEGE
Thandalam, Chennai-602015

Name: Keerthna S

Register No: 231801083

Year / Branch / Section: 2nd / AI&DS / FA

Semester: IV

Academic Year: 2024 - 2025

INDEX

S.No.	Date	Title	Page No
1.	21/1/25	Azure Devops Environment Setup.	3
2.	21/1/25	Azure Devops Project Setup and User Story Management.	6
3.	11/2/25	Setting Up Epics, Features, And User Stories for Project Planning.	11
4.	18/2/25	Sprint Planning.	14
5.	25/2/25	Poker Estimation.	17
6.	04/3/25	Designing Class and Sequence Diagrams for Project Architecture.	18
7.	25/3/25	Designing Architectural and ER Diagrams for Project Structure.	20
8.	15/4/25	Testing – Test Plans and Test Cases.	22
9.	22/4/25	Load Testing and Pipelines.	39
10.	22/4/25	GitHub: Project Structure & Naming Conventions.	44

EXP NO: 1

AZURE DEVOPS ENVIRONMENT SETUP

Aim:

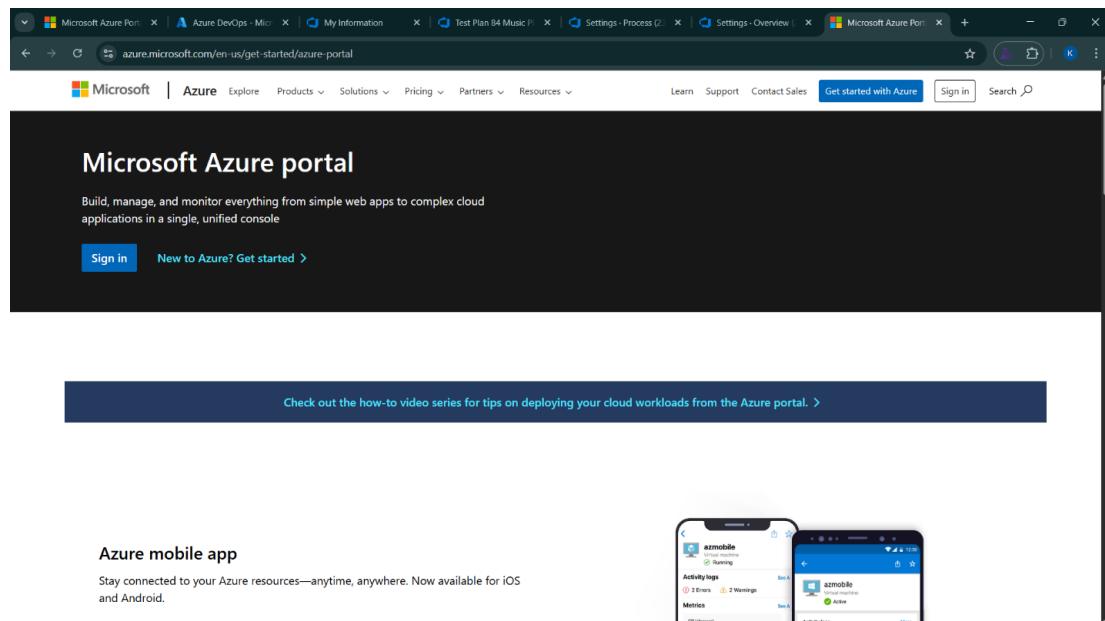
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

INSTALLATION

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



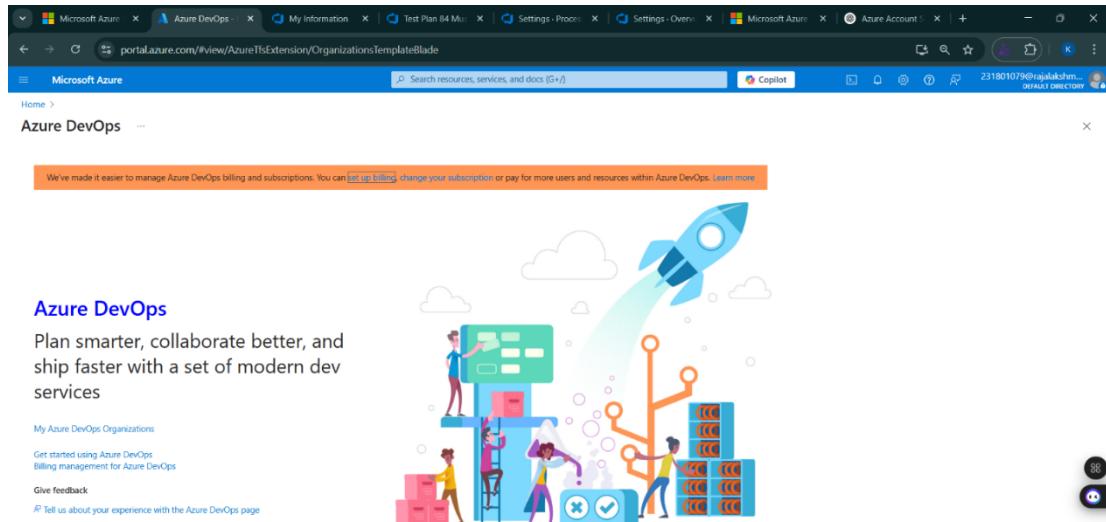
2. Azure home page

The screenshot shows the Microsoft Azure home page. At the top right, there is a user profile with the email 231801081@rajalakshmi.edu.in and a message about a multi-factor authentication sign-in. Below the header, there's a search bar and a Copilot button. The main area is divided into sections: 'Azure services' (Create a resource, Azure Load Testing, Subscriptions, Quickstart Center, Azure AI services, Kubernetes services, Virtual machines, App Services, Storage accounts, More services), 'Resources' (Recent and Favorite items like MusicPlaylistBatchCreator, Resource group, and Subscription), 'Navigate' (links to Subscriptions, Resource groups, All resources, and Dashboard), and 'Tools' (Microsoft Learn, Azure Monitor, Microsoft Defender for Cloud, and Cost Management).

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

This screenshot is similar to the previous one but includes a search overlay in the center. The search term 'DevOps' is entered, and the results show 'All' (Services (7), Marketplace (31), More (4)). Under 'Services', items like 'Azure Native New Relic Service', 'Managed DevOps Pools', and 'Azure DevOps organizations' are listed. Under 'Marketplace', items like 'Static Web App', 'Rocky Linux 9', and 'Build Agents for Azure DevOps' are shown. The rest of the interface is identical to the first screenshot.

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.

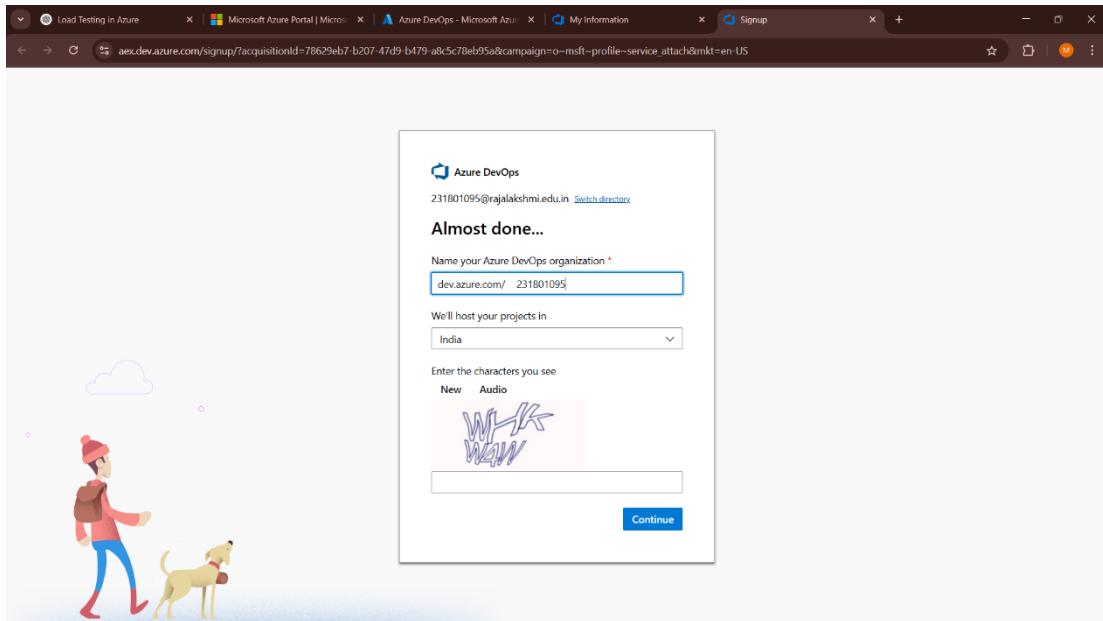
EXP NO: 2

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

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 *

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.

By creating this project, you agree to the Azure DevOps [code of conduct](#)

Advanced

Version control [?](#)

Git

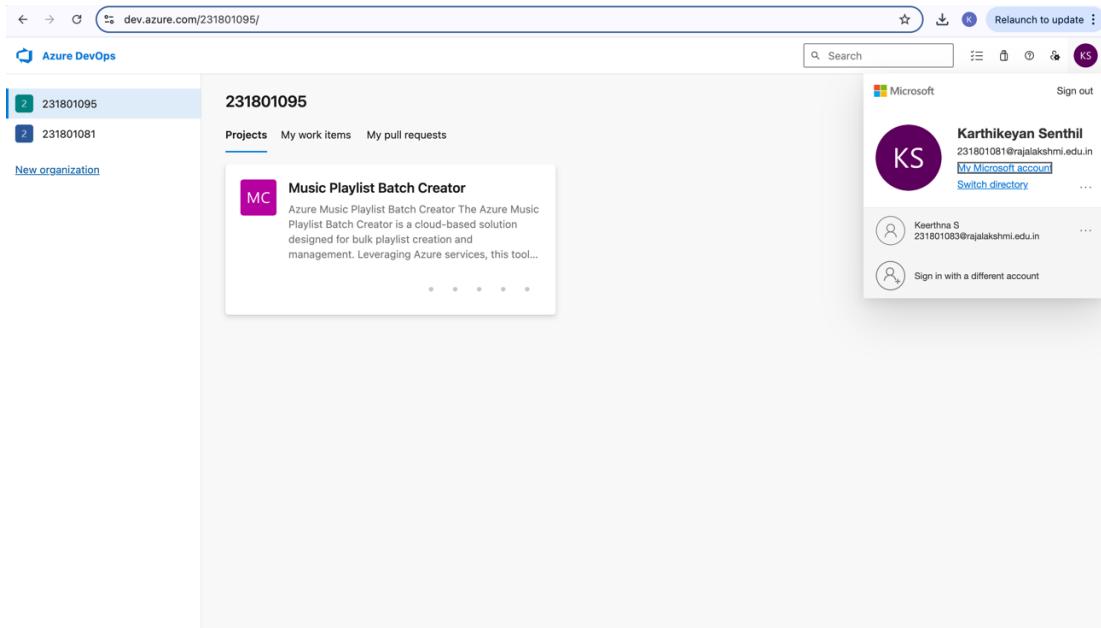
Work item process [?](#)

231801095 Agile

Cancel Create

The screenshot shows the 'Create new project' dialog box. At the top, it says 'Create new project'. Below that is a 'Project name *' field containing 'Music Playlist Batch Creator'. There is also a 'Description' field which is empty. Under 'Visibility', there are two options: 'Public' (selected) and 'Private'. The 'Public' option is described as allowing anyone on the internet to view the project, while 'Private' allows only people given access to view it. Below this, a note states that by creating the project, the user agrees to the Azure DevOps code of conduct. Under 'Advanced', there are dropdown menus for 'Version control' (set to 'Git') and 'Work item process' (set to '231801095 Agile'). At the bottom right are 'Cancel' and 'Create' buttons.

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

A screenshot of the Azure DevOps project summary page for 'Music Playlist Batch Creator'. The left sidebar shows the project navigation with 'Overview' selected. The main content area includes sections for 'About this project' (describing the tool as a cloud-based solution for bulk playlist creation), 'Project stats' (showing 1 work item and 0 work items), and 'Members' (listing five team members with their initials: KS, MK, JP, Ks, and Ks). The top navigation bar shows multiple tabs open, including 'My Information', 'Summary', 'Settings - Pro', 'Settings - Org', and 'Google Search'.

5. To manage user stories:

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

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

The screenshot shows the Azure DevOps Boards Backlog page. The left sidebar includes options like Overview, Boards, Work items, Backlogs (selected), Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The main area displays a backlog titled 'Music Playlist Batch Creator Team'. It lists six user stories with columns for Order, Work Item Type, Title, State, Effort, Story Points, Business Value, Value Area, and Tags. The first story is an Epic titled 'User Authentication & Profile Management'.

Order	Work Item Type	Title	State	Effort	Story Points	Business Value	Value Area	Tags
1	Epic	> User Authentication & Profile Management	New				Business	
2	Epic	> Develop a system that allows users to create and manage m...	New				Business	
3	Epic	> Fetch and integrate music data from external sources for pl...	New				Business	
4	Epic	> Enable users to edit, customize, and share their playlists wit...	New				Business	
5	Epic	> Improve system performance and enhance user experience	New				Business	
6	Epic	> Test Epic	New				Business	

The screenshot shows the Microsoft account sign-in page. At the top, there are icons for a star, download, and profile (labeled 'KS'). A blue button says 'Relaunch to update ::'. Below that is a search bar with the placeholder 'Search'. To the right are icons for a list, a bag, a question mark, a gear, and another profile icon (labeled 'KS'). The Microsoft logo is on the left, and 'Sign out' is on the right. In the center, there's a large purple circular profile picture with the letters 'KS'. The user information displayed is 'Karthikeyan Senthil' and the email '231801081@rajalakshmi.edu.in'. There are links for 'My Microsoft account' and 'Switch directory'.

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 DevOps interface for a project titled "Music Playlist Batch Creator Team". The left sidebar navigation includes "Overview", "Boards", "Work items", "Backlogs", "Sprints", "Queries", "Delivery Plans", "Analytics views", "Repos", "Pipelines", "Test Plans", and "Artifacts". The main area displays a backlog board titled "Backlog - Analytics". The backlog table has columns for Order, Work Item Type, Title, State, Effort, Story Points, Business Value, Value Area, and Tags. The backlog contains the following items:

Order	Work Item Type	Title	State	Effort	Story Points	Business Value	Value Area	Tags
1	Epic	User Authentication & Profile Management	New				Business	
2	Epic	Develop a system that allows users to create and manage m...	New				Business	
	Feature	Auto-Playlist Creation Based on user preference	New				Business	
	User Story	As a user, I should be able to create playlists based on...	Resolved		3		Business	
	Feature	Bulk Song Addition	New				Business	
	Feature	Data Collection	New				Business	
3	Epic	Fetch and Integrate music data from external sources for pl...	New				Business	
4	Epic	Enable users to edit, customize, and share their playlists wit...	New				Business	
5	Epic	Improve system performance and enhance user experience	New				Business	
6	Epic	Test Epic	New				Business	

1.Fill in Epics

The screenshot shows the Azure DevOps interface for a project titled "Music Playlist Batch Creator". The left sidebar navigation includes "Overview", "Boards", "Work items", "Backlogs", "Sprints", "Queries", "Delivery Plans", and "Analytics views". The main area displays a work item details page for an epic named "EPIC 13 - User Authentication & Profile Management". The work item details include:

- Recently updated**: EPIC 13
- Title**: 13 - User Authentication & Profile Management
- Assignee**: Karthikeyan Senthil
- Comments**: 0
- Add Tag**
- Status**: New
- Area**: Music Playlist Batch Creator
- Reason**: New
- Iteration**: Music Playlist Batch Creator\Iteration 3
- Save**, **Follow**, **Details** buttons

The "Description" section is empty with a placeholder "Click to add Description.". The "Planning" section includes fields for Priority (2), Risk, Business Value, Time Criticality, Start Date, and Target Date. The "Deployment" section provides instructions for tracking releases. The "Development" section includes an "Add link" button and a note about linking to Azure Repos.

2.Fill in Features

The screenshot shows the Azure DevOps interface for a work item titled "13 User Authentication & Profile Management". The work item is categorized under "EPIC 13" and assigned to "Karthikeyan Senthil". It has 0 comments and no tags. The status is "New" and the reason is also "New". The area is "Music Playlist Batch Creator" and the iteration is "Music Playlist Batch Creator\Iteration 3".

Description: Click to add Description.

Planning: Priority: 2, Risk: , Business Value: , Time Criticality: .

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

Development: Add link: Link an Azure Repos [commit](#), [pull request](#) or [branch](#) to see the status of your development. You can also [create a branch](#) to get started.

3.Fill in User Story Details

The screenshot shows the same work item "13 User Authentication & Profile Management" with additional details filled in. The "Start Date" and "Target Date" fields now have dropdown menus showing "Select a date...".

Classification: Value area: Business

Related Work: Add link: Add an existing work item as a parent. Child: A list of related work items including "57 Homepage Implementation", "17 OAuth Integration (Google, Spotify)", "59 Playlist Creation Page", "61 Trending Song Page", and "14 User Registration & Login".

Result:

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

EXP NO: 4

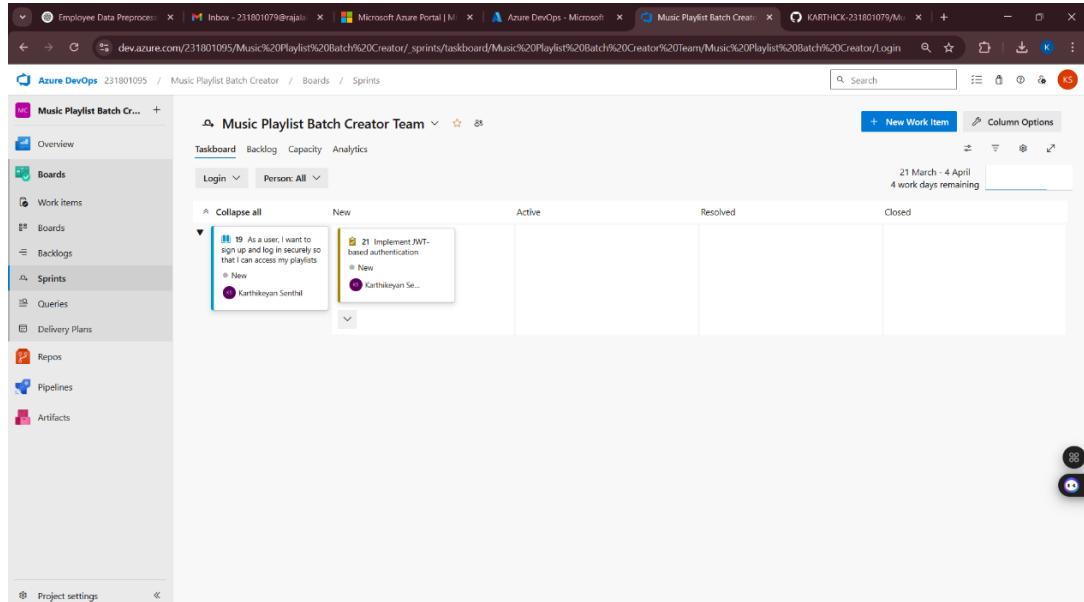
SPRINT PLANNING

Aim:

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

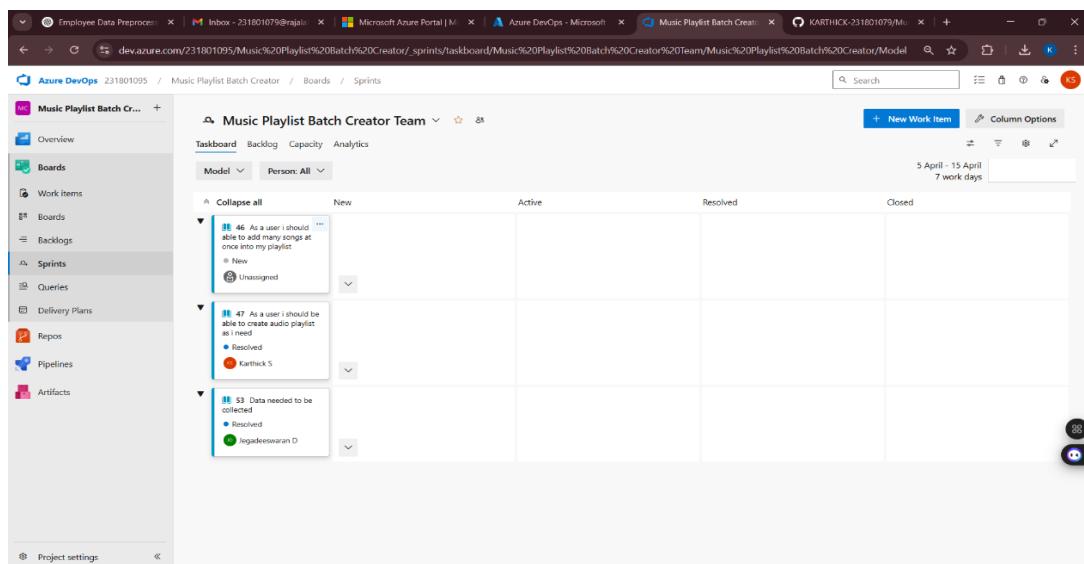
Sprint Planning

Sprint 1



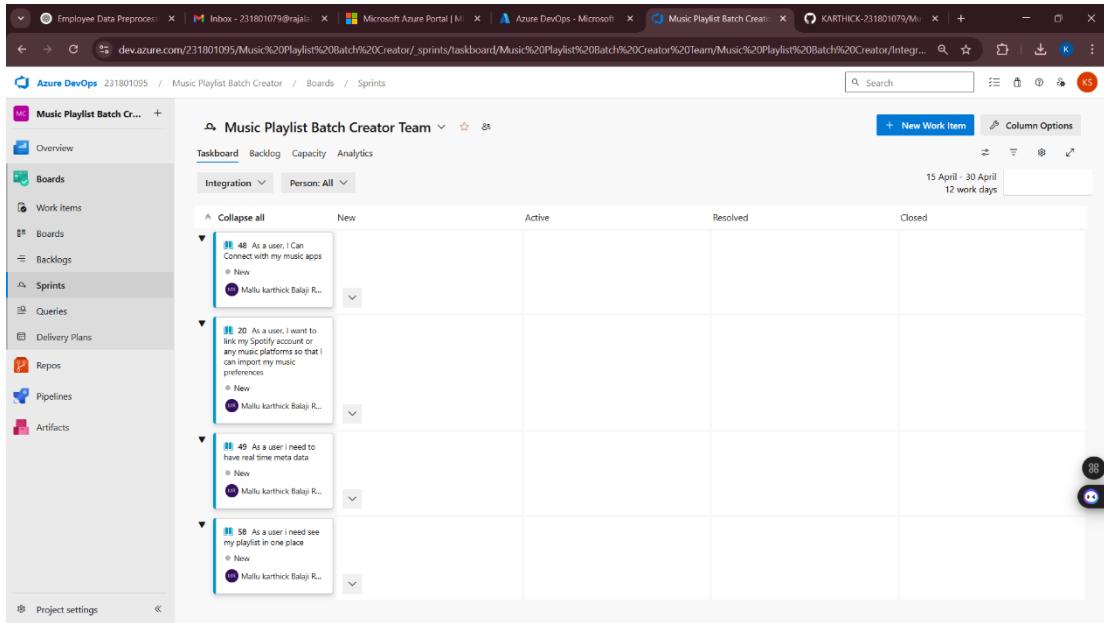
The screenshot shows the Azure DevOps interface for the 'Music Playlist Batch Creator' project. The left sidebar includes 'Overview', 'Boards', 'Work items', 'Backlogs', 'Sprints', 'Queries', 'Delivery Plans', 'Repos', 'Pipelines', and 'Artifacts'. The main area displays the 'Taskboard' for the 'Music Playlist Batch Creator Team'. The backlog section shows two user stories: '19 As a user, I want to sign up and log in securely so that I can access my playlists' and '21 Implement JWT-based authentication'. Both stories are categorized as 'New' and assigned to 'Karthikeyan Senthil'. The taskboard also shows columns for 'Active', 'Resolved', and 'Closed' tasks.

Sprint 2



The screenshot shows the Azure DevOps interface for the 'Music Playlist Batch Creator' project. The left sidebar includes 'Overview', 'Boards', 'Work items', 'Backlogs', 'Sprints', 'Queries', 'Delivery Plans', 'Repos', 'Pipelines', and 'Artifacts'. The main area displays the 'Taskboard' for the 'Music Playlist Batch Creator Team'. The backlog section shows three user stories: '46 As a user I should be able to add many songs at once into my playlist', '47 As a user I should be able to create audio playlist as I need', and '43 Data needed to be collected'. The first two stories are marked as 'New' and assigned to 'Karthick S'. The third story is marked as 'Resolved' and assigned to 'legadeeswaran D'. The taskboard also shows columns for 'Active', 'Resolved', and 'Closed' tasks.

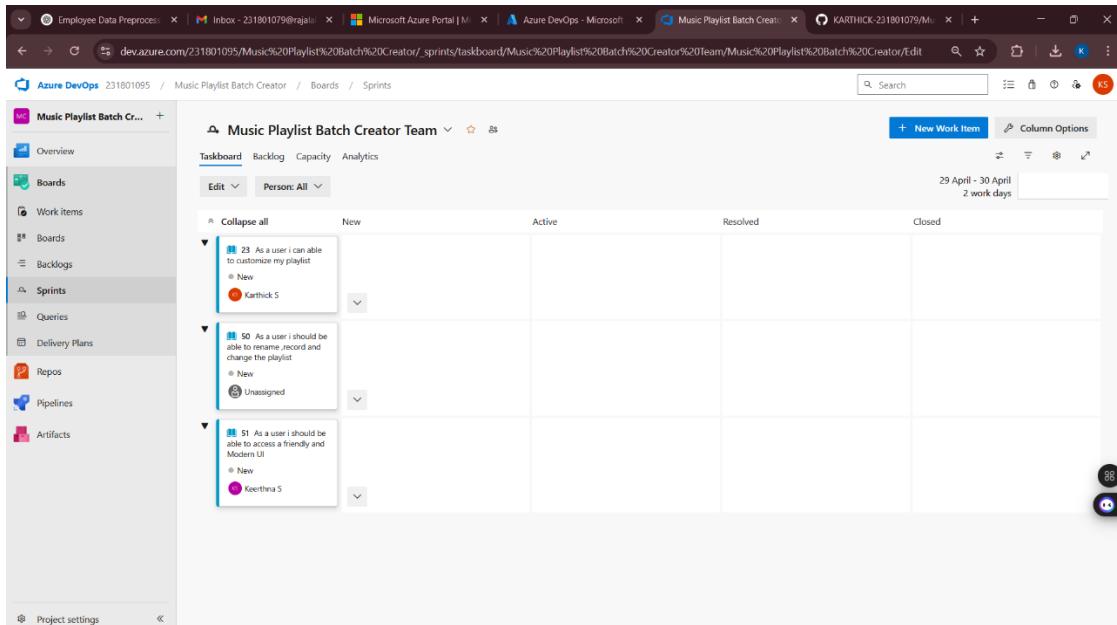
Sprint 3



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

- #48 As a user, I can Connect with my music apps (New, assigned to Mallu karthick Balaji R.)
- #20 As a user, I want to link my Spotify account or any music platforms so that I can import my music preferences (New, assigned to Mallu karthick Balaji R.)
- #49 As a user I need to have real time meta data (New, assigned to Mallu karthick Balaji R.)
- #58 As a user I need see my playlist in one place (New, assigned to Mallu karthick Balaji R.)

Sprint 4



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

- #23 As a user I can able to customize my playlist (New, assigned to Karthick S.)
- #50 As a user I should be able to rename, record and change the playlist (New, Unassigned)
- #51 As a user I should be able to access a friendly and Modern UI (New, assigned to Keethra S.)

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 the Azure DevOps interface for a work item titled "USER STORY 19". The story is described as: "19 As a user, I want to sign up and log in securely so that I can access my playlists". It was created by Karthikeyan Senthil. The work item is currently "New". The area is "Music Playlist Batch Creator" and the iteration is "Music Playlist Batch Creator\Login". The planning section shows Story Points as 2 and Priority as 2. The deployment section includes a note about tracking releases. The classification section indicates the value area is "Business". The development section provides instructions for linking to Azure Repos. The discussion section is currently empty.

Result:

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

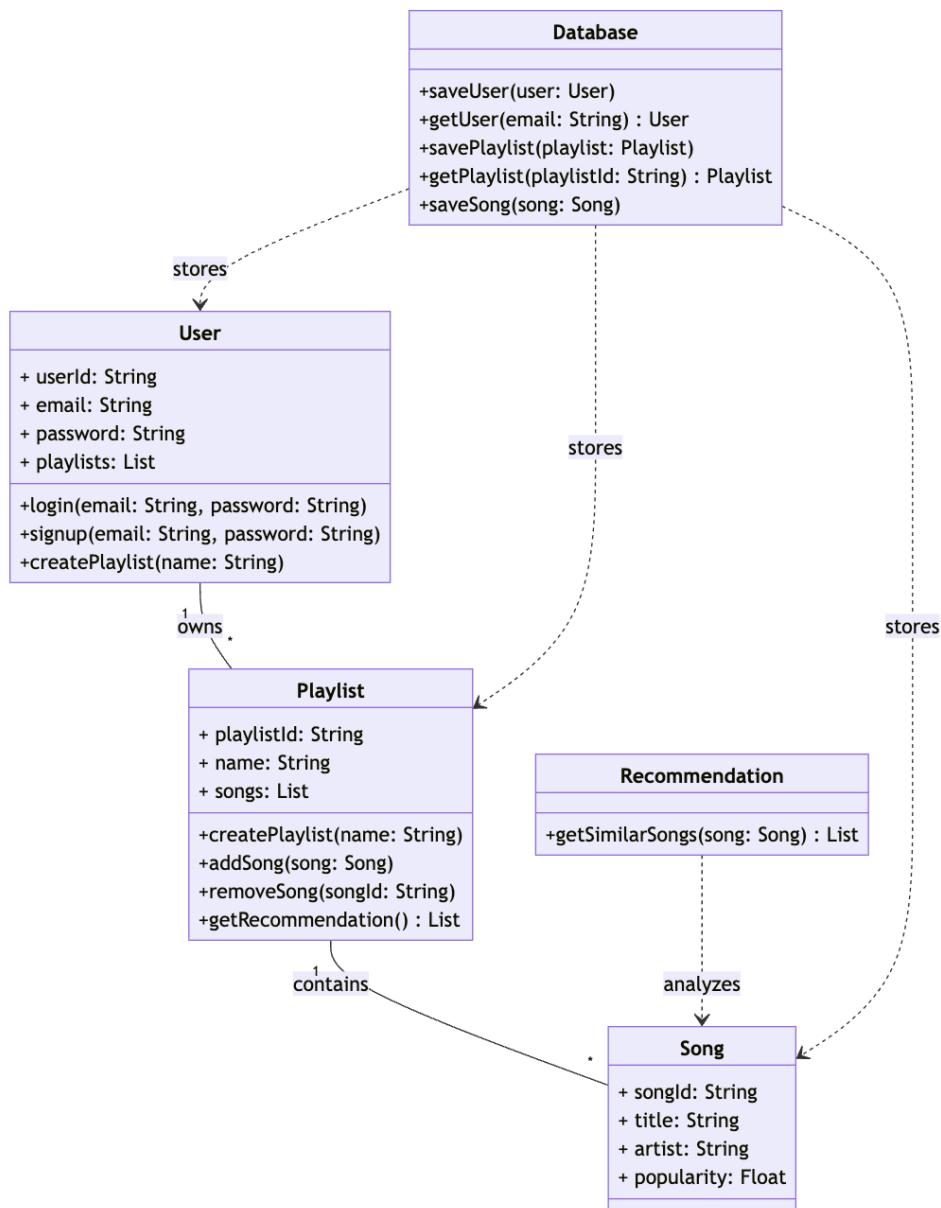
EXP NO: 6

DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE

Aim:

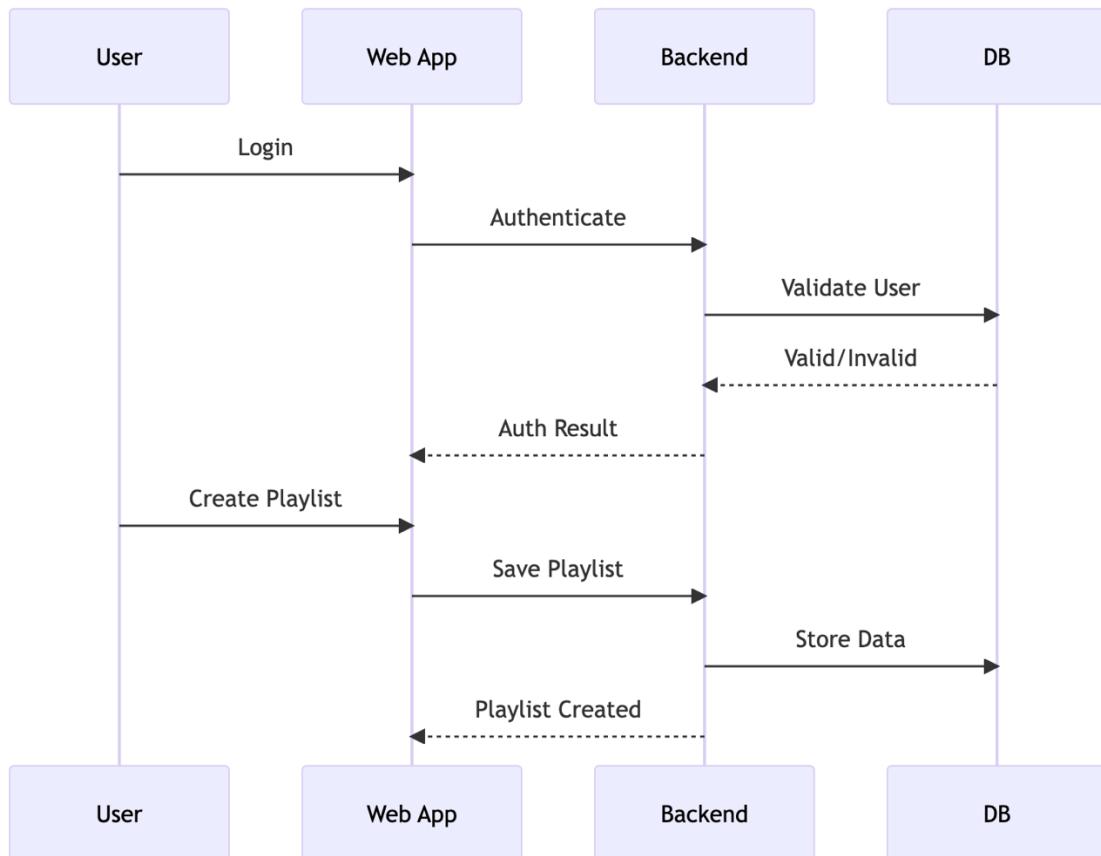
To Design a Class Diagram and Sequence Diagram for the given Project.

6A. Class Diagram



6B. Sequence Diagram

Karthikeyan Senthil 1 Apr



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

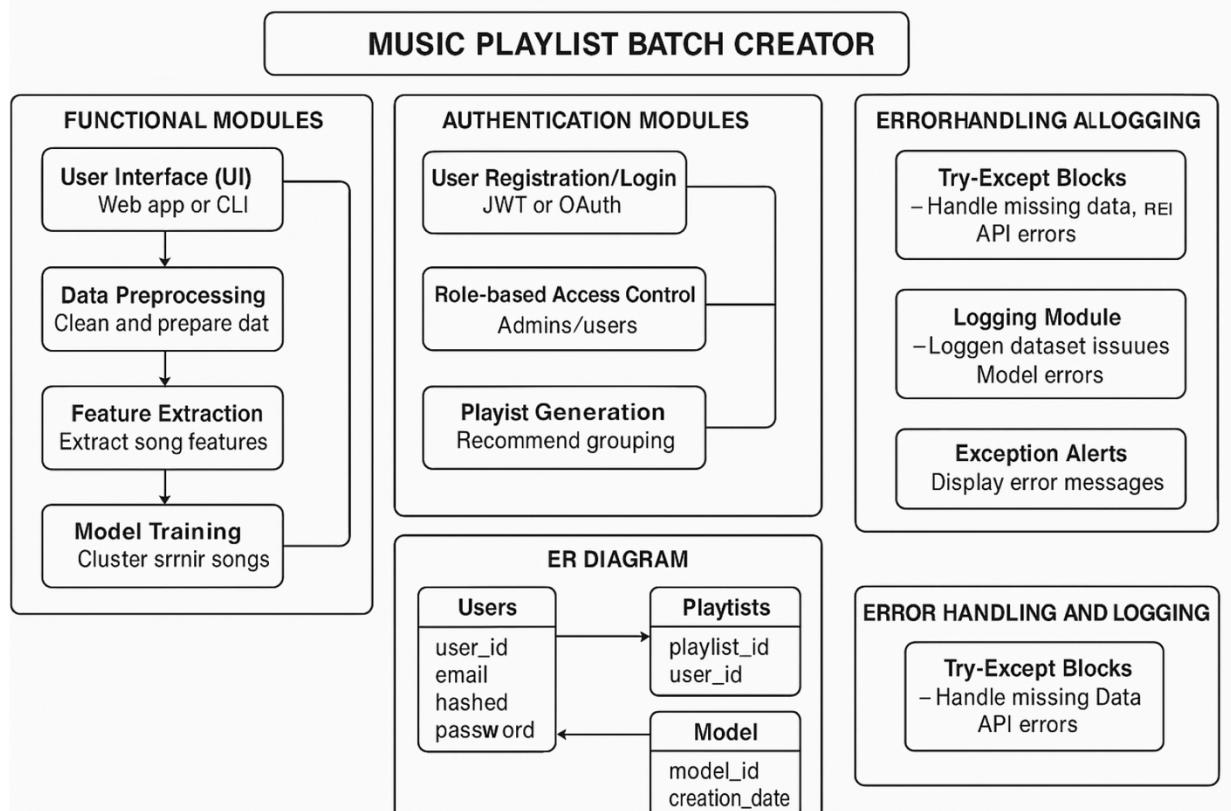
Aim:

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

7A&B. Architectural Diagram and ER Diagram

Karthikeyan Senthil 1 Apr

Architectural 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

Title	Order	Test Case Id	Assigned To	State
TC01 - Successful Sign Up	1	78	Karthikeyan Se... Design	
TC02 - Existing Email	2	80	Karthikeyan Se... Design	
Requirement based suite	3	81	Karthikeyan Se... Design	
Query based suite	4	82	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 the Microsoft Azure DevOps Test Plan interface. A test case titled "77 TC06 – Playlist Loading Failure" is displayed. The test case details include:

- Assigned To:** Karthick S
- Comments:** 0
- Add Tag:** None
- Stage:** Design
- Area:** Music Playlist Batch Creator
- Reason:** New
- Iteration:** Music Playlist Batch Creator/Integration

The **Steps** section contains two steps:

Step	Action	Expected result	Attachments
1.	Disconnect from internet	Network is offline	
2.	Navigate to "My Playlists"	Error message "Unable to load playlists" is shown	

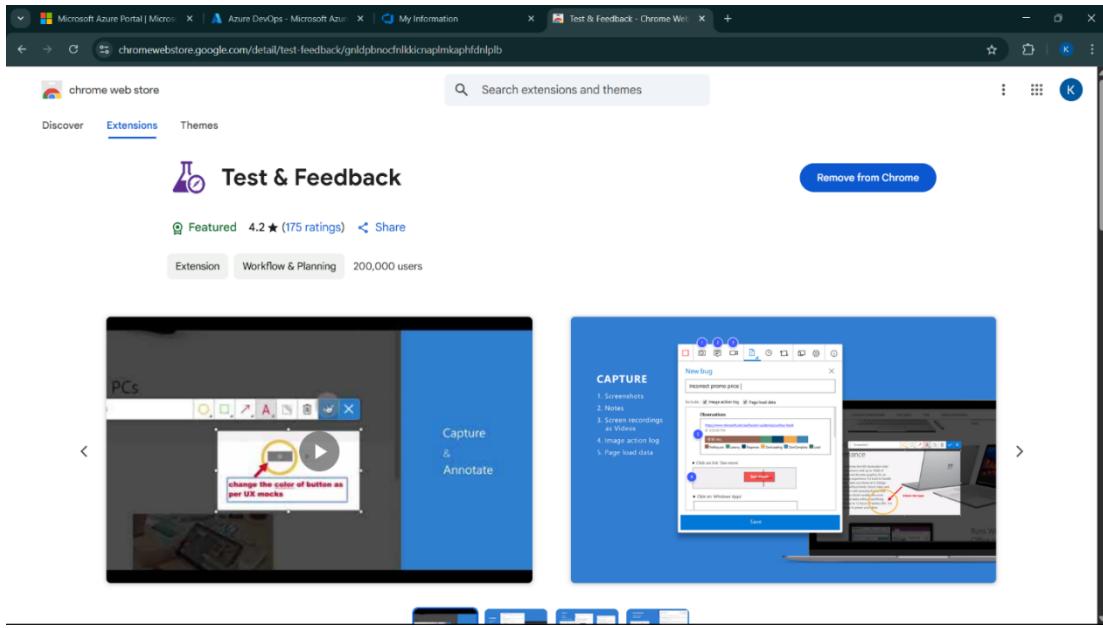
The **Status** section indicates:

- Type:** Error Path
- Priority:** Z
- Automation status:** Not Automated

The screenshot shows the Azure DevOps interface for a project titled "Music Playlist Batch ...". The left sidebar is expanded to show "Work items". A specific work item, "TEST CASE 80" (TC02 – Secure Login), is selected. The main area displays the test case details: State (Design), Area (Music Playlist Batch Creator), Reason (New), Iteration (Music Playlist Batch Creator>Login), and Steps tab selected. The "Steps" section contains three steps: 1. Go to the Login page (Expected result: Login form is displayed). 2. Enter valid email and password (Expected result: Fields accept data without error). 3. Click on "Login" (Expected result: User is logged in and redirected to d). The "Custom" section includes fields for Priority (2), Type (Happy Path), and Automation status (Not Automated). The "Status" section shows Priority (2) and Automation status (Not Automated). A "Parameter values" section is also present.

4. Installation of test

The screenshot shows the Chrome Web Store page for the "Test & Feedback" extension. The extension is listed with a rating of 4.2 stars from 175 ratings. It is categorized under "Workflow & Planning" and has 200,000 users. The "Add to Chrome" button is visible. Two screenshots are shown: one demonstrating annotation on a mobile device screen, and another showing the extension's interface for capturing and managing bugs and screenshots.



Test and feedback

Showing it as an extension

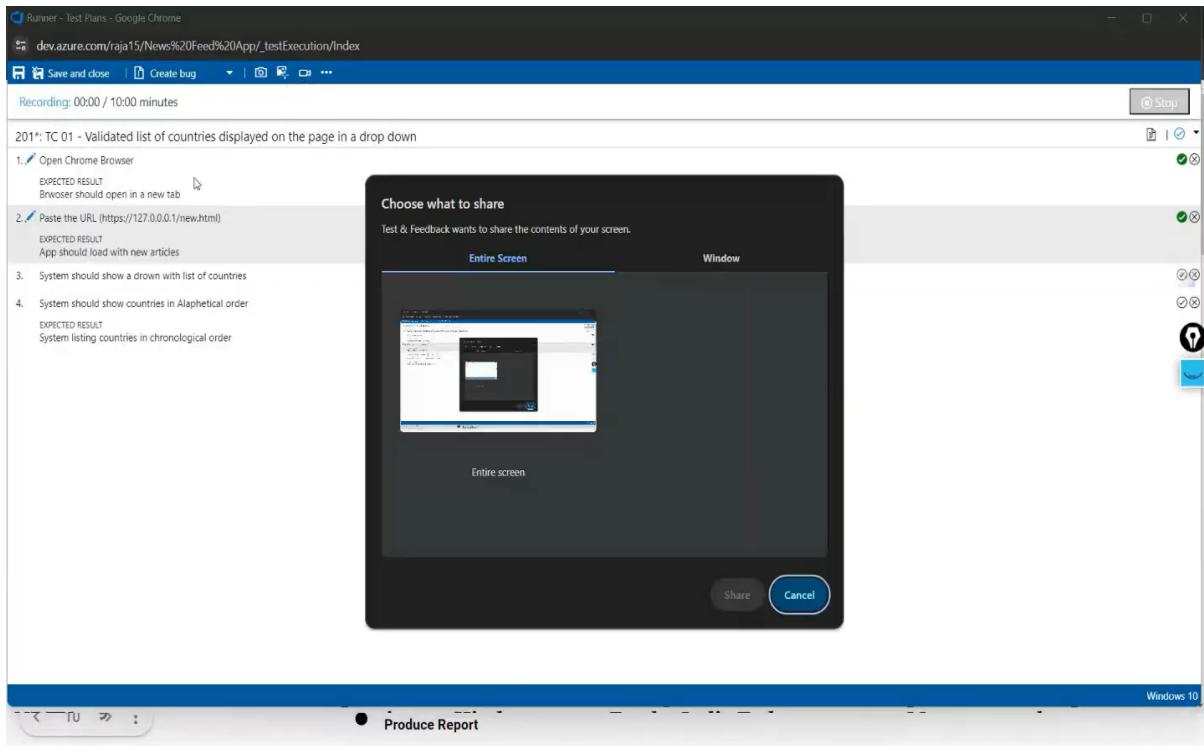
A screenshot of a web browser window showing the Azure DevOps Test Plan interface for a project named "Music Playlist Batch Creator". The left sidebar is open, showing "Test Plans" selected. In the main content area, a test plan titled "TS01 - User Login (ID: 86)" is displayed, listing four test cases: "Title", "TC01 - Successful Sign Up", "TC02 - Secure Login", "TC03 - Sign Up with Existing Email", and "TC04 - Login with Wrong Password". On the far right, a floating "Extensions" pane is open, showing a list of installed extensions under "Full access": "Copy Text from Picture", "Dark Reader", "Monica: ChatGPT AI Assist...", "Selectext: Copy text from V...", and "Test & Feedback". The "Test & Feedback" extension is highlighted with a blue border. A small circular icon with a question mark is visible in the bottom right corner of the browser window.

5. Running the test cases

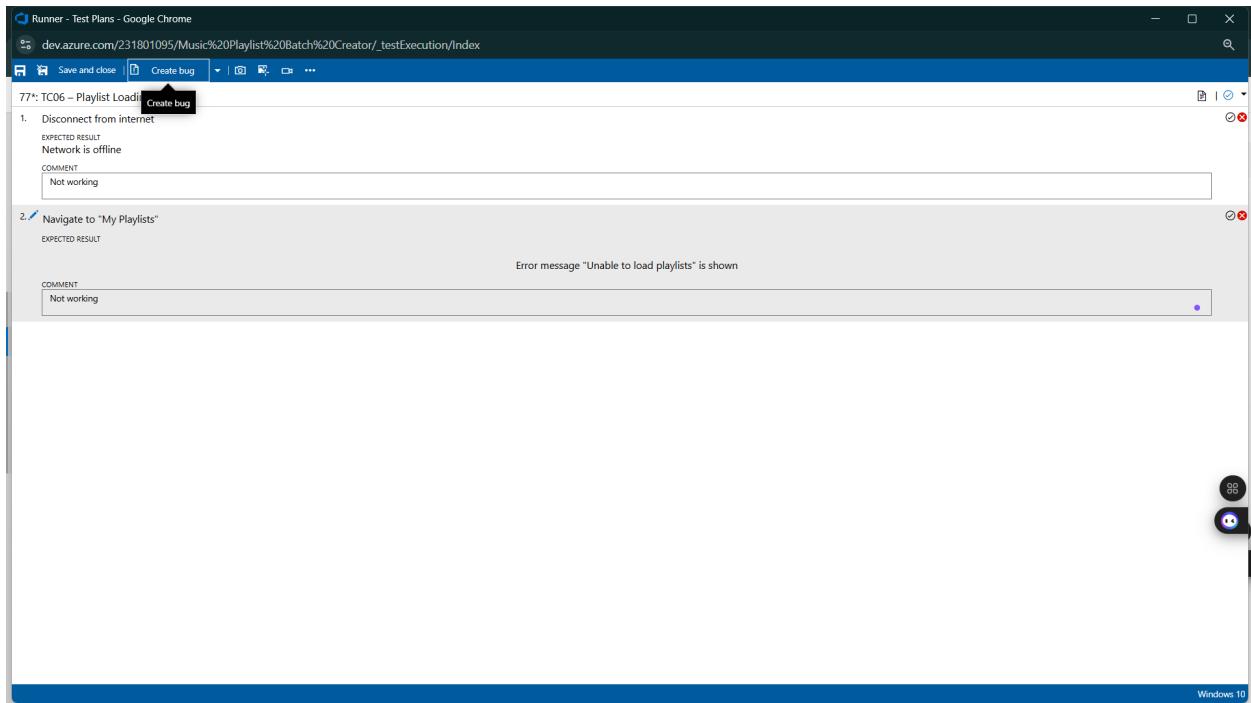
The screenshot shows the Azure DevOps Test Plan interface. On the left, there's a navigation sidebar with options like Overview, Boards, Repos, Pipelines, Test Plans, Test plans, Progress report, Parameters, Configurations, Runs, and Artifacts. The 'Test plans' section is currently selected. In the center, a test suite named 'TS02 - View Playlists (ID: 87)' is displayed. The 'Execute' tab is selected. Under 'Test Points (2 items)', there are two entries: 'TC05 – View Playlist Page' (Passed) and 'TC06 – Playlist Loading Failure' (Passed). A context menu is open over the first entry, showing options like 'View execution history', 'Mark Outcome', 'Run', 'Reset test to active', 'Edit test case', 'Assign tester', and 'View test result'. The status bar at the bottom indicates 'Apr 10 - Apr 17' and '100% run, 100% passed. View report'.

The screenshot shows a browser window titled 'Runner - Test Plans - Google Chrome' with the URL 'dev.azure.com/231801095/Music%20Playlist%20Batch%20Creator/_testExecution/Index'. The page displays a test step for 'TC05 – View Playlist Page'. Step 1: 'Log in successfully' (EXPECTED RESULT: User is redirected to dashboard). Step 2: 'Navigate to "My Playlists" section' (EXPECTED RESULT: All created playlists are displayed clearly). The status bar at the bottom indicates '75% TC05 – View Playlist Page'.

6.Recording the test case



7.Creating the bug



Runner - Test Plans - Google Chrome
dev.azure.com/231801095/Music%20Playlist%20Batch%20Creator/_testExecution/Index

Save and close Create bug ...

77: TC06 – Playlist Loading Failure

1. Disconnect from internet NEW BUG*

2. Navigate to "My Playlists" EXPECTED

TB01 - Playlist loading spinner keeps spinning indefinitely on poor network

Unassigned 0 comments Add tag TB01 - Playlist loading spinner keeps spinning indefinitely on poor network Save & Close ...

State: New Area: Music Playlist Batch Creator Reason: New Iteration: Music Playlist Batch Creator

Repro Steps

18-04-2025 03:23 Bug filed on "TC06 – Playlist Loading Failure"

Step no. Result Title

1. Failed Disconnect from internet

Expected Result Network is offline

Comments: Page Not loading

2. Failed Navigate to "My Playlists"

Expected Result

Error message "Unable to load playlists" is shown

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 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: 77 TC06 – Playlist Loading Failure Updated 10-04-2025, ● Design

System Info

Found in Build

Windows 10

Microsoft Azure Portal | Azure DevOps - Micro | My Information | Test Plan 84 Music P | Runs - Test Plans | Settings - Overview | Bug Report Playlist | +

dev.azure.com/231801095/Music%20Playlist%20Batch%20Creator/_testManagement/runs?a=resultSummary&runId=48&resultId=100000

Azure DevOps 231801095 / Music Playlist Batch Creator / Test Plans / Runs

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

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

Unassigned 0 comments Add tag

State: New Area: Music Playlist Batch Creator Reason: New Iteration: Music Playlist Batch Creator

System Info

Browser - Name	Google Chrome 135
Browser - Language	en-IN
Browser - Height	864
Browser - Width	1536
Browser - User agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/135.0.0.0 Safari/537.36
Operating system - Name	Windows NT 10.0; Win64; x64
Operating system - Architecture	x64_64
Operating system - Processor mode	11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz
Operating system - Number of processors	4
Memory - Available	814784512
Memory - Capacity	8216240128
Display - Pixels per inch (X axis)	120
Display - Pixels per inch (Y axis)	120
Display - Device pixel ratio	1.25

Discussion

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

Project settings

Name: SystemInformation-2025-04-18T03-23-58.168Z.json Size: 1K

8. Test case results

The screenshot shows the Azure DevOps interface for a test plan named 'Music Playlist Batch Creator'. On the left, the navigation pane is visible with options like Overview, Boards, Repos, Pipelines, Test Plans, and Artifacts. The 'Test plans' section is selected. In the center, a test suite named 'TS02 - View Playlists (ID: 87)' is displayed under 'Test Suites'. This suite contains two test points: 'TC05 - View Playlist Page' (which is checked) and 'TC06 - Playlist Loading Failure'. A detailed view of 'TC05 - View Playlist Page' is shown on the right, titled 'TC05 - View Playlist Page'. It includes a 'Test Case Results' table with columns: Outcome, TimeStamp, Configuration, Run by, Tester, and Test PL. The table lists several entries, mostly 'Passed' outcomes with various timestamps and configurations (Windows 10, Windows 11, etc.). A link 'Open execution history for current test point' is also present.

9. Test report summary

The screenshot shows the Azure DevOps interface for a work item named 'BUG 203'. The left sidebar shows the project structure with 'News Feed App' selected. The main area displays the details of the bug report. The work item has the following details:
- Title: BUG 203
- Description: 203 - BG 01 - Countries Drop down Not Available on the page
- Reporter: rajesh prabhu
- 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.: 2.
- Result: Passed
- Title: Paste the URL (<https://127.0.0.1/new.html>)
- Expected Result: App should load with new articles
- Step no.: 3.
- Result: Failed
- Title: System should show a dropdown with list of countries
- Expected Result: Produce Report
The right side of the screen shows sections for Planning, Deployment, Development, and Effort (Hours), along with deployment status and related work links.

- Assigning bug to the developer and changing state

Bug Detail:

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

State: New
Reason: New
Iteration: Music Playlist Batch Creator

Repro Steps:

18-04-2025 03:23 Bug filed on "TC06 – Playlist Loading Failure"

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

Test Configuration: Windows 10

Planning:

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

Effort (Hours):

- Original Estimate: 0
- Remaining: 0
- Completed: 0

Deployment:

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

Development:

+ Add link
Link an Azure Repos commit, pull request or branch to see the status of your development. You can also [create a branch](#) to get started.

Related Work:

+ Add link
Add an existing work item as a parent
77 TC06 – Playlist Loading Failure

System Info:

10. Progress report

Progress report:

Summary:

- 1 test plans
- 14 Test points
- 14 (14 / 14) Test points run (100% Run)
- 100% (14 / 14) Pass rate
- 14 Passed

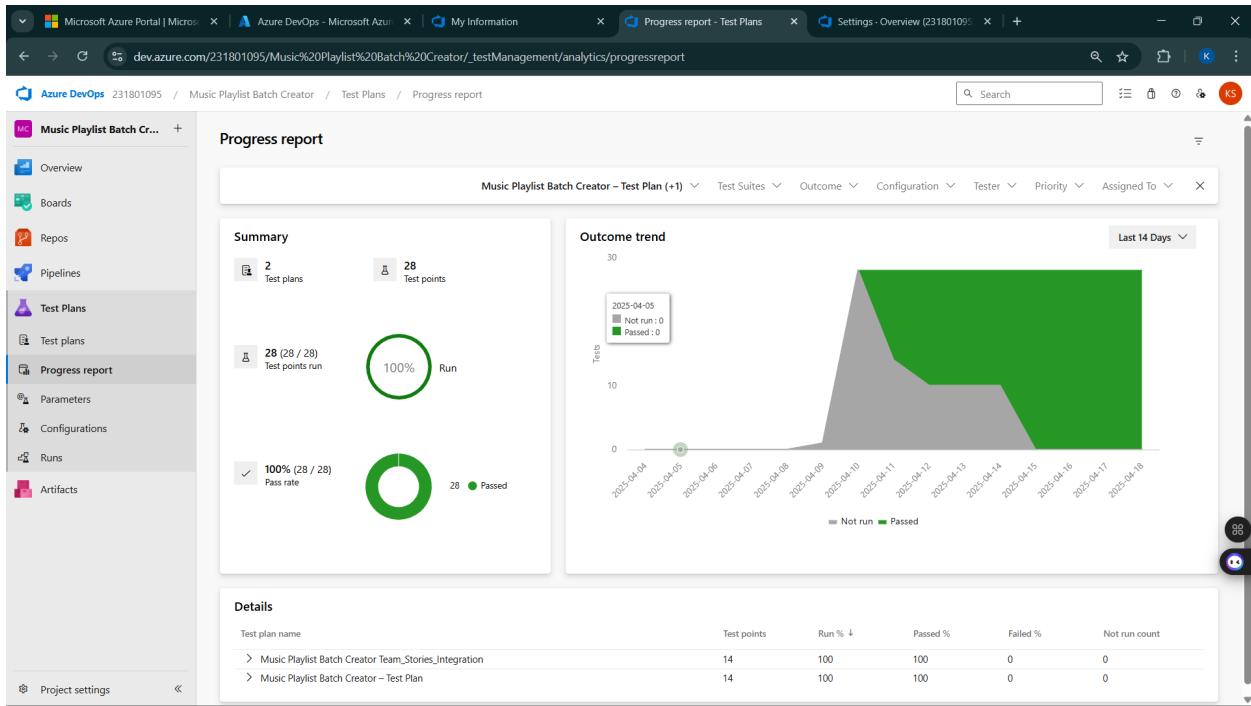
Outcome trend:

Last 14 Days

Music Playlist Batch Creator – Test Plan

Test Suite: Outcome Configuration Tester Priority Assigned To

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

Organization Settings

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 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' section of the Azure DevOps Settings - Process page. The 'Processes' tab is selected. The table lists four templates:

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' section of the Azure DevOps Settings - Process page. The 'Processes' tab is selected. The table lists five templates, including a newly created one:

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
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 'Add a field to Test Case' dialog box over a background of the Azure DevOps settings interface. The dialog has tabs for 'Definition', 'Options', and 'Layout'. Under 'Definition', the 'Create a field' option is selected, with 'Name' set to 'Type' and 'Type' set to 'Text (single line)'. A description is provided: 'Optionally provide a description for the field'. At the bottom are 'Add field' and 'Cancel' buttons.

The screenshot shows the 'Work-item types' section of the Azure DevOps settings. It lists a single work-item type: 'Music Playlist Batch Creator'. The 'Description' column contains the text: 'The Azure Music Playlist Batch Creator is a cloud-based solution designed for bulk playlist creation and management. Leverage...'. The 'Projects' tab is selected in the navigation bar.

The screenshot shows the Azure DevOps Settings - Process page. The URL in the browser is dev.azure.com/231801095/_settings/process?type-id=231801095Agile.TestCase&process-name=231801095%20Agile&_a=layout. The page title is "All processes > 231801095 Agile > Test Case". On the left, there is a sidebar with "Organization Settings" for "231801095" and a search bar. The main content area shows a "Steps" section with a "Text (multiple lines)" input field. To the right, there are sections for "Custom", "Recent test results", "Deployment", "Development", "Related Work", and "Status". A vertical scroll bar is visible on the right side of the main content area.

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:

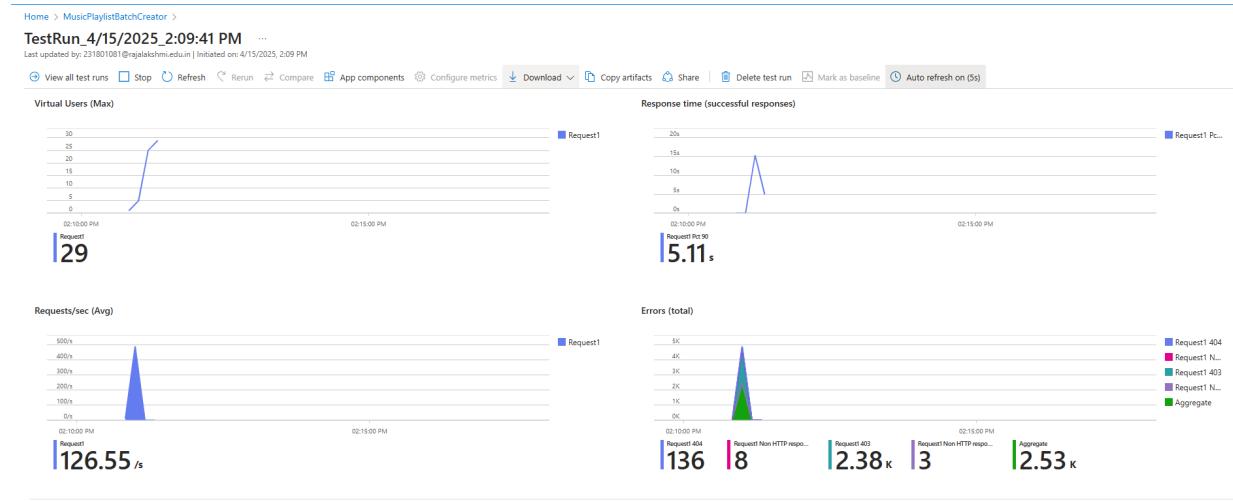
1. Sign in to Azure Portal
Go to <https://portal.azure.com> and log in.
2. Create the Resource
 - o Go to *Create a resource* → Search for “Azure Load Testing”.
 - o Select Azure Load Testing and click Create.
3. Fill in the Configuration Details
 - o *Subscription*: Choose your Azure subscription.
 - o *Resource Group*: Create new or select an existing one.
 - o *Name*: Provide a unique name (no special characters).
 - o *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.

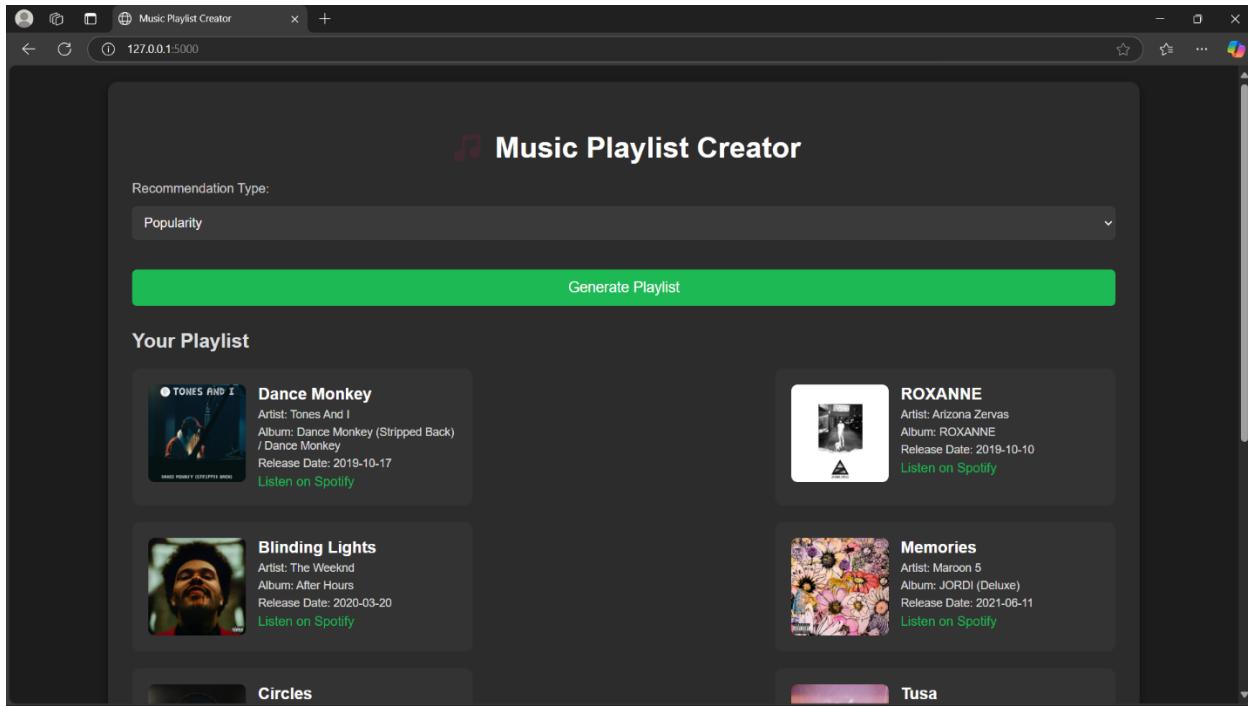
Steps to Create and Run a Load Test:

Once your resource is ready:

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

Load Testing





Pipelines

Description:

This experiment demonstrates how to connect a GitHub-hosted Flask-based music recommendation project with Azure DevOps. The pipeline will automatically install dependencies, run basic tests, and publish artifacts. This ensures that every commit triggers checks for reliability and smooth deployment.

Steps:

1. Connect GitHub to Azure DevOps:
 - o In Azure DevOps, create a new project.
 - o Create a pipeline and select GitHub as the source.
 - o Authorize access to your GitHub repository, ensuring that Azure DevOps can pull the repository for your pipeline.
2. Create azure-pipelines.yml in Your Repo Root:
 - o In your GitHub repository, create a new file called azure-pipelines.yml in the root directory.
 - o Add the following basic pipeline configuration for Python and Flask

yaml Code

trigger:

```
- main # Trigger pipeline when changes are pushed to the main branch
```

pool:

```
vmImage: ubuntu-latest # Use a hosted Ubuntu agent
```

steps:

```
# Step 1: Checkout the code from GitHub
- checkout: self

# Step 2: Set up Python environment
- task: UsePythonVersion@0
inputs:
versionSpec: '3.x' # Use the latest Python 3.x version displayName: "Set up Python"

# Step 3: Install dependencies from the correct path
- script: |
    python -m pip install --upgrade pip
    pip install -r project/requirements.txt # Adjusted path to requirements.txt displayName: "Install dependencies"

# Step 4:
Run a simple Python script to check the environment
- script: |
    python -c "print('Hello from Music Playlist Batch Creator!')"
    displayName: "Run a Python script"
```

3. Pipeline Tasks Include:

- o Setting up the Python environment using the UsePythonVersion task.
- o Installing project dependencies from project/requirements.txt. Make sure the path to requirements.txt is correct (it is located under the project folder).
- o Running a simple Python script to verify that Python is set up correctly and the pipeline works.

4. Run and Monitor Pipeline:

- o Commit changes to the main branch of your repository to trigger the pipeline in Azure DevOps.
- o Monitor the logs in the Azure DevOps portal to view logs, errors, or success messages and ensure everything runs smoothly.

The screenshot shows the Azure DevOps Pipelines interface for a project named "Music Playlist Batch Creator". The pipeline is titled "#20250424.3 • Pipeline 2". The summary indicates that this run is being retained as one of 3 recent runs by main (Branch). The pipeline was manually run by Karthick S. The repository and version information shows "Music Playlist Batch Creator (9)" at commit "main a87bd670". The run started "Just now" and took "24s". There are 0 work items and 0 artifacts related to this run. The "Jobs" section lists a single job that has completed successfully in 6s. The pipeline configuration is visible on the right side of the screen.

Result:

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

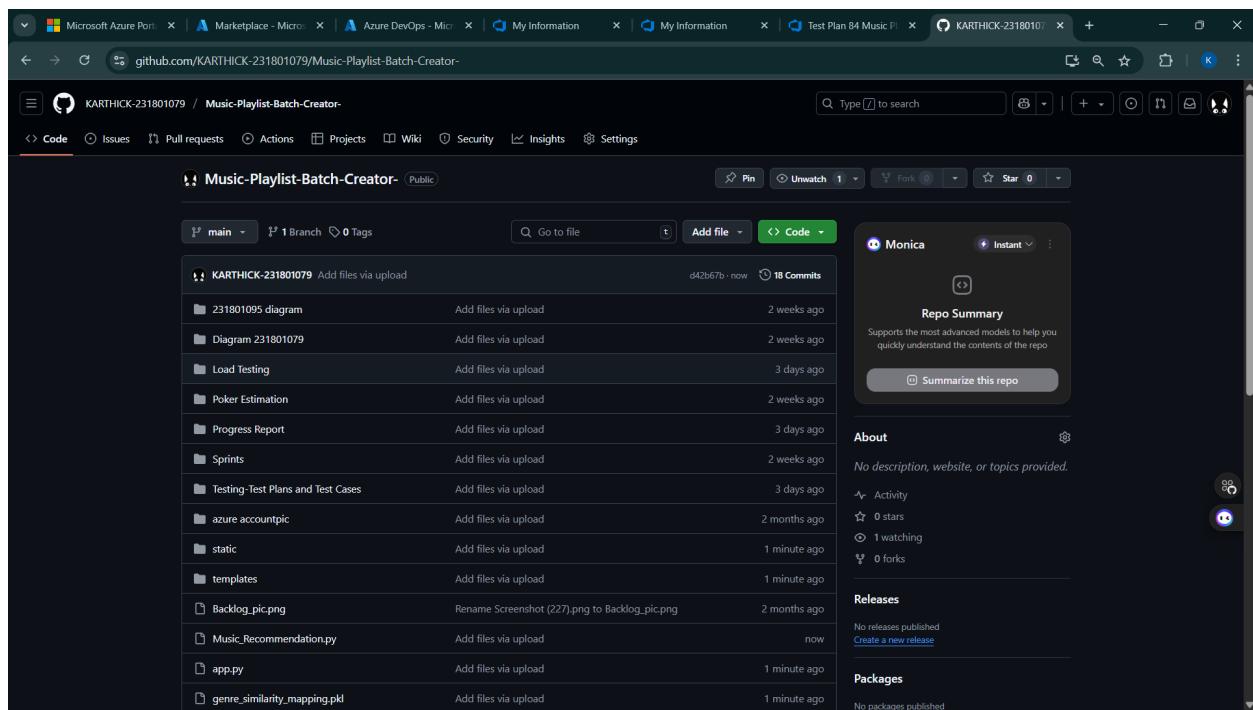
EXP NO: 10

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



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.