



**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: Karthick S

Register No: 231801079

Year / Branch / Section: 2<sup>nd</sup> / AI&DS / FA

Semester: IV

Academic Year: 2024 – 2025

**RAJALAKSHMI ENGINEERING COLLEGE**  
**RAJALAKSHMI NAGAR, THANDALAM - 602 105.**



**RAJALAKSHMI  
ENGINEERING COLLEGE**

CS23432  
Software Construction

**Laboratory Record Note Book**

NAME ..Karthick ..S.....

BRANCH ..A.I.K.P.S - FA.....

UNIVERSITY REGISTER No ..2116.23.18010.79.....

COLLEGE ROLL No ..23.18.010.79.....

SEMESTER ..I.....

ACADEMIC YEAR ..20.24 - 25.....



**RAJALAKSHMI  
ENGINEERING COLLEGE**

An AUTONOMOUS Institution  
Affiliated to ANNA UNIVERSITY, Chennai

**BONAFIDE CERTIFICATE**

NAME ...Karthick... S.....

ACADEMIC YEAR .....2024 - 25..... SEMESTER .....IV..... BRANCH.....Artificial Intelligence & Data Science

UNIVERSITY REGISTER NO.

2116231801079

Certified that this is the bonafide record of work done by the above student in the

CS 23432

.....Software Construction..... Laboratory during the year 2024 - 2025

**Signature of Faculty - in - Charge**

**Submitted for the Practical Examination held on.....**

**External Examiner**

**Internal Examiner**

## INDEX

Name :	Karthick S	Branch :	AIA Dp	Sec :	FA	Roll No :	2116231801079
S.No.	Date	Title		Page No.	Teacher's Sign/Remarks		
1.	21/1/25	Azure Devops Environment Setup		6			
2.	21/1/25	Azure Devops Project Setup and User Story Management		7			
3.	11/2/25	Setting Up Epic, Feature And User Stories for Project Planning		14			
4.	18/2/25	Sprint Planning		17			
5.	25/2/25	Poker Estimation		20			
6.	04/3/25	Designing class and Sequence Diagrams for Project Architecture		21			
7.	25/3/25	Designing Architectural and ER Diagrams for Project Structure		23			
8.	15/4/25	Testing - Test Plan and Test Cases		25			
9.	22/4/25	Load Testing and Pipelining		32			
10.	22/4/25	GitHub: Project Structure & Naming Conventions		47			
		<i>Completed</i>					

# INDEX

S.No.	Date	Title	Page No
1.	21/1/25	Azure Devops Environment Setup.	6
2.	21/1/25	Azure Devops Project Setup and User Story Management.	9
3.	11/2/25	Setting Up Epics, Features, And User Stories for Project Planning.	14
4.	18/2/25	Sprint Planning.	17
5.	25/2/25	Poker Estimation.	20
6.	04/3/25	Designing Class and Sequence Diagrams for Project Architecture.	21
7.	25/3/25	Designing Architectural and ER Diagrams for Project Structure.	23
8.	15/4/25	Testing – Test Plans and Test Cases.	25
9.	22/4/25	Load Testing and Pipelines.	42
10.	22/4/25	GitHub: Project Structure & Naming Conventions.	47

**EXP NO: 1**

## AZURE DEVOPS ENVIRONMENT SETUP

### **Aim:**

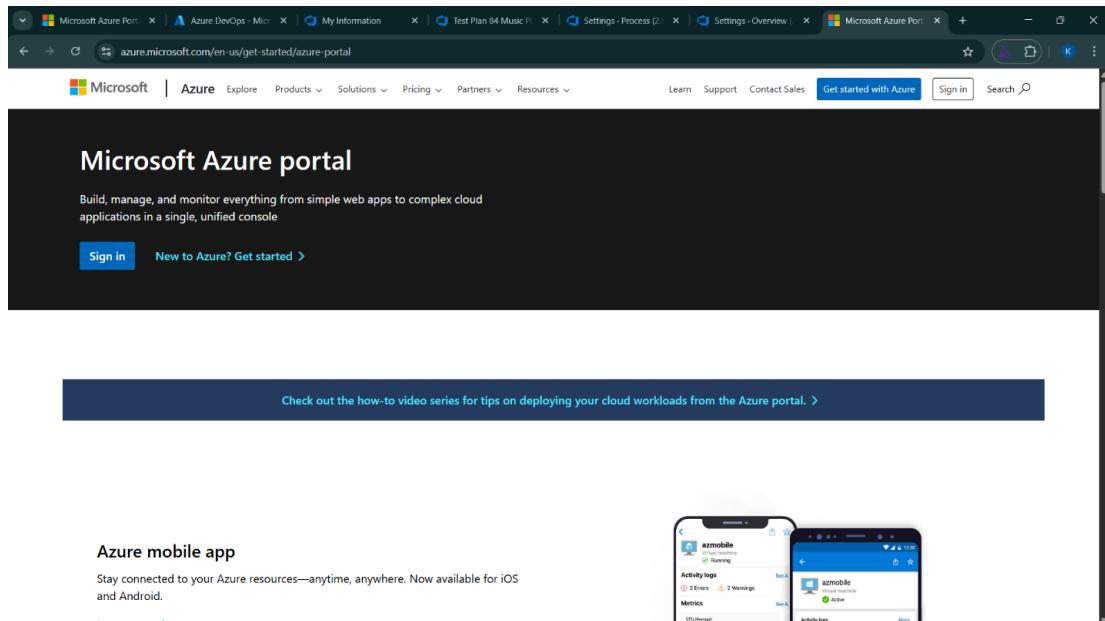
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

### **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>



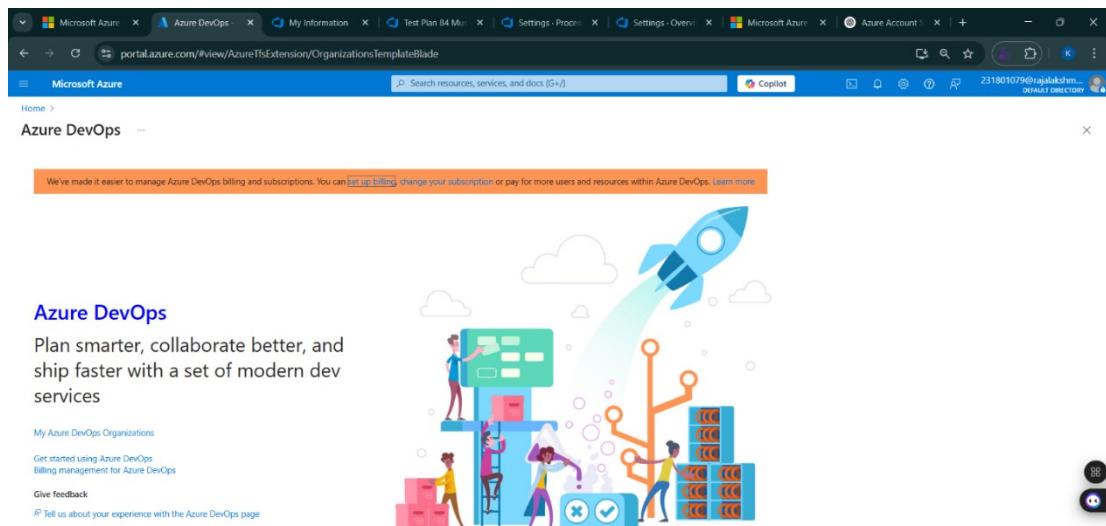
## 2. Azure home page

The screenshot shows the Microsoft Azure home page. At the top, there's a navigation bar with various tabs like Microsoft Azure, Azure DevOps, My Information, Test Plan 84 Mu..., Settings - Process, Settings - Overview, Microsoft Azure, and Azure Account. Below the navigation bar is the search bar with the placeholder "Search resources, services, and docs (S+)" and a Copilot button. The main content area is titled "Azure services" and includes a "Create a resource" button, icons for Azure DevOps organizations, Subscriptions, Dashboard hub, Resource groups, Azure Load Testing, Quickstart Center, Azure AI services, Kubernetes services, and More services. Below this is a "Resources" section with a "Recent" tab showing two items: "Music" (Azure Load Testing) and "Music\_playlist\_Batch\_Creator" (Resource group), both last viewed 3 days ago. There's also a "See all" link. A "Navigate" section has links for Subscriptions, Resource groups, All resources, and Dashboard. A "Tools" section includes Microsoft Learn, Azure Monitor, Microsoft Defender for Cloud, and Cost Management. At the bottom, there's a "Useful links" section and an "Azure mobile app" link.

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 again, but with a search query "Azure DevOps" entered into the search bar. The search results are displayed below the search bar. The "Services" section shows results like Azure Native New Relic Service, Managed DevOps Pools, Azure DevOps organizations, and Azure Native Dynatrace Service. The "Marketplace" section shows results like Static Web App, Rocky Linux 9, Build Agents for Azure DevOps, and InfluxDB Cloud (Official Version). The "Documentation" section includes links to DevOps architecture design - Azure Architecture Center, Secure your Azure DevOps - Azure DevOps, Course AZ-400T00-A: Designing and Implementing Microsoft DevOps solutions - T..., and Managed DevOps Pools Overview - Managed DevOps Pools. The "Microsoft Entra ID" section shows Azure DevOps as a service principal. The rest of the page layout is similar to the first screenshot, including the "Azure mobile app" link at the bottom.

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.

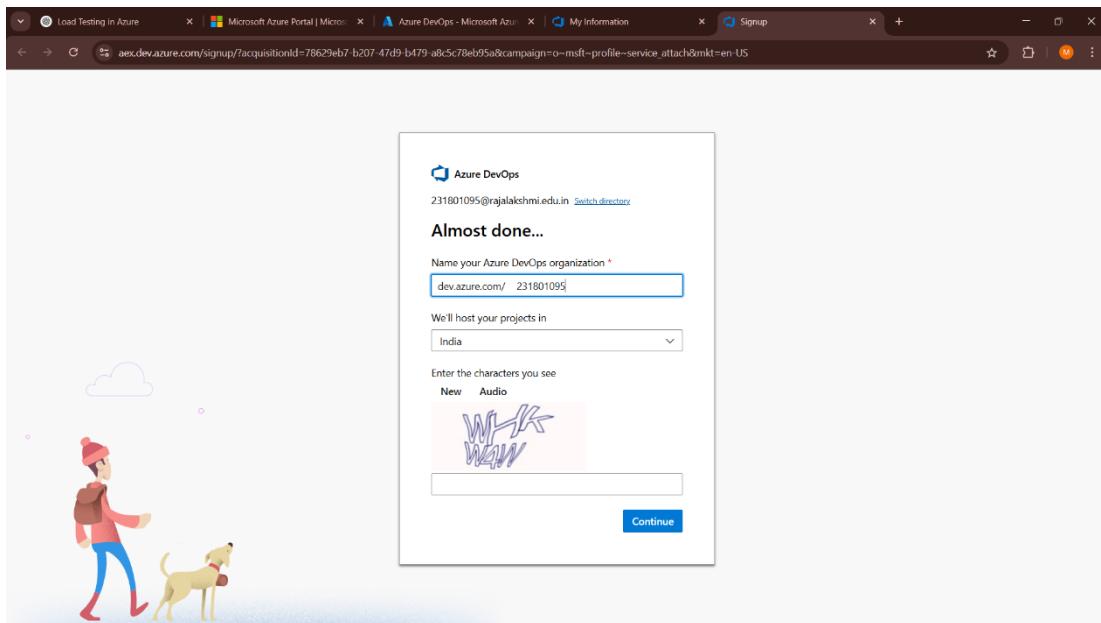
**EXP NO: 2**

## **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. The 'Project name' field contains 'Music Playlist Batch Creator'. The 'Visibility' section is expanded, showing two options: 'Public' (selected) and 'Private'. The 'Public' option is described as allowing anyone on the internet to view the project. The 'Advanced' section at the bottom includes dropdowns 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.

The screenshot shows the Azure DevOps Organizations dashboard. At the top, there's a navigation bar with various links like Microsoft Account, Azure DevOps, My Information, Signup, Test Plan 84, Settings - Project, Settings - Overview, Microsoft Azure, and Google Search. The user is signed in as Karthick S. On the left, there's a profile card for Karthick S with a large orange circular icon containing 'KS'. Below the profile, it shows contact information: 231801079@rajalakshmi.edu.in, rajalakshmi.edu.in, India, and the email 231801079@rajalakshmi.edu.in. A section for 'Visual Studio Dev Essentials' is present, mentioning benefits for building and deploying apps. On the right, the main dashboard shows 'Azure DevOps Organizations' with a link to dev.azure.com/231801095 (Member). It lists a single project: 'Music Playlist Batch Creator'. To the right of the project, there's a 'Actions' menu with options: Open in Visual Studio, Manage security, Browse extensions, and Leave. A 'Create new organization' button is also visible.

#### 4. Project dashboard

The screenshot shows the Azure DevOps Project dashboard for 'Music Playlist Batch Creator'. The left sidebar includes links for Overview, Summary, Dashboards, Wiki, Boards, Repos, Pipelines, Test Plans, and Artifacts. The main content area has a header 'Music Playlist Batch Creator' with a 'Public' button and an 'Invite' button. The dashboard is divided into several sections: '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 5 members with their icons). The 'About this project' section also includes sections for Key Features, Use Cases, Technical Stack, and Future Enhancements.

## 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 page. The left sidebar is titled 'Music Playlist Batch Cr...' and includes sections for Overview, Boards, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The main area is titled 'Backlog' and shows a table with the following data:

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 Azure DevOps work items page. The top navigation bar includes a search bar, a star icon, a refresh icon, a user icon (KS), and a more options icon. The main area shows a backlog with a 'New Work Item' button. A user profile overlay is displayed for 'Karthick S' (231801079@rajalakshmi.edu.in). The profile includes a red circular icon with 'KS', a 'View account' link, and a 'Switch directory' link. Below the profile, there are three 'Business' status indicators. At the bottom of the page, there is a 'Sign in with a different account' button.

**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 the 'Music Playlist Batch Creator' project. The left sidebar navigation includes 'Overview', 'Boards', 'Work items', 'Backlogs' (selected), 'Sprints', 'Queries', 'Delivery Plans', 'Analytics views', 'Repos', 'Pipelines', 'Test Plans', and 'Artifacts'. The main content area displays the 'Backlog' for the 'Music Playlist Batch Creator Team'. A table lists work items with columns for Order, Work Item Type, Title, State, Effort, Story Points, Business Value, Value Area, and Tags. The backlog contains several Epics, Features, and a User Story. For example, Epic 1 is 'User Authentication & Profile Management' (State: New). Feature 2 is 'Develop a system that allows users to create and manage m...' (State: New). The User Story is 'As a user, I should be able to create playlists based on...' (State: Resolved, Story Points: 3). Other items include 'Bulk Song Addition', 'Data Collection', 'Fetch and Integrate music data from external sources for pl...', 'Enable users to edit, customize, and share their playlists wit...', 'Improve system performance and enhance user experience', and 'Test Epic'.

### **1. Fill in Epics**

The screenshot shows the 'New Epic' creation dialog in the Azure DevOps interface. The dialog title is 'NEW EPIC'. The description field contains the text: 'Develop a system that allows users to create and manage multiple playlists in batches'. Below the description, there are fields for 'Karthick S' (Assignee), '0 Comments', 'Add Tag', 'Status: New', 'Area: Music Playlist Batch Creator', 'Reason: New', 'Iteration: Music Playlist Batch Creator/Model', and 'Iteration: Music Playlist Batch Creator/Model'. The dialog is divided into sections: 'Planning' (Priority: 2, Risk: Low, Effort, Business Value, Time Criticality), 'Deployment' (Instructions to track releases via Releases and deployment status reporting), 'Development' (Instructions to link Azure Repos commit, pull request or branch), and 'Related Work' (Add link to existing work item). At the bottom, there are 'Classification' fields for 'Value area: Business' and a note to 'Add an existing work item as a parent'.

## 2.Fill in Features

The screenshot shows the 'Details' view of a work item titled 'FEATURE 26'. The work item ID is 26, assigned to Karthick S, and is in the 'New' state under the 'Music Playlist Batch Creator' area and iteration. The 'Description' section contains placeholder text: 'Click to add Description.' The 'Planning' section shows priority 2 and risk level. The 'Deployment' section includes a note about tracking releases and a link to 'Releases'. The 'Development' section provides instructions for linking to Azure Repos. The 'Related Work' section lists another work item, '25 Develop a system that allows users to creat...', which is also in the 'New' state. The 'Classification' section indicates the value area is 'Business'.

## 3.Fill in User Story Details

The screenshot shows the 'Details' view of a work item titled 'USER STORY 47'. The work item ID is 47, assigned to Karthick S, and is in the 'Resolved' state under the 'Music Playlist Batch Creator' area and iteration. The 'Description' section contains placeholder text: 'Click to add Description.' The 'Acceptance Criteria' section contains placeholder text: 'Click to add Acceptance Criteria.' The 'Discussion' section contains placeholder text: 'Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request.' The 'Planning' section shows story points 3 and priority 2. The 'Deployment' section includes a note about tracking releases and a link to 'Releases'. The 'Development' section provides instructions for linking to Azure Repos. The 'Related Work' section lists another work item, '26 Auto-Playlist Creation Based on user prefer...', which is also in the 'New' state. The 'Classification' section indicates the value area is 'Business'.

**Result:**

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

**EXP NO: 4**

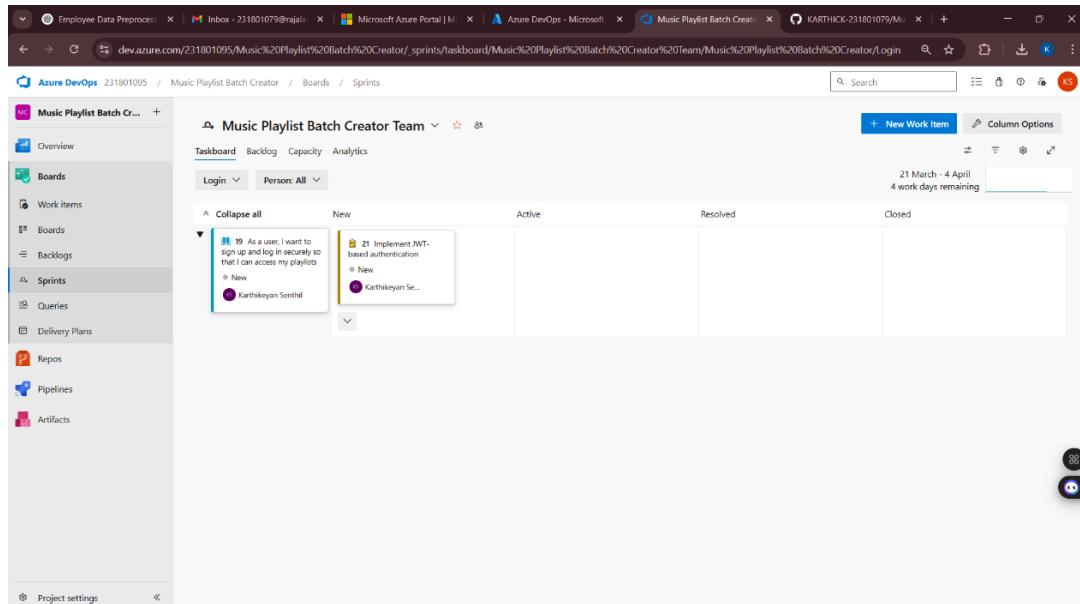
# SPRINT PLANNING

## Aim:

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

## Sprint Planning

### Sprint 1

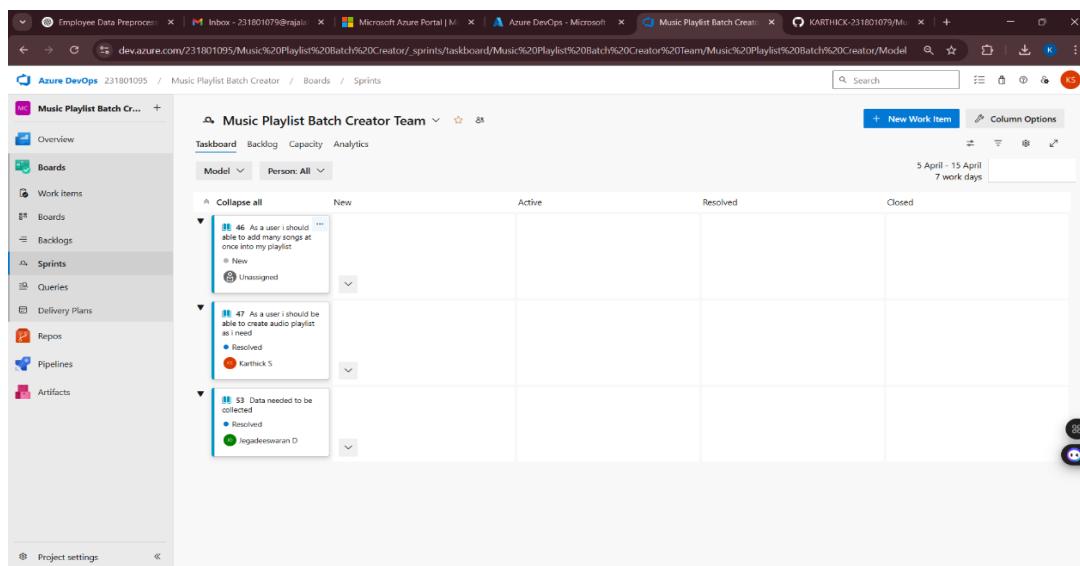


The screenshot shows the Azure DevOps interface for the 'Music Playlist Batch Creator' project. The left sidebar has 'Boards' selected. The main area is the 'Taskboard' for the 'Music Playlist Batch Creator Team'. The backlog section shows two user stories:

- As a user, I want to sign up and log in securely so that I can access my playlists
- As a user, I want to implement JWT-based authentication

Both stories are listed under the 'New' column and are assigned to 'Karthikeyan Senthil'.

### Sprint 2

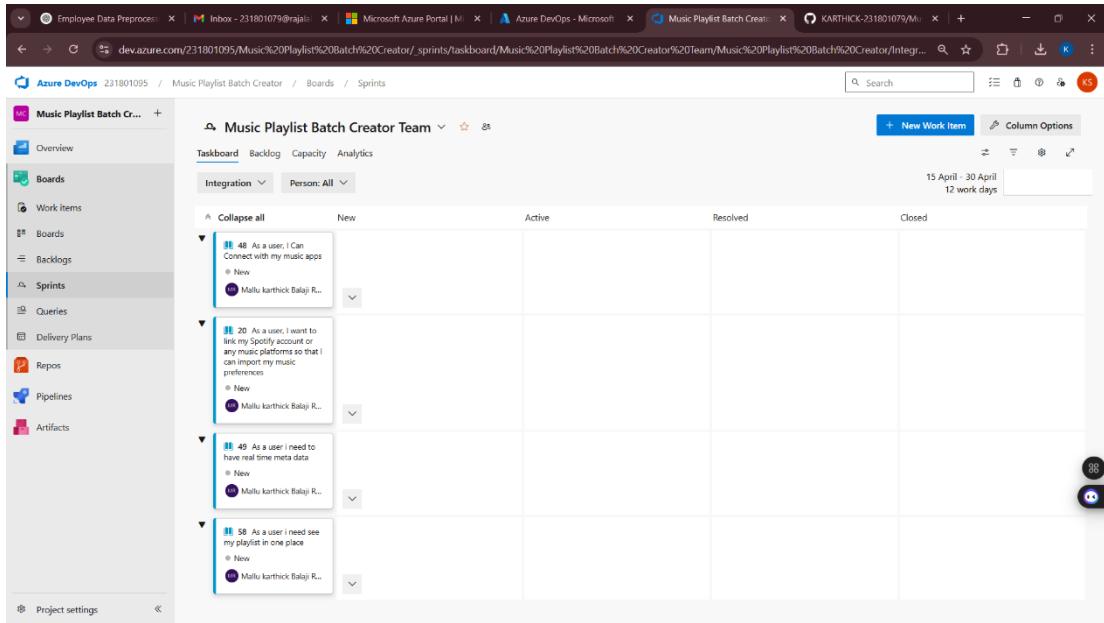


The screenshot shows the Azure DevOps interface for the 'Music Playlist Batch Creator' project. The left sidebar has 'Boards' selected. The main area is the 'Taskboard' for the 'Music Playlist Batch Creator Team'. The backlog section shows three user stories:

- As a user, I should be able to add many songs at once into my playlist
- As a user I should be able to create audio playlist as i need
- Data needed to be collected

The first two stories are marked as 'Unassigned', while the third is marked as 'Resolved' and assigned to 'Karthick S'.

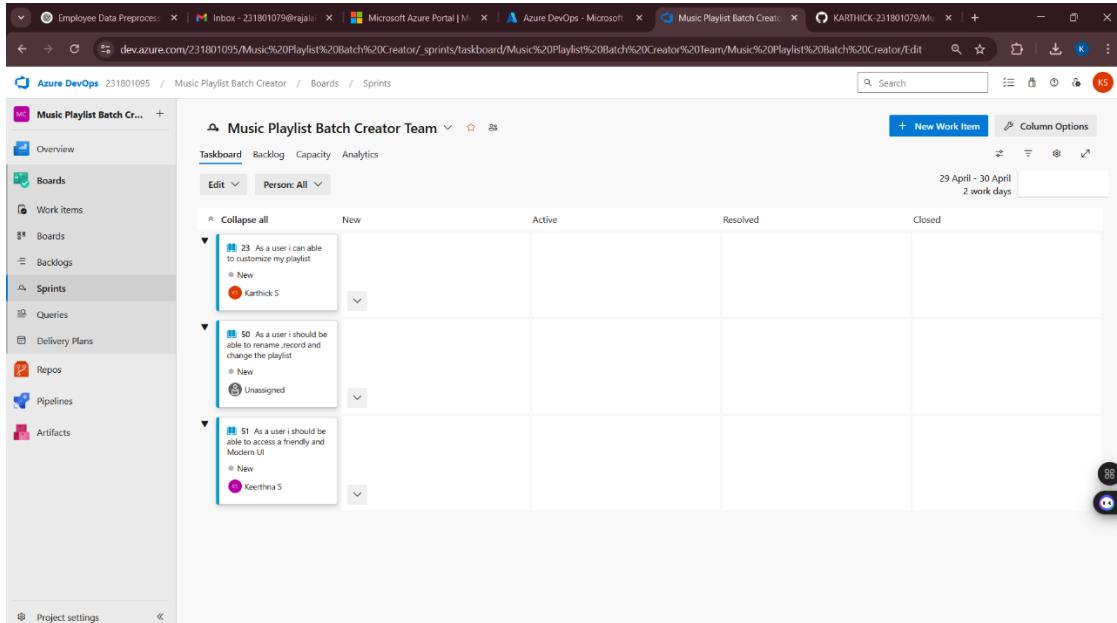
## Sprint 3



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 backlog items visible:

- 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 backlog items visible:

- 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 a Microsoft Azure DevOps User Story card for item 47. The card has the following details:

- User Story:** 47 As a user i should be able to create audio playlist as i need
- Assignee:** Karthick S
- Comments:** 0
- Add Tag:** Add Tag
- Status:** Resolved
- Area:** Music Playlist Batch Creator
- Reason:** Code complete and unit b
- Iteration:** Music Playlist Batch Creator/Model
- Story Points:** 3
- Priority:** 2
- Risk:** 0
- Description:** Click to add Description.
- Acceptance Criteria:** Click to add Acceptance Criteria.
- Classification:** Value area: Business
- Planning:** Story Points: 3
- 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: Parent: 26 Auto-Playlist Creation Based on user preference. Updated Feb 18. New.

### **Result:**

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

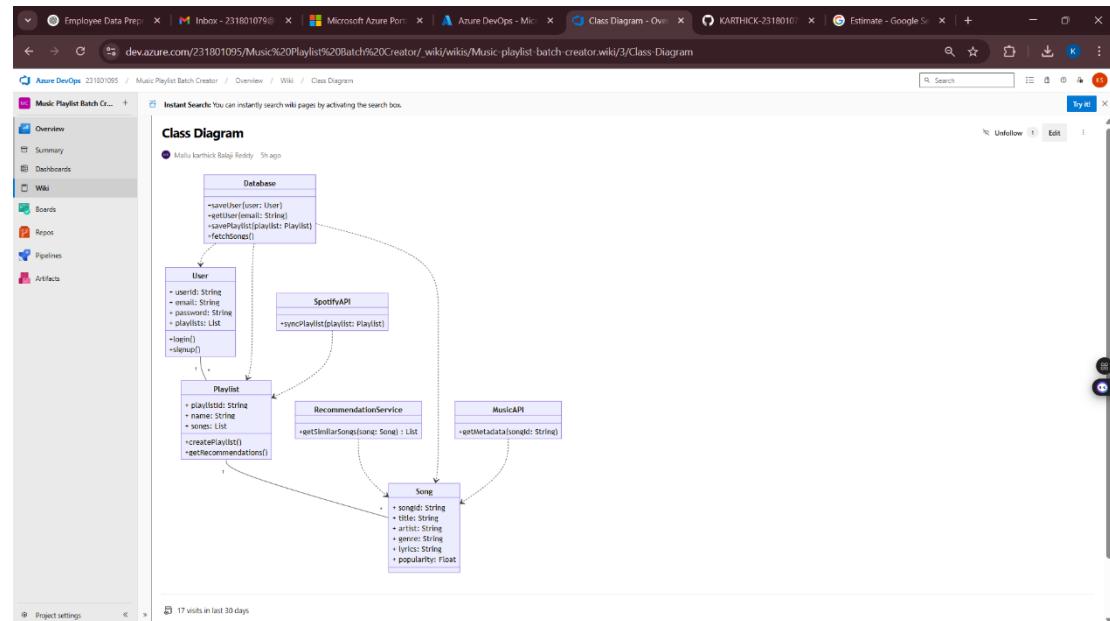
**EXP NO: 6**

## **DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE**

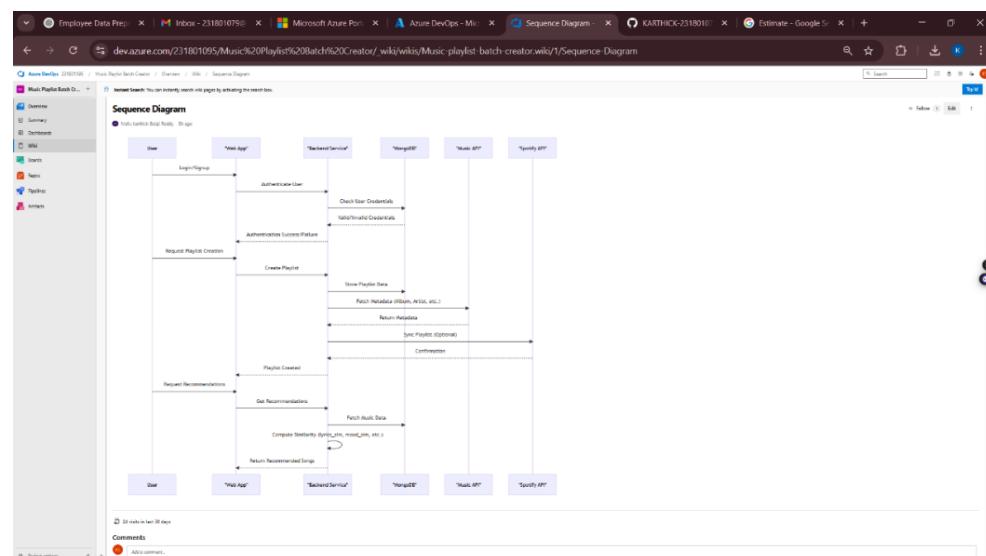
### **Aim:**

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.

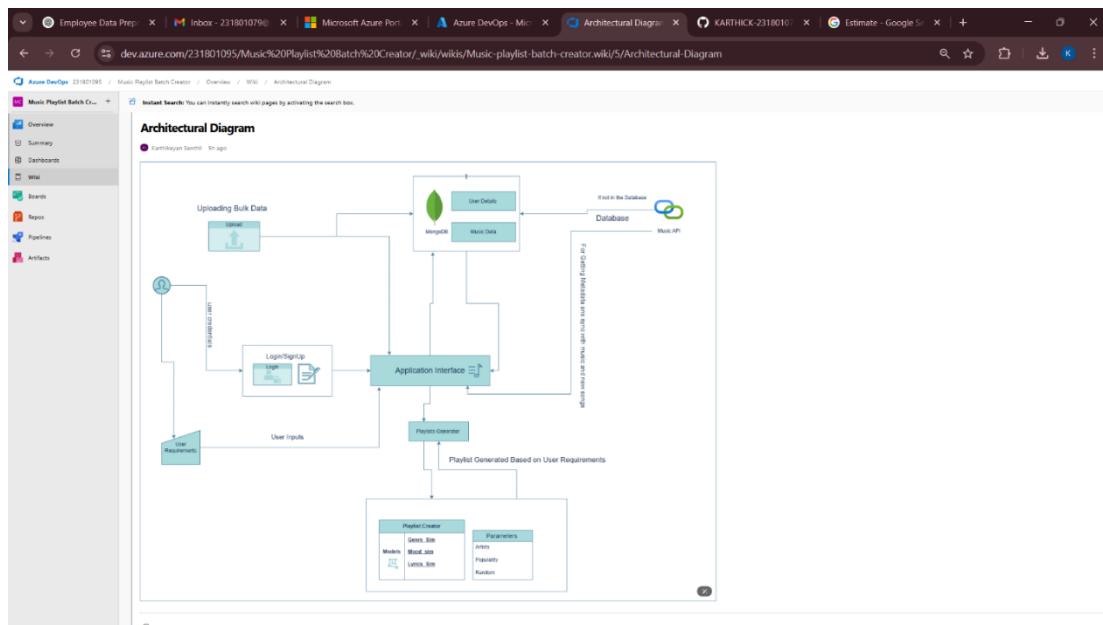
**EXP NO: 7**

## **DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE**

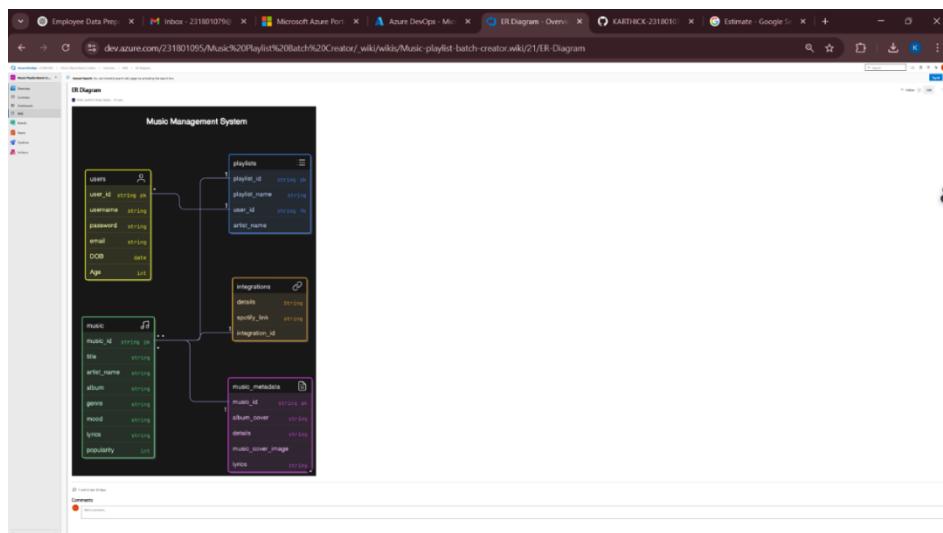
### **Aim:**

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

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

**Aim:**

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

**Test Planning and Test Case****Test Case Design Procedure****1. Understand Core Features of the Application**

- 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

Order	Test Case Id	Assigned To	State
1	78	Karthikeyan Se... Design	
2	80	Karthikeyan Se... Design	
3	81	Karthikeyan Se... Design	
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 Azure DevOps Test Plan interface. A test case titled '77 TC06 – Playlist Loading Failure' is displayed. The test case details include:

- Owner: Karthick S
- Comments: 0
- Add Tag: (button)
- Status: 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	

Below the steps, there is a note: "Click or type here to add a step".

The right side of the screen shows the 'Summary' tab of the test case view, which includes sections for 'Custom' (Type: Error Path), 'Status' (Priority: Z, Automation status: Not Automated), and 'Associated Automation' (with a link to 'Process 231801095').

The screenshot shows the Azure DevOps Test Plan interface. A test case titled "TC05 – View Playlist Page" is selected. The test case details include:

- State:** Design
- Reason:** New
- Area:** Music Playlist Batch Creator
- Iteration:** Music Playlist Batch Creator\Integration

The **Steps** section contains two steps:

1. Log in successfully. Expected result: User is redirected to dashboard.
2. Navigate to "My Playlists" section. Expected result: All created playlists are displayed clearly.

The right side of the screen displays the **Custom** section with the following details:

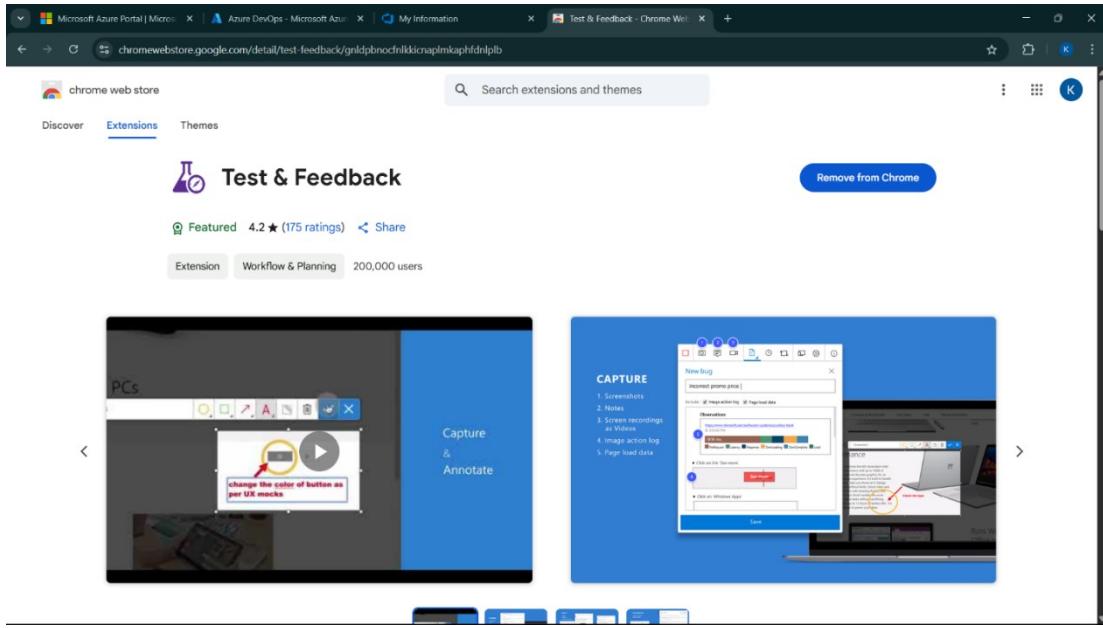
- Type:** Happy Path
- Status:** Priority 2, Automation status: Not Automated

#### 4. Installation of test

The screenshot shows the Chrome Web Store page for the "Test & Feedback" extension. The extension details are as follows:

- Featured:** 4.2 ★ (175 ratings)
- Category:** Workflow & Planning
- Users:** 200,000 users

The extension is described as a tool for capturing screenshots, recording screens, and logging image actions. It includes features like "Capture & Annotate" and "New bug".



## Test and feedback

Showing it as an extension

## 5. Running the test cases

The screenshot shows the Azure DevOps Test Plans 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, under 'Test Suites', there's a list of suites: 'Music Playlist Batch Creator - T...', 'TS01 - User Login (4)', 'TS02 - View Playlists (2)', 'TS03 - Real-Time Met...', 'TS04 - Playlist Editing (4)', and 'TS05 - Smart Playlist ...'. 'TS02 - View Playlists (2)' is selected. Below it, the 'Test Points (2 items)' table lists two entries:

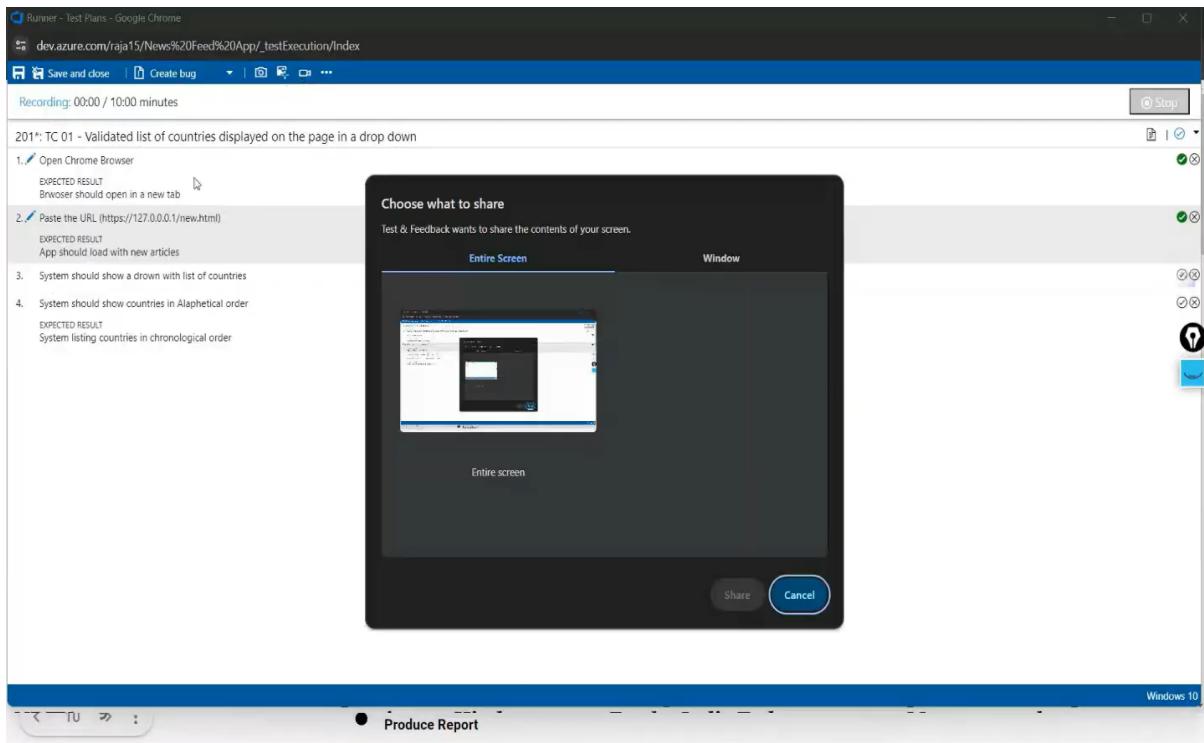
Title	Outcome	Order	Test Case Id	Configuration	Tester
<input checked="" type="checkbox"/> TC05 – View Playlist Page	Passed	1	75	Windows 10	Malu karthick B...
<input type="checkbox"/> TC06 – Playlist Loading Failure	Passed	2	77	Windows 10	Malu karthick B...

A context menu is open over the first row, showing options: View execution history, Mark Outcome, Run, Reset test to active, Edit test case, Assign tester, and View test result.

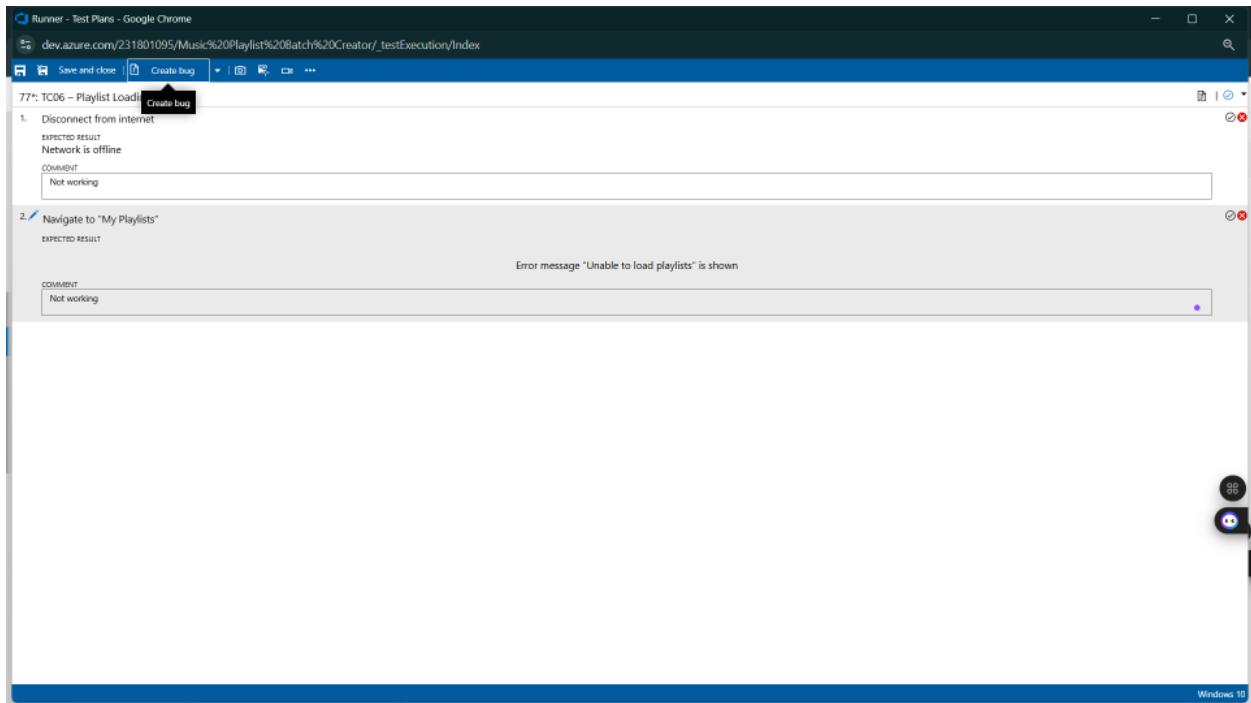
The screenshot shows a browser window titled 'Runner - Test Plans - Google Chrome' displaying the results of test case TC05. The URL is dev.azure.com/231801095/Music%20Playlist%20Batch%20Creator/\_testExecution/Index. The page shows the following steps and results:

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

## 6.Recording the test case



## 7.Creating the bug



The screenshot shows a bug entry in the Azure DevOps Test Plan interface. The title is "TB01 - Playlist loading spinner keeps spinning indefinitely on poor network". The bug is categorized under "Music Playlist Batch Creator" and is currently "Unassigned". The "Repro Steps" section details two failed steps: disconnecting from the internet and navigating to "My Playlists". The "Comments" field notes an error message: "Unable to load playlists" is shown. The "Test Configuration" is listed as Windows 10. The "Planning" and "Deployment" sections provide standard project management details.

This screenshot shows the same bug entry as above, but with the "System Info" section expanded. The "System Info" table contains the following data:

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	x86_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

## 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 progress report: '100% run, 100% passed. [View report](#)'. Below this, 'Test Suites' are listed: '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'. To the right, a modal window titled 'TC05 - View Playlist Page' shows 'Test Case Results' with a table:

Outcome	TimeStamp	Configuration	Run by	Tester	Test PL
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	Karthikyan Senthil	Malu karthick Balaji	... Music
Passed	Saturday	Windows 10	Malu karthick Balaji	Malu karthick Balaji	... Music
Failed	Saturday	Windows 10	Malu 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 a work item. The left sidebar 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 shows 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
- Assignee:** rajesh prabhu
- Comments:** 0
- Add Tag:** None
- State:** New
- Reason:** New
- Iteration:** News Feed App
- Repro Step:**
  - Step no.: 1. Result: Passed. Title: Open Chrome Browser. Description: 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>). Description: Expected Result: App should load with new articles.
  - Step no.: 3. Result: Failed. Title: System should show a dropdown with list of countries. Description: Produce Report.
- Planning:**
  - Resolved Reason: Active
  - Story Points: 1
  - Priority: 2
  - Severity: 3 - Medium
  - Activity: None
- Deployment:** None
- Development:**
  - Add link: Link an Azure Repos commit, pull request or branch to see the status of your development. You can also create a branch to get started.
- Effort (Hours):** Original Estimate: None
- Related Work:** None

- Assigning bug to the developer and changing state

The screenshot shows a detailed view of a test plan in Azure DevOps. A specific test step has failed:

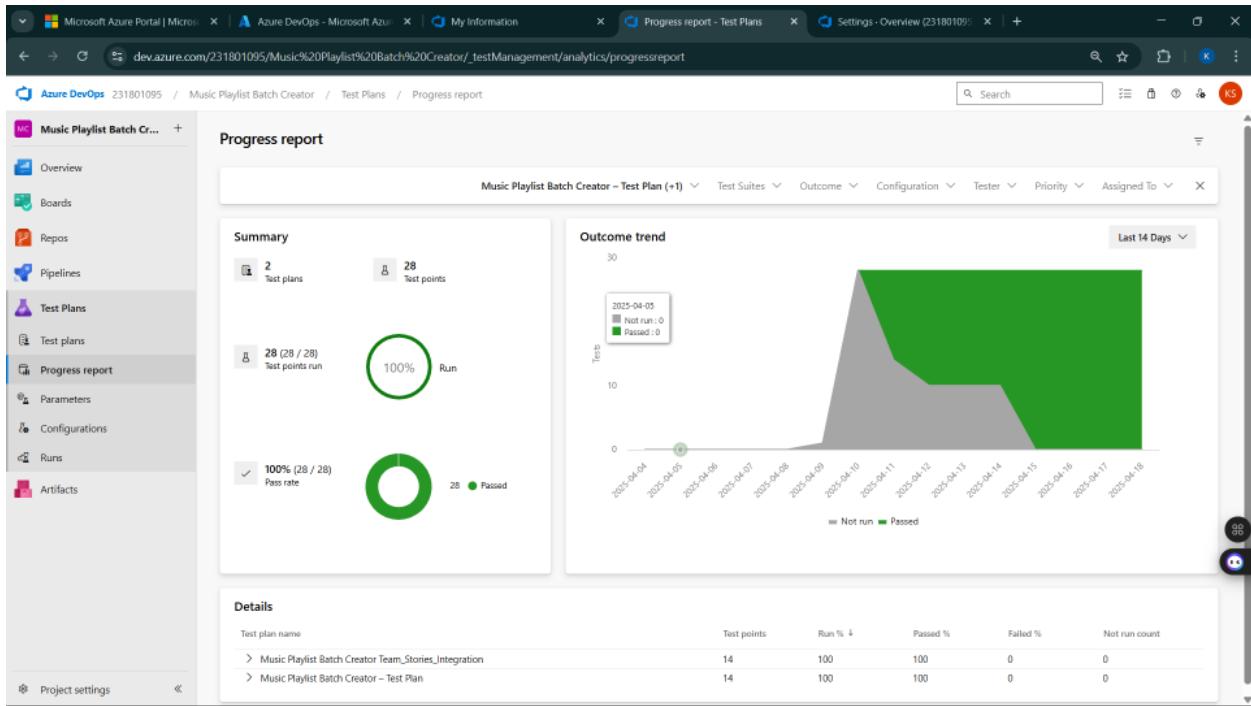
- Step no.**: 1
- Result**: Failed
- Title**: Disconnect from internet
- Comments**: Page Not loading
- Expected Result**: Network is offline
- Actual Result**: Error message "Unable to load playlists" is shown

The test configuration is set to Windows 10. The deployment section indicates it's linked to a release item. Related work and system info sections are also visible.

## 10. Progress report

The screenshot shows the progress report for the test plan. It includes the following data:

Test plan name	Test points	Run %	Passed %	Failed %	Not run count
Music Playlist Batch Creator - Test Plan	14	100	100	0	0
T501 - User Login	4	100	100	0	0
T502 - View Playlists	2	100	100	0	0
T503 - Real-Time Metadata	2	100	100	0	0
T504 - Playlist Editing	4	100	100	0	0
T505 - Smart Playlist Creation	2	100	100	0	0



## 11. Changing the test template

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 page. The 'Processes' tab is selected. 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 page. The 'Processes' tab is selected. 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
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 overlaid on the Azure DevOps settings interface. The dialog has a 'Definition' tab selected, showing options to 'Use an existing field' (with 'Acceptance Criteria' selected) or 'Create a field'. The 'Create a field' section includes fields for 'Name' (set to 'Type'), 'Type' (set to 'Text (single line)'), and 'Description' (with the placeholder 'Optionally provide a description for the field'). At the bottom are 'Add Field' and 'Cancel' buttons.

The screenshot shows the 'Work item types' page in the Azure DevOps settings. It lists a single work item type: 'Music Playlist Batch Creator'. The table has columns for 'Name' (containing 'Music Playlist Batch Creator') and 'Description' (containing the text 'The Azure Music Playlist Batch Creator is a cloud-based solution designed for bulk playlist creation and management. Levera...'). The 'General' section of the sidebar is visible on the left.

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](https://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". The left sidebar is titled "Organization Settings" and includes sections for General, Security, Boards, Pipelines, and Process. The "Process" section is currently selected. The main content area shows a "Steps" field with the placeholder "Text (multiple lines)". To the right, there are several sections: "Custom" (Type: Text (single line)), "Recent test results" (Recent test case results), "Deployment" (Deployments), "Development" (Links), "Related Work" (Links), and "Status" (Priority: Integer, Automation 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	<b>LOAD TESTING AND PIPELINES</b>
-----------	-----------------------------------

## **LOAD TESTING AND PIPELINES**

### **Aim:**

To create an Azure Load Testing resource and run a load test to evaluate the performance of a target endpoint and to create and demonstrate an Azure DevOps pipeline for automating application builds, tests, and deployment.

### **Load Testing**

#### **Azure Load Testing:**

Azure Load Testing allows you to simulate high traffic and stress tests for your web applications and APIs to understand how they perform under load. It helps identify performance bottlenecks, scalability issues, and optimize resource usage before deployment.

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

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

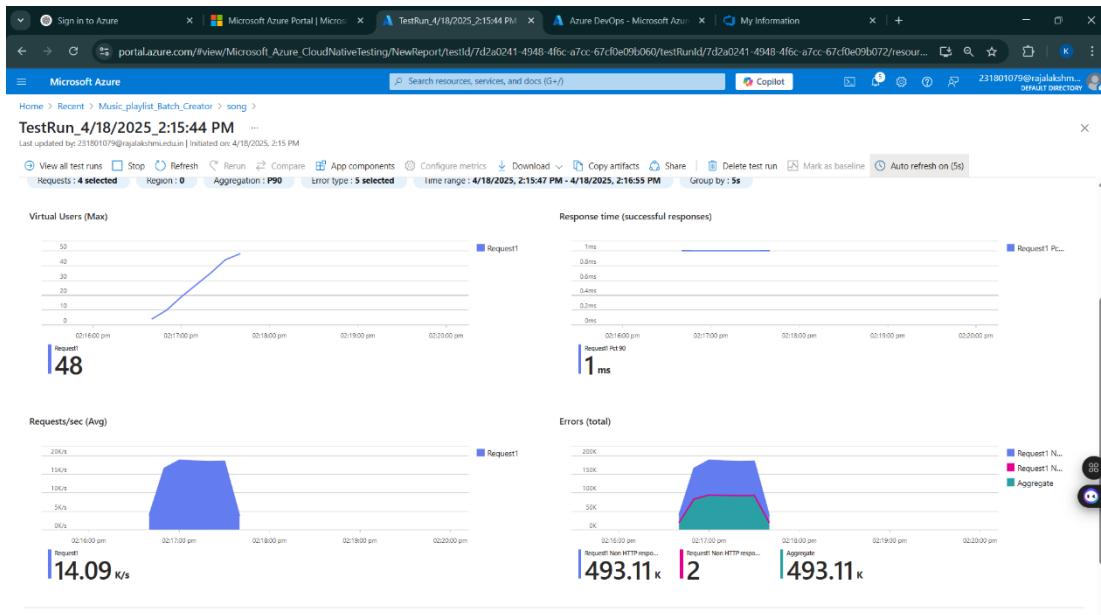
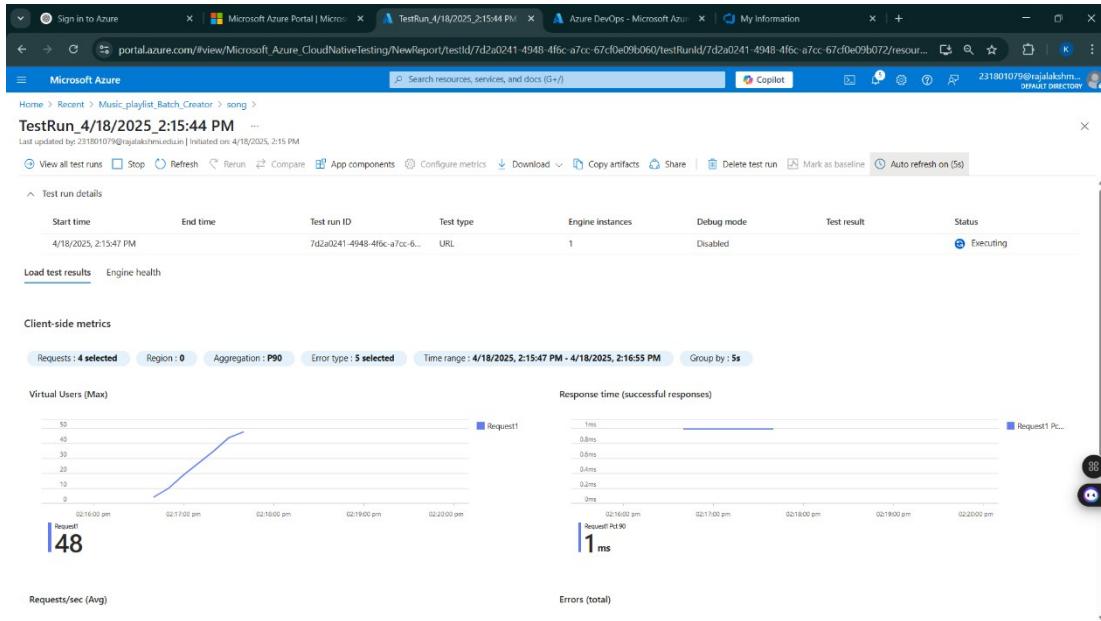
1. Sign in to Azure Portal  
Go to <https://portal.azure.com> and log in.
2. Create the Resource
  - o Go to *Create a resource* → Search for “Azure Load Testing”.
  - o Select Azure Load Testing and click Create.
3. Fill in the Configuration Details
  - o *Subscription*: Choose your Azure subscription.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

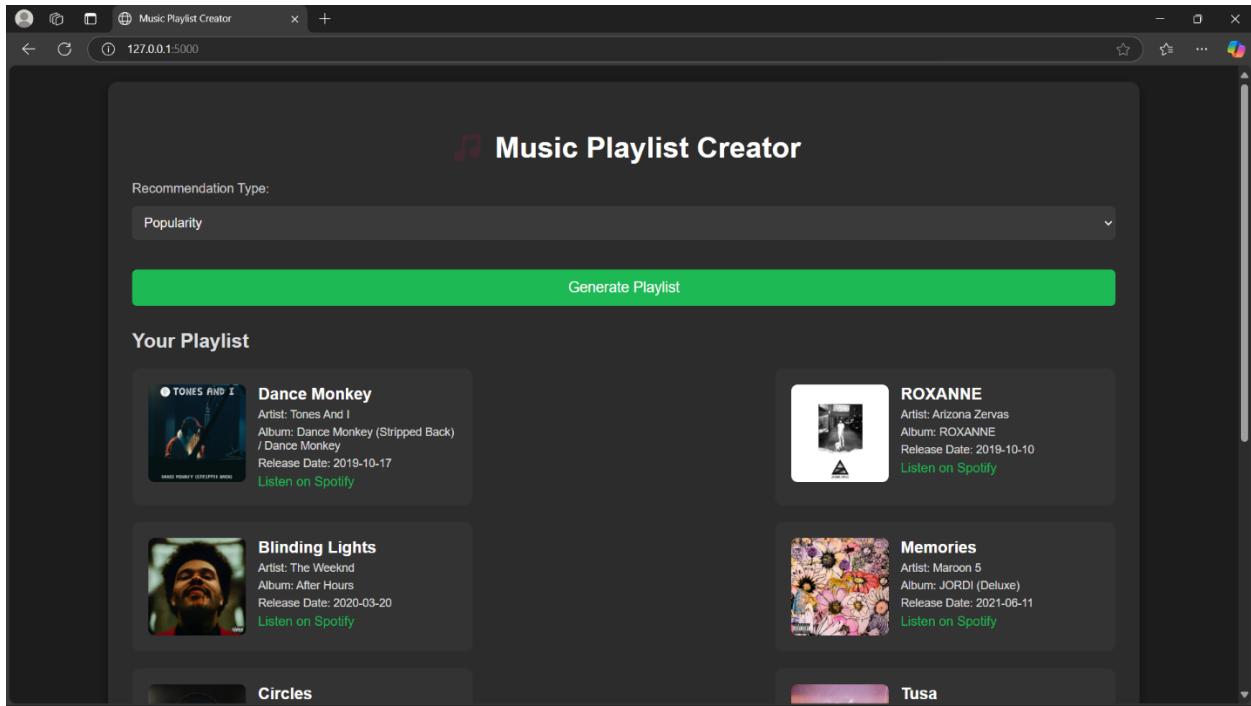
#### **Steps to Create and Run a Load Test:**

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:

## yml 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:

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

### 4. Run and Monitor Pipeline:

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

## Pipeline

The screenshot shows the Azure DevOps Pipeline page for a project named 'Music Playlist Batch Creator'. The pipeline run is identified as '#20250424.3 • Pipeline 2' and was triggered by 'Music Playlist Batch Creator (9)'. A note indicates that this run is being retained as one of three recent runs by the main branch. The summary section shows the repository and version information: 'Repository and version' (Music Playlist Batch Creator) and 'Branch' (main). The run started 'Just now' and took '24s'. Related items include 0 work items and 0 artifacts. A link to 'View 52 changes' and 'Tests and coverage' is provided, along with a 'Get started' button. Below the summary, a 'Jobs' table lists a single job named 'Job' which completed successfully in 6s. The pipeline navigation bar on the left includes options like Overview, Boards, Repos, Pipelines, Environments, Releases, Library, Task groups, Deployment groups, Test Plans, and Artifacts.

### Result:

Successfully created the Azure Load Testing resource and executed a load test to assess the performance of the specified endpoint and also demonstrated pipelines in azure devops.

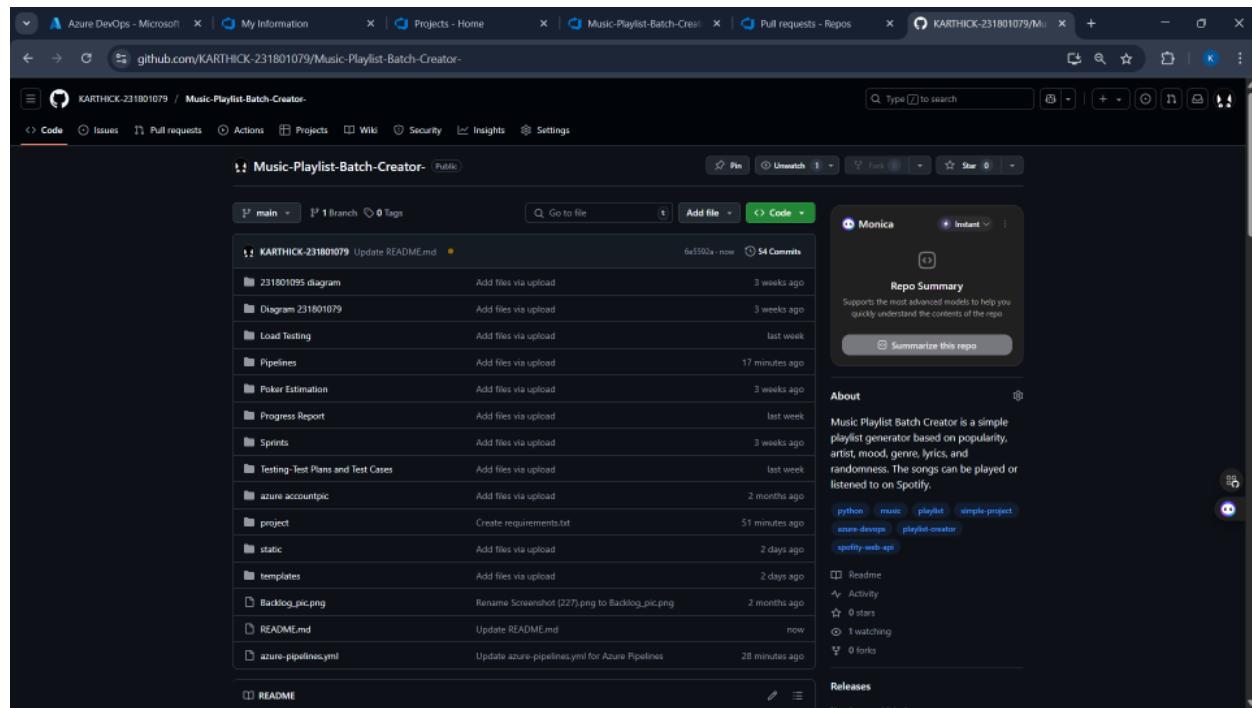
**EXP NO: 10**

## **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.