



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

**DEPARTMENT OF COMPUTER SCIENCE AND  
ENGINEERING LAB MANUAL**

**CS23432 – Software Construction**

**(REGULATION 2023)**

**RAJALAKSHMI ENGINEERING COLLEGE  
Thandalam, Chennai-602015**

Name: Daksh Khinvasara

Register No: 231801025

Year / Branch / Section: 2<sup>nd</sup> / AI&DS / FA

Semester: IV

Academic Year: 2024 - 2025



**RAJALAKSHMI**  
**ENGINEERING COLLEGE**  
An AUTONOMOUS Institution  
Affiliated to Anna University, Chennai

### BONAFIDE CERTIFICATE

NAME ..... D.AKSHI ..... KHIN.NVASARA .....

ACADEMIC YEAR ..... 2024 - 2025 ..... SEMESTER ..... IV ..... BRANCH...AI&PS - FA.....

UNIVERSITY REGISTER No.

2116231801025

Certified that this is the bonafide record of work done by the above student in the  
ES23432 - Software Construction Laboratory during the year **2024 - 2025**

**Signature of Faculty - in - Charge**

**Submitted for the Practical Examination held on.....**

**External Examiner**

**Internal Examiner**



**CamScanner**

## INDEX

Name: Daksh

Branch: AI&DS

Sec: FA

Roll No: 231801025

S.No.	Date	Title	Page No.	Teacher's Sign/Remarks
1	22/1/25	Azure Devops Environment Setup	4	
2	22/1/25	Azure Devops Project setup and User Story Management	8	
3	29/1/25	Setting up Epics, Features and User Stories for Project Planning	12	
4	12/2/25	Sprint planning	19	
5	19/2/25	Point Estimation	18	
6	26/2/25	Designing Class and sequence Diagram for Project Architecture	19	
7	5/3/25	Designing Architectural and ER Diagrams for project structure	21	
8	26/3/25	Testing - Test Plans and Test Cases	23	
9	16/4/25	Load Testing and Pipelines	38	
10	23/4/25	GitHub: Project Structure and Naming conventions	43	

## INDEX

S.No.	Date	Title
1.	22/1/25	Azure Devops Environment Setup.
2.	22/1/25	Azure Devops Project Setup and User Story Management.
3.	29/1/25	Setting Up Epics, Features, And User Stories for Project Planning.
4.	12/2/25	Sprint Planning.
5.	19/2/25	Poker Estimation.
6.	26/2/25	Designing Class and Sequence Diagrams for Project Architecture.
7.	05/3/25	Designing Architectural and ER Diagrams for Project Structure.
8.	26/3/25	Testing – Test Plans and Test Cases.
9.	16/4/25	Load Testing and Pipelines.
10.	23/4/25	GitHub: Project Structure & Naming Conventions.

**EXP NO: 1**

## AZURE DEVOPS ENVIRONMENT SETUP

### **Aim:**

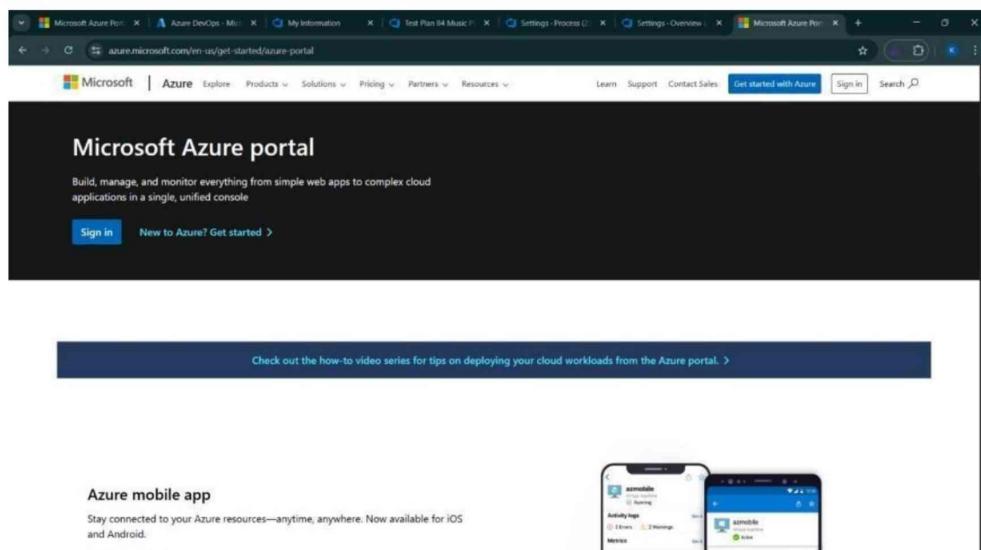
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

### **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.

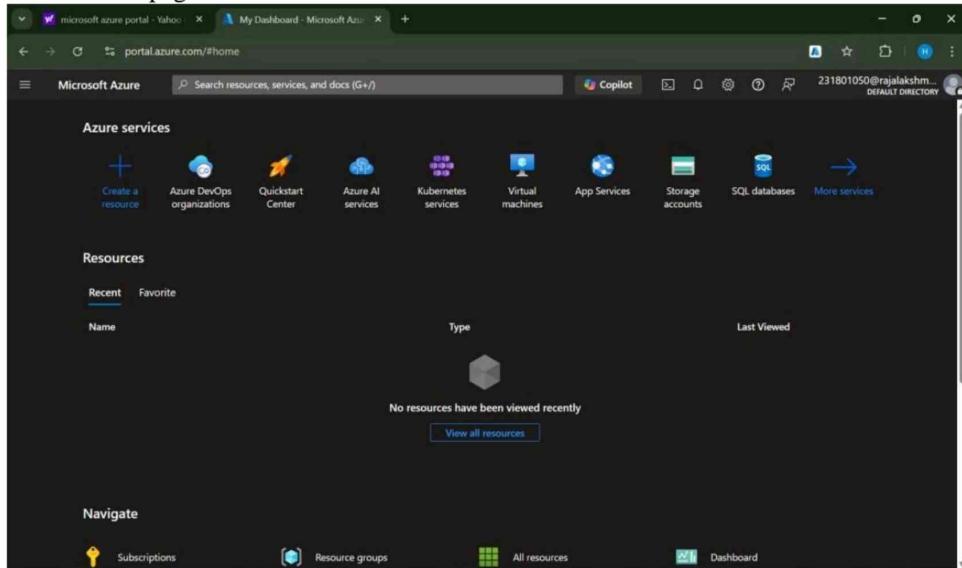
If you don't have a Microsoft account, you can create one here: <https://signup.live.com/?lic=1>



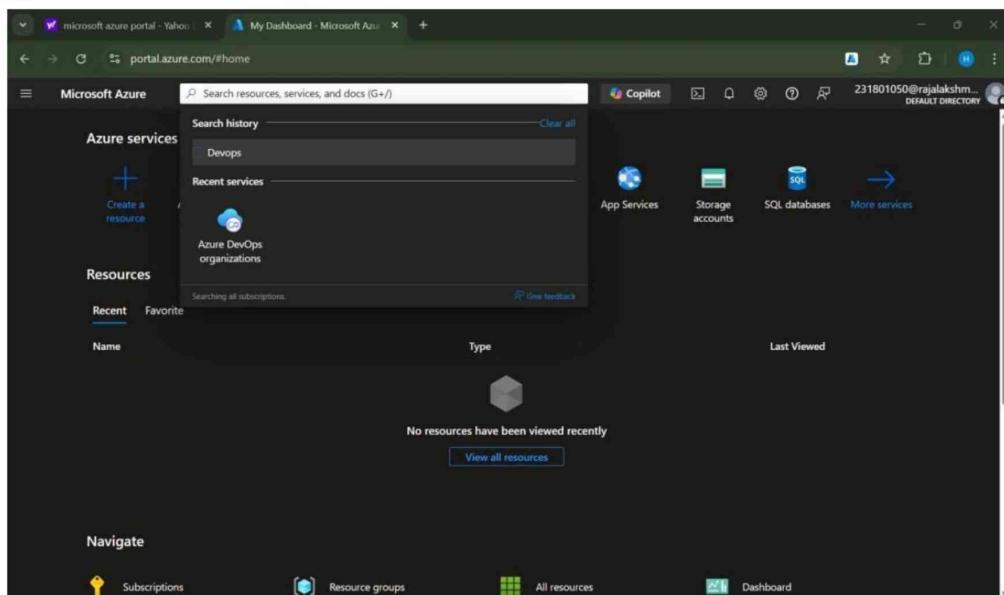
2116231801025

CS23432

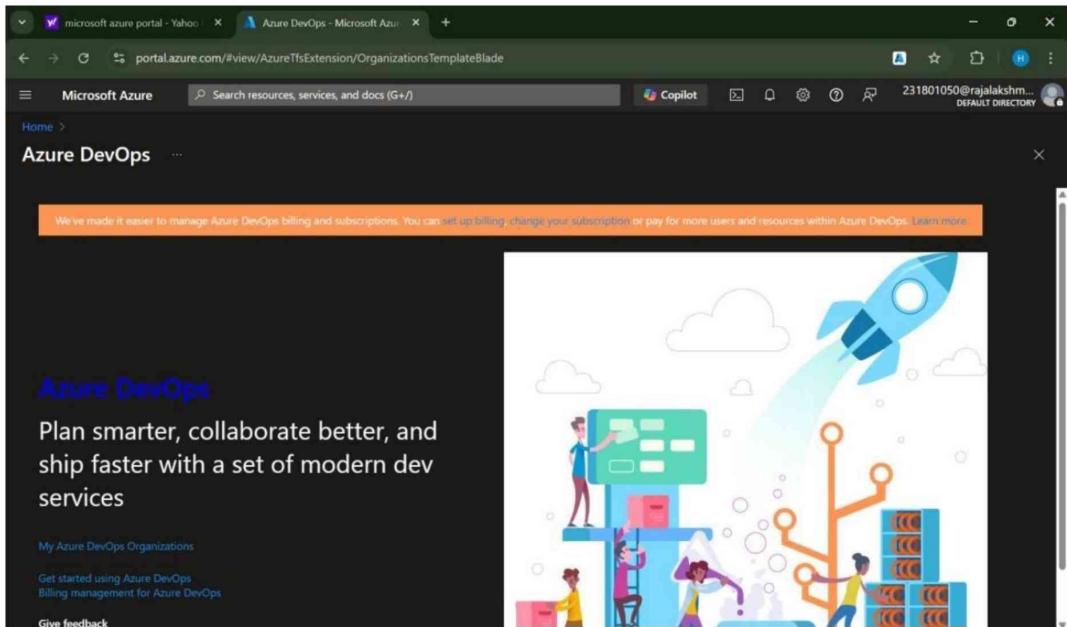
2. Azure home page



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.

2116231801025

CS23432



CamScanner

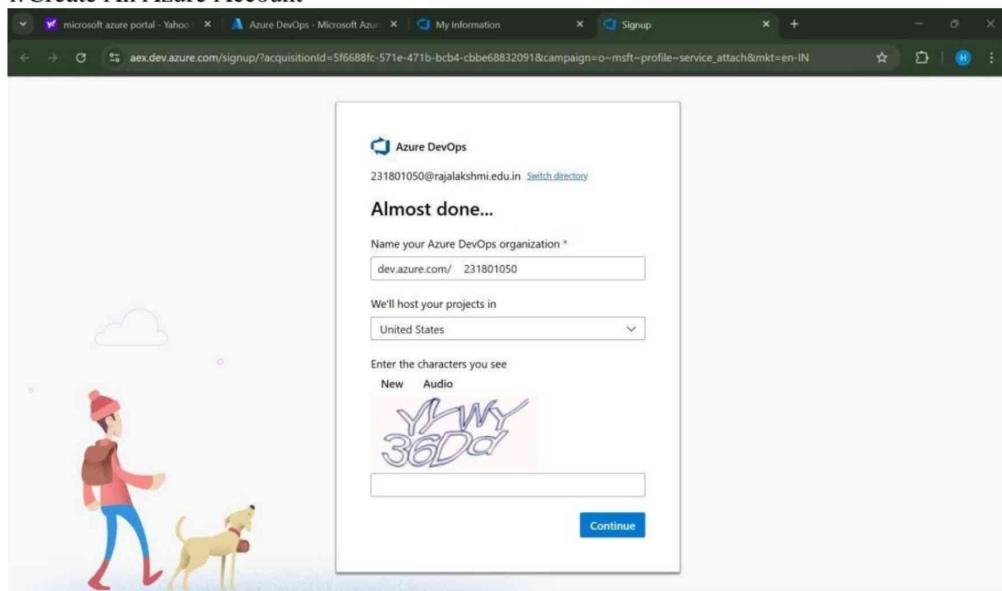
**EXP NO: 2**

## **AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT**

### **Aim:**

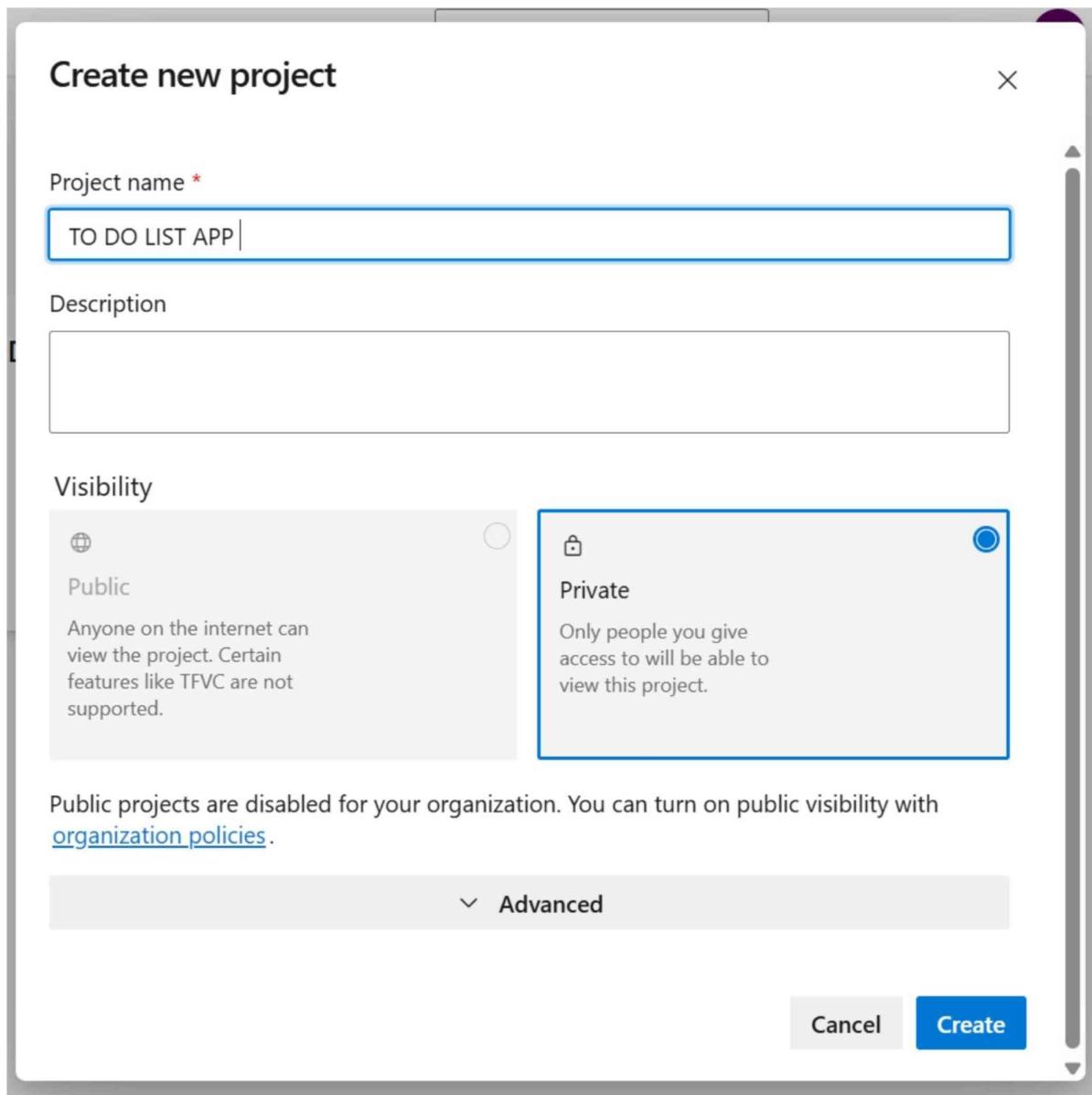
To set up an Azure DevOps project for efficient collaboration and agile work management.

#### **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.

2116231801025

CS23432

The screenshot shows the Azure DevOps Organizations dashboard. On the left, there is a sidebar with the user's profile picture (a purple circle with 'HY'), name (Harish Tutu YT), email (231801050@rajalakshmi.edu.in), and account information (Microsoft account, India). Below this is a section for Visual Studio Dev Essentials. On the right, the main area displays 'Azure DevOps Organizations' with a 'Create new organization' button. Under 'Projects', there is a list with one item: 'dev.azure.com/TO-DO-LIST-APP (Owner)' which is 'TO-DO LIST APP WITH REMINDERS'. There is also a 'New project' link and an 'Actions' section with a 'Open in Visual Studio' button.

#### 4. Project dashboard

The screenshot shows the project overview for 'TO-DO LIST APP WITH REMINDERS'. The left sidebar includes links for Overview, Summary (which is selected), Dashboards, Wiki, Boards, Repos, Pipelines, Test Plans, and Artifacts. The main content area has a title 'TO-DO LIST APP WITH REMINDERS' and a 'Public' status. It features sections for 'About this project' (describing it as a cloud-based To-Do List App with reminders using Azure services like App Service, Functions, SQL Database, and Notification Hubs), 'Project stats' (showing 12 work items created and 0 completed), and 'Members' (listing 5 members with their initials: DR, HY, GR, DS, and DK, each represented by a colored circular icon). A 'Project settings' link is at the bottom left.

2116231801025

CS23432



CamScanner

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 a project titled "TO-DO LIST APP WITH REMINDERS". The left sidebar is visible with options like Overview, Boards, Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Project settings. The "Work items" section is selected. The main area displays a table of work items under the heading "Recently updated". There is one item listed:

ID	Title	Assigned To	State	Area Path
1	DEVELOP A TO-DO LIST APP WITH REMINDERS	Unassigned	New	TO-DO LIST

2116231801025

CS23432

**Result:**

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

2116231801025

CS23432



CamScanner

**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 'Work items' section of the Azure DevOps interface. On the left, there's a sidebar with options like Overview, Boards, Work items, and Repos. The 'Work items' option is selected. The main area displays a table of backlog items:

ID	Title	Assigned To	State	Area Path
1	💡 DEVELOP A TO-DO LIST APP WITH REMINDERS	Unassigned	New	TO-DO LIST
4	⌚ Reminder Notifications	Unassigned	New	TO-DO LIST
5	📌 Task Categorization & Priority Levels	Unassigned	New	TO-DO LIST
3	📝 Task Creation, Editing, and Deletion	Unassigned	New	TO-DO LIST
2	👤 User Authentication & Profile Management	Unassigned	New	TO-DO LIST

### **1. Fill in Epics**

The screenshot shows the 'Edit work item' screen for an epic. The title is 'EPIC 1: DEVELOP A TO-DO LIST APP WITH REMINDERS'. The 'Description' field contains a brief overview of the epic. The 'Planning' section includes fields for Priority (set to 2) and Risk. The 'Deployment' section has a note about tracking releases. The 'Development' section includes fields for Start Date and Add link.

2116231801025

CS23432



CamScanner

## 2. Fill in Features

The screenshot shows the Azure DevOps interface for a project named 'TO-DO LIST APP WITH REMINDERS'. A specific work item, 'Reminder Notifications', is selected. The work item details include:

- Title:** FEATURE 4 - Reminder Notifications
- Status:** New
- Area:** TO-DO LIST APP WITH REMINDERS
- Iteration:** TO-DO LIST APP WITH REMINDERS
- Description:** Users should receive timely reminders for their pending tasks.
- Key Functionalities:**
  - Set reminders for tasks.
  - Receive push notifications before the task deadline.
  - Snooze or dismiss reminders.
- Planning:** Priority: 2, Risk: 1, Effort: 1, Business Value: 1
- Deployment:** To track releases associated with this work item, go to [Releases](#) and turn on deployment status reporting for Boards in your pipeline's Options menu. [Learn more about deployment status reporting](#)
- Development:** Add link, Link an Azure Repos

## 3. Fill in User Story Details

The screenshot shows the Azure DevOps interface for a project named 'TO-DO LIST APP WITH REMINDERS'. A specific work item, 'As a user, I want to create a new task so that I can keep track of my work.', is selected. The work item details include:

- Title:** USER STORY 9
- Status:** New
- Area:** TO-DO LIST APP WITH REMINDERS
- Iteration:** TO-DO LIST APP WITH REMINDERS\Sprint 2
- Description:** Users should be able to add a task with a title, description, and due date.
- Acceptance Criteria:**
  - Users can enter a task title and description.
  - Users can set a due date.
  - Task should be saved and displayed in the task list.
- Planning:** Story Points: 5, Priority: 2, Risk: 1
- Classification:** Value area: Business
- 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

2116231801025

CS23432



CamScanner

**Result:**

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

2116231801025

CS23432



CamScanner

**EXP NO: 4**

## SPRINT PLANNING

### Aim:

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

### Sprint Planning

#### Sprint 1

The screenshot shows the Azure DevOps Boards Sprints page for the 'TO-DO LIST APP WITH REMINDERS' project. The left sidebar is visible with 'Boards' selected. The main area displays the 'Taskboard' for 'Sprint 1' from February 21 to March 21. There are four user stories listed:

- User Story 6: As a user, I want to register and log in so that I can access my tasks securely. Status: New, Assigned to Dhanush R.
- User Story 18: Implement user authentication (Sign-up, Login, Logout). Status: New, Assigned to Dhanush R.
- User Story 7: As a user, I want to manage my profile so that I can update my details. Status: New, Assigned to Unassigned.
- User Story 19: Develop profile update functionality (name, email, password). Status: New, Assigned to Dhanush R.

#### Sprint 2

The screenshot shows the Azure DevOps Boards Sprints page for the 'TO-DO LIST APP WITH REMINDERS' project. The left sidebar is visible with 'Boards' selected. The main area displays the 'Taskboard' for 'Sprint 2' from March 22 to April 22. There are four user stories listed:

- User Story 9: As a user, I want to create a new task so that I can keep track of my work. Status: New, Assigned to Unassigned.
- User Story 21: Develop UI & API for adding new tasks. Status: New, Assigned to Harish Tutu YT.
- User Story 10: As a user, I want to edit an existing task so that I can update my plans. Status: New, Assigned to Unassigned.
- User Story 22: Implement functionality for modifying task details. Status: New, Assigned to Dalish Khinvasa...

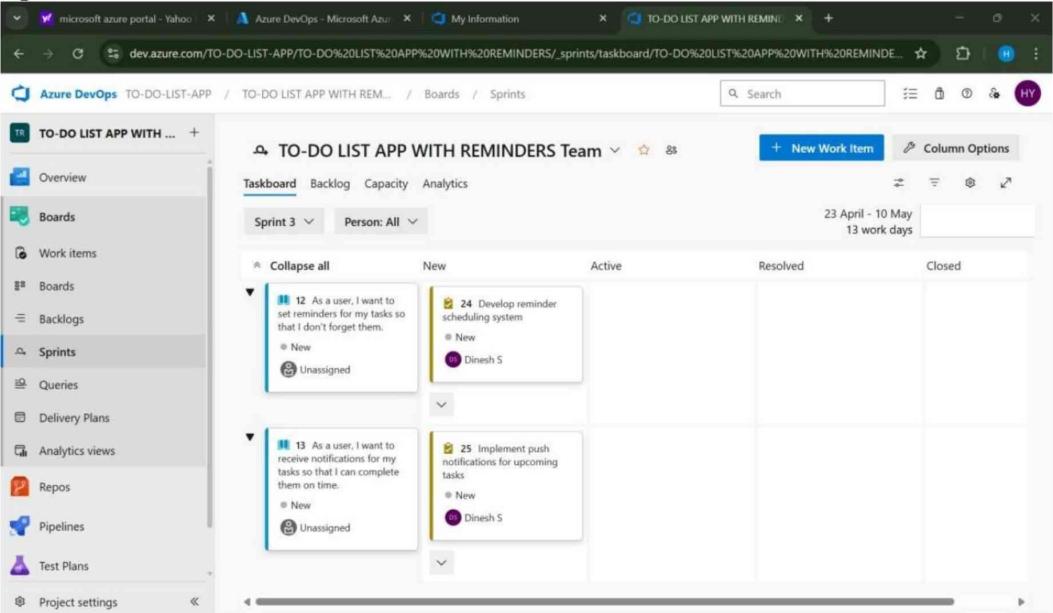
2116231801025

CS23432



CamScanner

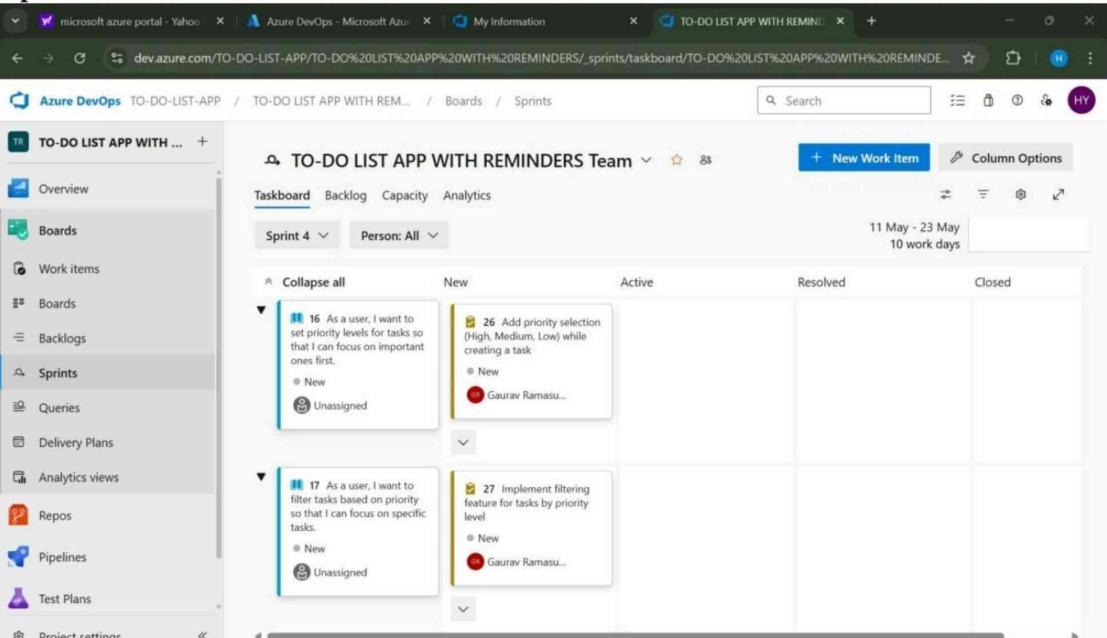
### Sprint 3



The screenshot shows the Azure DevOps Taskboard for the 'TO-DO LIST APP WITH REMINDERS' project. The current sprint is 'Sprint 3' from April 23 to May 13, which spans 13 work days. The board has four columns: New, Active, Resolved, and Closed. There are two tasks in the 'New' column:

- Task 12: As a user, I want to set reminders for my tasks so that I don't forget them. Status: New, Assigned to Dinesh S.
- Task 13: As a user, I want to receive notifications for my tasks so that I can complete them on time. Status: New, Unassigned.

### Sprint 4



The screenshot shows the Azure DevOps Taskboard for the 'TO-DO LIST APP WITH REMINDERS' project. The current sprint is 'Sprint 4' from May 11 to May 23, which spans 10 work days. The board has four columns: New, Active, Resolved, and Closed. There are two tasks in the 'New' column:

- Task 16: As a user, I want to set priority levels for tasks so that I can focus on important ones first. Status: New, Unassigned.
- Task 17: As a user, I want to filter tasks based on priority so that I can focus on specific tasks. Status: New, Unassigned.

Two other tasks have been moved to the 'Resolved' column:

- Task 24: Develop reminder scheduling system. Status: Resolved, Assigned to Dinesh S.
- Task 25: Implement push notifications for upcoming tasks. Status: Resolved, Assigned to Dinesh S.

2116231801025

CS23432



CamScanner

**Result:**

The Sprints are created for the To-do list app with reminders project.

2116231801025

CS23432



CamScanner

**EXP NO: 5**

## POKER ESTIMATION

### Aim:

Create Poker Estimation for the user stories - To-do list app with reminders project.

### Poker Estimation

The screenshot shows a detailed view of a User Story in Azure DevOps. The story is titled "As a user, I want to register and log in so that I can access my tasks securely." It is categorized under the "TO-DO LIST APP WITH REMINDERS" project. The "Planning" section indicates 3 Story Points and Priority 1. The "Deployment" section provides instructions for tracking releases. The "Classification" section shows the value area as "Business". The "Development" section includes a link to an Azure Repos commit or pull request. The card is labeled "USER STORY 6" and was last updated by Harish Tutu YT on Feb 21.

### Result:

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

2116231801025

CS23432



CamScanner

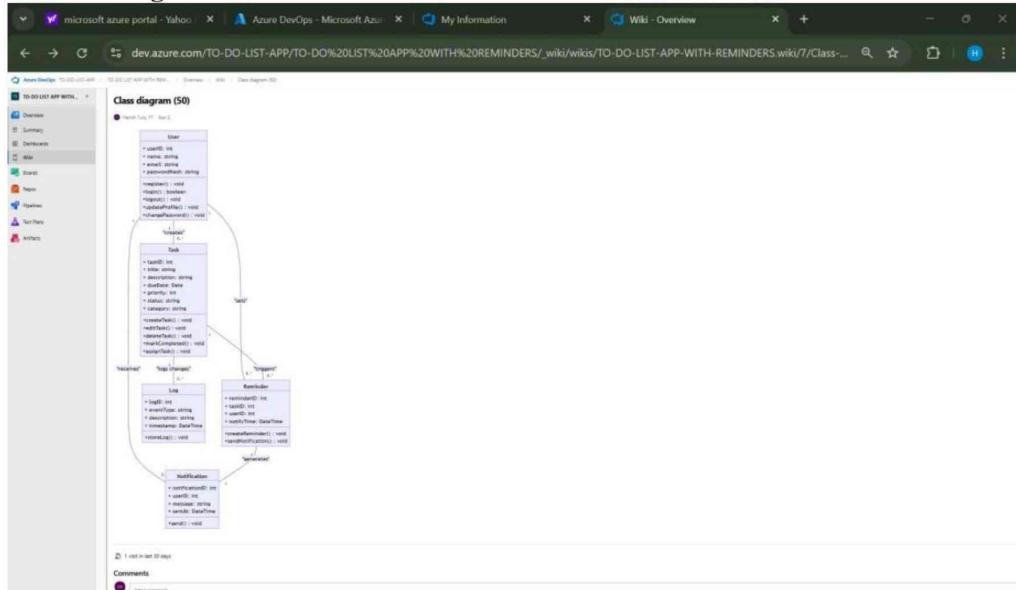
**EXP NO: 6**

## **DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE**

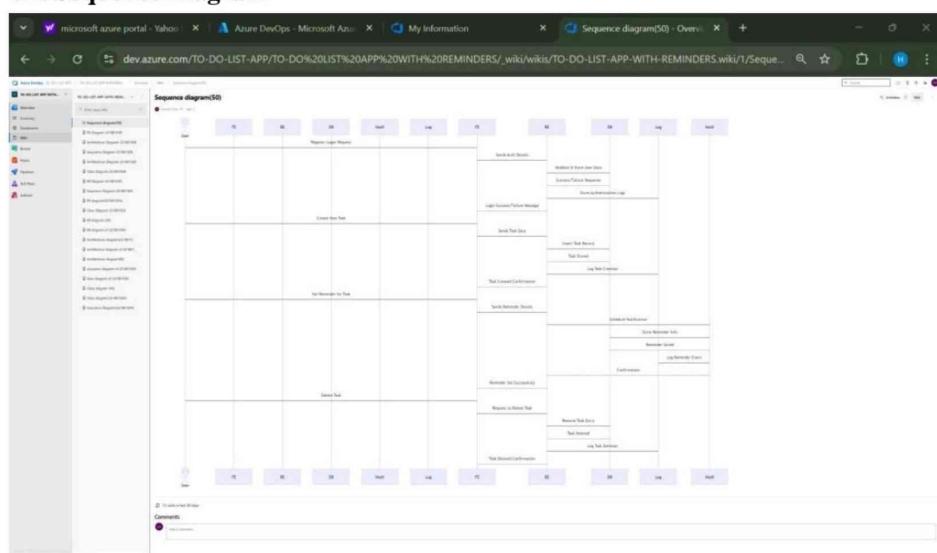
### **Aim:**

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

### **6A. Class Diagram**



### **6B. Sequence Diagram**



2116231801025

CS23432



CamScanner

**Result:**

The Class Diagram and Sequence Diagram is designed Successfully for the To-do list app with reminders project.

2116231801025

CS23432

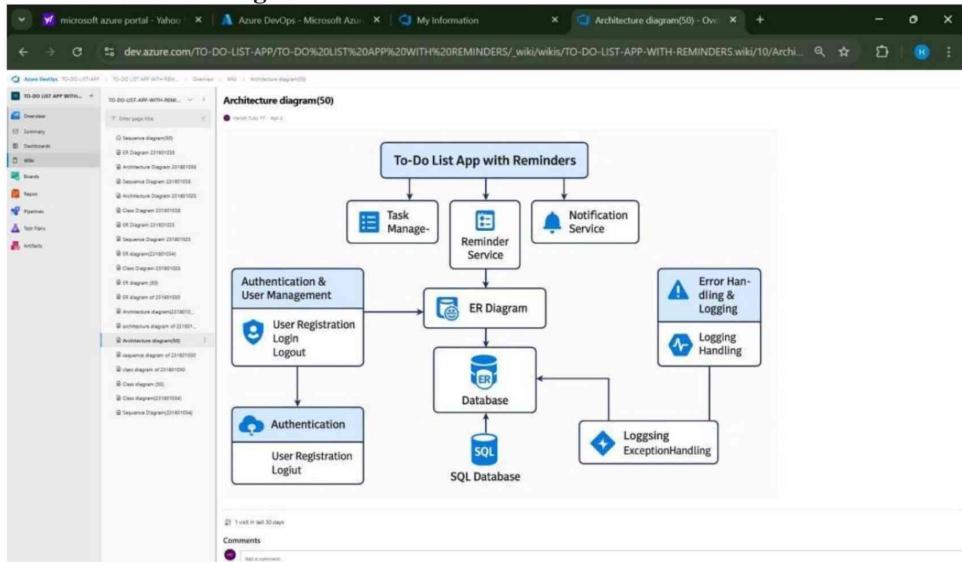
**EXP NO: 7**

## **DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE**

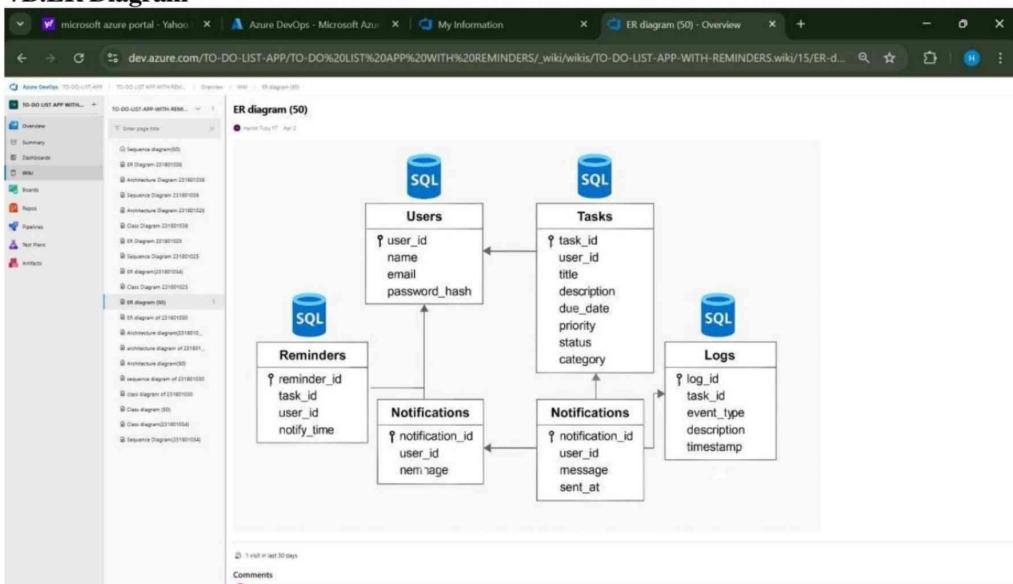
### **Aim:**

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

### **7A. Architectural Diagram**



### **7B. ER Diagram**



2116231801025

CS23432



CamScanner

**Result:**

The Architecture Diagram and ER Diagram is designed Successfully for the To-do list app with reminders project.

2116231801025

CS23432



CamScanner

<b>EXP NO: 8</b>	<b>TESTING – TEST PLANS AND TEST CASES</b>
------------------	--

**Aim:**

To give test cases for the To-Do List App showcasing both the happy path (expected scenarios) and error path (unexpected scenarios).

**Test Planning and Test Case****Test Case Design Procedure****1. Understand Core Features of the Application**

- User signup, login, logout, and profile management
- Creating, editing, and deleting tasks
- Setting task reminders and receiving notifications
- Setting and filtering task priorities

**2. Define User Interactions**

- Each test case is based on real user actions like registering, creating tasks, setting reminders, etc.

**3. Design Happy Path Test Cases**

- These validate that all core functionalities work as expected under normal conditions..

**4. Design Error Path Test Cases**

- Simulate negative or unexpected behavior like login failures or invalid inputs.

**5. Break Down Steps and Expected Results**

- Each test case includes step-by-step user actions and the expected outcome

**6. Use Clear Naming and IDs**

- Example: TC01-Successful-Login, TC05-Task-Reminder-Failure

**7. Separate Test Suites**

- Group test cases based on modules: Authentication, Task Management, Reminders, Priority Handling, Security.

## 8. Prioritize and Review

- High-priority test cases are assigned to core features like login, task creation, and reminder notifications.

### 1. New test plan

The screenshot shows the 'Test Plan 28 TO-DO LIST APP' page in the Azure DevOps interface. The test plan is titled '28 TO-DO LIST APP' and is created by 'Harish Tutu YT'. It is set to 'Active' status and belongs to the 'TO-DO LIST APP WITH REMINDERS' area. The 'Reason' is listed as 'New test plan' and the 'Iteration' is 'TO-DO LIST APP WITH REMINDERS'. The 'Timelines' section shows a start date of 15-04-2025 15:09 and a finish date of 22-04-2025 15:09. The 'Description' and 'Discussion' sections are currently empty. A 'Project settings' button is at the bottom left.

### 2. Test suite

The screenshot shows the 'Test Plan 28 TO-DO LIST APP / Test Suites' page. The left sidebar shows project navigation with 'Test Plans' selected. In the main area, a 'Task management (ID: 31)' card is displayed under the 'Current' tab. It shows a timeline from April 15 to April 22, with 100% run and 100% passed. Below this, a 'Test Suites' section lists 'TO-DO LIST APP' and 'Authentication (3)'. A context menu is open over the 'TO-DO LIST APP' entry, showing options like 'New Suite', 'Static suite', 'Requirement based suite', and 'Query based suite'. Other menu items include 'Assign configurations', 'Export', 'Open', 'Assign testers to run all tests', 'Rename', 'Delete', and 'Import test suites'.

2116231801025

CS23432



CamScanner

### **3. Test case**

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

To-Do List App – Test Plans

#### **Test Suites**

##### **Test Suit: TS01 - User Authentication**

###### **1. TC01 – User Registration & Login**

Action 1: Open the app and click “Register”.

Expected Result: Registration form appears.

Action 2: Enter valid name, email, and password, then click “Submit”.

Expected Result: Account created, redirected to login page.

Action 3: Enter credentials and click “Login”.

Expected Result: Dashboard loads, user successfully logged in.

###### **2. TC02 – Manage Profile**

Action 1: Click on “Profile” in the navigation bar.

Expected Result: Profile details are displayed.

Action 2: Click “Edit”, change user info (e.g., name).

Expected Result: Editable fields appear.

Action 3: Click “Save Changes”.

Expected Result: Profile updated confirmation appears.

###### **3. TC03 – Logout**

Action 1: Click on “Logout” button from dashboard.

Expected Result: Logout confirmation popup appears.

2116231801025

CS23432



CamScanner

Action 2: Confirm logout.

Expected Result: User is logged out.

Action 3: Try accessing dashboard URL directly.

Expected Result: Redirected to login page.

### **Test Suit: TS02 – Task management**

#### 1. TC04 – Create Task

Action 1: Click “Add New Task”.

Expected Result: Task creation form opens.

Action 2: Fill in task title, due date, priority, and click “Save”.

Expected Result: Task added to list.

Action 3: View the dashboard.

Expected Result: Newly created task appears in “Upcoming Tasks”

#### 2. TC05 – Edit Task

Action 1: Click the “Edit” icon next to a task.

Expected Result: Task fields become editable.

Action 2: Change title or date.

Expected Result: Fields accept new values.

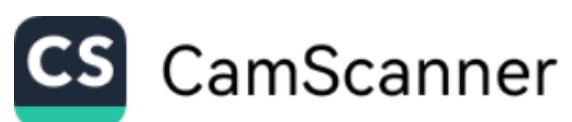
Action 3: Click “Update”.

Expected Result: Task details are updated in task list.

#### 3. TC06 – Delete Task

2116231801025

CS23432



Action 1: Click on the “Delete” button of a task.

Expected Result: Confirmation dialog appears.

Action 2: Click “Yes, Delete”.

Expected Result: Task is removed from the list.

Action 3: Refresh the page.

Expected Result: Task no longer appears.

### **Test Suit: TS03 – Reminder & notification**

#### 1. TC07 – Set Reminders

Action 1: Click “Set Reminder” on a task.

Expected Result: Date and time input appears.

Action 2: Set future time and save.

Expected Result: Reminder is saved.

Action 3: Wait till reminder time.

Expected Result: Notification or alert is triggered.

#### 2. TC08 – Notifications

Action 1: Ensure a task has a reminder set.

Expected Result: Reminder time shows in UI.

Action 2: Wait until the task time arrives.

Expected Result: App shows a popup or notification.

Action 3: Click the notification.

Expected Result: User navigates to the task.

2116231801025

CS23432



CamScanner

### 3. TC09 – Set Task Priority

Action 1: Click “New Task”, set priority as “High”.

Expected Result: Priority dropdown is available and saves input.

Action 2: Save the task.

Expected Result: Priority level shows in task card.

Action 3: Hover or click task for details.

Expected Result: Priority tag (e.g., red for high) is visible.

### 4. TC10 – Filter Tasks by Priority

Action 1: Click “Filter” and choose “High Priority”.

Expected Result: Filter activates.

Action 2: View task list.

Expected Result: Only high-priority tasks are visible.

Action 3: Remove filter.

Expected Result: All tasks become visible again.

## Test Cases

The screenshot shows the Azure DevOps Test Plan interface. A test case titled "33 User Registration & Login" is displayed. The test case details include:

- Owner: Dinesh S
- Comments: 0
- Add Tag: Available
- State: Design
- Area: TO-DO LIST APP WITH REMINDERS
- Reason: New
- Iteration: TO-DO LIST APP WITH REMINDERS

The "Steps" tab is selected, showing the following steps:

- Open the app and click "Register". Expected result: Registration form appears.
- Enter valid name, email, and password, then click "Submit". Expected result: Account created, redirected to dashboard.
- Enter credentials and click "Login". Expected result: Dashboard loads, user successfully logged in.

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

On the right side of the screen, there are sections for "Deployment" and "Development". The deployment section provides instructions on tracking releases associated with the work item. The development section provides instructions on linking to Azure Repos commits, pull requests, or branches.

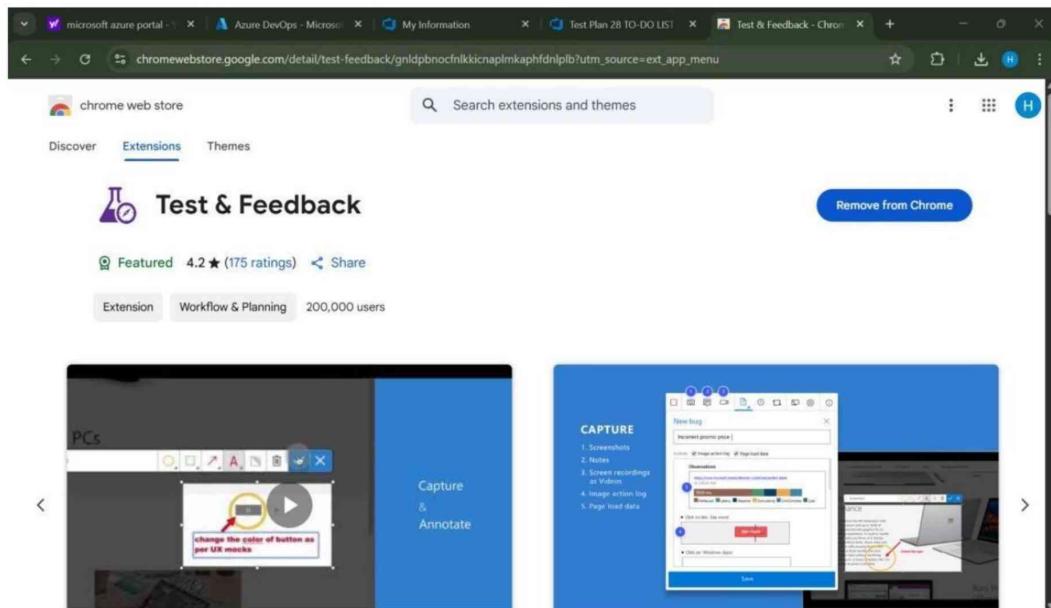
2116231801025

CS23432



CamScanner

#### 4. Installation of test



2116231801025

CS23432

Test and feedback  
Showing it as an extension

The screenshot shows the Azure DevOps Test Plan interface. On the left, there's a sidebar with various icons for Test Cases, Test Suites, Pipelines, and Artifacts. The main area displays a test case titled "33 User Registration & Login". The "Steps" section contains three steps: 1. Open the app and click "Register". Expected result: Registration form appears. 2. Enter valid name, email, and password, then click "Submit". Expected result: Account created, redirected to dashboard. 3. Enter credentials and click "Login". Expected result: Dashboard loads, user successfully logged in. To the right of the test case, a modal window titled "Extensions" is open, listing extensions with "Full access": "Allow Copy - Select & En...", "Blend & Run: Enable copy...", "Test & Feedback" (which is highlighted with a purple border), and "Manage extensions". Below the extensions, there's a note about tracking releases and deployment status reporting. At the bottom of the modal, there's a "Development" section with a "Add link" button.

2116231801025

CS23432



CamScanner

## 5. Running the test cases

The screenshot shows the Azure DevOps interface for a 'TO-DO LIST APP' test plan. On the left, the navigation menu includes 'Overview', 'Boards', 'Repos', 'Pipelines', 'Test Plans' (selected), 'Test plans', 'Progress report', 'Parameters', 'Configurations', 'Runs', 'Artifacts', and 'Project settings'. The main area displays a 'Test Suites' section with 'TO-DO LIST APP' expanded, showing 'Authentication (3)' selected. A 'Test Points (3 items)' table lists three entries: 'User Registration & Login' (Passed), 'Manage Profile' (Passed), and 'Logout' (Passed). A context menu is open over the 'User Registration & Login' row, with options like 'Run for web application', 'Run for desktop application', 'Run with options', 'Mark Outcome' (set to Passed), 'Edit test case', 'Assign tester', and 'View test result'.

6.

The screenshot shows a browser window titled 'Runner - Test Plans - Google Chrome' with the URL 'dev.azure.com/TO-DO-LIST-APP/TO-DO%20LIST%20APP%20WITH%20REMINDERS/\_testExecution/Index'. The page displays a test step for 'User Registration & Login':  
1. Open the app and click "Register".  
EXPECTED RESULT  
Registration form appears.  
2. Enter valid name, email, and password, then click "Submit".  
EXPECTED RESULT  
Account created, redirected to login page.  
3. Enter credentials and click "Login".  
EXPECTED RESULT  
Dashboard loads, user successfully logged in.

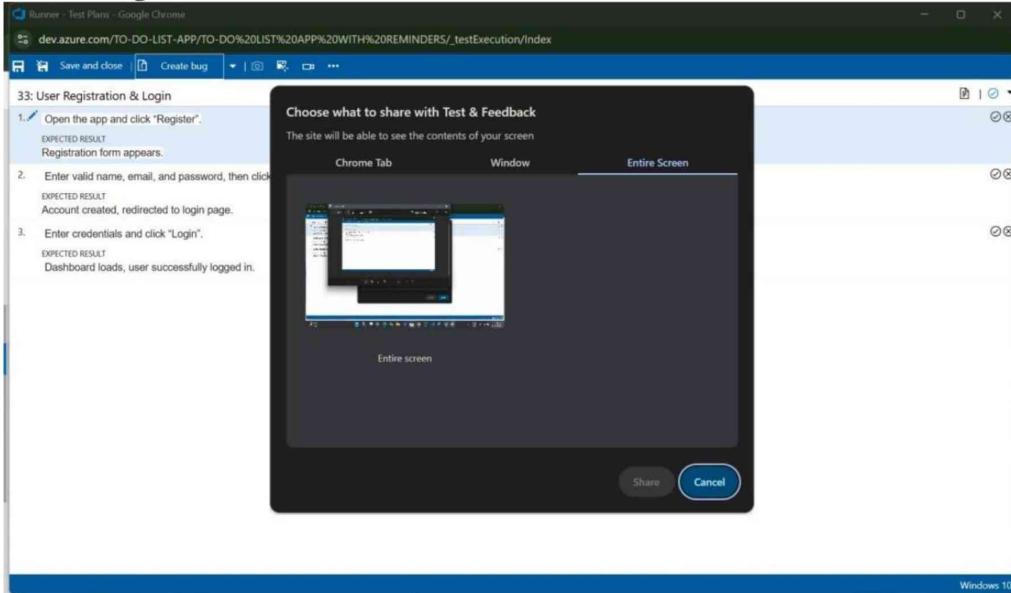
2116231801025

CS23432



CamScanner

## 6. Recording the test case



## 7. Creating the bug



The screenshot shows two instances of the Azure DevOps Bug Details page. The top instance displays a bug titled "loading due to poor network" with the following repro steps:

- Step no.** Result Title
- None** Open the app and click "Register".  
Expected Result  
Registration form appears.
- None** Enter valid name, email, and password, then click "Submit".  
Expected Result

The bottom instance shows the same bug with additional environment details:

Browser - Name	Google Chrome 135
Browser - Language	en-US
Browser - Height	768
Browser - Width	1296
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	i686_64
Operating system - Processor model	13th Gen Intel(R) Core(TM) i7-1360P
Operating system - Number of processors	16
Memory - Available	4136333312
Memory - Capacity	16849256448
Display - Pixels per inch (X axis)	144
Display - Pixels per inch (Y axis)	144

2116231801025

CS23432

## 8. Test case results

A screenshot of the Azure DevOps interface showing the 'Test Case Results' table. The table lists test runs for 'Task management' test cases. The columns include Outcome, TimeStamp, Configuration, Run by, Tester, and Test Plan.

Outcome	TimeStamp	Configuration	Run by	Tester	Test Plan
Passed	2h ago	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC
Failed	2h ago	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC
Passed	2h ago	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC
Passed	Wednesday	Windows 10	Gaurav Ramasubram...	Harish Tutu YT	TO-DC
Passed	Wednesday	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC
In Progress	Wednesday	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC
Passed	Wednesday	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC
Passed	Wednesday	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC
Failed	Wednesday	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC
Failed	Wednesday	Windows 10	Harish Tutu YT	Harish Tutu YT	TO-DC

## 9. Test report summary

A screenshot of the Azure DevOps interface showing a detailed view of a test step. The step is titled 'Create Task' and is associated with 'TEST CASE 34'. The step details the action 'Click "Add New Task..."' and its expected result 'Task creation form opens'. The step is part of a larger test plan for 'TO-DO LIST APP WITH REMINDERS'.

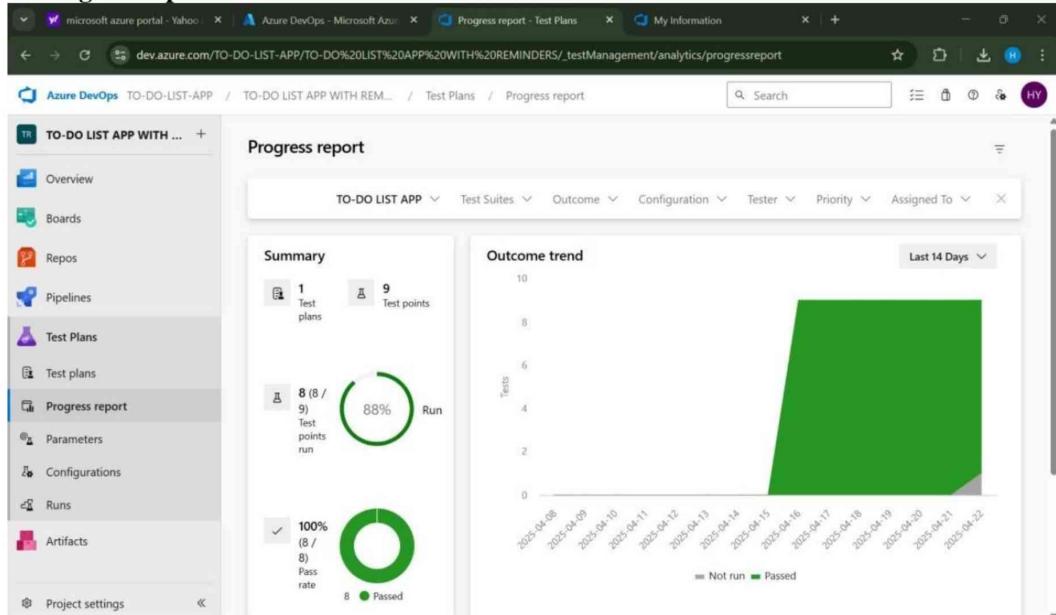
2116231801025

CS23432



CamScanner

## 10. Progress report



2116231801025

CS23432



CamScanner

**Result:**

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

2116231801025

CS23432



CamScanner

**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:

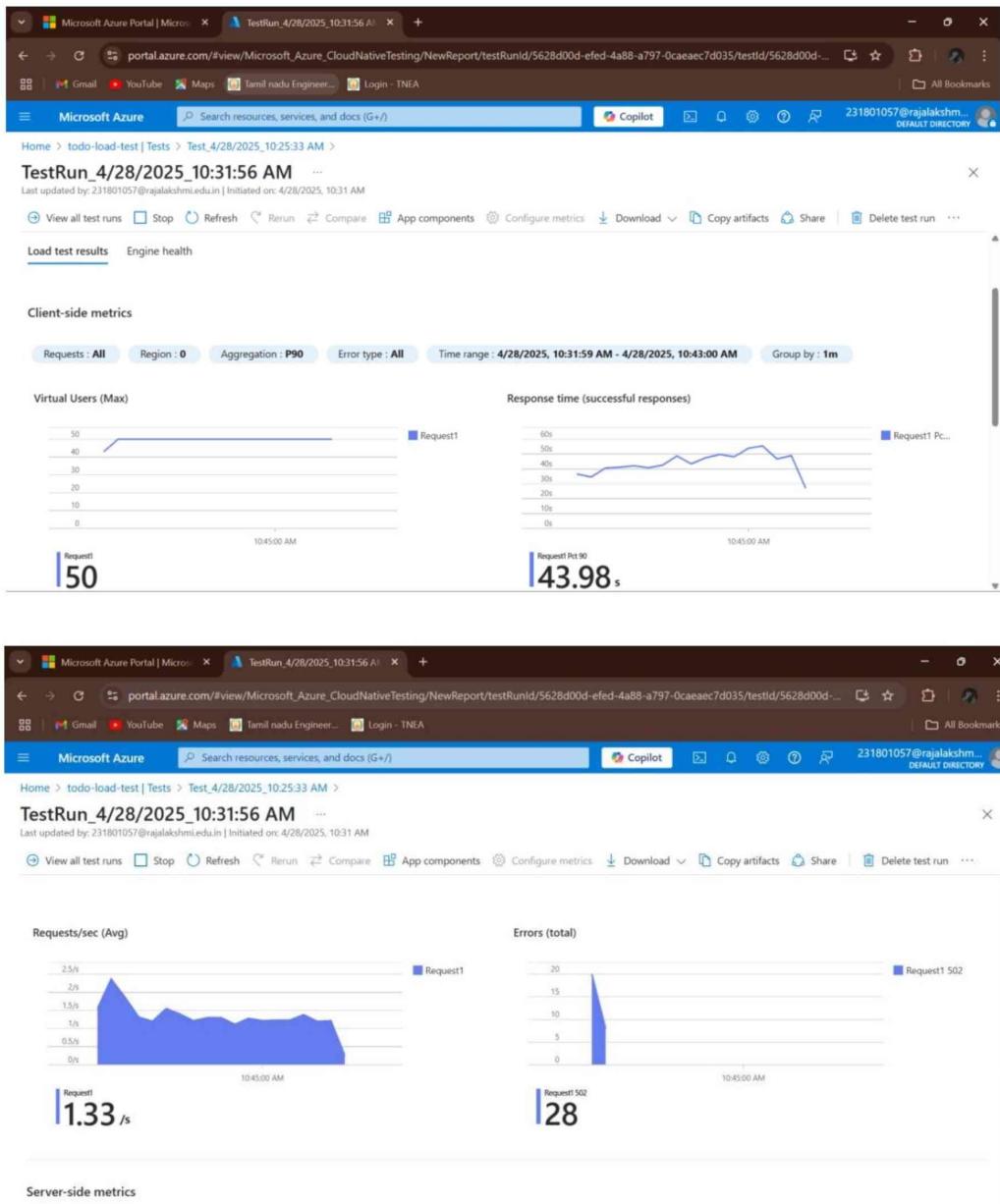
1. Sign in to Azure Portal  
Go to <https://portal.azure.com> and log in.
2. Create the Resource
  - o Go to *Create a resource* → Search for “Azure Load Testing”.
  - o Select Azure Load Testing and click Create.
3. Fill in the Configuration Details
  - o *Subscription*: Choose your Azure subscription.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

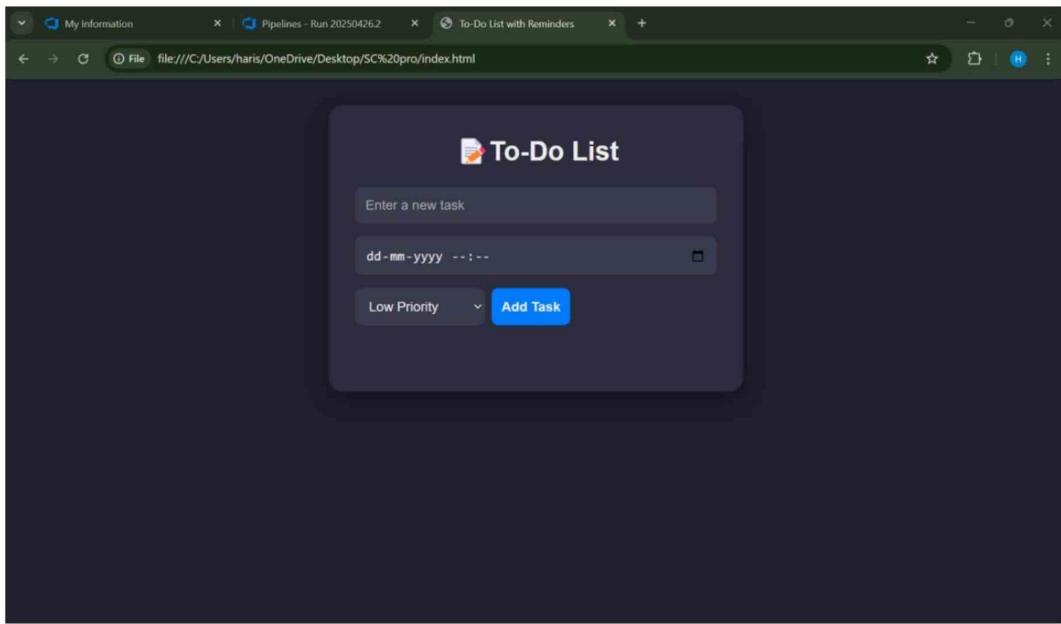


2116231801025

CS23432



CamScanner



## Pipelines

### Description:

This experiment demonstrates connecting a GitHub-hosted Flask-based music recommendation project with Azure DevOps. The pipeline will automatically install dependencies, run basic tests, and publish artifacts. This ensures that every commit triggers checks for reliability and smooth deployment.

Steps:

1. Connect GitHub to Azure DevOps:
  - o In Azure DevOps, create a new project.
  - o Create a pipeline and select GitHub as the source.
  - o Authorize access to your GitHub repository, ensuring that Azure DevOps can pull the repository for your pipeline.
2. Create azure-pipelines.yml in Your Repo Root:
  - o In your GitHub repository, create a new file called azure-pipelines.yml in the root directory.
  - o Add the following basic pipeline configuration for Python and Flask:

```

yml Code
trigger:
- main # Trigger pipeline when changes are pushed to the main branch

pool:
vmImage: ubuntu-latest # Use a hosted Ubuntu agent

steps:
# Step 1: Checkout the code from GitHub
- checkout: self

# Step 2: Set up Python environment
- task: UsePythonVersion@0
inputs:
versionSpec: '3.x' # Use the latest Python 3.x version
displayName: "Set up Python"

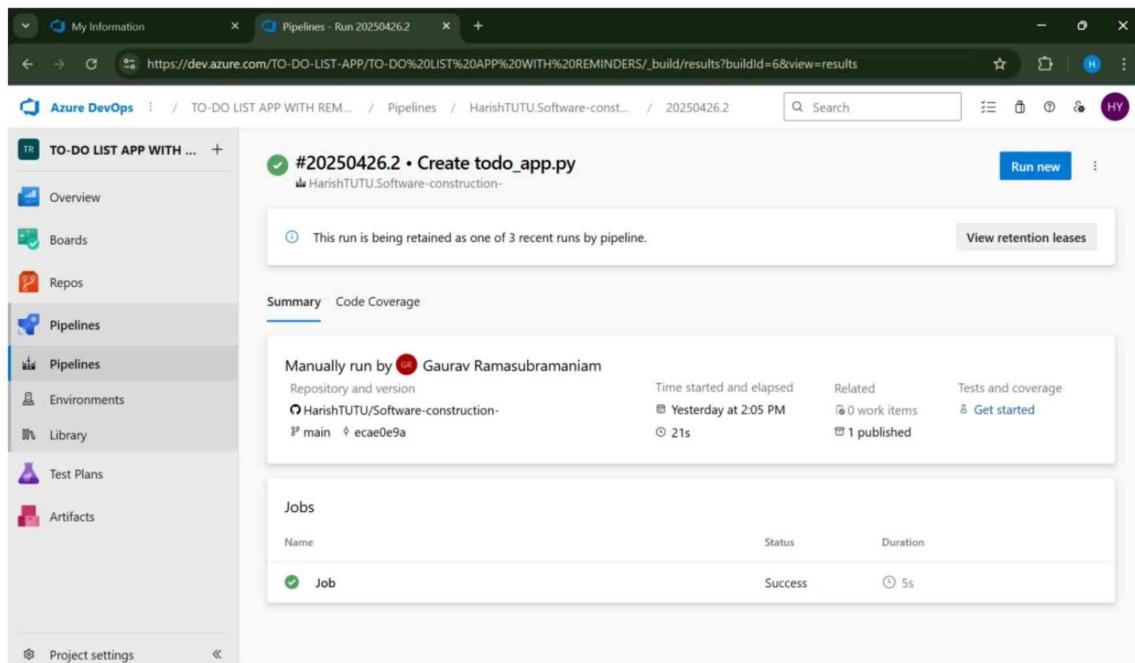
# Step 3: Install dependencies from the correct path
- script: |
  python -m pip install --upgrade pip
  pip install -r project/requirements.txt # Adjusted path to requirements.txt
  displayName: "Install dependencies"

# Step 4: Run a simple Python script to check the environment
- script: |
  python -c "print('Hello from TO-DO LIST APP WITH REMINDERS!')"
  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. Ensure the path to requirements.txt is correct (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 results page for a run titled '#20250426.2 • Create todo\_app.py'. The pipeline was triggered manually by Gaurav Ramasubramanian. The repository is HarishTUTU/Software-construction, branch main, commit ecae0e9a. The run started yesterday at 2:05 PM and took 21s. There are 0 work items related and 1 published. The 'Jobs' section shows one job named 'Job' which was successful and took 5s.

Name	Status	Duration
Job	Success	5s

## 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.

2116231801025

CS23432

**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 TO-DO LIST APP WITH REMINDERS project.

### **GitHub Project Structure**

The screenshot shows the GitHub repository page for 'TO-DO-LIST-APP-WITH-REMINDERS'. The repository is public and has 1 branch and 0 tags. The commit history shows 10 commits from 'HarishTUTU' made 1 minute ago. The commits are listed as follows:

Commit	Message	Time Ago
Architecture Diagram	Add files via upload	10 minutes ago
Backlog	Add files via upload	10 minutes ago
Pipelines	Add files via upload	10 minutes ago
Poker Estimation	Add files via upload	1 minute ago
Progress Report	Add files via upload	10 minutes ago
Project	Add files via upload	1 minute ago
Sequence Diagram	Add files via upload	10 minutes ago
Sprints	Add files via upload	1 minute ago
Test Plans And Test Cases	Add files via upload	10 minutes ago
azure-pipelines.yml	Rename azure-pipelines-1.yml to azure-pipelines.yml	44 minutes ago

The repository has 0 stars, 1 watching, and 0 forks. It has no releases or packages published. The language distribution is 70.1% HTML and 29.9% Python.

### **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.

2116231801025

CS23432



CamScanner