



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: Mallu Karthick Balaji Reddy

Register No: 231801095

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

Semester: IV

Academic Year: 2024 - 2025

INDEX

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

EXP NO: 1

AZURE DEVOPS ENVIRONMENT SETUP

Aim:

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

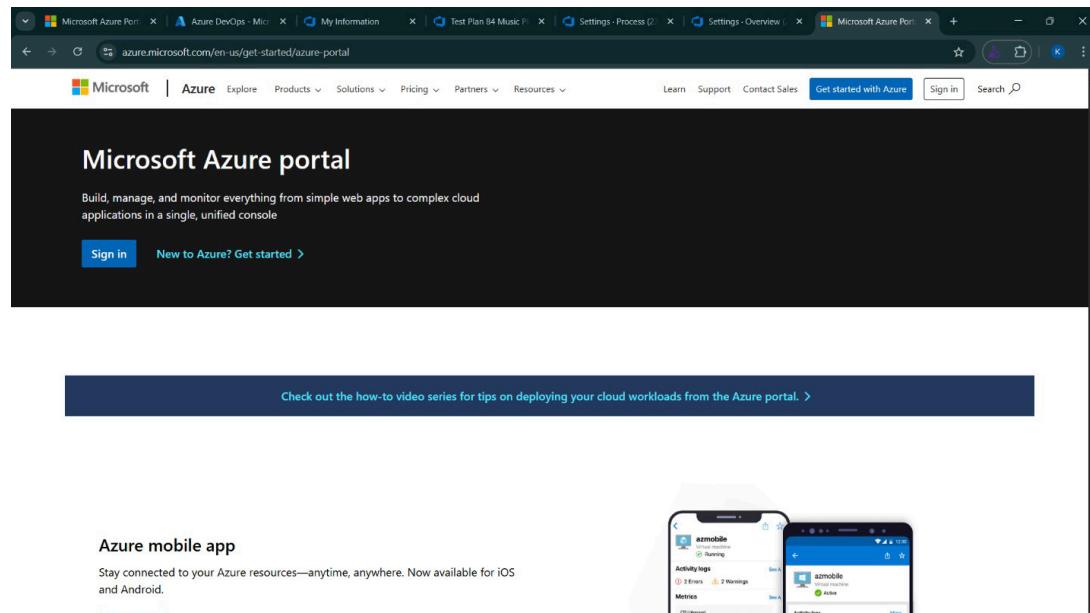
INSTALLATION

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

<https://azure.microsoft.com/en-us/get-started/azure-portal>.

Sign in using your Microsoft account credentials.

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



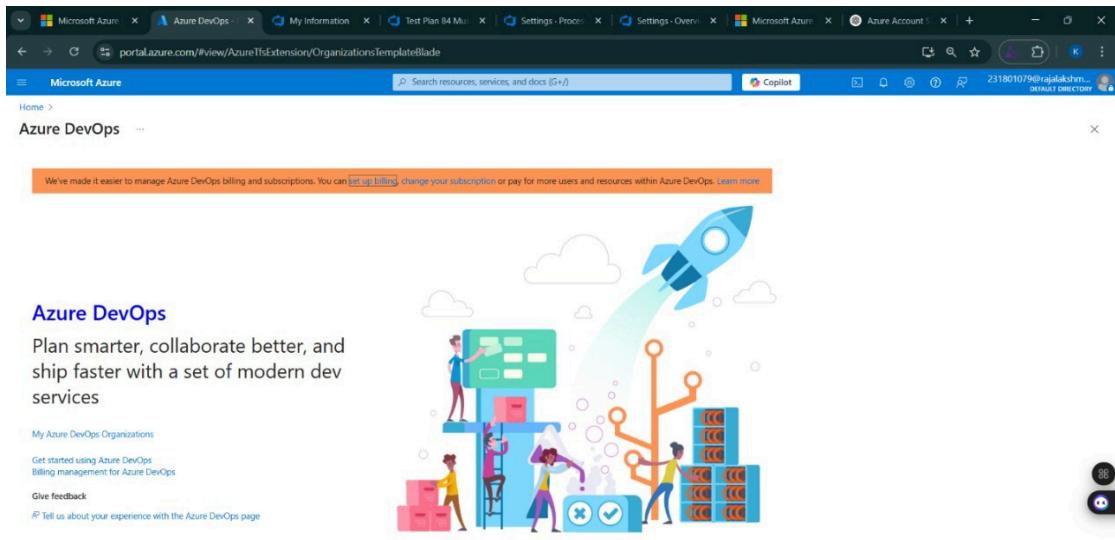
2. Azure home page

The screenshot shows the Microsoft Azure home page. At the top, there's a search bar with placeholder text "Search resources, services, and docs (G+Y)". Below the search bar is a navigation bar with links for "Copilot", "Azure Account", and a user profile. The main content area is titled "Azure services" and features a "Create a resource" button. It also includes icons for "Azure DevOps organizations", "Subscriptions", "Dashboard hub", "Resource groups", "Quickstart Center", "Azure AI services", "Kubernetes services", and a "More services" link. Below this is a "Resources" section with tabs for "Recent" and "Favorite". Under "Recent", there are two items: "Music" (Type: Azure Load Testing) and "Music_playlist_Batch_Creator" (Type: Resource group). A "See all" link is also present. Further down are sections for "Navigate" (with links to "Subscriptions", "Resource groups", "All resources", and "Dashboard"), "Tools" (with links to "Microsoft Learn", "Azure Monitor", "Microsoft Defender for Cloud", and "Cost Management"), and "Useful links" (with links to "Azure mobile app" and "https://portal.azure.com/#blade/Azure/BladeExtension/OrganizationsTemplateBlade").

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

This screenshot is similar to the previous one, showing the Microsoft Azure home page. However, the search bar at the top now contains the text "DevOps". The rest of the interface, including the "Azure services" dashboard, "Resources" section, "Tools" section, and "Useful links" section, remains the same as in the first screenshot.

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



Result:

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

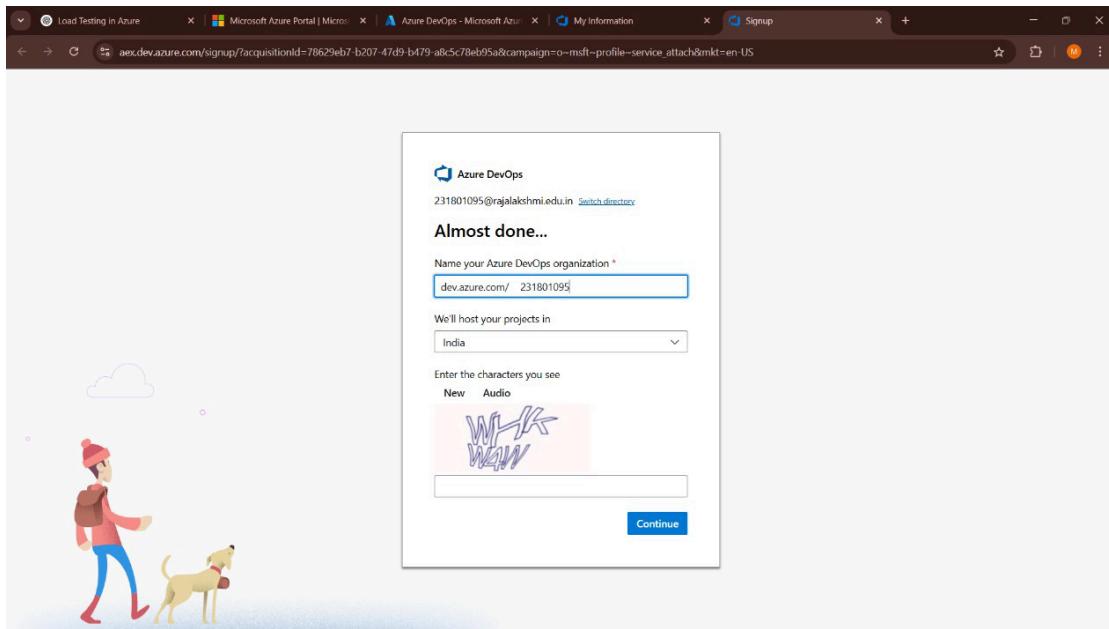
EXP NO: 2

AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

Aim:

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

1. Create An Azure Account



2. Create the First Project in Your Organization

a. After the organization is set up, you'll need to create your first **project**. This is where you'll begin to manage code, pipelines, work items, and more.

- 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. The 'Project name' field contains 'Music Playlist Batch Creator'. The 'Visibility' section has two options: 'Public' (selected) and 'Private'. The 'Public' option is described as allowing anyone on the internet to view the project, noting that certain features like TFVC are not supported. The 'Private' option is described as allowing only people given access to view the project. Below the visibility section, a note states: 'By creating this project, you agree to the Azure DevOps [code of conduct](#)'. The 'Advanced' section is expanded, showing 'Version control' set to 'Git' and 'Work item process' set to '231801095 Agile'. At the bottom right are 'Cancel' and 'Create' buttons.

3. Once logged in, ensure you are in the correct organization. If you're part of multiple organizations, you can switch between them from the top left corner (next to your user profile). Click on the Organization name, and you should be taken to the Azure DevOps Organization Home page.

4. Project dashboard

5. To manage user stories:

- From the left-hand navigation menu, click on **Boards**. This will take you to the main **Boards** page, where you can manage work items, backlogs, and sprints.
- On the **work items** page, you'll see the option to **Add a work item** at the top. Alternatively, you can find a + button or **Add New Work Item** depending on the view you're in. From the **Add a work item** dropdown, select **User Story**. This will open a form to enter details for the new User Story.

The screenshot shows the Azure DevOps interface for the 'Music Playlist Batch Creator Team' project. The left sidebar is the navigation menu with options like Work items, Boards, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The main area is titled 'Backlog' under 'Music Playlist Batch Creator Team'. It displays a table of work items with columns: Order, Work Item Type, Title, State, Effort, Story Points, Business Value, Value Area, and Tags. There are six items listed, all labeled as 'Epic' and 'New'. The first item is 'User Authentication & Profile Management'. A 'New Work Item' button is visible at the top right of the backlog table.

The screenshot shows the Microsoft Azure portal dashboard. At the top, there are icons for Copilot, a search bar, and a user profile for '231801095@rajalakshmi.edu.in'. Below the header, there are cards for 'Virtual machines' (with a blue icon) and 'Resource groups' (with a grey icon). On the right, the user's name '231801095@rajalakshmi...' is displayed, along with links to 'Sign out', 'My Microsoft account', and 'Switch directory'. The background shows a blurred view of other Azure services.

Resultt:

**Successfully created an Azure DevOps
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 'Music Playlist Batch Creator Team'. The backlog is organized into several levels of hierarchy:

- Level 1 (Epics):**
 - User Authentication & Profile Management
 - Develop a system that allows users to create and manage music playlists
 - Auto-Playlist Creation Based on user preference
 - As a user, I should be able to create playlists based on my location
 - Bulk Song Addition
 - Data Collection
- Level 2 (Features):**
 - Fetch and integrate music data from external sources for playlist enhancement
 - Enable users to edit, customize, and share their playlists with friends
 - Improve system performance and enhance user experience
 - Test Epic
- Level 3 (User Stories):**
 - As a user, I should be able to create playlists based on my location
 - As a user, I should be able to fetch and integrate music data from external sources for playlist enhancement
 - As a user, I should be able to enable users to edit, customize, and share their playlists with friends
 - As a user, I should be able to improve system performance and enhance user experience

1. Fill in Epics

The screenshot shows the 'EPIC 33' creation dialog in Azure DevOps. The epic details are as follows:

- Title:** Fetch and integrate music data from external sources for playlist enhancement
- Assignee:** Mallu karthick Balaji Reddy
- Comments:** 0
- Tags:** Add Tag
- State:** New
- Reason:** New
- Area:** Music Playlist Batch Creator
- Iteration:** Music Playlist Batch Creator\Integration

The dialog is divided into several sections:

- Description:** Click to add Description.
- Planning:** Priority: 2, Risk: 1, Effort: 1, Business Value: 1, Time Criticality: 1.
- 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:** Add link: Add a classification to this work item.

2. Fill in Features

This screenshot shows the 'Details' view for a Feature work item. The top navigation bar includes 'Save and Close', 'Follow', and 'Details' buttons. The main area is divided into several sections:

- Description:** A large text input field with placeholder text: "Add a comment. Use # to link a work item, @ to mention a person, or ! to link a pull request." Below it is a link to "switch to Markdown editor".
- Planning:** Fields for Priority (set to 3), Risk, Effort, Business Value, Time Criticality, Start Date (Select a date...), and Target Date (Select a date...).
- Deployment:** A note: "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:** A note: "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:** A section showing 3 items.

3. Fill in User Story Details

This screenshot shows the 'Details' view for a User Story work item. The top navigation bar includes 'Save and Close', 'Follow', and 'Details' buttons. The main area is divided into several sections:

- Description:** A large text input field with placeholder text: "Click to add Acceptance Criteria." Below it is a link to "switch to Markdown editor".
- Acceptance Criteria:** A note: "Click to add Acceptance Criteria."
- Planning:** Fields for Story Points, Priority (set to 3), and Risk.
- Classification:** Fields for Value area (set to Business).
- Deployment:** A note: "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:** A note: "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."

Result:

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

EXP NO: 4

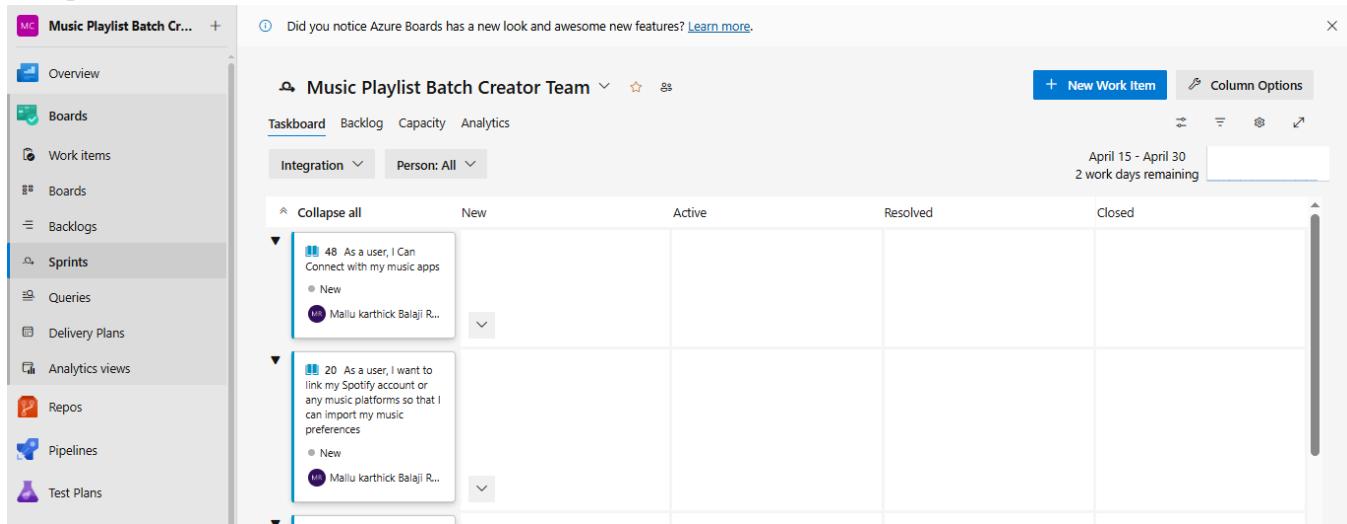
SPRINT PLANNING

Aim:

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

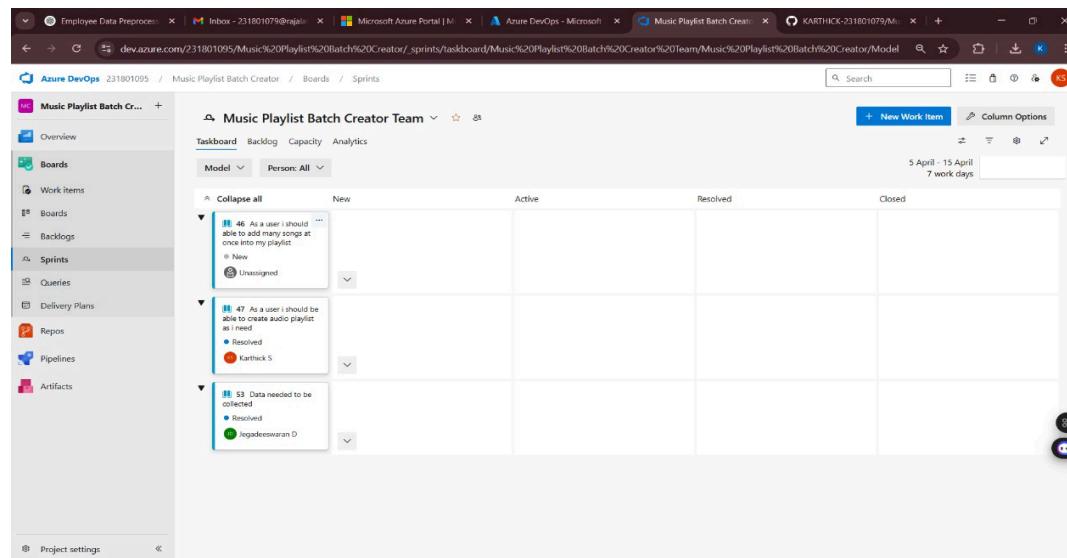
Sprint Planning

Sprint 1



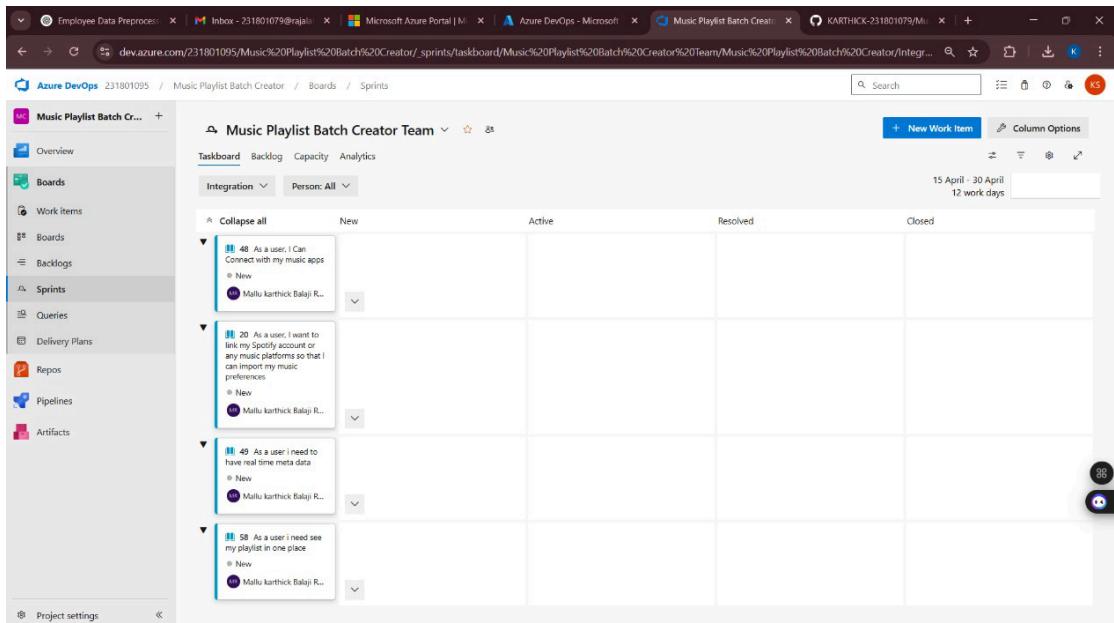
The screenshot shows the Azure Boards interface for the 'Music Playlist Batch Creator Team'. The left sidebar navigation includes 'Overview', 'Boards', 'Work items', 'Backlogs', 'Sprints', 'Queries', 'Delivery Plans', 'Analytics views', 'Repos', 'Pipelines', and 'Test Plans'. The main area displays a 'Taskboard' for the 'Integration' column under the 'Person: All' filter. Two user stories are visible: #48 'As a user, I Can Connect with my music apps' and #20 'As a user, I want to link my Spotify account or any music platforms so that I can import my music preferences'. Both stories are marked as 'New' and are assigned to 'Mallu karthick Balaji R..'. The top right corner shows the sprint duration as 'April 15 - April 30' with '2 work days remaining'.

Sprint 2



The screenshot shows the Azure Boards interface for the 'Music Playlist Batch Creator Team'. The left sidebar navigation includes 'Overview', 'Boards', 'Work items', 'Backlogs', 'Sprints', 'Queries', 'Delivery Plans', 'Repos', 'Pipelines', and 'Artifacts'. The main area displays a 'Taskboard' for the 'Model' column under the 'Person: All' filter. Three user stories are visible: #48 'As a user i should ... able to add many songs at once into my playlist' (status: New, assigned to 'Unassigned'), #47 'As a user i should be able to create audio playlist as i need' (status: Resolved, assigned to 'Karthick S'), and #33 'Data needed to be collected' (status: Resolved, assigned to 'legadeeswaran D'). The top right corner shows the sprint duration as '5 April - 15 April' with '7 work days' remaining.

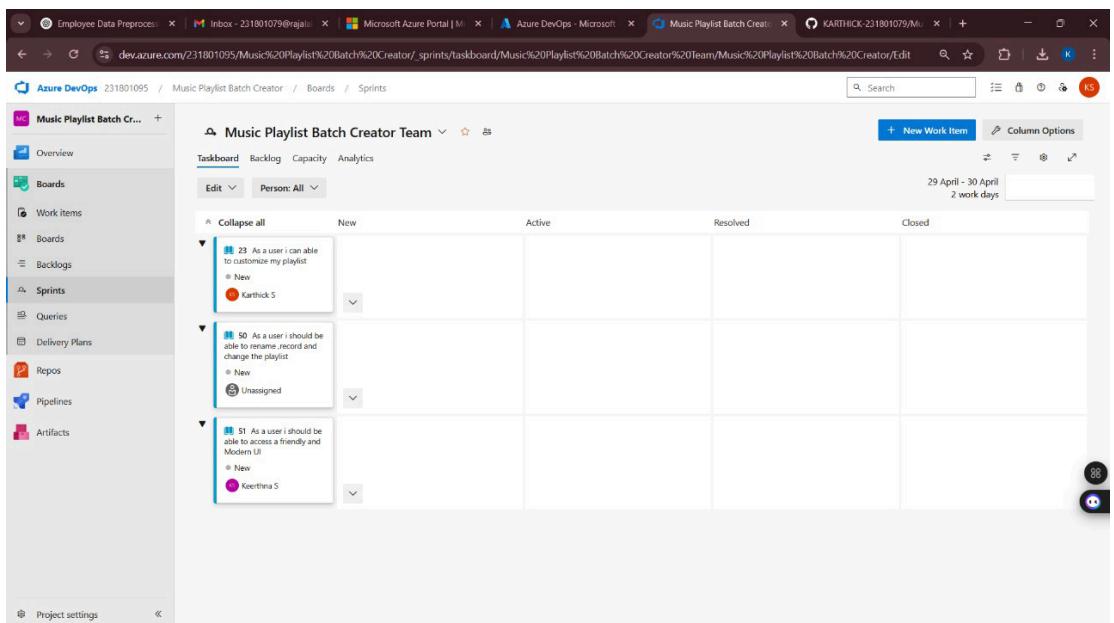
Sprint 3



A screenshot of the Azure DevOps Boards interface for the 'Music Playlist Batch Creator' project. The left sidebar shows navigation options like Overview, Boards, Work items, and Sprints. The main area displays a backlog board for the 'Music Playlist Batch Creator Team'. The backlog is organized into columns: New, Active, Resolved, and Closed. There are four work items visible in the 'New' column:

- #48 As a user, I can Connect with my music apps
Status: New
Assigned to: Malu 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
Status: New
Assigned to: Malu karthick Balaji R...
- #49 As a user I need to have real time meta data
Status: New
Assigned to: Malu karthick Balaji R...
- #58 As a user i need see my playlist in one place
Status: New
Assigned to: Malu karthick Balaji R...

Sprint 4



A screenshot of the Azure DevOps Boards interface for the 'Music Playlist Batch Creator' project, showing the state after Sprint 4. The left sidebar remains the same. The backlog board now shows three work items in the 'New' column:

- #23 As a user i can able to customize my playlist
Status: New
Assigned to: Karthick S
- #50 As a user i should be able to rename record and change the playlist
Status: New
Assigned to: Unsigned
- #51 As a user i should be able to access a friendly and Modern UI
Status: New
Assigned to: Keerthna 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 story ID 50. The title of the story is "As a user i should be able to rename ,record and change the playlist". The story has 0 comments and no tags. It is set to "New" status in the "State" field and is assigned to the "Music Playlist Batch Creator" area. The reason for the story is "New" and it is part of the "Music Playlist Batch Creator>Edit" iteration. The story has 5 Story Points, priority 2, and risk level 2 - Medium. The classification is "Value area" and "Business". There is a note to track related items via deployment pipeline. The discussion section is empty, and there are links to add a link and related work. The development section is also empty.

Result:

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

EXP NO: 6

DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE

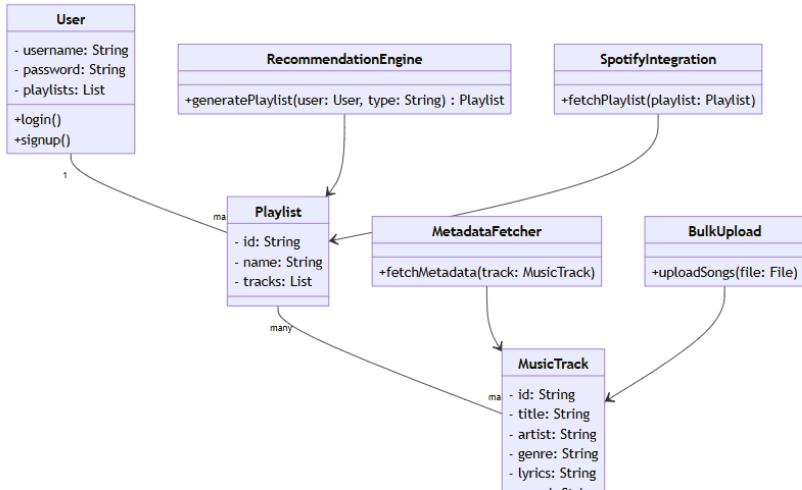
Aim:

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

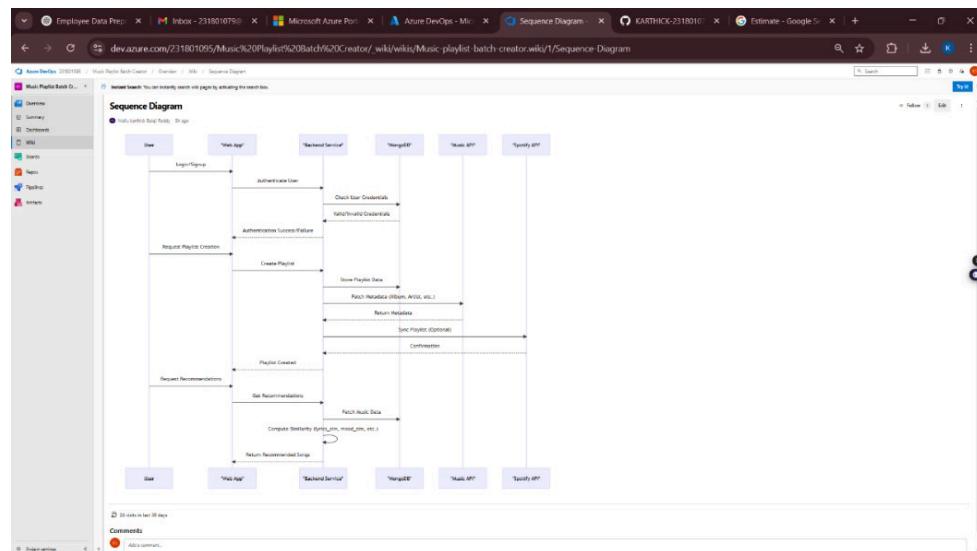
6A. Class Diagram

231801095 Class Diagram

Mallu karthick Balaji Reddy Apr 1



6B. Sequence Diagram



Result:

Creator.

The Class Diagram and Sequence Diagram is designed
Successfully for the Music Playlist Batch

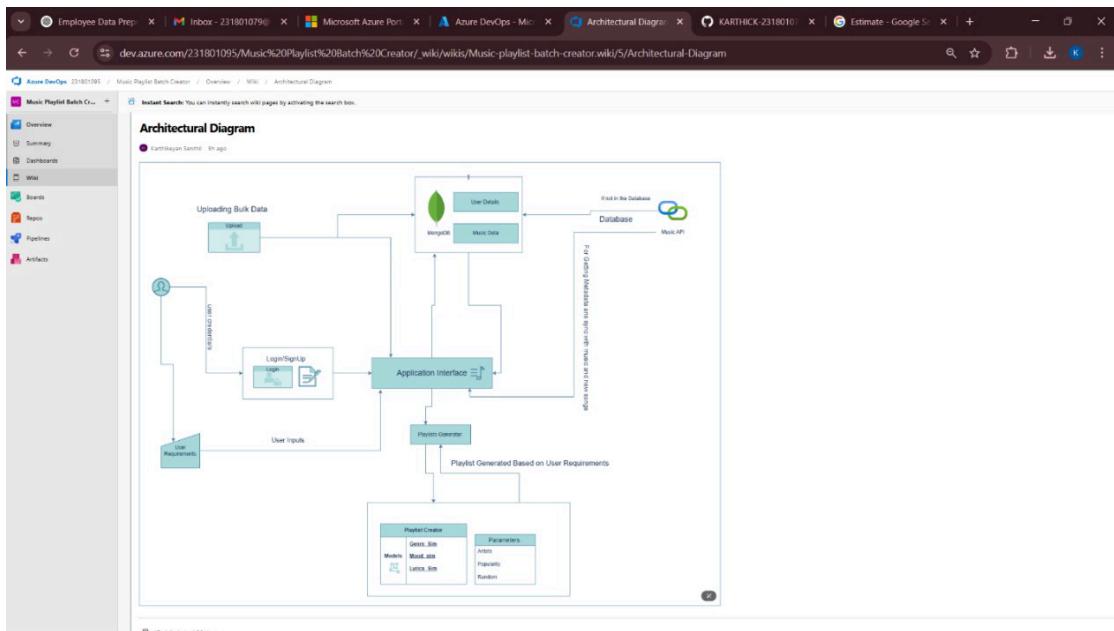
EXP NO: 7

DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE

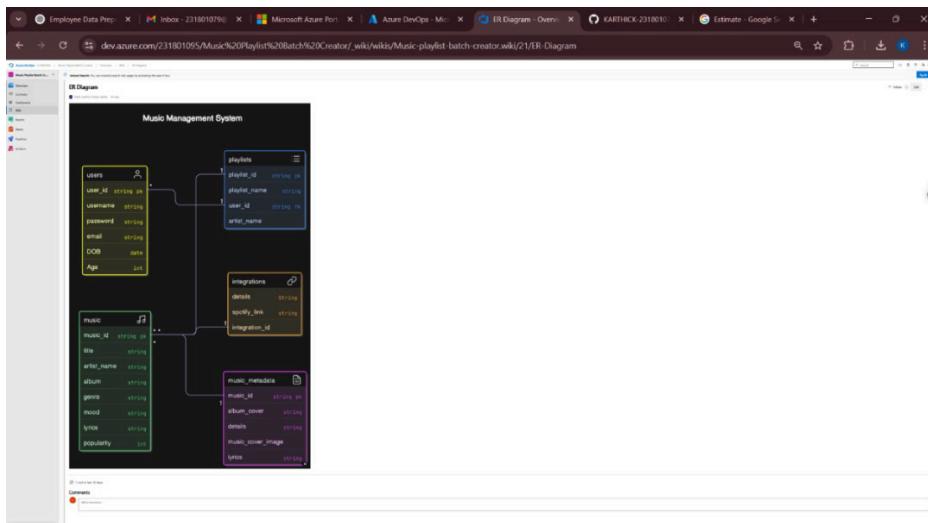
Aim:

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

7A. Architectural Diagram



7B.ER Diagram



Result:

Creator

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

EXP NO: 8	TESTING – TEST PLANS AND TEST CASES
------------------	--------------------------------------------

Aim:

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

Test Planning and Test Case

Test Case Design Procedure

1. Understand Core Features of the Application

- User Signup & Login
- Viewing and Managing Playlists
- Fetching Real-time Metadata
- Editing playlists (rename, reorder, record)
- Creating smart audio playlists based on categories (mood, genre, artist, etc.)

2. Define User Interactions

- Each test case simulates a real user behaviour (e.g., logging in, renaming a playlist, adding a song).

3. Design Happy Path Test Cases

- Focused on validating that all features function as expected under normal conditions.
- Example: User logs in successfully, adds item to playlist, or creates a category-based playlist.

4. Design Error Path Test Cases

- Simulate negative or unexpected scenarios to test robustness and error handling.
- Example: Login fails with invalid credentials, save fails when offline, no recommendations found.

5. Break Down Steps and Expected Results

- Each test case contains step-by-step actions and a corresponding expected outcome.
- Ensures clarity for both testers and automation scripts.

6. Use Clear Naming and IDs

- Test cases are named clearly (e.g., TC01 – Successful Login, TC10 – Save Playlist Fails).
- Helps in quick identification and linking to user stories or features.

7. Separate Test Suites

- Grouped test cases based on functionality (e.g., Login, Playlist Editing, Recommendation System).

- Improves organization and test execution flow in Azure DevOps.

8. Prioritize and Review

- Critical user actions are marked high-priority.
- Reviewed for completeness and traceability against feature requirements.

1. New test plan

New Test Plan

Name: Music Playlist Batch Creator - Test Plan

Area Path: Music Playlist Batch Creator

Iteration: Music Playlist Batch Creator\Integration

Create Cancel

2. Test suite

Title	Order	Test Case Id	Assigned To	State
TCD1 - Successful Sign Up	1	78	Karthikeyan Se...	Design
Static suite	2	80	Karthikeyan Se...	Design
Requirement based suite	3	81	Karthikeyan Se...	Design
Query based suite	4	82	Karthikeyan Se...	Design

3. Test case

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

Music Playlist Batch Creator – Test Plans

USER STORIES

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

Test Suites

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

1. TC01 – Successful Sign Up

○ **Action:**

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

○ **Expected Results:**

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

○ **Type:** Happy Path

2. TC02 – Secure Login

○ **Action:**

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

○ **Expected Results:**

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

○ **Type:** Happy Path

3. TC03 – Sign Up with Existing Email

- **Action:**
 - Go to the Sign-Up page.
 - Enter a name and an already registered email.
 - Click on "Sign Up".
- **Expected Results:**

- Fields accept data.
- Error message "Email already registered" is displayed.
- **Type:** Error Path

4. TC04 – Login with Wrong Password

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

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

1. TC05 – View Playlist Page

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

2. TC06 – Playlist Loading Failure

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

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

1. TC07 – Real-Time Metadata Display

- **Action:**

- Play a song.
 - Observe the metadata panel.
- **Expected Results:**
 - Metadata (title, artist, album, duration) is displayed and updates in real time.
 - **Type:** Happy Path
2. TC08 – Metadata Not Updating
 - **Action:**

- Play a different song.
- Observe the metadata panel.
- **Expected Results:**
 - Metadata remains static or shows default/fallback message.
- **Type:** Error Path

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

1. TC09 – Rename Playlist Successfully

- **Action:**
 - Navigate to "My Playlists".
 - Click "Rename" next to a playlist.
 - Enter a new name and click "Save".
- **Expected Results:**
 - Playlist name updates successfully.
- **Type:** Happy Path

2. TC10 – Rename with Blank Name

- **Action:**
 - Click "Rename" on a playlist.
 - Leave the field blank.
 - Click "Save".
- **Expected Results:**
 - Error message "Playlist name cannot be empty" is shown.
- **Type:** Error Path

3. TC11 – Change Playlist Order

- **Action:**
 - Open a playlist.
 - Drag and drop songs to reorder.
 - Click "Save".
- **Expected Results:**
 - Playlist order is updated and saved.
- **Type:** Happy Path

4. TC12 – Change Playlist Order Fails

- **Action:**
 - Login and go to "My Playlists".

- Select a playlist.
 - Go offline or simulate server error.
 - Reorder songs and click “Save Order”.
- **Expected Results:**
 - Error message: "Failed to update order. Please check your connection".
 - **Type:** Error Path

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

1. TC13 – Generate Playlist Based on Various Categories

- Action:

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

- Expected Results:

- Playlist is generated based on selected mood and categories.

- Type: Happy Path

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

- Action:

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

- Expected Results:

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

- Type: Error Path

Test Cases

The screenshot shows the Azure DevOps Test Plan interface. A test case titled "77 TC06 – Playlist Loading Failure" is displayed. The test case details include:

- Start: Staged
- Area: Music Playlist Batch Creator
- Iteration: Music Playlist Batch Creator/Integration
- Reason: New
- Owner: Karthick S
- Comments: 0
- Tags: Add Tag

The "Steps" tab shows the following steps:

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

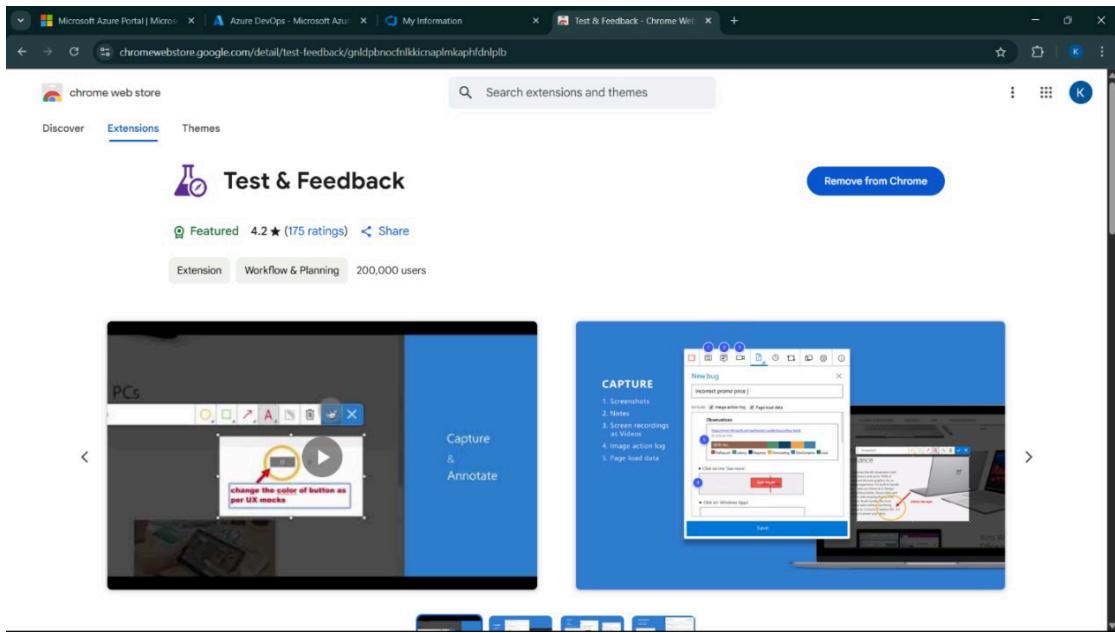
The "Status" section indicates:

- Type: Error Path
- Priority: 2
- Automation status: Not Automated

A screenshot of the Azure DevOps Test Plan interface. The left sidebar shows navigation links like Overview, Board, Repositories, Pipelines, Test Plans, Programs, Parameters, Configuration, Runs, and Artifacts. The main area displays a test case titled "TC05 - View Playlist Page" for the "Music Playlist Batch Creator" project. The test case is assigned to "Karthick S" and has a status of "Design". It includes a "Steps" section with two steps: "Log in successfully" (Expected result: User is redirected to dashboard) and "Navigate to 'My Playlists' section" (Expected result: All created playlists are displayed clearly). A note says "Click or type here to add a step". The "Status" section shows a priority of 2 and an automation status of "Not Automated". A "Custom" tab is also visible.

4. Installation of test

A screenshot of the Chrome Web Store page for the "Test & Feedback" extension. The extension is listed as "Featured" with a rating of 4.2 stars and 175 ratings, and it has 200,000 users. The "Add to Chrome" button is prominently displayed. The page shows a preview of the extension's features, including "Capture & Annotate" (with a screenshot of a video player interface) and "Newbug" (with a screenshot of a bug reporting interface showing a "Newbug" dialog and a recording of a user interaction).



Test and feedback Showing it as an extension

5. Running the test cases

The screenshot shows the Azure DevOps Test Plan interface. On the left, the navigation bar includes 'Overview', 'Boards', 'Repos', 'Pipelines', 'Test Plans', 'Test plans' (selected), 'Progress report', 'Parameters', 'Configurations', 'Runs', and 'Artifacts'. Under 'Test plans', 'Music Playlist Batch Creator - T...' is expanded, showing 'TS01 - User Login (4)', 'TS02 - View Playlists (2)', 'TS03 - Real-Time Met...', 'TS04 - Playlist Editing (4)', and 'TS05 - Smart Playlist ...'. The 'TS02 - View Playlists (ID: 87)' page is displayed, with the 'Execute' tab selected. A table lists 'Test Points (2 items)':

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...

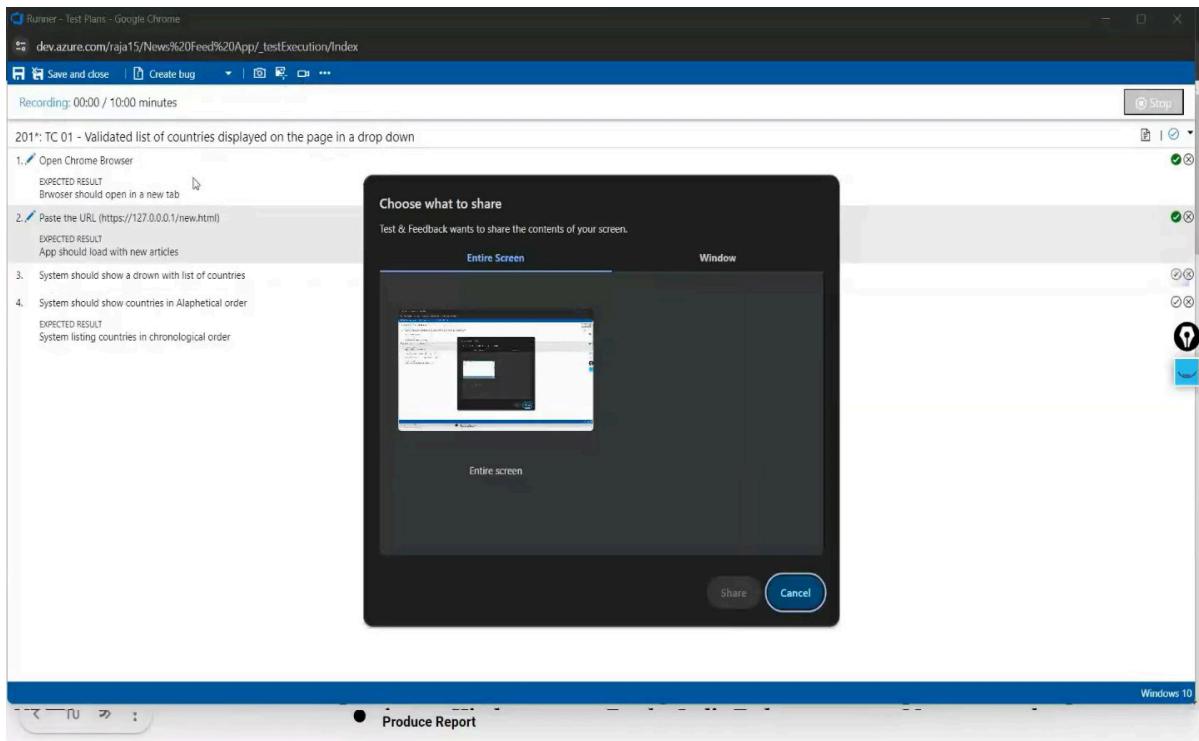
 A context menu is open over the first row, with options: 'Run for web application' (selected), 'Run for desktop application', and 'Run with options'.

The screenshot shows a browser window titled 'Runner - Test Plans - Google Chrome' displaying the results for 'TC05 – View Playlist Page'. The steps listed are:

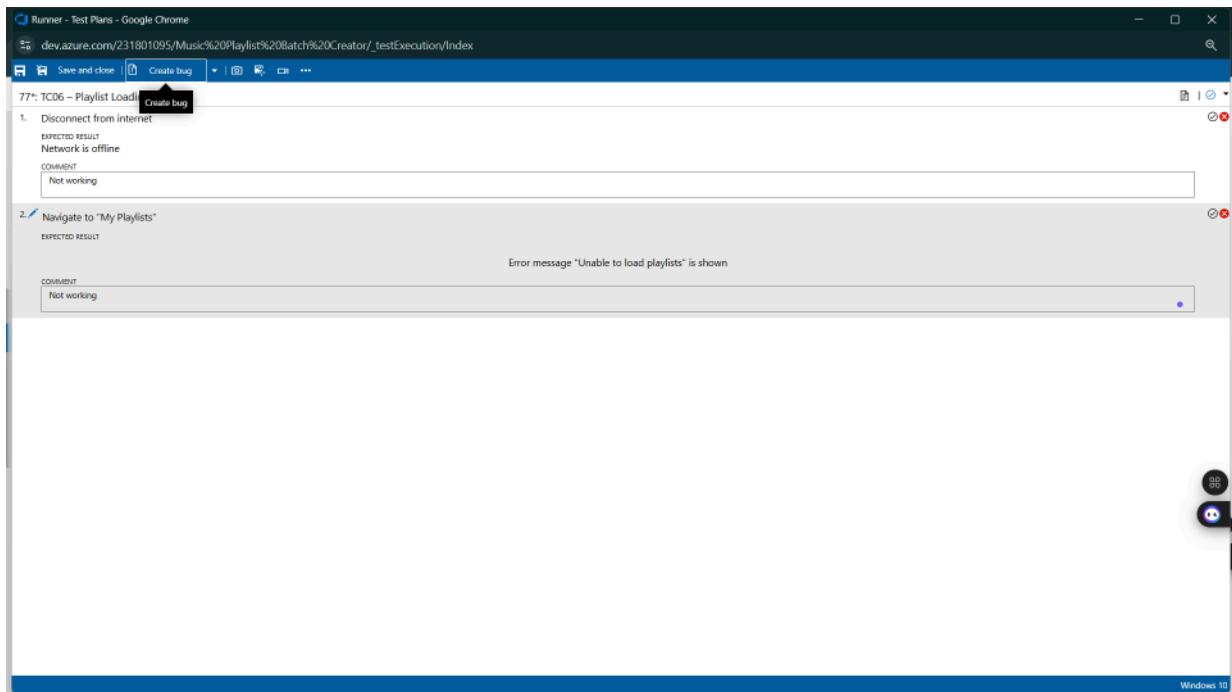
1. Log in successfully
EXPECTED RESULT
User is redirected to dashboard
2. Navigate to "My Playlists" section
EXPECTED RESULT
All created playlists are displayed clearly

 The status for step 1 is 'PASSED' and for step 2 is 'PASSED'.

6. Recording the test case



7. Creating the bug



The screenshot shows a bug detail page in Azure DevOps. The title of the bug is "TB01 - Playlist loading spinner keeps spinning indefinitely on poor network". The bug was filed on "18-04-2025 03:23" and is categorized under "Music Playlist Batch Creator". The "Repro Steps" section details two steps: 1. Disconnect from internet (Expected Result: Network is offline) and 2. Navigate to "My Playlists" (Expected Result: Error message "Unable to load playlists" is shown). The "Test Configuration" is listed as Windows 10. The "Planning" section shows the bug is Unassigned, has 0 comments, and an "Add tag" button. The "Deployment" section includes links to Releases and Boards. The "Development" section has an "Add link" button and a note about linking to Azure Repos. The "Effort (Hours)" section shows Original Estimate and Remaining values. The "Related Work" section allows linking to existing work items. The "System Info" section lists system configuration details like Browser Name, Language, and Processor model. The "Discussion" section is empty.

This screenshot shows the same bug detail page as the previous one, but with more detailed system information visible. The "System Info" table includes additional rows such as Operating system - Name (Windows NT 10.0), Operating system - Architecture (x86_64), Operating system - Processor model (11th Gen Intel(R) Core(TM) i3-1115G4 @ 3.00GHz), and Display - Device pixel ratio (1.25). The "Discussion" section contains a placeholder message: "Add a comment. Use # to link a work item, ! to link a pull request, or @ to mention a person." The "Project settings" section at the bottom shows a file named "SystemInformation-2025-04-18T03-23-58.169Z.json" with a size of 1K.

8. Test case results

The screenshot shows the Azure DevOps Test Plan interface. On the left, the navigation bar includes 'Overview', 'Boards', 'Repos', 'Pipelines', 'Test Plans', 'Test plans' (selected), 'Progress report', 'Parameters', 'Configurations', 'Runs', and 'Artifacts'. Under 'Test plans', 'Music Playlist Batch Creator...' is selected, showing 'Test Suites' (TS01 - User Login 4, TS02 - View Playlists 2, TS03 - Real-Time Met... 3, TS04 - Playlist Editing 4, TS05 - Smart Playlist ... 1). The main area displays 'TS02 - View Playlists (ID: 87)' with 'Test Points (2 items)'. One test point, 'TC05 - View Playlist Page', is checked and highlighted in blue. 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	

9. Test report summary

The screenshot shows the Azure DevOps Work Items interface. On the left, the navigation bar includes 'Overview', 'Boards', 'Work items' (selected), 'Backlogs', 'Sprints', 'Queries', 'Delivery Plans', 'Analytics views', 'Repos', 'Pipelines', 'Test Plans', and 'Artifacts'. The main area displays a work item for 'BUG 203: BG 01 - Countries Drop down Not Available on the page' assigned to 'rajeshr prabhu'. The work item details include:

- State:** New (highlighted)
- Reason:** Active
- Repro Step:** Active, Resolved, Closed (with note: "Validated list of countries displayed on the page")
- 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 Section should show a dropdown with list of countries
Produce Report
- Planning:** Resolved Reason, Story Points, Priority (2), Severity (3 - Medium), Activity
- Deployment:** To track releases associated with this work item, go to [Releases](#) and turn on deployment status reporting for Boards in your pipeline's Options menu. [Learn more about deployment stat 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.
- Effort (Hours):** Original Estimate
- Related Work:** (partial view)

- Assigning bug to the developer and changing state

Bug Detail:

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

Step no. Result Title

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

Test Configuration: Windows 10

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

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

Development:
Link an Azure Repos commit, pull request or branch to see the status of your development. You can also create a branch to get started.

Related Work:
Add link: Add an existing work item as a parent
Tested By: 77 TC06 - Playlist Loading Failure
Updated 10-04-2025. ◉ Design

System Info:

10. Progress report

Progress report

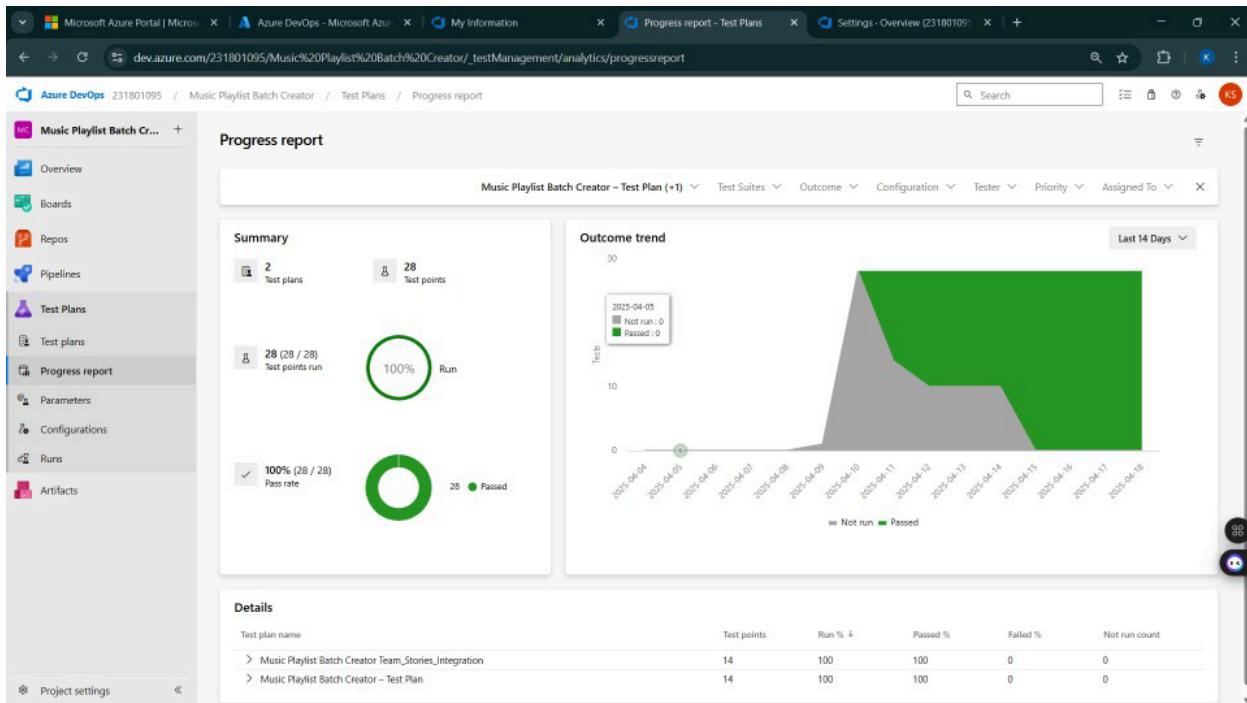
Summary:

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

Outcome trend: Last 14 Days

Details:

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

All processes

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

Process

- General
- Overview
- Projects
- Users
- Billing
- Global notifications
- Usage
- Extensions
- Microsoft Entra
- Security
- Security overview
- Policies
- Permissions
- Boards
- Pipelines
- Agent pools
- Settings
- Deployment pools

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

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

The screenshot shows the 'All processes' list in the Azure DevOps Settings - Process section. The '231801095 Agile (default)' template is selected, indicated by a highlighted row. The list includes:

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

12. View the new test case template

The screenshot shows the 'Add a field to Test Case' dialog box over a background of the Azure DevOps settings interface. The dialog has a 'Definition' tab selected, showing options to 'Use an existing field' (with 'Acceptance Criteria' selected) or 'Create a field'. A 'Create a field' section is filled out with 'Name' set to 'Type' and 'Type' set to 'Text (single line)'. There is also a 'Description' field with placeholder text. At the bottom are 'Add field' and 'Cancel' buttons.

The screenshot shows the 'Work item types' page under the 'Process' tab. It lists a single work item type named 'Music Playlist Batch Creator' with a detailed description: 'Azure Music Playlist Batch Creator The Azure Music Playlist Batch Creator is a cloud-based solution designed for bulk playlist creation and management. Levera...'. Other tabs visible include 'Backlog levels' and 'Projects'.

The screenshot shows the Azure DevOps Settings - Process page for a specific organization setting (231801095). The left sidebar is titled 'Organization Settings' and lists various categories like General, Security, Boards, Pipelines, and Process. The 'Process' category is selected. The main content area is titled 'All processes > 231801095 Agile > Test Case'. It displays a configuration interface with tabs for 'Steps', 'Summary', and 'Associated Aut...'. A 'Custom' section is visible on the right, containing fields for 'Recent test results', 'Deployment', 'Development', 'Related Work', and 'Status'. The 'Status' section includes fields for 'Priority' (an integer) and 'Automation status'.

Result:

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

EXP NO: 9	LOAD TESTING AND 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:

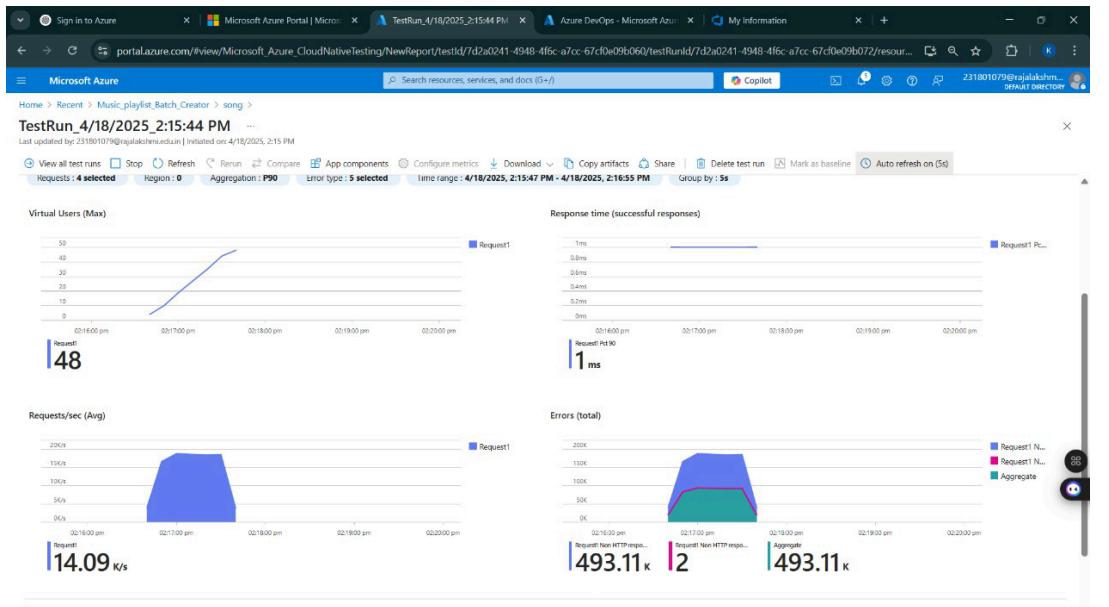
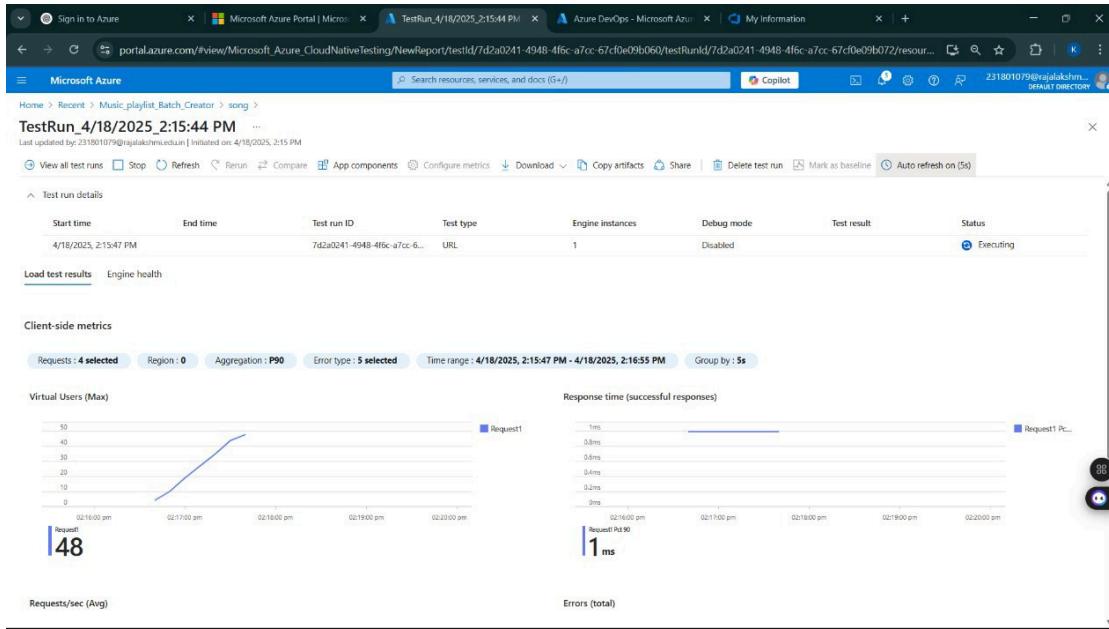
1. Sign in to Azure Portal
Go to <https://portal.azure.com> and log in.
2. Create the Resource
 - o Go to *Create a resource* → Search for “Azure Load Testing”.
 - o Select Azure Load Testing and click Create.
3. Fill in the Configuration Details
 - o *Subscription*: Choose your Azure subscription.
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

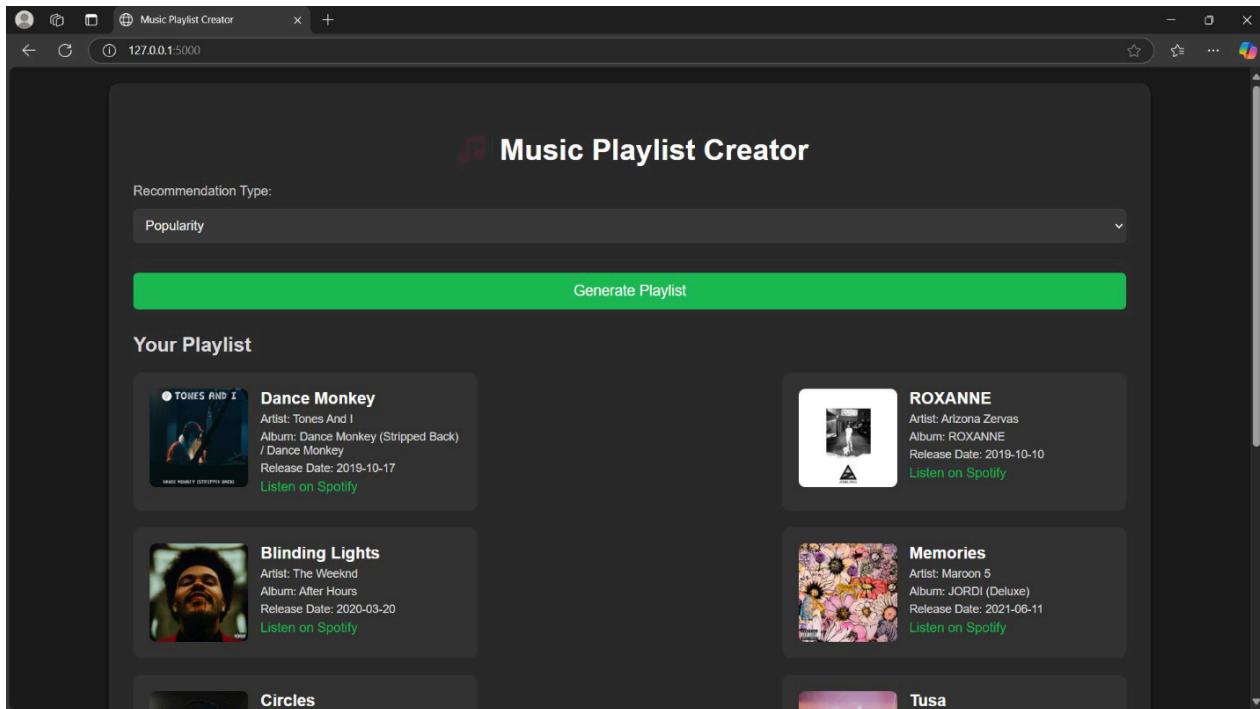
Steps to Create and Run a Load Test:

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"
```

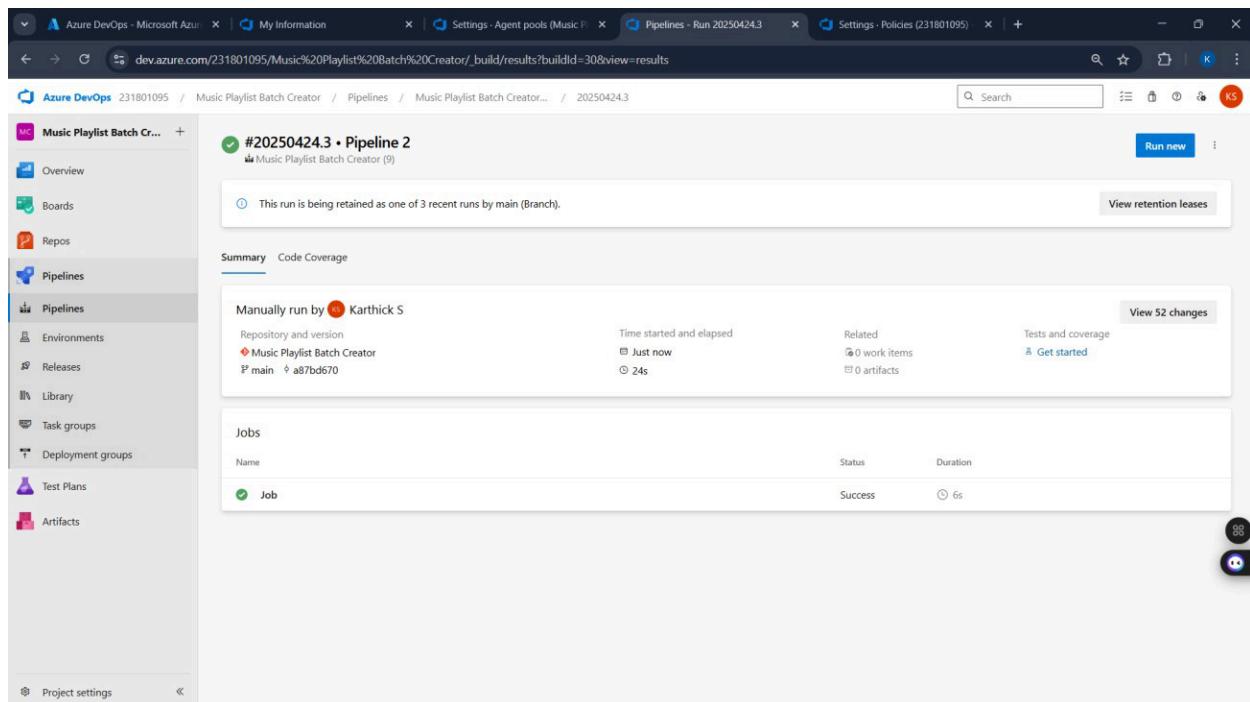
3. Pipeline Tasks Include:

- 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 interface for a pipeline run. The left sidebar is titled 'Music Playlist Batch Creator' and includes sections for Overview, Boards, Repos, Pipelines, Environments, Releases, Library, Task groups, Deployment groups, Test Plans, and Artifacts. The 'Pipelines' section is currently selected. The main content area displays a summary of a recent pipeline run:

- Run ID:** #20250424.3 • Pipeline 2
- Trigger:** Music Playlist Batch Creator (9)
- Status:** Retained (as one of 3 recent runs by main (Branch))
- Manually run by:** Karthick S
- Repository and version:** Music Playlist Batch Creator, main (a87bd670)
- Time started and elapsed:** Just now, 24s
- Related:** 0 work items, 0 artifacts
- Tests and coverage:** View 52 changes, Get started
- Jobs:** A single job named 'Job' is listed as successful (Duration: 6s).

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.

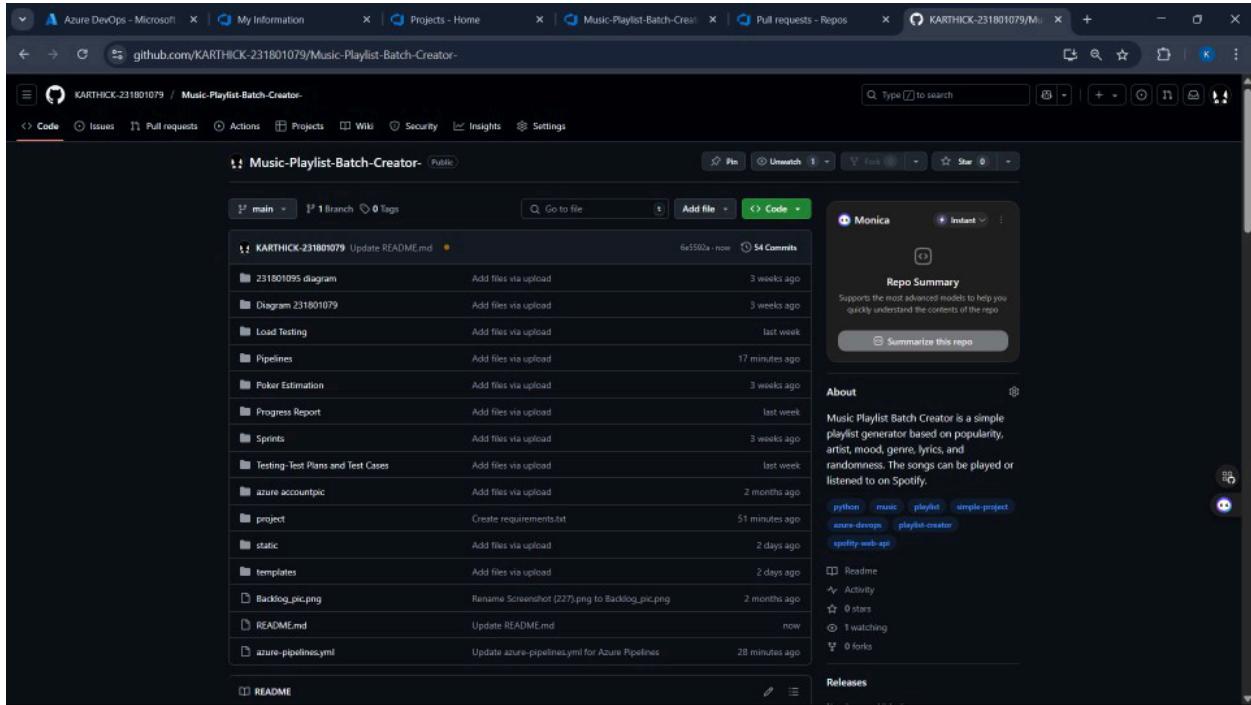
EXP NO: 10

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.