



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

## **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING 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

# INDEX

S.No.	Date	Title
1.	21/1/25	Azure Devops Environment Setup.
2.	21/1/25	Azure Devops Project Setup and User Story Management.
3.	28/1/25	Setting Up Epics, Features, And User Stories for Project Planning.
4.	11/2/25	Sprint Planning.
5.	18/2/25	Poker Estimation.
6.	25/2/25	Designing Class and Sequence Diagrams for Project Architecture.
7.	04/3/25	Designing Architectural and ER Diagrams for Project Structure.
8.	25/3/25	Testing – Test Plans and Test Cases.
9.	15/4/25	Load Testing and Pipelines.
10.	22/4/25	GitHub: Project Structure & Naming Conventions.

**EXP NO: 1**

## **AZURE DEVOPS ENVIRONMENT SETUP**

### **Aim:**

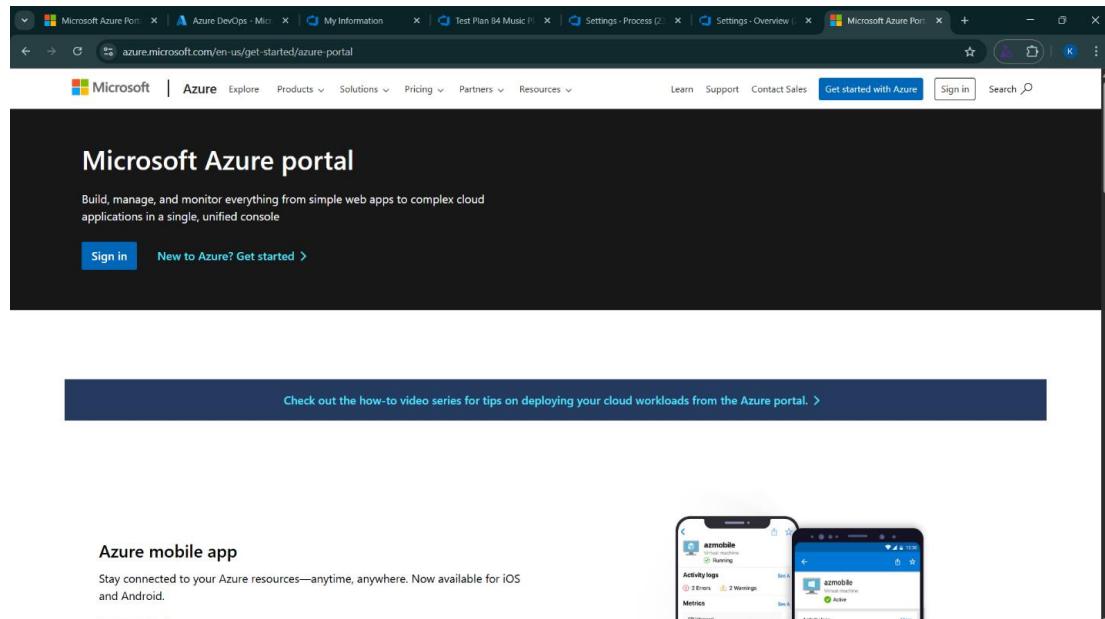
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

### **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

Welcome to Azure!

Don't have a subscription? Check out the following options.

**Start with an Azure free trial**  
Get \$200 free credit toward Azure products and services, plus 12 months of popular free services.

**Manage Microsoft Entra ID**  
Manage access, set smart policies, and enhance security with Microsoft Entra ID.

**Azure for Students**  
Get free software, Azure credit, or access Azure Dev Tools for Teaching after you verify your academic status.

**Azure services**

**Resources**

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

is and I

DevOps

All Services Marketplace More (4)

Services

- Azure Native New Relic Service
- Managed DevOps Pools
- Azure DevOps organizations
- Azure Native Dynatrace Service

Kubernetes services More services

Marketplace

- Static Web App
- Rocky Linux 9
- Build Agents for Azure DevOps
- InfluxDB Cloud (Official Version)

Last Viewed

- 3 days ago
- 3 days ago

Documentation

- DevOps architecture design - Azure Architecture Center
- Secure your Azure DevOps - Azure DevOps
- Course AZ-400T00-A: Designing and Implementing Microsoft DevOps solutions - T...
- Managed DevOps Pools Overview - Managed DevOps Pools

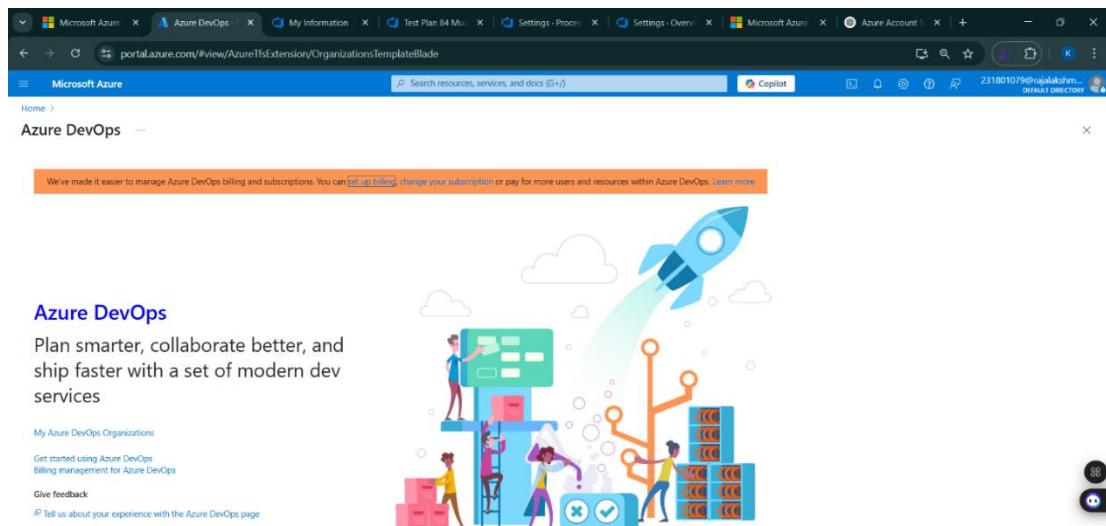
Microsoft Entra ID

- Azure DevOps
- Continue searching in Microsoft Entra ID

Cost Management

Analyze and optimize your cloud spend for free

4. Click on the **My Azure DevOps Organization** link and create an organization and you should be taken to the Azure DevOps Organization Home page.



### Result:

Successfully accessed the Azure DevOps environment and created a new organization through the Azure portal.

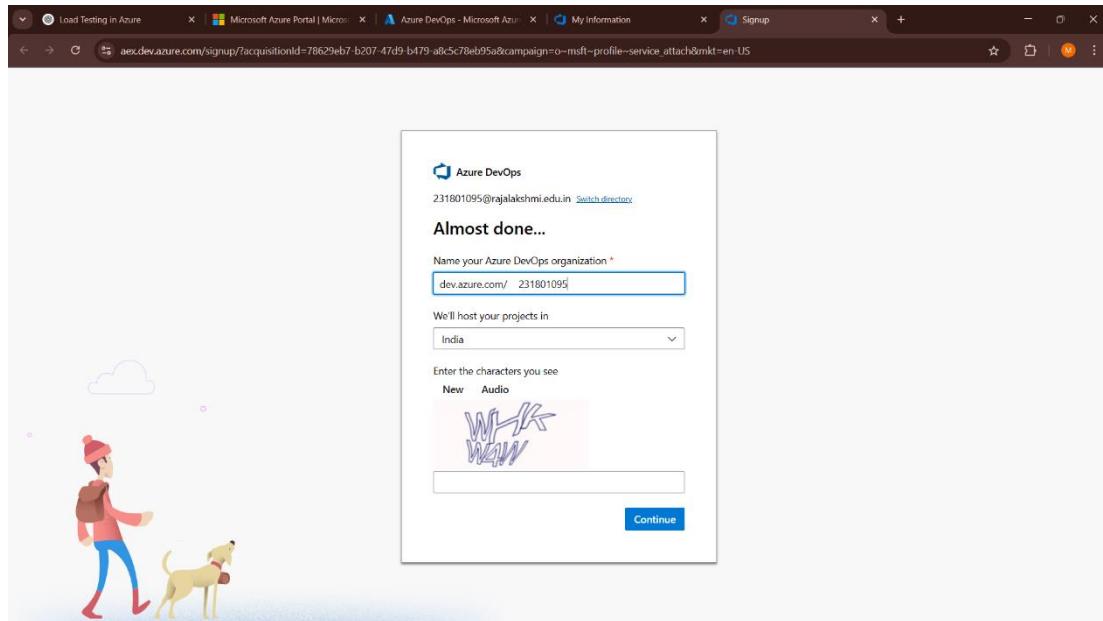
**EXP NO: 2**

## **AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT**

### **Aim:**

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

### **1.Create An Azure Account**



### **2.Create the First Project in Your Organization**

- After the organization is set up, you'll need to create your first **project**. This is where you'll begin to manage code, pipelines, work items, and more.
- On the organization's **Home page**, click on the **New Project** button.
- Enter the project name, description, and visibility options:
  - Name:** Choose a name for the project (e.g., **LMS**).
  - Description:** Optionally, add a description to provide more context about the project.
  - Visibility:** Choose whether you want the project to be **Private** (accessible only to those invited) or **Public** (accessible to anyone).
- Once you've filled out the details, click **Create** to set up your first project.

## Create new project

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, the title 'Create new project' is displayed. Below it, the 'Project name \*' field contains 'Music Playlist Batch Creator'. The 'Description' field is empty. Under 'Visibility', the 'Public' option is selected, indicated by a blue border around its panel. The 'Private' option is also shown. Below the visibility section, a note states: 'By creating this project, you agree to the Azure DevOps [code of conduct](#)'. There is an 'Advanced' button above the configuration sections. The 'Version control' dropdown is set to 'Git'. The 'Work item process' dropdown is set to '231801095 Agile'. At the bottom right, there 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 blue header bar with the Microsoft logo and the user name "Abhinavu Prasad" along with a "Sign out" link. Below the header, on the left, is a circular profile picture with the letters "AP". To the right of the profile picture, the user's name "Abhinavu Prasad" and email "231801002@rajalakshmi.edu.in" are displayed, with a "Edit profile" link. Below this, there's contact information: "India" with a location icon and the email "231801002@rajalakshmi.edu.in". A horizontal line separates this from a section titled "Visual Studio Dev Essentials" which says "Get everything you need to build and deploy your app on any platform." and "Use your benefits". On the right side of the dashboard, under "Azure DevOps Organizations", there's a list of projects: "SMS-Text Messaging Service" (Owner), "SMS-Text 2", and "New project". There's also a link to "dev.azure.com/231801032 (Member)". At the top right of this section, there's a "Create new organization" button and an "Actions" link with a "Open in Visual Studio" option. The overall interface is white with blue and grey accents.

#### 4. Project dashboard

The screenshot shows the Azure DevOps project summary page for the "SMS-Text Messaging Service" project. At the top, there's a navigation bar with the project ID "231801002 / SMS-Text Messaging Service / Overview / Summary". To the right of the navigation bar are search, filter, and public/share buttons. The main content area has a title "SS SMS-Text Messaging Service". On the left, there's a sidebar with various icons for project management tasks like backlog, boards, and reports. The main content is divided into sections: "About this project" (which contains a brief description of the system as a cloud-native SMS and text messaging application), "Project stats" (which shows 0 work items created and 0 work items completed over the last 7 days), and "Members" (which lists six team members represented by their initials in colored circles). The overall interface is clean and modern, using a light blue and white color scheme.

## 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 Boards page in Microsoft Azure DevOps. The left sidebar has icons for Home, Boards, Backlogs, Analytics, and More. The main area shows the 'SMS-Text Messaging Service Team' backlog under the 'Backlog' tab. The backlog table has columns: Order, Work Item Type, Title, State, Effort, Business Area, and Tags. There are 15 items listed, including an Epic, several Features, User Stories, and a Bug. The 'Planning' pane on the right shows iterations: Iteration 4 (Planned Effort: 5 days), Iteration 5 (Planned Effort: 2 days), Iteration 2 (No work scheduled yet), and Iteration 3 (No work scheduled yet). A search bar at the top right contains 'Search'.

The screenshot shows the Microsoft account sign-in page. At the top, there's a search bar, a 'Sign out' link, and a 'Microsoft' logo. Below that, a large blue circular profile picture with the letters 'AP' is displayed. To the right of the profile picture, the name 'Abhinavu Prasad' and the email '231801002@rajalakshmi.edu.in' are shown, along with links for 'My Microsoft account' and 'Switch directory'. At the bottom, there's a 'Sign in with a different account' button with a person icon. The left side of the page has a 'Work Item' button and a 'Tags' section.

**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 Backlog board for the 'SMS-Text Messaging Service Team'. The backlog is organized into several levels of hierarchy:

- Level 1 (Epics):** SMS-Text Messaging Service (New, Business)
- Level 2 (Features):** login page (Closed, Business), to provide the user a login page to log in (Resolved, Business), Search feature (Closed, Business), to provide searching feature for users (New, Business), Interface (Closed, Business), chatting interface for user to send messages to other ... (Active, Business), homepage interface (New, Business)
- Level 3 (User Stories):** Profile Page (New, Business), to change username and profile picture (New, Business), logout page (New, Business), grouping feature (New, Business), to provide the ability to make groups among users (New, Business)
- Level 4 (Tasks):** searched result is not visible (New, Business)

On the right side of the board, there is a 'Planning' panel titled 'Drag and drop work items to include them in a sprint.' It lists three sprints:

- Iteration 4:** 28/03/2025 - 03/05/2025, Planned Effort: 5 - 32 working days
- Iteration 5:** 05/05/2025 - 10/06/2025, Planned Effort: 2 - 32 working days
- Iteration 2:** No work scheduled yet
- Iteration 3:** (No work scheduled yet)

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

The screenshot shows the 'New Epic' creation dialog in Azure DevOps. The dialog has the following fields filled in:

- Description:** Develop a system that allows users to create and manage multiple playlists in batches
- Planning:** Priority: 2, Risk: 1, Business Value: 1, Time Criticality: 1, Start Date: Select a date..., Target Date: Select a date...
- 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
- Classification:** Value area: Business

## 2.Fill in Features

A screenshot of a Feature work item in Azure DevOps. The top navigation bar shows 'FEATURE 2'. Below it, the work item details are as follows:

- Title:** 2 login page
- ID:** 231801003
- Comments:** 0
- Add Tag:** Add Tag
- Save and Close:** Save and Close
- Follow:** Follow
- Updated:** Updated

Below the title, there are sections for State (Closed), Reason (Acceptance tests pass), Area (SMS-Text Messaging Service), Iteration (SMS-Text Messaging Service), and Details.

The main content area has tabs for Description, Planning, Deployment, Discussion, and Development.

- Description:** Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request. (with a 'switch to Markdown editor' link)
- Planning:** Priority: 2, Risk: 2, Effort: 1, Business Value: 1, Time Criticality: 1, Start Date: Select a date..., Target Date: [empty]
- 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 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

A screenshot of a User Story work item in Azure DevOps. The top navigation bar shows 'USER STORY 14'. Below it, the work item details are as follows:

- Title:** 14 to provide searching feature for users
- Assignee:** Dharshana Srinivasulu
- Comments:** 0
- Add Tag:** Add Tag
- Save and Close:** Save and Close
- Follow:** Follow
- Updated:** Updated by Dharshana Srinivasulu: Feb 26

Below the title, there are sections for State (New), Reason (New), Area (SMS-Text Messaging Service), Iteration (SMS-Text Messaging Service\Iteration 1), and Details.

The main content area has tabs for Description, Planning, Deployment, Acceptance Criteria, Classification, Development, and Discussion.

- Description:** Click to add Description.
- Planning:** Story Points: 5, Priority: 2, Risk: 2
- 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.
- Acceptance Criteria:** Click to add Acceptance Criteria.
- Classification:** [empty]
- 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.
- Discussion:** Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request.

**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 a taskboard interface for the "SMS-Text Messaging Service Team". The top navigation bar includes "Taskboard", "Backlog", "Capacity", and "Analytics". Below this, filters show "Iteration 1" and "Person: All". The main area displays a grid with columns for "New", "Active", and "Resolved". A single user story is visible in the "New" column, with a tooltip providing details: "6 to provide the user a login page to log in", status "Resolved", and ID "231801003".

#### **Sprint 2**

The screenshot shows a taskboard interface for the "SMS-Text Messaging Service Team". The top navigation bar includes "Taskboard", "Backlog", "Capacity", and "Analytics". Below this, filters show "Iteration 2" and "Person: All". The main area displays a grid with columns for "New", "Active", and "Resolved". A single user story is visible in the "New" column, with a tooltip providing details: "23 homepage interfac...", status "New", and assigned to "Gauhar SR".

## Sprint 3

The Taskboard for Sprint 3 displays two work items:

- Work Item 13:** To change username and profile picture. Status: New. Assigned to Dharshana Srinivasulu.
- Work Item 14:** To provide searching feature for users. Status: New. Assigned to Dharshana Srinivasulu.

## Sprint 4

The Taskboard for Sprint 4 displays two work items:

- Work Item 7:** Chatting interface for user to send messages to other user. Status: Active. Assigned to Boopesh R.
- Work Item 16:** To provide the ability to make groups among users. Status: New. Assigned to Abhinavu Prasad.

Timeline: 28 March - 3 May, 5 work days remaining.

**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 detailed view of a User Story card in Azure DevOps. The card has the following details:

- Title:** USER STORY 7  
7 chatting interface for user to send messages to other user
- Assignee:** Boopesh R
- Comments:** 0 Comments
- Add Tag:** Add Tag
- State:** Active
- Area:** SMS-Text Messaging Service
- Reason:** Acceptance tests fail
- Iteration:** SMS-Text Messaging Service\Iteration 4
- Updated by:** Abhinav Prasad: Apr
- Details:** 2 comments, 0 attachments

The card is divided into several sections:

- Description:** Click to add Description.
- Acceptance Criteria:** Click to add Acceptance Criteria.
- Discussion:** Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request.
- Planning:** Story Points: 5, Priority: 2, Risk.
- Classification:**
- Deployment:** To track releases associated with this work item, go to [Releases](#) and turn on deployment status reporting for Boards in your pipeline's Options menu. [Learn more about deployment status reporting](#).
- Development:** Add link. Link an Azure Repos [commit](#), [pull request](#) or [branch](#) to see the status of your development. You can also [create a branch](#) to get started.
- Related Work:**

### **Result:**

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

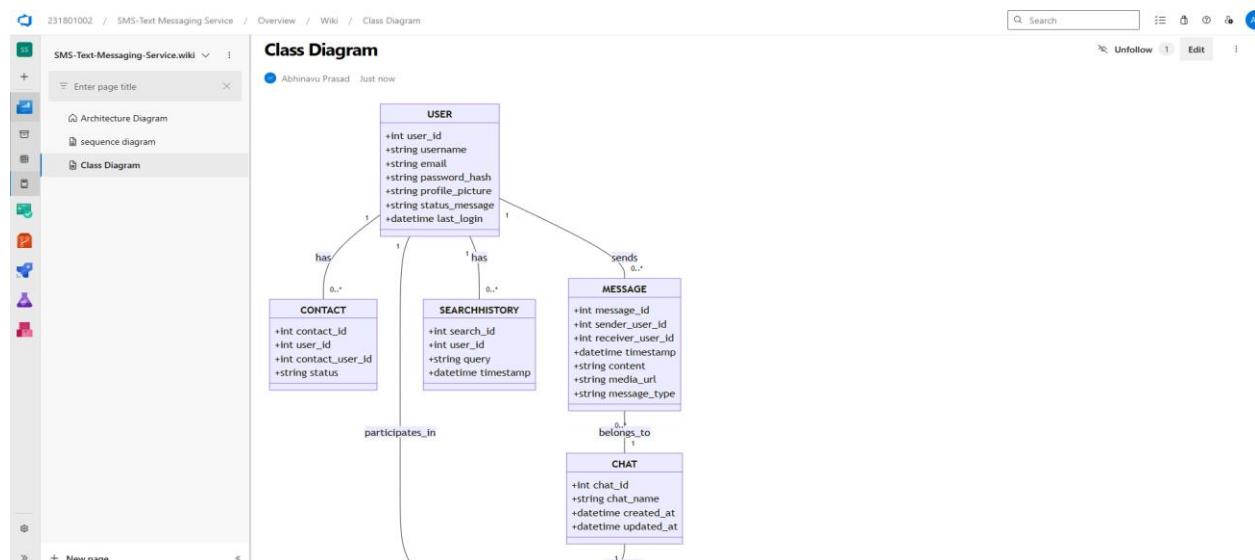
**EXP NO: 6**

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

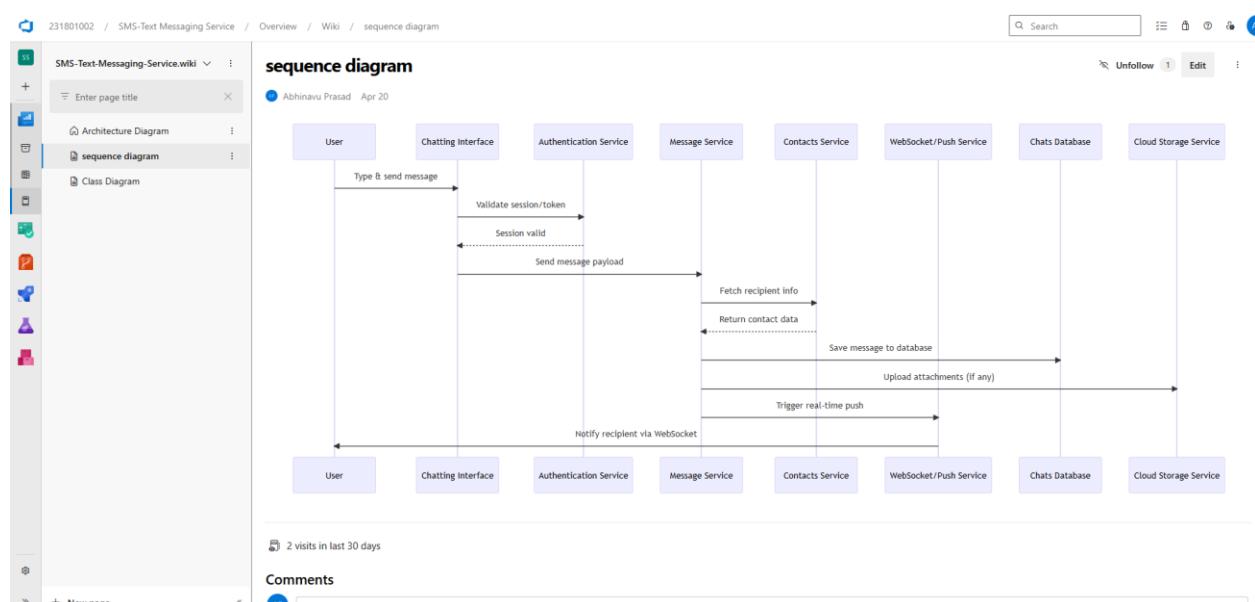
### **Aim:**

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.

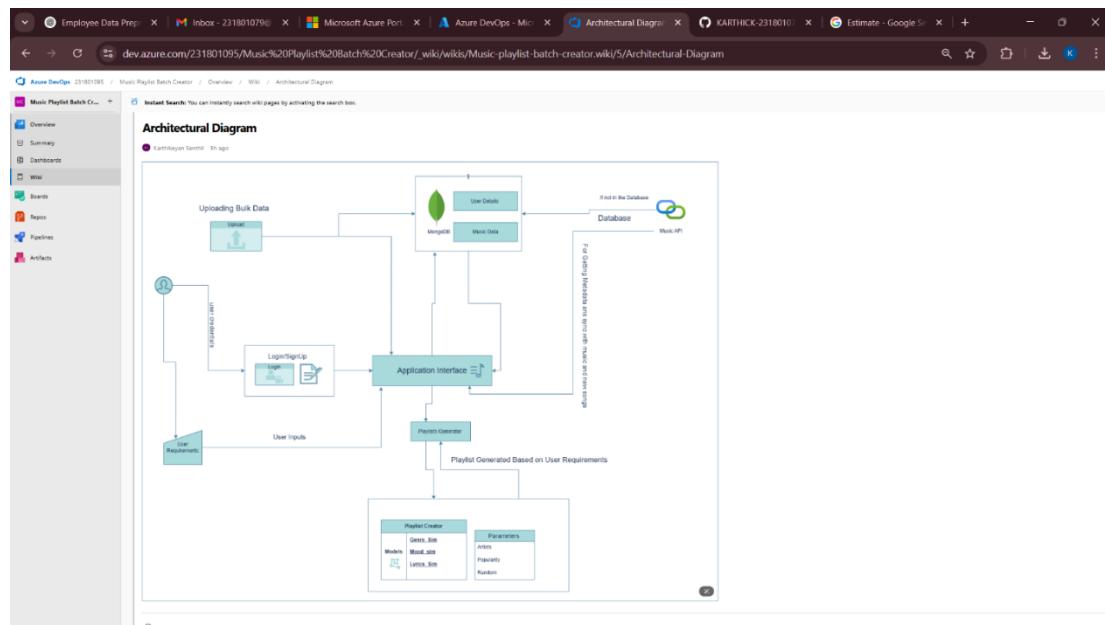
**EXP NO: 7**

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

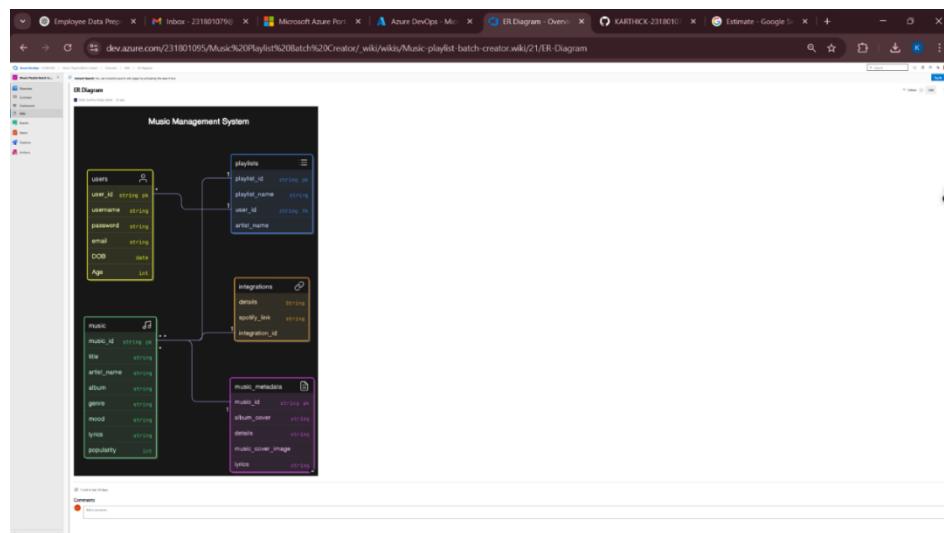
### **Aim:**

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

Title	Outcome	Order	Test Case Id	Configuration	Tester
tc1 check search feature	Passed	1	28	Windows 10	Abhinavu Prasad
tc2 check sending and receiving of data	Passed	2	32	Windows 10	Abhinavu Prasad
tc3 check group	Passed	3	41	Windows 10	Abhinavu Prasad

### **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

- o Action:

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

- o Expected Results:

- Playlist is generated based on selected mood and categories.

- o Type: Happy Path

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

- o Action:

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

- o Expected Results:

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

- o 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 are as follows:

- Category:** Music
- Assignee:** Karthick S
- Status:** Design
- Area:** Music Playlist Batch Creator
- Iteration:** Music Playlist Batch Creator/Integration
- Reason:** New
- Last Update:** Updated by Karthick S 14m ago

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 **Custom** section shows the type is "Error Path". The **Status** section indicates Priority Z and Automation status Not Automated.

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

- Owner:** Karthick S
- Comments:** 0 Comments
- Add Tag:**
- Status:** Design
- Reason:** New
- 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 **Custom** section shows the following details:

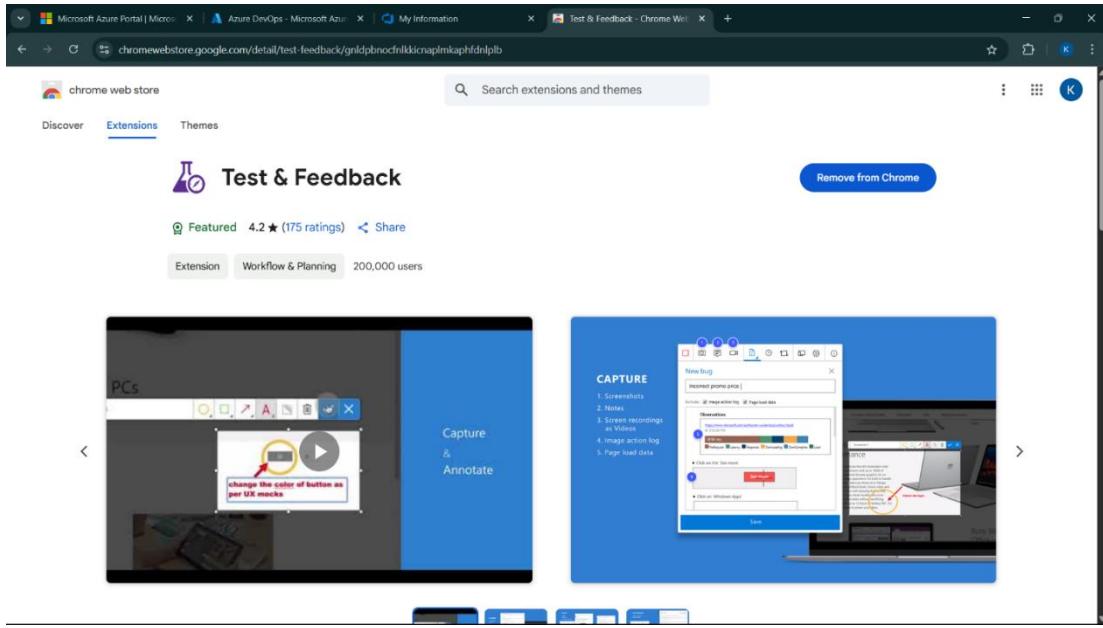
- Type:** Happy Path
- Status:** Priority 2
- Automation status:** Not Automated

#### 4. Installation of test

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

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

The extension is described as a tool for capturing screenshots, screen recordings, and image action logs. It includes features like 'Capture & Annotate' and 'New bug' reporting. The 'Add to Chrome' button is visible at the top right.



## Test and feedback

Showing it as an extension

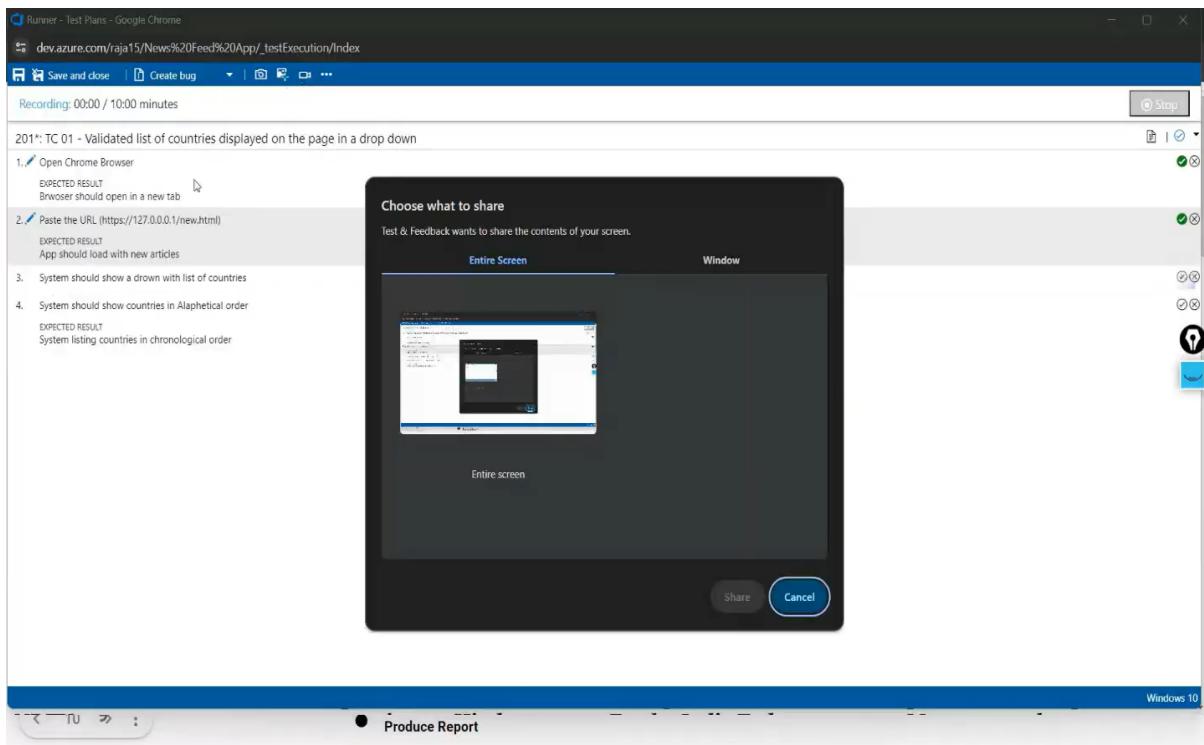
A screenshot of a web browser window showing the Azure DevOps Test Plan interface for a project named "Music Playlist Batch Creator". The "Test Plans" section is selected in the sidebar. A modal window titled "Extensions" is open, listing the "Test &amp; Feedback" extension as having "Full access" and being installed. Other extensions listed include "Copy Text from Picture", "Dark Reader", "Monica: ChatGPT AI Assist...", "Selectext: Copy text from V...", and "Test &amp; Feedback".

## 5. Running the test cases

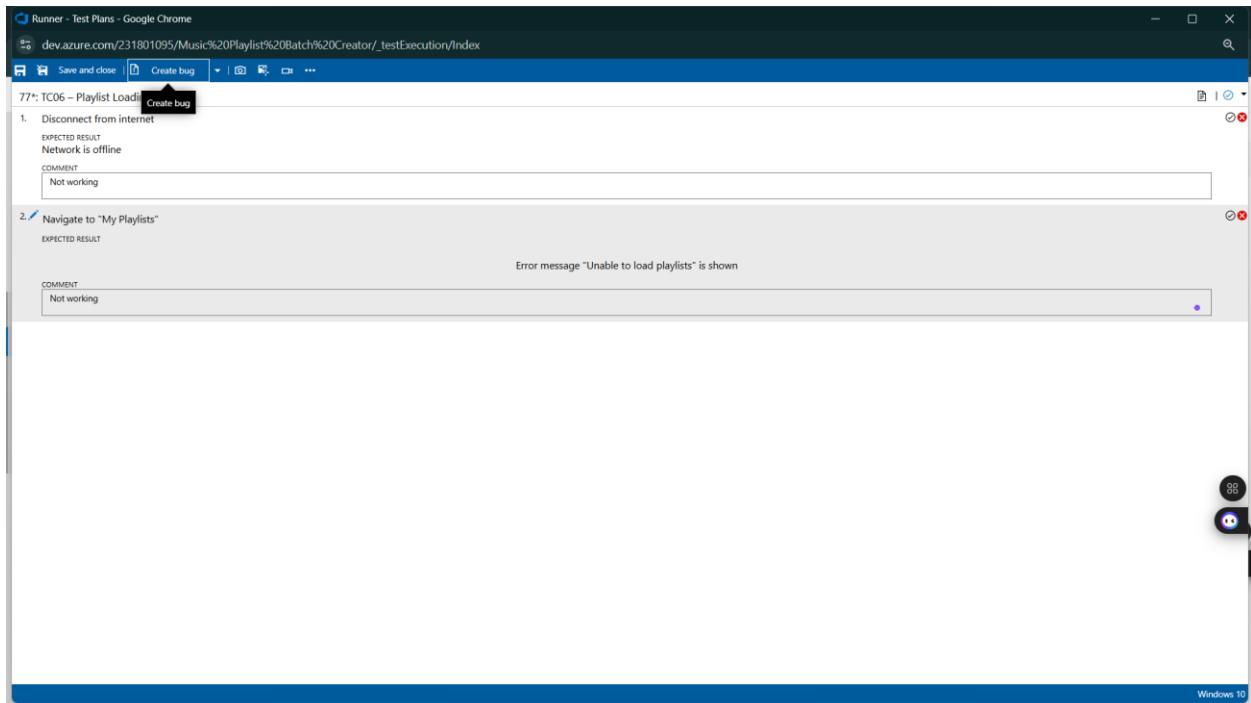
The screenshot shows the Azure DevOps Test Plan interface. On the left, there's a navigation sidebar with options like Overview, Boards, Repos, Pipelines, Test Plans, Test plans, Progress report, Parameters, Configurations, Runs, and Artifacts. Under Test Plans, 'Music Playlist Batch Creator' is selected. In the center, under 'Test Suites', 'TS02 - View Playlists (ID: 87)' is selected. The 'Execute' tab is active, showing 'Test Points (2 items)'. The first item is 'TC05 – View Playlist Page', which has a green 'Passed' status, an order of 1, and a test case ID of 75. The second item is 'TC06 – Playlist Loading Failure', also passed. A context menu is open over the first test point, listing options: View execution history, Mark Outcome (checkbox checked), Run (checkbox checked), Reset test to active, Edit test case, Assign tester, and View test result.

The screenshot shows the 'Test Execution' page for 'TC05 – View Playlist Page'. It displays a single test step: '1. Log in successfully'. Below it, the 'EXPECTED RESULT' is listed as 'User is redirected to dashboard'. At the bottom, there's a note: '2. Navigate to "My Playlists" section' and 'EXPECTED RESULT' 'All created playlists are displayed clearly'. The browser title is 'Runner - Test Plans - Google Chrome'.

## 6.Recording the test case



## 7.Creating the bug



The screenshot shows a bug work item in the Azure DevOps Test Plan interface. The title is "TB01 - Playlist loading spinner keeps spinning indefinitely on poor network". The work item is unassigned and has a priority of 2 and severity of 3 - Medium. It was filed on 18-04-2025 at 03:23. The repro steps describe a failure to connect to the internet and navigate to "My Playlists". The test configuration is Windows 10. The deployment section indicates it's found in build.

This screenshot shows the same bug work item from the previous screen, but with more detailed system information listed under "System Info". The system info table includes:

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	125

2116231801002

CS23432

## 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', '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 under 'Test Suites'. It contains two test points: 'TC05 – View Playlist Page' (selected) and 'TC06 – Playlist Loading Failure'. A modal window titled 'TC05 – View Playlist Page' is open, showing the 'Test Case Results' 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	Karthikayen 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

## 9. Test report summary

The screenshot shows the Azure DevOps interface for work items. The left sidebar includes 'Overview', 'Boards', 'Work items', 'Backlogs', 'Sprints', 'Queries', 'Delivery Plans', 'Analytics views', 'Repos', 'Pipelines', 'Test Plans', and 'Artifacts'. The 'Work items' section is selected. A specific bug report for 'BUG 203' is displayed, titled '203 - BG 01 - Countries Drop down Not Available on the page'. The details show:

- State:** New
- Reason:** New
- Repro Step:** Active
- Iteration:** News Feed App
- Step no. Result Title**
- 1. Passed Open Chrome Browser  
Expected Result: Browser should open in a new tab
- 2. Passed Paste the URL (<https://127.0.0.1/new.html>)  
Expected Result: App should load with new articles
- 3. Failed System should show a dropdown with list of countries  
Action: Produce Report

The right side of the screen displays sections for 'Planning', 'Deployment', 'Development', and 'Effort (Hours)'. There is also a 'Details' pane with various status indicators and links.

- Assigning bug to the developer and changing state

The screenshot shows a detailed view of a test plan in Azure DevOps. On the left, the navigation bar includes 'Test Plans' under 'Music Playlist Batch Creator'. The main area displays a test run titled 'Run 48 - TS02 - View Playlists (Manual) / TC06 – Playlist Loading Failure'. A specific test step, '92 TB01 - Playlist loading spinner keeps spinning indefinitely on poor network', is highlighted. This step has two steps: '1. Failed' (Disconnect from internet) and '2. Failed' (Navigate to "My Playlists"). Both steps show an 'Expected Result' of 'Network is offline' and a 'Comments' section indicating 'Page Not loading'. The 'Test Configuration' is set to 'Windows 10'. To the right, sections for 'Planning' (Priority 2, Severity 3 - Medium), 'Deployment' (status reporting), 'Development' (link to Azure Repos), and 'Related Work' (a linked work item) are visible. A 'System Info' sidebar on the far right shows 'Last 14 Days' activity.

## 10. Progress report

The screenshot shows the 'Progress report' section in Azure DevOps. The left sidebar highlights 'Progress report' under 'Test Plans'. The main area features a 'Summary' card with 2 test plans, 28 test points, and 28 (28/28) test points run, all at 100% completion. Below this is an 'Outcome trend' chart for the last 14 days, showing a sharp increase in green 'Passed' status from April 5th to April 10th, reaching a plateau of 28 passed tests by April 18th. A legend indicates that grey areas represent 'Not run' and green areas represent 'Passed'. At the bottom, a 'Details' table provides a breakdown of test plans, test points, run percentages, and failure counts.

Test plan name	Test points	Run %	Passed %	Failed %	Not run count
> Music Playlist Batch Creator Team_Stories_Integration	14	100	100	0	0
> Music Playlist Batch Creator – Test Plan	14	100	100	0	0

## 11. Changing the test template

The screenshot shows the 'All processes' list in the Azure DevOps Settings - Process section. The 'Agile' template is selected, indicated by a blue border around its row. The 'Basic' template is also visible.

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

The screenshot shows the 'All processes' list in the Azure DevOps Settings - Process section. The 'Agile' template is selected, indicated by a blue border around its row. The 'Basic' template is also visible.

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' view in the Azure DevOps Settings - Process section. The 'Agile' template is selected, specifically '231801095 Agile (default)'. The table lists various process templates with their descriptions and team project counts.

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. A new field named 'Acceptance Criteria' is being created, defined as a 'Text (single line)' type. The dialog also includes options for using an existing field or creating a new one, and a description field.

Add a field to Test Case

Definition: Add a field to store custom, queryable data about your work items.

Options: Use an existing field  
Create a field

Layout: Field: Acceptance Criteria

Create a field

Name: Type

Type: Text (single line)

Description: Optionally provide a description for the field

Add field Cancel

Microsoft Azure Portal | My Information | Azure DevOps - Microsoft | My Information | Test Plan 84 Music Playlist | Settings - Process (231801095) | Settings - Overview (231801095) | + | - | ○ | ×

dev.azure.com/231801095/\_settings/process?process-name=231801095%20Agile&\_a=projects

Azure DevOps 231801095 / Settings / Process

All processes > 231801095 Agile

Work-item types Backlog levels Projects

Name	Description
Music Playlist Batch Creator	Azure Music Playlist Batch Creator The Azure Music Playlist Batch Creator is a cloud-based solution designed for bulk playlist creation and management. Levera...

Organization Settings... 231801095

General

- Overview
- Projects
- Users
- Billing
- Global notifications
- Usage
- Extensions
- Microsoft Entra

Security

- Security overview
- Policies
- Permissions

Boards

- Process

Pipelines

- Agent pools
- Settings
- Deployment pools

Search Settings

Microsoft Azure Portal | My Information | Azure DevOps - Microsoft | My Information | Test Plan 84 Music Playlist | Settings - Process (231801095) | Settings - Overview (231801095) | + | - | ○ | ×

dev.azure.com/231801095/\_settings/process?type-id=231801095Agile.TestCase&process-name=231801095%20Agile&\_a=layout

Azure DevOps 231801095 / Settings / Process

All processes > 231801095 Agile > Test Case

Layout States Rules

New field New group New page Get extensions

Steps Summary Associated Aut...

Custom

Type Text (single line)

Recent test results

Recent test case results

Deployment

Development

Related Work

Status

Add a field ...

Organization Settings... 231801095

General

- Overview
- Projects
- Users
- Billing
- Global notifications
- Usage
- Extensions
- Microsoft Entra

Security

- Security overview
- Policies
- Permissions

Boards

- Process

Pipelines

- Agent pools
- Settings
- Deployment pools

Search Settings

**Result:**

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

<b>EXP NO: 9</b>	<b>LOAD TESTING AND PIPELINES</b>
------------------	-----------------------------------

**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.

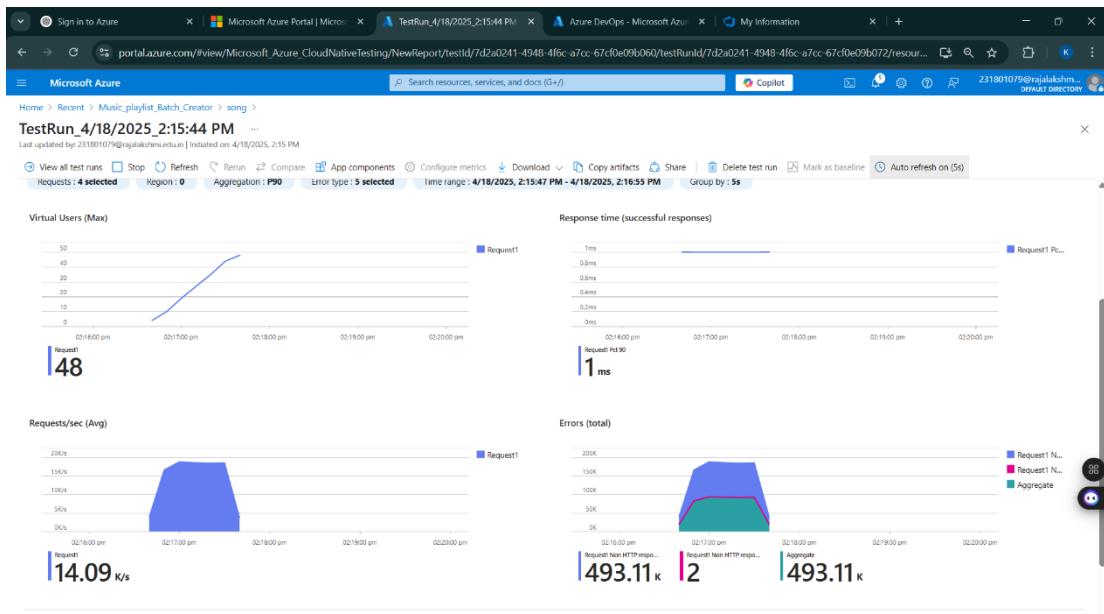
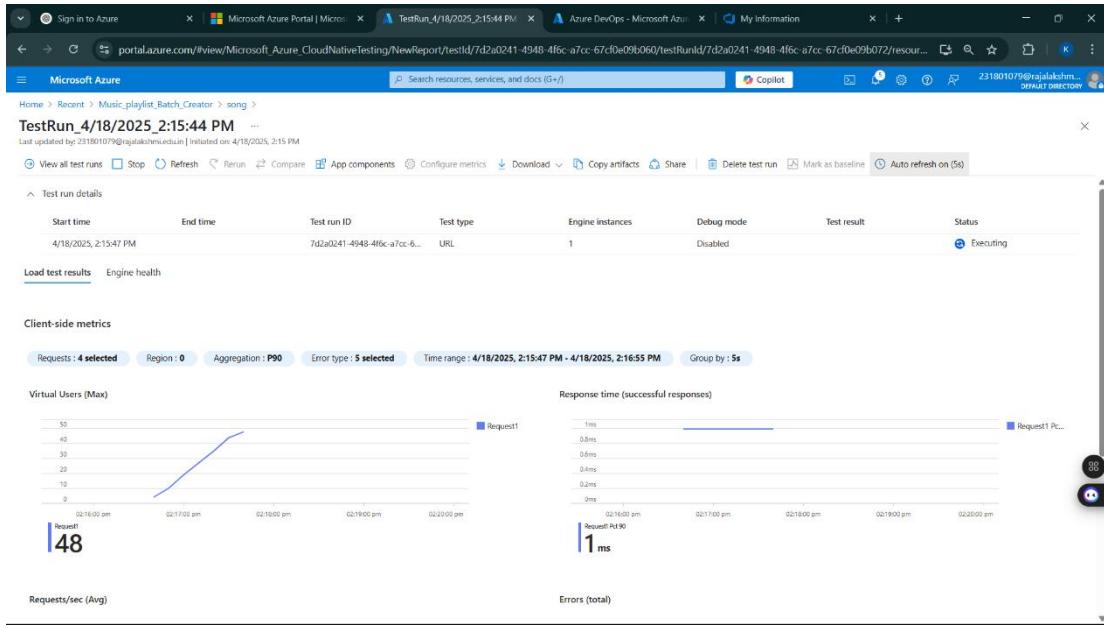
2. Create the Resource
  - Go to *Create a resource* → Search for “Azure Load Testing”.
  - Select Azure Load Testing and click Create.
3. Fill in the Configuration Details
  - *Subscription*: Choose your Azure subscription.
  - *Resource Group*: Create new or select an existing one.
  - *Name*: Provide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

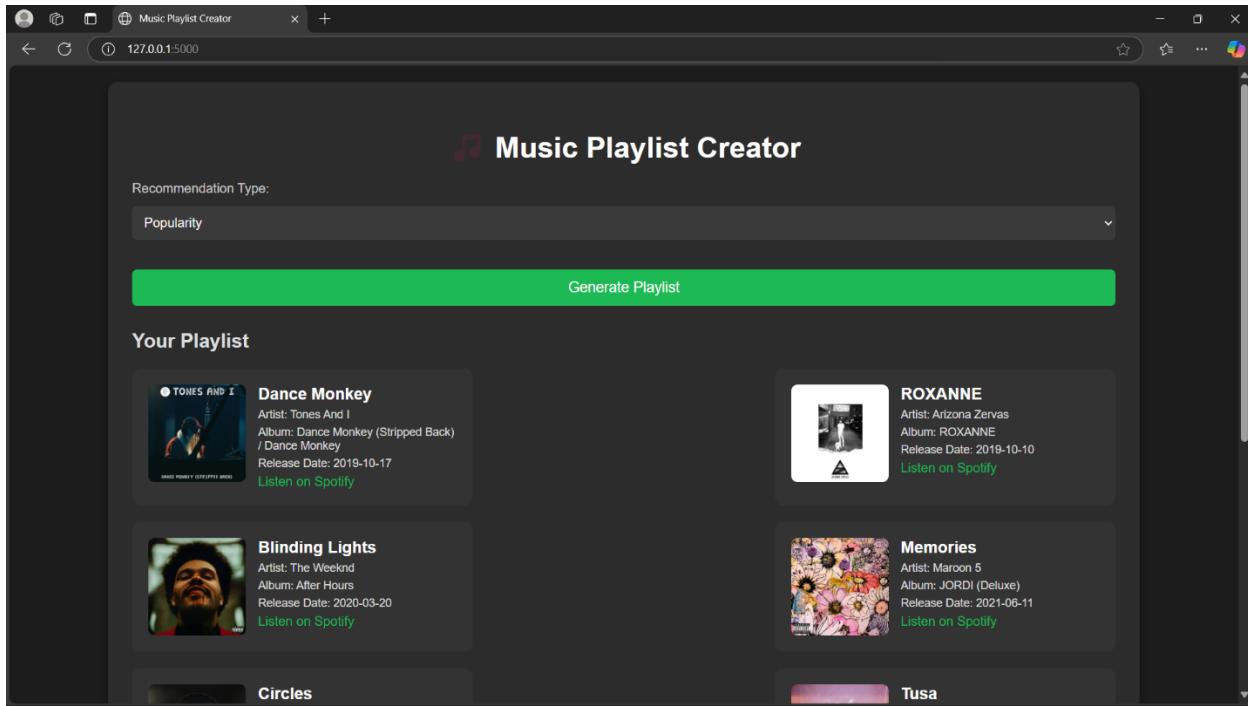
### **Steps to Create and Run a Load Test:**

Once your resource is ready:

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

### **Load Testing**





## 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' and is titled 'Pipeline 2'. The run was manually triggered by 'Karthick S'. The summary indicates that the run is retained as one of three recent runs by the main branch. The repository and version information shows 'Music Playlist Batch Creator' at commit 'main a87bd670'. The time started and elapsed is 'Just now' with a duration of '24s'. There are no related work items or artifacts. The 'Jobs' section lists a single job named 'Job' which has completed successfully in 6 seconds. The pipeline interface includes a sidebar with options like Overview, Boards, Repos, Pipelines, Environments, Releases, Library, Task groups, Deployment groups, Test Plans, and Artifacts. A search bar and a 'Run new' button are also visible.

## 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.

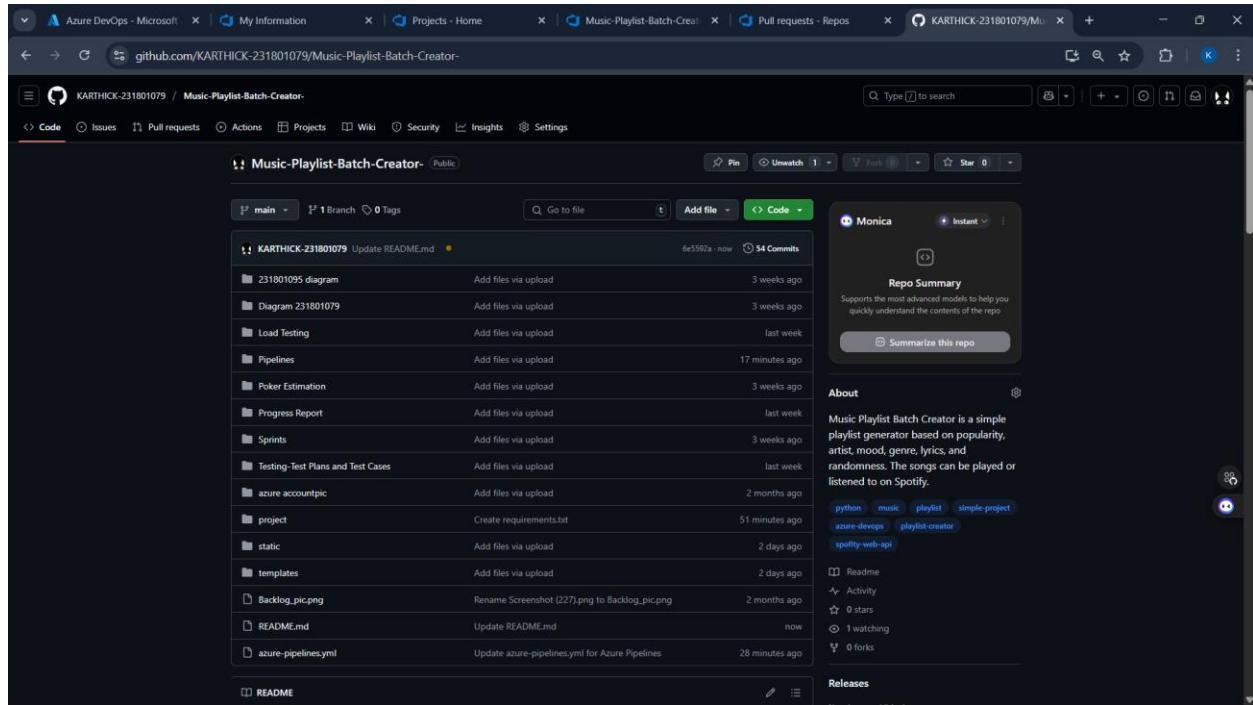
**EXP NO: 10**

## **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.