



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

DEPARTMENT OF INFORMATION TECHNOLOGY
LAB MANUAL

CS23432 – Software Construction

(REGULATION 2023)

RAJALAKSHMI ENGINEERING COLLEGE
Thandalam, Chennai – 602015

Name: SANDHIYA S K
Register No: 231001177
Year / Branch / Section: 2nd /IT/ FC
Semester: IV
Academic Year: 2024 - 2025

INDEX

S.No.	Date	Title
1.	23/01/25	Azure Devops Environment Setup.
2.	06/02/25	Azure Devops Project Setup and User Story Management.
3.	20/02/25	Setting Up Epics, Features, And User Stories for Project Planning.
4.	13/03/25	Sprint Planning.
5.	20/03/25	Poker Estimation.
6.	26/03/25	Designing Class and Sequence Diagrams for Project Architecture.
7.	03/04/25	Designing Architectural and ER Diagrams for Project Structure.
8.	10/04/25	Testing – Test Plans and Test Cases.
9.	17/04/25	Load Testing and Performance Testing.
10.	24/04/25	GitHub: Project Structure & Naming Conventions.

EXP NO: 1

AZURE DEVOPS ENVIRONMENT SETUP

Aim:

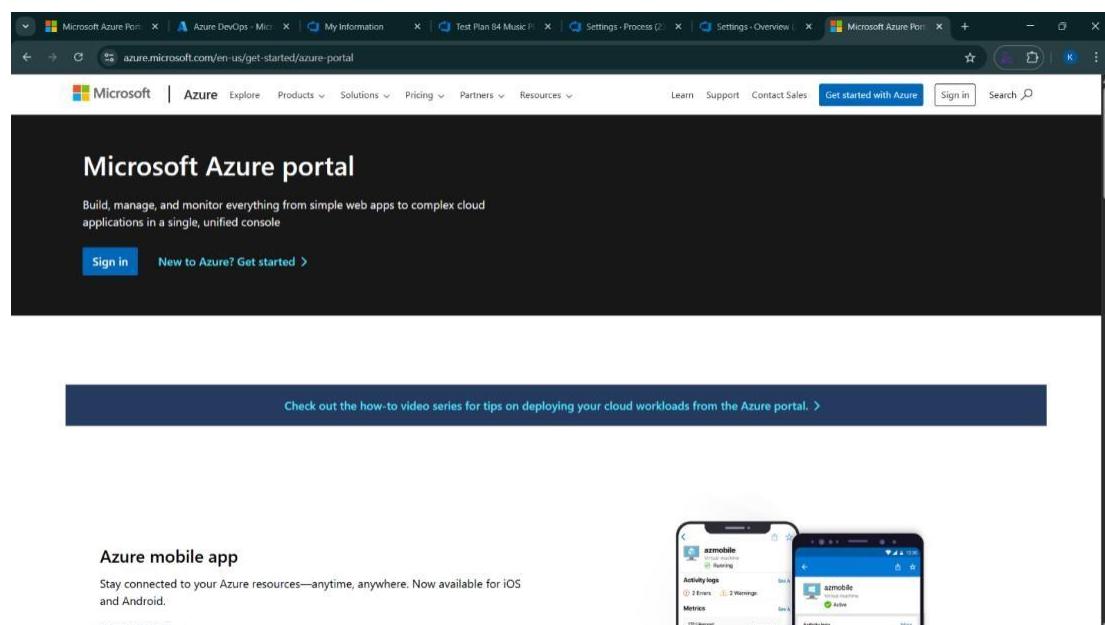
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

INSTALLATION

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

Sign in using your Microsoft account credentials.

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



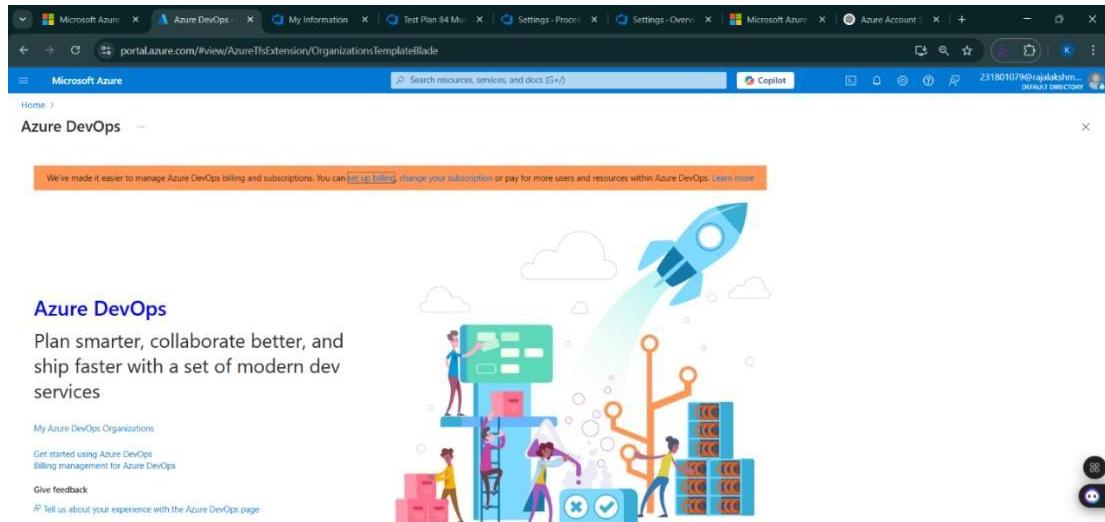
2. Azure home page

The screenshot shows the Microsoft Azure home page. At the top, there's a navigation bar with tabs like 'Microsoft Azure', 'Azure DevOps', 'My Information', etc. Below the navigation bar is the 'Azure services' section, which includes a 'Create a resource' button and icons for 'Azure DevOps organizations', 'Subscriptions', 'Dashboard hub', 'Resource groups', 'Azure Load Testing', 'Quickstart Center', 'Azure AI services', 'Kubernetes services', and 'More services'. Under the 'Resources' section, there are 'Recent' and 'Favorite' tabs. The 'Recent' tab shows two items: 'Music' (Azure Load Testing) and 'Music_playlist_Batch_Creator' (Resource group), both last viewed 3 days ago. Below this is a 'Navigate' section with links to 'Subscriptions', 'Resource groups', 'All resources', and 'Dashboard'. The 'Tools' section includes links to 'Microsoft Learn', 'Azure Monitor', 'Microsoft Defender for Cloud', and 'Cost Management'. The 'Useful links' section features a link to 'Azure mobile app'.

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

This screenshot is similar to the previous one but with a search bar at the top containing the text 'DevOps'. The search results are displayed below the search bar. In the 'Services' section, 'Azure DevOps organizations' is highlighted. Other visible service suggestions include 'Azure Native New Relic Service', 'Managed DevOps Pools', 'Azure Native Dynatrace Service', and 'Static Web App'. The 'Marketplace' section lists 'Rocky Linux 9', 'Build Agents for Azure DevOps', and 'InfluxDB Cloud (Official Version)'. The 'Documentation' section includes links to 'DevOps architecture design - Azure Architecture Center', 'Secure your Azure DevOps - Azure DevOps', and 'Course A2-400T00-A: Designing and Implementing Microsoft DevOps solutions - T...'. The 'Tools' section shows 'Microsoft Entra ID', 'Azure DevOps', and 'Continue searching in Microsoft Entra ID'.

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



Result:

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

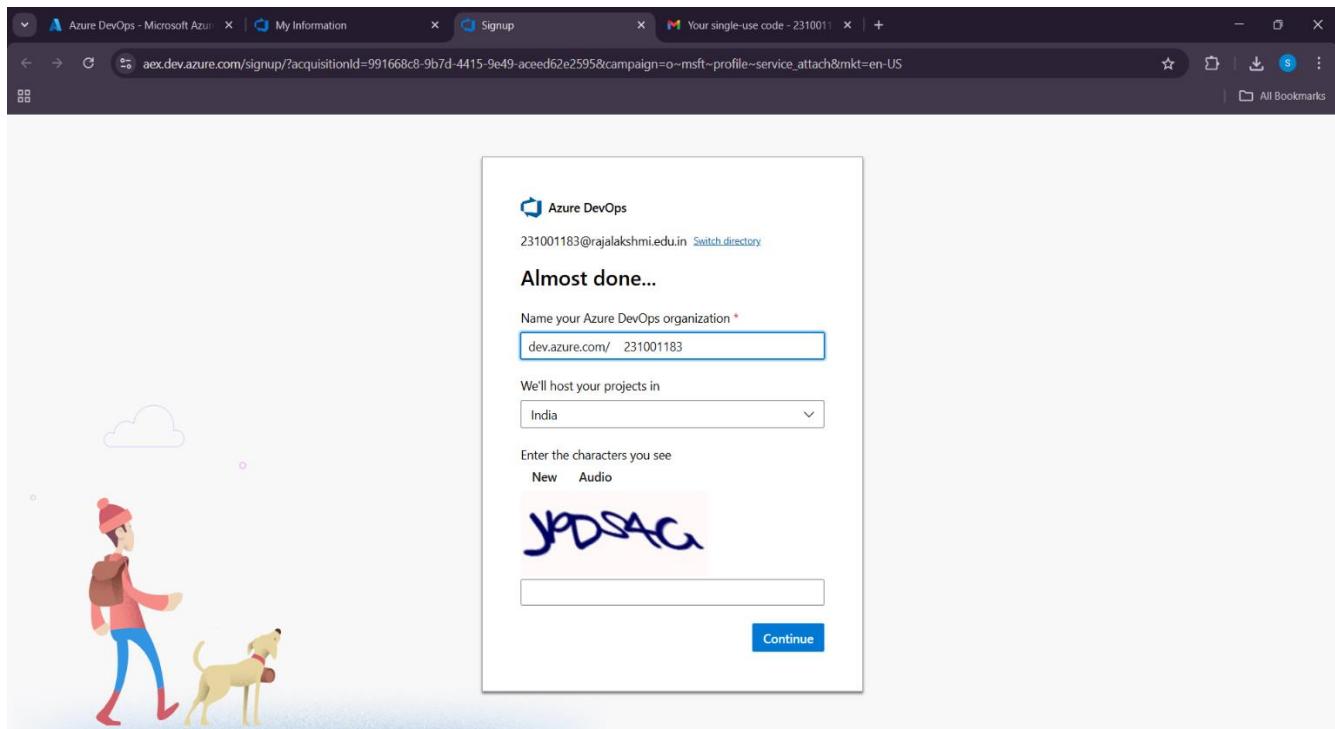
EXP NO: 2

AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

Aim:

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

1. Create An Azure Account



2. Create the First Project in Your Organization

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

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

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

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

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

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

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

Create new project

X

Project name *

SDMS (Student Data Management System)

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.

By creating this project, you agree to the Azure DevOps [code of conduct](#)

^ Advanced

Version control

Git

Work item process

Agile

Cancel

Create

- 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.

Azure DevOps Organizations

Sanjula S Sign out

Create new organization

Projects Actions

SDMS (Student Data Management System) Open in Visual Studio

Student Data Management System

New project

dev.azure.com/231001183 (Owner)

dev.azure.com/SoftwareConstructionProject (Member)

Microsoft account

India

231001183@rajalakshmi.edu.in

Visual Studio Dev Essentials

Get everything you need to build and deploy your app on any platform.

Use your benefits

1:54 pm 5/5/2025

4. Project dashboard

Azure DevOps SoftwareConstructionProject / SDMS / Overview / Summary

Search

Private Invite

SDMS

About this project

This project focuses on developing a Student Database Management System (SDMS) utilizing Microsoft Azure as the cloud platform. The primary objective is to store, manage, and access student-related data efficiently and securely in a digital format.

The system is designed to handle key student information such as personal details, academic records, attendance, and other relevant data. Leveraging Azure enables the system to benefit from cloud-based storage, high scalability, enhanced data security, and remote accessibility.

Key features of the system include:

- Adding, viewing, updating, and deleting student data
- Secure login and access control
- Reporting and data analysis capabilities
- A user-friendly interface for administrators, faculty, or students

This project demonstrates how modern cloud services like Microsoft Azure can streamline and enhance educational data management, resulting in a more efficient, reliable, and accessible system.

Project stats Period: Last 7 days

Boards

8 Work items 0 Work items

Members 5

SP SK SS SR SS

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. The left sidebar is titled 'SDMS' and includes options like Overview, Boards, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Project settings. The 'Backlogs' option is selected. The main area is titled 'SDMS Team' and shows a 'Backlog' table. The table has columns for Order, Work Item Type, Title, State, Effort, Business, Value Area, and Tags. There are six backlog items, all labeled 'Epic' and 'New', under the 'Business' category. The titles include 'Notifications & Compliance', 'Administration & Security', 'Student Self-Service', 'Attendance Management', 'Academic Management', and 'Student Enrollment & Information Management'. A search bar at the top right contains the text 'Search'.

The screenshot shows the Microsoft account sign-in page. At the top, there is a search bar, a Microsoft logo, and a 'Sign out' link. Below the Microsoft logo is a large red oval containing the letters 'SS'. To the right of the oval, the name 'Sanjula S' and the email address '231001183@rajalakshmi.edu.in' are displayed, along with links for 'My Microsoft account' and 'Switch directory'. On the left, there is a sidebar with categories: Business, Business, Business, Business, Business, and Business. At the bottom, there is a link for 'Sign in with a different account' and a circular profile icon.

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

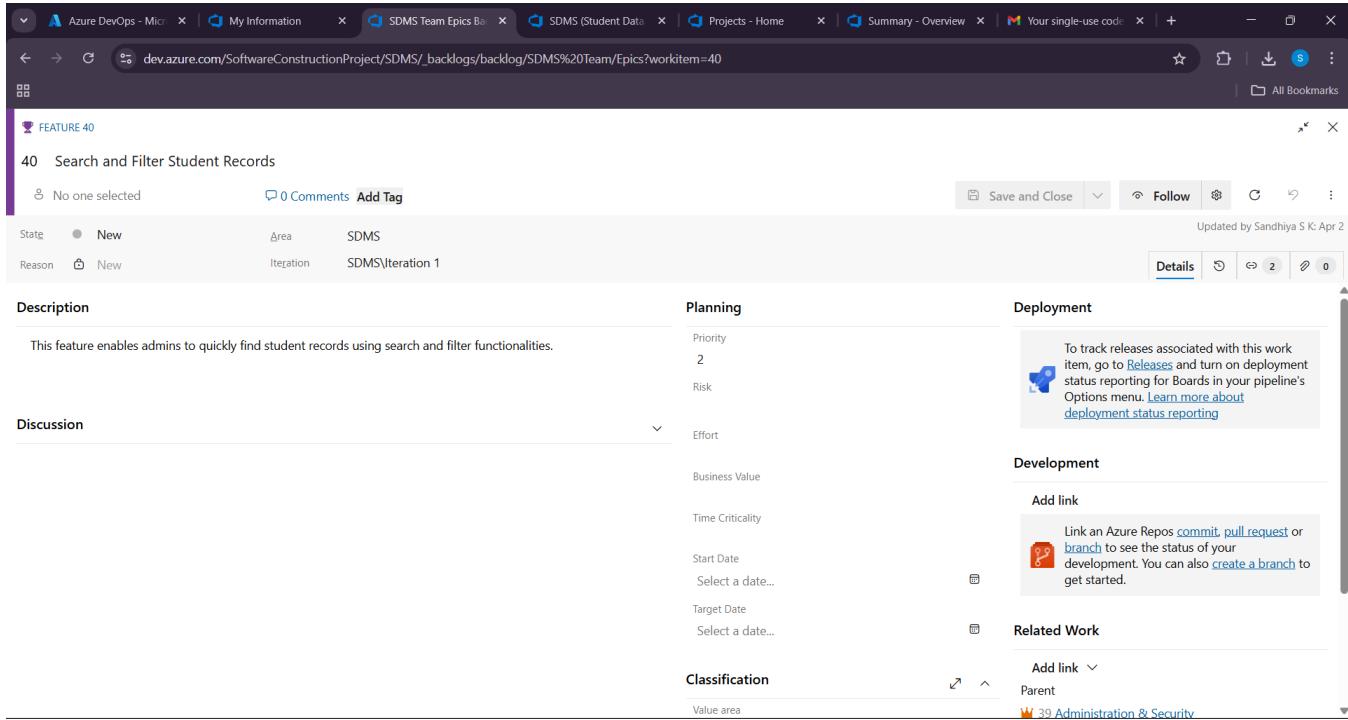
The screenshot shows the Azure DevOps interface for a project named "SoftwareConstructionProject". The left sidebar is open, showing options like Overview, Boards, Work items, Sprints, Queries, Delivery Plans, Analytics views, Repos, Pipelines, Test Plans, and Project settings. The main area is titled "SDMS Team" and displays a "Backlog" view. The backlog table has columns for Order, Work Item Type, Title, State, Effort, Business Value, Area, and Tags. There are 85 items in the backlog. The first few items are:

Order	Work Item Type	Title	State	Effort	Business Value	Area	Tags
1	Epic	> Notifications & Compliance	New			Business	
2	Epic	> Administration & Security	New			Business	
	Feature	> Search and Filter Student Records	New			Business	
	User Story	Search and Filter Student Records	New			Business	
	Feature	> Role-Based Access Control	New			Business	
3	Epic	> Student Self-Service	New			Business	
4	Epic	> Attendance Management	New			Business	
	+	+ 5 Epic > Academic Management	New			Business	
6	Epic	> Student Enrollment & Information Management	New			Business	

1. Fill in Epics

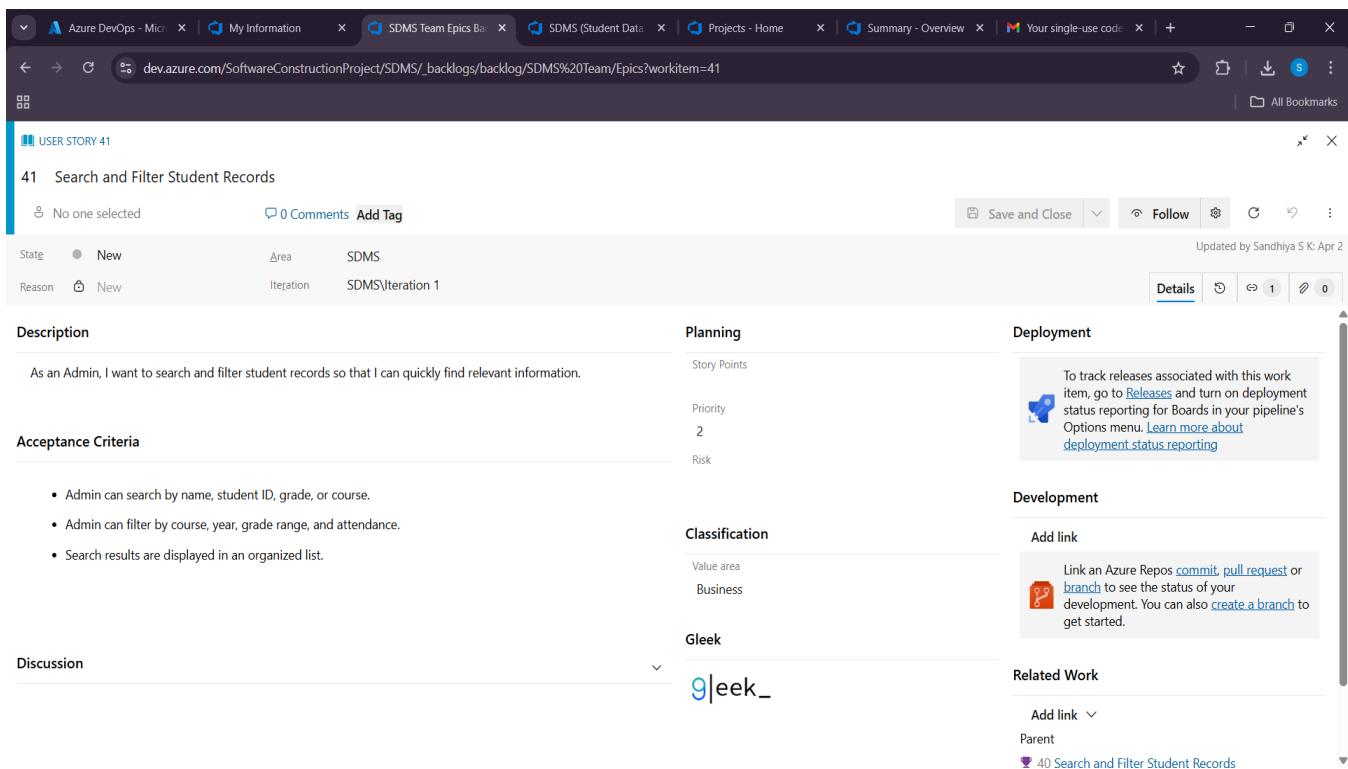
The screenshot shows the details of an epic named "EPIC 39: Administration & Security". The top navigation bar includes links for My Information, SDMS Team, SDMS (Student Data), Projects - Home, Summary - Overview, and Your single-use code. The main content area shows the epic's title, state (New), area (SDMS), iteration (SDMS\Iteration 1), and a description: "This epic focuses on administrative capabilities, including searching student records, managing user roles, and ensuring data security." The "Planning" section includes priority (2), risk, effort, business value, and time criticality. The "Deployment" section includes deployment status and a link to Azure Repos. The "Development" section includes a link to commit and pull request. The "Related Work" section lists a child epic ("42 Role-Based Access Control") and a related user story ("40 Search and Filter Student Records").

2. Fill in Features



A screenshot of the Azure DevOps interface showing a feature card. The title is "FEATURE 40" and the description is "40 Search and Filter Student Records". The feature is marked as "New" and assigned to the "SDMS" area. The "Reason" is "New" and the "Iteration" is "SDMS\Iteration 1". The "Description" section states: "This feature enables admins to quickly find student records using search and filter functionalities." The "Planning" section shows a priority of 2 and a risk level of 1. The "Deployment" section includes a note about tracking releases and deployment status reporting. The "Development" section has a placeholder for linking an Azure Repos branch. The "Related Work" section lists a parent item and a link to "Administration & Security".

3. Fill in User Story Details



A screenshot of the Azure DevOps interface showing a user story card. The title is "USER STORY 41" and the description is "41 Search and Filter Student Records". The user story is marked as "New" and assigned to the "SDMS" area. The "Reason" is "New" and the "Iteration" is "SDMS\Iteration 1". The "Description" section states: "As an Admin, I want to search and filter student records so that I can quickly find relevant information." The "Acceptance Criteria" section lists: "Admin can search by name, student ID, grade, or course.", "Admin can filter by course, year, grade range, and attendance.", and "Search results are displayed in an organized list.". The "Planning" section shows a story point of 1 and a priority of 2. The "Deployment" section includes a note about tracking releases and deployment status reporting. The "Development" section has a placeholder for linking an Azure Repos branch. The "Related Work" section lists a parent item and a link to "Administration & Security".

Result:

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

EXP NO: 4

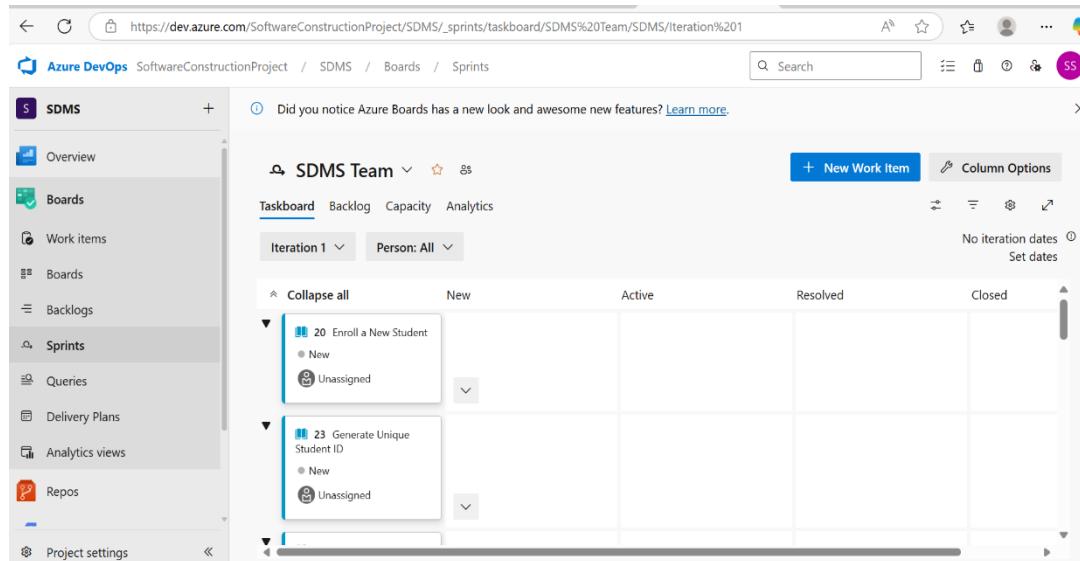
SPRINT PLANNING

Aim:

To assign user story to specific sprint for the Student Database Management System Project.

Sprint Planning

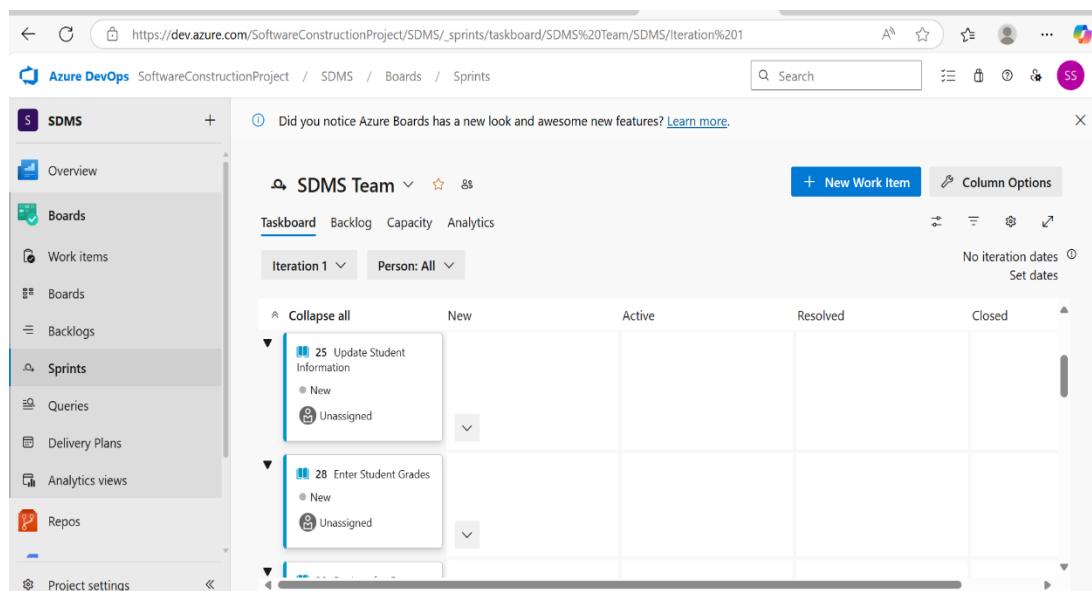
Sprint 1



The screenshot shows the Azure Boards Taskboard for the SDMS project. The left sidebar has 'Sprints' selected. The main area shows a table with two rows, each representing a user story. Both stories are in the 'New' state and are unassigned. The columns are labeled: New, Active, Resolved, and Closed.

New	Active	Resolved	Closed
20 Enroll a New Student ● New 👤 Unassigned			
23 Generate Unique Student ID ● New 👤 Unassigned			

Sprint 2



The screenshot shows the Azure Boards Taskboard for the SDMS project. The left sidebar has 'Sprints' selected. The main area shows a table with two rows, each representing a user story. Both stories are in the 'New' state and are unassigned. The columns are labeled: New, Active, Resolved, and Closed.

New	Active	Resolved	Closed
25 Update Student Information ● New 👤 Unassigned			
28 Enter Student Grades ● New 👤 Unassigned			

Sprint 3

The screenshot shows the Azure Boards Taskboard for the SDMS Team. The left sidebar is titled 'SDMS' and includes options like Overview, Boards, Work items, Backlogs, Sprints (selected), Queries, Delivery Plans, Analytics views, Repos, and Project settings. The main area is titled 'SDMS Team' and shows a Taskboard with 'Iteration 1'. The backlog contains two items:

- 30 Register for Courses (New, Unassigned)
- 33 Track Student Attendance (New, Unassigned)

Filter options include 'Person: All' and 'Iteration 1'. Buttons for '+ New Work Item' and 'Column Options' are at the top right.

Sprint 4

The screenshot shows the Azure Boards Taskboard for the SDMS Team. The left sidebar is titled 'SDMS' and includes options like Overview, Boards (selected), Work items, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, and Project settings. The main area is titled 'SDMS Team' and shows a Taskboard with 'Iteration 1'. The backlog contains two items:

- 36 Generate Academic Report (New, Unassigned)
- 38 View Student Profile (New, Unassigned)

Filter options include 'Person: All' and 'Iteration 1'. Buttons for '+ New Work Item' and 'Column Options' are at the top right.

Sprint 5

The screenshot shows the Azure DevOps Boards Taskboard for the SDMS team. The sidebar on the left lists project navigation options like Overview, Boards, Work items, etc. The main area displays a taskboard with columns for New, Active, Resolved, and Closed work items. Two specific tasks are highlighted with blue boxes:

- 41 Search and Filter Student Records**: Status: New, Unassigned.
- 44 Manage User Roles and Permissions**: Status: New, Unassigned.

Sprint 6

The screenshot shows the Azure DevOps Boards Taskboard for the SDMS team. The sidebar on the left lists project navigation options like Overview, Boards, Work items, etc. The main area displays a taskboard with columns for New, Active, Resolved, and Closed work items. Two specific tasks are highlighted with blue boxes:

- 47 Receive Academic Notifications**: Status: New, Unassigned.
- 49 Perform Data Backup and Recovery**: Status: New, Unassigned.

Sprint 7

The screenshot shows the Azure DevOps Taskboard for the SDMS Team, specifically for Sprint 7. The board displays two tasks:

- 49 Perform Data Backup and Recovery**: Status: New, Assigned To: Unassigned
- 51 Ensure Data Privacy and Compliance**: Status: New, Assigned To: Unassigned

The Taskboard interface includes a sidebar with project navigation options like Overview, Boards, Backlogs, Sprints, Queries, Delivery Plans, Analytics views, Repos, and Project settings. The main header bar shows the URL https://dev.azure.com/SoftwareConstructionProject/SDMS/_sprints/taskboard/SDMS%20Team/SDMS/Iteration%201.

Result:

The Sprints are created for the Student Database Management System project.

EXP NO: 5

POKER ESTIMATION

Aim:

Create Poker Estimation for the user stories - Student Database Management System Project.

Poker Estimation

The screenshot shows a detailed view of a User Story card in Azure DevOps. The card is titled "47 Receive Academic Notifications". It includes sections for Description, Planning, Deployment, Classification, and Development. The Description section contains a bullet point about receiving notifications. The Planning section shows Story Points as 2 and Priority as 2. The Deployment section has a note about tracking releases. The Classification section lists Value area as Business and Gleek as aleek. The Development section provides instructions for linking to Azure Repos. The card also shows basic metadata like State (New), Area (SDMS), and Iteration (SDMS\Iteration 1).

Result:

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

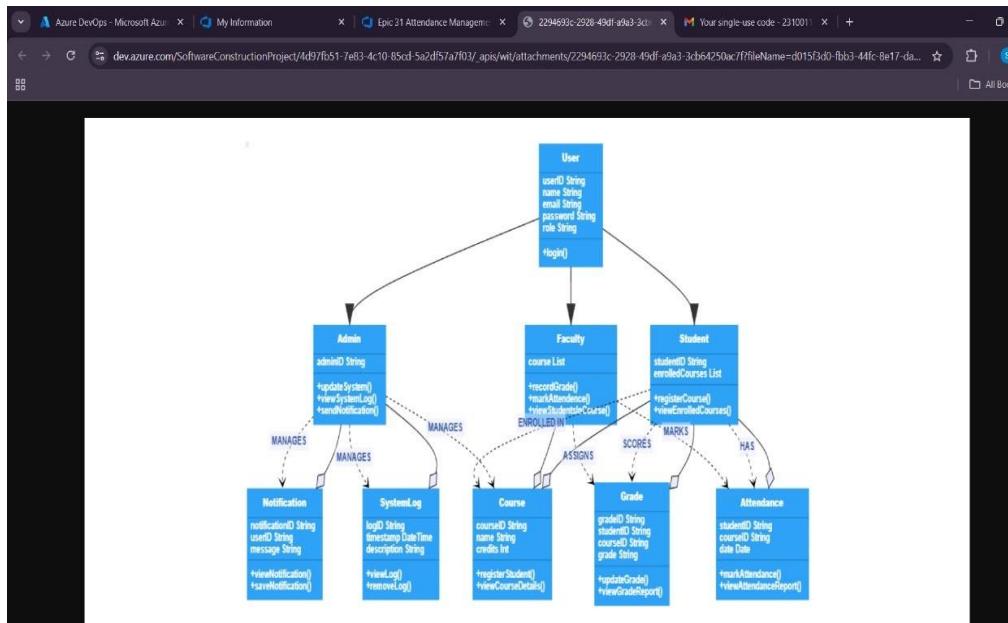
EXP NO: 6

DESIGNING CLASS AND SEQUENCE DIAGRAMS FOR PROJECT ARCHITECTURE

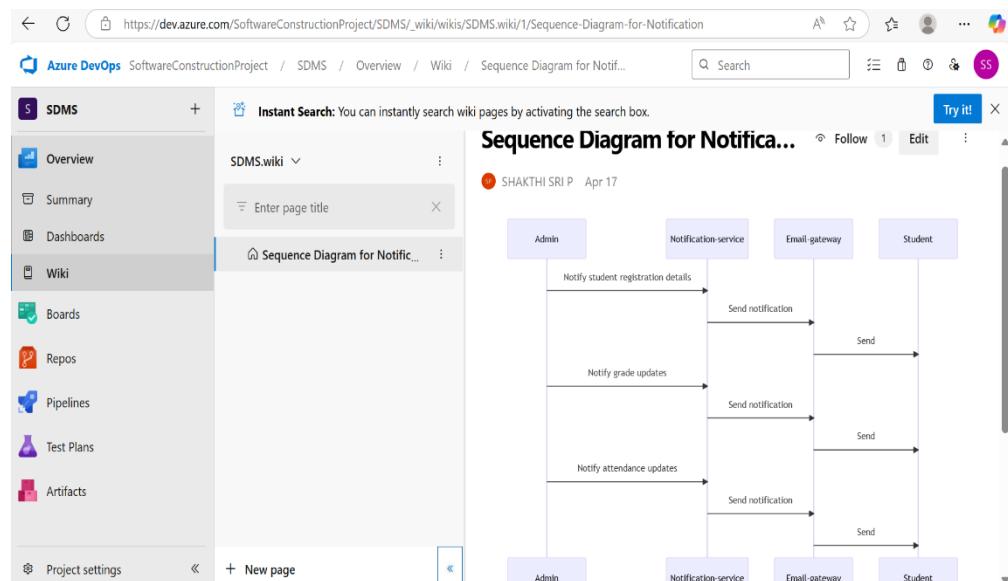
Aim:

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 Student Database Management System.

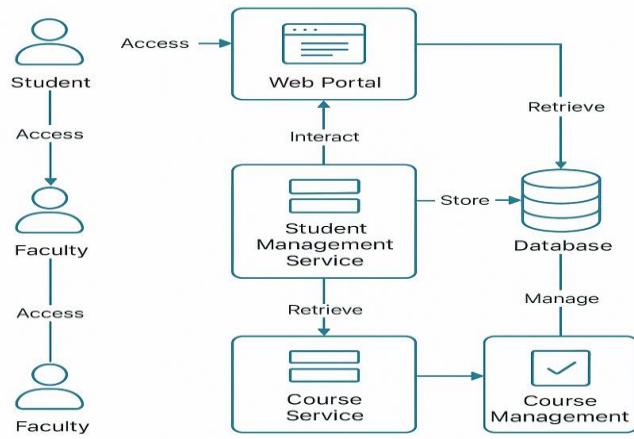
EXP NO: 7

DESIGNING ARCHITECTURAL AND ER DIAGRAMS FOR PROJECT STRUCTURE

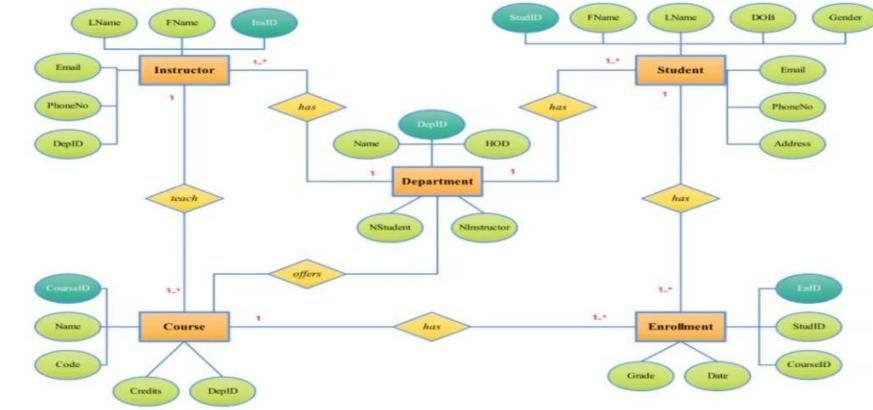
Aim:

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 Student Database Management System.

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. Understand Core Features of the Application**

- Student Signup & Login
- Adding, Viewing, and Editing Student Records
- Fetching Academic and Attendance Reports
- Managing Courses and Enrollments
- Generating and Exporting Performance Reports

2. Define User Interactions

- Each test case simulates real user behavior (e.g., admin logging in, adding a student, updating course info).

3. Design Happy Path Test Cases

- Focused on validating that all features function correctly under expected inputs and conditions.
- **Example:** Admin logs in successfully, adds a new student, enrolls them in a course, and views their report.

4. Design Error Path Test Cases

- Simulate invalid actions and unexpected scenarios to verify error handling and robustness.
- **Example:** Login fails with incorrect credentials, adding a student with missing mandatory fields, trying to enroll in a full course.

5. Break Down Steps and Expected Results

- Each test case should contain clear step-by-step actions with corresponding expected outcomes.
- Helps testers and automation tools follow the exact flow of operations and validations.

6. Use Clear Naming and IDs

- Name test cases clearly (e.g., TC01 – Admin Login Success, TC07 – Add Student Validation Error).
- Enables quick reference and linking to specific features or user stories.

7. Separate Test Suites

- Group test cases based on core modules (e.g., Authentication, Student Management, Course Management, Reports).

- Improves organization and systematic test execution (e.g., in tools like Azure DevOps or Jira).

8. Prioritize and Review

- High-priority is assigned to critical tasks (e.g., adding/editing student data, generating reports).
- All test cases are reviewed for coverage, clarity, and alignment with system requirements

1. Test plans

The screenshot shows the Azure DevOps interface for 'SDMS' under 'Testing'. On the left sidebar, 'Test plans' is selected. In the main area, a 'Test Suites' section lists 'Testing', '26 : Academic Management (1)', and '18 : Student Enrollment & Inf...'. A detailed view for '18 : Student Enrollment & Information Management (ID: 56)' is open, showing 'Test Points (2 items)'. The table includes columns for Title, Outcome (Passed or Active), Order, and Test Case Id. The first row for 'Student login' has an 'Outcome' of 'Passed' (green dot) and 'Order' of 1, 'Test Case Id' of 55. The second row for 'Admin login' has an 'Outcome' of 'Active' (blue dot) and 'Order' of 2, 'Test Case Id' of 57.

Title	Outcome	Order	Test Case Id
Student login	Passed	1	55
Admin login	Active	2	57

test plans

2. Test case creation

The screenshot shows the 'TEST CASE 55' page. At the top, it says '55 Student login' and 'Sandhiya S K'. Below that, there are fields for 'State' (Design), 'Reason' (New), 'Area' (SDMS), and 'Iteration' (SDMS\Iteration 1). On the right, there are buttons for 'Save and Close', 'Follow', and other options. The 'Steps' tab is active, showing a table with steps like 'go to application.', 'login with username as "thanks" and password as "studentpass1".', and 'click enter.'. The 'Expected result' column contains 'student dashboard must be seen'. The 'Deployment' tab indicates 'Updated by Sandhiya S K: Just now'. The 'Development' tab has a note about tracking releases. The 'Related Work' tab shows an 'Add link' button.

3.Execution

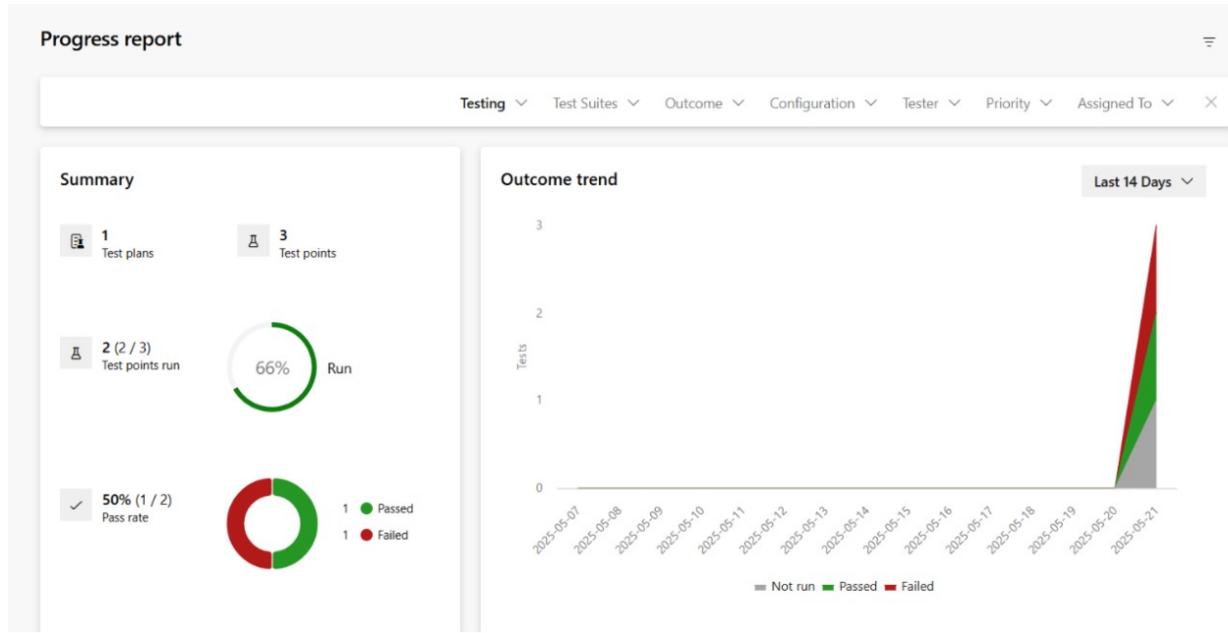


4.Results

A screenshot of the Azure DevOps Test Plans interface. On the left, a sidebar shows 'SDMS' and 'Test Plans' selected. The main area shows a 'Testing' view with a 'Test Suites' section. A test suite named '18 : Student Enrollment & Information Management (ID: 56)' is selected. The 'Execute' tab is active, showing 'Test Points (2 items)'. A table lists two test points: 'Student login' and 'Admin login', both marked as 'Passed'. The table includes columns for 'Title', 'Outcome', 'Order', and 'Test Case Id'. The 'Run for web application' button is visible at the top right of the table.

results

5.Final Progress Report



1. Student Login

The **Student Login** module is responsible for allowing authenticated access to student-specific data. Testing ensures that only valid student users can access the system and that security, accessibility, and session behavior function correctly in the Azure-hosted environment.

Testing Objectives:

- Validate authentication flow using test credentials.
- Ensure proper error handling on invalid attempts.
- Verify response time and session handling via Azure services.
- Test login availability across different environments.

2. Admin Login

The **Admin Login** module enables secure access for authorized personnel to perform administrative operations. Testing in the Azure environment focuses on ensuring proper access control, authentication, and session integrity for admin users.

Testing Objectives:

- Ensure only authorized users can log in to the admin panel.
- Validate role-based access control.
- Test authentication failures and system response.
- Check Azure-hosted identity services (if used) for reliability.

3.Admin Management

The **Admin Management** module includes all backend operations performed by the admin, such as managing student records, generating reports, or adjusting user access. Testing ensures data integrity, permission checks, and system reliability under Azure's infrastructure.

Testing Objectives:

- Verify admin operations like create, update, and delete student records.
- Ensure appropriate validations for all data fields.
- Test concurrency and data consistency in a multi-user environment.
- Validate that operations reflect accurately in the cloud database

Result:

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

231001177

CS23432

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.

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.

Microsoft Azure

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

Copilot

231001177@rajalakshmi...
DEFAULT DIRECTORY (231001177...)

Home > Azure Load Testing >

Create a load testing resource

Basics Encryption Tags Review + create

Azure Load Testing is a fully managed load-testing service that makes it easy to generate high-scale load and identify performance bottlenecks. [Learn more](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *

Resource group * [Create new](#)

Instance details

Name *

Region *

[Previous](#) [Next](#) [Review + create](#)

Home > Azure Load Testing >

Create a load testing resource

 Validation passed.

Basics Encryption Tags [Review + create](#)

Basics

Subscription	Azure for Students
Resource group	SDMS
Name	SDMS
Region	East US

Encryption

Encryption type	MMK
-----------------	-----

[Previous](#) [Next](#) [Create](#)

Microsoft Azure Search resources, services, and docs (G+) Copilot

Home >

Microsoft.CloudNativeTesting1747839500538 | Overview



Deployment

X

«



Delete



Cancel



Redeploy



Download



Refresh

Overview



Inputs



Outputs



Template

Your deployment is complete



Deployment name : Microsoft.CloudNativeTesting1747839500538

Start time : 5/21/2025, 8:28:49 PM

Subscription : Azure for Students

Correlation ID : 54bc6e25-964c-4f45-9d62-4ed9d67413af

Resource group : SDMS

Deployment details

Next steps

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

The screenshot shows the Microsoft Azure Load Testing interface. At the top, there's a navigation bar with 'Microsoft Azure', a search bar, and various icons. Below it, the URL is 'Home > Microsoft.CloudNativeTesting1747839500538 | Overview > SDMS | Tests >'. The main title is 'Test_5/21/2025_8:30:37 PM' with a timestamp of '8:30:37 PM'. A note says 'Last updated by: 231001177@rajalakshmi.edu.in | Last updated on: 5/21/2025, 8:38 PM'. Below the title are buttons for 'View all tests', 'Run', 'Configure', 'Delete test', 'Convert to JMX Test', 'Clone', and 'Refresh'. A section titled 'Test details' shows a table with one row: Name 'Test_5/21/2025_8:30:37 PM', Test ID 'c0421290-d956-482c-bdf1-046502...', Test type 'URL', Description 'Authentication Verification', Updated on '5/21/2025, 8:38:43 PM', and Updated by '231001177@rajalakshmi.edu.in'. A 'Schedule' section indicates no schedule is present. The 'Test runs' tab is selected, showing a table with two rows of test runs. The columns are Name, Description, Duration (in minutes), Virtual users (average), Virtual user hours, Trigger, Run by, Test result, Status, and an action column. The first run is 'TestRun_5/21/2025_8:39:56 PM' with status 'Not Applicable' and 'Provisioning'. The second run is 'TestRun_5/21/2025_8:38:42 PM' with status 'Not Applicable' and 'Executing'.

Name	Test ID	Test type	Description	Updated on	Updated by
Test_5/21/2025_8:30:37 PM	c0421290-d956-482c-bdf1-046502...	URL	Authentication Verification	5/21/2025, 8:38:43 PM	231001177@rajalakshmi.edu.in

Name	Description	Duration (in minutes)	Virtual users (average)	Virtual user hours	Trigger	Run by	Test result	Status
TestRun_5/21/2025_8:39:56 PM					Manual	231001177@rajalaksh...	Not Applicable	Provisioning
TestRun_5/21/2025_8:38:42 PM					Manual	231001177@rajalaksh...	Not Applicable	Executing

Home > Test_5/21/2025_8:30:37 PM >

TestRun_5/21/2025_8:39:56 PM

Last updated by: 231001177@rajalakshmi.edu.in | Initiated on: 5/21/2025, 8:40 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 \(10s\)](#)

5/21/2025, 8:40:08 PM

1

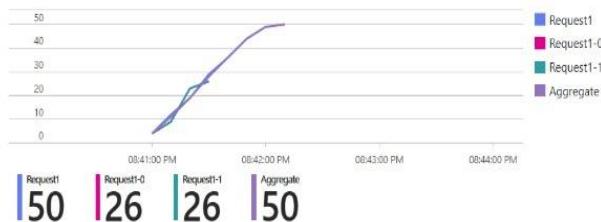
Executing

Load test results Engine health

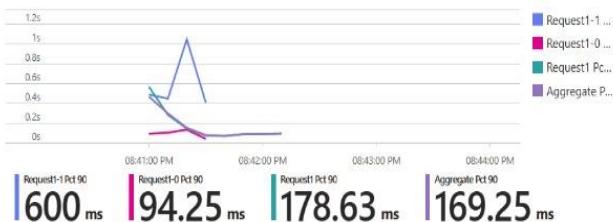
Client-side metrics

Requests : All Region : 0 Aggregation : P90 Error type : All Time range : 5/21/2025, 8:40:08 PM - 5/21/2025, 8:43:55 PM Group by : 10s

Virtual Users (Max)



Response time (successful responses)

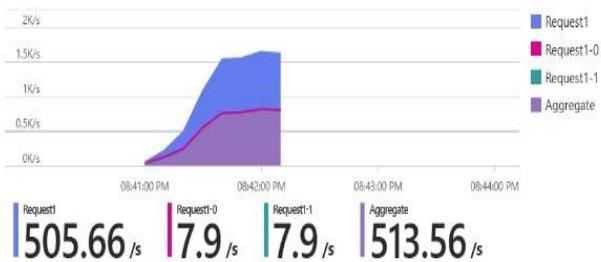


[View all test runs](#) [Stop](#) [Refresh](#) [Rerun](#) [Compare](#) [App components](#) [Configure metrics](#) [Download](#) [Copy artifacts](#) [Share](#) [Delete test run](#) [Mark as baseline](#) [Auto refresh on \(10s\)](#)

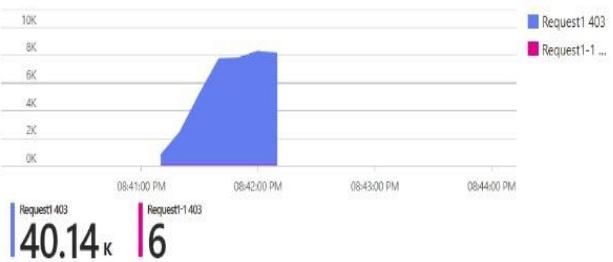
100 120 120 100

1000 ms 94.25 ms 110.00 ms 109.25 ms

Requests/sec (Avg)



Errors (total)



Server-side metrics

You have not configured any app components. Configure app components to see server side metrics.

Login

Login as:

Admin

Username:

admin1

Password:

.....

Login

Welcome Admin, admin01!

[Logout](#)

[View Student List](#)

[View Faculty List](#)

[Add Student](#)

[Add Faculty](#)

[View Courses](#)

Students

[Back to Dashboard](#)

ID	Name	Email	Attendance	Grades	Enrolled Courses	
S001	Tom Hanks	tom@university.edu	{"CS101": 89.0, "CS102": 90}	CS101:A, CS102:B	Introduction to Computer Science, Data Structures	Edit Delete
S002	Shakthi	shak@gmail.com	{"CS104": 90}	CS104:A+	Operating Systems	Edit Delete
S003	Meera Patel	meera@gmail.com	{"MA101": 90}	MA101:O	Probability	Edit Delete
S004	Arjun V	arjun@gmail.com	{"CS102": 78}	CS102:A+	Data Structures	Edit Delete
S005	Sai Krishna	sai@gmail.com	{"CS101": 89.0}	CS101:O	Introduction to Computer Science	Edit Delete
S006	Priya V	pri@gmail.com	None	None	Python	Edit Delete
S007	Priya K	pri@gmail.com	{"MA101": 90}	MA101:O	Probability	Edit Delete

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 Student Database Management System project.

GitHub Project Structure

This screenshot shows a GitHub repository named "Student-Database-Management-System---Microsoft-Azure". The "Issues" tab is selected, displaying seven open issues. The issues are categorized under "Architecture Diagram", "ER diagram", "Sequence diagram", "Class diagram", "Epic", "Features", and "User story". Each issue has a green circular icon with a white dot and a number, indicating its ID. The issues were opened at different times, ranging from 1 minute ago to 9 minutes ago. The interface includes standard GitHub navigation and search tools.

This screenshot shows a GitHub repository named "SC-Project" (Public). A blue banner at the top indicates that "@Sandhiya1211 has invited you to collaborate on this repository". Below the banner, there is a "View invitation" button. The repository details show 1 branch and 0 tags. A file named "sdms.zip" was added via upload last week. On the right side, there are sections for "About", "Releases", and "Packages", all of which currently have no information. The repository has 1 commit and 0 stars, with 1 person watching it and 0 forks.

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.