

EXP NO: 1

AZURE DEVOPS ENVIRONMENT SETUP

Aim:

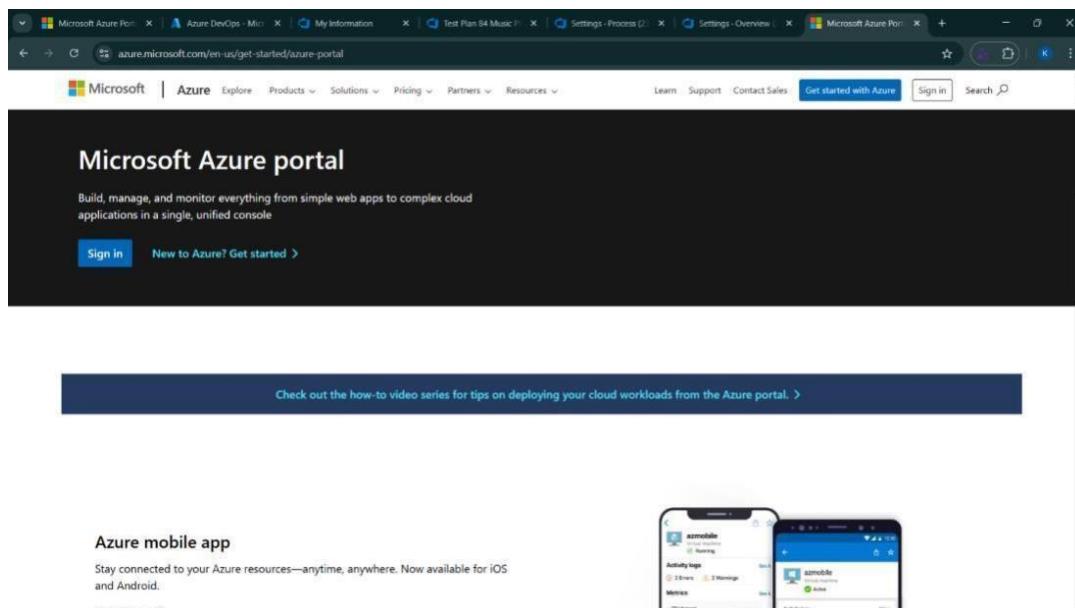
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

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>



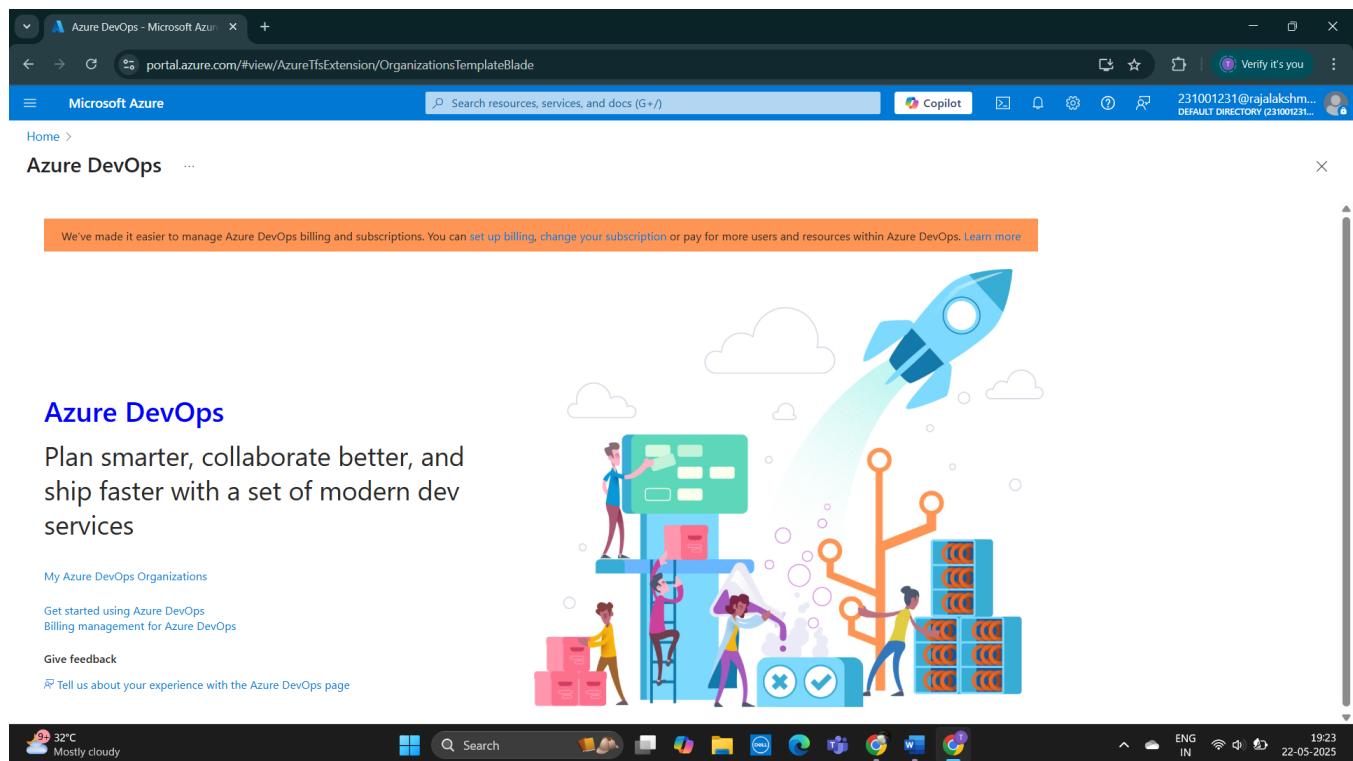
2. Azure home page

The screenshot shows the Microsoft Azure home page. At the top, there's a navigation bar with a back/forward button, a refresh icon, a search bar containing "portal.azure.com/#home", and a user profile icon. Below the navigation bar is a blue header bar with the text "Microsoft Azure". A search bar with the placeholder "Search resources, services, and docs (G+)" is located above the main content area. On the right side of the header, there are icons for Copilot, a gear (settings), a question mark (help), and a person (profile). The main content area starts with a "Welcome to Azure!" message and a note about free trials. It features three large cards: "Start with an Azure free trial" (blue key icon), "Manage Microsoft Entra ID" (shield icon), and "Azure for Students" (pencil writing on a tablet icon). Below these are sections for "Azure services" (with links to Create a resource, Azure DevOps organizations, Quickstart Center, Azure AI foundry, Kubernetes services, Virtual machines, App Services, Storage accounts, and SQL databases) and "Resources" (with links to Hot days ahead, Search, and other system icons). The bottom of the screen shows a taskbar with the Azure portal icon, a weather widget (32°C), and a system status bar.

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

This screenshot shows the Microsoft Azure home page again, but with a search term "Devops" entered into the search bar. The search results are displayed in a sidebar on the left, under the "Services" tab. The results include "Azure Native New Relic Service", "Managed DevOps Pools", "Azure DevOps organizations", "Azure Native Dynatrace Service", "Static Web App", "Rocky Linux 9", "Build Agents for Azure DevOps", and "Azure Native New Relic Service : New Relic Azure Cloud Monitoring - Start f...". Below the search results, there are sections for "Documentation" (with links to "Introduction to DevOps - Training" and "Course AZ-400T00-A: Designing and Implementing Microsoft DevOps solutions...") and "Continue searching in Microsoft Entra ID". The rest of the page is identical to the first screenshot, including the "Azure services" section and the bottom taskbar.

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.

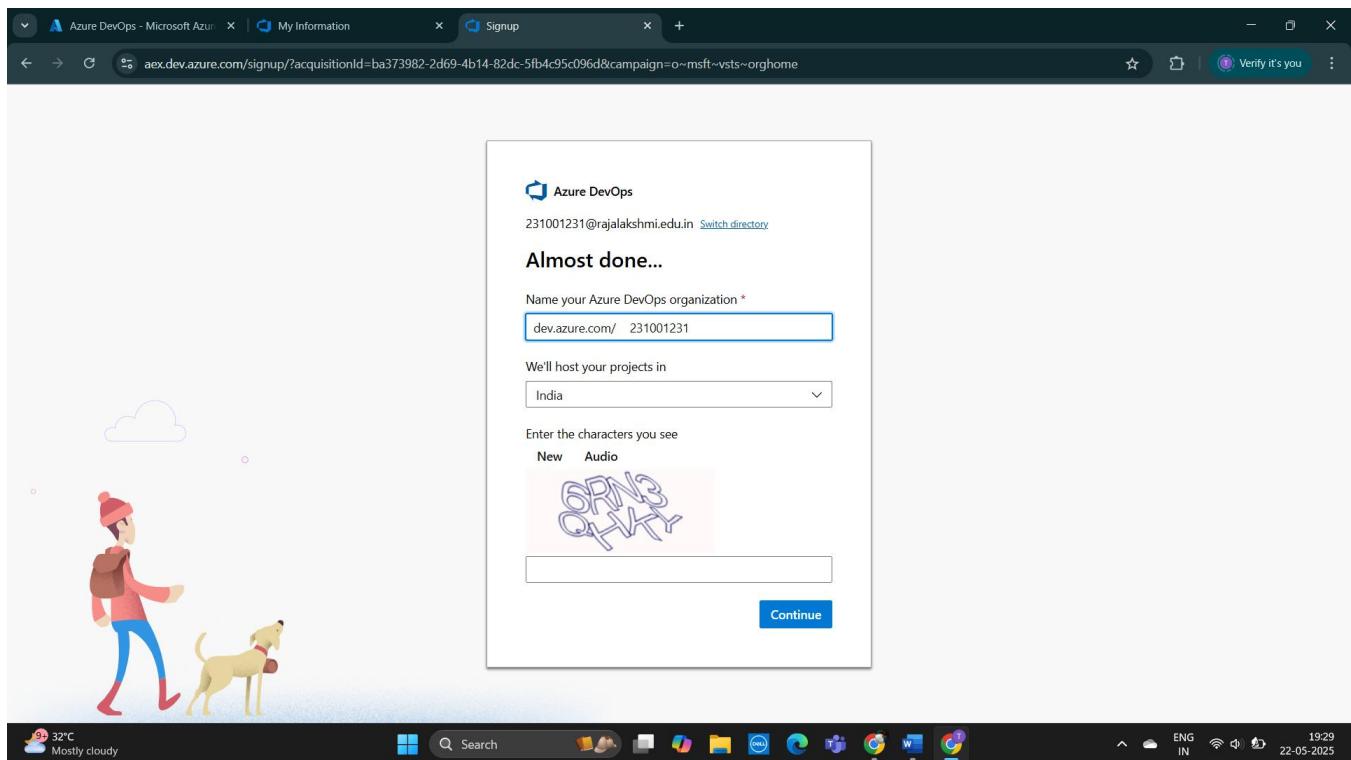
EXP NO: 2

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.

The screenshot shows the 'Create a project to get started' page in Azure DevOps. At the top right, there is a search bar with a magnifying glass icon and a red circular badge with the letter 'W'. To its right are icons for three-dot menu, refresh, help, and user profile.

Create a project to get started

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 [?](#)

+ Create 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.

The screenshot shows the Azure DevOps 'My Information' page. At the top, there's a purple circular profile picture with 'TV' in white. Below it, the name 'Tharun V' is displayed with the email '231001231@rajalakshmi.edu.in'. A dropdown menu shows 'Microsoft account'. Below this, location information is listed as 'India' with the email '231001231@rajalakshmi.edu.in'. A section titled 'Visual Studio Dev Essentials' offers benefits for building and deploying apps. On the right, the 'Azure DevOps Organizations' page is visible, showing 'dev.azure.com/231001231' (Owner) and 'dev.azure.com/231001248' (Member). A 'Create new organization' button is at the top right. The system tray at the bottom shows weather (32°C, mostly cloudy), system icons, and the date/time (22-05-2025).

4. Project dashboard

The screenshot shows the Azure DevOps 'Fitness Application' project dashboard. The left sidebar has a 'Fitness Application' header and links for Overview, Summary, Dashboards, Wiki, Boards, Repos, Pipelines, Test Plans, and Artifacts. The main content area has a 'Fitness Application' title and a 'About this project' section detailing features like User Registration and Authentication, Workout Plans and Recommendations, Diet and Nutrition Logging, Progress Tracking, Notifications and Reminders, and Security and Compliance. To the right, 'Project stats' show work items created (15), work items completed (0), pull requests opened (0), and commits by 0 authors. A 'Members' section lists five team members with their initials (VT, YV, VS, TV, YF). The system tray at the bottom shows weather (32°C, mostly cloudy), system icons, and the date/time (22-05-2025).

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 interface for the 'Fitness Application' project. The left sidebar includes options like Overview, Boards, Work Items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Artifacts. The main area displays the 'Backlog' for the 'Fitness Application Team'. A table lists 14 User Stories, each with a title, state (New), story point value (3-13), business area (Business), and iteration path (Fitness Application\Iteration 1). Buttons for 'New Work Item' and 'Add to top' are visible above the table. To the right, a 'Planning' section shows Iteration 1 with a planned effort of 126, Iteration 2 with no work scheduled yet, and Iteration 3 with no work scheduled yet. A 'New Sprint' button is also present. The bottom of the screen shows a taskbar with various icons and system status information.

The screenshot shows the Microsoft sign-in page. It features the Microsoft logo and a large purple circular profile picture containing a white 'TV'. The profile name is 'Tharun V' and the email address is '231001231@rajalakshmi.edu.in'. Below the profile, there are links for 'My Microsoft account' and 'Switch directory'. At the bottom, there's a link to 'Sign in with a different account' and a sign-in button.

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 'Fitness Application' project. The left sidebar is the navigation menu, and the main area is the 'Work items' list. The list contains 15 items, each with a small icon, ID, title, assignee, state, area path, and tags. The items are related to user login, registration, and various fitness features.

ID	Title	Assigned To	State	Area Path	Tags
16	Video Demonstrations for Exercises	Unassigned	New	Fitness Application	
15	AI-Based Workout Recommendations	Unassigned	New	Fitness Application	
41	Email Notification for Account Changes	Unassigned	New	Fitness Application	
9	Two-Factor Authentication (2FA)	Unassigned	New	Fitness Application	
8	Forgot Password	Unassigned	New	Fitness Application	
7	User Login with Credentials	Unassigned	New	Fitness Application	
6	User Registration with Email	Unassigned	New	Fitness Application	
40	Email Verification	Unassigned	New	Fitness Application	
39	code	Unassigned	New	Fitness Application	
38	Personalization and Smart Adaptation	Unassigned	New	Fitness Application	
36	User Customization and Adaptability	Unassigned	New	Fitness Application	
34	Email-Based User Registration & Account Setup	Unassigned	New	Fitness Application	
5	User Registration and Authentication	Unassigned	New	Fitness Application	
21	Diet and Nutrition Logging	Unassigned	New	Fitness Application	
13	Workout Plans and Recommendations	Unassigned	New	Fitness Application	

1. Fill in Epics

The screenshot shows the 'Edit work item' page for an 'EPIC' named 'User Registration and Authentication'. The page has tabs for 'Details', 'Development', and 'Related Work'. Under 'Details', there are sections for 'Description', 'Planning', and 'Deployment'. The 'Description' section has a note to add a comment. The 'Planning' section includes fields for Priority (set to 2), Risk, and Effort. The 'Deployment' section provides instructions on tracking releases. The 'Development' tab has a 'Add link' section for Azure Repos. The 'Related Work' tab shows a single linked task.

2. Fill in Features

The screenshot shows the Azure DevOps interface for a work item titled "FEATURE 38 Personalization and Smart Adaptation". The work item is in the "New" state, assigned to the "Fitness Application" area, and is part of Iteration 1. The "Description" section lists ten sub-items: 1. Food Diary & Meal Logging, 2. AI-Powered Meal Suggestions, 3. Calorie & Macronutrient Tracking, 4. Integration with Wearables & Health Apps, 5. Water Intake Tracking, 6. Meal Planning & Scheduling, 7. Food Insights & Analytics, 8. Diet Compatibility & Restrictions, 9. Community & Social Features, and 10. AI-Powered Chatbot for Nutrition Guidance. The "Planning" section includes fields for Priority (2), Risk, Effort, Business Value, Time Criticality, Start Date, and Target Date. The "Deployment" section provides instructions for tracking releases. The "Development" section includes an "Add link" button and a note about linking to Azure Repos. The "Related Work" section shows a linked work item for "Diet and Nutrition Logging".

3. Fill in User Story Details

The screenshot shows the Azure DevOps interface for a work item titled "USER STORY 40 Email Verification". The work item is in the "New" state, assigned to the "Fitness Application" area, and is part of Iteration 1. The "Description" section contains the text: "As a newly registered user, I want to verify my email address after registration, so that my account is activated and I can begin using the platform." The "Acceptance Criteria" section lists two items: 1. A verification email with a unique link is sent to the registered email. 2. Clicking the verification link activates the user's account. The "Planning" section includes fields for Story Points (2), Priority (2), and Risk. The "Deployment" section provides instructions for tracking releases. The "Development" section includes an "Add link" button and a note about linking to Azure Repos. The "Related Work" section shows a linked work item for "User Registration and Authentication".

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

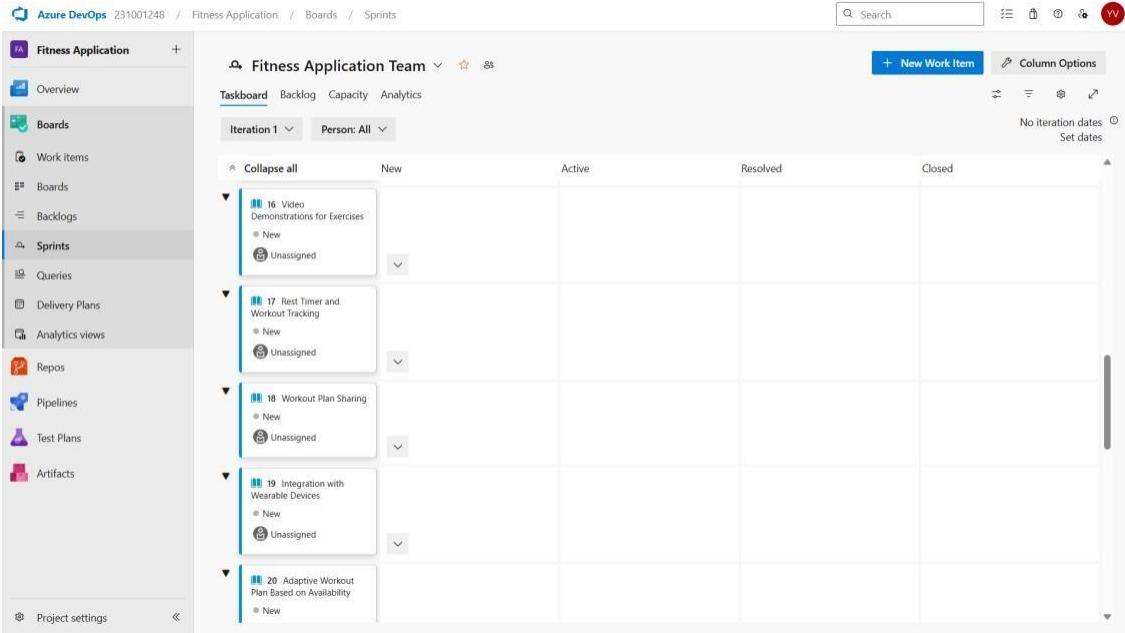
Sprint 1

The screenshot shows the Azure DevOps interface for the 'Fitness Application' project. The left sidebar is open with the 'Sprints' option selected. The main area displays a 'Taskboard' for the 'Fitness Application Team'. The 'Iteration 1' column is active, showing five user stories. Each story is represented by a card with a blue icon, a number, and a brief description. The first story is '22 Food Logging from Database' with status 'New' and 'Unassigned'. The second is '6 User Registration with Email' also 'New' and 'Unassigned'. The third is '7 User Login with Credentials' also 'New' and 'Unassigned'. The fourth is '8 Forgot Password' also 'New' and 'Unassigned'. The fifth is '9 Two-Factor Authentication (2FA)' also 'New' and 'Unassigned'. The columns represent 'New', 'Active', 'Resolved', and 'Closed'.

Sprint 2

The screenshot shows the Azure DevOps interface for the 'Fitness Application' project. The left sidebar is open with the 'Sprints' option selected. The main area displays a 'Taskboard' for the 'Fitness Application Team'. The 'Iteration 1' column is active, showing five user stories. Each story is represented by a card with a blue icon, a number, and a brief description. The first story is '10 Profile Management' with status 'New' and 'Unassigned'. The second is '11 Account Deletion' also 'New' and 'Unassigned'. The third is '12 Biometric Login' also 'New' and 'Unassigned'. The fourth is '14 Personalized Workout Plan Generation' also 'New' and 'Unassigned'. The fifth is '15 AI-Based Workout Recommendations' also 'New' and 'Unassigned'. The columns represent 'New', 'Active', 'Resolved', and 'Closed'.

Sprint 3

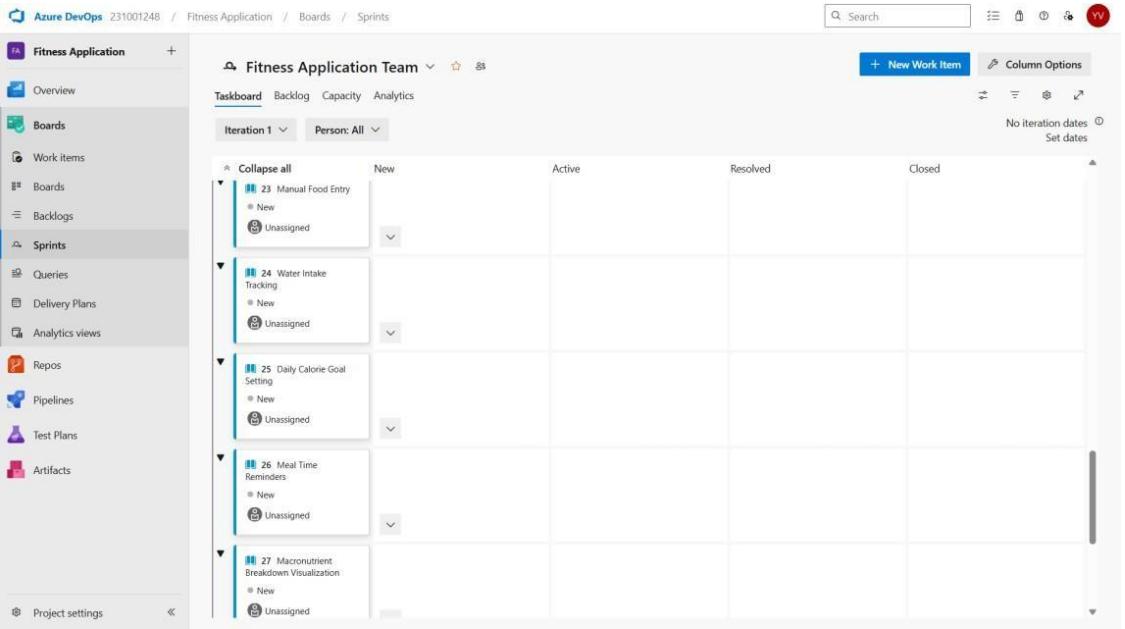


The screenshot shows the Azure DevOps Taskboard for Sprint 3. The left sidebar is titled "Fitness Application" and includes links for Overview, Boards, Work items, Backlogs, Sprints (selected), Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, Artifacts, and Project settings. The main area is titled "Fitness Application Team" and shows a taskboard for "Iteration 1". The columns are labeled "New", "Active", "Resolved", and "Closed". There are five collapsed work items:

- 16 Video Demonstrations for Exercises (New, Unassigned)
- 17 Rest Timer and Workout Tracking (New, Unassigned)
- 18 Workout Plan Sharing (New, Unassigned)
- 19 Integration with Wearable Devices (New, Unassigned)
- 20 Adaptive Workout Plan Based on Availability (New, Unassigned)

At the top right, there are buttons for "New Work Item" and "Column Options", and a message stating "No iteration dates Set dates". A search bar is at the top center.

Sprint 4



The screenshot shows the Azure DevOps Taskboard for Sprint 4. The left sidebar is identical to Sprint 3. The main area is titled "Fitness Application Team" and shows a taskboard for "Iteration 1". The columns are labeled "New", "Active", "Resolved", and "Closed". There are five collapsed work items:

- 23 Manual Food Entry (New, Unassigned)
- 24 Water Intake Tracking (New, Unassigned)
- 25 Daily Calorie Goal Setting (New, Unassigned)
- 26 Meal Time Reminders (New, Unassigned)
- 27 Macronutrient Breakdown Visualization (New, Unassigned)

At the top right, there are buttons for "New Work Item" and "Column Options", and a message stating "No iteration dates Set dates". A search bar is at the top center.

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

The screenshot shows a user story card for "Food Logging from Database".

Header: USER STORY 22
22. Food Logging from Database

Details:
State: New
Reason: New
Area: Fitness Application
Iteration: Fitness Application\Iteration 1

Description:
As a user, I want to log my daily meals from a food database, so that I can track my calorie intake easily.

Acceptance Criteria:
1. Users can search for food items from a preloaded database.
2. Users can add multiple food items to a meal.
3. The system automatically calculates calories, protein, carbs, and fat.

Planning:
Story Points: 3
Priority: 2
Risk:

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](#)

Classification:
Value area: Business

Development:
Add link: Link an Azure Repos [commit](#), [pull request](#) or [branch](#) to see the status of your development. You can also [create a branch](#) to get started.

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

Related Work:
Add link: Parent
38 Personalization and Smart Adaptation
Updated Mar 27 • New

Result:

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

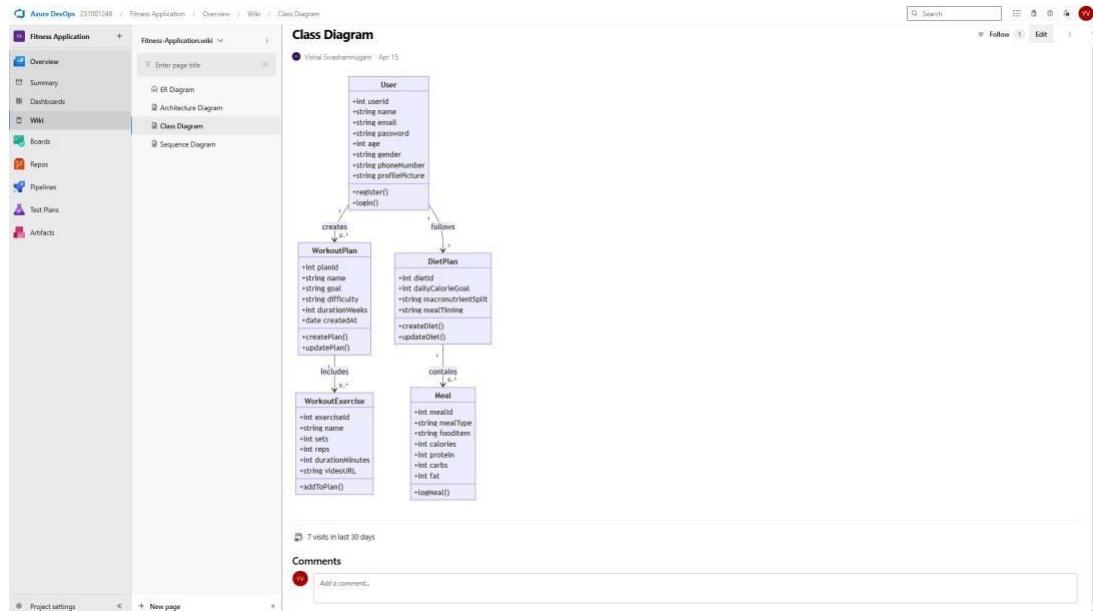
EXP NO: 6

DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE

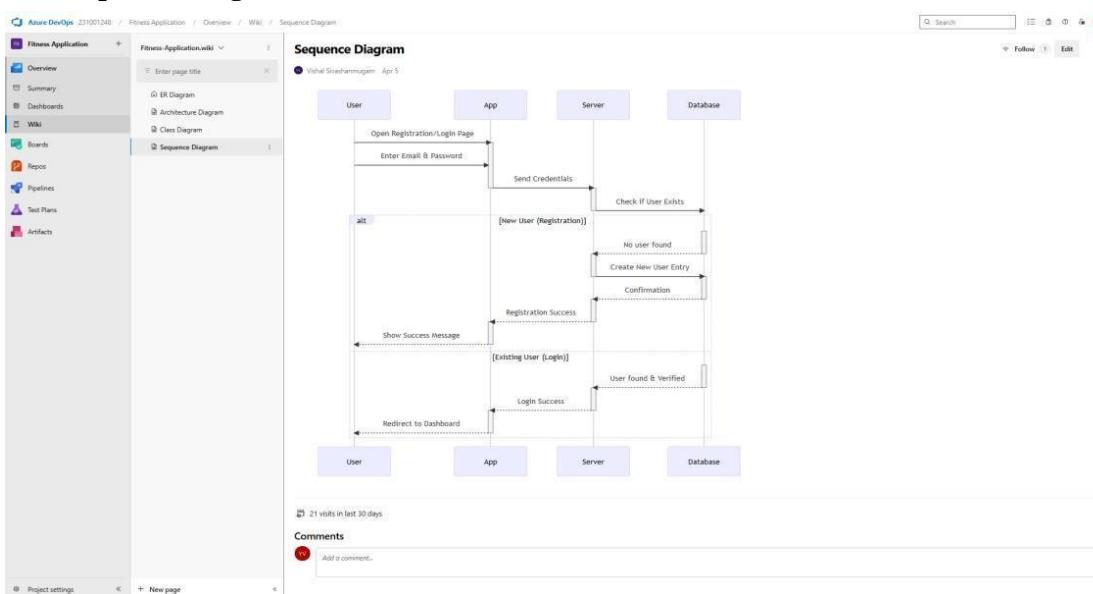
Aim:

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.

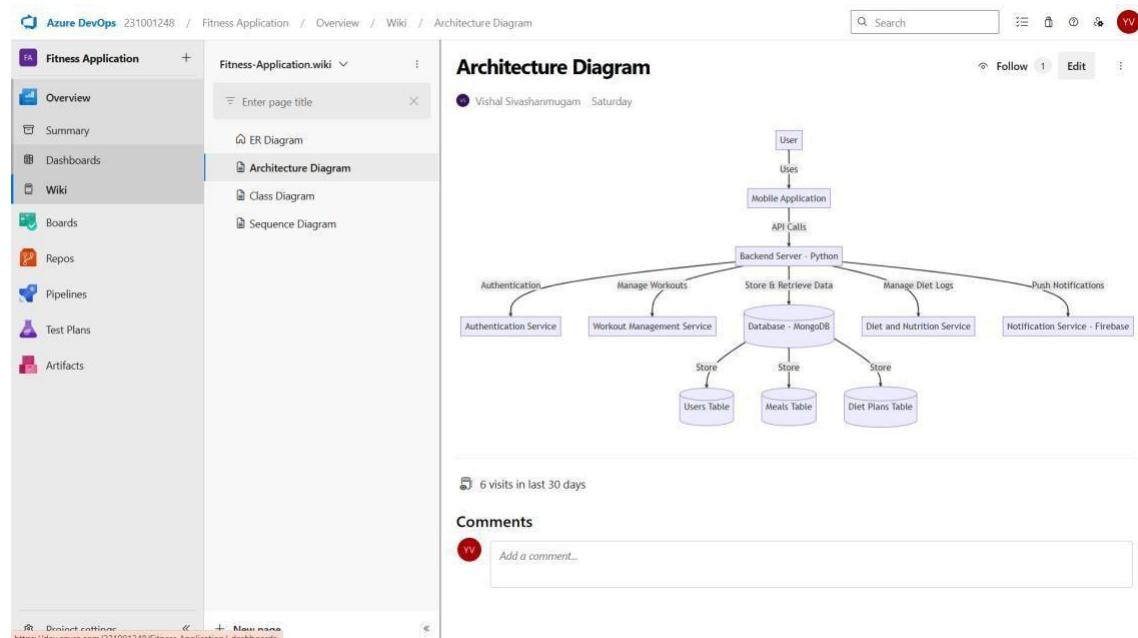
EXP NO: 7

DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE

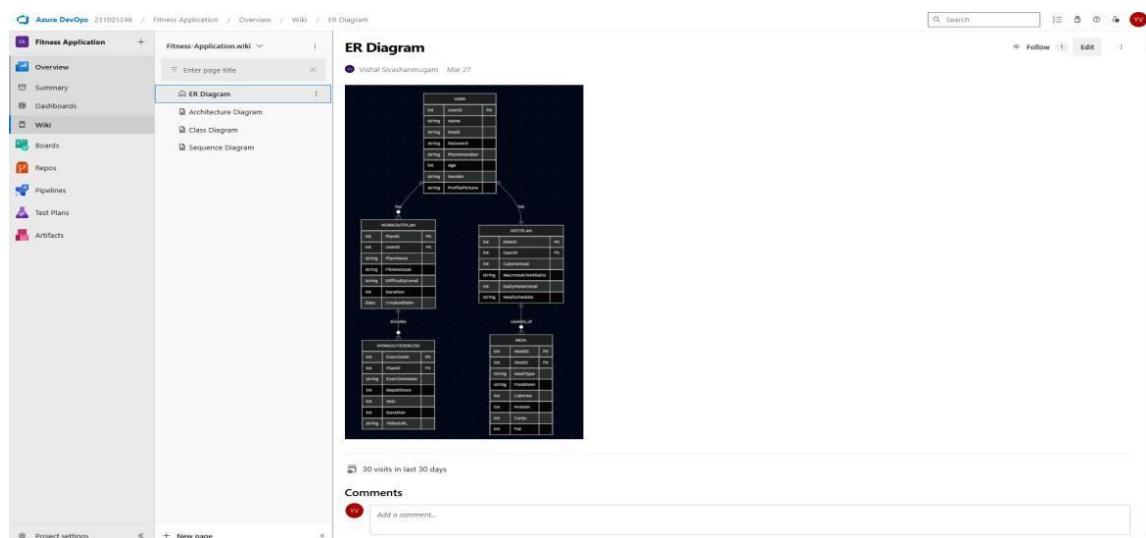
Aim:

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 traceability.

1. New test plan

The screenshot shows the 'New Test Plan' creation interface in Azure DevOps. The left sidebar is for the 'Fitness Application' project, showing options like Overview, Boards, Repos, Pipelines, Test Plans, and Artifacts. The 'Test Plans' section is currently selected. The main area is titled 'New Test Plan' and contains fields for 'Name' (set to 'Fitness Application test plan'), 'Area Path' (set to 'Fitness Application'), and 'Iteration' (set to 'Fitness Application'). At the bottom are 'Create' and 'Cancel' buttons.

2. Test suite

The screenshot shows the 'User Registration and Authentication' test suite execution page. The left sidebar is for the 'Fitness Application' project. The main area displays the 'User Registration and Authentication (ID: 44)' test suite details, including a 'Test Points (3 items)' table. The table lists three test points, all of which have passed:

Title	Outcome	Order	Test Case Id	Configuration	Tester
User Registration with Email	Passed	2	45	Windows 10	Vishal Sivas
User Login with Credentials	Passed	3	46	Windows 10	Vishal Sivas
Forgot Password	Passed	4	47	Windows 10	Vishal Sivas

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

The screenshot shows the Azure DevOps Test Plan interface. The URL in the browser is dev.azure.com/231001248/Fitness%20Application/_testPlans/execute?planId=42&suiteId=44. The page displays a test case titled '45 - User Registration with Email'. The test case details include:

- Owner: Vishal Sivashanmugam
- Status: Design
- Area: Fitness Application
- Iteration: Fitness Application\Iteration 1
- Reason: New

The 'Steps' section contains three steps:

Step	Action	Expected result	Attachments
1.	Users can enter their email and create a password.	Login Successful	
2.	The system sends a verification email.	Email verification sent	
3.	Users can complete registration only after email verification.	Verified	

Below the steps, there is a note: "Click or type here to add a step".

The right side of the screen shows sections for Deployment, Development, and Related Work, with various status indicators and links.

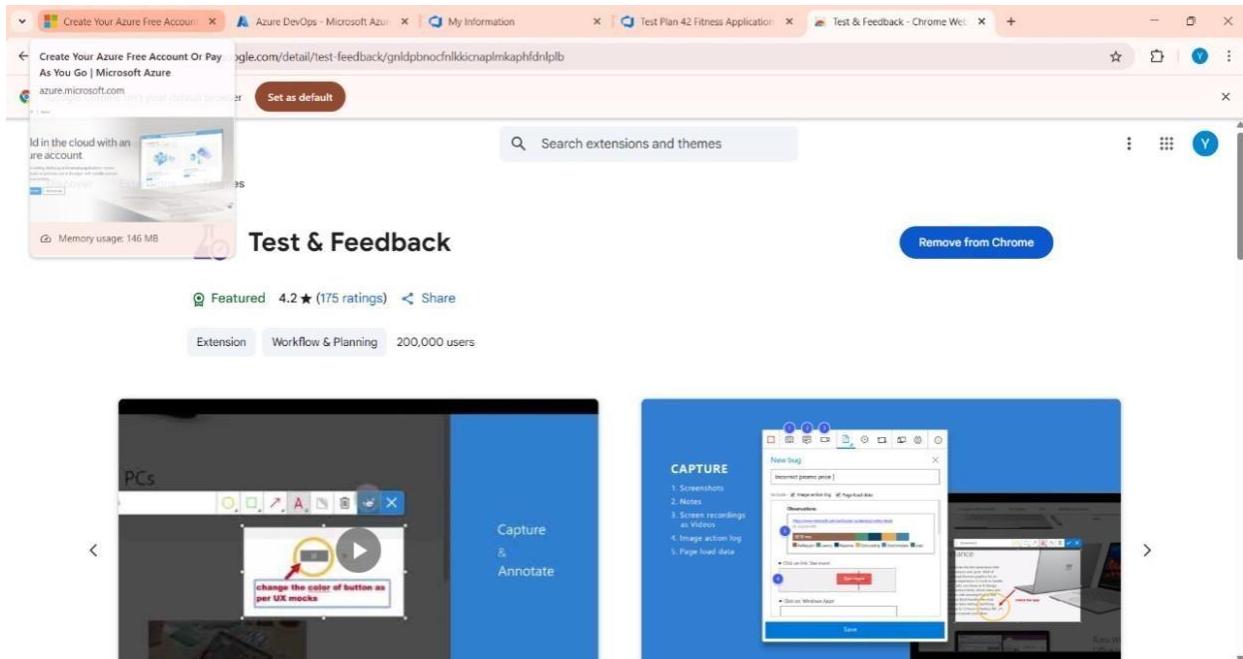
The screenshot shows the Azure DevOps Test Plan interface for a Fitness Application. A specific test case, 'User Login with Credentials' (ID 46), is selected. The test plan details include:

- Design**: State is 'Design', Area is 'Fitness Application', Iteration is 'Fitness Application\Iteration 1'.
- Steps**:
 - Step 1: 'The system should allow valid users to log in.' Expected result: 'Logged in'.
 - Step 2: 'Incorrect credentials should return an error message.' Expected result: 'Incorrect credentials'.
- Deployment**: A note about tracking releases associated with this work item.
- Development**: A note about linking to Azure Repos for commit, pull request, or branch status.
- Related Work**: An option to add a link to an existing work item.
- Status**: Updated by Vishal Sivashanmugam 8h ago.

4. Installation of test

The screenshot shows the Chrome Web Store page for the 'Test & Feedback' extension. Key details include:

- Extension**: Test & Feedback - Chrome Webstore
- Rating**: 4.2 stars (175 ratings)
- Users**: 200,000 users
- Description**: Capture & Annotate
- Screenshots**: Two screenshots are shown, demonstrating the extension's annotation and capture features.
- Capture**: A sidebar menu with options like Screenshots, Notes, Screen recordings as Videos, Image action log, and Page load data.
- Add to Chrome**: A prominent button to add the extension to a Chrome browser.



Test and feedback

Showing it as an extension

The screenshot shows the Azure DevOps Test Plans interface. On the left, the navigation bar includes 'Fitness Application', 'Test plans', and 'Test suites'. In the center, a 'User Registration and Authentication (ID: 44)' test plan is displayed with 'Test Points (3 items)'. On the right, a floating 'Extensions' panel is open, showing 'Test & Feedback' under 'Full access' extensions. The 'Manage extensions' button is also visible.

5. Running the test cases

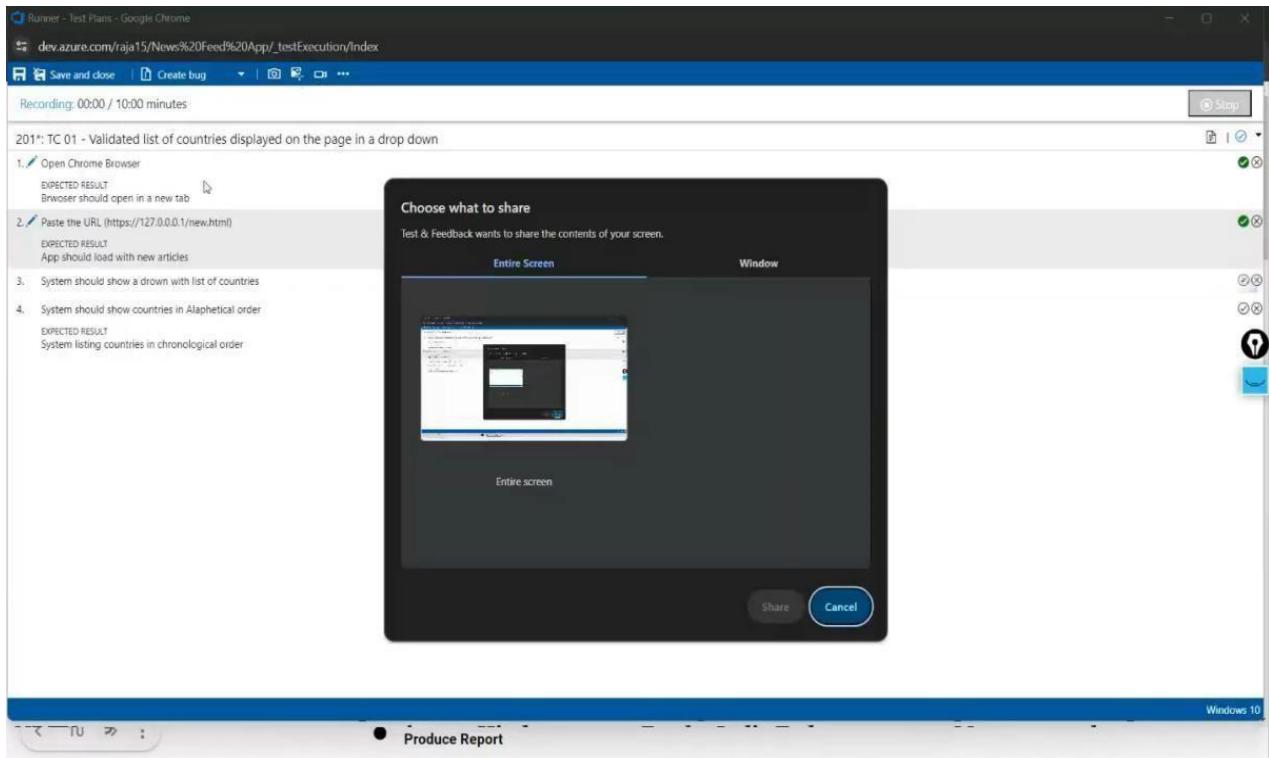
The screenshot shows the Azure DevOps Test Plan interface for a 'Fitness Application'. The left sidebar is titled 'Fitness Application' and includes sections for Overview, Boards, Repos, Pipelines, Test Plans, Test plans, Progress report, Parameters, Configurations, Runs, and Artifacts. The 'Test Plans' section is currently selected. The main area displays a 'User Registration and Authentication (ID: 44)' test plan. A context menu is open over a 'User Registration with Email' test point, listing options such as View execution history, Mark Outcome, Run, Reset test to active, Edit test case, Assign tester, and View test result. Below the test point, a table lists three test cases with columns for Title, Outcome (all Passed), Order (2, 3, 4), Test Case Id (45, 46, 47), Configuration (Windows 10), and Tester (Vishal Sivas). The URL in the browser is dev.azure.com/231001248/Fitness%20Application/_testPlans/execute?planId=44&suiteId=44.

The screenshot shows the 'Runner - Test Plans - Google Chrome' window displaying the results of a 'User Registration with Email' test. The test steps are listed as follows:

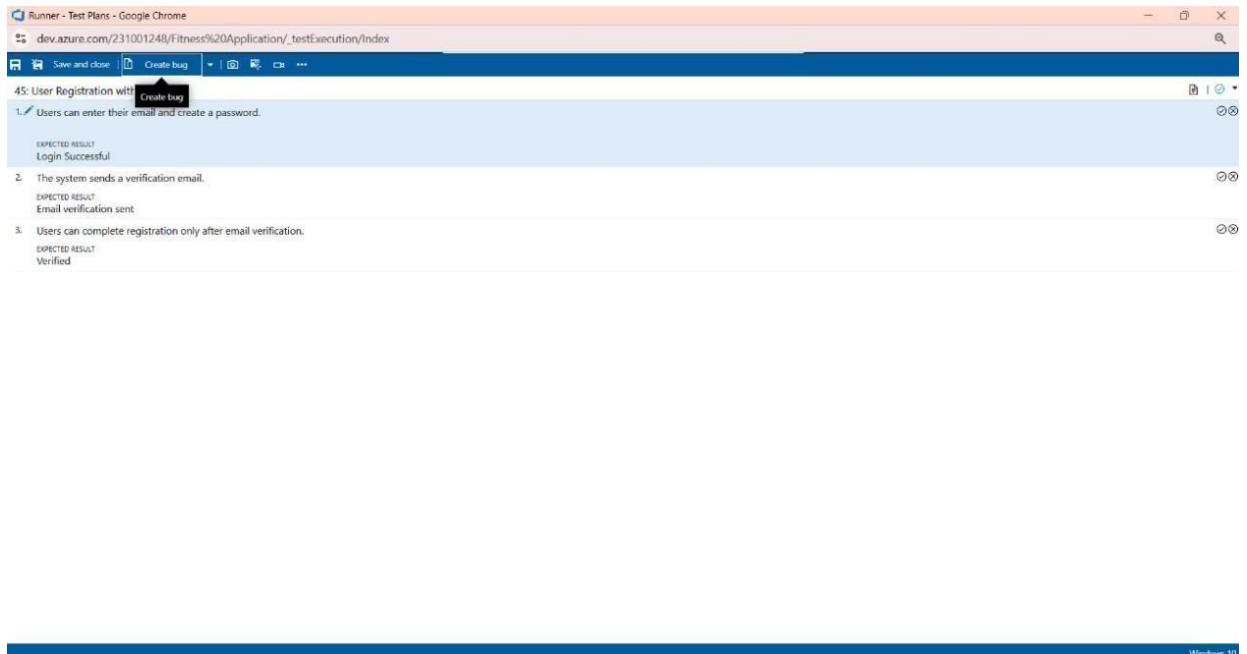
1. Users can enter their email and create a password.
EXPECTED RESULT: Login Successful
2. The system sends a verification email.
EXPECTED RESULT: Email verification sent
3. Users can complete registration only after email verification.
EXPECTED RESULT: Verified

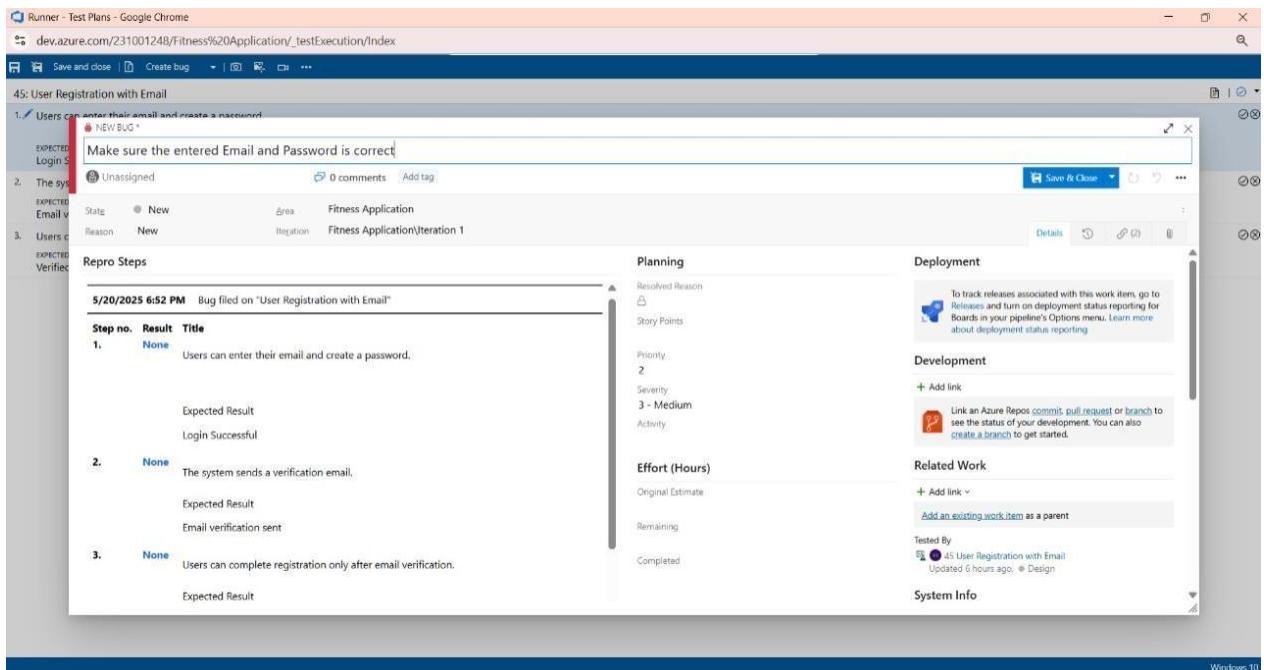
The status for each step is indicated by a green checkmark icon. The URL in the browser is dev.azure.com/231001248/Fitness%20Application/_testExecution/Index.

6. Recording the test case



7. Creating the bug





The screenshot shows a Microsoft Edge browser window titled "Create Your Azure Free Ac..." with the URL "dev.azure.com/231001248/Fitness%20Application/_testManagement/runs?_a=runCharts&runId=2". The page displays a test run summary for "Run 2 - User Registration and Authentication (Manual)".

Run Summary:

Run 2 - User Registration and Authentication (Manual)

Completed 6 hours ago. Ran for 01 minutes 50 seconds.

Run Type: Manual
Owner: Vidal Sivashanmugam
Tested build: not available
Release: not available
Release Stage: not available
Build platform: not available
Build flavor: not available
Test settings: MTM lab environment
Run time settings: not available

Comments:

No comments
Error message
No error message

Attachments: (0)

Outcome:

1 Passed

Outcome by priority:

2

8. Test case results

The screenshot shows the Azure DevOps interface for a 'Fitness Application' test plan. The left sidebar is the 'Test Plans' section, with 'Test plans' selected. The main area displays the 'User Registration and Authentication' test suite (ID: 44). Under 'Test Points (3 items)', the 'User Login with Credentials' point is selected and highlighted in blue. A modal window titled 'Test Case Results' is open, showing a single row for 'User Login with Credentials'. The row includes columns for Outcome (Passed), TimeSta... (6h ago), Configuration (Windows 10), Run by (Vishal Sivashan...), and Tester (Vishal Sivashan...). A link 'Open execution history for current test point' is at the bottom of the modal.

9. Test report summary

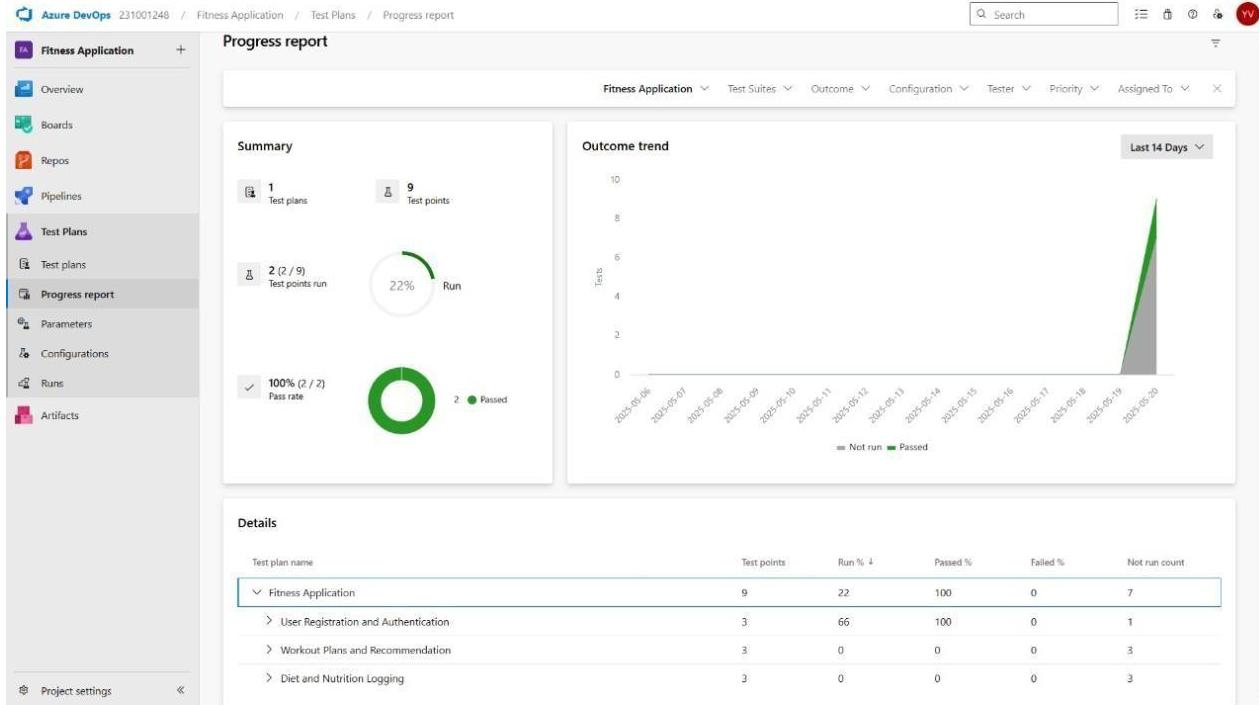
The screenshot shows the Azure DevOps interface for a 'Fitness Application' work item. The left sidebar is the 'Work items' section, with 'Work items' selected. The main area displays a work item titled 'TEST CASE 45' for 'User Registration with Email'. The work item details include: Status (Design), Area (Fitness Application), Reason (New), Iteration (Fitness Application/Iteration 1). The 'Steps' tab is active, showing three steps: 1. Users can enter their email and create a password. (Expected result: Login Successful). 2. The system sends a verification email. (Expected result: Email verification sent). 3. Users can complete registration only after email verification. (Expected result: Verified). Other tabs include 'Summary', 'Associated Automation', 'Deployment', 'Development', 'Related Work', and 'Status'. A note in the 'Deployment' section says: '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.' A note in the 'Development' section says: 'Add link' and '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.' A note in the 'Related Work' section says: 'Add link' and 'Add an existing work item as a parent.' A note in the 'Status' section says: 'Add a comment'.

- Assigning bug to the developer and changing state

The screenshot shows the Azure DevOps interface for a project named 'Fitness Application'. The left sidebar includes links for Overview, Boards, Work items, Boards, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, Artifacts, and Project settings. The main content area displays a work item titled 'BUG 57' with the description '57 Make sure the entered Email and Password is correct'. The work item status is 'New', assigned to 'Fitness Application', and the iteration is 'Fitness Application\Iteration 1'. The 'Repro Steps' section contains three steps: 1. 'None' - 'Users can enter their email and create a password.'; 2. 'None' - 'The system sends a verification email.'; 3. 'None' - 'Users can complete registration only after email verification.' The 'Planning' section shows a resolved reason, story points (2), priority (2), severity (3 - Medium), and activity. The 'Deployment' section provides instructions on tracking releases. The 'Development' section includes a link to a commit or branch. The 'Effort (Hours)' section shows original estimate, remaining, and completed hours. The 'Related Work' section allows adding links to existing work items.

10. Progress report

The screenshot shows the Azure DevOps interface for a project named 'Fitness Application'. The left sidebar includes links for Overview, Boards, Repos, Pipelines, Test Plans, Parameters, Configurations, Runs, Artifacts, and Project settings. The main content area displays a 'Progress report' for the 'Fitness Application' test plan. It shows a summary with 1 test plan, 9 test points, 2 test points run (22%), and a 100% pass rate (2/2). A circular progress bar indicates 2 Passed. The 'Outcome trend' chart tracks test results from May 26 to June 20, showing a sharp increase in successful runs starting around May 29. The chart includes a legend for 'Not run' (grey) and 'Passed' (green).



11. Changing the test template

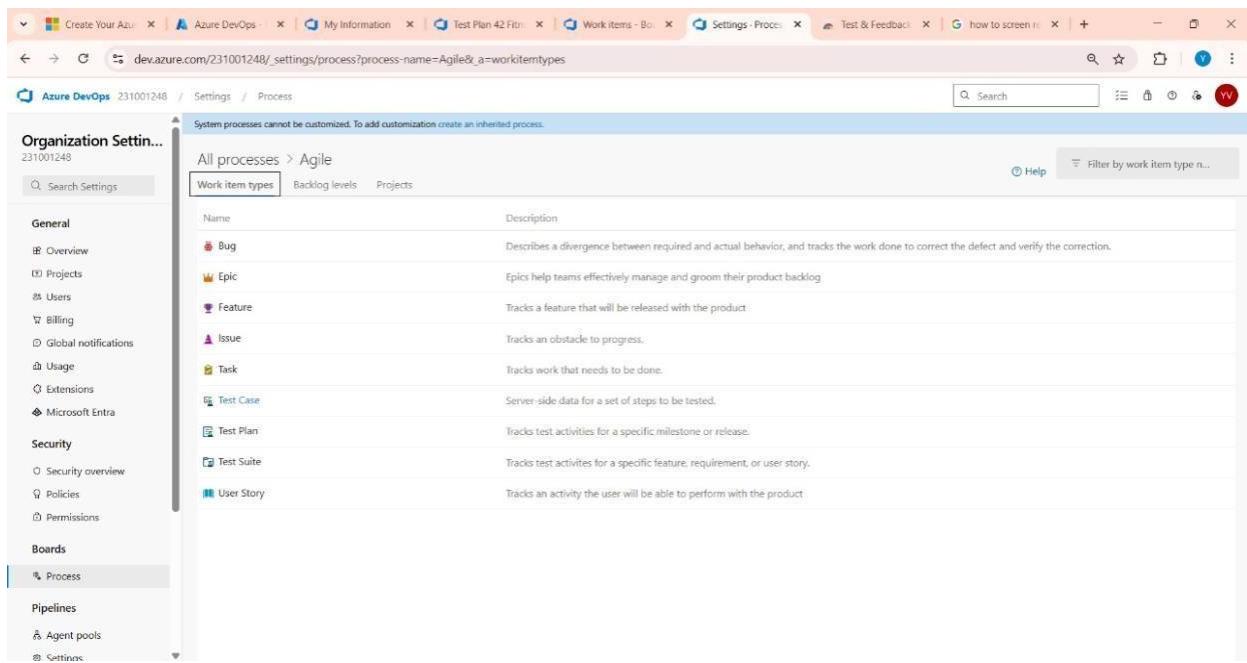
All processes

Name	Description	Team projects
Basic (default)	This template is flexible for any process and great for teams getting started with 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

The screenshot shows the Azure DevOps Settings - Process page. On the left, there's a sidebar with sections like General, Security, Boards, Pipelines, and Process (which is selected). The main area is titled "All processes" and lists four templates:

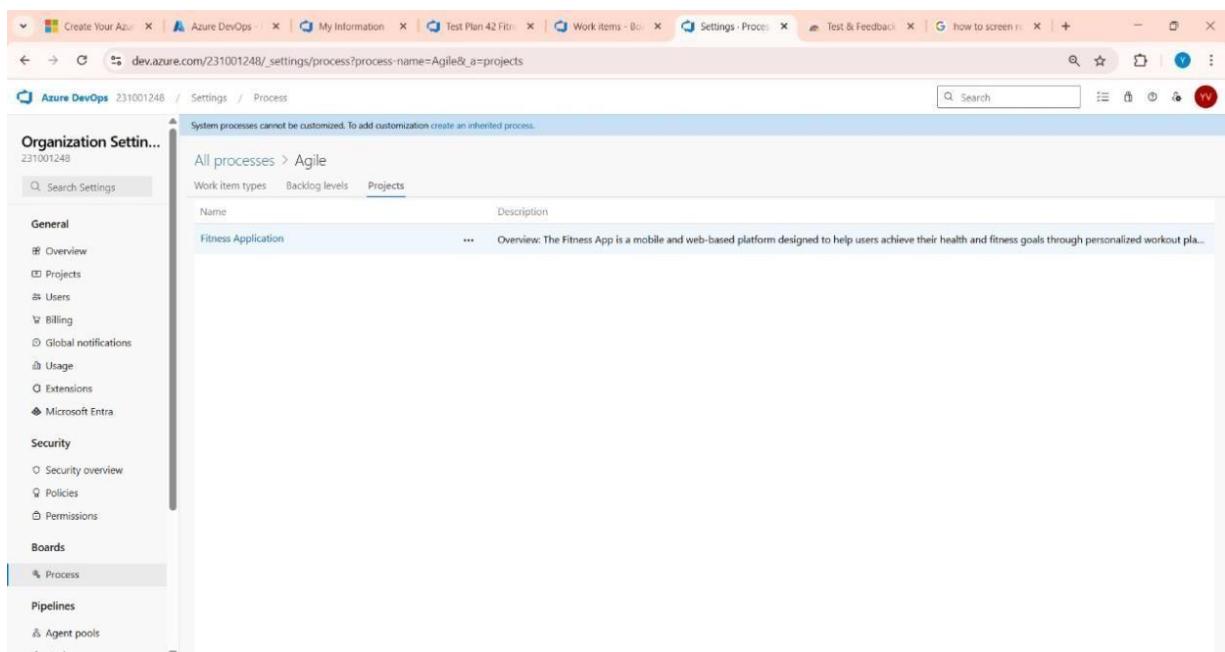
Name	Description	Team projects
Basic (default)	This template is flexible for any process and great for teams getting started with Azure DevOps.	0
Agile	This template is flexible and will work great for most teams using Agile planning methods, including those pract...	1
Scrum	This template is for teams 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



The screenshot shows the 'Organization Settings' page for a project named '231001248'. The left sidebar is collapsed, and the main content area displays the 'All processes > Agile' section. Under the 'Work item types' tab, there is a table listing the following work item types:

Name	Description
Bug	Describes a divergence between required and actual behavior, and tracks the work done to correct the defect and verify the correction.
Epic	Epics help teams effectively manage and groom their product backlog.
Feature	Tracks a feature that will be released with the product.
Issue	Tracks an obstacle to progress.
Task	Tracks work that needs to be done.
Test Case	Server-side data for a set of steps to be tested.
Test Plan	Tracks test activities for a specific milestone or release.
Test Suite	Tracks test activities for a specific feature, requirement, or user story.
User Story	Tracks an activity the user will be able to perform with the product.



The screenshot shows the same 'Organization Settings' page for project '231001248'. The left sidebar is collapsed, and the main content area displays the 'All processes > Agile' section. Under the 'Work item types' tab, there is a table listing the following work item types. A new entry, 'Fitness Application', has been added to the list:

Name	Description
Fitness Application	... Overview: The Fitness App is a mobile and web-based platform designed to help users achieve their health and fitness goals through personalized workout pla...

The screenshot shows the Azure DevOps Settings - Process page for the Agile test case. The left sidebar lists various settings categories like General, Security, Boards, Pipelines, and Process (which is selected). The main area displays the Agile Test Case configuration with tabs for Steps, Summary, and Associated Aut... (Summary is selected). The Steps tab contains a 'Text (multiple lines)' input field with placeholder 'Test (multiple lines)'. To the right, there are sections for Recent test results, Deployment, Development (with a 'Links' section), Related Work, Status, Priority (integer), and Automation status (text single line). A 'Add a field ...' button is located at the bottom right of the configuration 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 PERFORMANCE TESTING

Aim:

To develop a fitness application that helps users track workouts, set goals, and monitor their health.

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

TestRun_5/22/2025, 12:45:18 PM

Last updated by: 231001248@rajalakshmi.edu.in | Initiated on: 5/22/2025, 12:45 PM

View all test runs Stop Refresh Rerun Compare App components Configure metrics Download Copy artifacts Share Delete test run Mark as baseline Auto refresh on (10s)

Test run details

Start time	End time	Test run ID	Test type	Engine instances	Test result	Status
5/22/2025, 12:45:28 PM	5/22/2025, 12:49:16 PM	ec13a494-8a1e-429d-b741-dc94909f1020/testid/7f527efe-60c6-456a-9710-c6ce6a35d036/resourceId/%2Fsubscriptions%2F6e99c0c1-e723-4fb2-a505-908857e62aa...	URL	1	Not Applicable	Failed (View details)

Load test results Engine health

Test criteria

Client-side metrics

Metric	Aggregation	Condition	Threshold	Request name	Actual value	Result
Error	Percentage	Greater than	50			

Virtual Users (Max)

Request1
47

Response time (successful responses)

Request1 Pct...
1 ms

TestRun_5/22/2025, 12:45:18 PM

Last updated by: 231001248@rajalakshmi.edu.in | Initiated on: 5/22/2025, 12:45 PM

View all test runs Stop Refresh Rerun Compare App components Configure metrics Download Copy artifacts Share Delete test run Mark as baseline Auto refresh off

Client-side metrics

Virtual Users (Max)

Request1
47

Requests/sec (Avg)

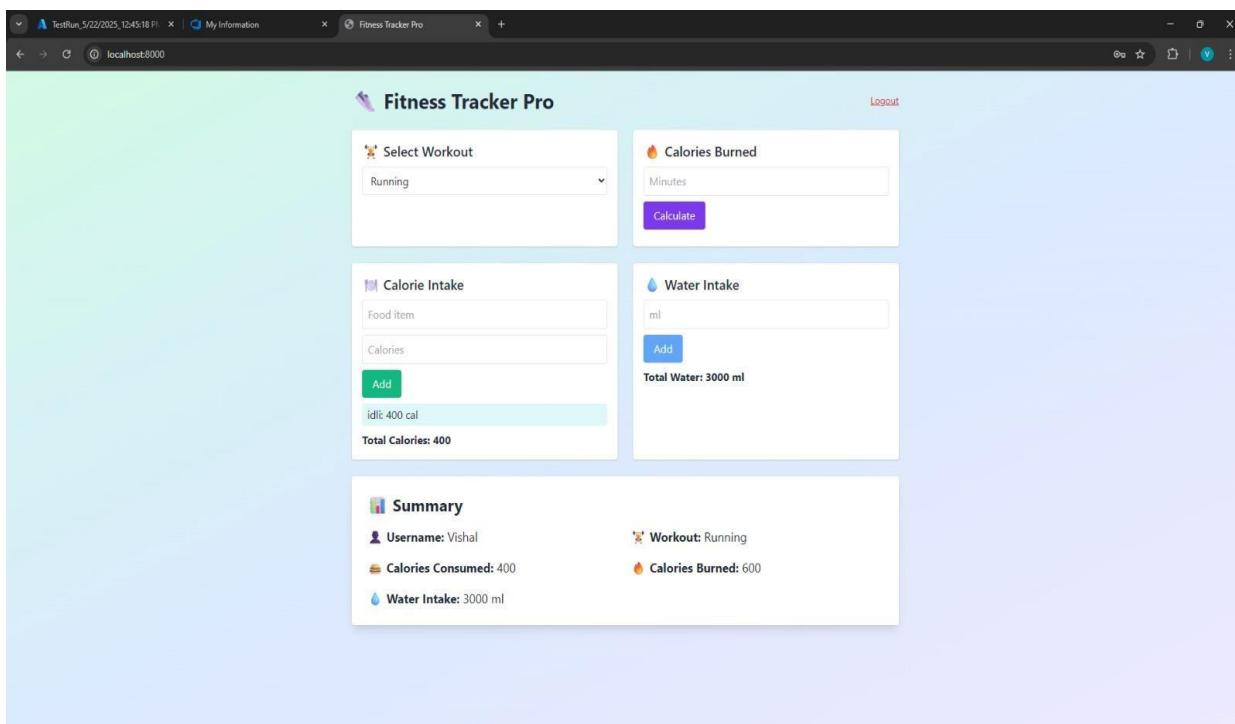
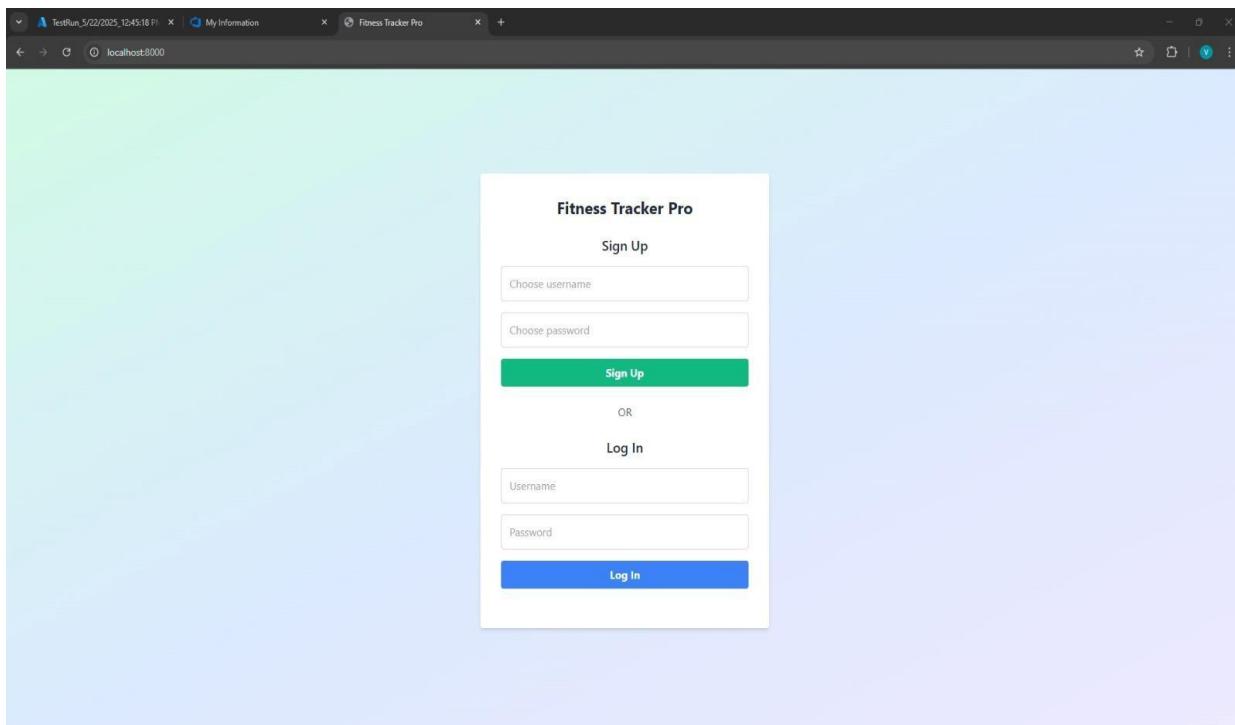
Request1
6.55 k/s

Errors (total)

Request1 Non-HTTP reqs...
458.49 k

Request1 HTTP reqs...
1

Aggregate
458.49 k



Result:

Successfully designed and developed a fitness application that enables users to track workouts, set fitness goals, and monitor health metrics to support a healthier lifestyle.

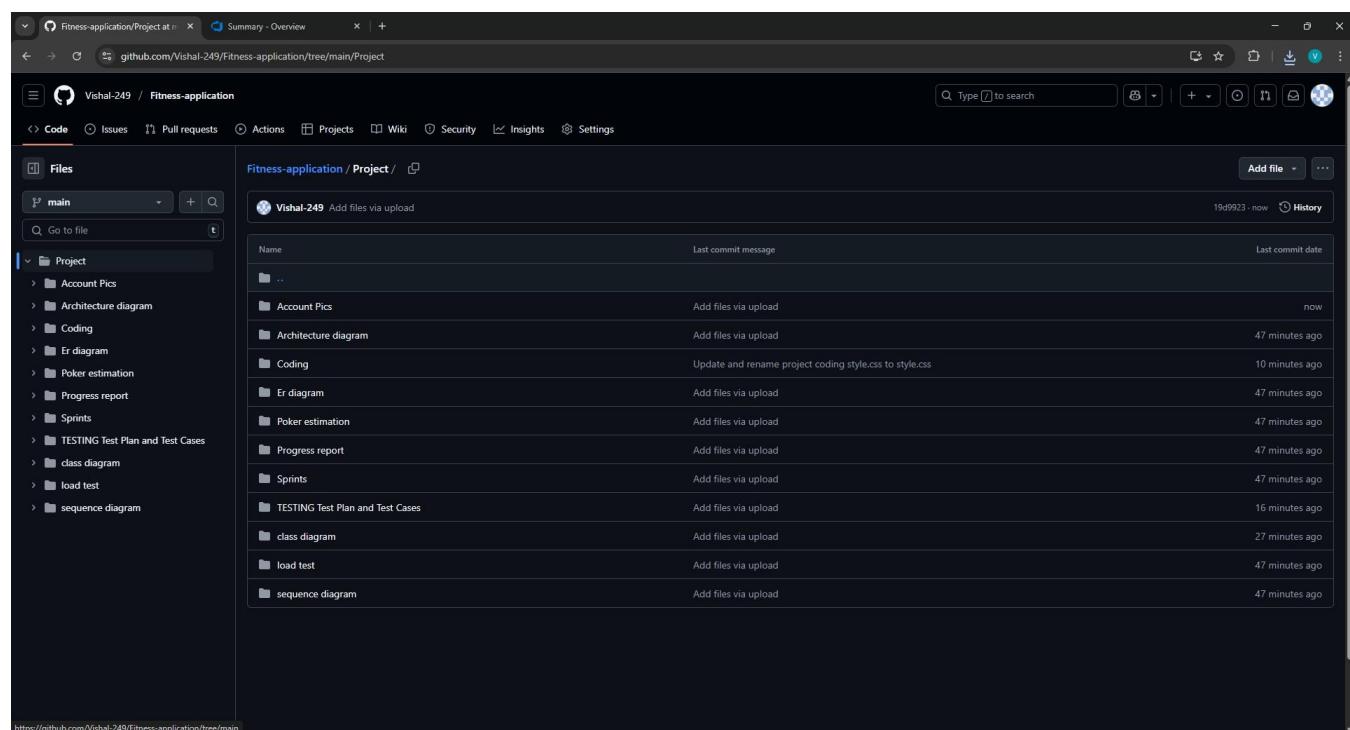
EXP NO: 10

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 Fitness Application project.

GitHub Project Structure



The screenshot shows a GitHub repository named 'Fitness-application' with a dark theme. The left sidebar displays a hierarchical project structure under the 'Project' folder, including 'Account Pics', 'Architecture diagram', 'Coding', 'Er diagram', 'Poker estimation', 'Progress report', 'Sprints', 'TESTING Test Plan and Test Cases', 'class diagram', 'load test', and 'sequence diagram'. The main area shows a commit history for a file named 'main'. The commits are as follows:

Name	Last commit message	Last commit date
..	Add files via upload	now
Account Pics	Add files via upload	now
Architecture diagram	Add files via upload	47 minutes ago
Coding	Update and rename project coding style.css to style.css	10 minutes ago
Er diagram	Add files via upload	47 minutes ago
Poker estimation	Add files via upload	47 minutes ago
Progress report	Add files via upload	47 minutes ago
Sprints	Add files via upload	47 minutes ago
TESTING Test Plan and Test Cases	Add files via upload	16 minutes ago
class diagram	Add files via upload	27 minutes ago
load test	Add files via upload	47 minutes ago
sequence diagram	Add files via upload	47 minutes ago

Result:

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