INDEX

EXP NO: 1	AZURE DEVOPS ENVIRONMENT SETUP

AIM

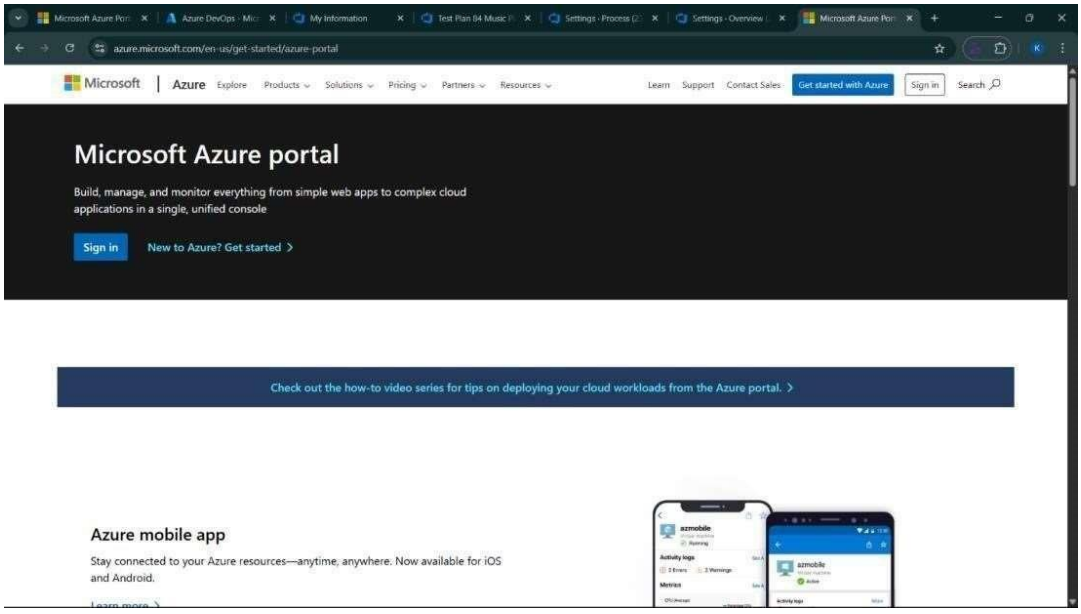
To set up and access the Azure DevOps environment by creating an organization through the Azure portal.

INSTALLATION

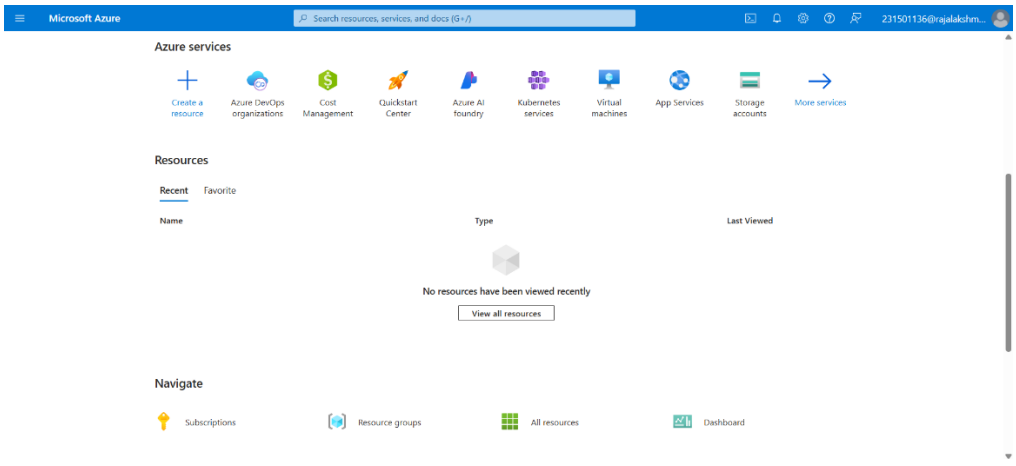
1.Open your web browser and go to the Azure website: https://azure.microsoft.com/enus/getstarted/azureportal.

Sign in using your Microsoft account credentials.

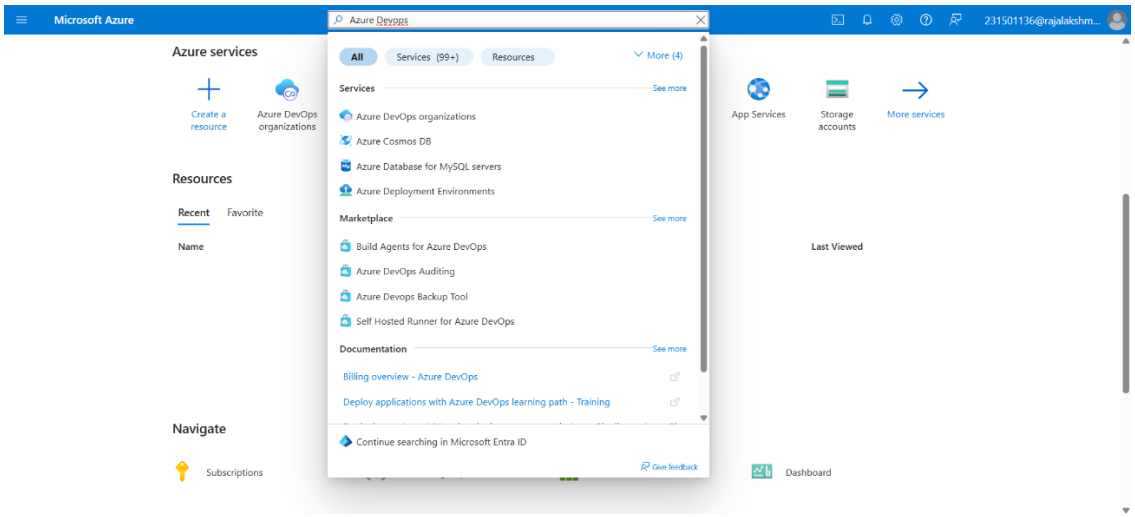
If you don't have a Microsoft account, you can create one here: https://signup.live.com/?lic=1



2.Azure home page



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



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

RESULT

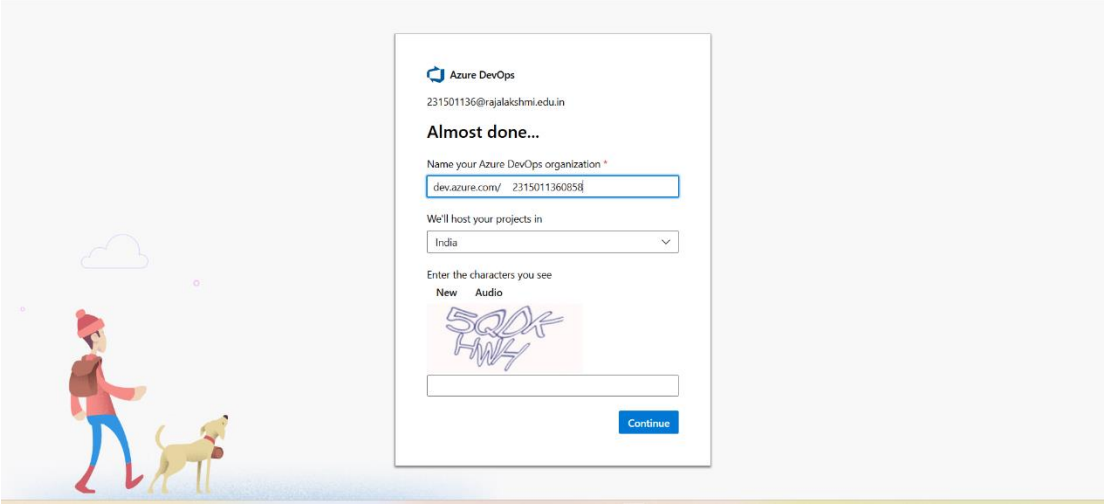
Successfully accessed the Azure DevOps environment and created a new organization through the Azure portal.

EXP NO: 2	AZURE DEVOPS PROJECT SETUP AND USER STORY MANAGEMENT

AIM

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

1.Create An Azure Account



2.Create the First Project in Your Organization

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

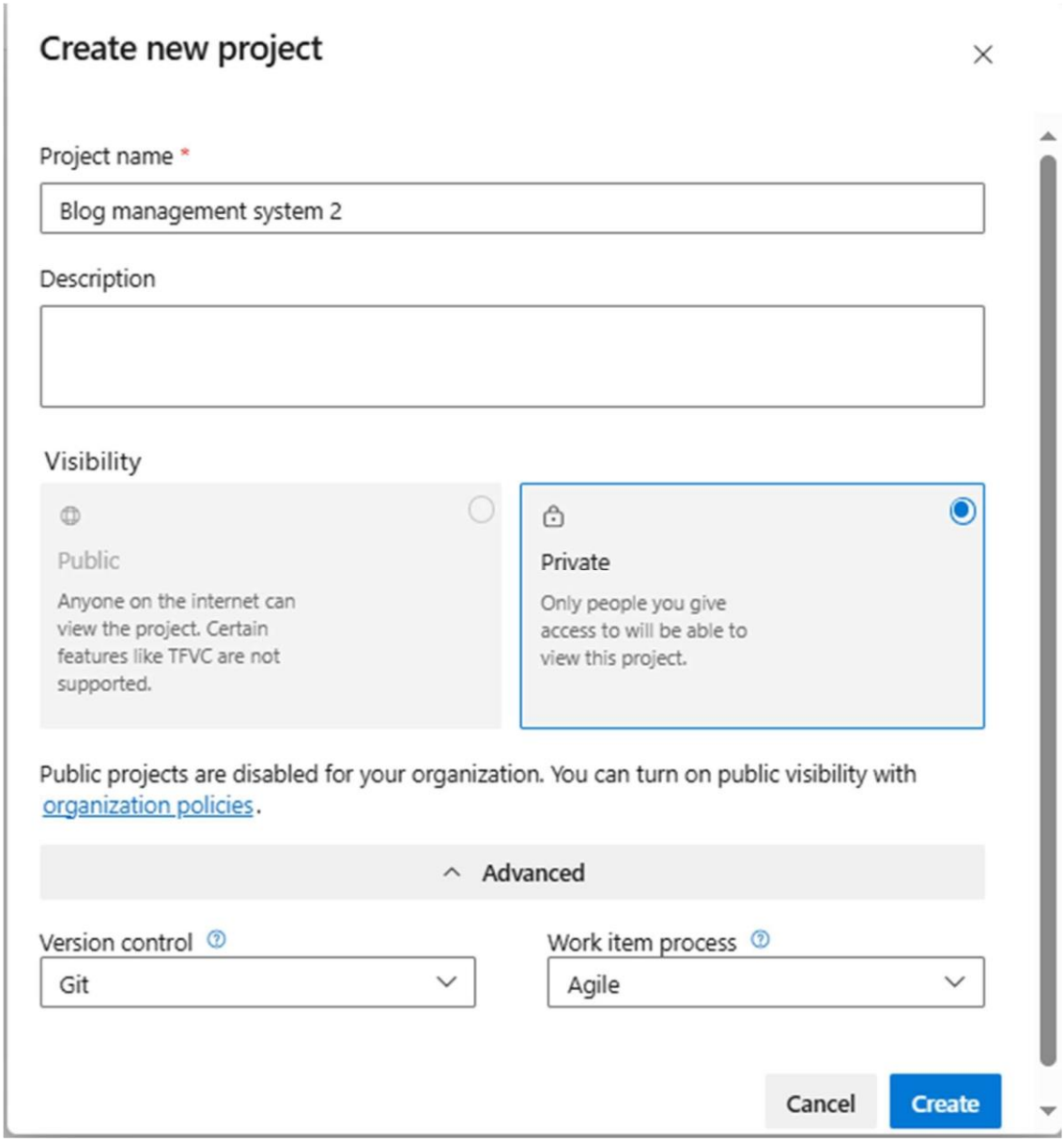
b. On the organization's Home page, click on the New Project button.

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

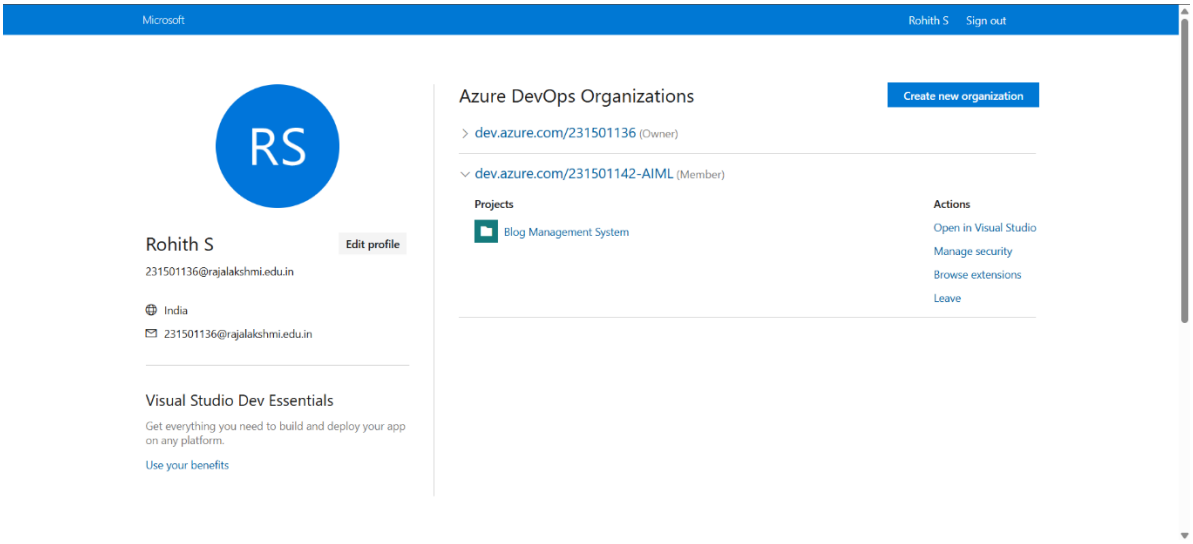
Name: Choose a name for the project (e.g., LMS).

Description: Optionally, add a description to provide more context about the project. Visibility: Choose whether you want the project to be Private (accessible only to those invited) or Public (accessible to anyone).

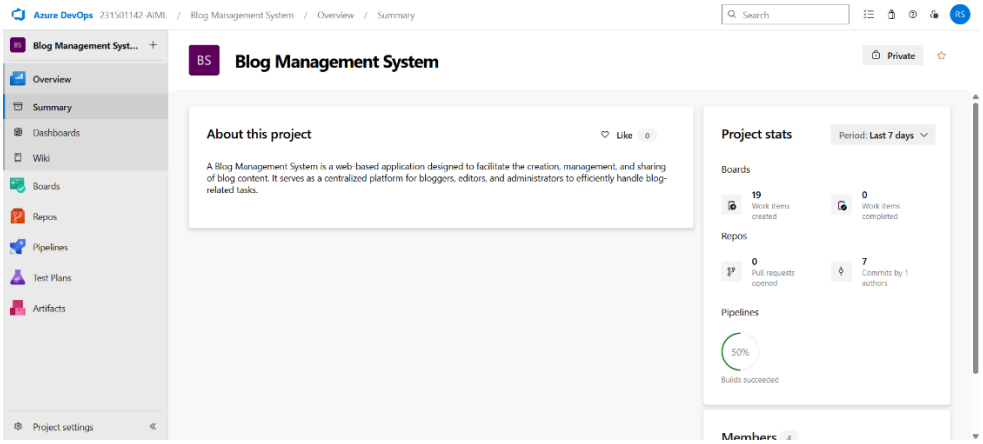
d. Once you've filled out the details, click Create to set up your first project.



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



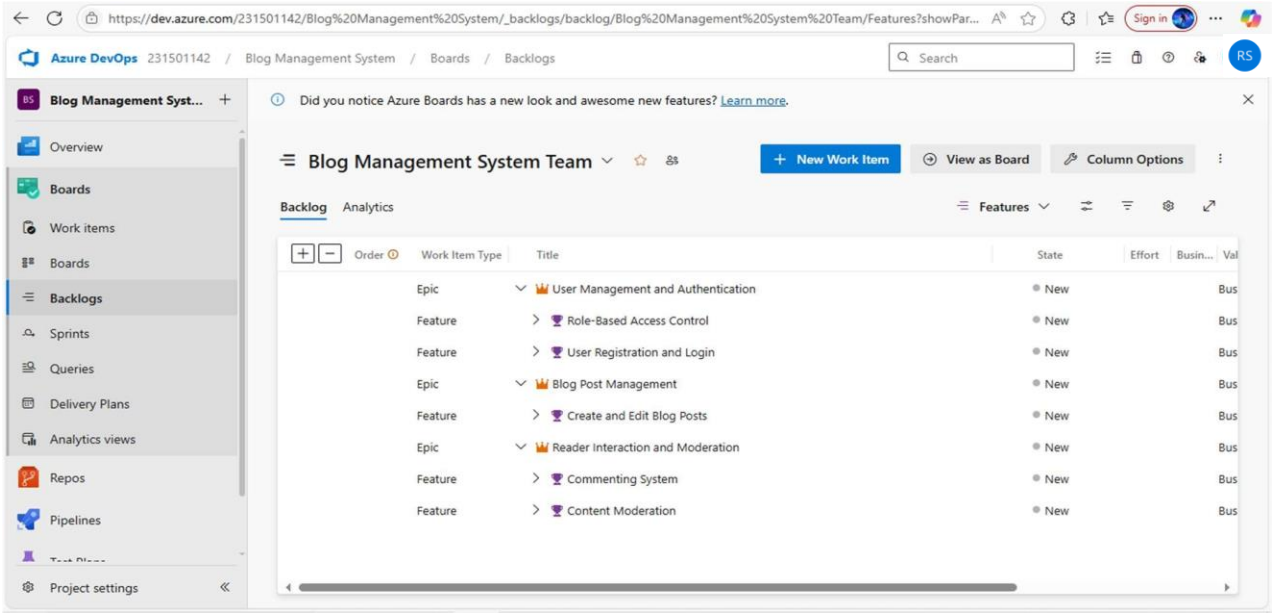
4.Project dashboard



5.To manage user stories:

a. From the left-hand navigation menu, click on Boards. This will take you to the main Boards page, where you can manage work items, backlogs, and sprints.

b. On the work items page, you'll see the option to Add a work item at the top. Alternatively, you can find a + button or Add New Work Item depending on the view you're in. From the Add a work item dropdown, select User Story. This will open a form to enter details for the new User Story.



RESULT

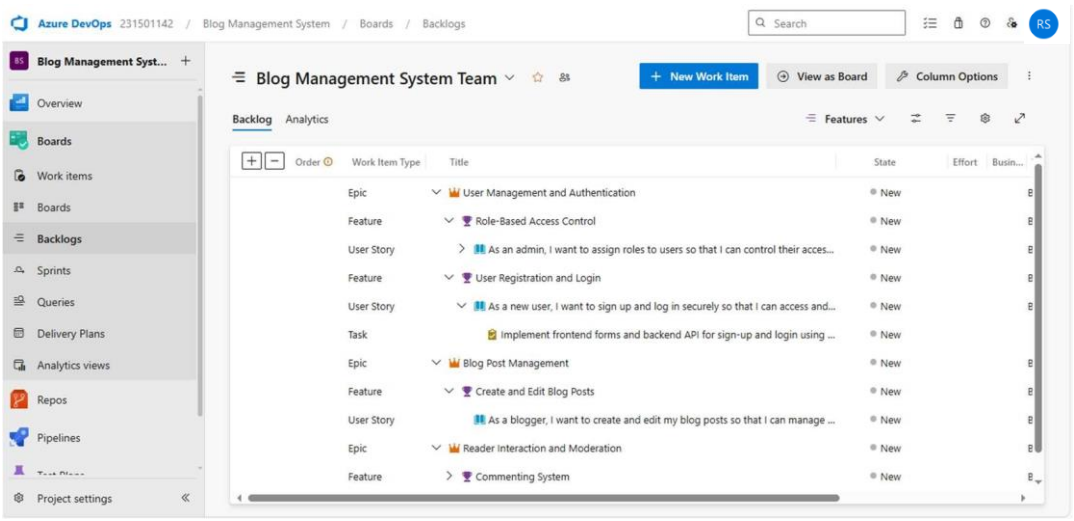
Successfully created an Azure DevOps project with user story management and agile workflow setup.

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

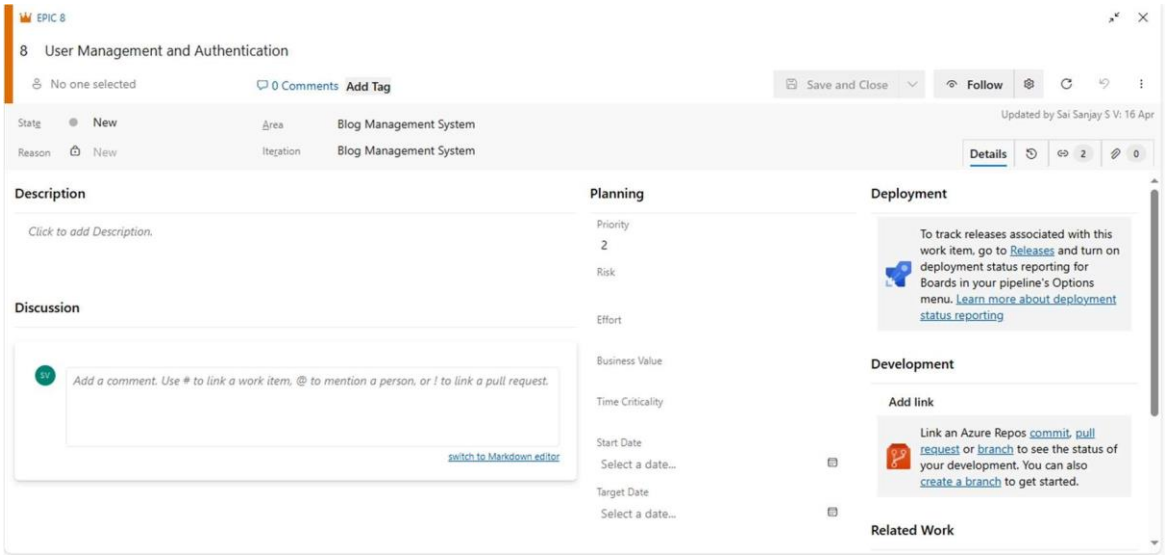
AIM

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

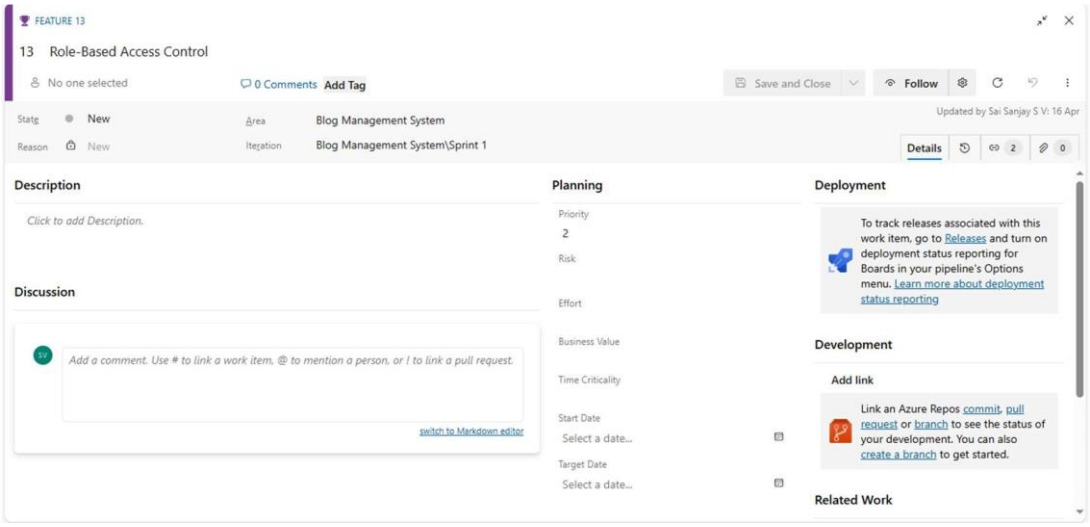
Create Epic, Features, User Stories, Task



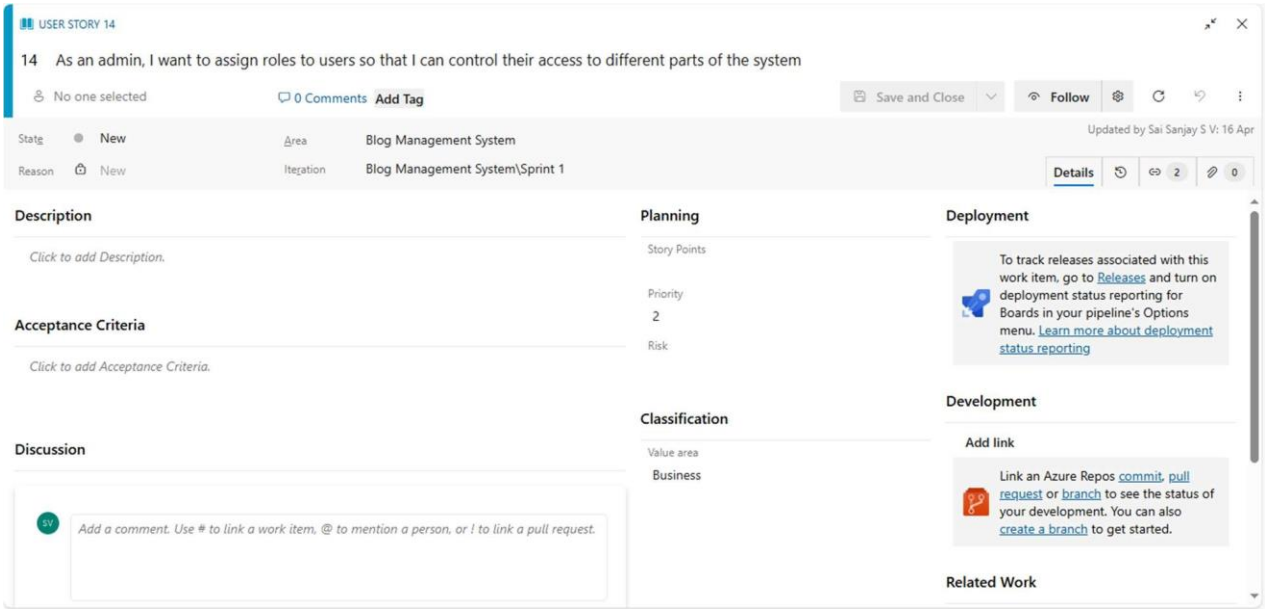
1.Fill in Epics



2.Fill in Features



3.Fill in User Story Details



RESULT

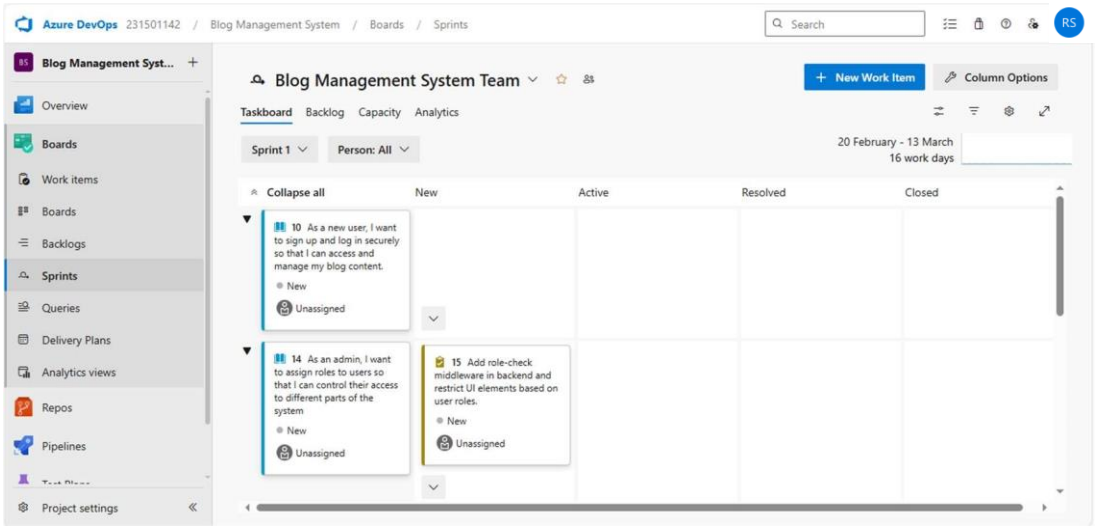
Thus, the creation of epics, features, user story and task has been created successfully.

EXP NO: 4	SPRINT PLANNING

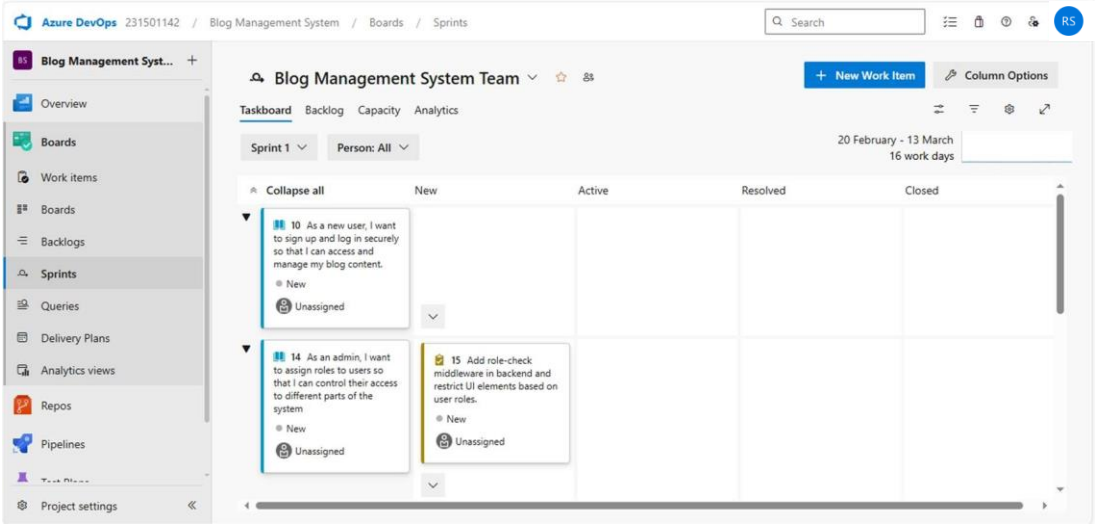
AIM

To assign user story to specific sprint for the Blog Management System.

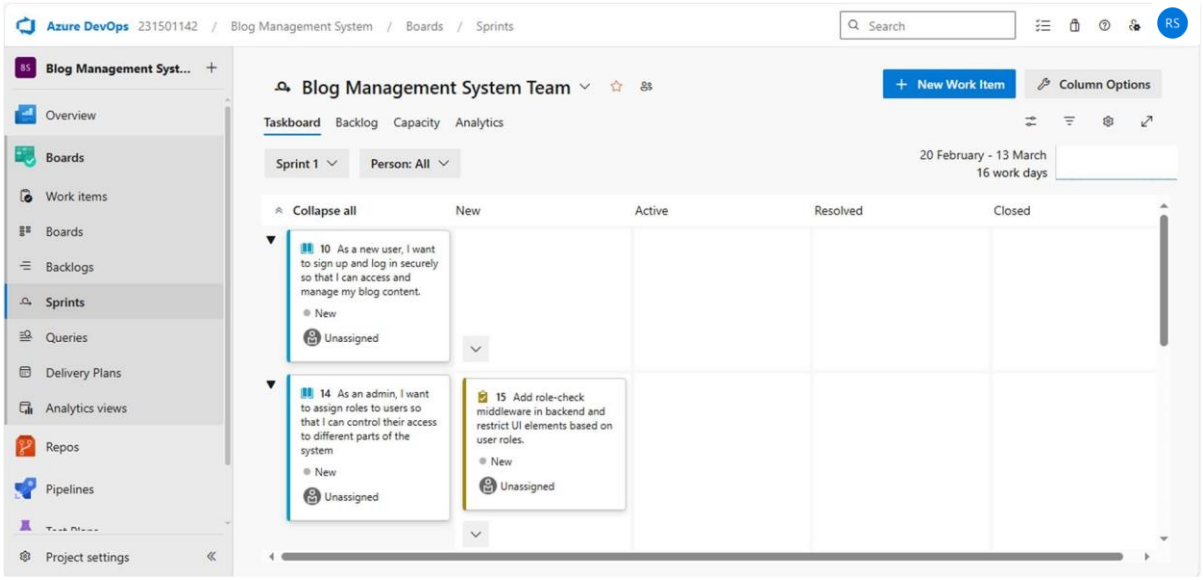
Sprint Planning Sprint 1



Sprint 2



Sprint 3



RESULT

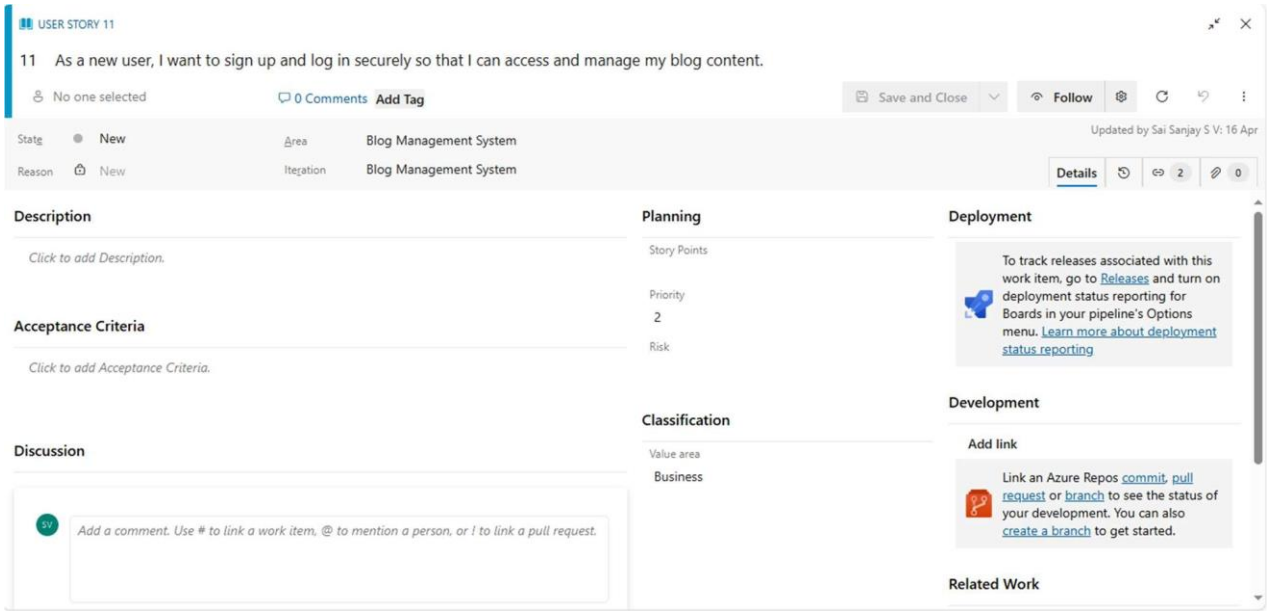
The Sprints have been created for Blog Management System.

EXP NO:5	POKER ESTIMATION

AIM

Create Poker Estimation for the user stories –Blog Management System.

Poker Estimation



RESULT

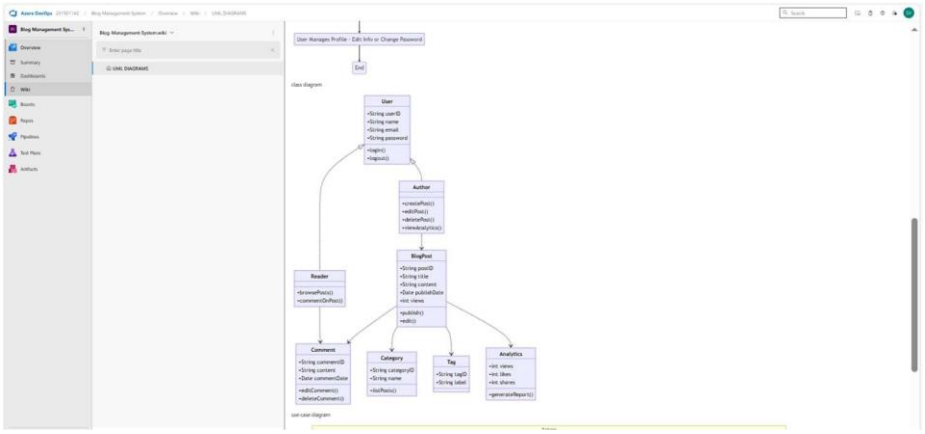
The Estimation/Story Points is created for the project using Poker Estimation.

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

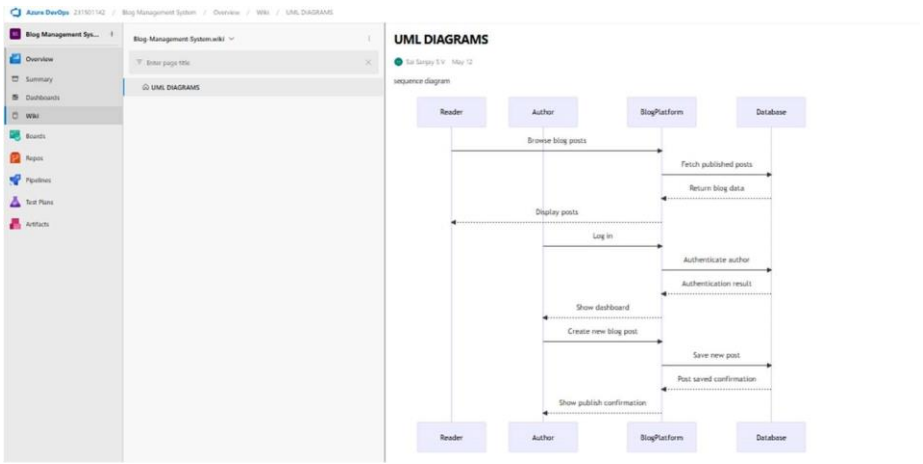
AIM

To Design a Class Diagram and Sequence Diagram for the given Project. 6A.

Class Diagram



6B. Sequence Diagram



RESULT

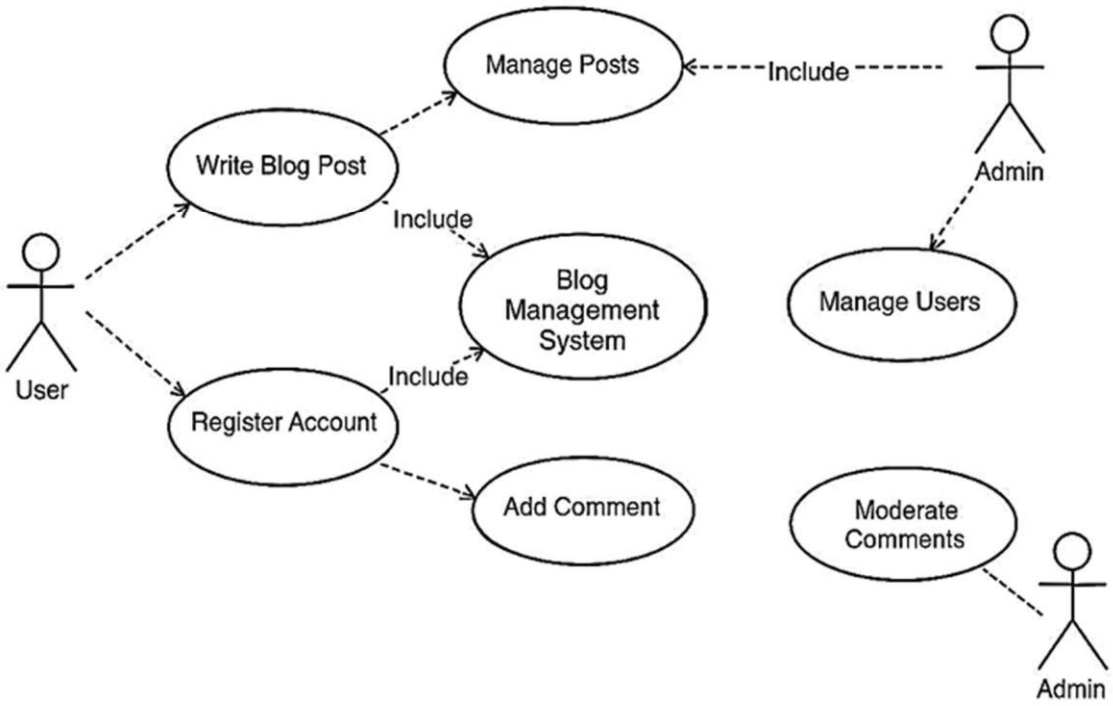
The Class Diagram and Sequence Diagram is designed Successfully for Blog Management System.

EXP NO: 7	DESIGNING USE-CASE AND ACTIVITY DIAGRAMS FOR PROJECT STRUCTURE

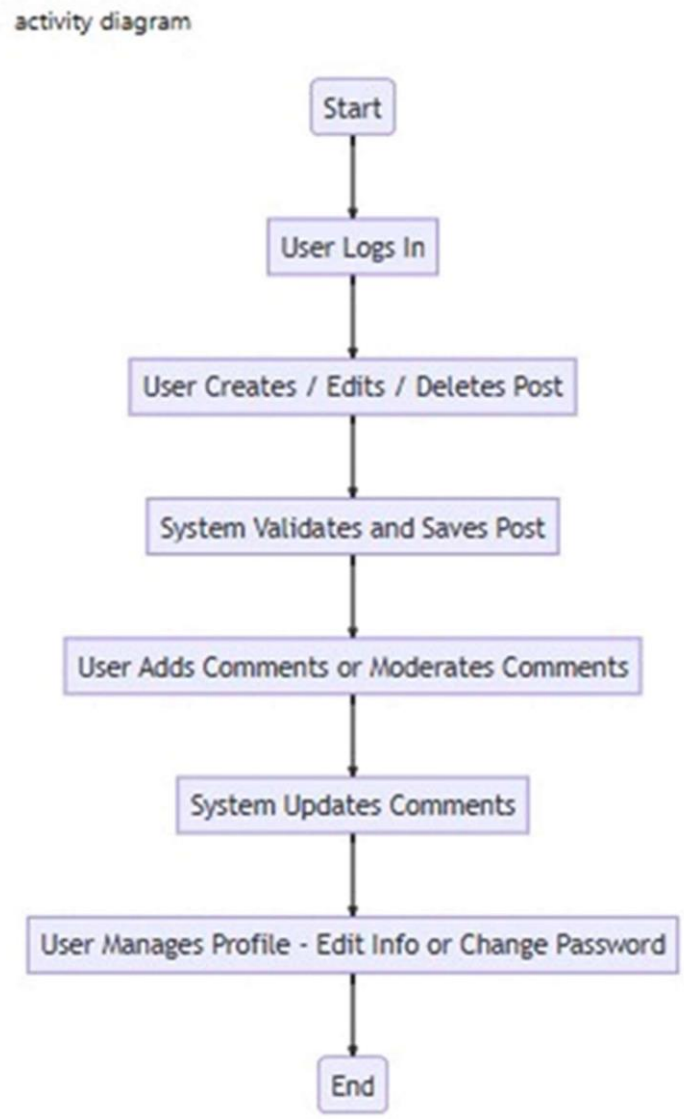
AIM

To Design an Use-Case Diagram and Activity Diagram for the given Project. 7A.

Use-Case Diagram



7B. Activity Diagram



RESULT

The Use-Case Diagram and Activity Diagram is designed Successfully for the Blog 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 Design Procedure

1. Understand Core Features of the Application

- User Login

2. Define User Interactions

- Each test case simulates a real user behaviour (e.g., logging in, submitting quizzes, viewing results)

3. Design Happy Path Test Cases

- Focused on validating that all core functionalities work correctly under normal conditions
- Example: Player registers and logs in, submits quizzes and views results

4. Design Error Path Test Cases

- Simulate invalid inputs, system issues or failed actions to ensure proper error handling.
- Example: Login with invalid credentials, submission without attachments, unauthorized access to admin panel.

5. Break Down Steps and Expected Results

- Each test case includes a clear sequence of actions and expected results. • Ensures both manual testers and automation tools can follow the process easily.

6. Use Clear Naming and IDs

- Test cases are uniquely identifies (e.g., TC01 – Valid Login, TC03 –

Invalid Password). • Facilities easy mapping to features and tracking in Azure DevOps.

7. Separate Test

• Grouped by functionality such as: o Login and Registration o Quizzes Submission o Viewing

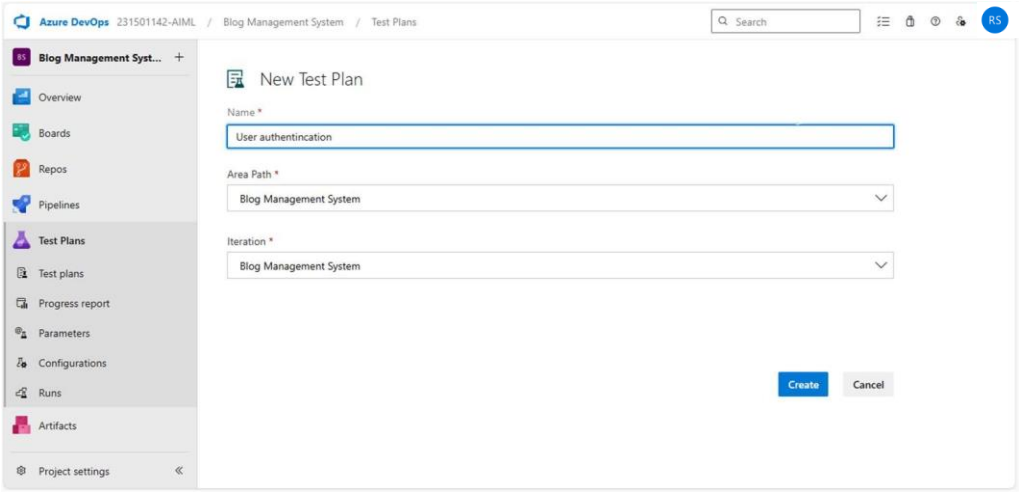
Results o Admin Functions • Improves organization and enables focused execution in Azure DevOps.

8. Prioritize and Review

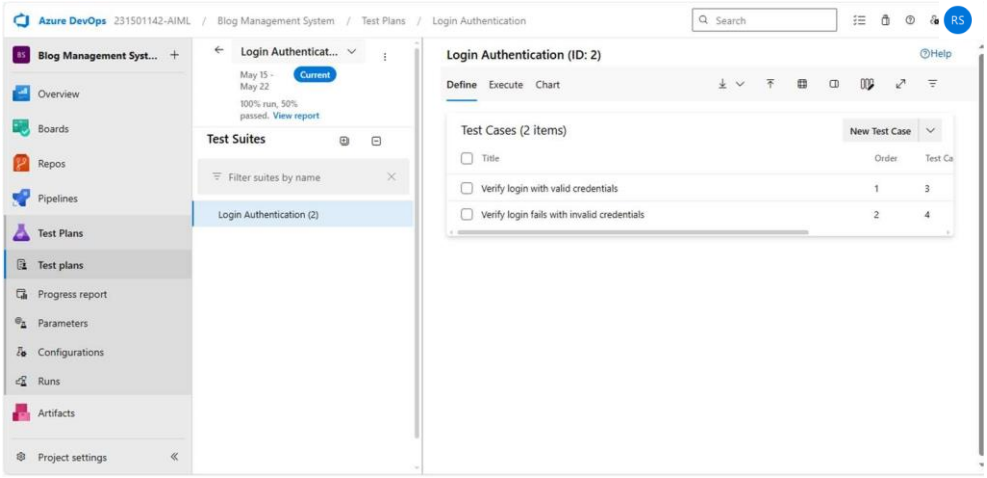
- High-priority assigned to critical workflows like login, quizzes and results.
- Reviewed for completeness, accuracy and alignment with user stories and features definition.

.

1.New test plan



2.Test suite



3.Test case

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

Blog Management System– Test Plans

USER STORIES

- As an admin, I want to log in to manage blog posts and users (ID: 3).
- As a writer, I want to register and log in so that I can create and manage my blog posts securely (ID: 4).
- As an admin, I want to assign roles and permissions to users so they can access the appropriate features (ID: 6).
- As a writer, I want to see only my posts and related analytics to manage my content efficiently (ID: 7).
- As an admin, I want to create and schedule blog posts with publication dates to control the content flow (ID: 10).

TEST SUITES

Test Suite: TS01 - Login Verification (ID: 23)

1. TC01 – Valid Login
   Action:

▪ Go to the Sign-Up page.
▪ Enter valid name, email, and password. ▪
Click “Login” button.

Expected Results:

▪ Login form is displayed.
▪ Fields accept valid credentials without error.
▪ Account is successfully logged in and redirected to the dashboard.

Type: Happy Path

2. TC02 – Invalid Username Action:

▪ Go to the Login page.
▪ Enter invalid or unregistered username. ▪
Click on "Login" button.

Expected Results:
▪ Login form is displayed.
▪ System displays an error message like “Invalid username”. ▪
User is not logged in and remains on the login page.
Type: Negative Path

3. TC03 – Invalid Password Action:

▪ Go to the Login page.
▪ Enter a valid username or email.
▪ Enter an incorrect or invalid password. ▪
Click on "Login" button.

Expected Results:
▪ Login form is displayed.
▪ System shows an error message like "Invalid username or password." ▪
User is not logged in and stays on the login page.

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 Suite: TS02 – Blog Post Creation (ID: 28)

1. TC05 – Valid Blog Post Creation
   Action:
   ▪ Navigate to the “Create Post” page.
   ▪ Enter valid blog title, content, and metadata. ▪
   Click the “Publish” or “Save Draft” button.

Expected Results:
▪ Blog post is successfully saved in the database.

Type: Happy Path

2. TC06 – Missing Title Action:
   ▪ Navigate to "Create Post" page.
   ▪ Leave the title field empty and fill in the content. ▪
   Click the “Publish” or “Save Draft” button.

Expected Results:
▪ System displays a validation error message like “Title is required”. ▪
Post is not submitted or saved in the database.

Type: Error Path

Test Suite: TS03 – Blog Analytics Viewing (ID: 32)

1. TC07 – View Analytics as Writer Action:
   ▪ Log in as a writer.
   ▪ Navigate to “My Analytics” section. ▪
   Select a blog post to view analytics.

Expected Results:
▪ Writer sees views, likes, and comment count for the selected post.

Type: Happy Path

2. TC08 – View All Analytics as Admin
   Action:
   ▪ Log in as an admin.

▪ Navigate to the “Analytics” or “All Posts Data” section. ▪
Select a post to view its performance.

Expected Results:
▪ Admin sees a list of posts with detailed analytics including author, views, likes, and comments. Type:

Happy Path

Test Suite: TS04 – Blog Publishing and Scheduling (ID: 36)

1. TC09 – Schedule Blog Post with All Fields Completed
   Action:
   ▪ Log in and navigate to the "Create Post" page.
   ▪ Fill in title, content, and metadata.
   ▪ Set a future publication date. ▪
   Click "Schedule".

Expected Results:
▪ Post is saved and scheduled to publish at the selected time.

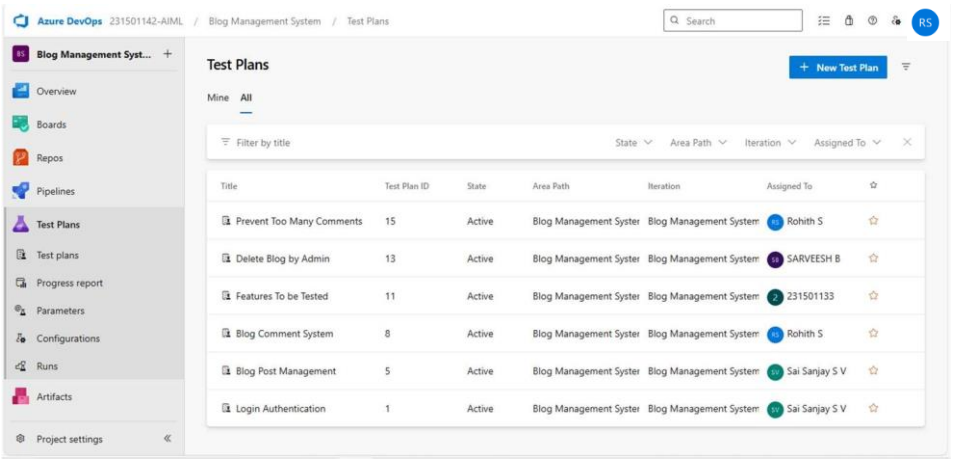
Type: Happy Path

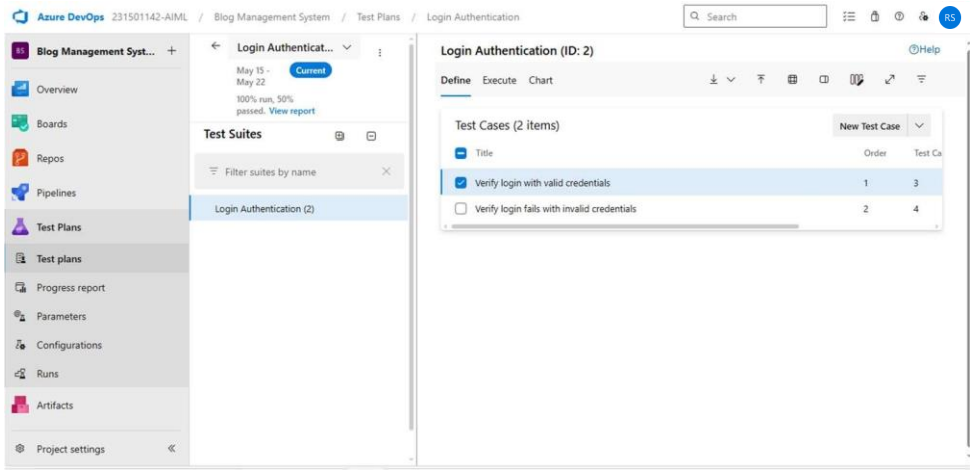
2. TC10 – Attempt to Schedule Without Content Action:
   ▪ Log in and go to "Create Post" page.
   ▪ Enter title but leave content empty. ▪
   Click "Schedule".

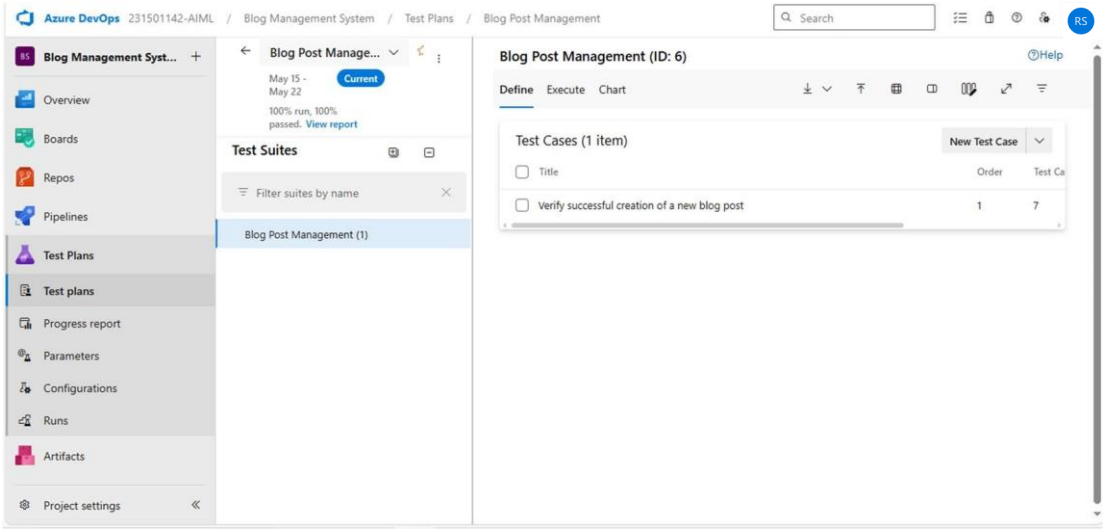
Expected Results:
▪ System displays a warning or prompt (e.g., “Content cannot be empty”).

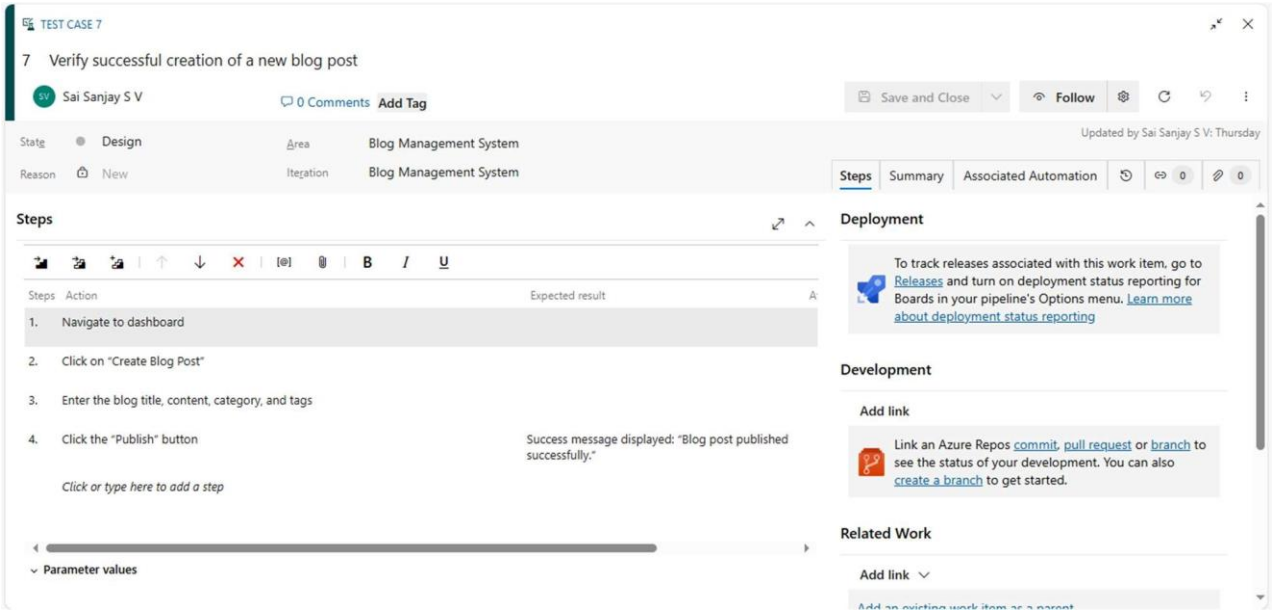
Type: Error Path

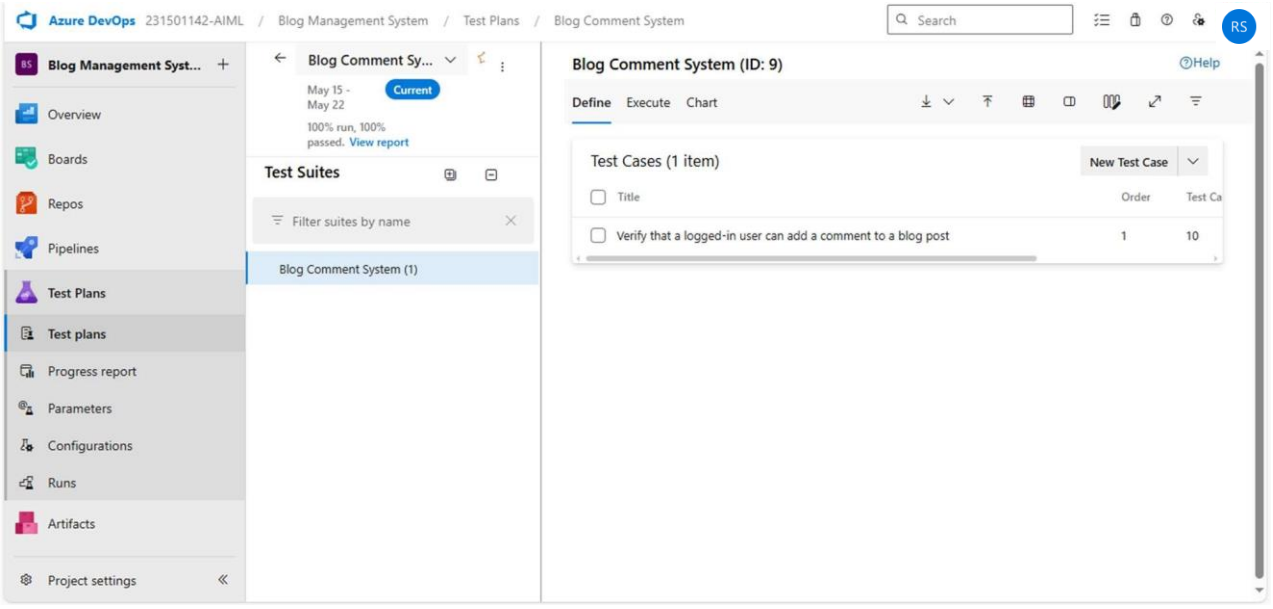
Test Cases

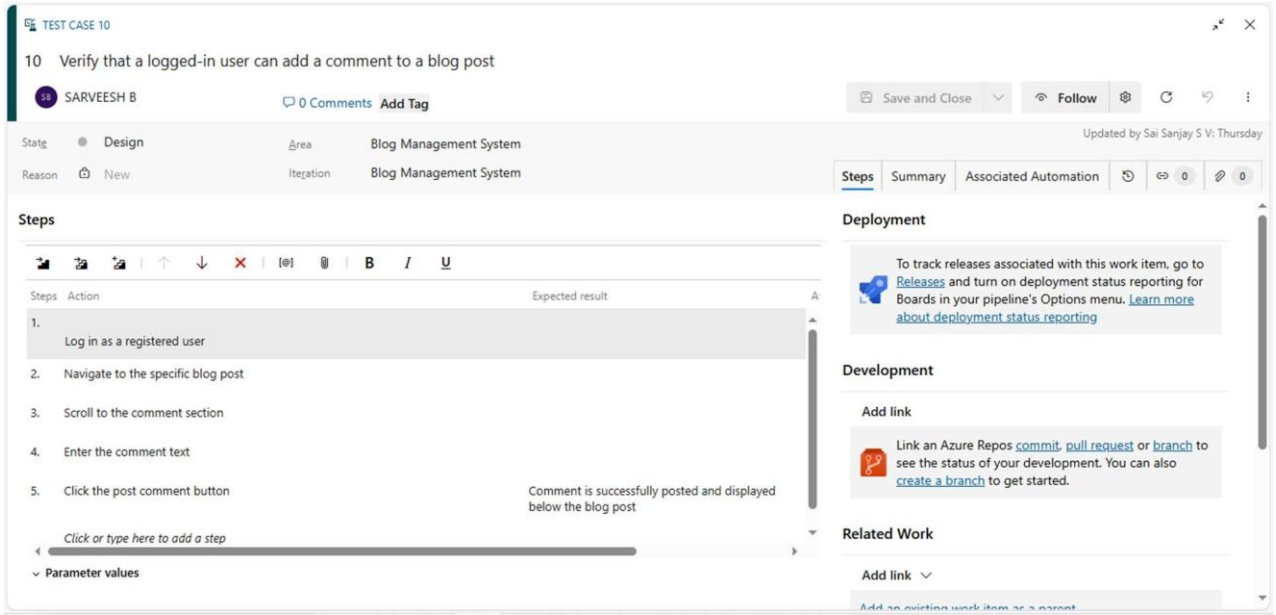

Azure DevOps 231501142-AIML / Blog Management System / Test Plans
Search
RS
Blog Management Syst...
Overview
Boards
Repos
Pipelines
Test Plans
Test plans
Progress report
Parameters
Configurations
Runs
Artifacts
Project settings
Test Plans
New Test Plan
Mine
All
Filter by title
State
Area Path
Iteration
Assigned To
Title
Test Plan ID
State
Area Path
Iteration
Assigned To
Prevent Too Many Comments 15 Active Blog Management Syster Blog Management System Rohith S
Delete Blog by Admin 13 Active Blog Management Syster Blog Management System SARVEESH B
Features To be Tested 11 Active Blog Management Syster Blog Management System 231501133
Blog Comment System 8 Active Blog Management Syster Blog Management System Rohith S
Blog Post Management 5 Active Blog Management Syster Blog Management System Sai Sanjay S V
Login Authentication 1 Active Blog Management Syster Blog Management System Sai Sanjay S V


Azure DevOps 231501142-AIML / Blog Management System / Test Plans / Login Authentication
Search
RS
Blog Management Syst...
Overview
Boards
Repos
Pipelines
Test Plans
Test plans
Progress report
Parameters
Configurations
Runs
Artifacts
Project settings
Login Authenticat...
May 15 - May 22
Current
100% run, 50% passed. View report
Test Suites
Filter suites by name
Login Authentication (2)
Login Authentication (ID: 2)
Help
Define
Execute
Chart
Test Cases (2 items)
New Test Case
Title
Order
Test Ca
Verify login with valid credentials 1 3
Verify login fails with invalid credentials 2 4

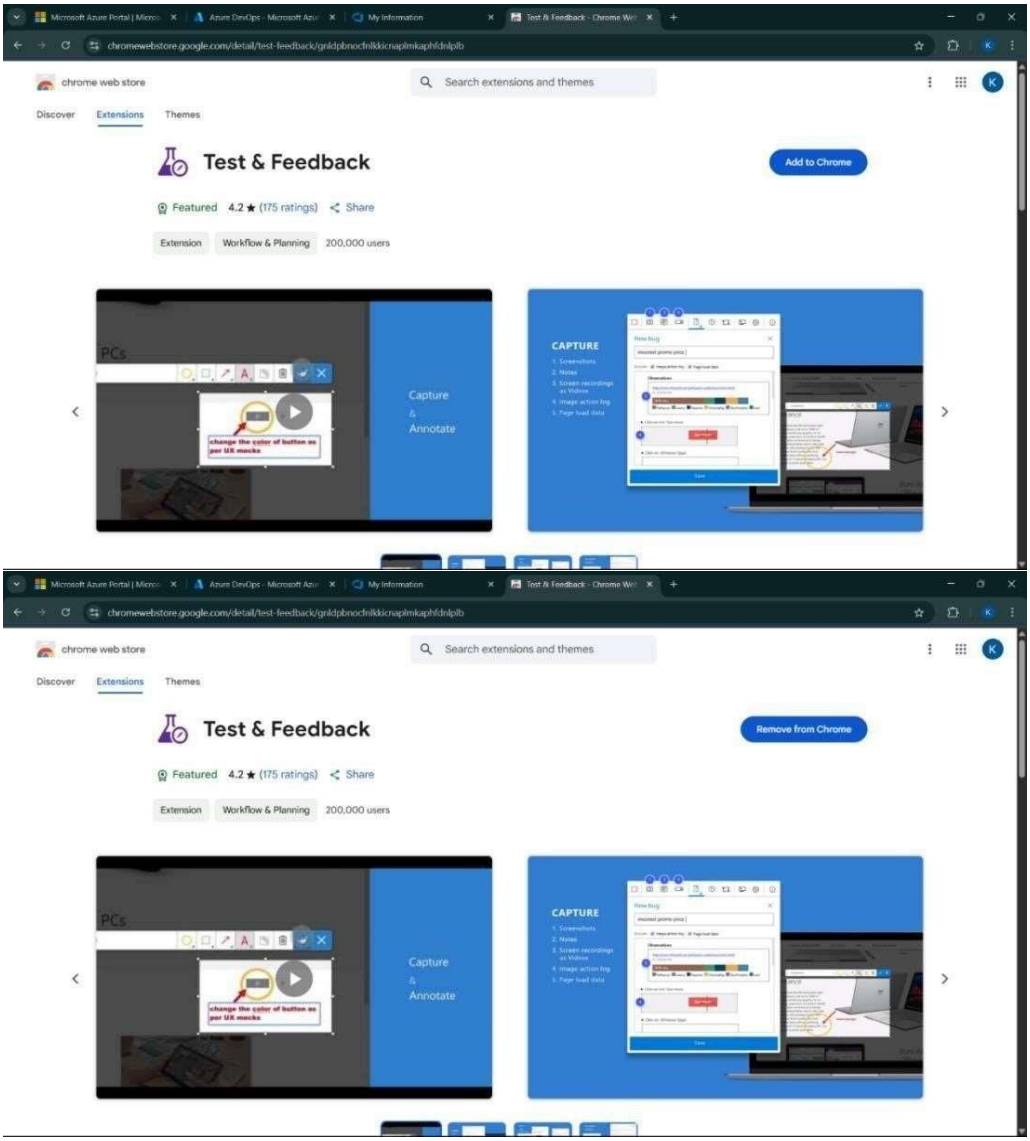

Azure DevOps 231501142-AIML / Blog Management System / Test Plans / Blog Post Management
Search
RS
Blog Management Syst...
Overview
Boards
Repos
Pipelines
Test Plans
Test plans
Progress report
Parameters
Configurations
Runs
Artifacts
Project settings
Blog Post Manage...
May 15 - May 22
Current
100% run, 100% passed. View report
Test Suites
Filter suites by name
Blog Post Management (1)
Blog Post Management (ID: 6)
Help
Define
Execute
Chart
Test Cases (1 item)
New Test Case
Title
Order
Test Ca
Verify successful creation of a new blog post 1 7


TEST CASE 7
7 Verify successful creation of a new blog post
Sai Sanjay S V
0 Comments Add Tag
Save and Close
Follow
State Design
Reason New
Area Blog Management System
Iteration Blog Management System
Updated by Sai Sanjay S V: Thursday
Steps Summary Associated Automation
Steps
Steps Action Expected result
1. Navigate to dashboard
2. Click on "Create Blog Post"
3. Enter the blog title, content, category, and tags
4. Click the "Publish" button
Success message displayed: "Blog post published successfully."
Click or type here to add a step
Parameter values
Deployment
To track releases associated with this work item, go to Releases and turn on deployment status reporting for Boards in your pipeline's Options menu. Learn more about deployment status reporting
Development
Add link
Link an Azure Repos commit, pull request or branch to see the status of your development. You can also create a branch to get started.
Related Work
Add link

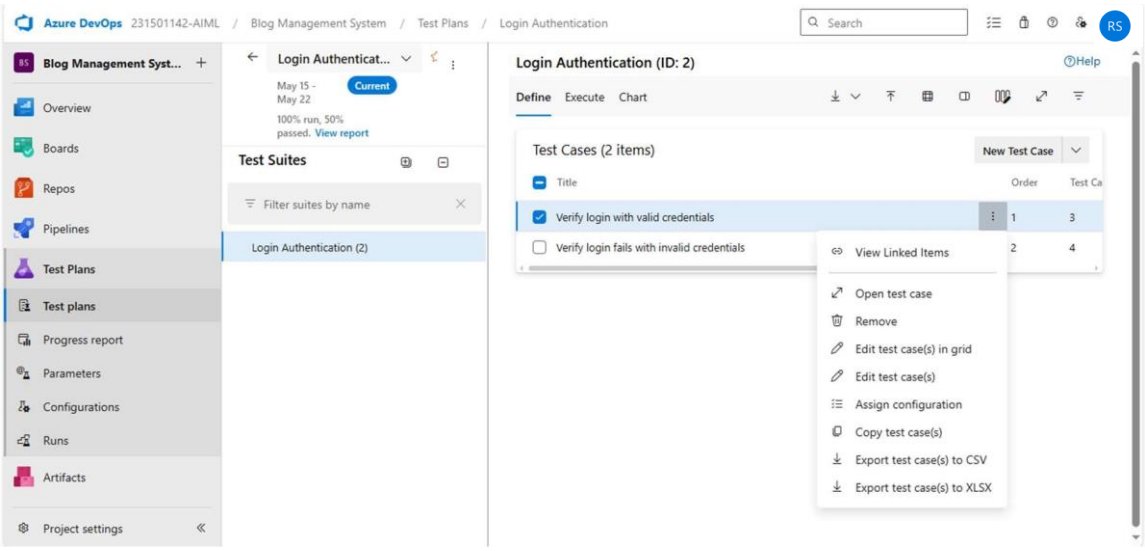

Azure DevOps 231501142-AIML / Blog Management System / Test Plans / Blog Comment System
Search
RS
Blog Management Syst...
Overview
Boards
Repos
Pipelines
Test Plans
Test plans
Progress report
Parameters
Configurations
Runs
Artifacts
Project settings
Blog Comment Sy...
May 15 - May 22
Current
100% run, 100% passed. View report
Test Suites
Filter suites by name
Blog Comment System (1)
Blog Comment System (ID: 9)
Help
Define Execute Chart
Test Cases (1 item)
New Test Case
Title
Order
Test Ca
Verify that a logged-in user can add a comment to a blog post
1
10

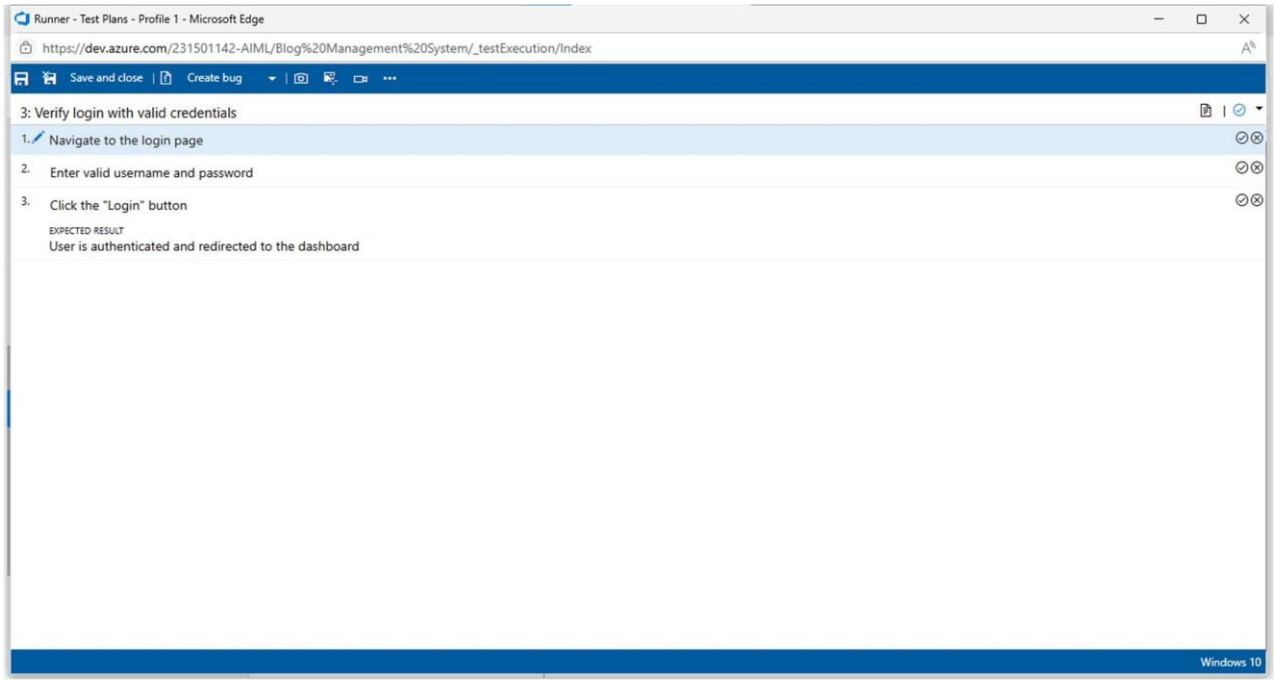


4.Installation of test

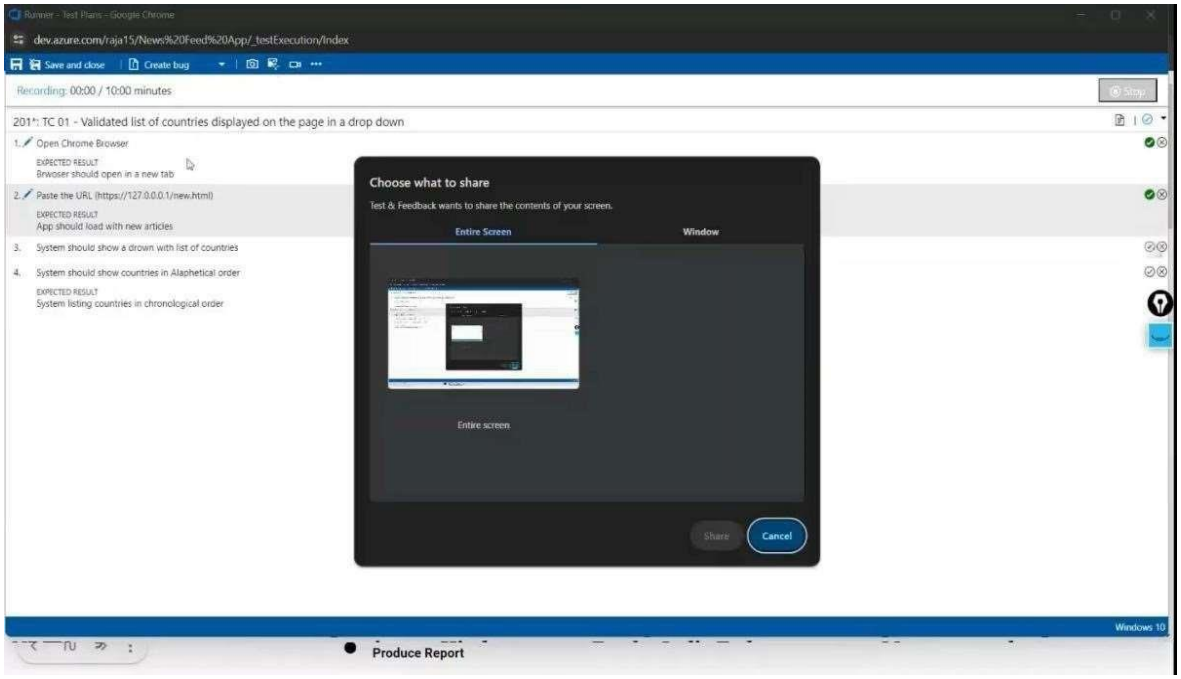


5.Running the test cases

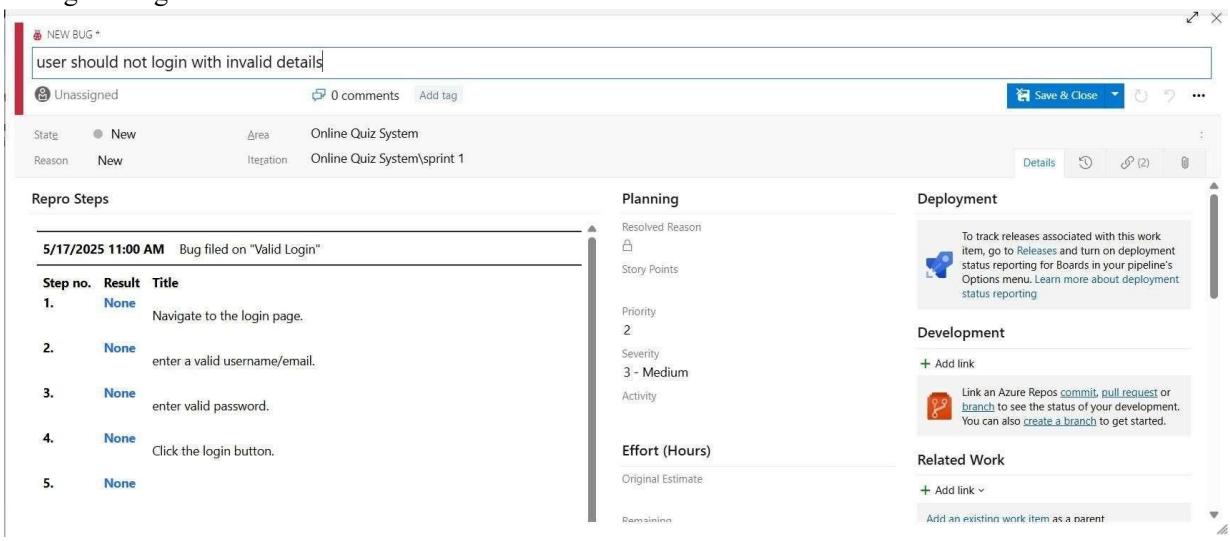


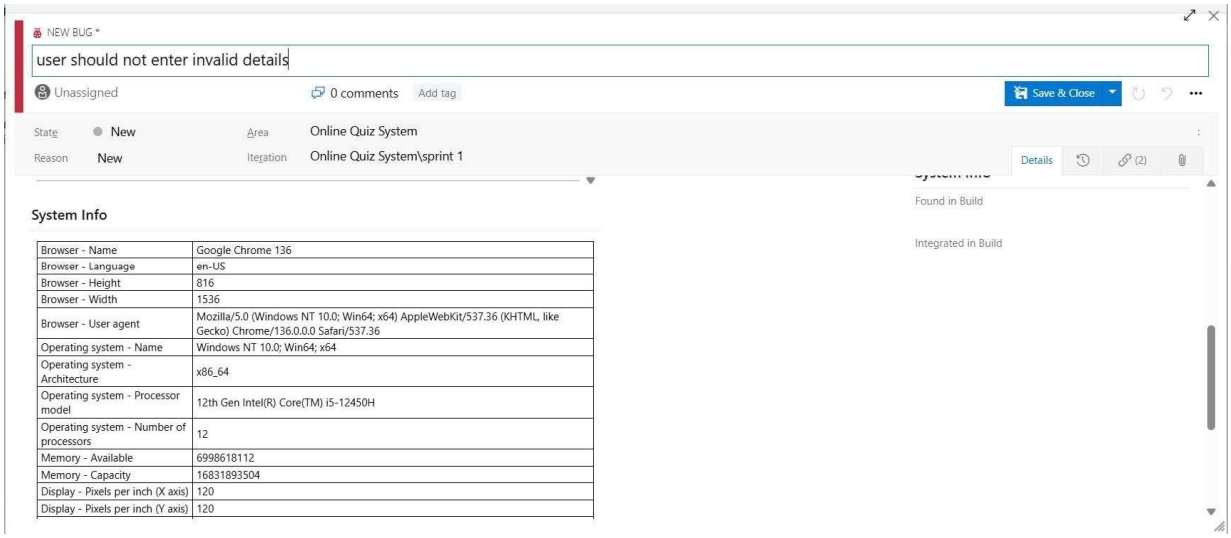


6.Recording the test case



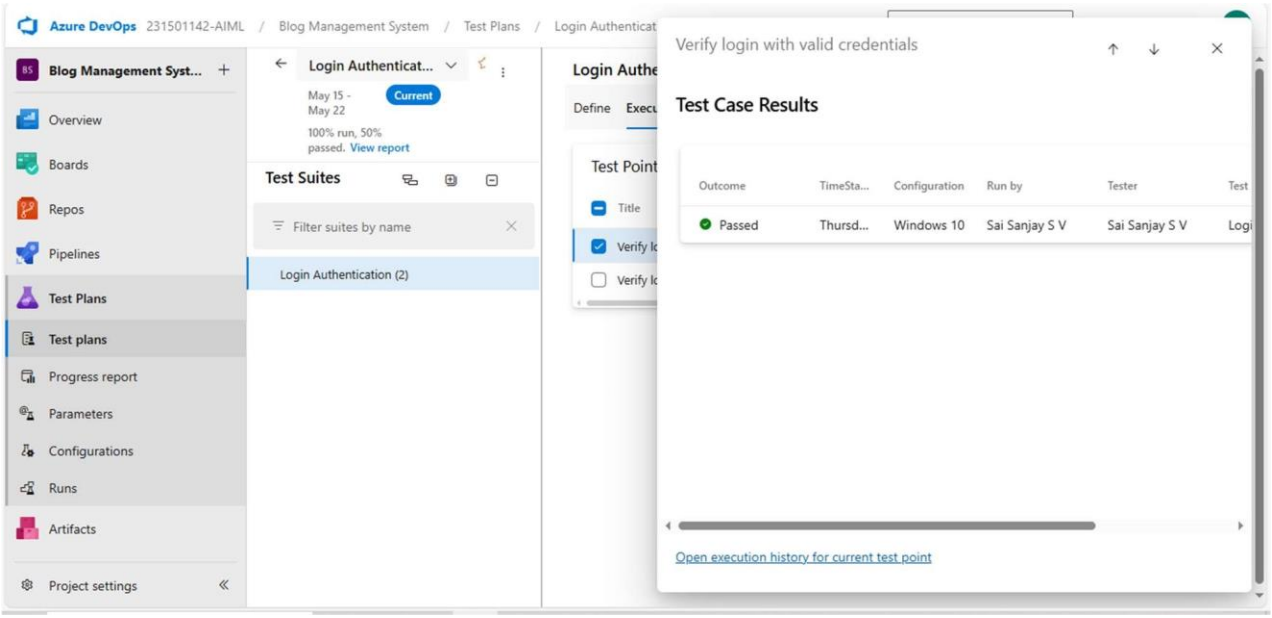
7.Creating the bug



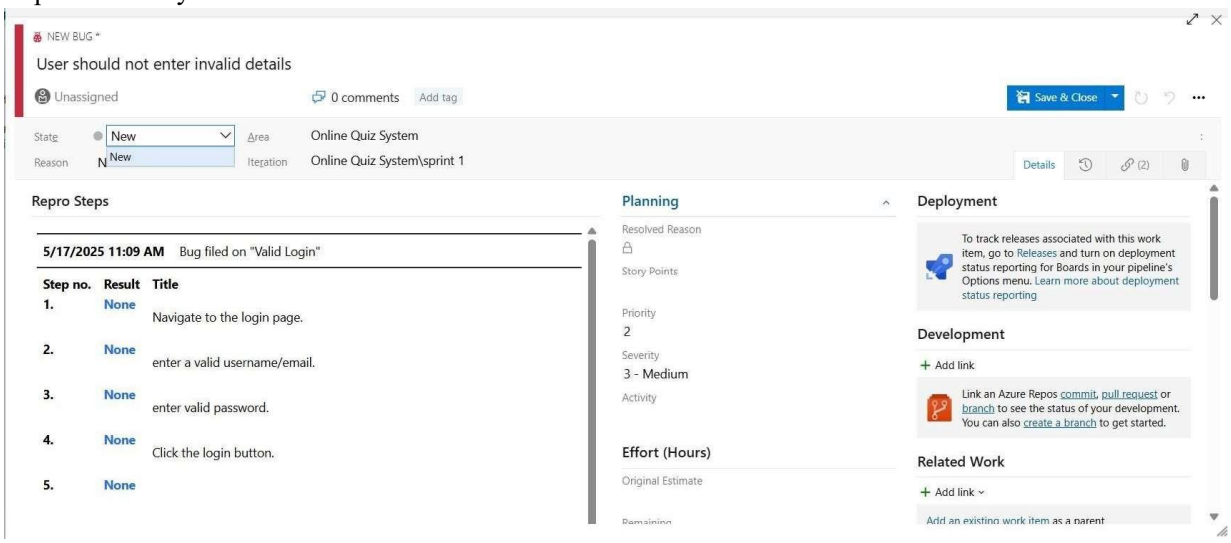


Browser - Name	Google Chrome 136
Browser - Language	en-US
Browser - Height	816
Browser - Width	1536
Browser - User agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/136.0.0.0 Safari/537.36
Operating system - Name	Windows NT 10.0; Win64; x64
Operating system - Architecture	x86_64
Operating system - Processor model	12th Gen Intel(R) Core(TM) i5-12450H
Operating system - Number of processors	12
Memory - Available	6998618112
Memory - Capacity	16831893504
Display - Pixels per inch (X axis)	120
Display - Pixels per inch (Y axis)	120

8.Test case results

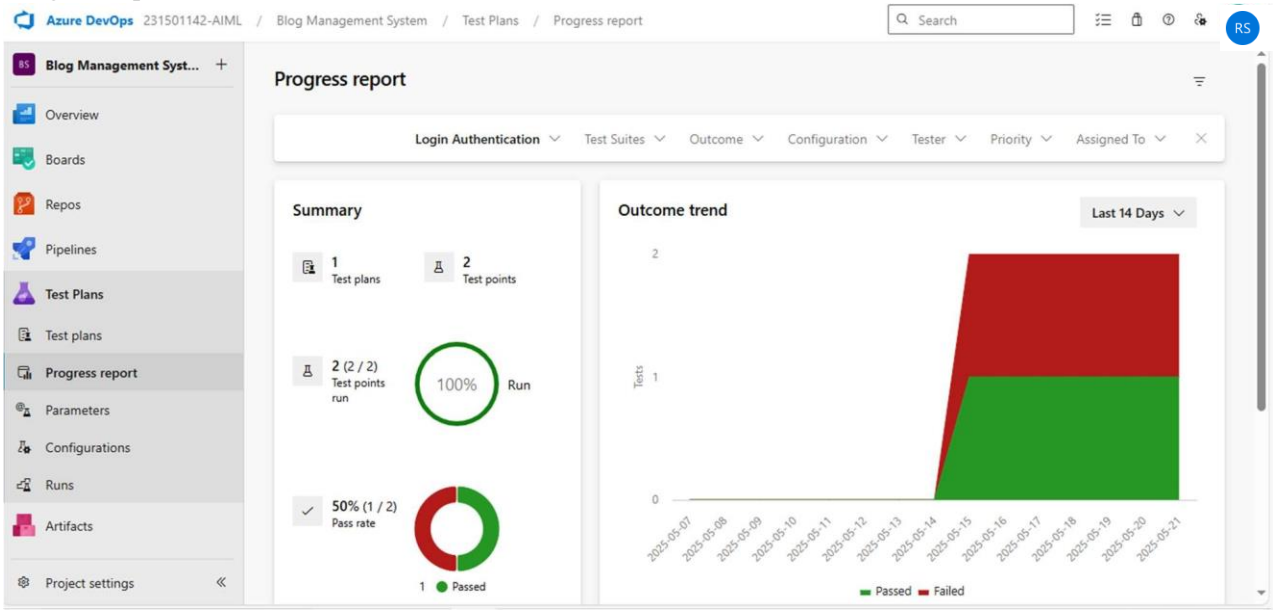


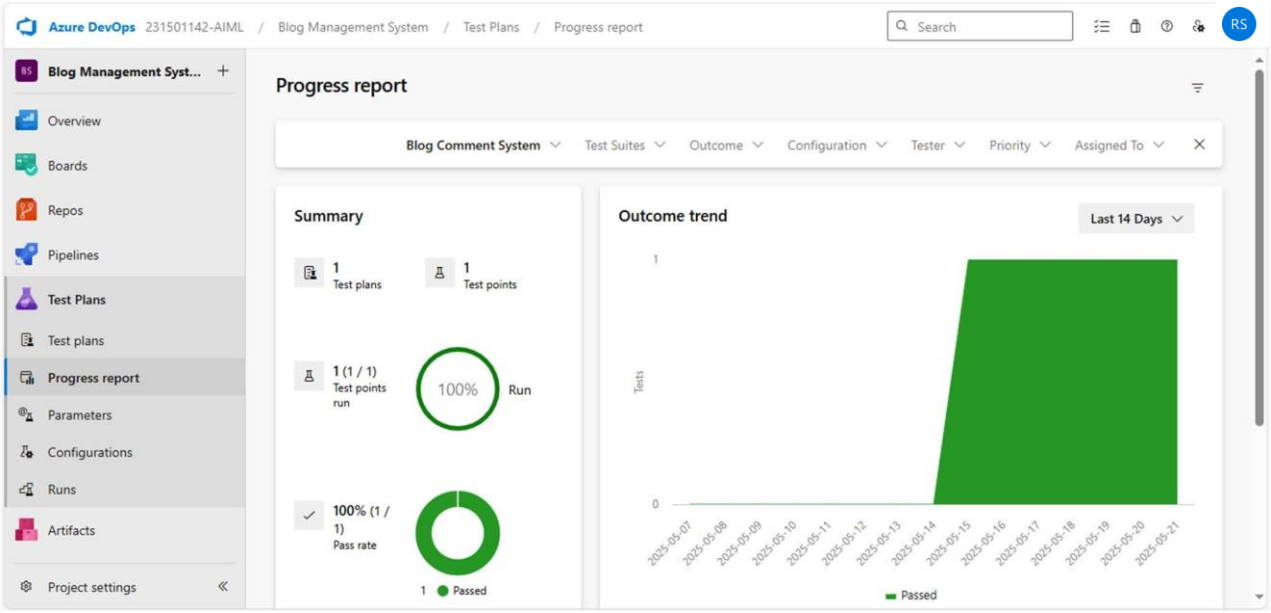
9.Test report summary

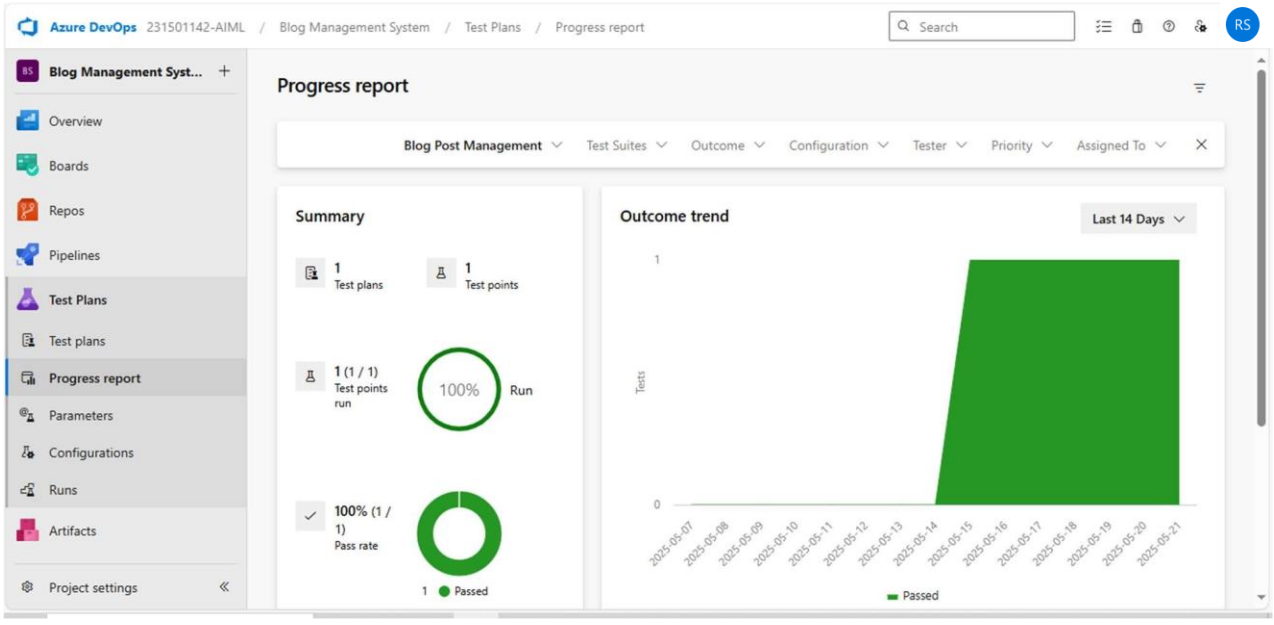


- Assigning bug to the developer and changing state

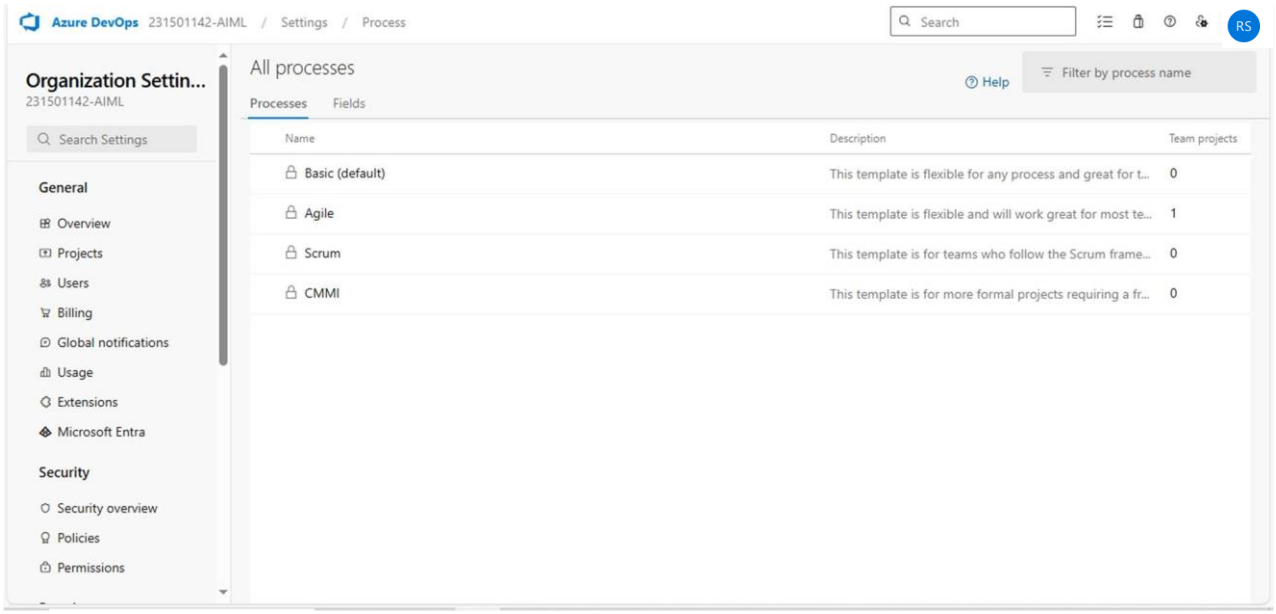
10.Progress report

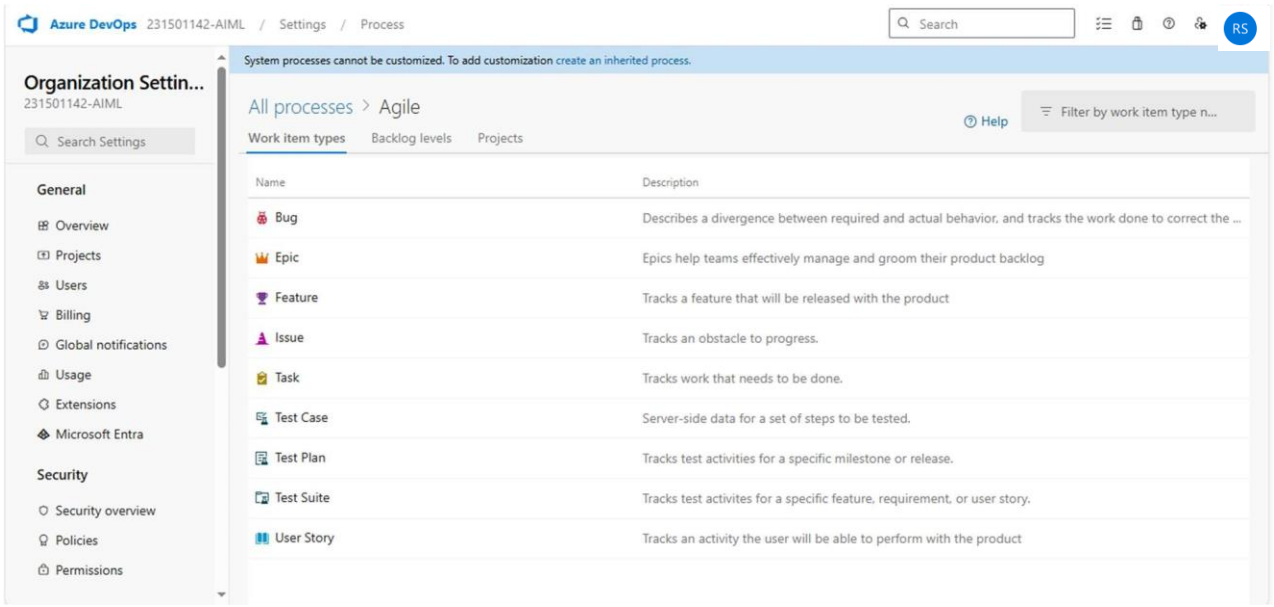






11.Changing the test template





RESULT

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

EXP NO: 9	CI/CD PIPELINES IN AZURE

AIM

To implement a Continuous Integration and Continuous Deployment (CI/CD) pipeline in Azure DevOps for automating the build, testing, and deployment process of the Blog Management System, ensuring faster delivery and improved software quality.

PROCEDURE

Steps to Create and implement pipelines in Azure:

1. Sign in to Azure DevOps and Navigate to Your Project
   Log in to dev.azure.com, select your organization, and open the project where your Student Management System code resides.

2. Connect a Code Repository (Azure Repos or GitHub)
   Ensure your application code is stored in a Git-based repository such as Azure Repos or GitHub. This will be the source for triggering builds and deployments in your pipeline.

3. Create a New Pipeline
   Go to the Pipelines section on the left panel and click “Create Pipeline”.
   Choose your source (e.g., Azure Repos Git or GitHub), and then select the repository containing your project code.

4. Choose the Pipeline Configuration
   You can select either the YAML-based pipeline (recommended for version control and automation) or the Classic Editor for a GUI-based setup. If using YAML, Azure DevOps will suggest a template or allow you to define your own.

5. Define Build Stage (CI - Continuous Integration) from YAML file.

6. Install dependencies (e.g., npm install, dotnet restore).

7. Build the application (dotnet build, npm run build).

8. Run unit tests (dotnet test, npm test).

9. Publish build artifacts to be used in the release stage.

10. Save and Run the Pipeline for the First Time
    Save the YAML or build definition and click “Run”.

    Azure will fetch the latest code and execute the defined build and test stages.

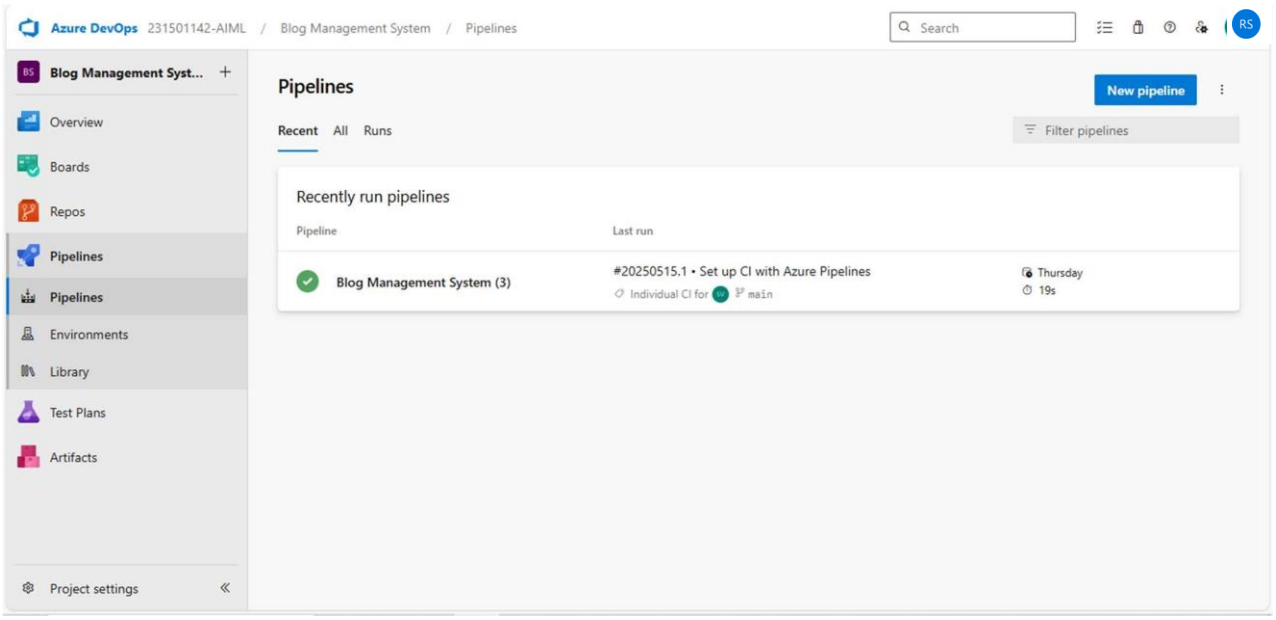
11. Configure Continuous Deployment (CD)
    Navigate to the Releases tab under Pipelines and click “New Release Pipeline”. Add an Artifact (from the build stage) and create a new Stage (e.g., Development, Production).

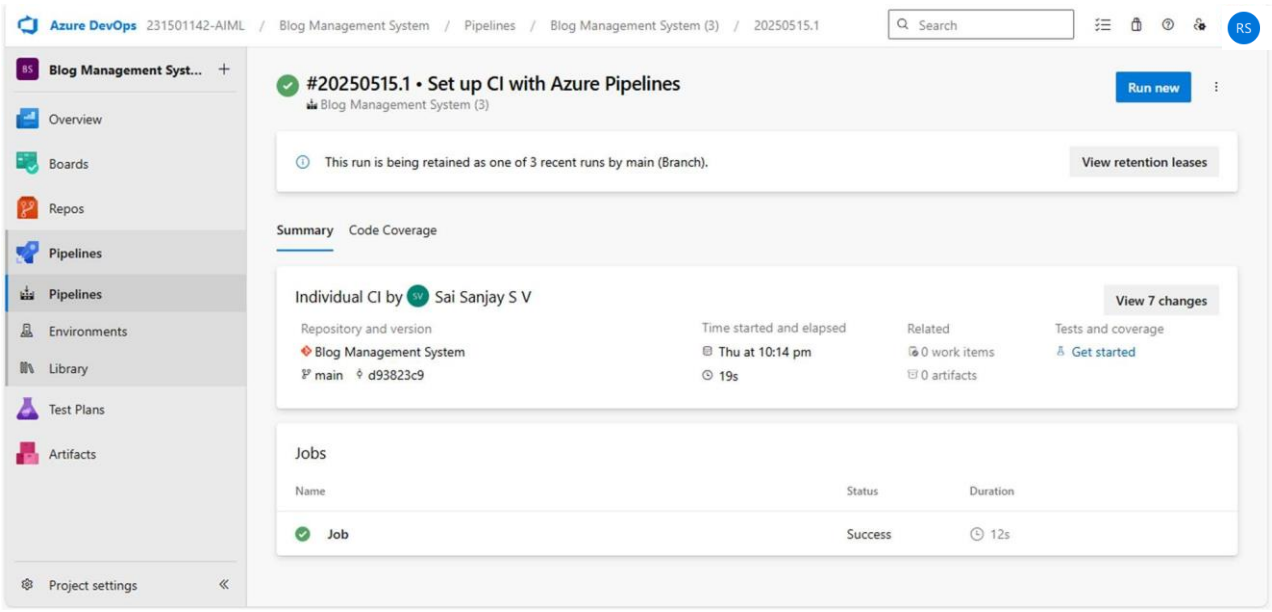
12. Configure the CD stage with deployment tasks such as deploying to Azure App Service, running database migrations or scripts, and restarting services using the Azure App Service Deploy task linked to your subscription and app details.

13. Set Triggers and Approvals
    Enable continuous deployment trigger so the release pipeline runs automatically after a successful build. For production environments, configure pre-deployment approvals to ensure manual verification before release.

14. Monitor Pipelines and Manage Logs
    View all pipeline runs under the Runs section.
    Check logs for build/test/deploy stages to debug any errors.
    You can also integrate email alerts or Microsoft Teams notifications for build failures.

15. Review and Maintain Pipelines
    Regularly update your pipeline tasks or YAML configurations as your application grows.
    Ensure pipeline runs are clean and artifacts are stored securely.
    Integrate quality gates and code coverage policies to maintain code quality.


Azure DevOps 231501142-AIML / Blog Management System / Pipelines
Search
RS
Blog Management Syst...
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Library
Test Plans
Artifacts
Project settings
Pipelines
New pipeline
Recent All Runs
Filter pipelines
Recently run pipelines
Pipeline
Last run
Blog Management System (3)
#20250515.1 • Set up CI with Azure Pipelines
Individual CI for SV main
Thursday
19s


Azure DevOps 231501142-AIML / Blog Management System / Pipelines / Blog Management System (3) / 20250515.1
Search
RS
Blog Management Syst...
Overview
Boards
Repos
Pipelines
Pipelines
Environments
Library
Test Plans
Artifacts
Project settings
#20250515.1 • Set up CI with Azure Pipelines
Blog Management System (3)
Run new
This run is being retained as one of 3 recent runs by main (Branch).
View retention leases
Summary Code Coverage
Individual CI by SV Sai Sanjay S V
View 7 changes
Repository and version
Blog Management System
main d93823c9
Time started and elapsed
Thu at 10:14 pm
19s
Related
0 work items
0 artifacts
Tests and coverage
Get started
Jobs
Name
Status
Duration
Job
Success
12s

RESULT

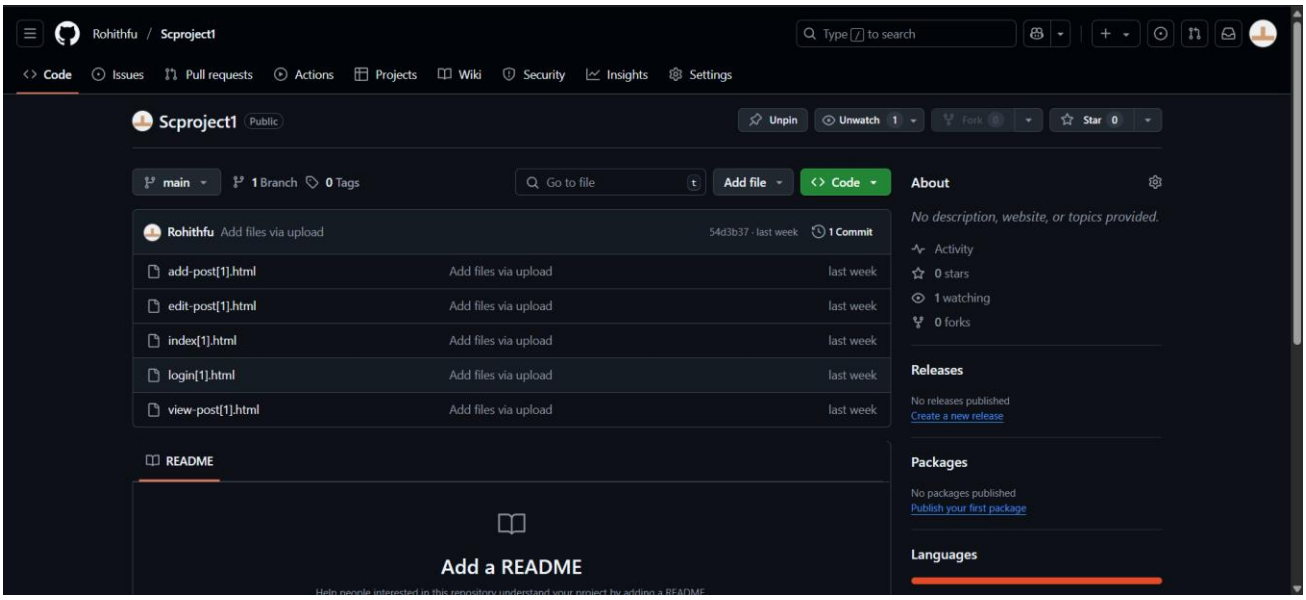
Thus, the pipelines for the given project Blog Management System has been executed successfully.

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 Blog Management System.

GitHub Project Structure



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