



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

**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND
DATA SCIENCE LAB RECORD**

CS23432 – Software Construction

(REGULATION 2023)

RAJALAKSHMI ENGINEERING COLLEGE
Thandalam, Chennai-602015

Name: Goutham Raj K

Register No: 231801044

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

Semester: IV

Academic Year: 2024 - 2025

INDEX

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

EXP NO: 1

AZURE DEVOPS ENVIRONMENT SETUP

Aim:

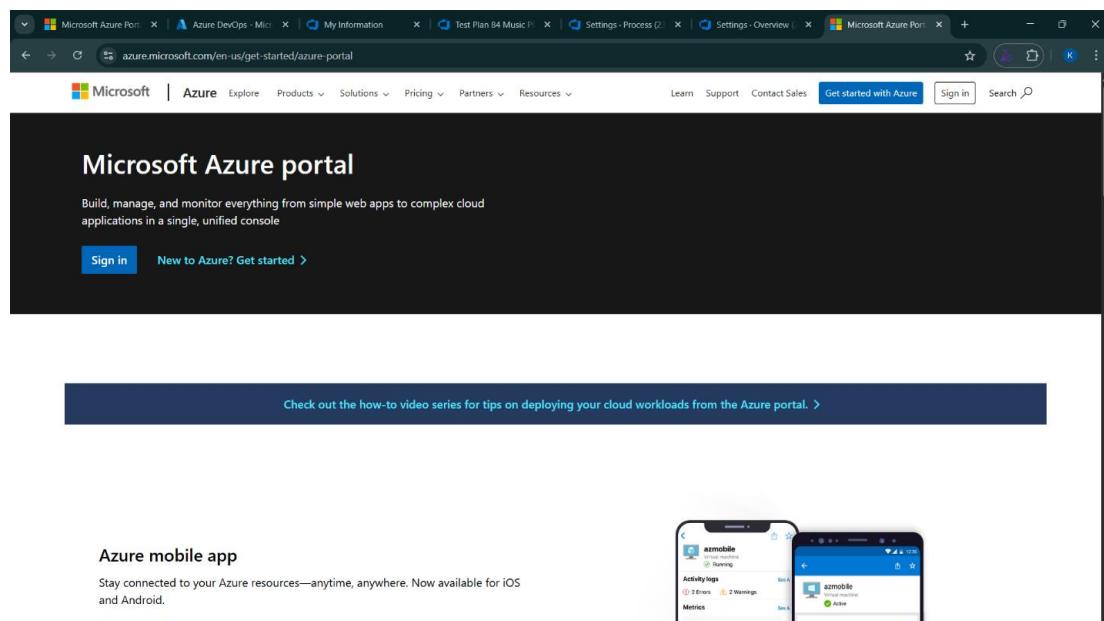
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

INSTALLATION

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

Sign in using your Microsoft account credentials.

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



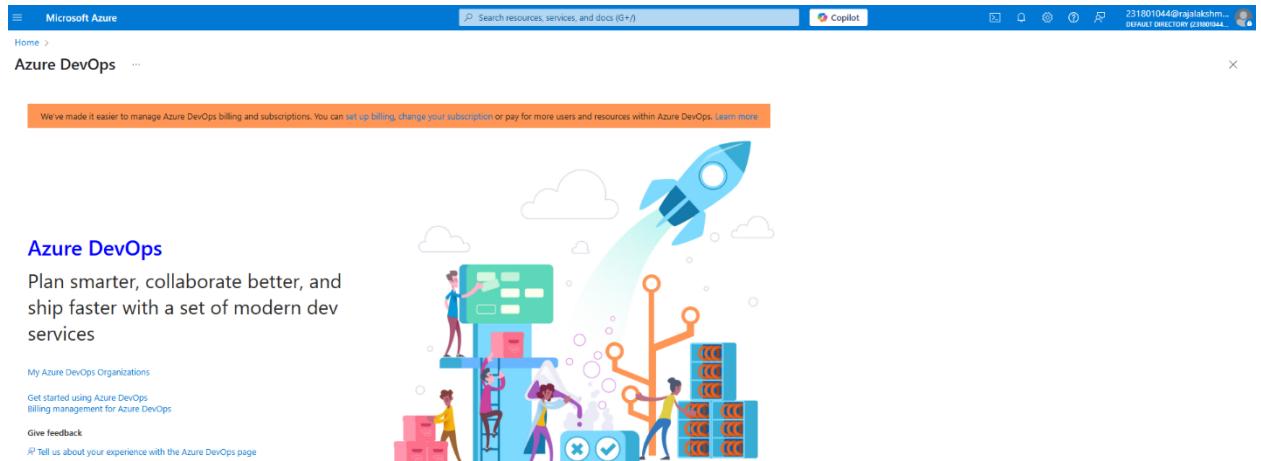
2. Azure home page

The screenshot shows the Microsoft Azure home page. At the top, there's a search bar with the placeholder "Search resources, services, and docs (G+)" and a Copilot button. The user's email "giridharanr777@outlook.com" is visible in the top right. Below the header, the "Azure services" section features a "Create a resource" button and icons for various services: DevOps organizations, Quickstart Center, AI services, Kubernetes services, Virtual machines, App Services, Storage accounts, and SQL databases. A "More services" link is also present. The "Resources" section shows a "Recent" tab selected, with a table for "Name", "Type", and "Last Viewed". A message indicates "No resources have been viewed recently" with a "View all resources" button. The "Navigate" section at the bottom has a "Recent" tab.

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

The screenshot shows the Microsoft Azure home page with a search bar containing the text "devops". The search results are displayed in a modal window. Under the "Services" category, items include "Azure Native New Relic Service", "Managed DevOps Pools", and "Azure DevOps organizations". Under "Marketplace", items include "Static Web App", "Rocky Linux 9", and "Build Agents for Azure DevOps". Under "Documentation", there's a link to "Secure your Azure DevOps - Azure DevOps". The "More services" section on the right includes icons for Services, Storage accounts, and SQL databases, along with a "More services" link. The "Navigate" section at the bottom has a "Recent" tab.

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



Result:

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

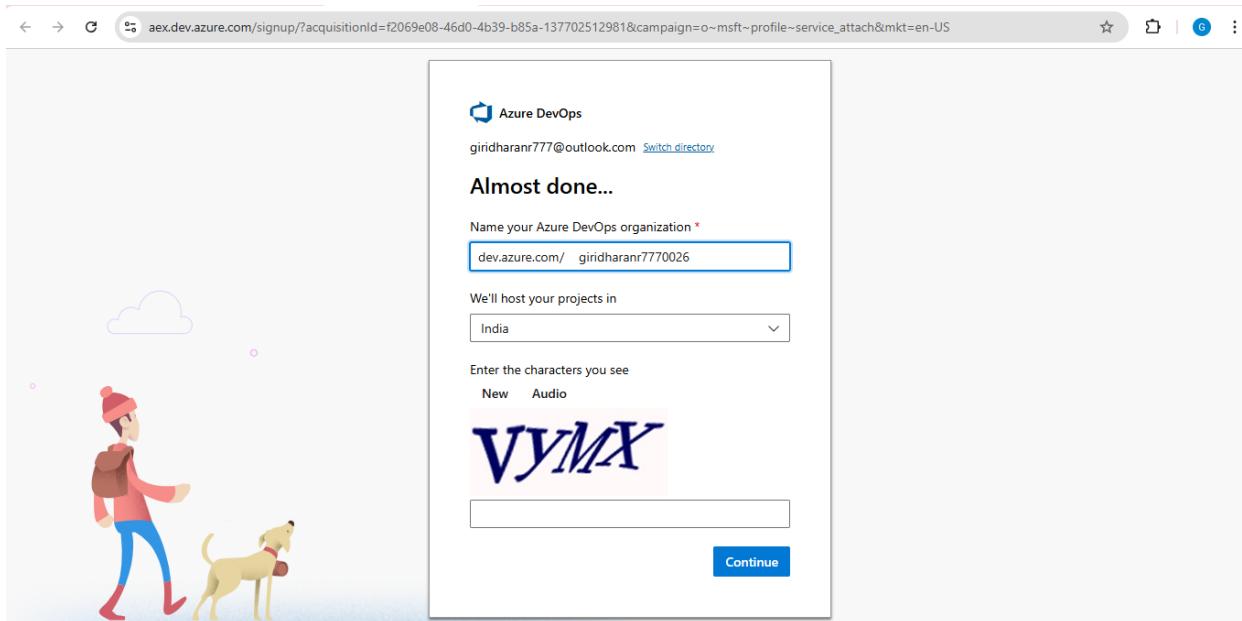
EXP NO: 2

AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

Aim:

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

1.Create An Azure Account



2.Create the First Project in Your Organization

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

Create new project

X

Project name *

Description

Weather App

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.

Advanced

Version control [?](#)

Git

Work item process [?](#)

Agile

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 profile page for Goutham Raj. At the top, there is a large orange circular profile picture with the letters 'GR' in white. Below it, the name 'Goutham Raj' and email '231801044@rajalakshmi.edu.in' are displayed, along with a 'Edit profile' button. A dropdown menu shows 'Microsoft account'. Below this, there is contact information: 'India' and the same email address again. A section for 'Visual Studio Dev Essentials' is present, encouraging users to build and deploy apps on any platform, with a 'Use your benefits' link.

4. Project dashboard

The screenshot shows the Azure DevOps project dashboard for 'Weather App'. The left sidebar has a navigation menu with 'WA Weather App' at the top, followed by 'Overview', 'Summary' (which is selected), 'Dashboards', 'Wiki', 'Boards', 'Repos', 'Pipelines', 'Test Plans', and 'Artifacts'. At the bottom of the sidebar are 'Project settings' and a back arrow. The main content area has a title 'Weather App' with a 'WA' icon. Below it is a 'About this project' section containing a description of the app's features. To the right are two cards: 'Project stats' showing '43 Work' and '0 Work' (with a 'Period: Last 7 days' dropdown), and 'Members' showing five team members with their initials (GR, Z, AP, GH) and profile icons.

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 'The Weather App' project. The left sidebar is the navigation menu with options like Overview, Boards, Work items, Repos, Pipelines, Test Plans, and Artifacts. The 'Work items' section is selected. The main area displays a table of work items under the 'Work items' tab. The columns are ID, Title, Assigned To, State, Area Path, and Tags. The first item, ID 8, is highlighted with a blue background and shows 'as a user I wantr to login' as the title, assigned to 'Goutham Raj' (status: New), and belongs to 'The Weather App'. Other items include 'test login' (status: Design), 'to create login feature' (status: New), and several unassigned items like 'signup and weather details page', 'weather icon', etc. The top right corner shows the user profile of 'Goutham Raj' with the email '231801044@rajalakshmi.edu.in' and a sign-in link.

ID	Title	Assigned To	State	Area Path	Tags
8	as a user I wantr to login	Goutham Raj	New	The Weather App	
10	test login	Goutham Raj	Design	The Weather App	4/29/2025 4:21:49 PM
9	to create login feature	Unassigned	New	The Weather App	4/29/2025 4:21:27 PM
5	signup and weather details page	Goutham Raj	Active	The Weather App	3/26/2025 7:44:35 AM
7	weather icon	Unassigned	New	The Weather App	2/12/2025 7:43:01 AM
6	Show weather conditions	Unassigned	New	The Weather App	2/12/2025 7:41:42 AM
4	UI design simple	Unassigned	New	The Weather App	2/12/2025 7:40:26 AM
3	View Current Temperature	Unassigned	New	The Weather App	2/12/2025 7:38:59 AM
2	Real-Time Weather updates	Unassigned	New	The Weather App	2/12/2025 7:37:26 AM
1	To build a weather app	Unassigned	New	The Weather App	2/12/2025 7:36:31 AM

Result:

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

EXP NO: 3

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

Aim:

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

Create Epic, Features, User Stories, Task

The screenshot shows the Azure DevOps interface for the 'Weather App' project. The left sidebar has 'Backlogs' selected. The main area displays the 'Backlog' tab of the 'Weather App Team' backlog. The backlog table includes columns for Order, Work Item Type, Title, State, Effort, Business Area, and Tags. The backlog items are as follows:

Order	Work Item Type	Title	State	Effort	Business Area	Tags
1	Epic	To Build a Weather APP	Active		Business	
	Feature	Forecast Prediction	Active		Architectural	
	User Story	ASa user,i want to view weather forecast for next 7 da...	New		Business	
	Task	Fetch 7 day forecast from API	New			
	Task	Display forecast with icons and descriptions	New			
	Task	optimize API calls for better performance	New			
	Feature	Fetch Weather Data	Active		Business	
	User Story	As a user, I want to view current weather conditions fo...	New		Business	
	Task	Connect to a weather API	New			
	Task	Retrieve temperature, humidity, and wind speed da...	New			
	Task	Display data in a user-friendly format.	New			

1.Fill in Epics

The screenshot shows the Azure DevOps interface for creating a new Work Item. The left sidebar is titled 'Weather App' and includes options like Overview, Boards, Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, and Project settings. The main area is titled 'Work Items' and shows a 'NEW EPIC' card for 'Real-Time Weather Information'. The card details include:

- Owner: Giridharan R
- Comments: 0
- Tags: Add Tag
- State: New
- Reason: New
- Area: Weather App
- Iteration: Weather App
- Time Criticality: Select a date...
- Start Date: Select a date...
- Target Date: Select a date...
- Classification: Value area: Business

On the right, there are sections for 'Development' (with a note about linking to Azure Repos), 'Related Work' (with a link to add an existing work item), and a 'Details' panel.

2.Fill in Features

The screenshot shows the Azure DevOps interface for creating a new Work Item. The left sidebar is titled 'Weather App' and includes options like Overview, Boards, Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, and Project settings. The main area is titled 'Backlog' and shows a 'FEATURE 20' card for 'Forecast Prediction'. The card details include:

- Owner: Giridharan R
- Comments: 0
- Tags: Add Tag
- State: Active
- Reason: Implementation started
- Area: Weather App
- Iteration: Weather App

The card has sections for 'Description' (Provides users with a 7-day weather forecast.), 'Planning' (Priority: 2, Risk: 2 - Medium, Effort: Business Value, Time Criticality: Select a date...), and '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](#)). On the right, there are sections for 'Development' (with a note about linking to Azure Repos) and a 'Details' panel.

3.Fill in User Story Details

← → ⌂ dev.azure.com/giridharanr777/Weather%20App/_backlogs/backlog/Weather%20App%20Team/Epics?workitem=37

Azure DevOps | giridharanr777 / Weather App / Boards / Backlogs

USER STORY 37*

Giridharan R 0 Comments Add Tag

Save and Close Follow Details

Updated by Giridharan R: Feb 19

WA Weather
Ov Overall
Boards Boards
Work Work Items
Backlogs Backlogs
Sprints Sprints
Quizzes Quizzes
Deliverables Deliverables
Analytics Analytics
Reports Reports
Pipelines Pipelines

Project settings

37 AS a user, I want to view weather forecast for next 7 days so that I can prepare accordingly.

State: New Area: Weather App Reason: New Iteration: Weather App\Sprint 4

Description

Acceptance Criteria

Click to add Acceptance Criteria.

Discussion

Planning Deployment

Story Points: 3 Priority: 2 Risk:

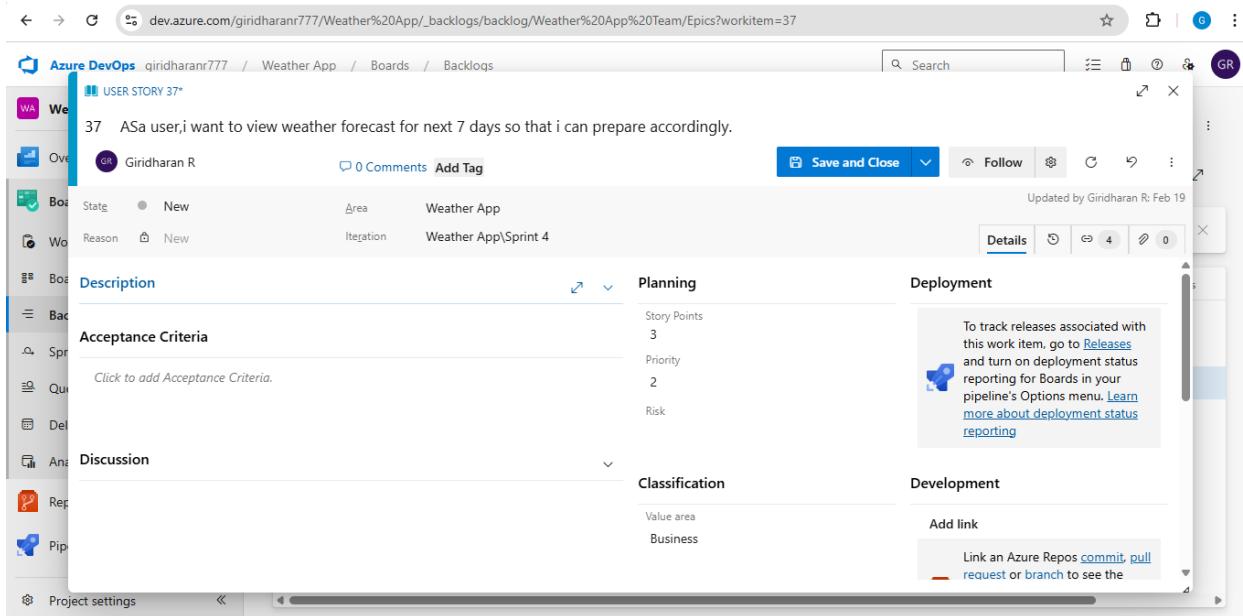
To track releases associated with this work item, go to [Releases](#) and turn on deployment status reporting for Boards in your pipeline's Options menu. [Learn more about deployment status reporting](#)

Classification Development

Value area: Business

Add link

Link an Azure Repos commit, pull request or branch to see the



Result:

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

EXP NO: 4

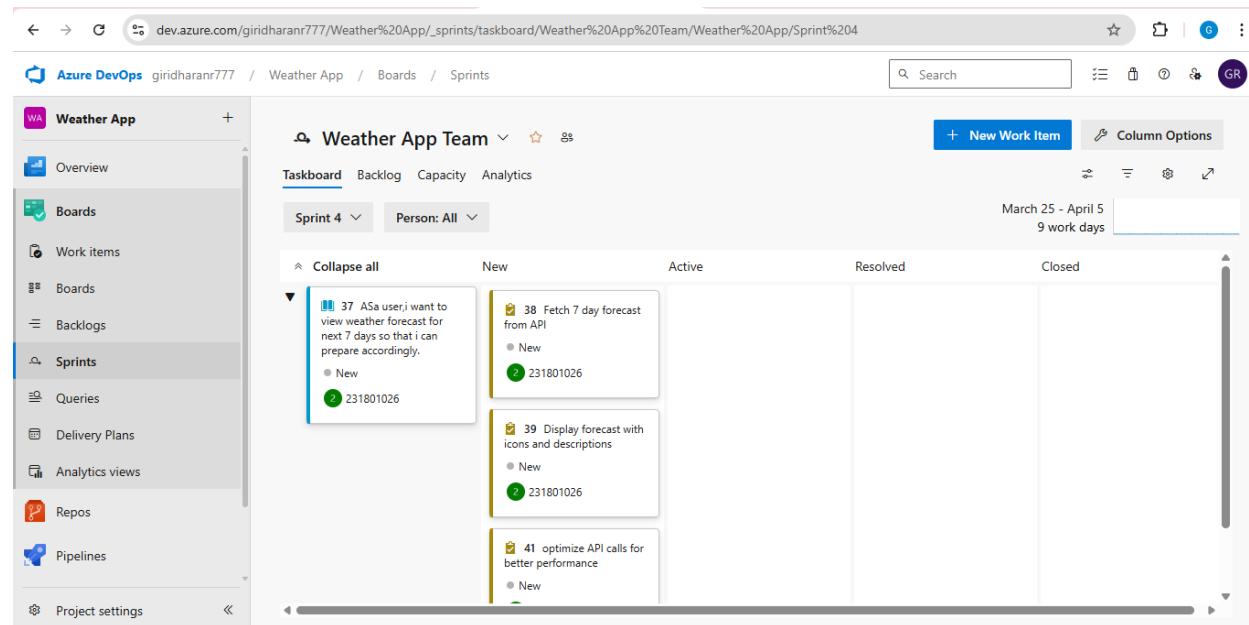
SPRINT PLANNING

Aim:

To assign user story to specific sprint for Weather App Project.

Sprint Planning

Sprint 1

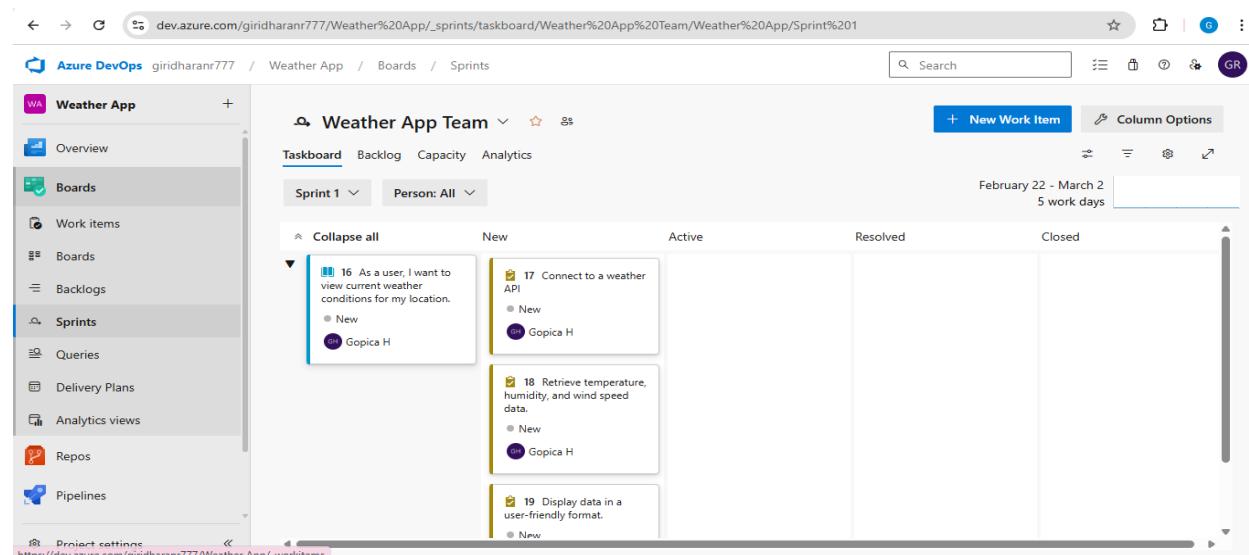


The screenshot shows the Azure DevOps Taskboard for the Weather App project. The sidebar on the left lists various project management sections like Overview, Boards, Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, and Pipelines. The main area displays the 'Weather App Team' taskboard for 'Sprint 4'. The backlog section shows three user stories:

- 37 As a user, I want to view weather forecast for next 7 days so that I can prepare accordingly. (New, 231801026)
- 38 Fetch 7 day forecast from API (New, 231801026)
- 39 Display forecast with icons and descriptions (New, 231801026)
- 41 Optimize API calls for better performance (New)

The taskboard also includes columns for New, Active, Resolved, and Closed tasks, and a summary at the bottom indicating the sprint duration from March 25 - April 5 (9 work days).

Sprint 2

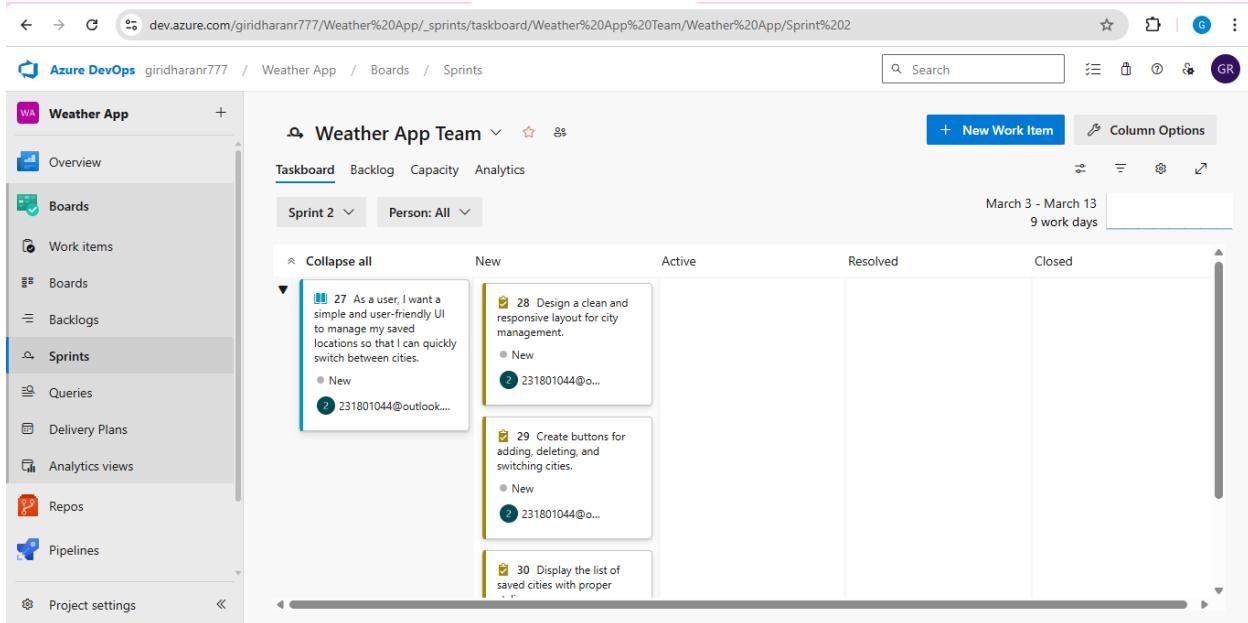


The screenshot shows the Azure DevOps Taskboard for the Weather App project. The sidebar on the left lists various project management sections like Overview, Boards, Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, and Pipelines. The main area displays the 'Weather App Team' taskboard for 'Sprint 1'. The backlog section shows four user stories:

- 16 As a user, I want to view current weather conditions for my location. (New, Gopica H)
- 17 Connect to a weather API (New, Gopica H)
- 18 Retrieve temperature, humidity, and wind speed data. (New, Gopica H)
- 19 Display data in a user-friendly format. (New)

The taskboard also includes columns for New, Active, Resolved, and Closed tasks, and a summary at the bottom indicating the sprint duration from February 22 - March 2 (5 work days).

Sprint 3



The screenshot shows the Azure DevOps Taskboard for the Weather App Team in Sprint 3. The backlog is organized into columns: New, Active, Resolved, and Closed. The 'New' column contains three items:

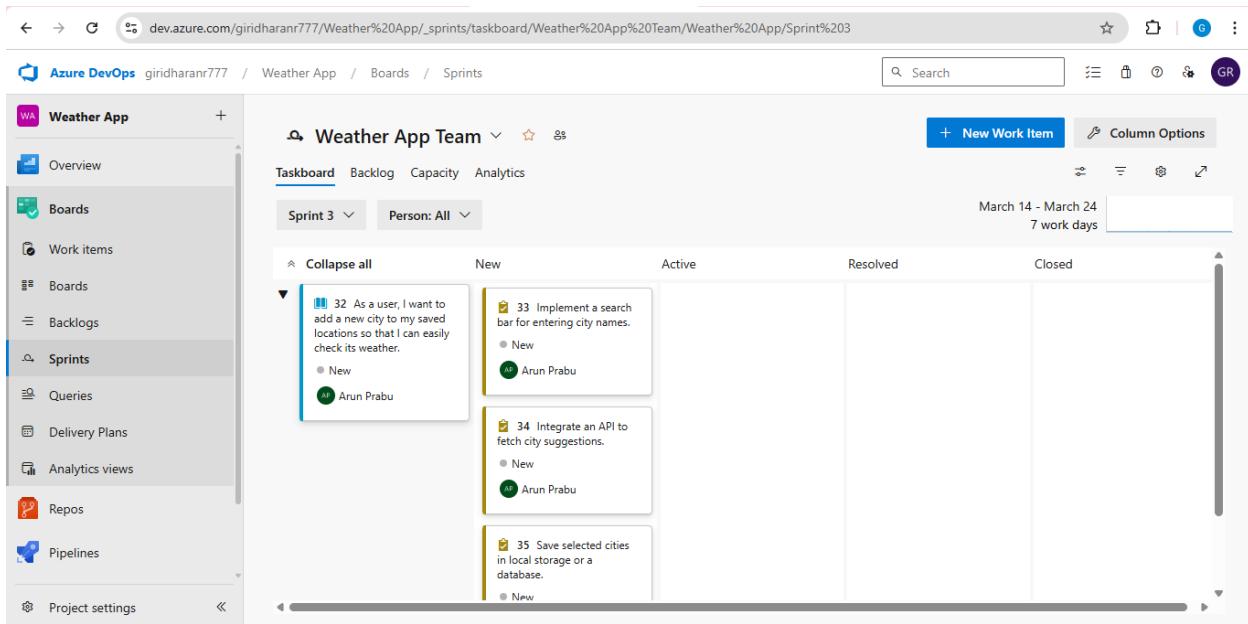
- Item 27: As a user, I want a simple and user-friendly UI to manage my saved locations so that I can quickly switch between cities. Status: New, Assigned to 231801044@outlook.com.
- Item 28: Design a clean and responsive layout for city management. Status: New, Assigned to 231801044@outlook.com.
- Item 29: Create buttons for adding, deleting, and switching cities. Status: New, Assigned to 231801044@outlook.com.

The 'Active' column contains one item:

- Item 30: Display the list of saved cities with proper status. Status: New, Assigned to 231801044@outlook.com.

The taskboard header shows the sprint period: March 3 - March 13, 9 work days.

Sprint 4



The screenshot shows the Azure DevOps Taskboard for the Weather App Team in Sprint 4. The backlog is organized into columns: New, Active, Resolved, and Closed. The 'New' column contains three items:

- Item 32: As a user, I want to add a new city to my saved locations so that I can easily check its weather. Status: New, Assigned to Arun Prabu.
- Item 33: Implement a search bar for entering city names. Status: New, Assigned to Arun Prabu.
- Item 34: Integrate an API to fetch city suggestions. Status: New, Assigned to Arun Prabu.

The 'Active' column contains one item:

- Item 35: Save selected cities in local storage or a database. Status: New, Assigned to Arun Prabu.

The taskboard header shows the sprint period: March 14 - March 24, 7 work days.

Result:

The Sprints are created for the Weather App Project.

EXP NO: 5

POKER ESTIMATION

Aim:

Create Poker Estimation for the user stories Weather App Project.

Poker Estimation

The screenshot shows the Azure DevOps interface for a User Story titled "USER STORY 37". The story description is: "37 AS a user,i want to view weather forecast for next 7 days so that i can prepare accordingly." The story is assigned to "Giridharan R". The "Planning" section shows Story Points as 3, Priority as 2, and Risk as 1. The "Classification" section indicates the value area is Business. The "Development" section provides a link to an Azure Repos repository. The sidebar on the left shows the project structure: Weather App > Boards > Backlogs. The top navigation bar includes links for Boards, Backlogs, Epics, and Work items.

Result:

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

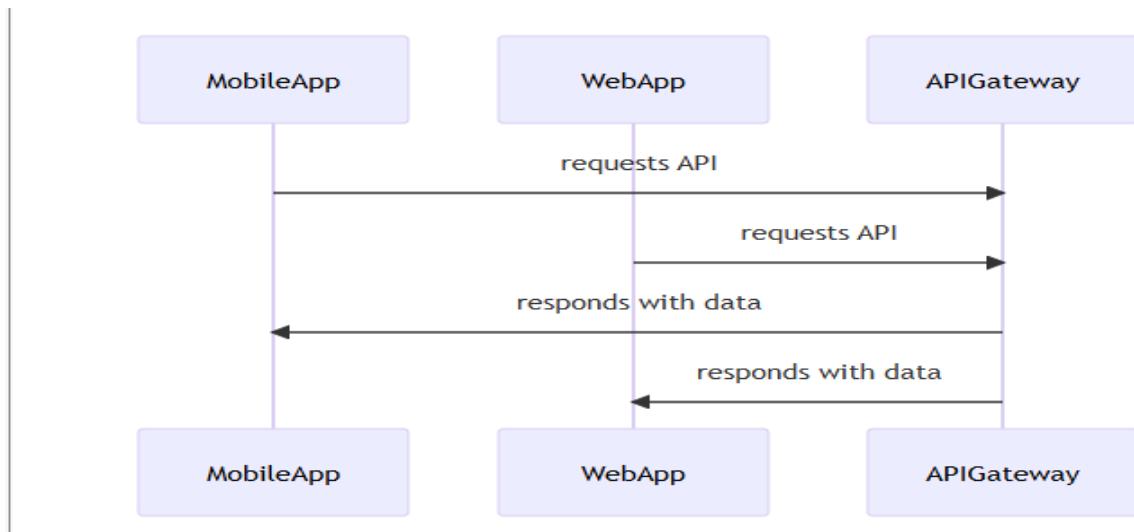
EXP NO: 6

DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE

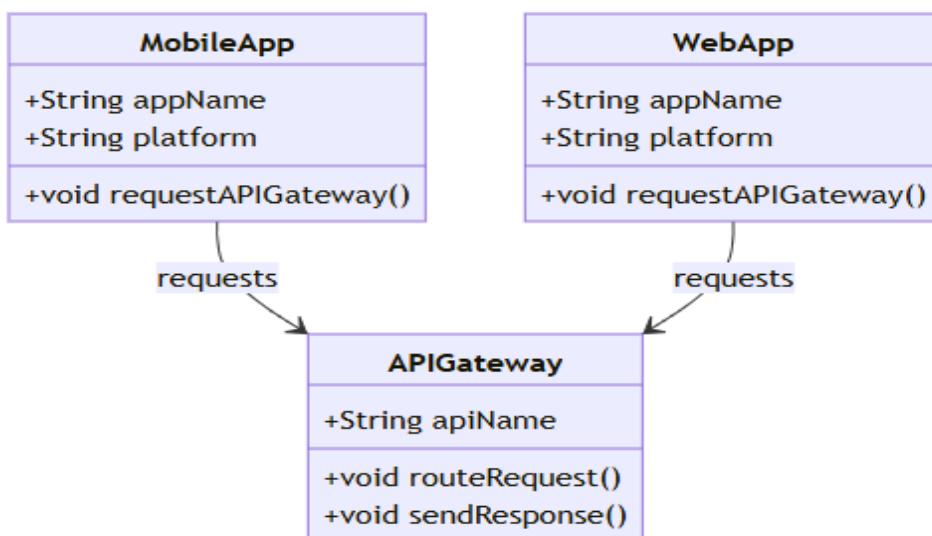
Aim:

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 Weather App.

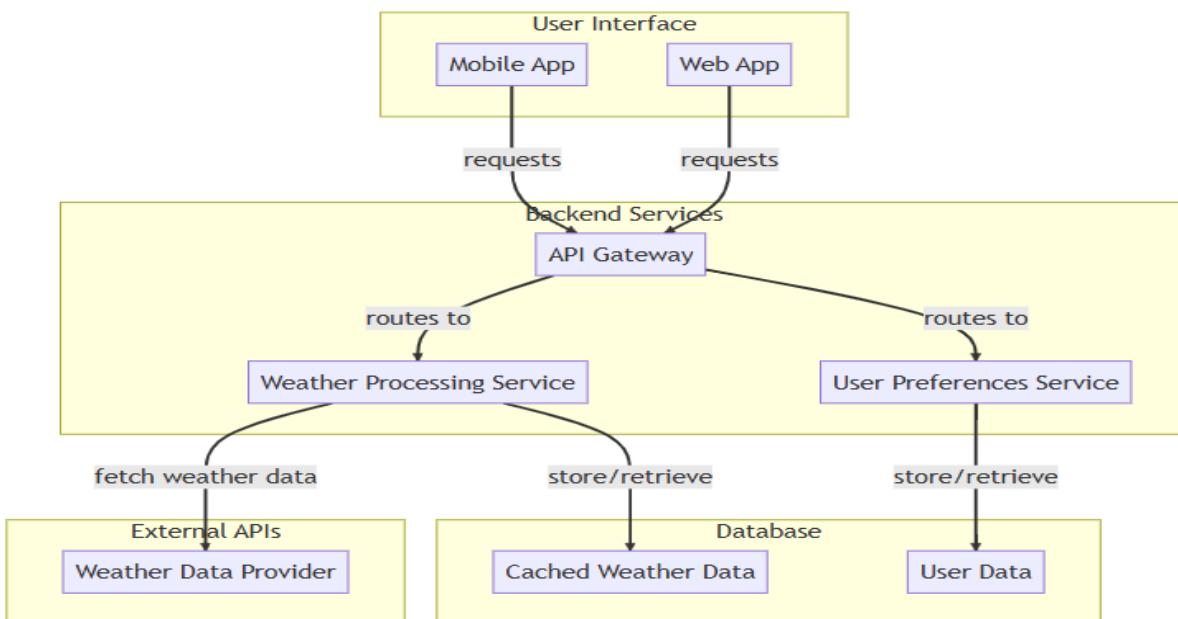
EXP NO: 7

DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE

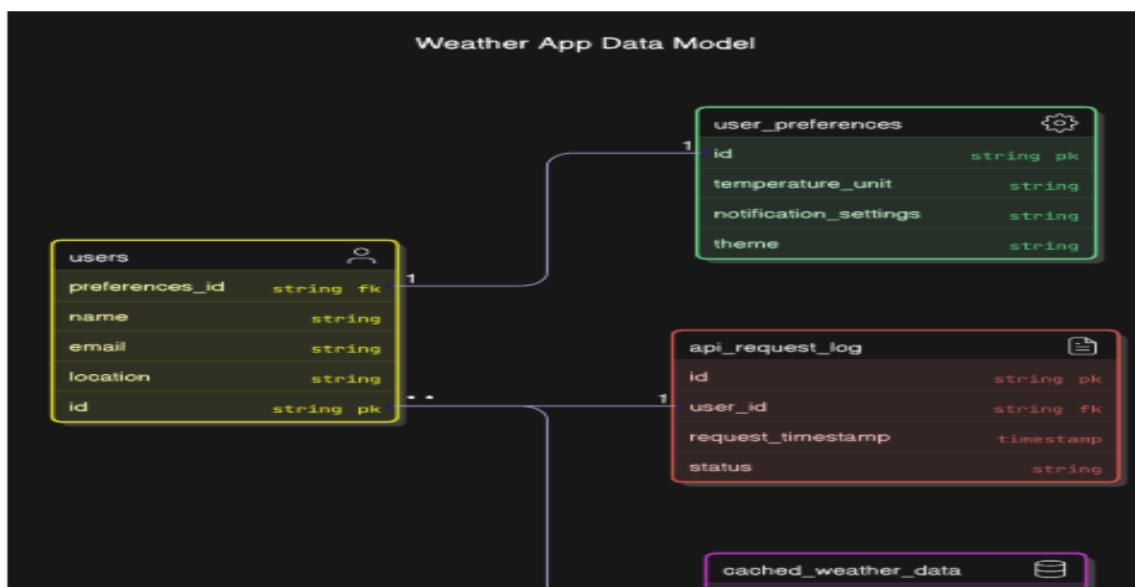
Aim:

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 Weather App.

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. Define User Interactions

User Signup & Login

Viewing and Managing Favorite Locations

Fetching Real-time Weather Data

Editing Weather Dashboards (rename, reorder, customize)

Creating Smart Weather Alerts based on conditions (temperature, humidity, location, etc.)

2. Design Happy Path Test Cases

Focused on validating that all features function as expected under normal conditions.

Example: User logs in successfully, checks the current weather for a city, or saves a favorite city for quick access.

3. 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.

4. 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.

5. Use Clear Naming and IDs

Test cases are named clearly (e.g., TC01 – Successful Login, TC10 – Forecast).

Helps in quick identification and linking to user stories or features.

6. Separate Test Suites

- Grouped test cases based on functionality (e.g., Login, Current Weather , Recommendation System).
- Improves organization and test execution flow in Azure DevOps.

7. Prioritize and Review

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

1.New test plan

The screenshot shows the 'New Test Plan' creation interface in Azure DevOps. The left sidebar shows the 'Test Plans' section is selected. The main form has the following fields filled:

- Name ***: Weather APP test Plans
- Area Path ***: Weather App
- Iteration ***: Weather App

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

2.Test suite

The screenshot shows the 'Actual Test Plans' execution interface in Azure DevOps. The left sidebar shows the 'Test plans' section is selected. The main area displays the 'Display (ID: 75)' test suite. The 'Execute' tab is active. The 'Test Points (3 items)' table is shown below:

Title	Outcome	C
Launch app and verify logo appears	Passed	1
Check that the current temperature is shown	Passed	2
Verify weather icon (e.g., sun, cloud) is visible	Failed	3

3.Test case

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

Weather App– Test Plans

USER STORIES

- As a user, I want to sign up and log in securely so that I can access my saved cities and settings (ID: 79).
- As a user, I need to see all my saved cities' weather in one place (ID: 76).
- As a user, I should be able to add a new city to my favorites list as needed (ID: 73).
- As a user, I should be able to rename, remove, or reorder my saved cities (ID: 68).
- As a user, I need to have real-time weather updates and alerts (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 Weather 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 Weather 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 Saved Cities (ID: 87)

1. **TC05 – View Saved Cities Page**
 - **Action:**
 - Log in successfully.
 - Navigate to "My Cities" section.
 - **Expected Results:**
 - All saved cities and their current weather data are displayed clearly.
 - **Type:** Happy Path
 2. **TC06 – Saved Cities Loading Failure**
 - **Action:**
 - Disconnect from the internet.
 - Navigate to "My Cities".
 - **Expected Results:**
 - Network is offline.
 - Error message "Unable to load saved cities" is shown.
 - **Type:** Error Path
-

Test Suit: TS03 - Real-Time Weather Updates (ID: 88)

1. **TC07 – Real-Time Weather Display**
 - **Action:**
 - Open a city's weather page.
 - Observe the weather data and updates.
 - **Expected Results:**
 - Weather data (temperature, humidity, wind speed, etc.) is displayed and updates in real time.
 - **Type:** Happy Path
 2. **TC08 – Weather Data Not Updating**
 - **Action:**
 - Open a city's weather page.
 - Simulate server delay or offline mode.
 - **Expected Results:**
 - Weather data remains static or shows fallback/default message.
 - **Type:** Error Path
-

Test Suit: TS04 - Manage Saved Cities (ID: 89)

1. **TC09 – Rename Saved City Successfully**
 - **Action:**
 - Navigate to "My Cities".
 - Click "Rename" next to a saved city.
 - Enter a new name and click "Save".
 - **Expected Results:**
 - City name updates successfully.
 - **Type:** Happy Path
2. **TC10 – Rename City with Blank Name**
 - **Action:**
 - Click "Rename" on a saved city.

- Leave the field blank.
 - Click "Save".
 - **Expected Results:**
 - Error message "City name cannot be empty" is shown.
 - **Type:** Error Path
- 3. TC11 – Change Cities Order**
- **Action:**
 - Open "My Cities".
 - Drag and drop cities to reorder.
 - Click "Save".
 - **Expected Results:**
 - Saved cities' order is updated and saved.
 - **Type:** Happy Path
- 4. TC12 – Change Cities Order Fails**
- **Action:**
 - Login and go to "My Cities".
 - Select a city.
 - Go offline or simulate server error.
 - Reorder cities and click "Save Order".
 - **Expected Results:**
 - Error message: "Failed to update order. Please check your connection".
 - **Type:** Error Path

Test Suit: TS05 - Smart Weather Alerts (ID: 90)

- 1. TC13 – Generate Weather Alerts Based on Conditions**
- **Action:**
 - Login with valid credentials.
 - Click on "Set Weather Alerts".
 - Select conditions like rain, storm, or temperature threshold.
 - Click "Save Alerts".
 - **Expected Results:**
 - Weather alerts are generated and saved based on selected conditions.
 - **Type:** Happy Path
- 2. TC14 – Fail to Set Weather Alerts Due to Missing Selection**
- **Action:**
 - Login with valid credentials.
 - Click on "Set Weather Alerts".
 - Leave alert conditions blank or select invalid options.
 - Click "Save Alerts".
 - **Expected Results:**
 - Error message: "Please select at least one valid alert condition".
 - **Type:** Error Path

Test Cases

dev.azure.com/giridharan777/Weather%20App/_testPlans/execute?planId=60&suitId=75

Azure DevOps giridharan777 / Weather App / Test Plans / Actual Test Plans

TEST CASE 76

76 Launch app and verify logo appears

0 Comments Add Tag

State: Design Area: Weather App
Reason: New Iteration: Weather App/Sprint 4

Steps Summary Associated Automation

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.

Steps

Steps	Action	Expected result
1.	City name and weather info change to reflect new location	App launches without crashing
2.	Observe the splash screen or header	App logo appears during splash or at to screen
3.	Wait for main UI to load	Main screen loads with logo still visible with weather data

Click or type here to add a step

Project settings

dev.azure.com/giridharan777/Weather%20App/_testPlans/execute?planId=60&suitId=75

Azure DevOps giridharan777 / Weather App / Test Plans / Actual Test Plans

TEST CASE 76

76 Launch app and verify logo appears

0 Comments Add Tag

State: Design Area: Weather App
Reason: New Iteration: Weather App/Sprint 4

Steps Summary Associated Automation

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.

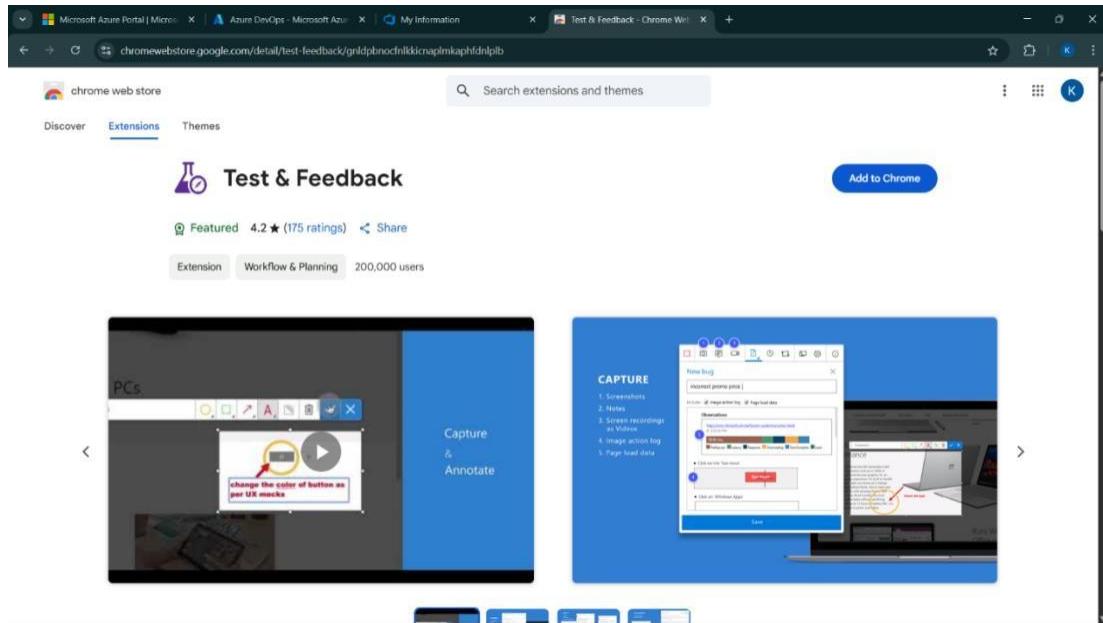
Steps

Steps	Action	Expected result
1.	City name and weather info change to reflect new location	App launches without crashing
2.	Observe the splash screen or header	App logo appears during splash or at to screen
3.	Wait for main UI to load	Main screen loads with logo still visible with weather data

Click or type here to add a step

Project settings

4. Installation of test



Test and feedback

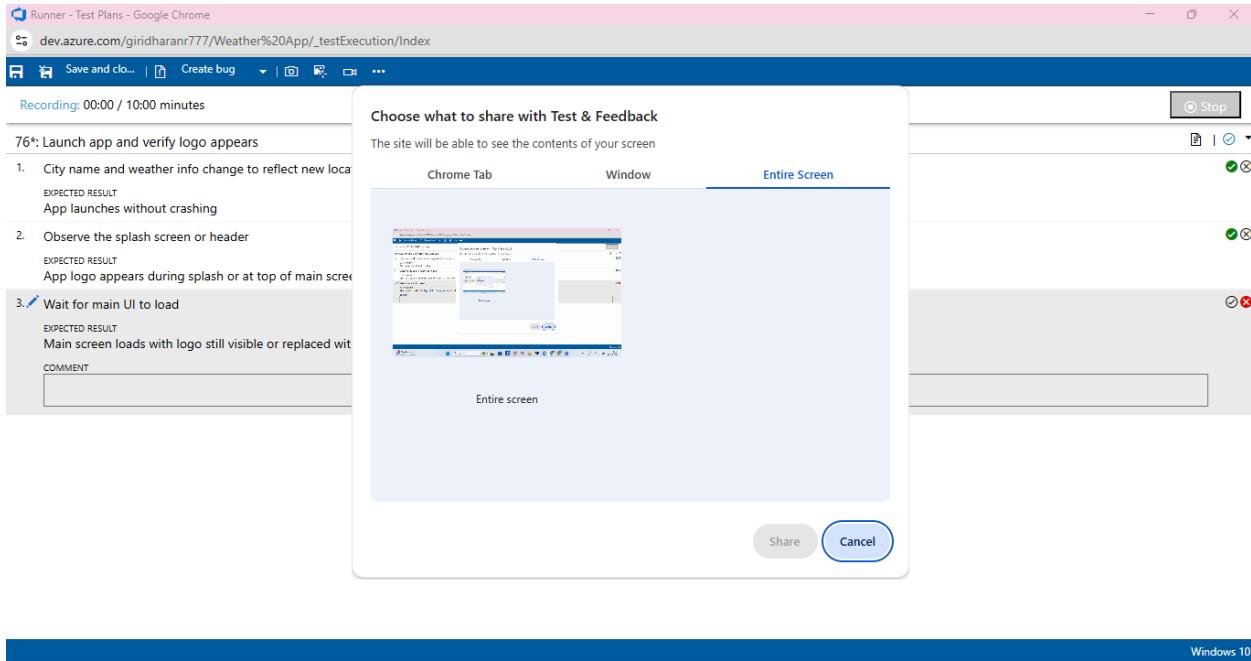
Showing it as an extension

A screenshot of the Azure DevOps Test Plans interface for a "Weather App". The left sidebar shows navigation options like Overview, Boards, Repos, Pipelines, Test Plans, and Artifacts. The "Test plans" option is selected. In the center, a "Display (ID: 75)" card is shown with tabs for Define, Execute, and Chart. The "Execute" tab is active, showing a search bar and a "Test Points (3 items)" table. The table lists three test points: "Launch app and verify logo appears" (Passed), "Check that the current temperature is shown" (Passed), and "Verify weather icon (e.g., sun, cloud) is visible" (Failed). On the right side, a floating "Extensions" pane is open, showing the "Test & Feedback" extension listed under "Full access". The extension details state: "These extensions can see and change information on this site." It lists two other extensions: "McAfee® WebAdvisor" and "Test & Feedback". A "Manage extensions" button is also present in the pane.

5. Running the test cases

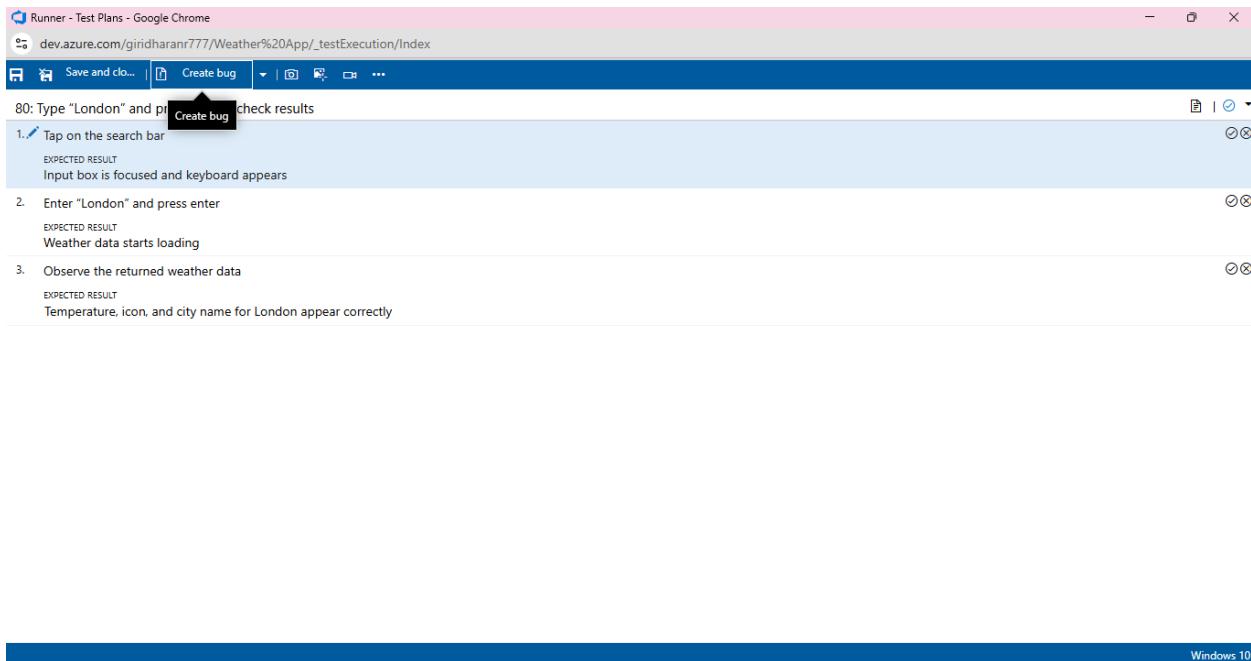
The screenshot shows the Azure DevOps Test Plans interface for a 'Weather App' project. The left sidebar navigation bar includes 'Overview', 'Boards', 'Repos', 'Pipelines', 'Test Plans' (selected), 'Progress report', 'Parameters', 'Configurations', 'Runs', and 'Artifacts'. The main content area displays 'Actual Test Plans' for 'Display' (ID: 75) from March 25 - April 5, showing a 100% run with 75% passed. A 'View report' link is present. The 'Test Suites' section lists 'Forecast' (3), 'Current Weather' (3), 'Location Search' (3), and 'Display' (3). The 'Display' suite is selected. The right side shows the 'Test Points (3 items)' for the 'Display' suite. One test point, 'Launch app and verify logo appears', is selected and has its details expanded. Other options like 'Run for web application', 'Run for desktop application', and 'Run with options' are also visible. Below this, a detailed view of the first test point shows steps: 1. City name and weather info change to reflect new location (Expected Result: App launches without crashing); 2. Observe the splash screen or header (Expected Result: App logo appears during splash or at top of main screen); 3. Wait for main UI to load (Expected Result: Main screen loads with logo still visible or replaced with weather data). A comment field is available for each step.

6.Recording the test case



Windows 10

7.Creating the bug



Windows 10

Runner - Test Plans - Google Chrome
dev.azure.com/giridharan777/Weather%20App/_testExecution/Index

Save and close Create bug ...

80*: Type "London" and press search, check results

1. Tap
EXPEC Input
2. Enter
EXPEC Weather
COM...
3. Observe
EXPEC Tem...

NEW BUG *
TB-01

Unassigned 0 comments Add tag Save & Close ...

State: New Area: Weather App
Reason: New Iteration: Weather App\Sprint 4

1. **Passed** Tap on the search bar
Expected Result: Input box is focused and keyboard appears

2. **Failed** Enter "London" and press enter
Expected Result: Weather data starts loading

3. **Passed** Observe the returned weather data

Priority: 2 Severity: 3 - Medium Activity:

Effort (Hours): Original Estimate Remaining Completed

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

Windows 10

Runner - Test Plans - Google Chrome
dev.azure.com/giridharan777/Weather%20App/_testExecution/Index

NEW BUG *
TB-01

Unassigned 0 comments Add tag Save & Close ...

State: New Area: Weather App
Reason: New Iteration: Weather App\Sprint 4

System Info: Found in Build Integrated in Build

Browser - Name	Google Chrome 135
Browser - Language	en-US
Browser - Height	736
Browser - Width	1382
Browser - User agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/135.0.0.0 Safari/537.36
Operating system - Name	Windows NT 10.0; Win64; x64
Operating system - Architecture	x86_64
Operating system - Processor model	Intel(R) Celeron(R) N4020 CPU @ 1.10GHz
Operating system - Number of processors	2
Memory - Available	1846099968
Memory - Capacity	8214425600
Display - Pixels per inch (X axis)	96
Display - Pixels per inch (Y axis)	96
Display - Device pixel ratio	1

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

8.Test case results

The screenshot shows the Azure DevOps Test Plans interface. On the left, the navigation bar includes 'Weather App', 'Overview', 'Boards', 'Repos', 'Pipelines', 'Test Plans', 'Test plans' (selected), 'Progress report', 'Parameters', 'Configurations', 'Runs', and 'Artifacts'. Under 'Test plans', 'Actual Test Plans' is selected, showing a summary for 'Mar 25 - Apr 5' with '100% run, 75% passed.' and a link to 'View report'. The main area displays 'Test Suites' with a filter for 'Actual Test Plans' showing 'Forecast (3)', 'Current Weather (3)', 'Location Search (3)' (selected), and 'Display (3)'. To the right, a modal window titled 'Test Case Results' shows a table with two rows:

Outcome	TimeStamp	Configuration	Run by	Tester	Test Plan
Passed	Saturday	Windows 10	Giridharan R	Giridharan R	Actual
Passed	Saturday	Windows 10	Giridharan R	Giridharan R	Actual

9. Test report summary

The screenshot shows the Azure DevOps Test Execution interface. A bug entry for 'TB-01' is displayed, titled 'NEW BUG *'. The bug details include:

- Title:** TB-01
- Area:** Weather App
- Iteration:** Weather App\Sprint 4
- State:** New
- Reason:** New
- Priority:** 2
- Severity:** 3 - Medium
- Activity:** Unassigned
- Effort (Hours):** Original Estimate, Remaining, Completed
- Development:** Details, 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

The test steps for 'TB-01' are listed as follows:

- 1. Passed**: Tap on the search bar
Expected Result: Input box is focused and keyboard appears
- 2. Failed**: Enter "London" and press enter
Expected Result: Weather data starts loading
- 3. Passed**: Observe the returned weather data

- Assigning bug to the developer and changing state

The screenshot shows the Azure Test Plan details page for a bug titled "TB-01". The bug was filed on 4/22/2025 at 3:17 PM with the title "Bug filed on 'Type 'London'' and press search, check results". The steps are listed as follows:

- Step no.**: 1, **Result**: Passed, **Title**: Tap on the search bar. Expected Result: Input box is focused and keyboard appears.
- Step no.**: 2, **Result**: Failed, **Title**: Enter "London" and press enter. Expected Result: Weather data starts loading.
- Step no.**: 3, **Result**: Passed, **Title**: Observe the returned weather data.

Planning section includes Resolved Reason, Story Points, Priority (2), Severity (3 - Medium), Activity, and Effort (Hours) (Original Estimate, Remaining, Completed).

Deployment section includes a note about tracking releases and deployment status reporting.

Development section includes a note about linking to Azure Repos commits or branches.

Related Work section includes an option to add an existing work item as a parent.

Tested By section lists 80 users who tested the bug.

10. Progress report

The screenshot shows the Azure DevOps Progress report for the Weather App. The left sidebar shows navigation options like Overview, Boards, Repos, Pipelines, Test Plans, Progress report (which is selected), Parameters, Configurations, Runs, Artifacts, and Project settings.

The main area displays the following data:

- Actual Test Plans**: 1 Test plan, 12 Test points.
- Summary**: 12 (12 / 12) Test points run (100% Run).
- Outcome trend**: A chart showing the number of tests over time from April 8 to April 22, with a sharp increase starting around April 18. The legend indicates green for Passed and red for Failed.
- Pass rate**: 75% (9 / 12) with 9 Passed and 3 Failed.

11. Changing the test template

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

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

The screenshot shows the same 'All processes' section, but the 'Agile' template is now selected. The 'Agile' row is highlighted with a blue background. The other rows have a light gray background.

The screenshot shows the 'All processes' section of the Azure DevOps Settings - Process page. The 'Agile' template is selected. The table lists the following process templates:

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

12. View the new test case template

The screenshot shows the 'Add a field to Test Case' dialog box. The 'Create a field' option is selected, and a new field named 'Acceptance Criteria' is being created with a type of 'Text (single line)'. The dialog also includes a description field and a 'Learn more' link.

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

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

Azure DevOps 231801095 / Settings / Process

All processes > 231801095 Agile

Work-item types Backlog levels Projects

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

Organization Settings... 231801095

General

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

Security

- Security overview
- Policies
- Permissions

Boards

- Process

Pipelines

- Agent pools
- Settings
- Deployment pools

Search Settings

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

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

Azure DevOps 231801095 / Settings / Process

All processes > 231801095 Agile > Test Case

Layout States Rules

New field New group New page Get extensions

Steps Summary Associated Aut...

Custom

Type Text (single line)

Recent test results

Recent test case results

Deployment

Deployments

Development

Links

Related Work

Links

Status

Priority Integer

Automation status

Add a field ...

Organization Settings... 231801095

General

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

Security

- Security overview
- Policies
- Permissions

Boards

- Process

Pipelines

- Agent pools
- Settings
- Deployment pools

Search Settings

Result:

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

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.

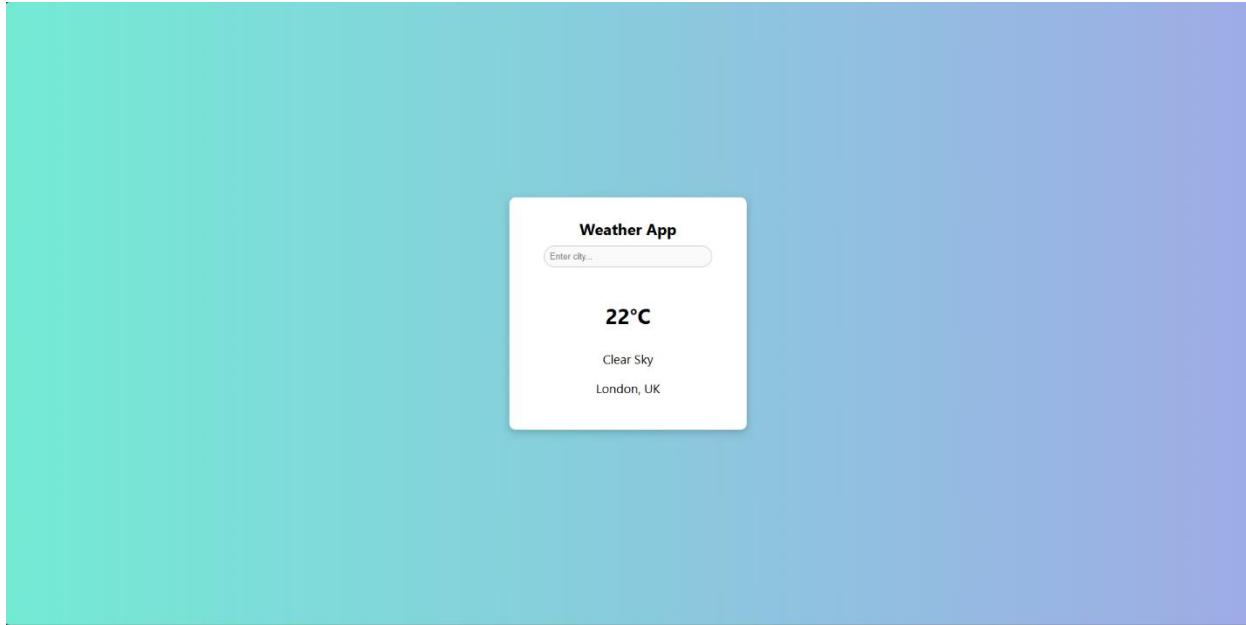
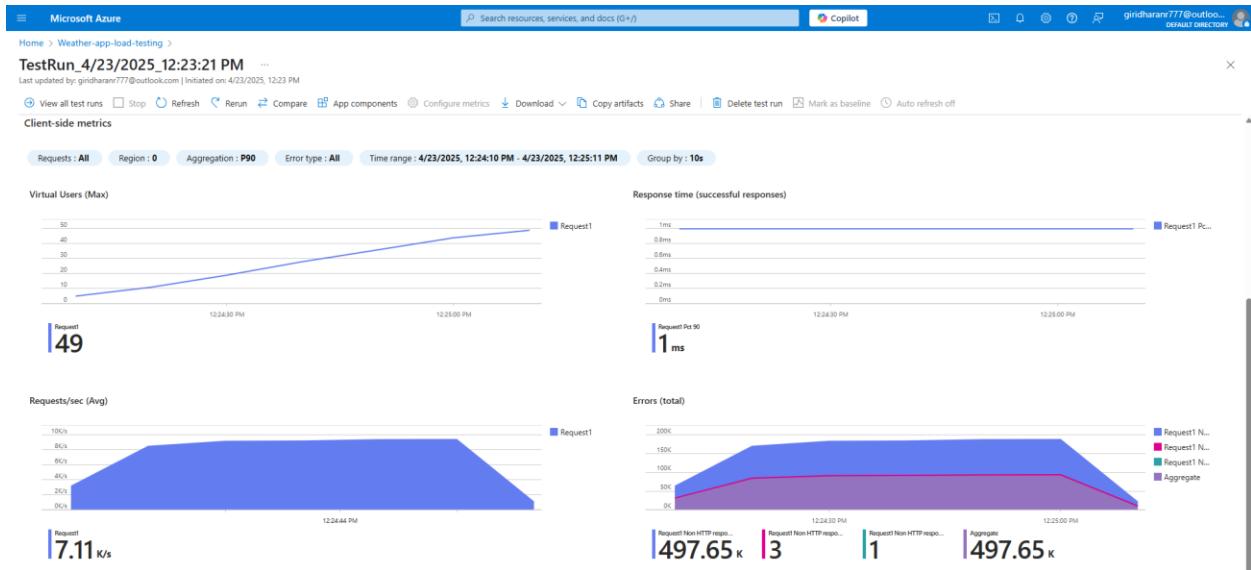
Steps to Create and Run a Load Test:

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 Weather App 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('Weather App')"
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. On the left, there's a sidebar with navigation links: Weather App (selected), Overview, Boards, Repos, Pipelines (selected), Pipelines, Environments, Library, Test Plans, and Artifacts. The main area is titled 'Pipelines' and shows a table for 'Recently run pipelines'. The table has columns for 'Pipeline', 'Last run', and 'Status'. One row is visible: 'Giri7772005040.Weather-App' was run on Wednesday at approximately 1 second ago. A 'New pipeline' button is located in the top right corner of the main area.

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.

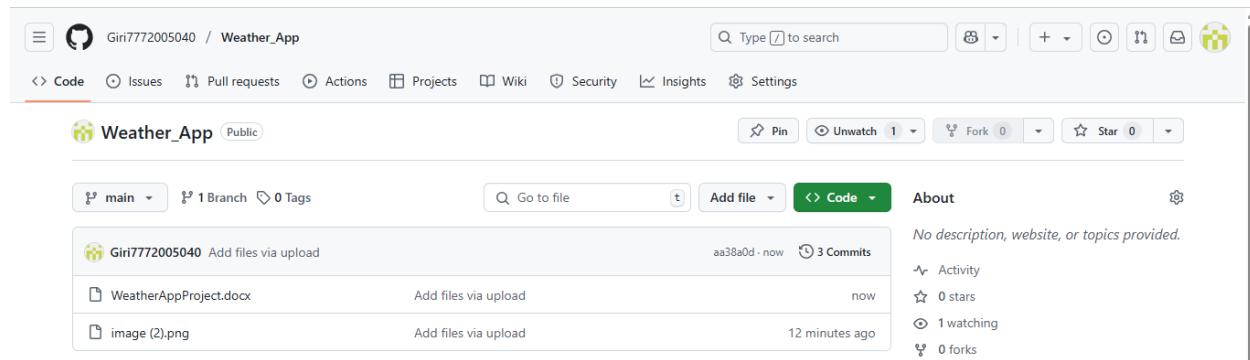
EXP NO: 10

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 Weather App.

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.