EXP NO: 1	AZURE DEVOPS ENVIRONMENT SETUP

Aim:

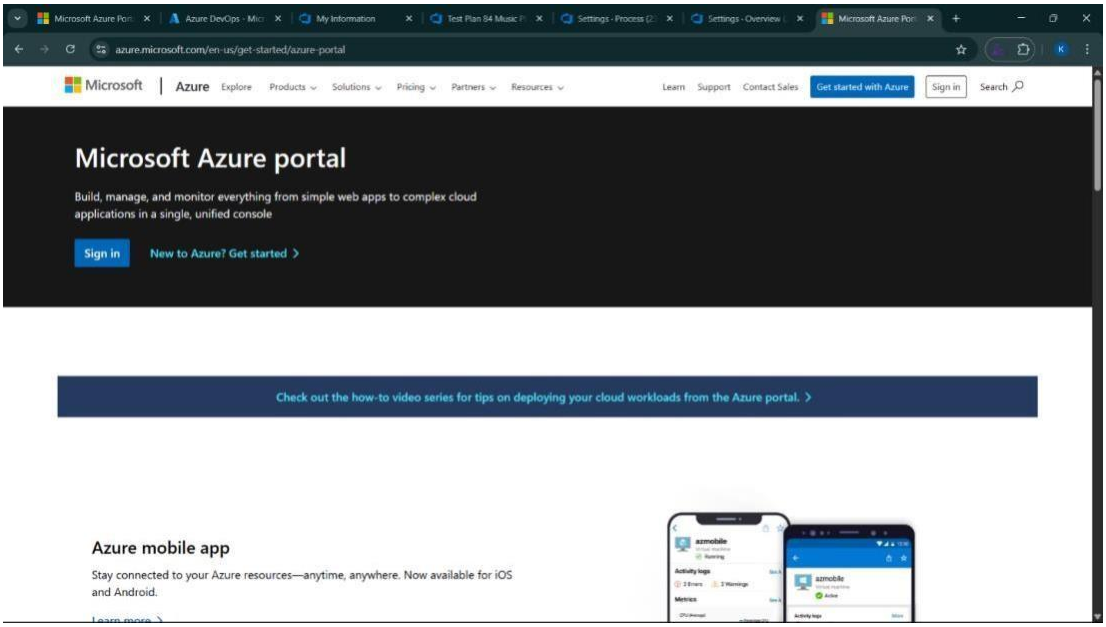
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

INSTALLATION

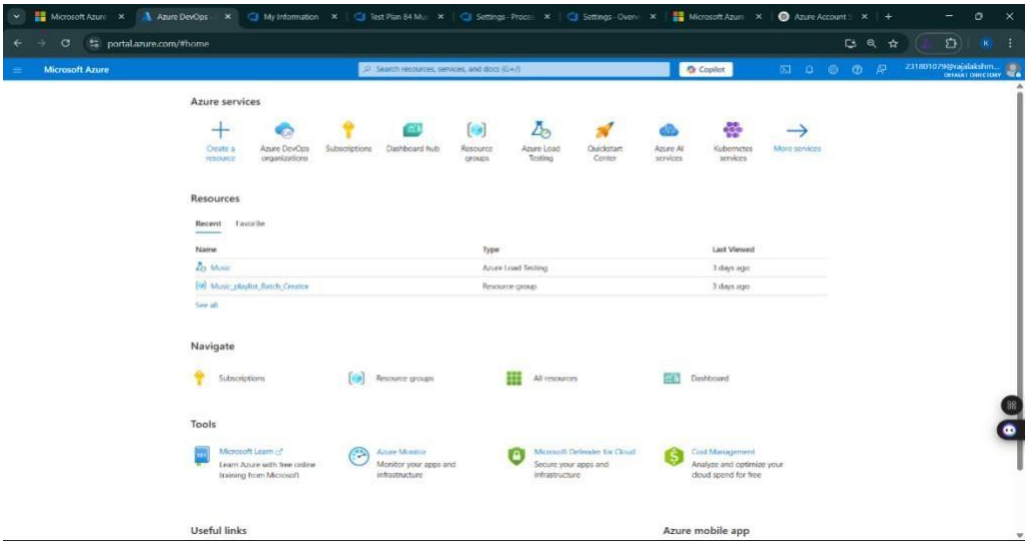
1. Open your web browser and go to the Azure website: https://azure.microsoft.com/en-us/get-started/azure-portal.

Sign in using your Microsoft account credentials.

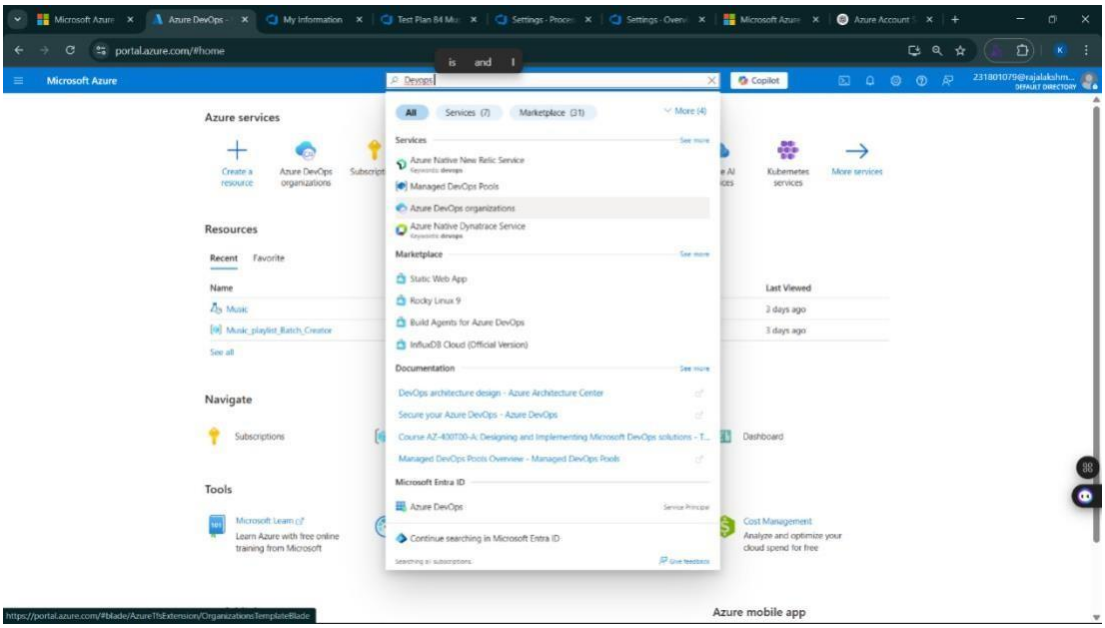
If you don't have a Microsoft account, you can create one here: https://signup.live.com/?lic=1



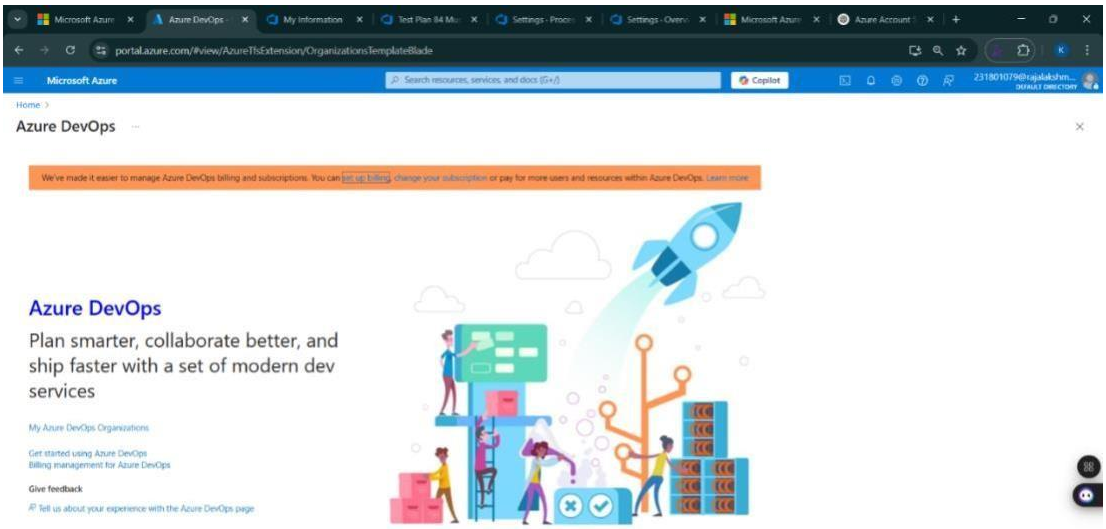
2. Azure home page



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



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:

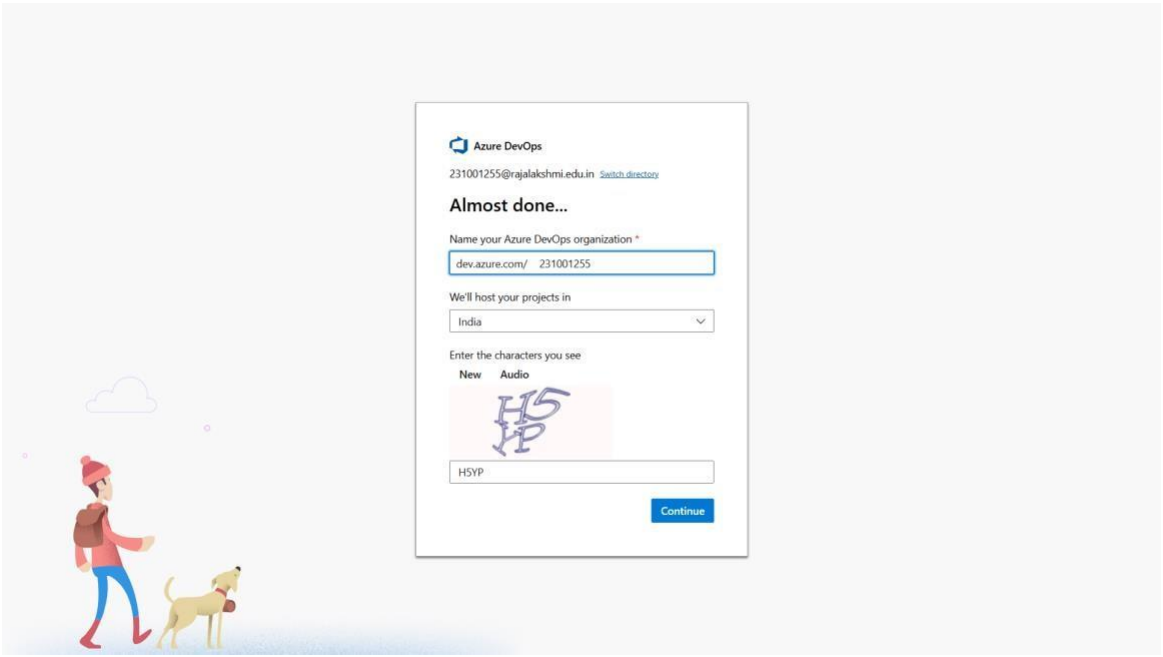
Successfully accessed the Azure DevOps environment and created a new organization through the Azure portal.

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.

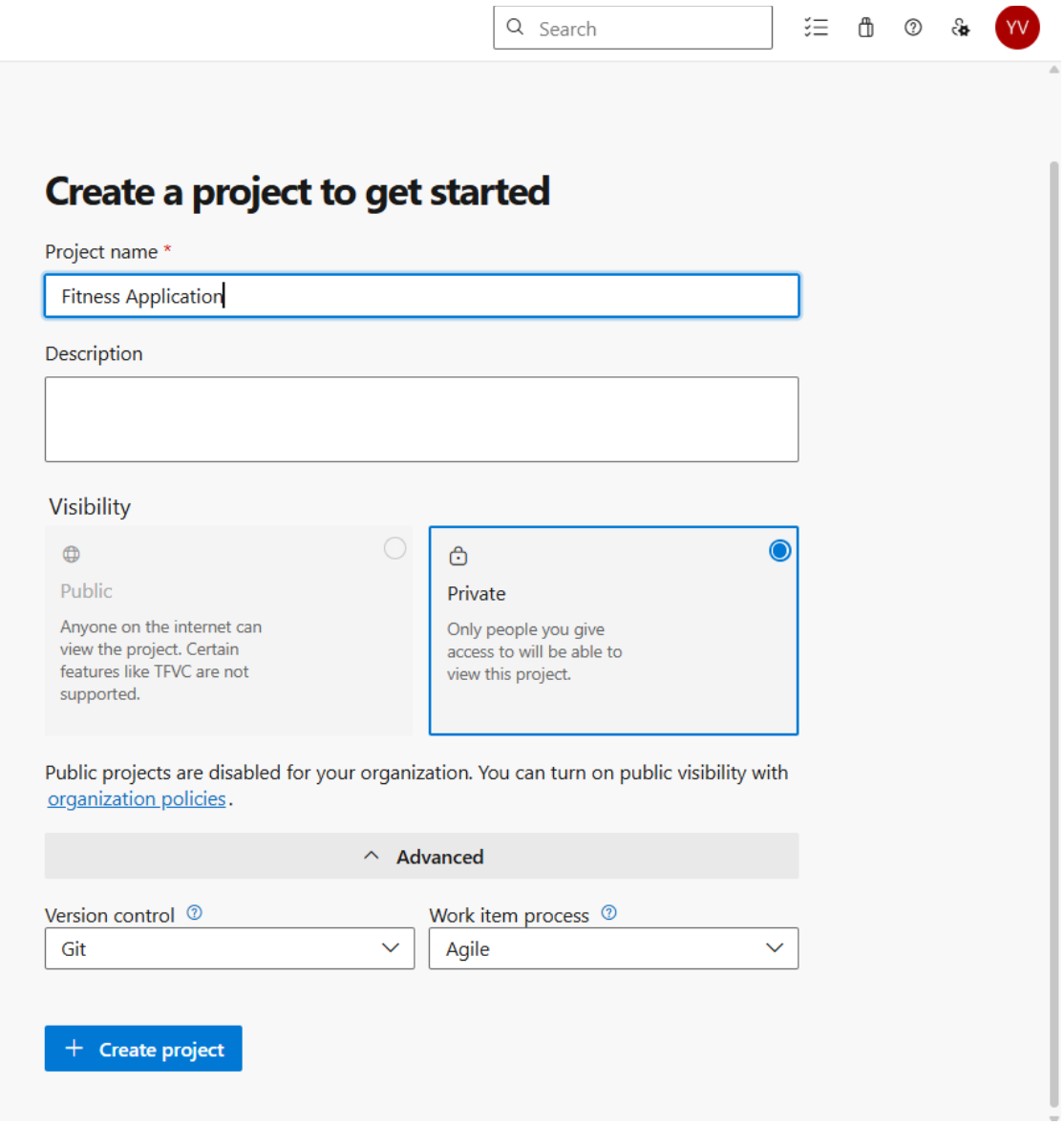
c. Enter the project name, description, and visibility options:

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

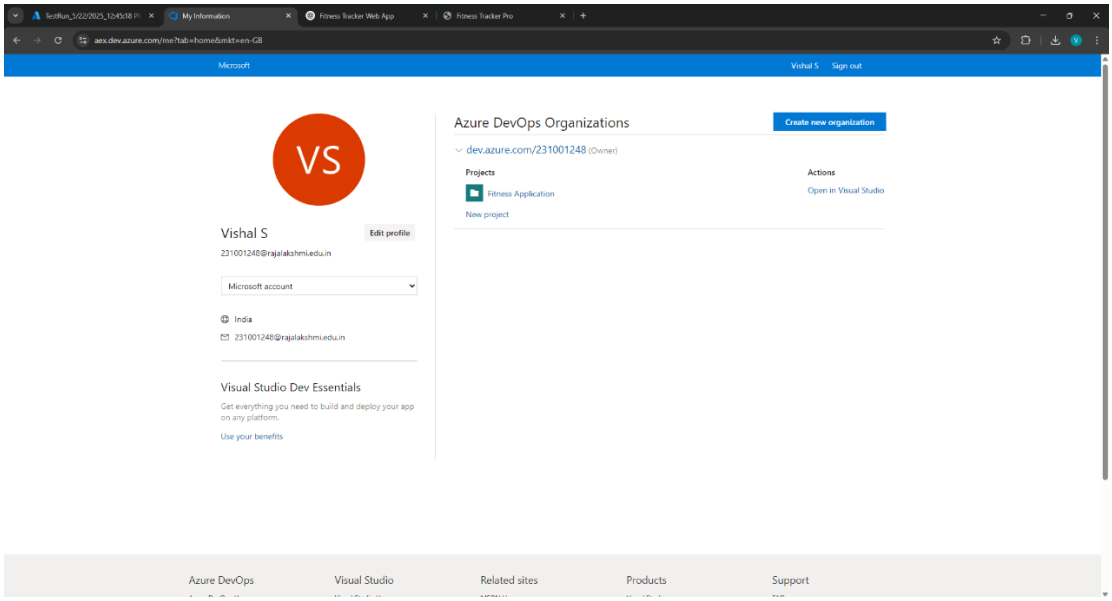
Description: Optionally, add a description to provide more context about the project.

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

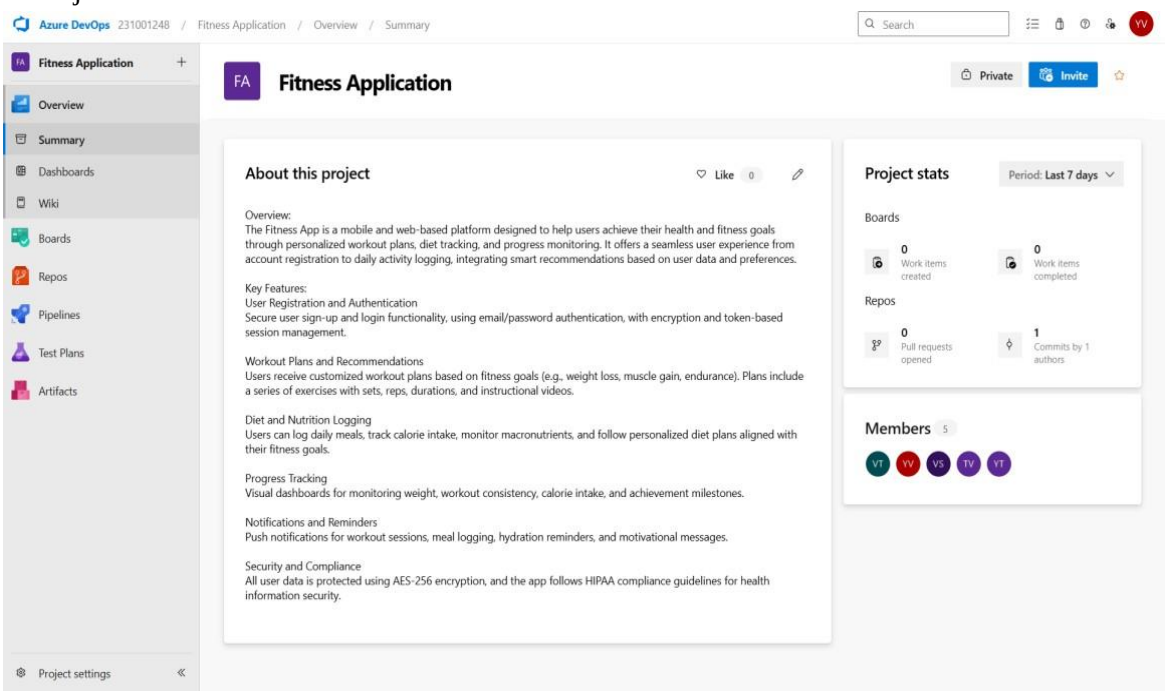
d. Once you've filled out the details, click **Create** to set up your first project.



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



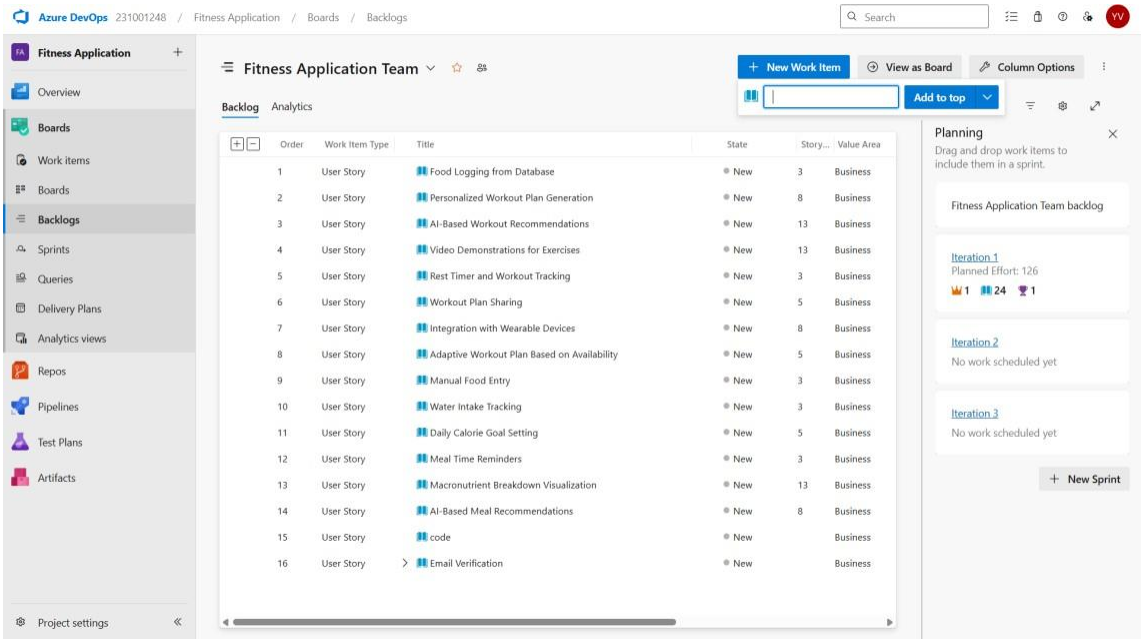
4. Project dashboard

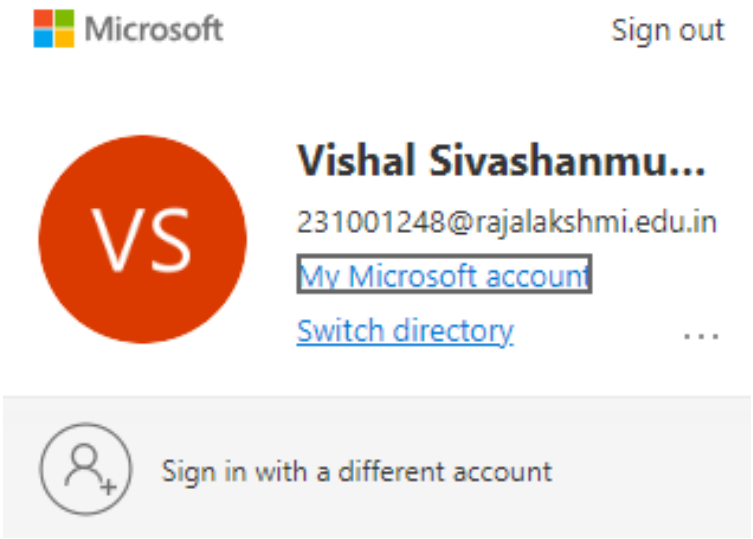


5. To manage user stories:

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.





Result:

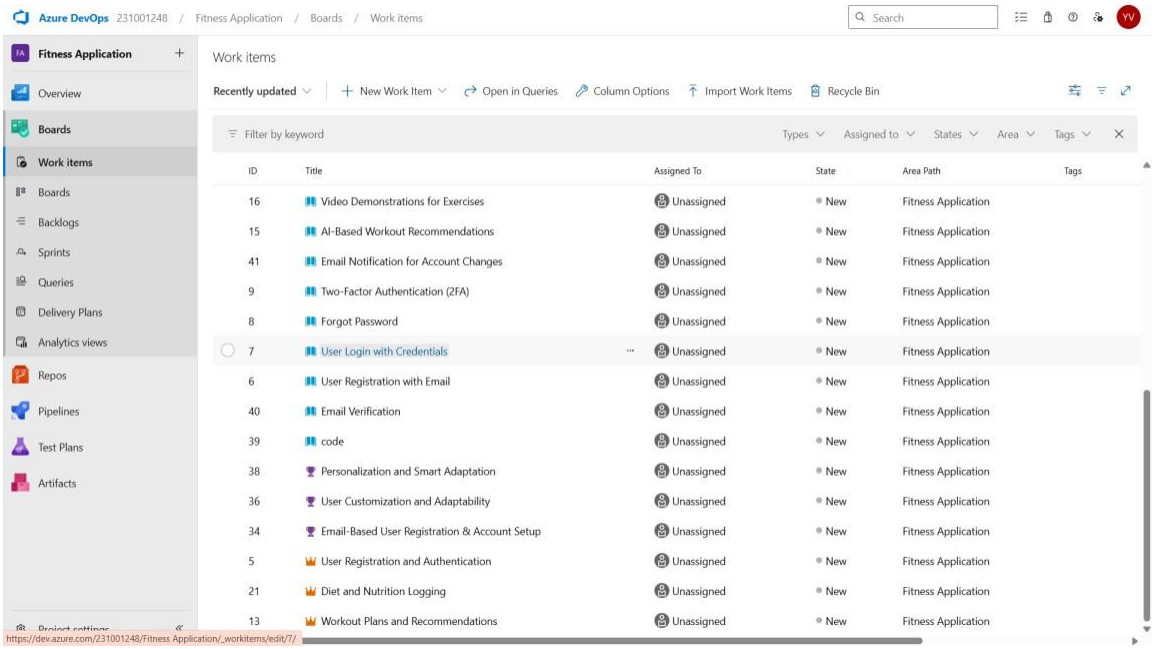
Successfully created an Azure DevOps project with user story management and agile workflow setup.

EXP NO: 3	SETTING UP EPICS, FEATURES, AND USER STORIES FOR PROJECT PLANNING

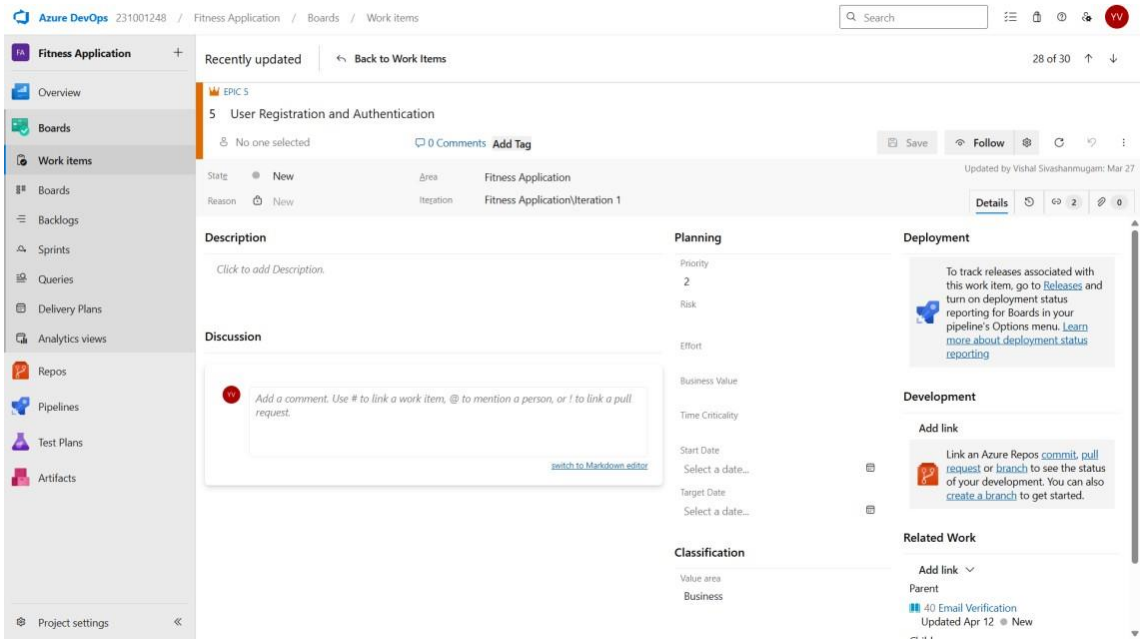
Aim:

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

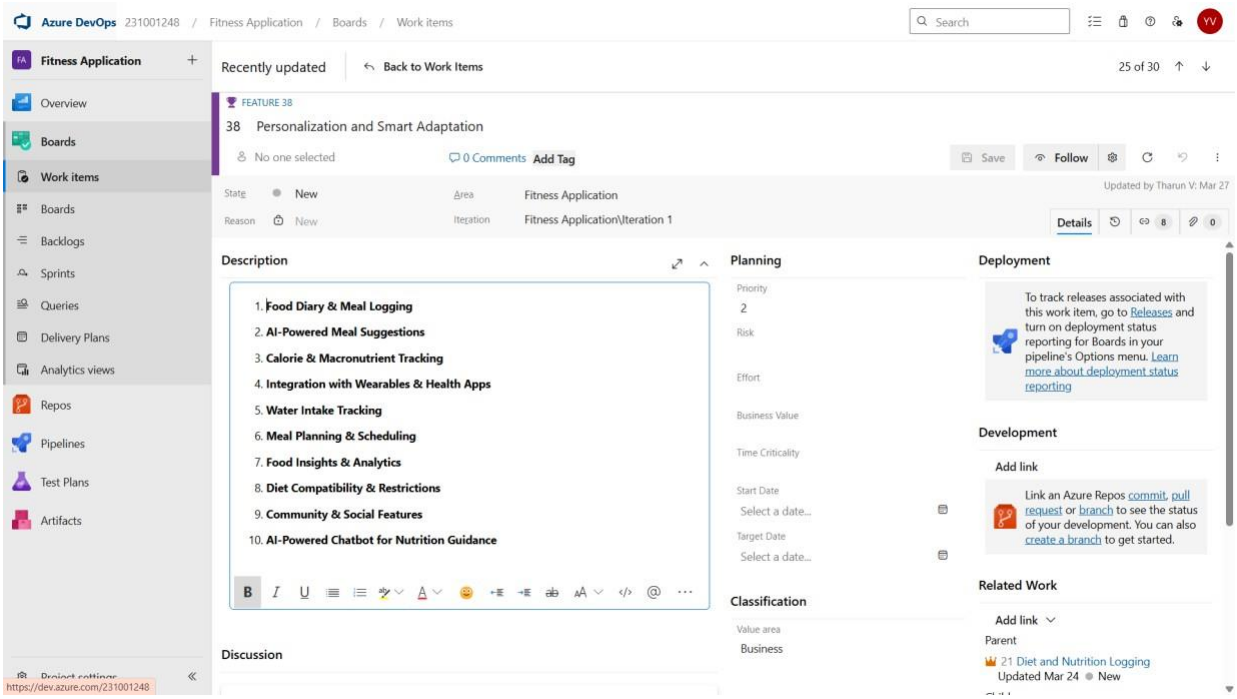
Create Epic, Features, User Stories, Task



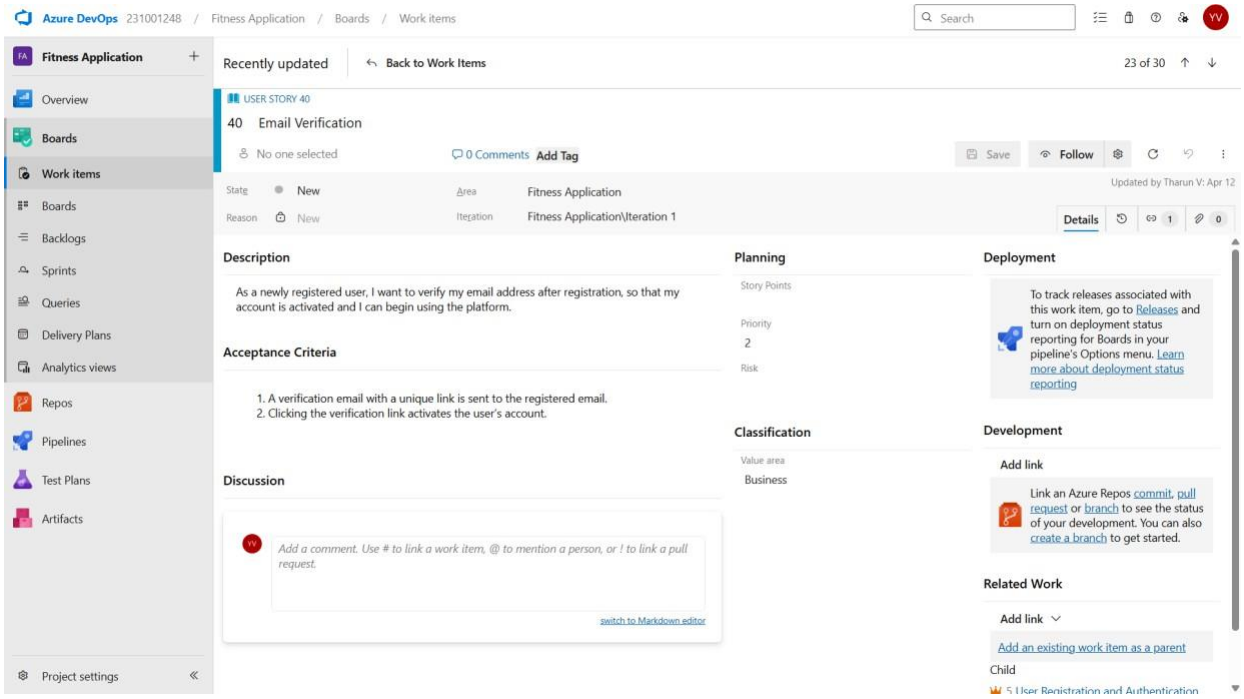
1. Fill in Epics



2. Fill in Features



3. Fill in User Story Details



Result:

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

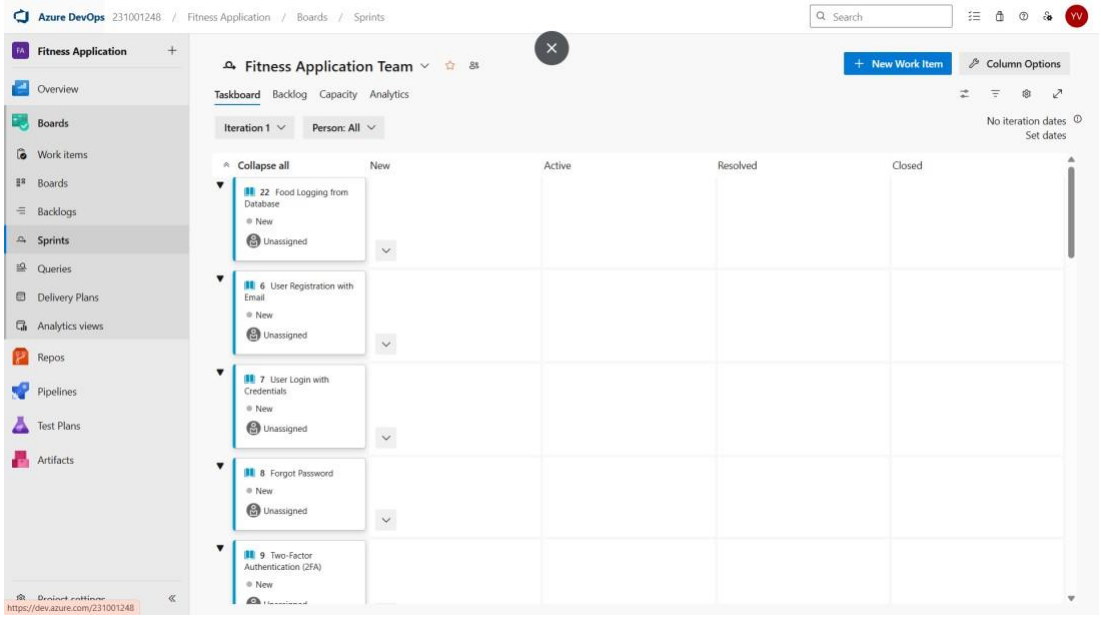
EXP NO: 4	SPRINT PLANNING

Aim:

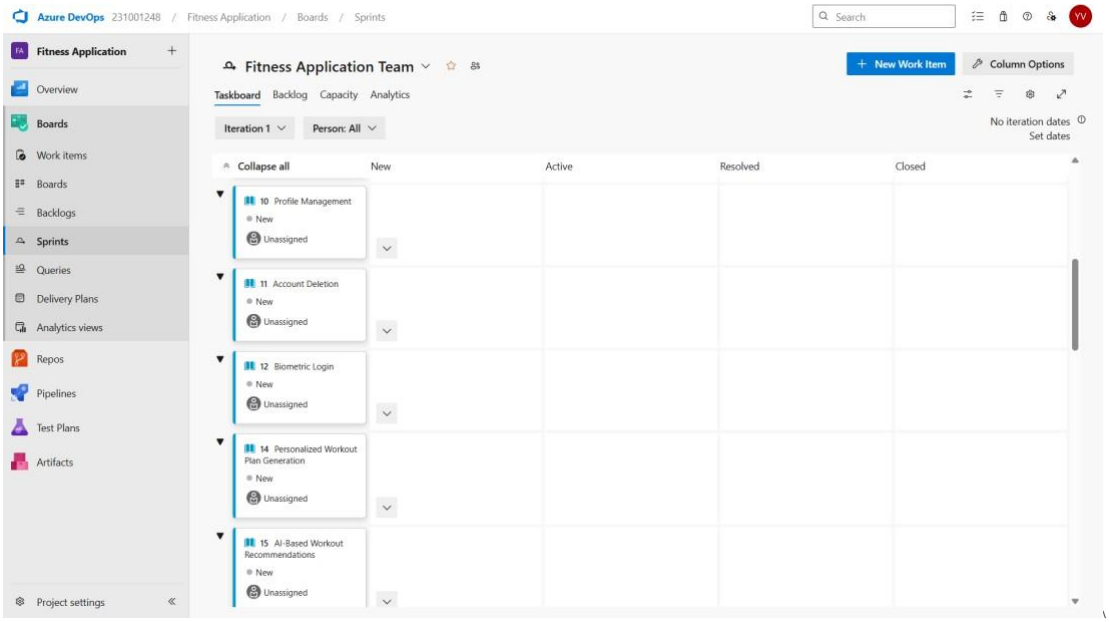
To assign user story to specific sprint for the Fitness Application Project.

Sprint Planning

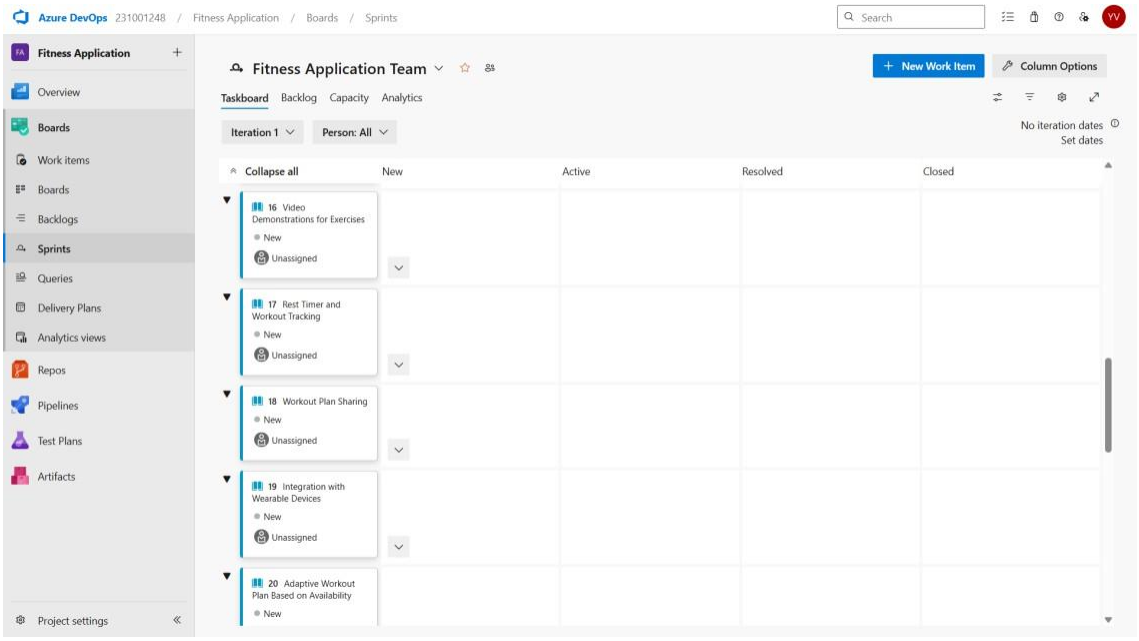
Sprint 1



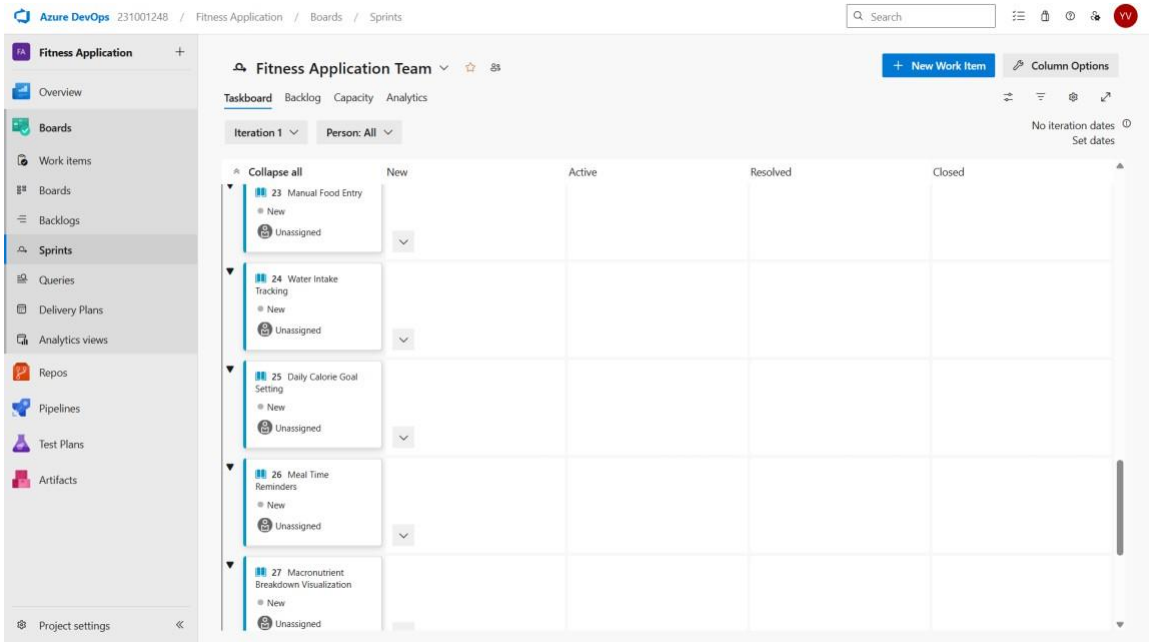
Sprint 2



Sprint 3



Sprint 4



Result:

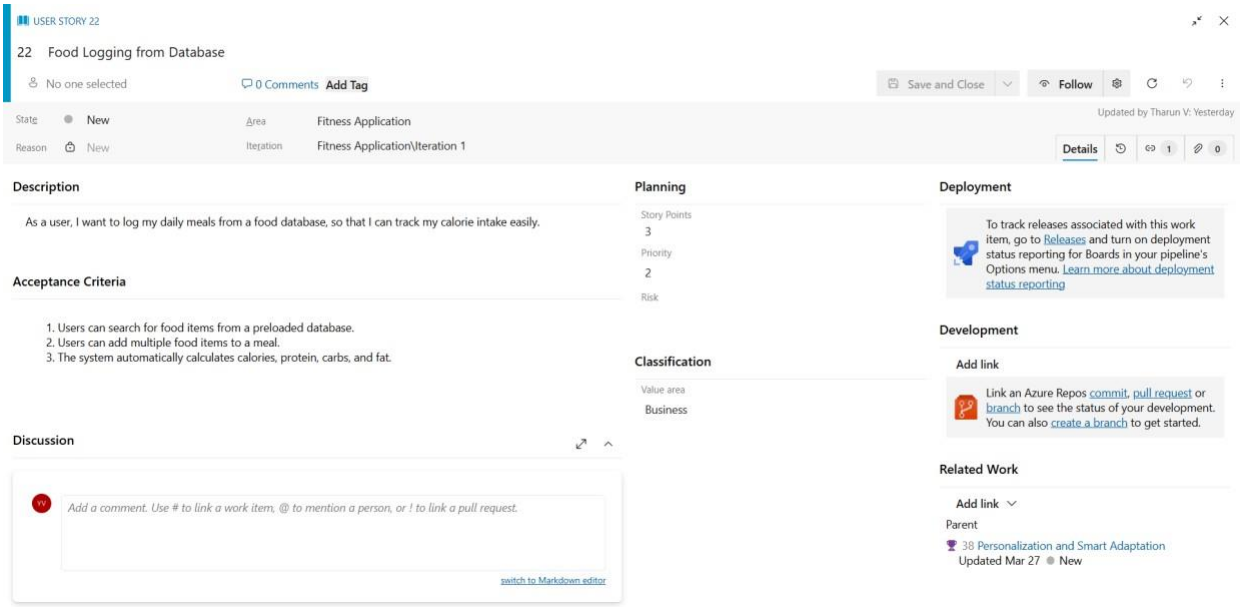
The Sprints are created for the Fitness Application Project.

EXP NO: 5	POKER ESTIMATION

Aim:

Create Poker Estimation for the user stories - Music Playlist Batch Creator Project.

Poker Estimation



Result:

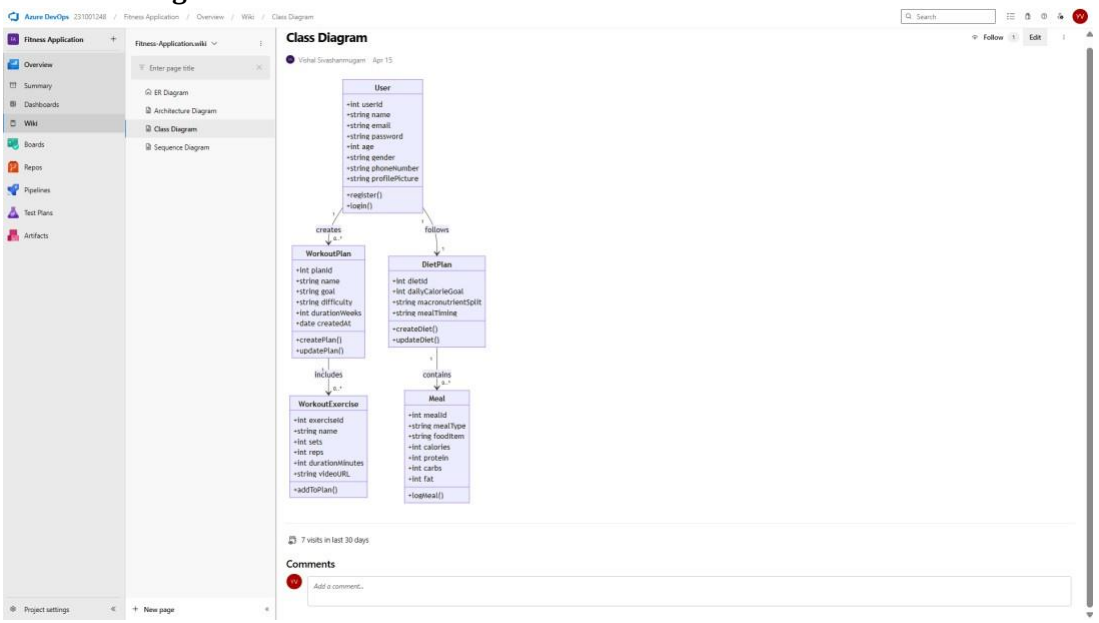
The Estimation/Story Points is created for the project using Poker Estimation.

EXP NO: 6	DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE

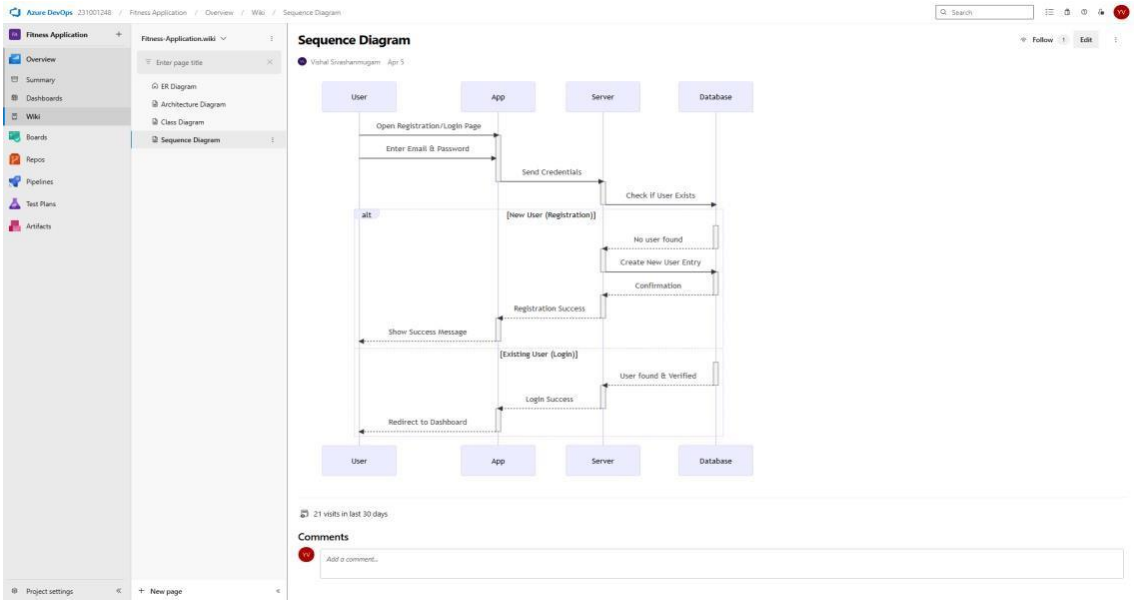
Aim:

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

6A. Class Diagram



6B. Sequence Diagram



Result:

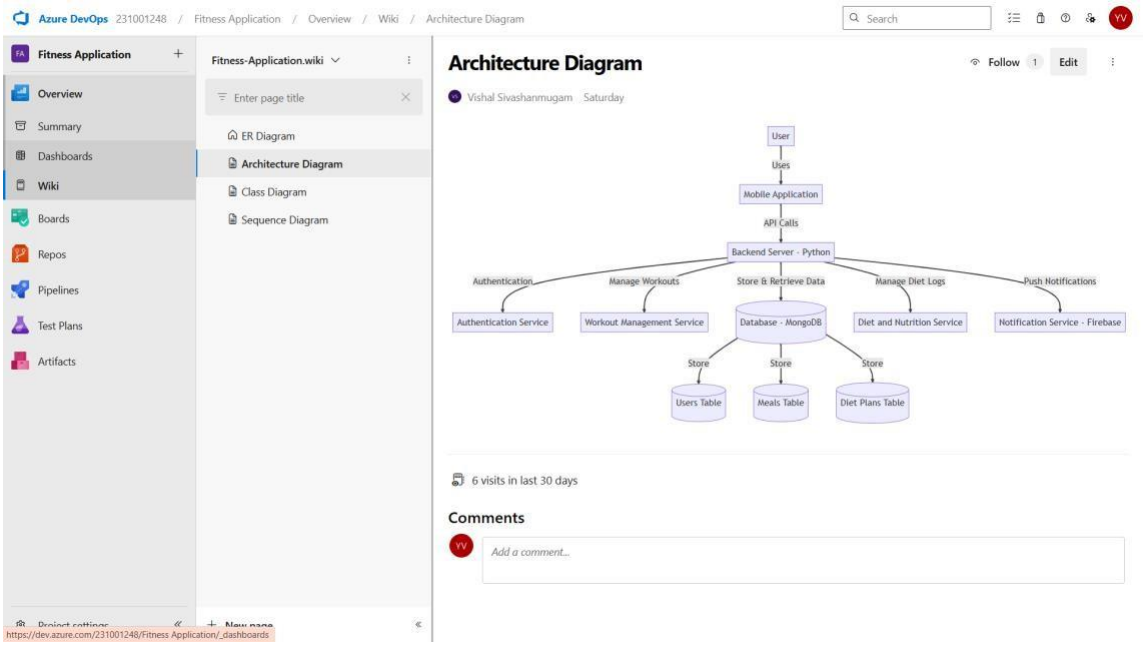
The Class Diagram and Sequence Diagram is designed Successfully for the Fitness Application.

EXP NO: 7	DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE

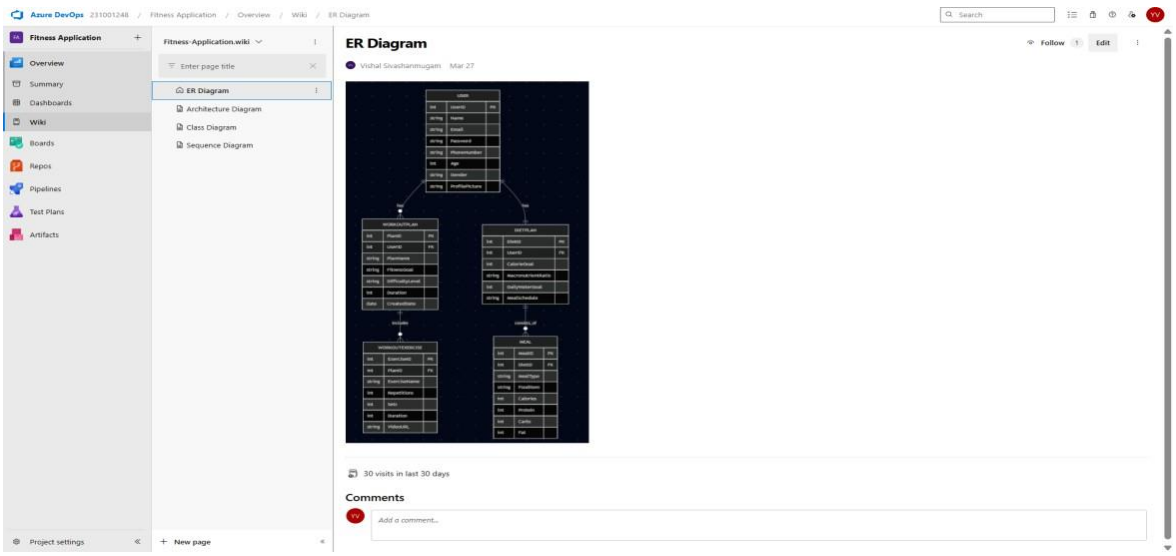
Aim:

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 Fitness Application.

EXP NO: 8	TESTING – TEST PLANS AND TEST CASES

Aim:

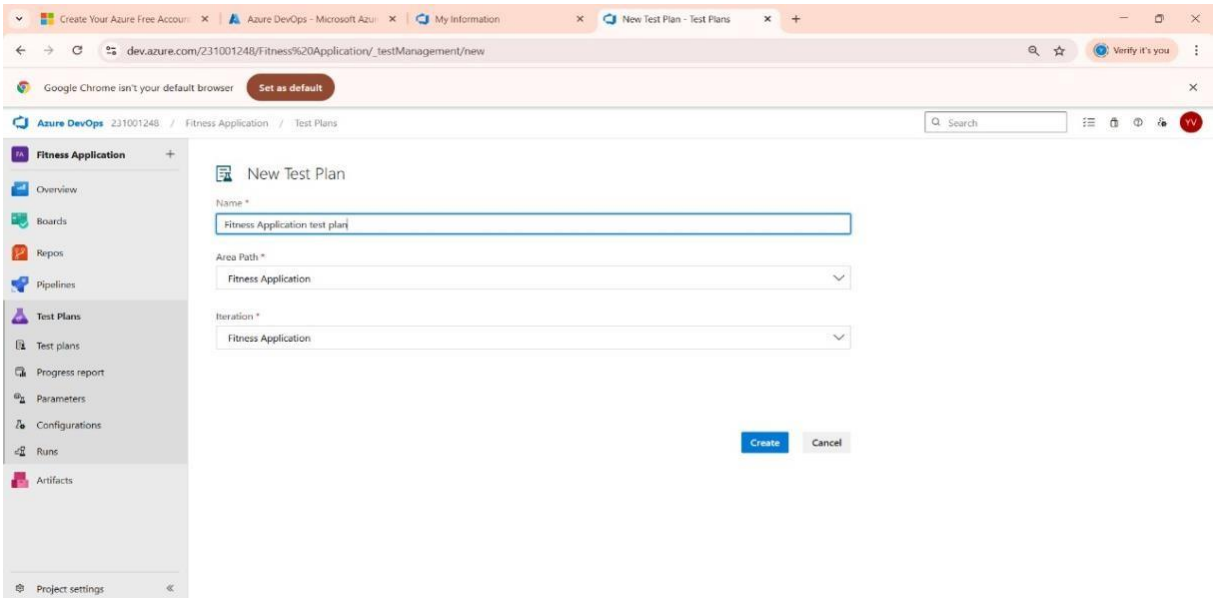
Test Plans and Test Case and write two test cases for at least five user stories showcasing the happy path and error scenarios in azure DevOps platform.

Test Planning and Test Case

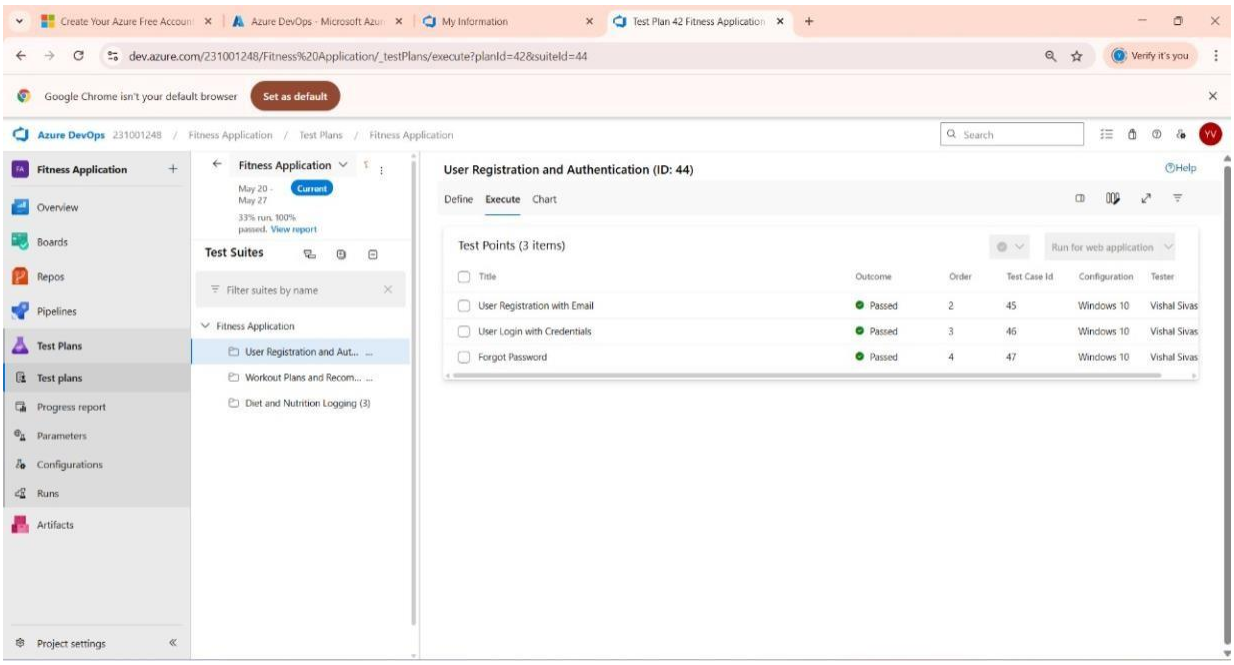
Test Case Design Procedure

1. **Identify Core Features**
    - o Include modules like Signup/Login, Workout Tracking, and Goal Setting.
    - o Cover health data monitoring, progress reports, and recommendations.
2. **Define User Actions**
    - o Simulate real user behaviour (e.g., logging in, starting a workout).
    - o Ensure actions align with functional and UI expectations.
3. **Happy Path Testing**
    - o Validate normal workflows like successful login or workout completion.
    - o Confirm system responds correctly to expected inputs.
4. **Error Path Testing**
    - o Test invalid scenarios like incorrect login or missing health data.
    - o Validate proper error messages and system stability.
5. **Step-wise Detailing**
    - o Break test cases into clear action-result pairs.
    - o Make each step easy to automate and verify.
6. **Clear Naming & IDs**
    - o Use readable, unique IDs (e.g., TC01 – Successful Goal Set).
    - o Helps in mapping tests to user stories and debugging.
7. **Organize Test Suites**
    - o Group by modules: Login, Workouts, Goals, Reports, etc.
    - o Improves structure and test execution flow in Azure DevOps.
8. **Prioritize & Review**
    - o Mark key features like health tracking as high-priority.
    - o Review all test cases for completeness and traceabilit

1. New test plan



2. Test suite



3. Test case

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

USER STORIES

- As a user, I want to sign up and log in securely so that I can track my fitness progress. (ID: 101)
- As a user, I need to view my daily workout summary in one place. (ID: 102)
- As a user, I should be able to start and stop a workout session easily. (ID: 103).
- As a user, I should be able to set and update fitness goals like steps or calories. (ID: 104).
- As a user, I need to receive real-time health stats like heart rate and calories burned. (ID: 105).

Test Suites

Test Suit: TS01 - User Login & Access (ID: 101)

1. **TC01 – Successful Sign Up**
    - **Action:**
        - Go to the Sign-Up page.
        - Enter valid name, email, and password.
        - Click "Sign Up".
    - **Expected Results:**
        - Sign-Up form is displayed.
        - Fields accept values without error.
        - Account is created, and the user is redirected to the dashboard.
    - **Type**: Happy Path
2. **TC02 – Secure Login**
    - **Action:**
        - Go to the Login page.
        - Enter valid email and password.
        - Click on "Login".
    - **Expected Results:**
        - Login form is displayed.
        - Fields accept data without error.
        - User is logged in and redirected to the dashboard.
    - **Type:** Happy Path
3. **TC03 – Sign Up with Existing Email**
    - **Action:**
        - Go to the Sign-Up page.
        - Enter a name and an already registered email.
        - Click on "Sign Up".
    - **Expected Results:**

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

4. **TC04 – Login with Wrong Password**
    - **Action:**
        - Go to the Login page.
        - Enter valid email and incorrect password.
        - Click on "Login".
    - **Expected Results:**
        - Input is accepted.
        - Error message "Invalid username or password" is shown.
    - **Type:** Error Path

Test Suit: TS02 – Workout Session (ID: 103)

1. **TC05 – Start Workout Session**
    - **Action:**
        - Log in successfully.
        - Navigate to "Workout" section.
    - **Expected Results:**
        - Workout session begins, timer starts, and tracking UI is displayed.
    - **Type:** Happy Path
2. TC06 – **Workout Session without Internet**
    - **Action:**
        - Disconnect from the internet.
        - Attempt to start a workout session.
    - **Expected Results:**
        - Network is offline.
        - Warning message "Cannot sync data. Workout will be saved offline." is shown.
    - **Type:** Error Path

Test Suit: TS03 - Real-Time Fitness Data (ID: 102)

1. **TC07 – Display Live Heart Rate**
    - **Action:**
        - Connect fitness band.
        - Observe heart rate panel.
    - **Expected Results:**
        - Real-time heart rate is displayed and updates continuously.
    - **Type:** Happy Path
2. TC08 – **No Heart Rate Data**
    - **Action:**

        - Start a workout session without connecting a fitness band.
        - Observe the heart rate panel.
    - **Expected Results:**
        - Heart rate panel shows "No device connected" or fallback message.
    - **Type:** Error Path

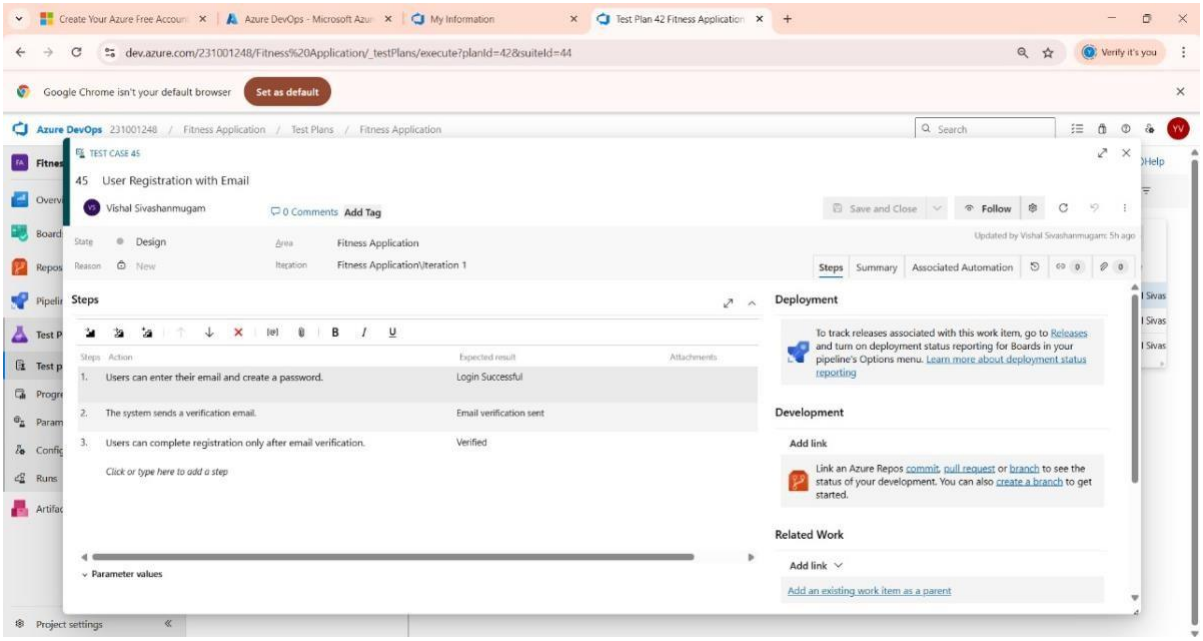
Test Suit: TS04 - Workout Editing (ID: 103)

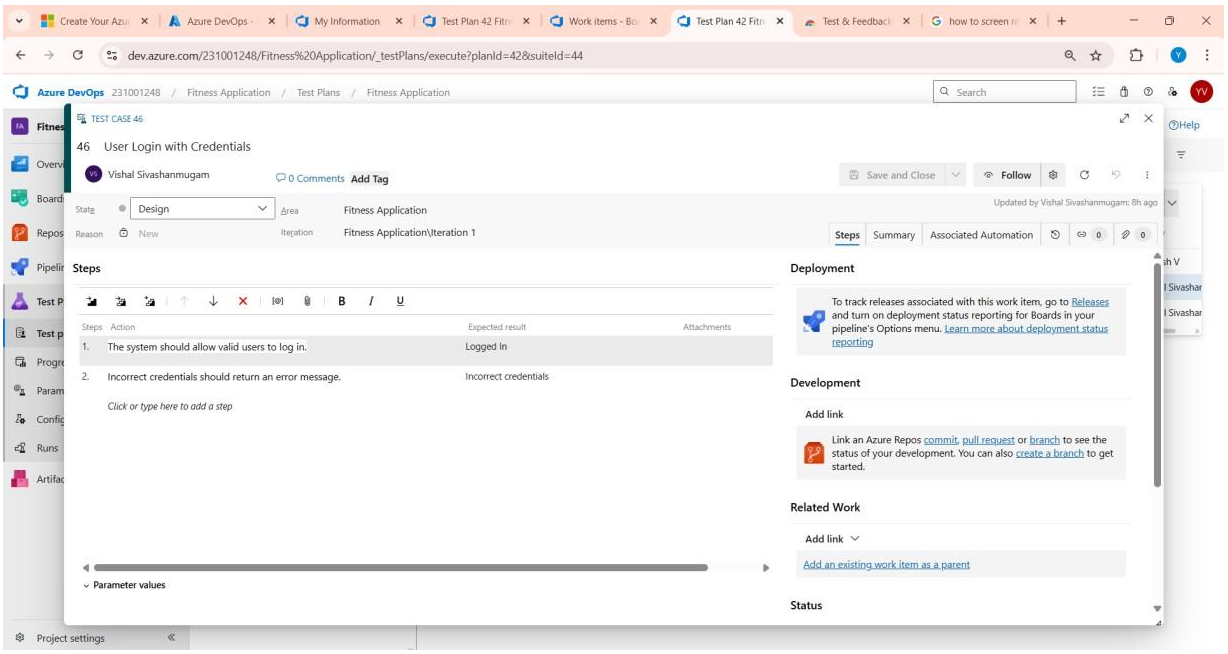
1. **TC09 – Rename Workout Successfully**
    - **Action:**
        - Navigate to "My Workout".
        - Click "Rename" next to a workout.
        - Enter a new name and click "Save".
    - **Expected Results:**
        - Workout name updates successfully.
    - **Type:** Happy Path
2. **TC10 – Rename with Blank Name**
    - **Action:**
        - Click "Rename" on a workout.
        - Leave the field blank.
        - Click "Save".
    - **Expected Results:**
        - Error message "Workout name cannot be empty" is shown.
    - **Type:** Error Path
3. **TC11 – Change Exercise Order in Workout**
    - **Action:**
        - Open a workout.
        - Drag and drop exercises to reorder.
        - Click "Save".
    - **Expected Results:**
        - Workout order is updated and saved.
    - **Type:** Happy Path
4. **TC12 – Change Exercise Order Fails Due to Network Error**
    - **Action:**
        - Login and go to “My Workouts”.
        - Select a workout.
        - Go offline or simulate server error.
        - Reorder exercise and click “Save Order”.
    - **Expected Results:**
        - Error message: "Failed to update order. Please check your connection".
    - **Type:** Error Path

Test Suit: TS05 - Smart Workout Plan Creation (ID: 104)

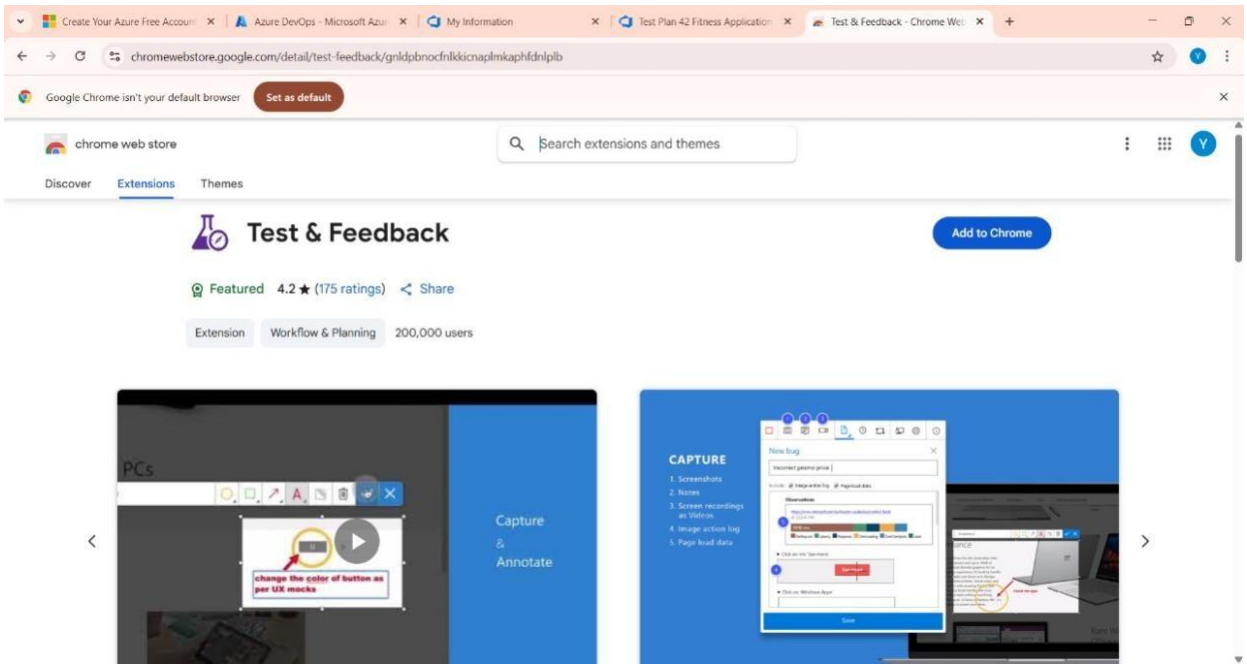
1. **TC13 – Generate Workout Plan Based on User Goals**
    - **Action:**
        - Login with valid credentials.
        - Click on "Generate Workout plan".
        - Select categories.
        - Click "Generate plan".
    - **Expected Results:**
        - Workout is generated based on selected goal and user profile.
    - **Type:** Happy Path
2. **TC14 – Fail to Generate Plan Due to Missing Goal Selection or Invalid Input**
    - **Action:**
        - Login with valid credentials.
        - Click on "Generate Workout plan".
        - Select categories.
        - Click "Generate plan".
    - **Expected Results:**
        - Error message: "Please select a fitness goal" or "Unable to generate plan with the given input" is shown.
    - **Type:** Error Path

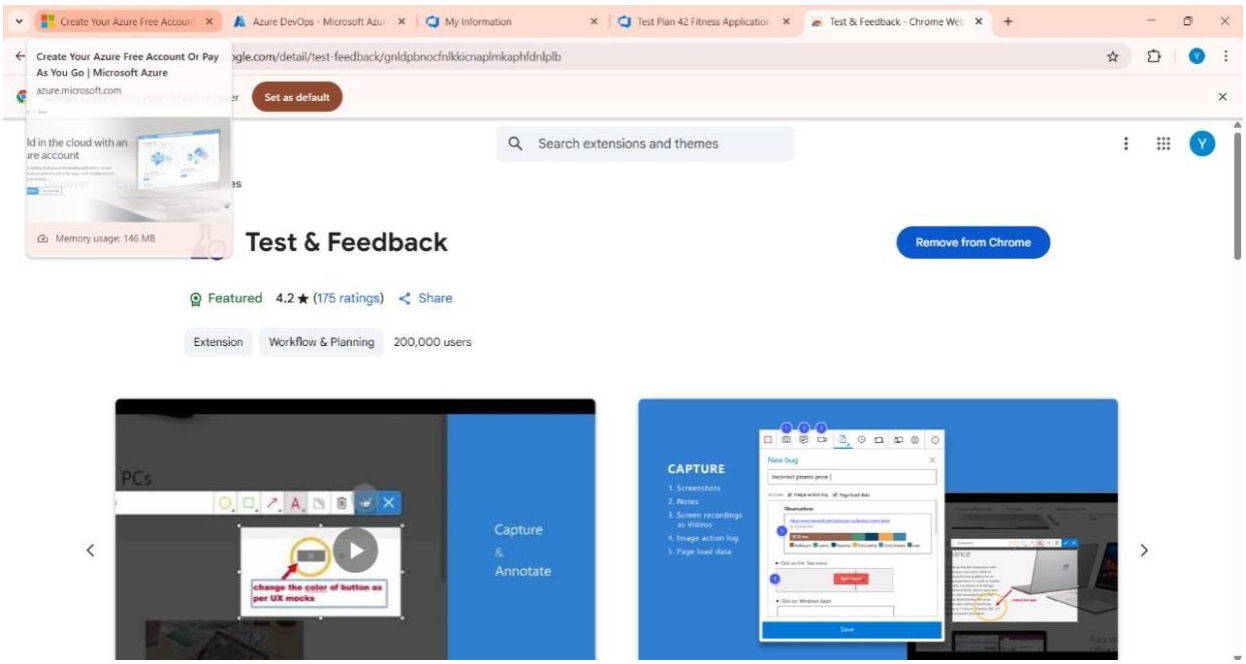
Test Cases



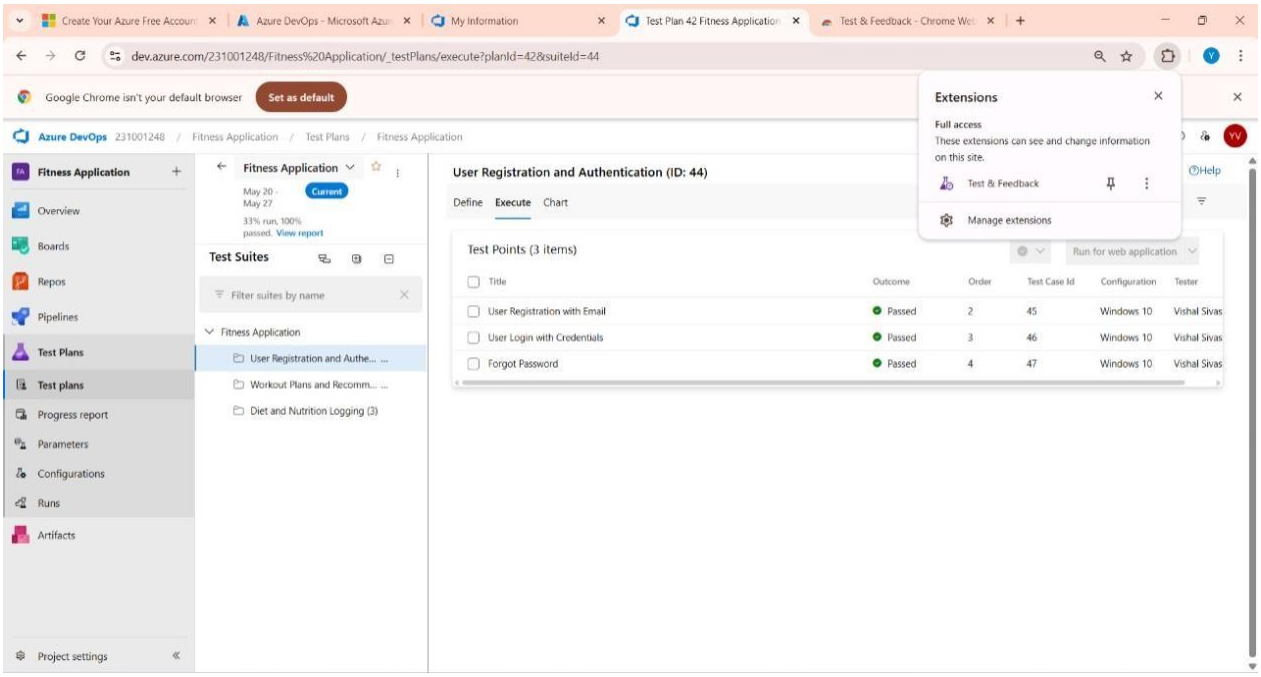


4. Installation of test

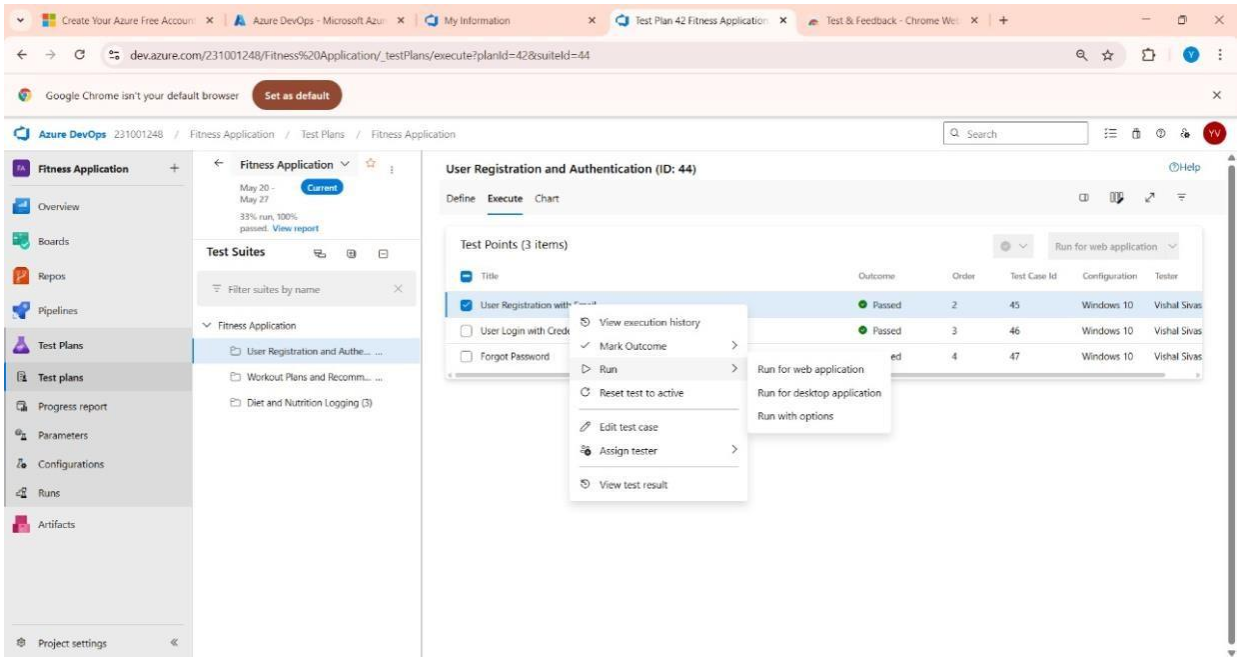


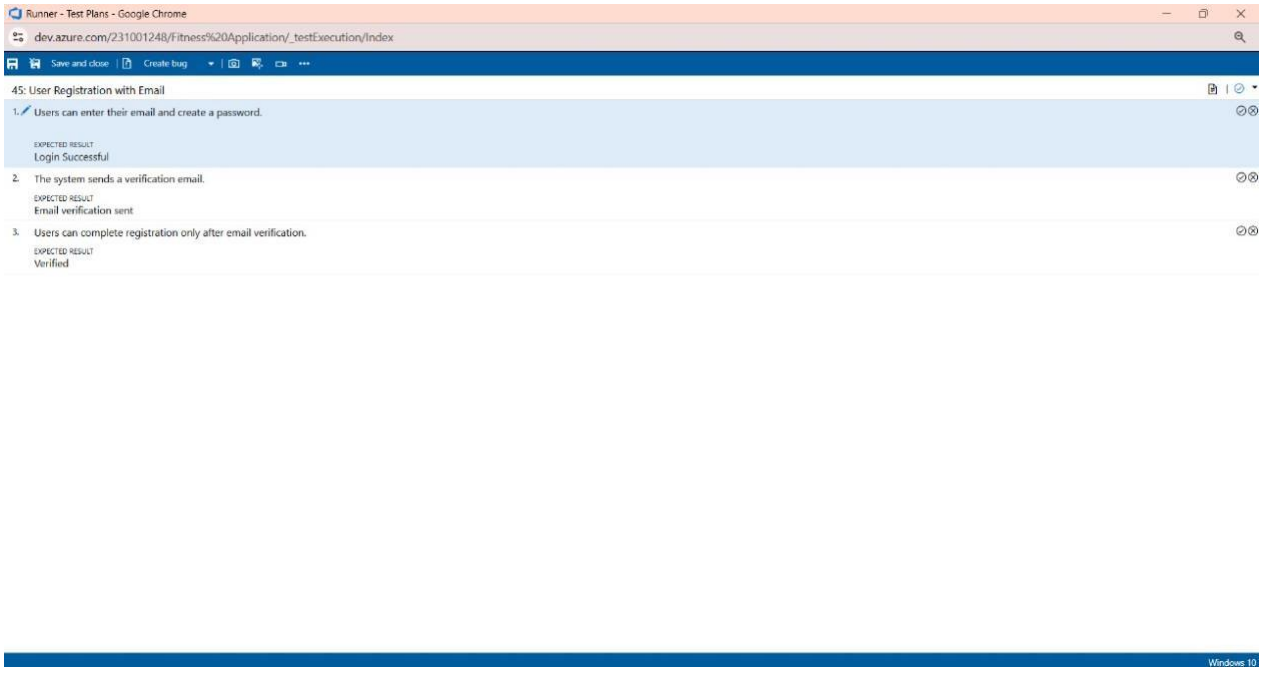


Test and feedback
Showing it as an extension

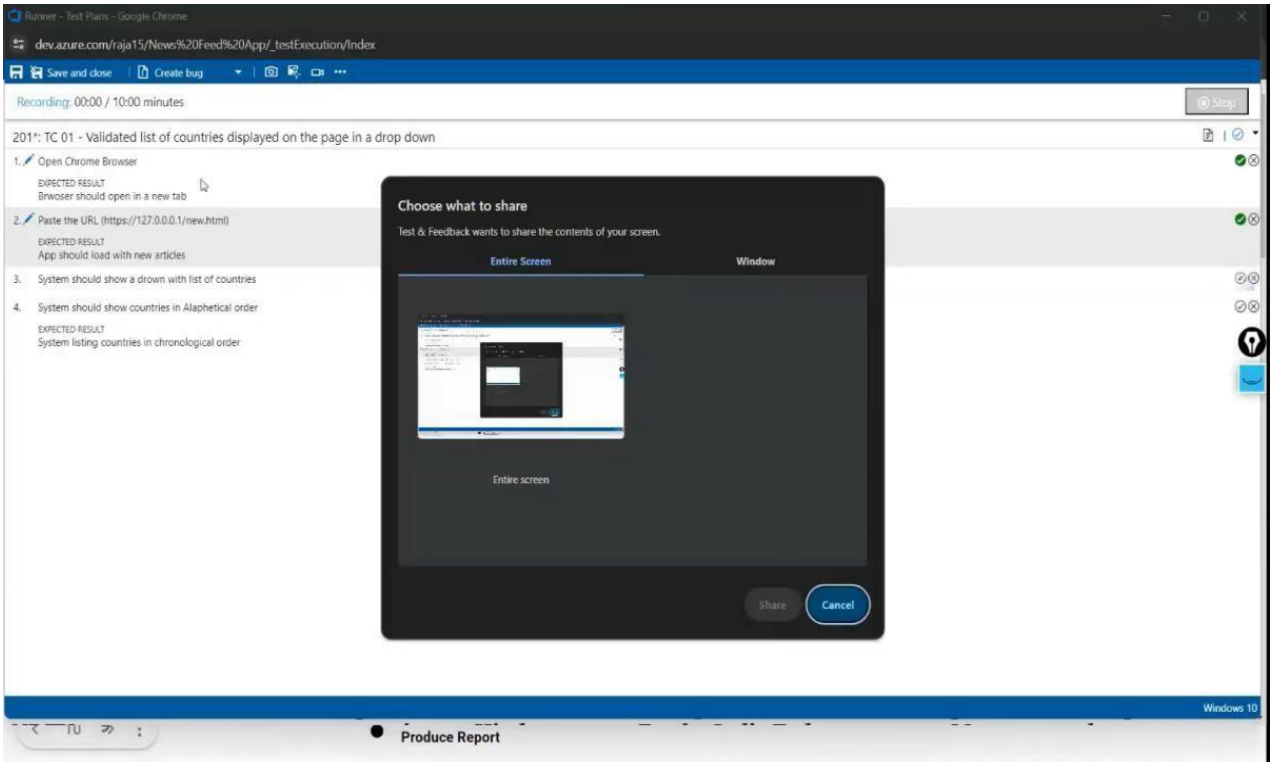


5. Running the test cases

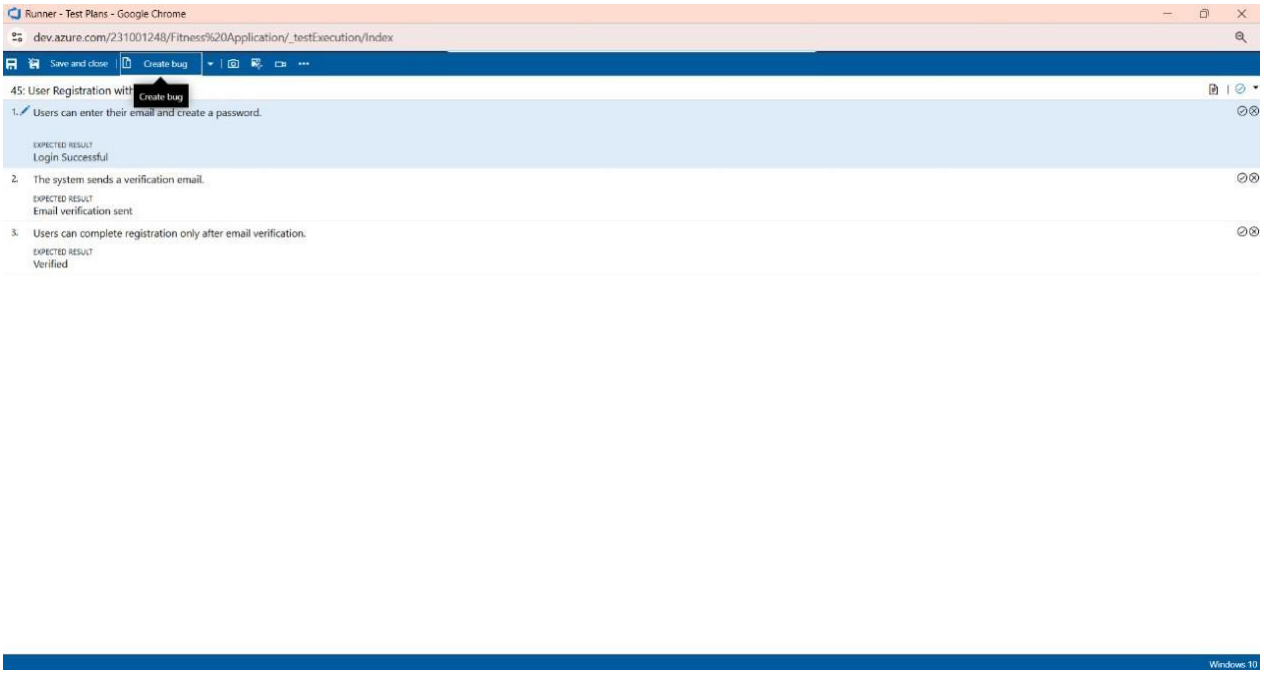


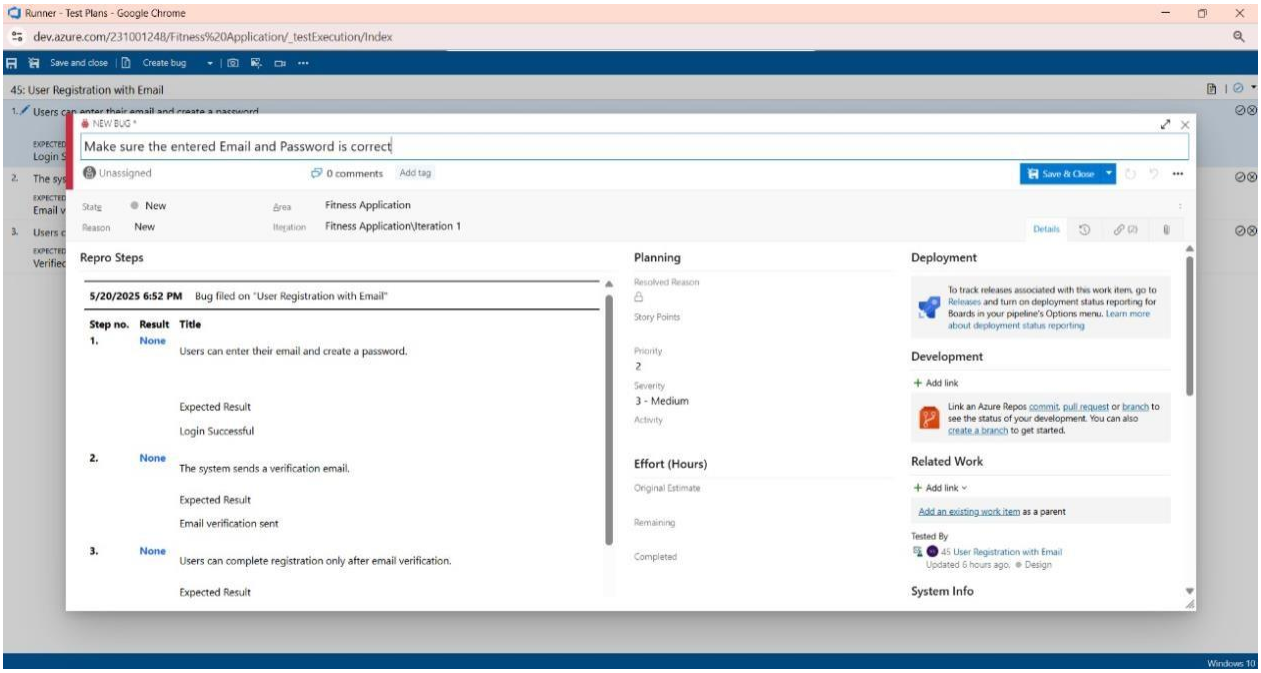


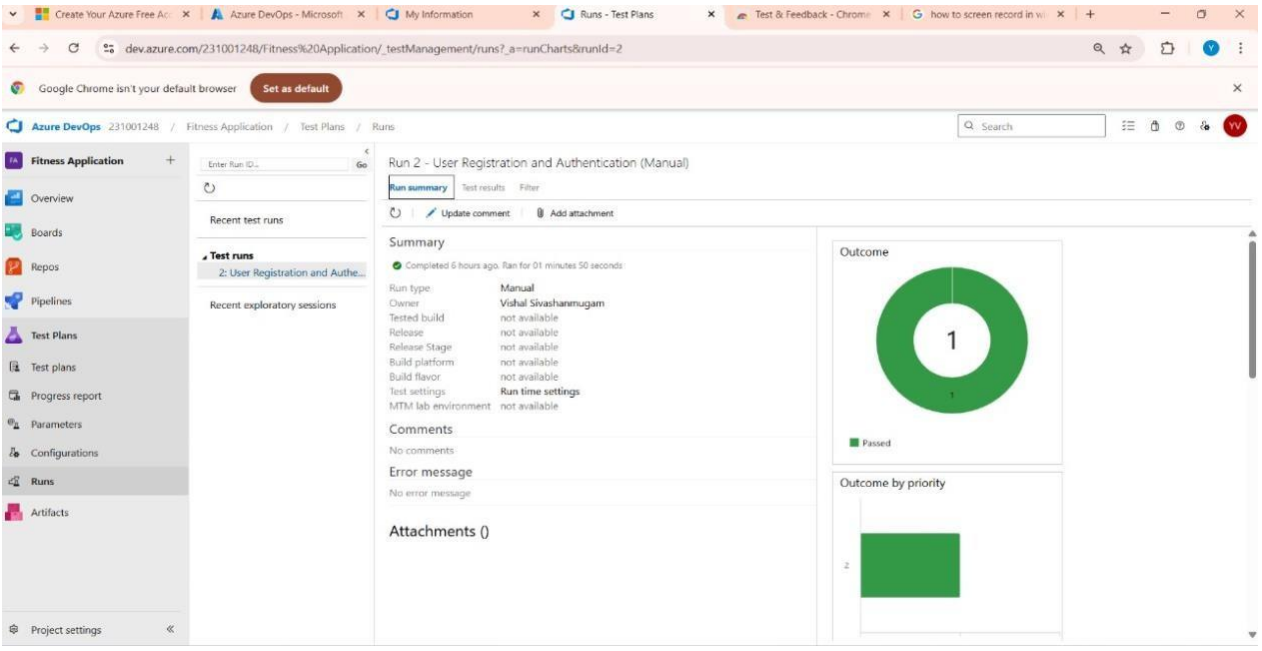
6. Recording the test case



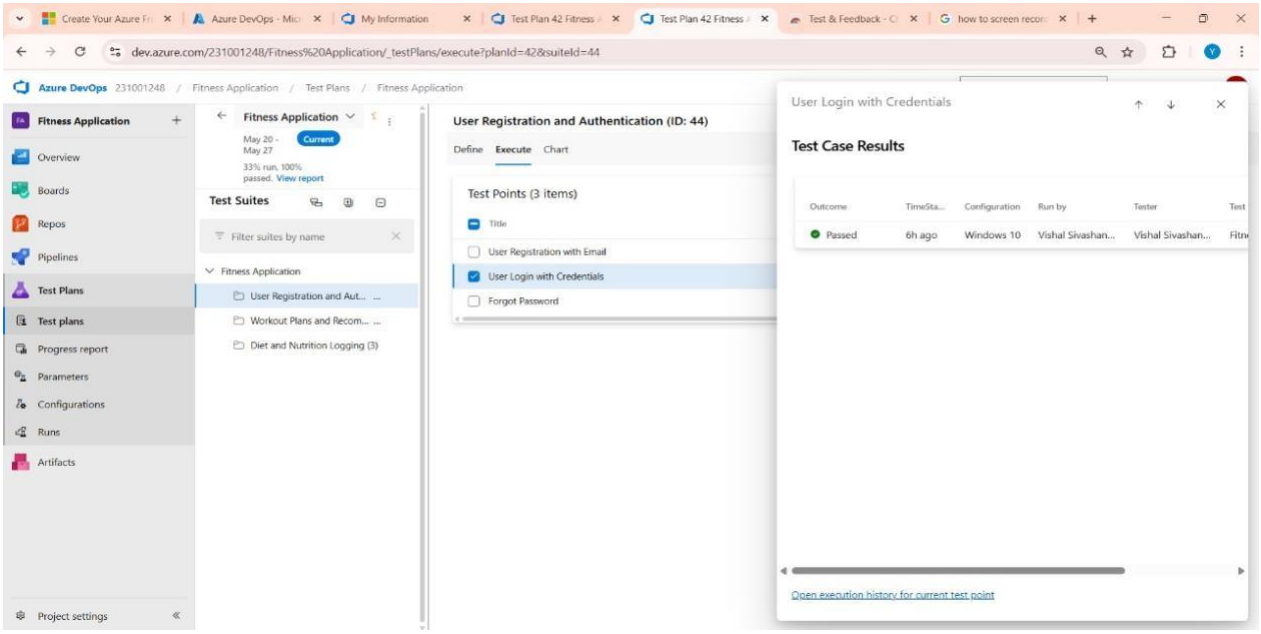
7. Creating the bug



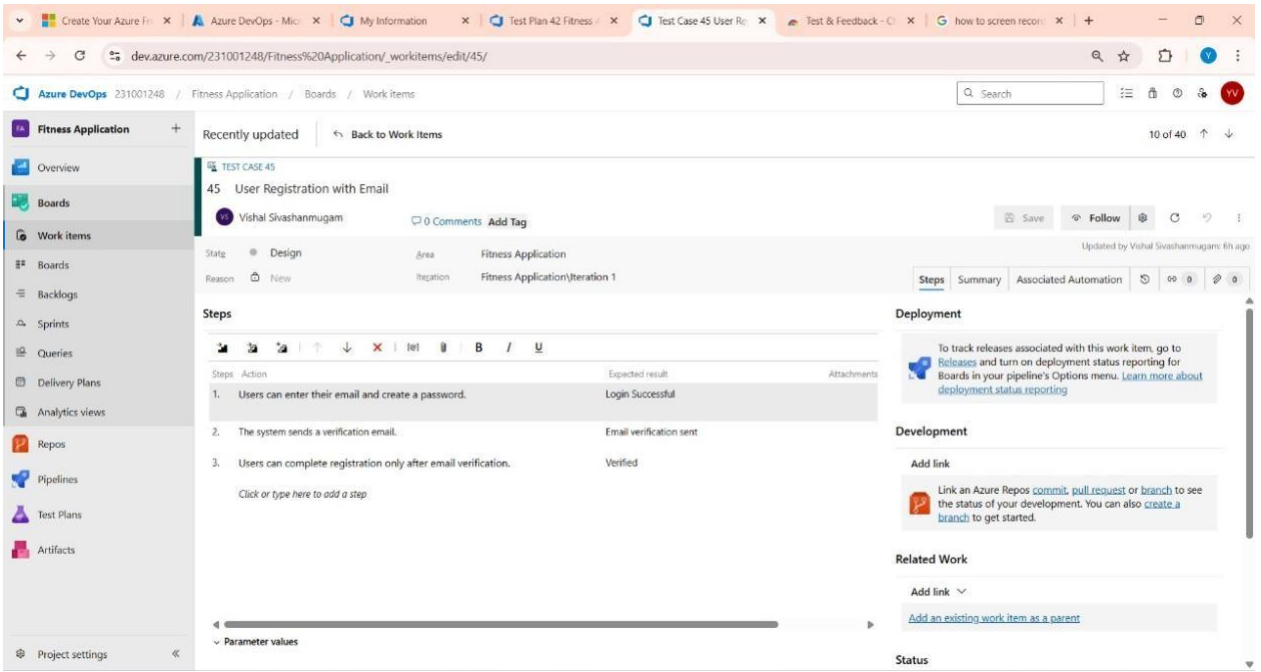

Runner - Test Plans - Google Chrome
dev.azure.com/231001248/Fitness%20Application/_testExecution/Index
Save and close
Create bug
45: User Registration with Email
NEW BUG *
Make sure the entered Email and Password is correct
Unassigned
0 comments
Add tag
Save & Close
State
New
Reason
New
Area
Fitness Application
Iteration
Fitness Application\Iteration 1
Details
Repro Steps
5/20/2025 6:52 PM Bug filed on "User Registration with Email"
Step no. Result Title
1. None
Users can enter their email and create a password.
Expected Result
Login Successful
2. None
The system sends a verification email.
Expected Result
Email verification sent
3. None
Users can complete registration only after email verification.
Expected Result
Planning
Resolved Reason
Story Points
Priority
2
Severity
3 - Medium
Activity
Effort (Hours)
Original Estimate
Remaining
Completed
Deployment
To track releases associated with this work item, go to Releases and turn on deployment status reporting for Boards in your pipeline's Options menu. Learn more about deployment status reporting
Development
+ Add link
Link an Azure Repos commit, pull request or branch to see the status of your development. You can also create a branch to get started.
Related Work
+ Add link
Add an existing work item as a parent
Tested By
45 User Registration with Email
Updated 6 hours ago, Design
System Info
Windows 10


dev.azure.com/231001248/Fitness%20Application/_testManagement/runs?_a=runCharts&runId=2
Google Chrome isn't your default browser
Set as default
Azure DevOps 231001248 / Fitness Application / Test Plans / Runs
Search
Fitness Application
Overview
Boards
Repos
Pipelines
Test Plans
Test plans
Progress report
Parameters
Configurations
Runs
Artifacts
Project settings
Recent test runs
Test runs
2: User Registration and Authe...
Recent exploratory sessions
Run 2 - User Registration and Authentication (Manual)
Run summary
Test results
Filter
Update comment
Add attachment
Summary
Completed 6 hours ago. Ran for 01 minutes 50 seconds
Run type Manual
Owner Vishal Sivashanmugam
Tested build not available
Release not available
Release Stage not available
Build platform not available
Build flavor not available
Test settings Run time settings
MTM lab environment not available
Comments
No comments
Error message
No error message
Attachments ()
Outcome
1
Passed
Outcome by priority

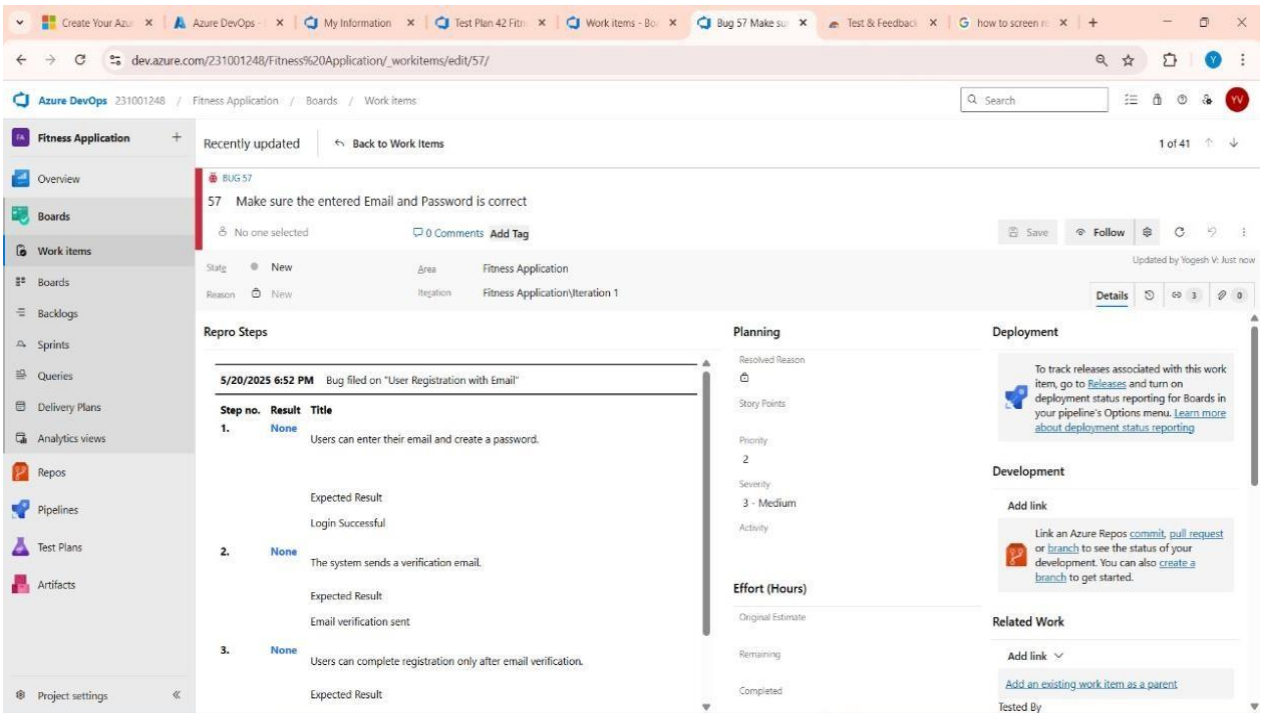
8. Test case results



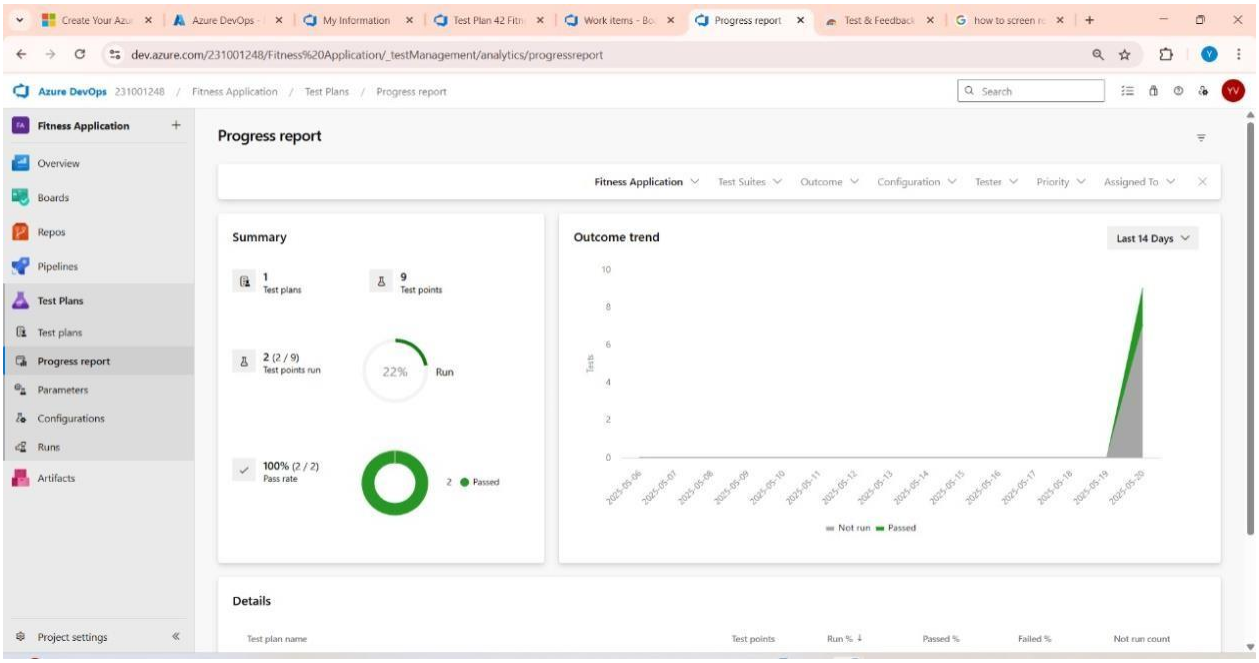
9. Test report summary

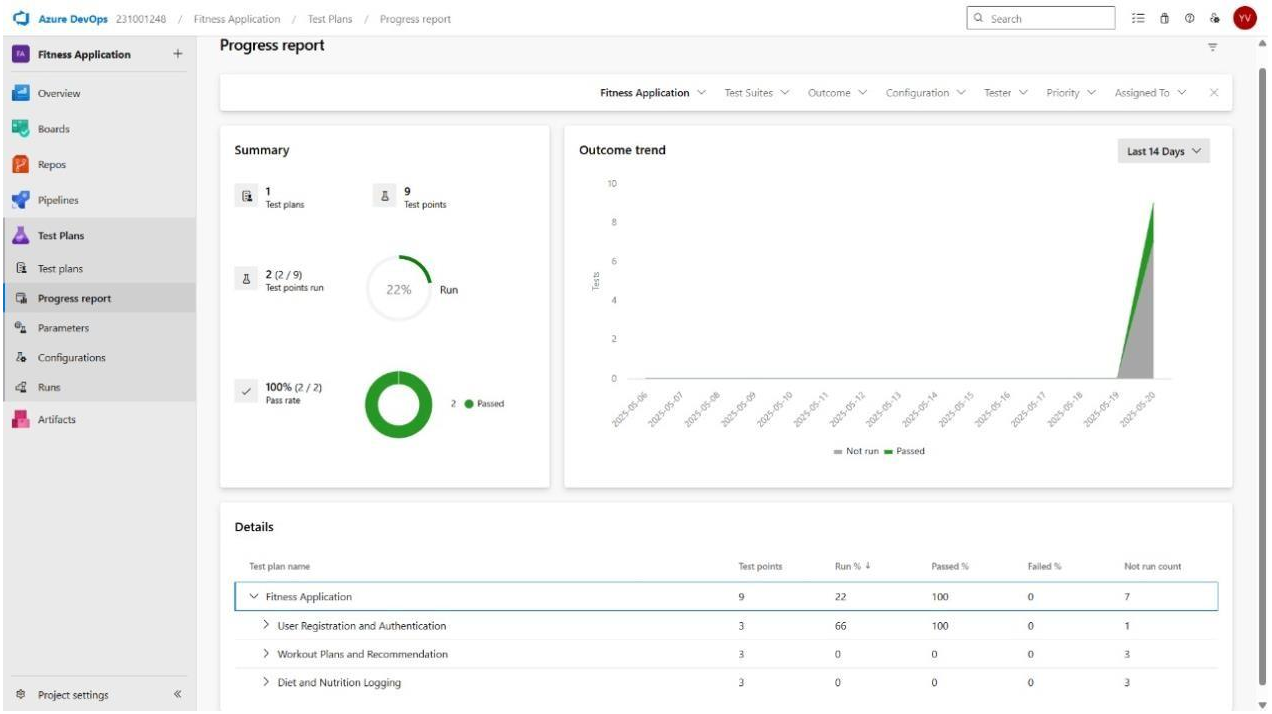


- Assigning bug to the developer and changing state

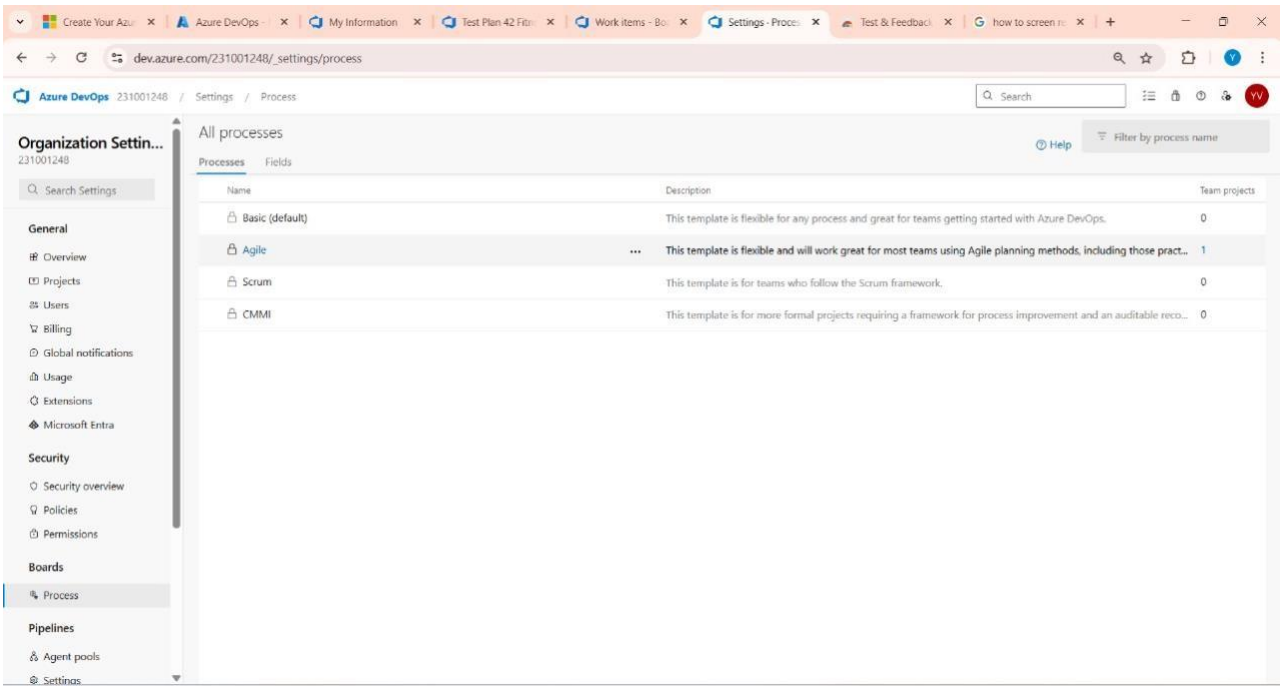


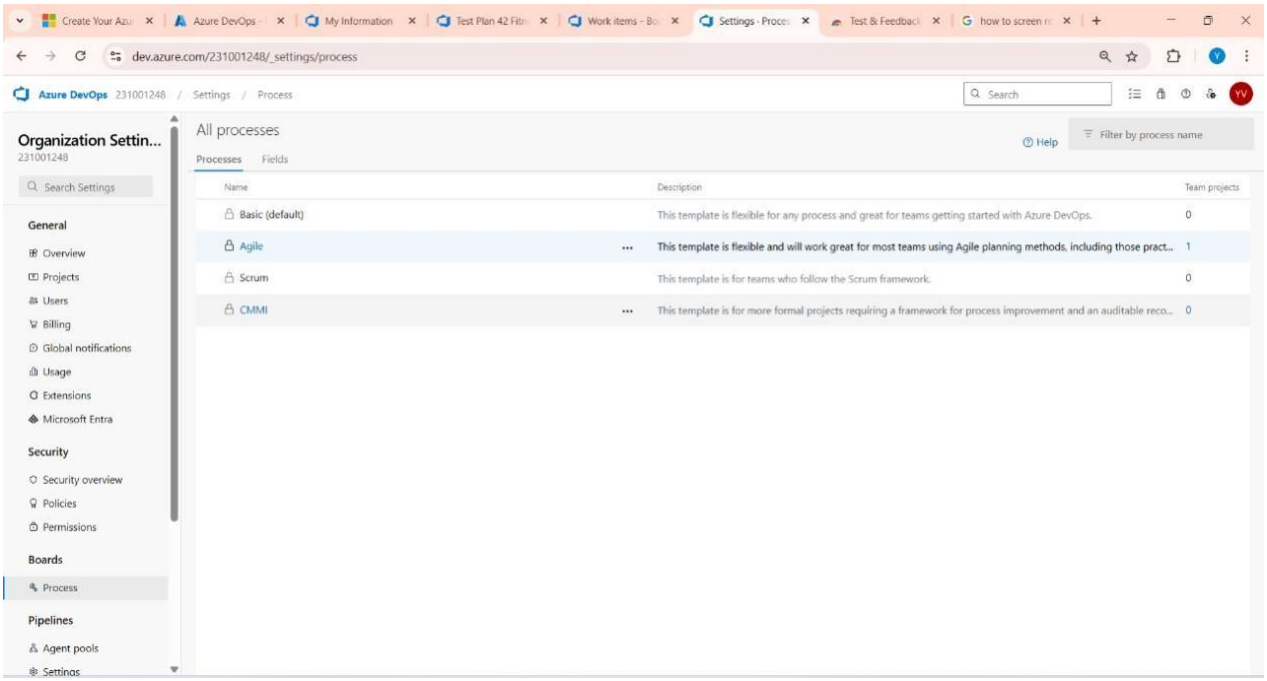
10. Progress report



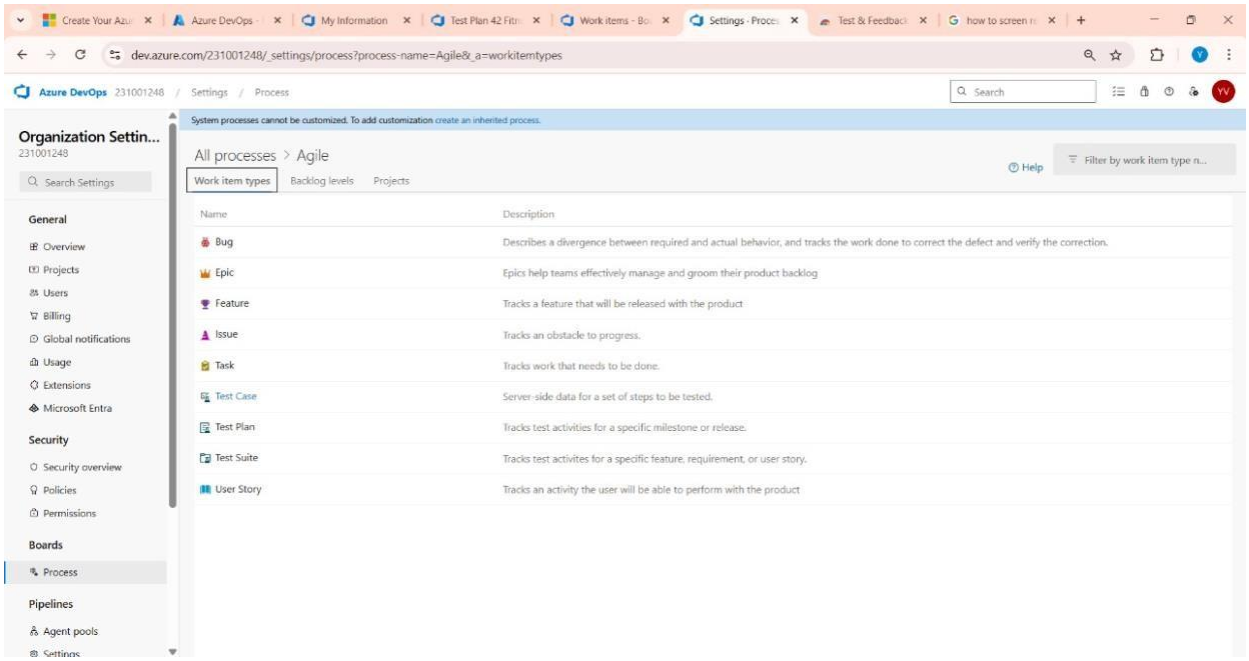


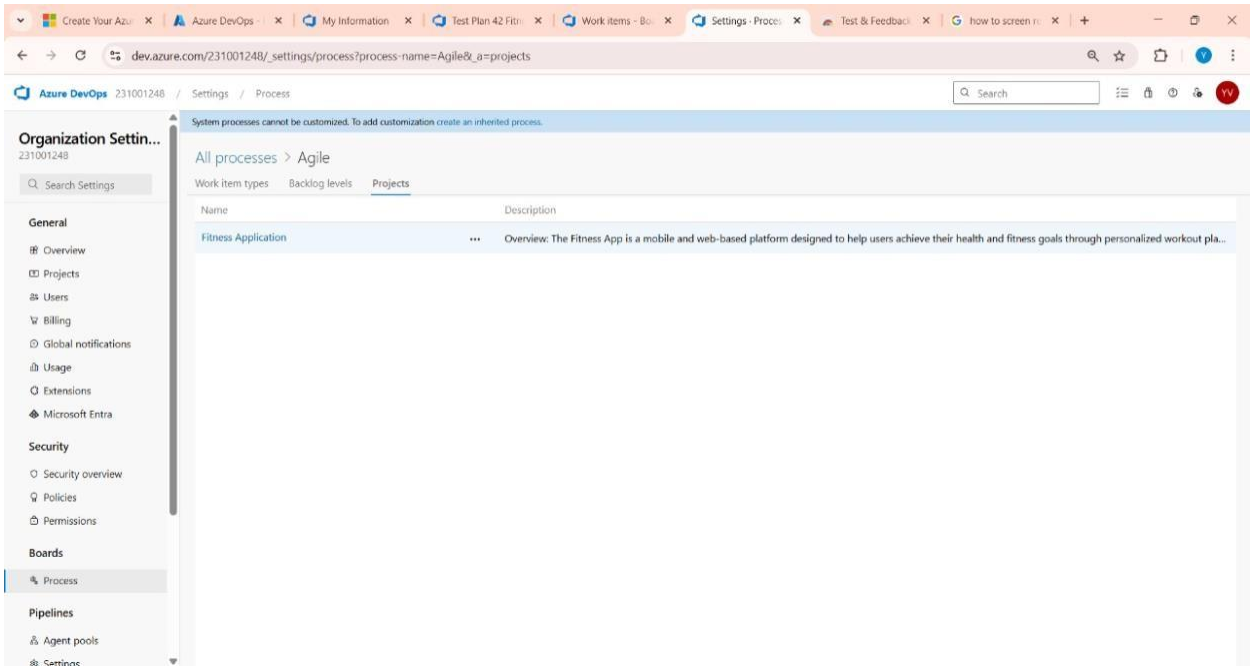
11. Changing the test template

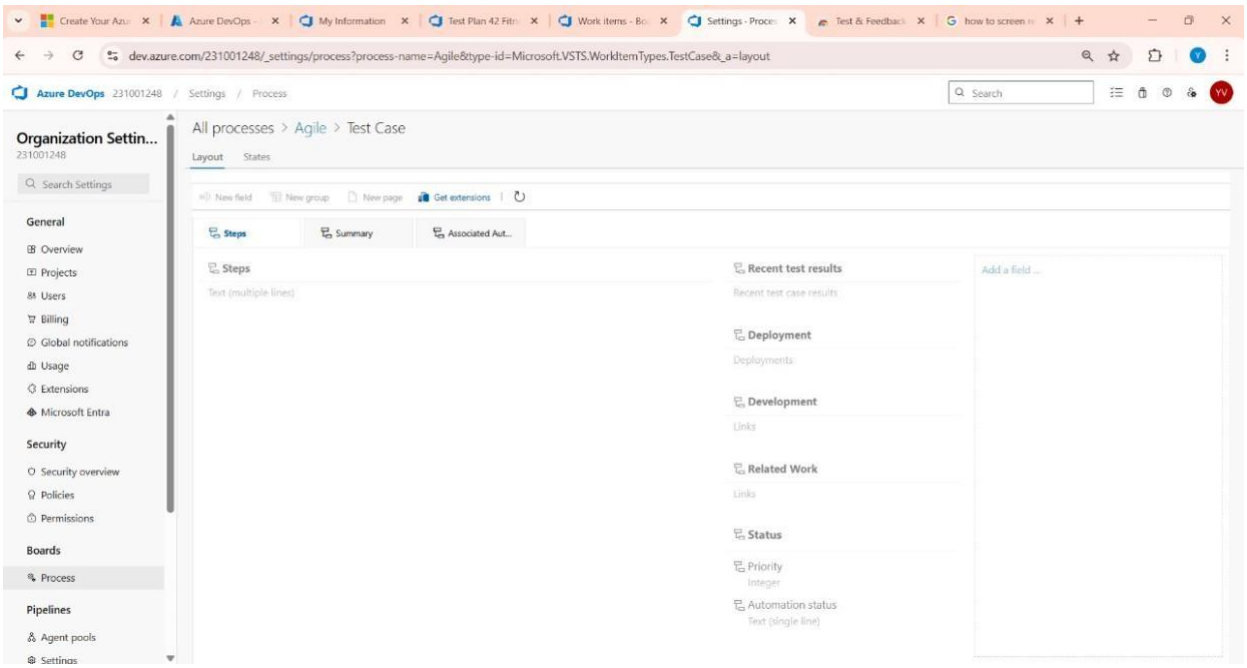



dev.azure.com/231001248/_settings/process
Azure DevOps 231001248 / Settings / Process
Search
Organization Settin...
231001248
Search Settings
General
Overview
Projects
Users
Billing
Global notifications
Usage
Extensions
Microsoft Entra
Security
Security overview
Policies
Permissions
Boards
Process
Pipelines
Agent pools
Settings
All processes
Help
Filter by process name
Processes
Fields
Name
Description
Team projects
Basic (default)
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...
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 improvement and an auditable reco...
0

12. View the new test case template







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 PERFORMANCE TESTING

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:

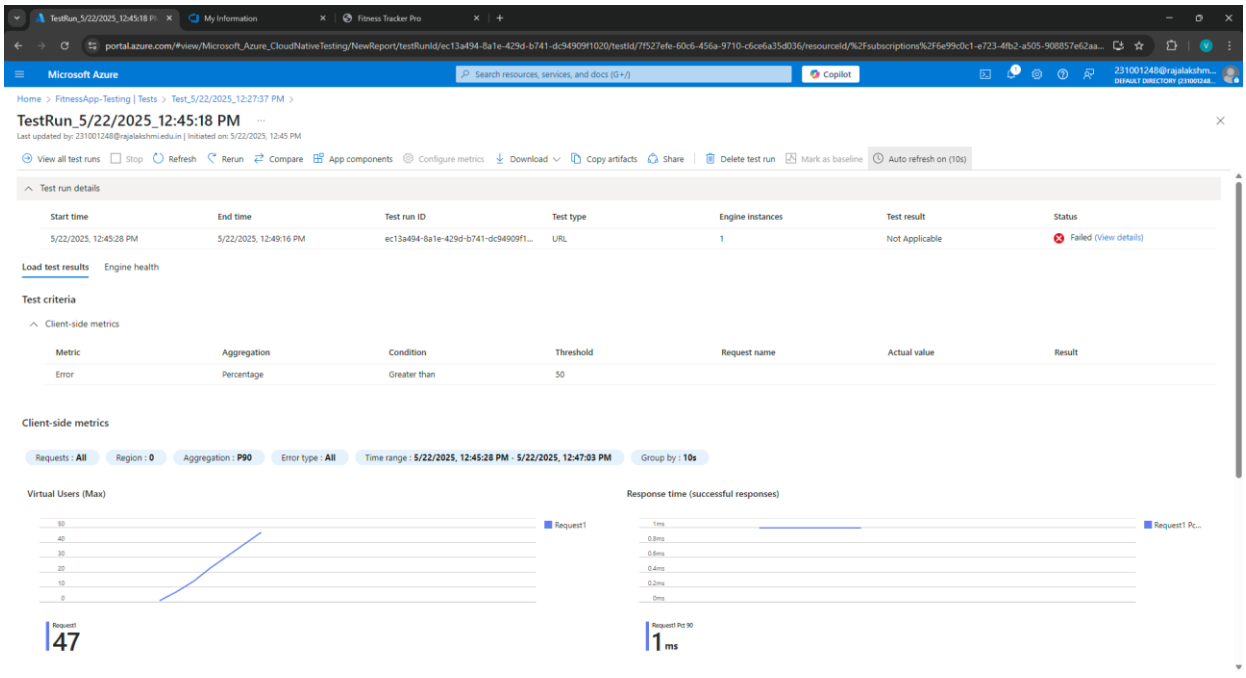
1. Sign in to Azure Portal
   Go to https://portal.azure.com and log in.
2. Create the Resource
   - Go to *Create a resource* → Search for "Azure Load Testing".
   - Select Azure Load Testing and click Create.
3. Fill in the Configuration Details
   - Subscription*:* Choose your Azure subscription.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

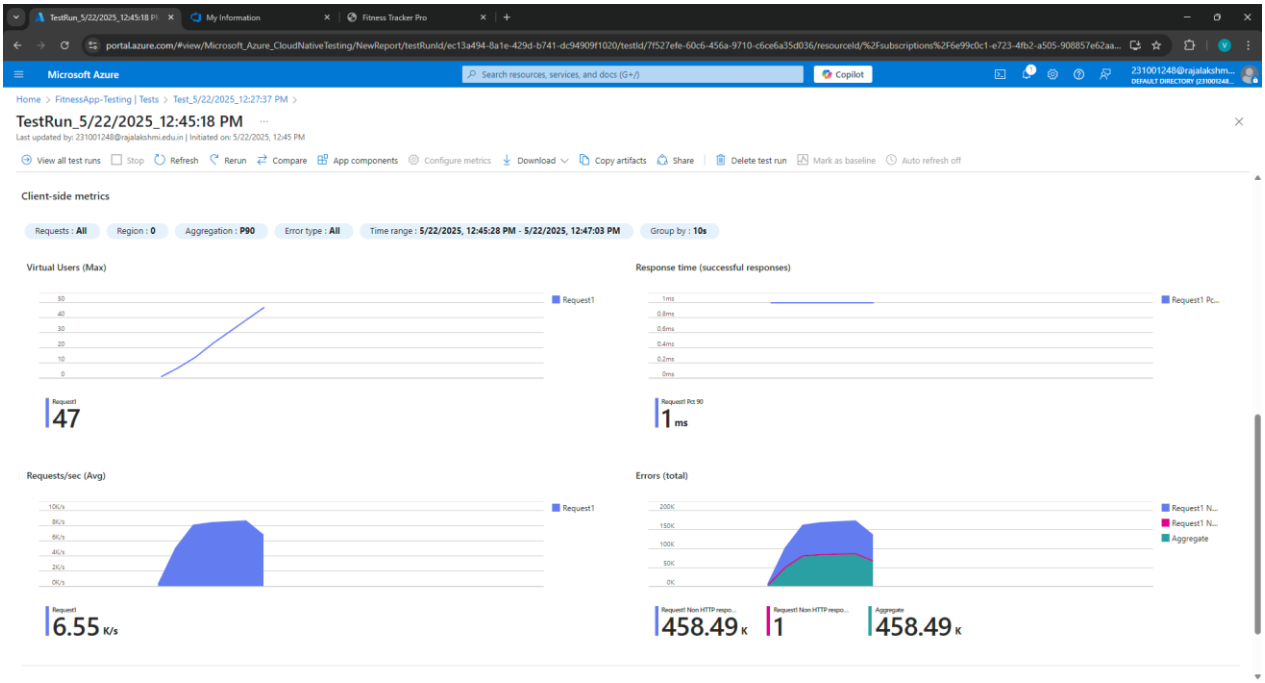
Steps to Create and Run a Load Test:

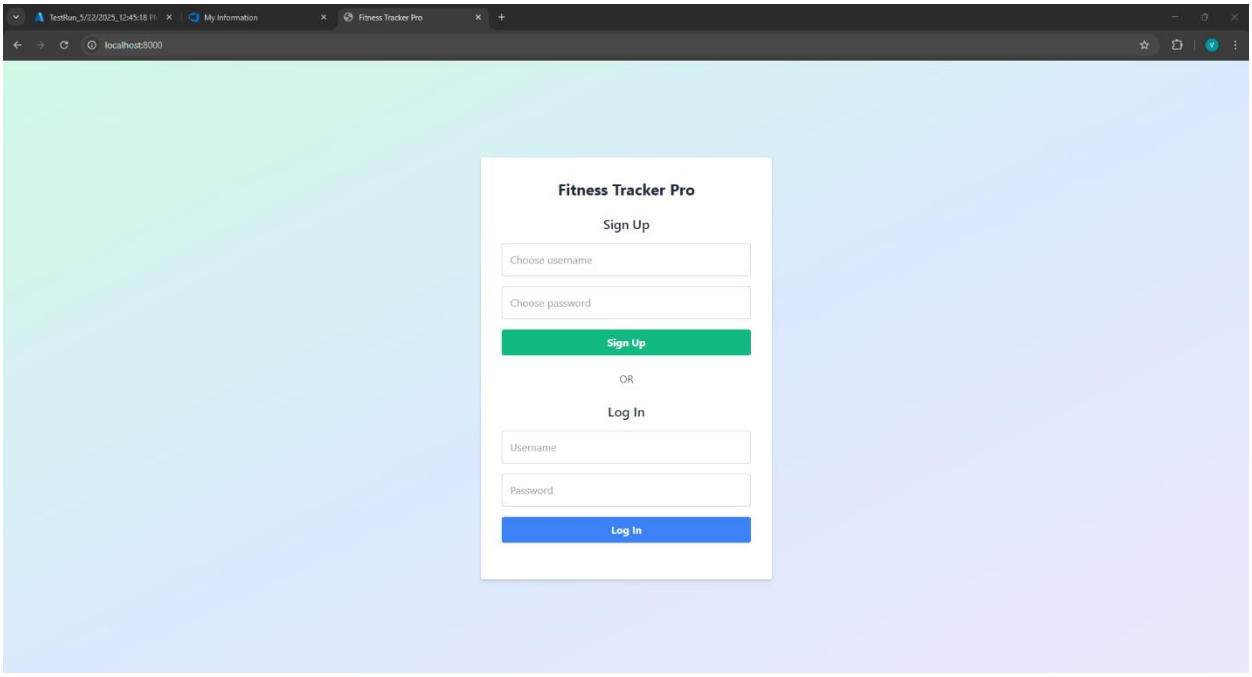
Once your resource is ready:

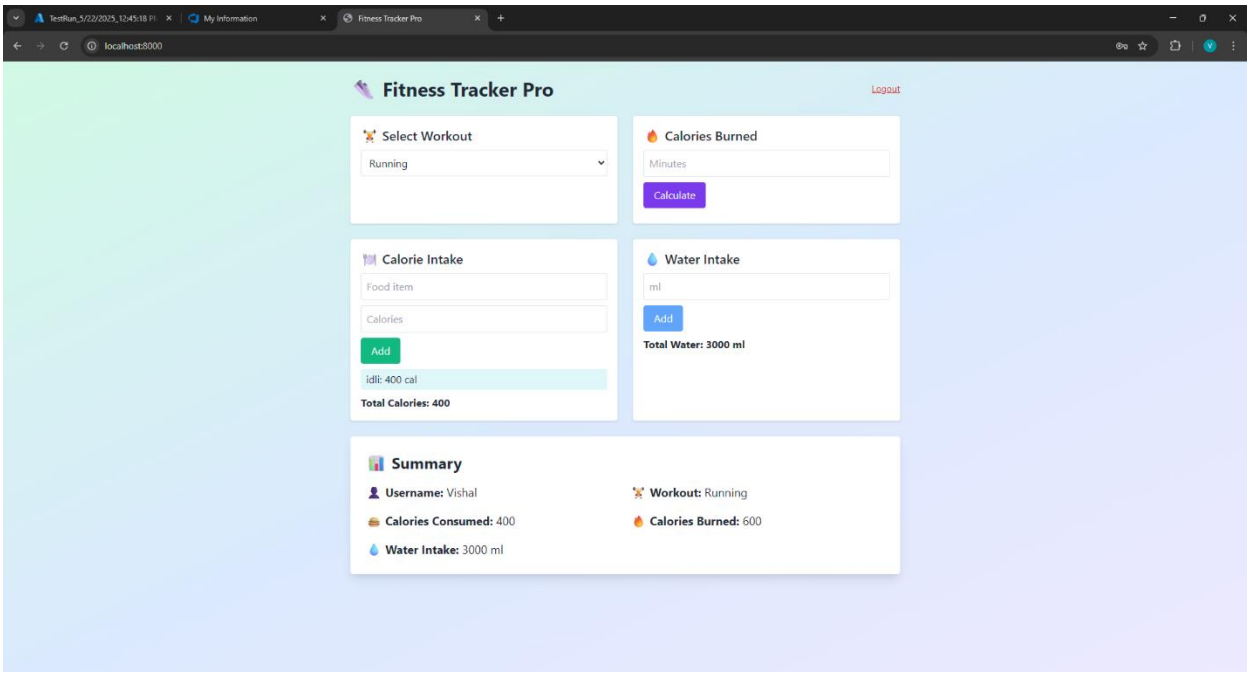
1. Go to your Azure Load Testing resource and click Add HTTP requests > Create.
2. Basics Tab
   - *Test Name:* Provide a unique name.
   - *Description:* (Optional) Add test purpose.
   - *Run After Creation:* Keep checked.
3. Load Settings
   - *Test URL:* Enter the target endpoint (e.g., https://yourapi.com/products).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

Load Testing









Result:

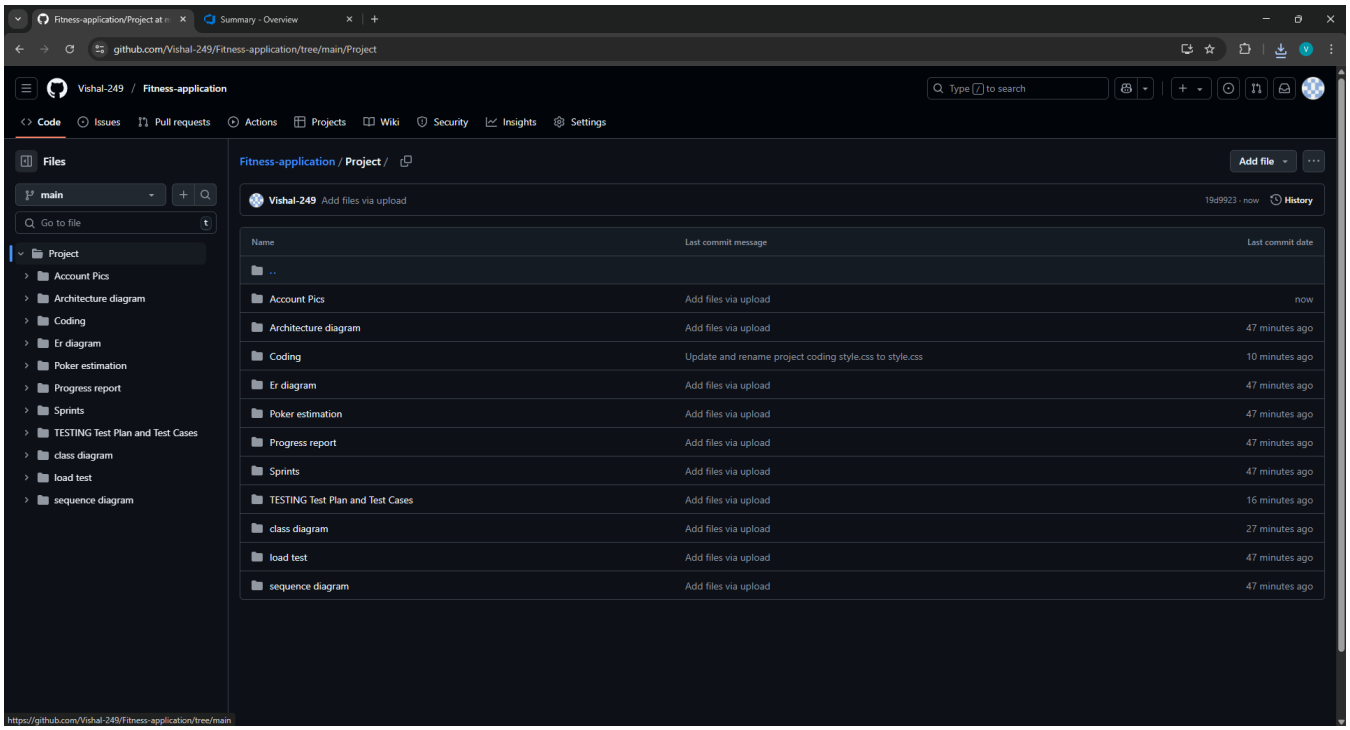
Successfully created the Azure Load Testing resource and executed a load test to assess the performance of the specified endpoint.

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 Music Playlist Batch Creator project.

GitHub Project Structure



Result:

The GitHub repository clearly displays the organized project structure and consistent naming conventions, making it easy for users and contributors to understand and navigate the codebase.