

EXP NO: 1

AZURE DEVOPS ENVIRONMENT SETUP

Aim:

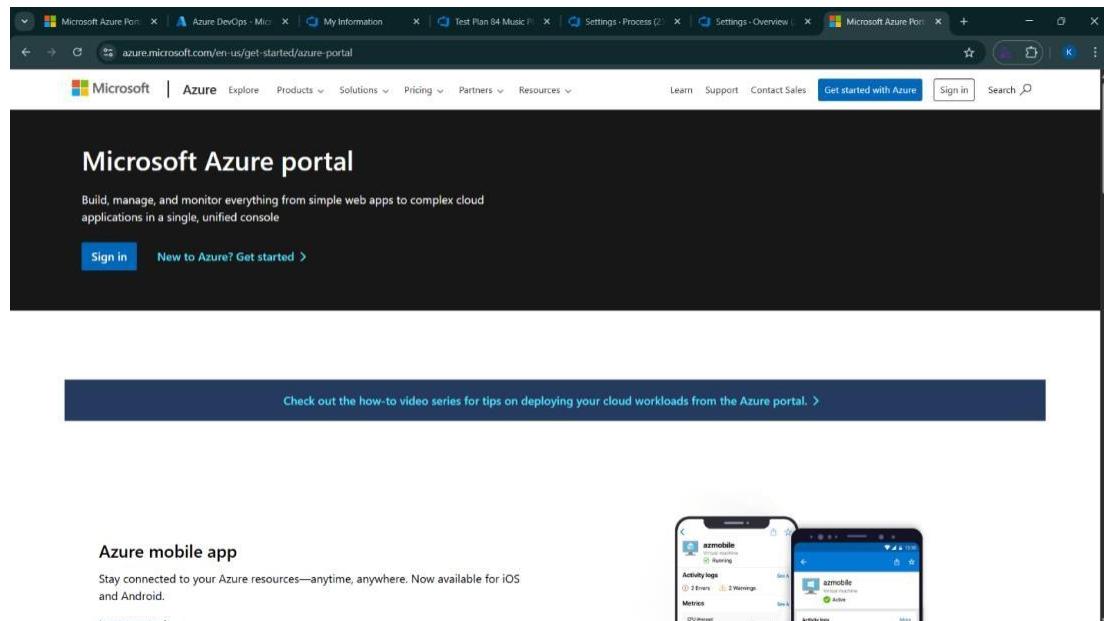
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

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+)" and a user profile icon. Below the search bar, the header "Welcome to Azure!" is displayed, followed by a note: "Don't have a subscription? Check out the following options." Three promotional cards are shown: "Start with an Azure free trial" (with a "Start" button), "Manage Microsoft Entra ID" (with a "View" button), and "Azure for Students" (with a "Start" button). Below these cards, the "Azure services" section is visible, featuring a "Create a resource" button and icons for various services like Quickstart Center, Azure AI services, Kubernetes services, Virtual machines, App Services, Storage accounts, SQL databases, Azure Cosmos DB, and More services.

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 Azure home page. However, a search overlay has been triggered, with the search term "Azure DevOps" entered into the search bar. The search results are displayed under the "All" tab, showing items such as "Azure DevOps organizations", "Azure Cosmos DB", "Azure Database for MySQL servers", "Azure Deployment Environments", "Build Agents for Azure DevOps", "Azure DevOps Auditing", "Azure DevOps Backup Tool", "Self Hosted Runner for Azure DevOps", "Secure your Azure DevOps - Azure DevOps", "Billing overview - Azure DevOps", and "Deploying to Azure VMs using deployment groups in Azure Pipelines - Azure PL...". The rest of the page content, including the "Azure services" section, remains visible.

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.

Microsoft Azure

Search resources, services, and docs (G+/)

Copilot

231801096@rajalakshmi...
DEFAULT DIRECTORY (231801096...)

Home > Azure DevOps ...

We've made it easier to manage Azure DevOps billing and subscriptions. You can [set up billing](#), [change your subscription](#) or [pay for more users and resources](#) within Azure DevOps. [Learn more](#)

Azure DevOps

Plan smarter, collaborate better, and ship faster with a set of modern dev services

My Azure DevOps Organizations

Get started using Azure DevOps
Billing management for Azure DevOps

Give feedback

Tell us about your experience with the Azure DevOps page



Result:

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

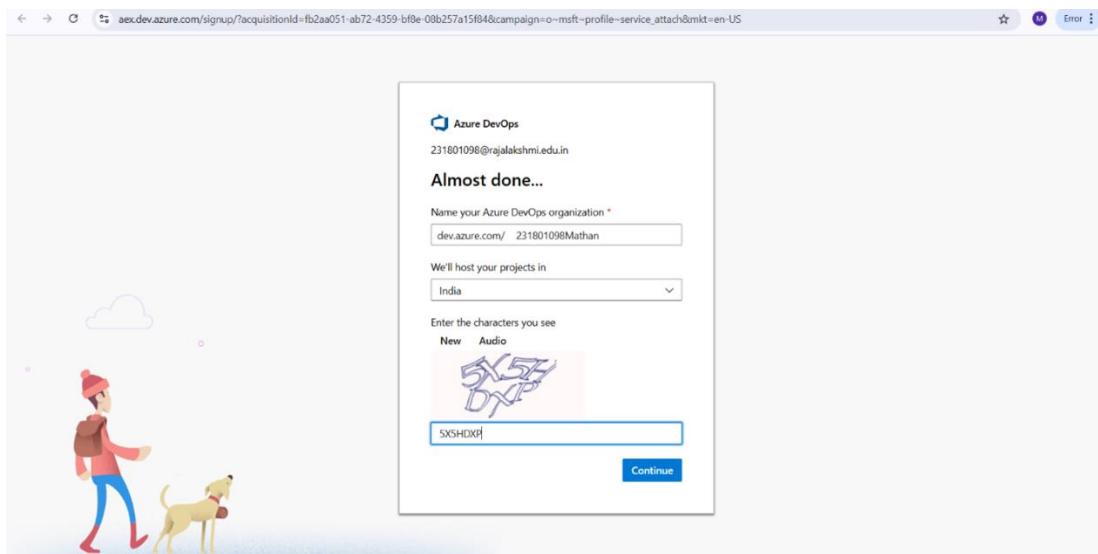
EXP NO: 2

AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

Aim:

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

1. Create An Azure Account



2. Create the First Project in Your Organization

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

b. On the organization's **Home page**, click on the **New Project** button.

c. Enter the project name, description, and visibility options:

Name: Choose a name for the project (e.g., **LMS**).

Description: Optionally, add a description to provide more context about the project.

Visibility: Choose whether you want the project to be **Private** (accessible only to those invited) or **Public** (accessible to anyone).

d. Once you've filled out the details, click **Create** to set up your first project.

Create new project

Project name *

Description

Visibility

Public
Anyone on the internet can view the project. Certain features like TFVC are not supported.

Private
Only people you give access to will be able to view this project.

Public projects are disabled for your organization. You can turn on public visibility with [organization policies](#).

Advanced

Version control [?](#)

Work item process [?](#)

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. On the left, there is a sidebar with the user's profile picture (a purple circle with a white 'M'), name (Manisha P), email (231801096@rajalakshmi.edu.in), and account information (Microsoft account, India). Below this is a section for 'Visual Studio Dev Essentials'. The main area displays the 'Azure DevOps Organizations' page with a 'Create new organization' button, a link to 'dev.azure.com/231801072 (Member)', and a message to 'Create a Team Project and start collaborating with your team now!'. On the right, there is a 'Actions' menu with options like 'Open in Visual Studio', 'Manage security', 'Browse extensions', and 'Leave'. A section titled 'Organizations Pending Deletion - Expand' is also visible.

4. Project dashboard

The screenshot shows the Azure Project dashboard for the project 'BATCH DATA ANALYSIS AND VISUALIZATION'. The top navigation bar includes links for Overview and Summary. The main content area features a 'About this project' section with a description of the project's goal: automating data ingestion, processing, storage, and visualization using Azure tools. To the right, there are two cards: 'Project stats' (showing 9 work items created and 0 work items completed) and 'Members' (listing five team members with their initials: JP, M, MS, KP, and a placeholder icon).

5. To manage user stories:

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

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

The screenshot shows the Azure DevOps Boards backlog page. At the top, there's a navigation bar with links for 'Organization Settings', 'Processes', and 'Fields'. Below that is a search bar and a 'Help' link. The main area is titled 'BATCH DATA ANALYSIS AND VISUALIZATION Team'. It features a 'Backlog' tab and an 'Analytics' tab. The backlog table has columns for Order, Work item type, Title, State, Effort, Business Area, and Tags. The data in the table is as follows:

Order	Work item type	Title	State	Effort	Business Area	Tags
1	Epic	Batch Data Analysis and Visualization on Azure	New		Business	
	Feature	User Authentication & Data Submission	New		Business	
	User Story	Secure User Login	New		Business	
	Task	Design Login User Interface (UI)	New			
	Task	Set Up Authentication System	New			
	Task	Validate User Credentials	New			
	Task	Implement Password Recovery Option	New			
	Task	Display Login Feedback Messages	New			
	User Story	Upload Batch Data	New		Business	
	Task	Design Upload Form Interface	New			
	Task	Support Multiple File Formats	New			
	Task	Validate File Before Upload	New			

Result:

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

EXP NO: 3

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

Aim:

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

Create Epic, Features, User Stories, Task

The screenshot shows the Azure Boards Backlog view for the 'BATCH DATA ANALYSIS AND VISUALIZATION Team'. The backlog is organized into sections: Backlog and Analytics. Under Backlog, there is a single Epic titled 'Batch Data Analysis and Visualization on Azure' which contains several child items: Feature ('User Authentication & Data Submission'), User Story ('Secure User Login'), and three Task items ('Design Login User Interface (UI)', 'Set Up Authentication System', 'Validate User Credentials'). Below these, another User Story item ('Upload Batch Data') has three associated Task items: 'Design Upload Form Interface', 'Support Multiple File Formats', and 'Validate File Before Upload'. All items are marked as 'New' and belong to the 'Business' category. The interface includes standard Azure navigation and search tools at the top.

1. Fill in Epics

The screenshot shows the details page for 'EPIC 141' titled 'Batch Data Analysis and Visualization on Azure'. The page includes fields for Status (New), Reason (New), Area (BATCH DATA ANALYSIS AND VISUALIZATION), and Iteration (BATCH DATA ANALYSIS AND VISUALIZATION/Automation, Monitoring and Error Handling). The 'Description' section contains a detailed text about building a cloud-based system for batch data analysis. The 'Planning' section shows priority (2), risk, effort, and business value. The 'Deployment' section provides instructions on tracking releases. The 'Development' section includes a link to an Azure Repos repository. The 'Related Work' section allows linking to existing work items. The bottom of the page shows classification and a value area.

2. Fill in Features

The screenshot shows a feature card in the Azure DevOps interface. The title is 'FEATURE 149' and the name is 'User Authentication & Data Submission'. The author is Mathan S. The status is 'New' and the area is 'BATCH DATA ANALYSIS AND VISUALIZATION'. The reason is 'New' and the iteration is 'BATCH DATA ANALYSIS AND VISUALIZATION\Automation, Monitoring and Error Handling'. The card has sections for 'Description', 'Planning', 'Deployment', 'Development', and 'Classification'. The 'Description' section contains a bullet point: 'Allows users to securely log in and submit batch data for processing.' The 'Planning' section includes fields for Priority (2), Risk, and Effort. The 'Deployment' section has a note about tracking releases. The 'Development' section includes an 'Add link' button and a note about Azure Repos. The 'Classification' section lists 'Value area' and 'Business'. The bottom of the screen shows a Windows taskbar with various icons.

3. Fill in User Story Details

The screenshot shows a user story card in the Azure DevOps interface. The title is 'USER STORY 217' and the name is 'Secure User Login'. The author is Mathan S. The status is 'New' and the area is 'BATCH DATA ANALYSIS AND VISUALIZATION'. The reason is 'New' and the iteration is 'BATCH DATA ANALYSIS AND VISUALIZATION\Automation, Monitoring and Error Handling'. The card has sections for 'Description', 'Acceptance Criteria', 'Discussion', 'Planning', 'Deployment', 'Development', and 'Classification'. The 'Description' section contains a bullet point: 'As a User, I want to log in securely so that I can access the platform to submit batch data.' The 'Acceptance Criteria' section has a placeholder 'Click to add Acceptance Criteria.'. The 'Planning' section includes fields for Priority (2), Risk, and Effort. The 'Deployment' section has a note about tracking releases. The 'Development' section includes an 'Add link' button and a note about Azure Repos. The 'Classification' section lists 'Value area' and 'Business'. The bottom of the screen shows a Windows taskbar with various icons.

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 Batch Data Analysis and Visualization

Sprint Planning

Sprints :

The screenshot shows the Azure Boards interface for the 'BATCH DATA ANALYSIS AND VISUALIZATION' team. The backlog board displays a list of user stories under the 'Automation, Monitoring and Error Handling' category. The stories are as follows:

Order	Title	State	Assigned To
1	Set Up Authentication System	New	Mathan S
1	Validate User Credentials	New	Mathan S
1	Implement Password Recovery Option	New	Mathan S
1	Display Login Feedback Messages	New	Mathan S
2	Upload Batch Data	New	Mathan S
2	Design Upload Form Interface	New	Mathan S
2	Support Multiple File Formats	New	
2	Validate File Before Upload	New	Mathan S
2	Store Uploaded Files in Cloud Storage	New	Mathan S
2	Display Upload Status	New	Mathan S
3	Store Uploaded Batch Data	New	Manisha P

231801096

CS23432

Result:

The Sprints are created for the Batch Data Analysis And Visualization.

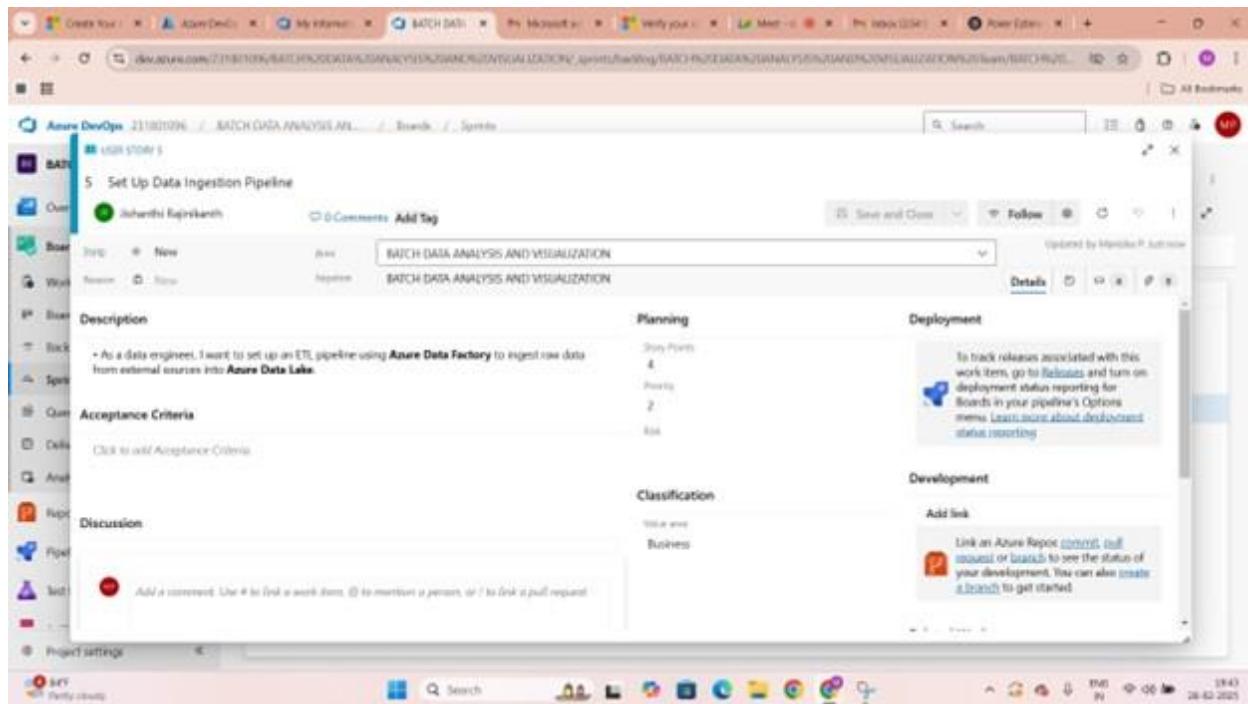
EXP NO: 5

POKER ESTIMATION

Aim:

Create Poker Estimation for the user stories - Batch Data Analysis And Visualization.

Poker Estimation



Result:

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

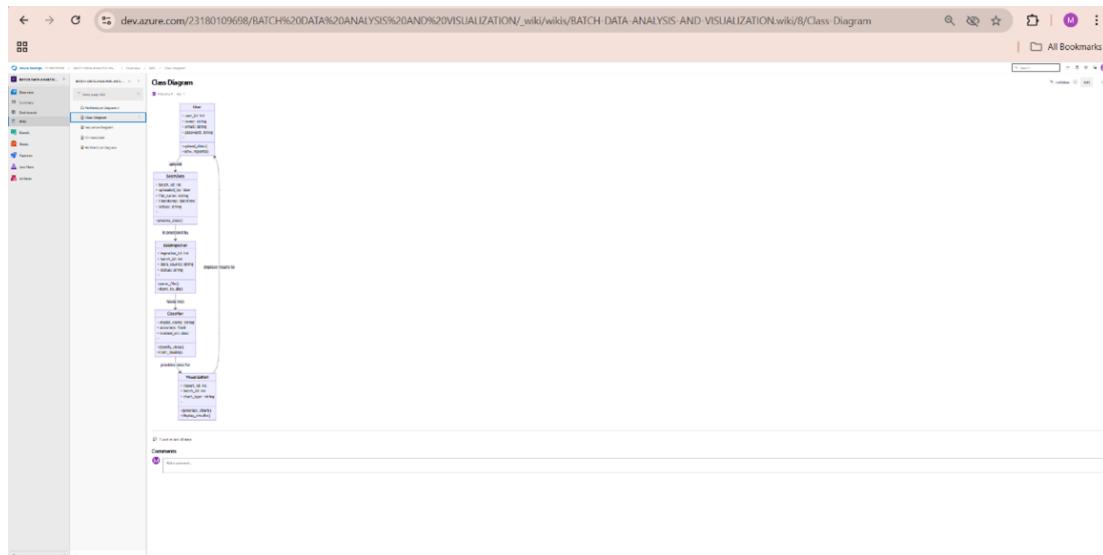
EXP NO: 6

DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE

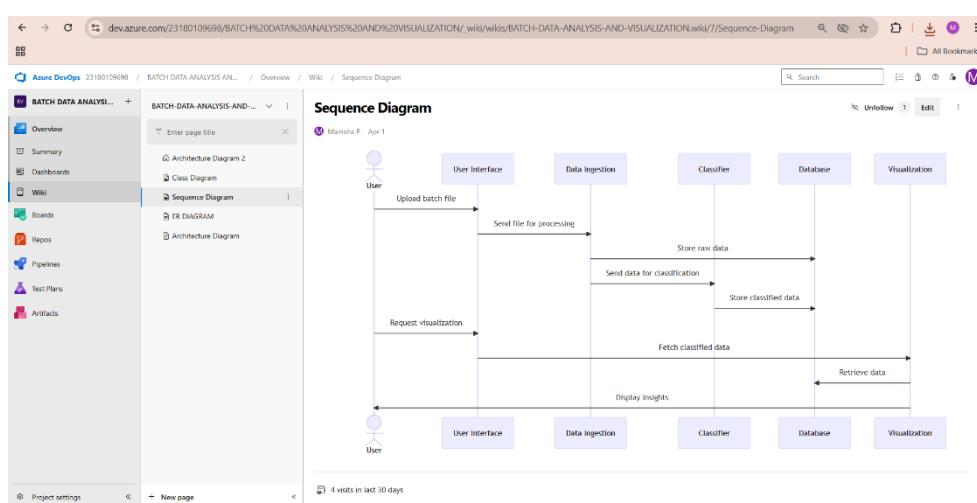
Aim:

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.

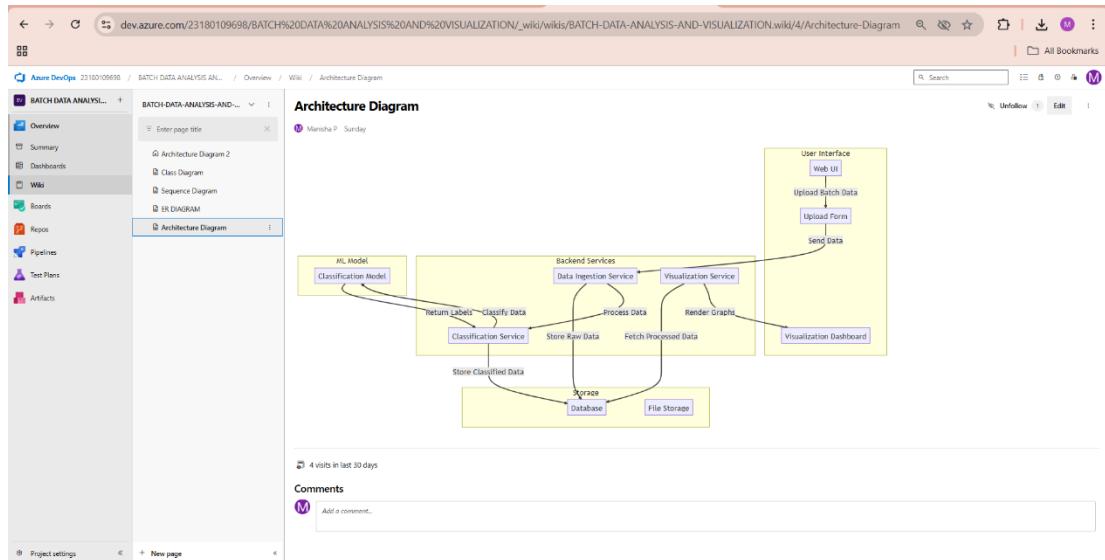
EXP NO: 7

DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE

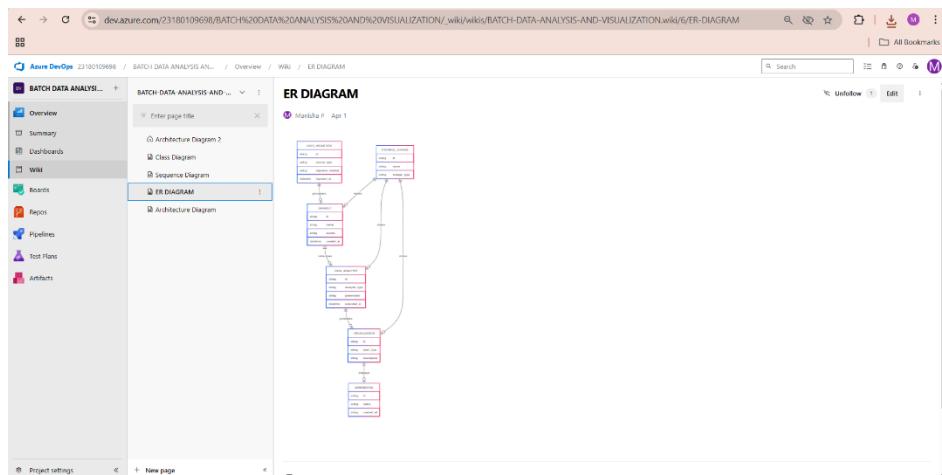
Aim:

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

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
- Allow users to securely create an account and log into the platform.
- Navigation to Data Upload Feature
- Enable users to navigate from the dashboard to the data upload section.
- Data Upload and Validation
- Allow users to upload CSV/Excel files and validate uploaded data for structure and format.
- Data Preprocessing and Cleaning
- Perform data cleaning activities like handling missing values, removing duplicates, and standardizing data.
- Data Visualization
- Generate visual insights (charts, graphs) based on processed data.

2. Define User Interactions

- Each test case simulates a real user behavior:
- Logging in to the platform.
- Uploading raw data files.
- Preprocessing and cleaning uploaded data.
- Generating charts and graphs from the processed data.
- Handling errors when invalid data or incorrect formats are uploaded.

3. Design Happy Path Test Cases

- Focused on validating that all features work properly under normal conditions:
- User logs in successfully with valid credentials.
- Data file (CSV/Excel) is uploaded successfully when correct format and structure are used.
- Missing values are handled correctly during preprocessing.
- Duplicate records are identified and removed.
- Graphs are generated correctly based on selected fields.

4. Design Error Path Test Cases

- Simulate negative or unexpected scenarios to test system robustness:
- Login fails when credentials are incorrect.
- Upload fails when file format is unsupported (e.g., .txt file).
- Upload fails when mandatory columns are missing in the file.
- Preprocessing fails when the dataset is empty.
- Visualization fails gracefully if selected fields are missing.

5. Break Down Steps and Expected Results

- Each test case includes:
- Step-by-step actions (e.g., click "Upload," select file, submit)
- Expected outcomes (e.g., "File uploaded successfully," "Error message displayed")

- This helps ensure clarity for manual testers and automation teams.

6. Use Clear Naming and IDs

- Test cases are clearly named and numbered for easy identification:
- TC01 – Successful Login
- TC02 – Successful Data Upload
- TC03 – Invalid Login Credentials
- TC04 – Upload File with Invalid Format
- TC05 – Generate Graph from Cleaned Data

7. Separate Test Suites

- Test cases are organized into logical groups for better execution:
- Functional Tests:
 - Login
 - Upload
 - Data Cleaning
 - Visualization
- UI Tests:
 - Navigation to Upload Section
 - Upload Button Visibility
 - Visualization Chart Display
- Edge Case Tests:
 - Invalid file formats
 - Missing fields in data
 - Uploading empty datasets
 - Login with empty fields

8. Prioritize and Review

- Critical user actions like Login, Upload, and Visualization are marked as High Priority.
- Test cases are reviewed for completeness and traceability to feature requirements.
- Priority factors considered:
 - Impact on system functionality
 - Business importance
 - Technical complexity

New test plan

The screenshot shows the 'New Test Plan' creation dialog in the Azure Test Management interface. The URL in the browser bar is https://dev.azure.com/23180109698/BATCH%20DATA%20ANALYSIS%20AND%20VISUALIZATION/_testManagement/new. The dialog has the following fields filled out:

- Name ***: LOGIN
- Area Path ***: BATCH DATA ANALYSIS AND VISUALIZATION
- Iteration ***: BATCH DATA ANALYSIS AND VISUALIZATION\Automation, Monitoring and Error Handling 2/25/2025 - 4/8/2025

At the bottom right of the dialog are 'Create' and 'Cancel' buttons.

- **Test suite**

The screenshot shows the Azure DevOps Test Plans interface. On the left, there's a sidebar with 'Test Suites' and a dropdown for 'DataVizBatch'. A context menu is open over the 'DataVizBatch' item, showing options like 'New Suite', 'Assign configurations', 'Export', 'Assign testers to run all tests', and 'Import test suites'. The main area displays a table titled 'Test Cases (3 items)'. The table has columns for 'Title', 'Order', 'Test Case Id', 'Assigned To', and 'State'. The rows are:

Title	Order	Test Case Id	Assigned To	State
upload large files(stress test)	1	234	Jishanthi Rajinika...	Design
remove content midway	2	235	Jishanthi Rajinika...	Design
load Datatypes	3	236	Jishanthi Rajinika...	Design

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 upload and view my data files. (ID: 101)
- As a user, I need to upload CSV/Excel files easily for analysis. (ID: 102)
- As a user, I want the system to clean and preprocess my uploaded data automatically. (ID: 103)
- As a user, I should be able to generate charts and graphs from my processed data. (ID: 104)
- As a user, I need clear error messages if upload or analysis fails. (ID: 105)

Test Suites

Test Suite: TS01 – User Login (ID: 110)

1. TC01 – Successful Login

- **Action:**
 - Go to the Login page.
 - Enter valid username and password.
 - Click "Login."
- **Expected Results:**
 - Login form accepts data.
 - User is redirected to the dashboard.

- **Type:** Happy Path

2. TC02 – Login with Invalid Credentials

- **Action:**

- Go to the Login page.
- Enter invalid username or password.
- Click "Login."

- **Expected Results:**

- Error message: "Invalid Username or Password" is displayed.

- **Type:** Error Path

Test Suite: TS02 – Data Upload (ID: 111)

1. TC03 – Successful Data Upload

- **Action:**

- Login successfully.
- Navigate to the "Upload" section.
- Select a valid CSV/Excel file.
- Click "Submit."

- **Expected Results:**

- File is uploaded successfully.
- Confirmation message: "Upload Successful."

- **Type:** Happy Path

2. TC04 – Upload Unsupported File Format

- **Action:**

- Login successfully.
- Navigate to "Upload."
- Select a .txt or .docx file.
- Click "Submit."

231801096

CS23432

- **Expected Results:**
 - Error message: "Unsupported file format."
- **Type:** Error Path

Test Suite: TS03 – Data Preprocessing (ID: 112)

1. TC05 – Successful Data Cleaning

- **Action:**
 - After successful upload, click "Start Preprocessing."
 - Allow the system to clean data (remove duplicates, handle missing values).
- **Expected Results:**
 - Data is cleaned and summary report is shown.
- **Type:** Happy Path

2. TC06 – Preprocessing Fails on Empty File

- **Action:**
 - Upload an empty CSV file.
 - Click "Start Preprocessing."
- **Expected Results:**
 - Error message: "Uploaded file is empty."
- **Type:** Error Path

Test Suite: TS04 – Data Visualization (ID: 113)

1. TC07 – Generate Graph Successfully

- **Action:**
 - After preprocessing, select fields for visualization.
 - Choose graph type (bar, pie, line).
 - Click "Generate Graph."
- **Expected Results:**
 - Graph is displayed correctly based on selected data.
- **Type:** Happy Path

2. TC08 – Graph Generation Fails without Selecting Fields

- **Action:**

- Skip selecting fields.
- Click "Generate Graph."

- **Expected Results:**

- Error message: "Please select fields to generate graph."

- **Type:** Error Path

Test Suite: TS05 – Error Handling & Alerts (ID: 114)

1. TC09 – Display Upload Error on Server Failure

- **Action:**

- Try uploading when server is down (simulate).

- **Expected Results:**

- Error message: "Server unavailable. Try again later."

- **Type:** Error Path

2. TC10 – Missing Mandatory Fields

- **Action:**

- Upload a file missing important columns (e.g., 'Name', 'Age').

- **Expected Results:**

- Error message: "Missing mandatory fields."

- **Type:** Error Path

Test Cases

The screenshot shows a Microsoft Test Case management interface. The test case is titled "upload large files(stress test)" and is assigned to Jishanthi Rajinikanth. It has 0 comments and no tags. The state is "Design". The iteration is "BATCH DATA ANALYSIS AND VISUALIZATION". The description is "BATCH DATA ANALYSIS AND VISUALIZATION/Automation, Monitoring and Error Handling".

Steps:

- User selects and uploads a .csv file with over 100,000 valid records.
- User waits for the upload and processing to complete.
- User monitors system performance during upload (e.g., response time of other UI elements).
- After upload, user checks if all records are inserted into the database or available in the batch list view.
- User checks for any error notifications or failure logs after upload.

Expected result:

- The system accepts the file and begins processing without crashing or freezing.
- System provides a loading indicator or progress bar, showing real-time progress until completion.
- The application remains responsive, and no significant performance lag is observed in the UI.
- All 100,000+ records are successfully saved and displayed in the system (or accessible via a link).
- No error messages are shown. System logs reflect successful processing of all records, and any invalid rows (if present) are clearly flagged with appropriate error messages.

Attachments:

Deployment:

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

Development:

Add link

[Link a GitHub commit, pull request or branch](#) to see the status of your development. You can also [create a branch](#) to get started. [Learn more](#)

Related Work:

Add link

[Add an existing work item as a parent](#)

Tests:

dev.azure.com/23180109698/BATCH%20DATA%20ANALYSIS%20AND%20VISUALIZATION/_testPlans/define?planId=223&suitId=233

TEST CASE 236

Upload File with Invalid Datatypes

Jishanthi Rajnikanth 0 Comments Add Tag

Save and Close Follow More All Bookmarks Updated by Manisha P. Monday

State: Design Area: BATCH DATA ANALYSIS AND VISUALIZATION Reason: New Iteration: BATCH DATA ANALYSIS AND VISUALIZATION\Automation, Monitoring and Error Handling

Steps

Steps	Action	Expected result	Attachments
1.	User uploads a .csv file containing invalid data types in specific fields (e.g., "ABC" in a quantity field that expects an integer)	System begins validating the file and detects the incorrect data types.	
2.	System processes the file and checks each record for data type compliance	Invalid records are flagged, and the system stops processing those rows while continuing with valid ones (if partial upload is supported).	
3.	System shows validation results	A clear error summary is displayed to the user: e.g., "Row 5: Invalid value in 'Quantity' field – expected number but found text."	

Click or type here to add a step

Parameter values

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 a GitHub [commit](#), [pull request](#) or [branch](#) to see the status of your development. You can also [create a branch](#) to get started. [Learn more](#)

Related Work

Add link

Add an existing work item as a parent

Tests

The screenshot shows a Microsoft Azure Test Case page. At the top, there are browser navigation buttons and a URL bar. Below the header, the test case title 'TEST CASE 236' and description 'Upload File with Invalid Datatypes' are displayed. The 'Steps' section contains three numbered steps with their corresponding actions and expected results. Step 1: 'User uploads a .csv file containing invalid data types in specific fields (e.g., "ABC" in a quantity field that expects an integer)' leads to 'System begins validating the file and detects the incorrect data types.'. Step 2: 'System processes the file and checks each record for data type compliance' leads to 'Invalid records are flagged, and the system stops processing those rows while continuing with valid ones (if partial upload is supported.)'. Step 3: 'System shows validation results' leads to 'A clear error summary is displayed to the user: e.g., "Row 5: Invalid value in 'Quantity' field – expected number but found text."'. Below the steps, there is a section for 'Parameter values'. To the right, there are sections for 'Deployment' (with a note about tracking releases), 'Development' (with GitHub integration options), and 'Related Work' (with links to add existing work items and tests). The bottom of the page has a footer with the ID '23180109698' and the name 'CS23432'.

dev.azure.com/23180109698/BATCH%20DATA%20ANALYSIS%20AND%20VISUALIZATION/_testPlans/define?planId=223&suitId=233

TEST CASE 235 cancel upload midway

Jishanthi Rajinikanth 0 Comments Add Tag

State: Design Area: BATCH DATA ANALYSIS AND VISUALIZATION
Reason: New Iteration: BATCH DATA ANALYSIS AND VISUALIZATION\Automation, Monitoring and Error Handling

Updated by Manisha P: Monday

Steps

Steps	Action	Expected result	Attachments
1.	User initiates the upload of a large .csv file (e.g., 100,000+ rows).	System immediately halts the upload process and stops reading the remaining data.	
2.	User clicks the "Cancel" button during the upload process.	System begins processing and shows a visible progress indicator (e.g., progress bar or spinner).	
3.	User checks the database or batch list after cancellation.	No new records from the canceled upload appear in the system. Partial uploads are not saved.	

Click or type here to add a step

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 a GitHub [commit](#), [pull request](#) or [branch](#) to see the status of your development. You can also [create a branch](#) to get started. [Learn more](#)

Related Work

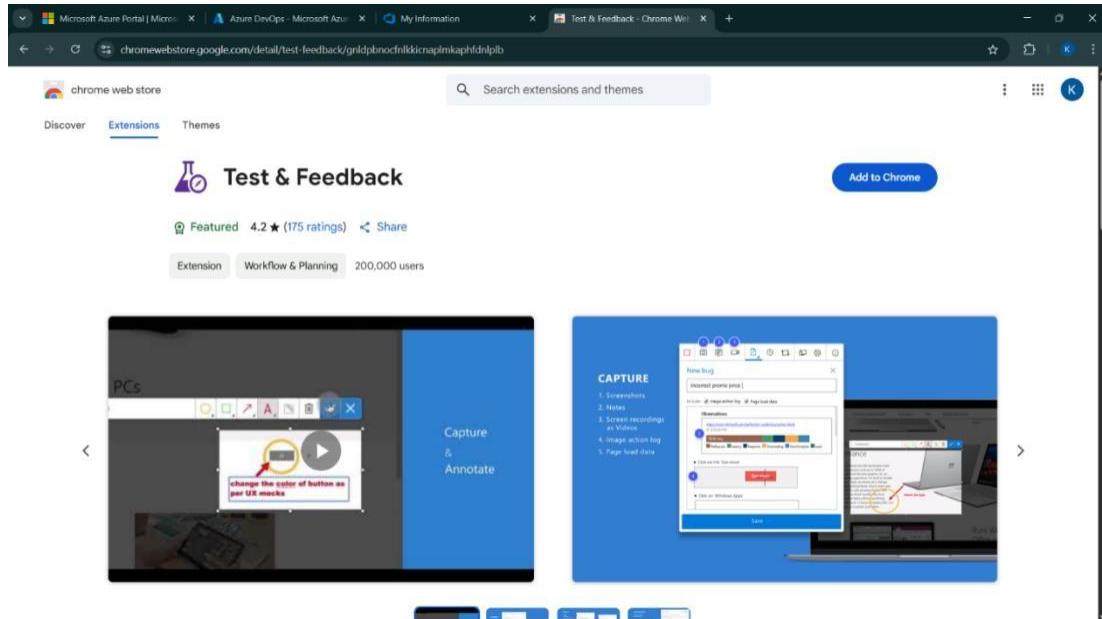
Add link

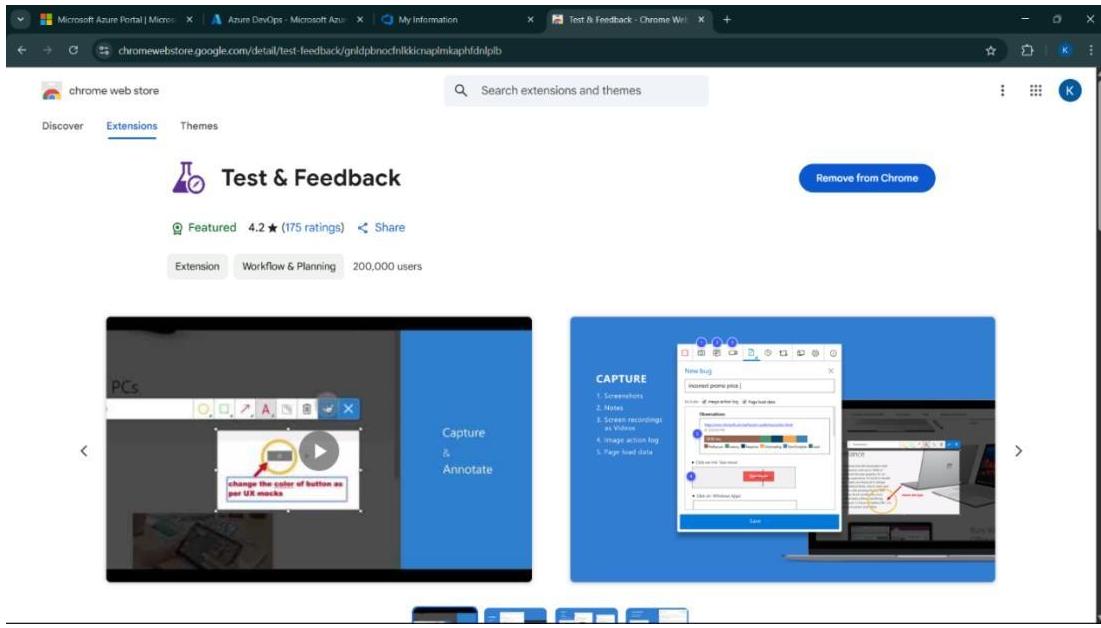
Add an existing work item as a parent

Tests

Parameter values

Installation of test





Test and feedback

Showing it as an extension

A screenshot of the Azure DevOps Test Plan interface for a project named "Music Playlist Batch Creator". The left sidebar shows navigation options like Overview, Boards, Repos, Pipelines, Test Plans, and Artifacts. The "Test plans" option is selected. In the main content area, a test plan titled "TS01 - User Login (ID: 86)" is displayed, showing test cases such as "TC01 - Successful Sign Up", "TC02 - Secure Login", "TC03 - Sign Up with Existing Email", and "TC04 - Login with Wrong Password". On the right side, a floating "Extensions" sidebar is open, listing extensions with "Full access": "Copy Text from Picture", "Dark Reader", "Monica: ChatGPT AI Assist...", "Selected: Copy Text from V...", and "Test & Feedback". A "Manage extensions" link is also present at the bottom of the sidebar.

- Running the test cases

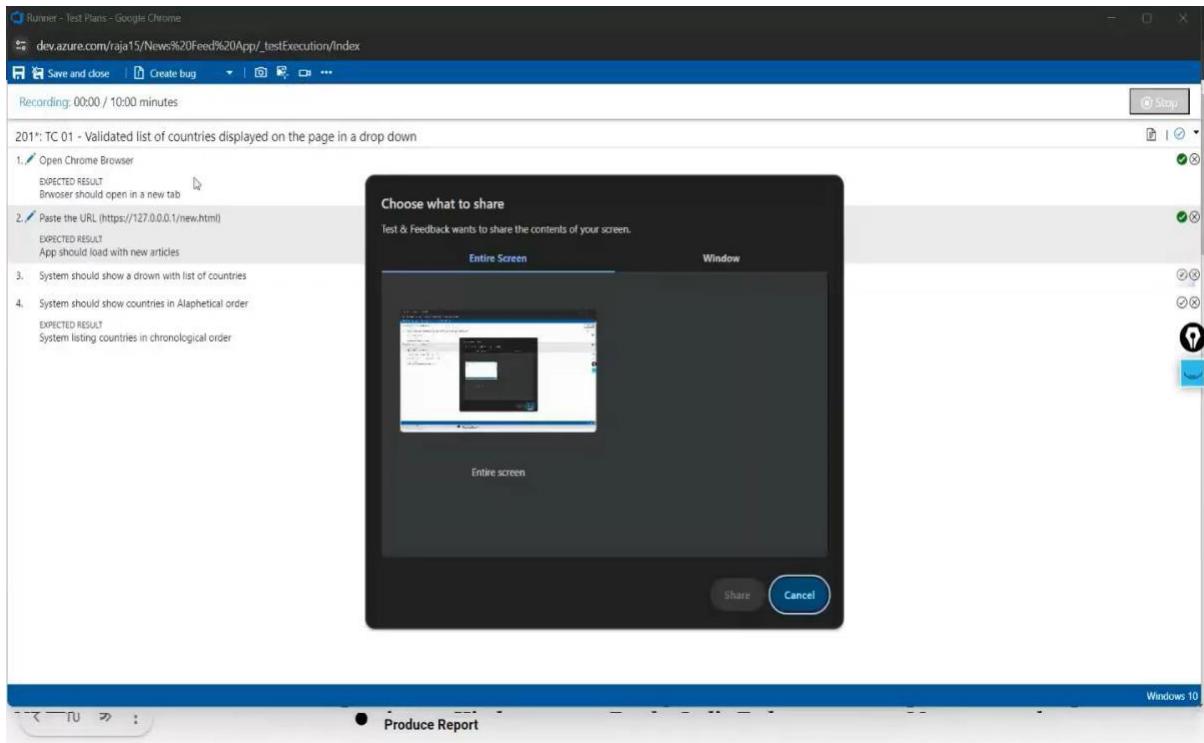
The screenshot shows the Azure DevOps Test Plans interface. On the left, the navigation bar includes 'Overview', 'Boards', 'Repos', 'Pipelines', 'Test Plans' (selected), 'Test plans', 'Progress report', 'Parameters', 'Configurations', 'Runs', and 'Project settings'. The main area displays a 'Test Suites' list under 'DataVizBatch (3)'. A specific test suite, '211 : Upload Batch Data (ID: 233)', is selected. The 'Execute' tab is active. A context menu is open over the first test point, 'upload large files(stress test)', listing options like 'Run for web application', 'Run for desktop application', and 'Run with options'. The test point table has columns for 'Title', 'Outcome', and 'Order'.

Title	Outcome	Order
<input checked="" type="checkbox"/> upload large files(stress test)	Passed	1
<input type="checkbox"/> cancel upload midway	Passed	2
<input type="checkbox"/> Upload File with Invalid Datatypes	Active	3

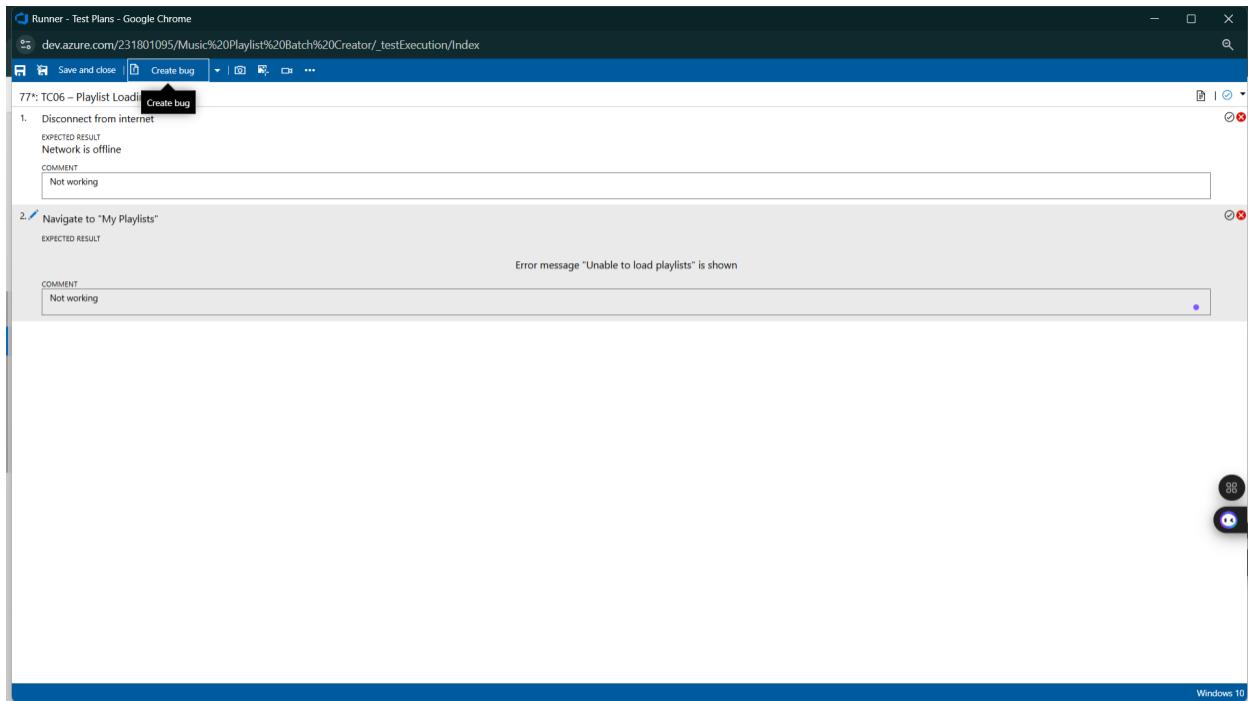
The screenshot shows the Azure DevOps Test Runs interface. The left sidebar includes 'Recent test runs' (selected), 'Test runs' (with a plus sign icon), 'Recent exploratory sessions', and other icons. The main area shows a table of 'Recent test runs' with columns: State, Run #, Title, Completed Date, Build Number, Failed, and Pass Rate. All runs are listed as 'Completed'.

State	Run #	Title	Completed Date	Build Number	Failed	Pass Rate
Completed	5	211 : Upload Batch Data (Manual)	4/21/2025 8:31:53 AM		0	100%
Completed	4	150 : View and Verify Charts (Manual)	4/21/2025 8:29:30 AM		0	100%
Completed	3	211 : Upload Batch Data (Manual)	4/21/2025 8:13:59 AM		0	100%
Completed	2	150 : View and Verify Charts (Manual)	4/21/2025 8:13:15 AM		0	100%

- Recording the test case



- Creating the bug



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

77: TC06 - Playlist Loading Failure

1. Disconnect from internet

TB01 - Playlist loading spinner keeps spinning indefinitely on poor network

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

2. Navigate to "My Playlists"

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

Repro Steps

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

Step no. Result Title
1. Failed Disconnect from internet
Expected Result Network is offline
Comments: Page Not loading
2. Failed Navigate to "My Playlists"
Expected Result
Error message "Unable to load playlists" is shown

Test Configuration: Windows 10

Planning Deployment

Resolved Reason 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

Story Points Priority 2
Severity 3 - Medium
Activity

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) Related Work
Original Estimate Remaining Completed
+ Add link Add an existing work item as a parent
Tested By 77 TC06 - Playlist Loading Failure Updated 10-04-2025, 8: Design

System Info Found in Build

Windows 10

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

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

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

Search

Music Playlist Batch Creator + Enter Run ID... Go Run 48 - TS02 - View Playlists (Manual) / TC06 - Playlist Loading Failure

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

Unassigned 0 comments Add tag Save & Close ... Follow ...

System Info Updated by Karthick S 8m ago

Overview Board Repos Pipelines Test Plans Test plans Program Parameters Configuration Runs Artifacts

Test Plan 84 Music | Run 48 - TS02 - View Playlists (Manual) / TC06 - Playlist Loading Failure

System Info Found in Build Integrated in Build

Test Info

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

Discussion

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

Recent attachments

Project settings <> Name SystemInformation-2025-04-18T03-23-58.168Z.json Size 1K

- Test case results

The screenshot shows the Azure DevOps Test Plans interface. On the left, there's a navigation pane with 'DataVizBatch' selected. Under 'Test Suites', '150 : View and Verify Charts (3)' is highlighted. The main area displays '150 : View and Verify Charts (ID: 230)' with tabs for 'Define', 'Execute', and 'Chart'. The 'Execute' tab is active, showing 'Test Points (3 items)'. One point, 'Validate supported file formats (CSV, JSON, Parquet)', is checked and highlighted in blue. To the right, a modal window titled 'Test Case Results' shows a table with one row: 'Passed' (Outcome), 'Monday' (TimeStamp), 'Windows 10' (Configuration), 'Manisha P' (Run by), 'Kavinya P' (Tester), and 'DataVizBatch' (Test Plan). A link 'Open execution history for current test point' is at the bottom.

- Test report summary

The screenshot shows the Azure DevOps Work Items interface. On the left, the 'News Feed App' project is selected. In the center, a bug report 'BUG 203' is displayed. The title is '203 - BG 01 - Countries Drop down Not Available on the page'. The 'Reason' dropdown is open, showing 'New' (selected), 'Resolved', and 'Closed'. The 'Repro Step' section shows three steps: 1. Passed (Result: Open Chrome Browser, Expected Result: Browser should open in a new tab), 2. Passed (Result: Paste the URL (https://127.0.0.1/new.html), Expected Result: App should load with new articles), and 3. Failed (Result: System should show a dropdown with list of countries, Expected Result: Produce Report). The 'Planning' section includes 'Story Points' (2), 'Priority' (2), 'Severity' (3 - Medium), and 'Activity'. The 'Deployment' section has a note about tracking releases. The 'Development' section has a note about linking to Azure Repos. The 'Effort (Hours)' section shows 'Original Estimate' (0). The 'Related Work' section is collapsed.

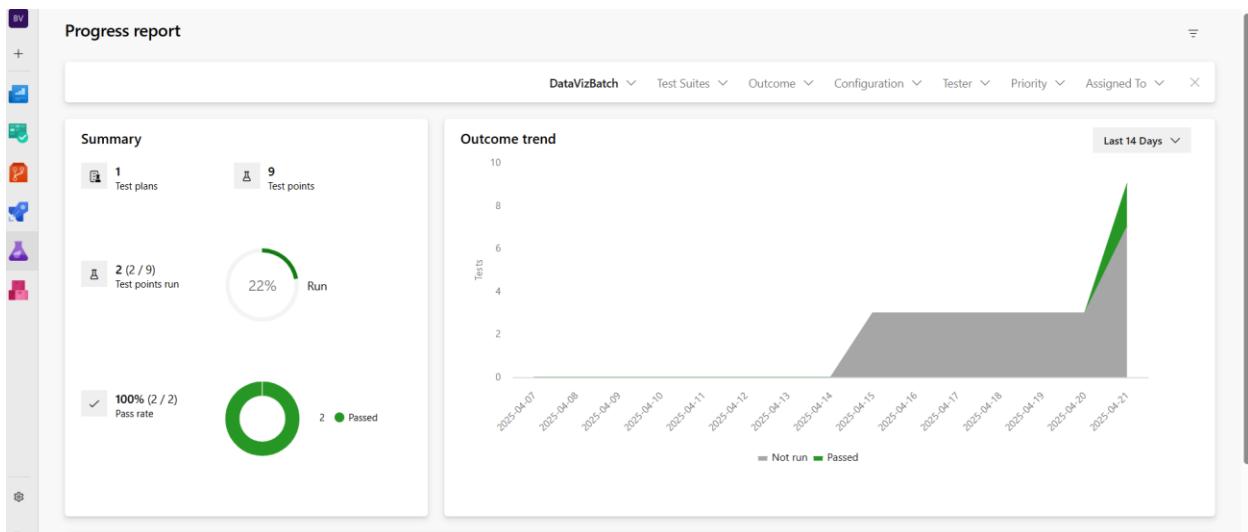
- Assigning bug to the developer and changing state

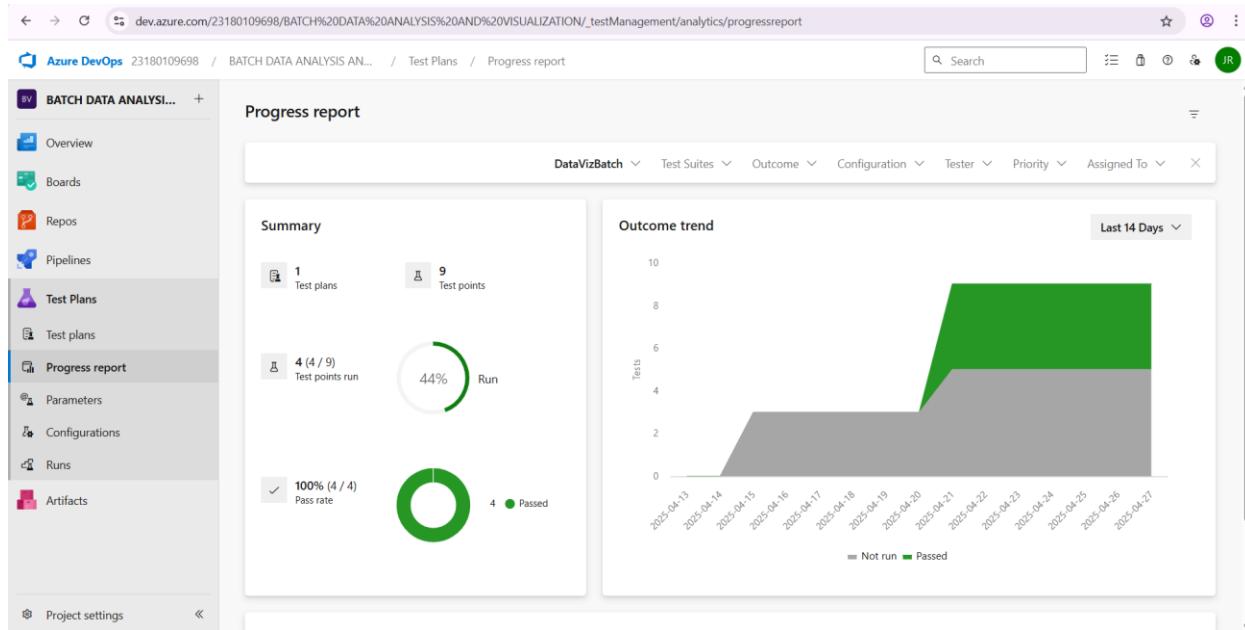
The screenshot shows a detailed view of a test step in the Azure DevOps Test Plan interface. The step is titled "92 TB01 - Playlist loading spinner keeps spinning indefinitely on poor network". It is categorized under "Repro Steps" and has two steps listed:

- Step no.**: 1, **Result**: Failed, **Title**: Disconnect from internet. Expected Result: Network is offline.
- Step no.**: 2, **Result**: Failed, **Title**: Navigate to "My Playlists". Expected Result: Error message "Unable to load playlists" is shown.

The "Planning" section includes fields for Resolved Reason, Story Points, Priority (2), Severity (3 - Medium), and Activity. The "Deployment" section shows a link to track releases associated with this work item. The "Development" section includes a link to an Azure Repos pull request. The "Related Work" section allows adding existing work items as parents. The "System Info" section shows the tester as "Karthick S" and the last update date as "Updated 10-04-2025".

- **Progress report**





- **Changing the test template**

All processes

Name	Description	Team projects
Basic (default)	This template is flexible for any process and great for teams getting started with Az...	0
Agile	This template is flexible and will work great for most teams using Agile planning me...	1
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 improv...	0

The screenshot shows the 'All processes' list in the Azure DevOps Settings - Process page. The 'Processes' tab is selected. The list includes:

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

The screenshot shows the 'All processes' list in the Azure DevOps Settings - Process page. The 'Processes' tab is selected. The list includes the standard templates and a new entry under the Agile category:

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

- View the new test case template

The screenshot shows the 'Add a field to Test Case' dialog box overlaid on the Azure DevOps settings interface. The dialog has a 'Definition' tab selected, showing options to 'Create a field' or 'Use an existing field'. Under 'Create a field', the 'Name' is set to 'Type' and the 'Type' is set to 'Text (single line)'. A 'Description' field is also present. At the bottom are 'Add field' and 'Cancel' buttons.

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

The screenshot shows the Azure DevOps Settings - Process page. The URL in the address bar is dev.azure.com/231801095/_settings/process?type-id=231801095Agile.TestCase&process-name=231801095%20Agile&_a=layout. The page title is "All processes > 231801095 Agile > Test Case". The left sidebar is titled "Organization Settings" and includes sections for General, Security, Boards, Pipelines, and Process. The "Process" section is currently selected. The main content area shows a form for creating a "Test Case". The "Steps" field is highlighted, containing the text "Text (multiple lines)". To the right of the form, there are several sections: "Custom" (Type: Text (single line)), "Recent test results" (Recent test case results), "Deployment" (Deployments), "Development" (Links), "Related Work" (Links), and "Status" (Priority: Integer, Automation status). A large "Add a field ..." button is located at the bottom right of the form area.

Result:

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

EXP NO: 9	LOAD TESTING AND PIPELINES
-----------	-----------------------------------

Aim:

To create an Azure Load Testing resource and run a load test to evaluate the performance of a target endpoint.

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

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

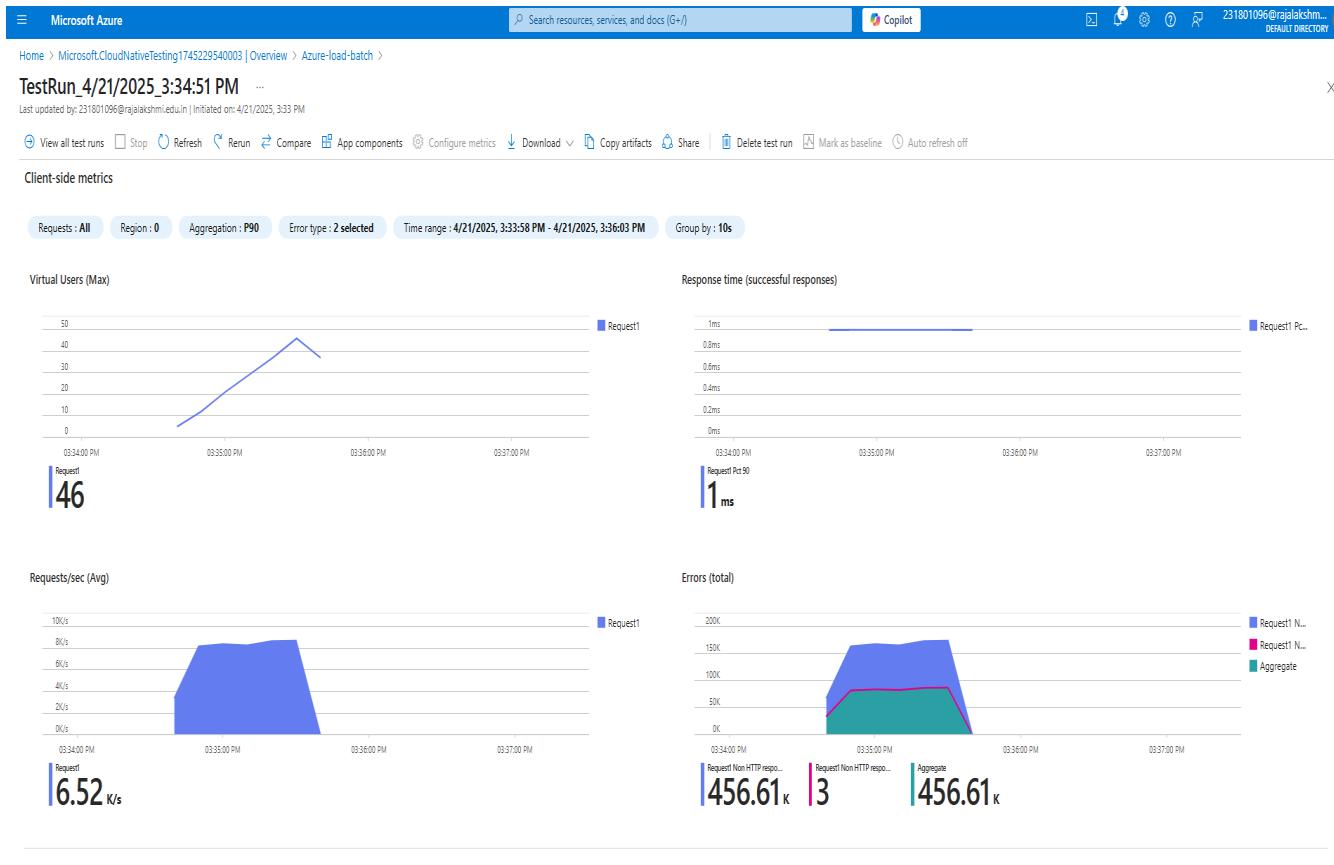
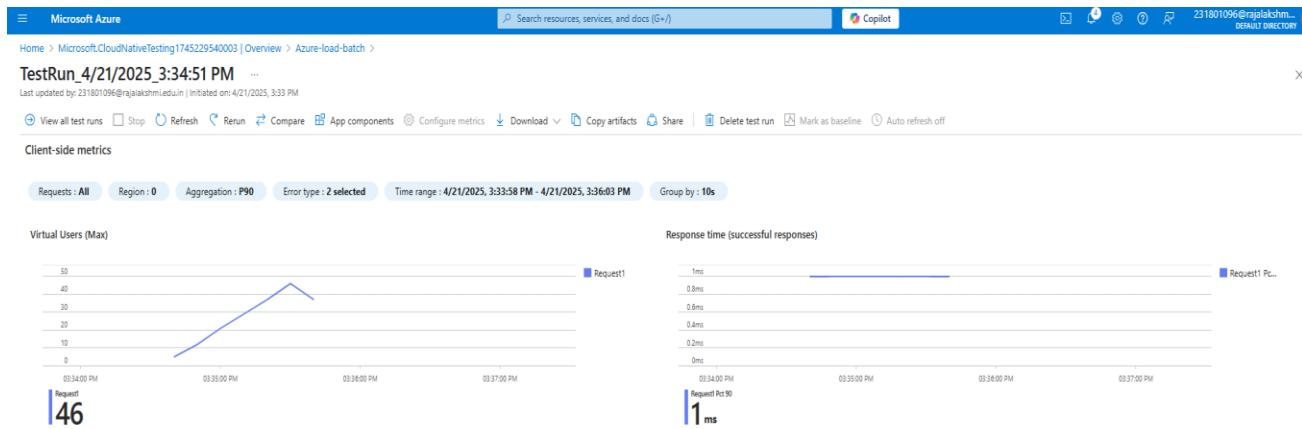
1. Sign in to Azure Portal
Go to <https://portal.azure.com> and log in.
2. Create the Resource
 - o Go to *Create a resource* → Search for “Azure Load Testing”.
 - o Select Azure Load Testing and click Create.
3. Fill in the Configuration Details
 - o *Subscription*: Choose your Azure subscription.
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

Steps to Create and Run a Load Test:

Once your resource is ready:

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

Load Testing



Server-side metrics

← → ⌂ http://127.0.0.1:5500/Batch data analytics and visualization.html 80% ☆ ⌂ ⌂ ⌂ ⌂ ⌂ ⌂

Batch Data Analysis & Visualization

Use Microsoft Azure tools to analyze, process, and visualize your batch data

Azure Synapse Analytics

Analyze massive amounts of data quickly using Azure Synapse. Integrate with big data tools and build high-performance pipelines for batch analytics.

[Visit Synapse Analytics](#)

Azure Data Factory

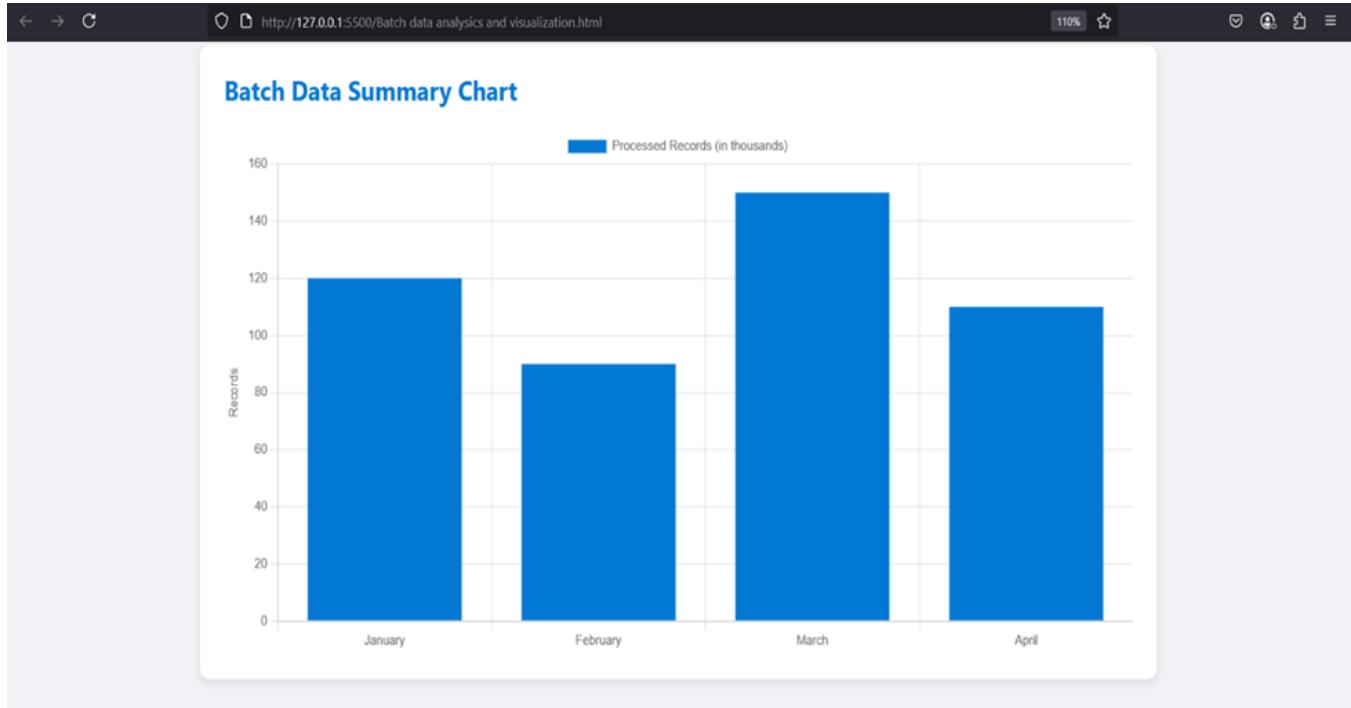
Build data pipelines, automate batch processing, and transform raw data at scale with Azure Data Factory's low-code interface.

[Visit Data Factory](#)

Power BI

Create stunning dashboards and interactive visualizations to explore your batch data insights with Power BI.

[Visit Power BI](#)



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:

- o Setting up the Python environment using the UsePythonVersion task.

- o Installing project dependencies from project/requirements.txt. Make sure the path to requirements.txt is correct (it is located under the project folder).

- o Running a simple Python script to verify that Python is set up correctly and the pipeline works.

4. Run and Monitor Pipeline:

- o Commit changes to the main branch of your repository to trigger the pipeline in Azure DevOps.

- o Monitor the logs in the Azure DevOps portal to view logs, errors, or success messages and ensure everything runs smoothly.

Pipeline:

The screenshot shows the Azure DevOps Pipelines interface for a project named "Music Playlist Batch Creator". A specific pipeline run, #20250424.3, is displayed. The summary indicates it was manually run by Karthick S. The run was started just now and completed successfully in 65 seconds. The repository is Music Playlist Batch Creator, branch main, commit a87bd670. There were 52 changes in this run. The pipeline consists of one job, which also completed successfully.

Name	Status	Duration
Job	Success	65

Result:

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

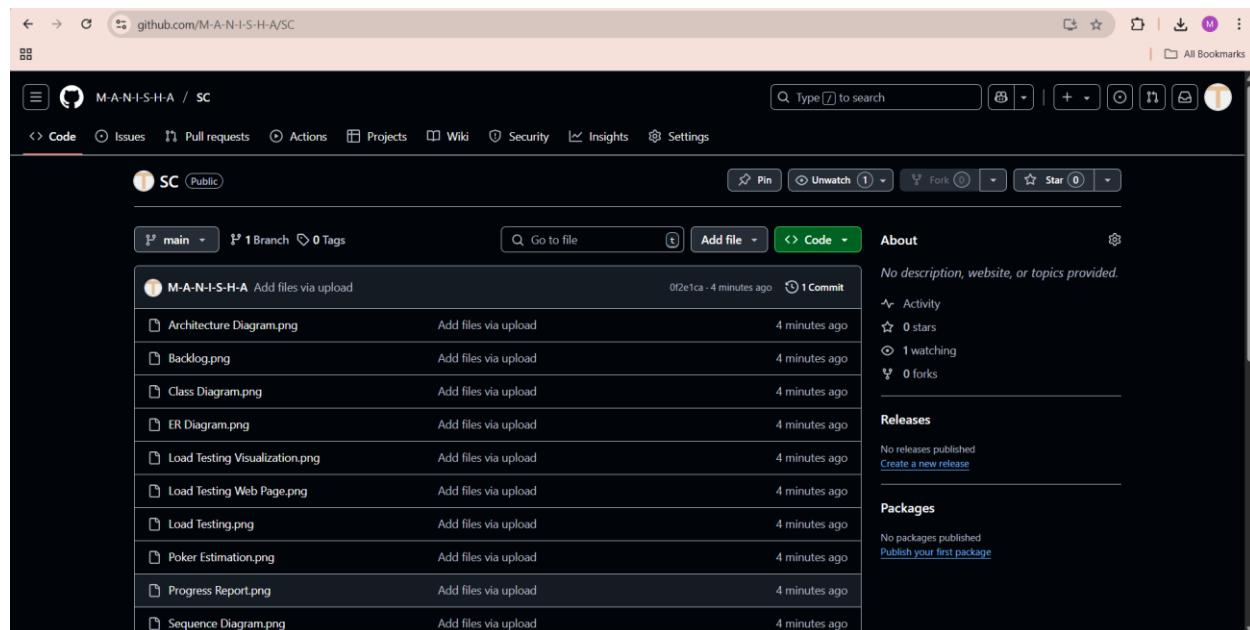
EXP NO: 10

GITHUB: PROJECT STRUCTURE & NAMING CONVENTIONS

Aim:

To provide a clear and organized view of the project's folder structure and file naming conventions, helping contributors and users easily understand, navigate, and extend the Batch Data Analysis And Visualization.

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.