Azure Test Plans is a service provided by Azure DevOps that offers a set of tools for planning, tracking, and executing tests. Here's how you can get started with Azure Test Plans:

### Setting Up Azure Test Plans

1. \*Prerequisites\*

- An Azure DevOps organization.

- A project in Azure DevOps.

- Basic understanding of testing processes and test cases.

2. \*Accessing Azure Test Plans\*

- Sign in to your Azure DevOps organization.

- Select your project.

- In the left-hand menu, click on "Test Plans."

### Creating a Test Plan

1. \*Create a New Test Plan\*

- In the Test Plans page, click on "New Test Plan."

- Provide a name and description for your test plan.

- Click "Create."

2. \*Adding Test Suites\*

- Once the test plan is created, you can add test suites to organize your test cases.

- Click on "New suite" and choose the type of suite (Static, Requirement-based, or Query-based).

- Name your test suite and add it to your test plan.

### Creating Test Cases

1. \*Create a New Test Case\*

- Within a test suite, click on "New Test Case."

- Fill in the details for the test case, such as title, steps, expected results, and any relevant attachments.

- Save the test case.

2. \*Bulk Add Test Cases\*

- You can also add multiple test cases at once using the "Grid" view.

- Click on "Grid" and use the grid editor to add test cases in bulk.

### Executing Test Cases

1. \*Assign Test Cases\*

- Assign test cases to testers.

- Select the test cases you want to assign, click on "Assign tester," and choose a user from your project.

2. \*Run Tests\*

- Click on "Execute" to start running the tests.

- Choose between web-based test execution or using the Azure Test Plans extension for Visual Studio.

- Mark each step as pass/fail and provide comments as needed.

- Create bugs directly from the test execution view if issues are found.

### Analyzing Test Results

1. \*View Test Results\*

- Go to the "Runs" tab to see the results of executed test cases.

- Review the outcomes, including pass/fail rates, and any logged bugs.

2. \*Generate Reports\*

- Use built-in reporting features to generate reports on test progress and results.

- Customize the reports based on your needs.

### Continuous Testing Integration

1. \*Integrate with Azure Pipelines\*

- Use Azure Pipelines to automate the execution of your tests as part of your CI/CD process.

- Add a task to your pipeline YAML to run the tests.

Example YAML snippet:

yaml

trigger:

- main

pool:

vmImage: 'ubuntu-latest'

steps:

- task: UseDotNet@2

inputs:

packageType: 'sdk'

version: '5.x'

installationPath: $(Agent.ToolsDirectory)/dotnet

- script: |

dotnet test

displayName: 'Run Tests'

By following these steps, you can set up, create, execute, and analyze tests using Azure Test Plans. If you have any specific questions or need further details, feel free to ask.

