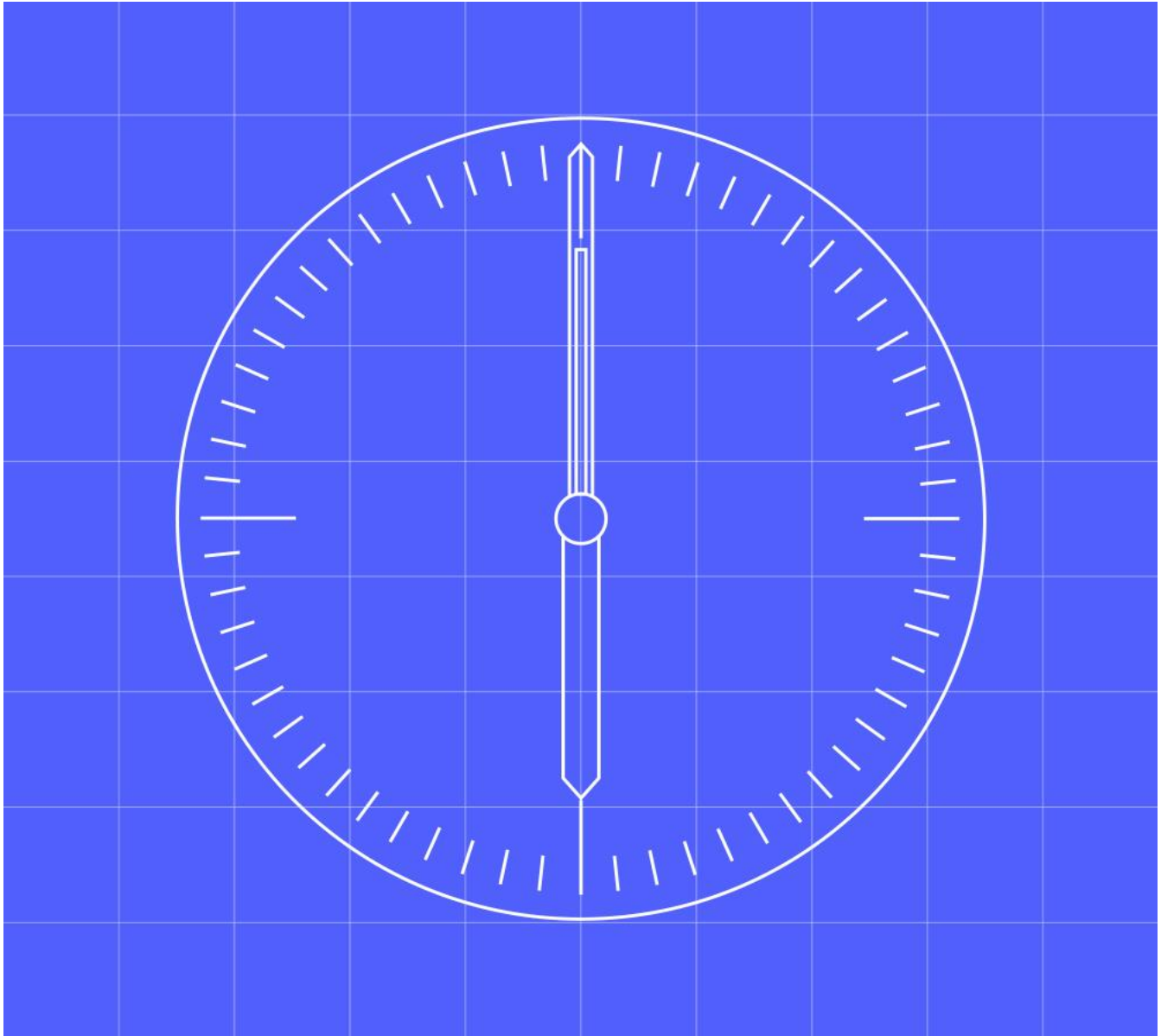


MHHS Programme

Test Management Tool User Guide



Document owner
Karthikeyan Ganesan
Status:
Final

Document number
MHHS-DEL1332
Date
02 October 2024

Version
Version 4.0
Classification
Public

1 Contents

1	Contents	1
1.1	Change Record	2
1.2	Reviewers	2
1.3	Terminology	2
1.4	Document References	2
2	Executive Summary	4
3	Introduction	5
3.1	Document Purpose	5
3.2	Security	5
3.3	User Roles	5
3.4	Reviews and Approvals	6
3.5	Summary of Changes	6
4	User: Test Manager	7
4.1	Manage Test Cases across the project	7
4.2	Manage traceability between defects and test cases	7
4.3	Consolidated view of defects and test execution on dashboard	9
5	User: Tester	10
5.1	Execute tests in the test runner	10
5.2	Raise Defects	12
5.3	Re-test when the Defects are fixed	12
5.4	Track the progress of defects	13
5.5	Report at project level on defects and test executions	13
6	User: Defect Manager	15
6.1	Analyse the Defects	15
6.2	Report on defect status	15
7	Report writer	19
7.1	Building queries and reports for the dashboard	19

1.1 Change Record

Date	Author(s)	Version	Change Detail
26/05/2023	Karthikeyan Ganesan	0.1	Initial Draft
13/06/2023	Karthikeyan Ganesan	0.2	Updates following PPC review
16/06/2023	Karthikeyan Ganesan	1.0	Updates following SRO review
05/10/2023	Karthikeyan Ganesan	1.1	Updates following ADO enhancements
06/10/2023	Karthikeyan Ganesan	2.0	Updates following SRO review
21/02/2024	Karthikeyan Ganesan	3.0	Updates following ADO enhancements
26/09/2024	Miren Sookdeo, Salman Bukhari, Karthikeyan Ganesan	3.1	Updates following changes to cohort anonymity changes., changes to sections related to defect template and raising defects. Updated reviewers list.
02/10/2024	Miren Sookdeo, Salman Bukhari, Karthikeyan Ganesan	4.0	Updates following SRO review

1.2 Reviewers

Reviewer	Role	Version
Nigel Hunt	LDP SI Test Manager	1.0
Bushra Ali	Programme Party Coordinator (PPC) Lead	1.0
Smitha Pichrikat	SRO Client Delivery Manager	1.0
Simon Berry	LDP SI Test Support Manager	2.0, 3.0
Nicola Farley	MHHS Qualifications Manager	2.0
Kiran Raj	MHHS SIT Functional Test Lead	2.0, 3.0, 4.0
Dean Bailey	LDP SI Defect Manager	2.0
Carol-Anne Smith	LDP SI Defect Manager	3.0
Sreeja Dutta	LDP SI Environment and Release Manager	3.0
Simon Berry	LDP SI Test Support Manager	3.0
Roger Robar	LDP SI Programme Test Manager	3.1, 4.0
Dominic Mooney	LDP SI SIT Manager	3.1, 4.0
Kiran Raj	SRO SIT Functional Test Lead	3.1, 4.0
Heath Thomas	SRO SIT Functional Test Lead	3.1, 4.0
Salman Bukhari	LDP SI Defect Manager	3.1, 4.0

1.3 Terminology

Term	Description
Various	For terminology, see Programme Glossary on the MHHS portal: Programme Glossary (SharePoint.com)

1.4 Document References

Document	Name
MHHS-DEL1704	MHHS ADO Training Guide #3 Raising a Defect in ADO
MHHS-DEL1705	MHHS ADO Training Guide #1 Release Management Form
MHHS-DEL1706	MHHS ADO Training Guide #1 Executing Test in ADO
MHHS-DEL1703	MHHS ADO Training Guide #2 Defect Status Overview
MHHS-DEL1707	MHHS ADO Training Guide #1 Creating Queries and Dashboards in ADO

MHHS-DEL1736	MHHS ADO Training Guide #1 Environments Downtime Form
MHHS-DEL 1701	MHHS ADO Training Guide #1 Permissions and Access Requests
MHHS-DEL1796	Defect and Release Management Lifecycle diagram

2 Executive Summary

The Market-wide Half Hourly Settlement programme (MHHS) when completed will contribute to a more cost-effective electricity system, encouraging more flexible use of energy and helping consumers lower their bills. The responsibility for success is shared between all parties and stakeholders, with everyone working together to make sure the programme is delivered and in the highest possible quality.

The purpose of this document is to provide test participants with a guide that contains screenshots for the ADO processes that will aid them in using the MHHS Programme ADO test management tool.

3 Introduction

3.1 Document Purpose

The MHHS Programme has developed this Guide to assist programme participants in using the MHHS Test Management Tool (Azure DevOps (ADO)) during testing. The ADO solution helps participants effectively manage the requirements, test cases, defects and reporting in various user roles during testing and associated activities.

3.2 Security

The Programme Test phases, have the inherent requirement to enable multiple participants to co-exist in a Test Management tool that allows the required security built in, whilst enabling key activities to be undertaken, such as:

1. Capture programme requirements and link them to test cases for traceability.
2. Test cases can be made available for execution by Programme participants in their own ADO Cohort folder.
3. Test execution can be run in isolation. It can facilitate end-to-end use cases and invite testers from different programme participants into one secure project.
4. Defects can be raised, triaged, and progressed through the development lifecycle.
5. Change requests can be progressed through the development lifecycle.
6. Project reporting showing defect states, test execution progress and all other specified reports.
7. Ensure data from one participant is not shared across another participant

MHHS Programme uses Azure DevOps (ADO) as the base for all our tooling. It was originally agreed that each cohort will operate in their own ADO project due to the Business Segregation plan but following FTIG 62 on 27th September 2024 it was agreed by all Programme Participants that anonymity will be waived from Cycle 3 onward therefore there is no longer a requirement for all Cohorts and Cohort pairings to have their own Azure DevOps project. All cohorts will use a single project, with the cohort pairings preserved within the project structure.

For point 7 above, participants have the facility to securely store sensitive information in SharePoint and link the data to the work item in ADO. Security of the storage area will be managed by Microsoft AD groups ensuring that users not within a cohort are not able to access potentially commercially confidential information. Central Parties, LDP SI teams and the SRO will have access to all data.

Users in ADO are assigned different roles with specific permissions based on their responsibilities within the project.

Please note that code repositories and release pipelines are out of scope for this Test Management Tool.

3.3 User Roles

Each participant, in their assigned projects within ADO, will have users assigned different roles with specific permissions based on their responsibilities within the project.

Note: This function is optional, and programme participants can choose whether to follow them based on roles. It is not mandatory to use all roles

The following users are included in this guide:

- Test Manager

- Tester
- Defect Manager
- Task Manager
- Report Writer

3.4 Reviews and Approvals

The document will be reviewed by the following team members:

- Heath Thomas, LDP SI Test Support Manager
- Salman Bukhari, LDP SI Defect Manager
- Carol-Anne Smith, LDP SI Defect Manager
- Sreeja Dutta, LDP SI Environment and Release Manager

Upon completion of LDP review, it will then go through a formal SRO team review by:

- Smitha Pichrikat, SRO Function Client Delivery Manager
- Nicola Farley, MHHS Qualifications Manager
- Kiran Raj, MHHS SIT Functional Test Lead
- Adrian Ackroyd, SRO Function Programme Test Manager

3.5 Summary of Changes

This document has undergone change from v3.0 to v4.0 to reflect the amended processes and additional fields required to support SIT Functional Testing. The nature of these changes is due to;

- Enhancements made to ADO processes and screens

4 User: Test Manager

Test Plans in ADO allow you to manage and track testing activities. A test plan contains one or more test suites, which are containers for grouping related test cases. Test Cases, on the other hand, are individual units of testing, describing a specific test scenario that must be performed to verify the functionality of a system or application. Test cases can also be linked to the respective requirements using the parent/child linking. Together, these three entities provide a comprehensive framework for managing tests and users can generate charts / reports for testing status and outcomes. Furthermore, these reports can be added to a dashboard to provide real-time visibility into project status and progress.

4.1 Manage Test Cases across the project

The following button notifies the person of any changes, the Test Manager makes changes to the Test Case, these changes will be live on subsequent test runs. Be aware this will impact only future test runs.

Test Case Reference

Test Case Title	Key Journey Events	Step No.	Business Process	BP Step	Traceability Requirement	Service Producing Trigger	Input/Trigger	Message Input Code
SITFTS-0130 - Smart Metered	Change of Data Service (Change Data Service only)							
	Supplier Appointment Request	1	BP003	10	MHHS-BR-SU-019 MHHS-BR-SU-020 MHHS-BR-SU-021	SUPC	F-031	[DSApp]
		2	BP003	30		DIP	PUB-031	[DSApp]
		3	BP003	40	MHHS-BR-RS-043 MHHS-BR-RS-044	DIP	PUB-031	[DSApp]

Test Case Description (1/2)

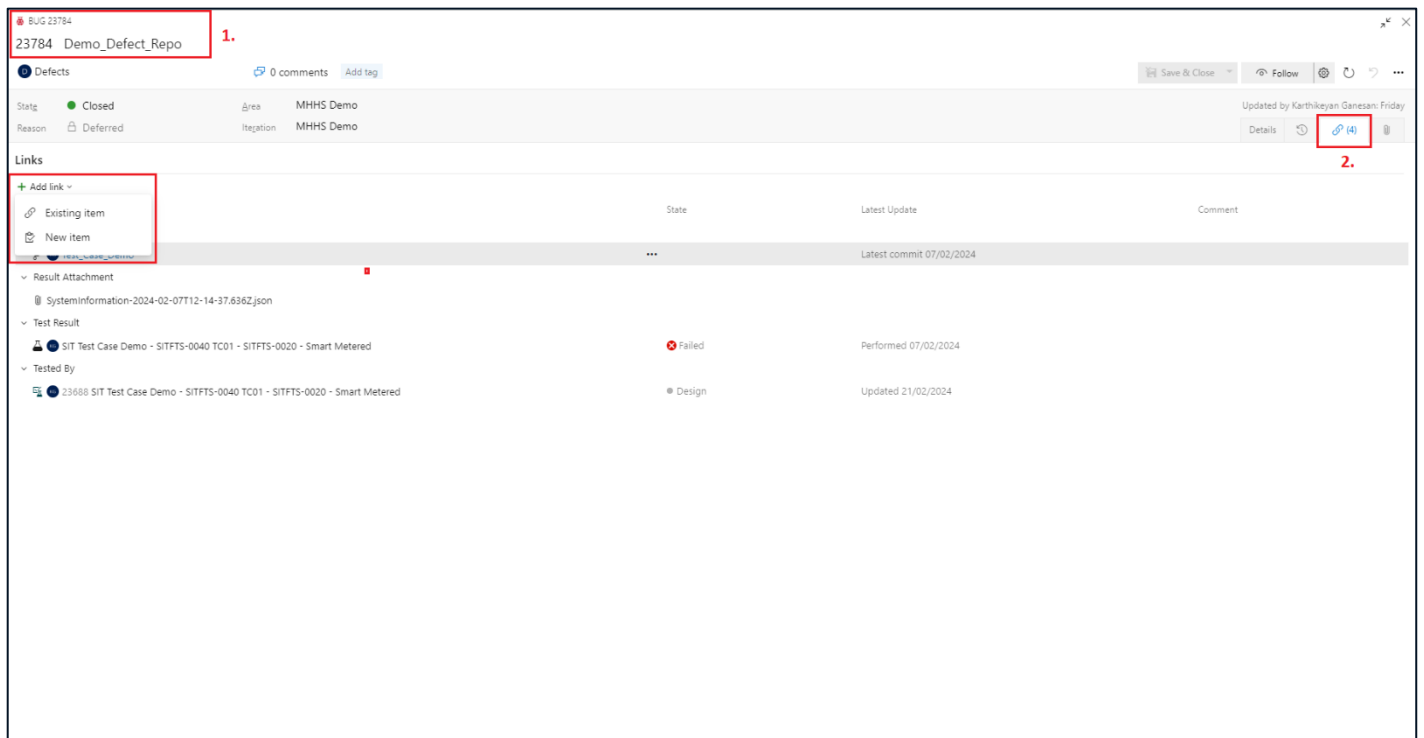
Test Case Description (2/2)

Test Preconditions

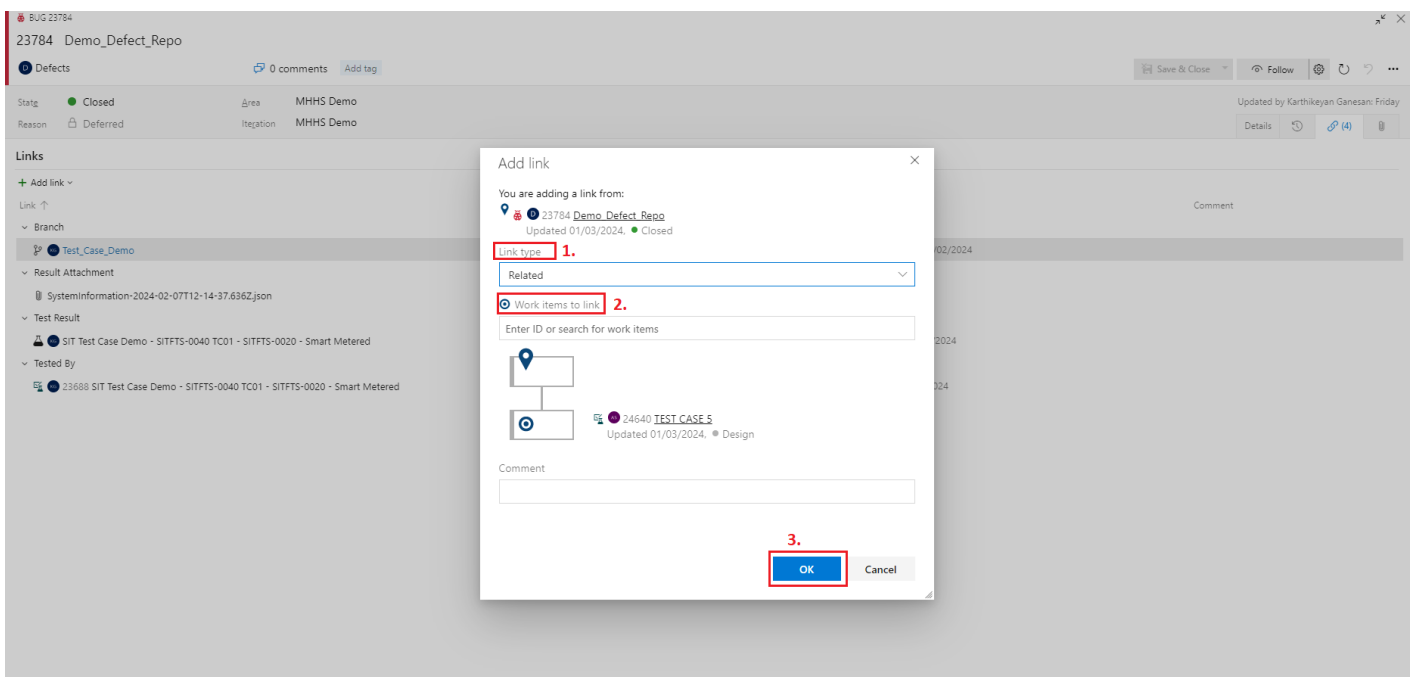
4.2 Manage traceability between defects and test cases

Follow these steps to link Defects with Test Case:

1. Open the respective Defect
2. Under Links tab, click on 'Existing item' from 'Add link'



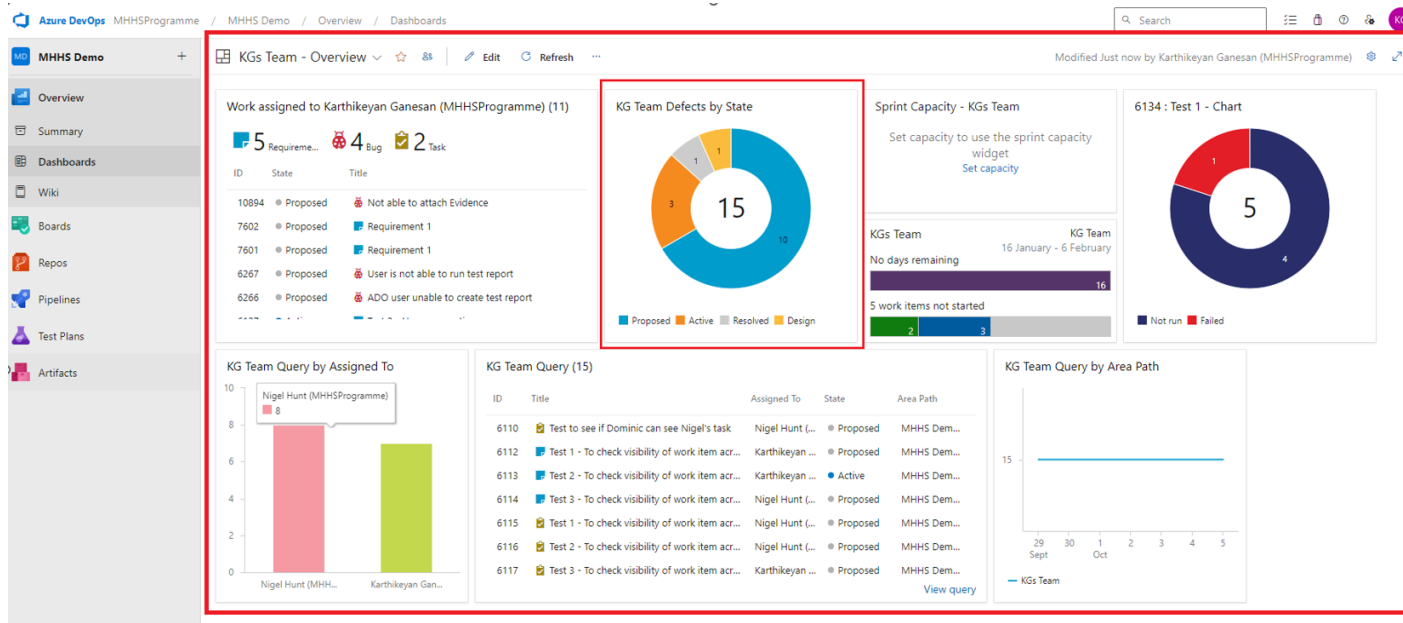
1. Choose the link type to be 'related' to link Test Case
2. Enter the Test Case name or ID to link
3. Click on OK to link the work items



4.3 Consolidated view of defects and test execution on dashboard

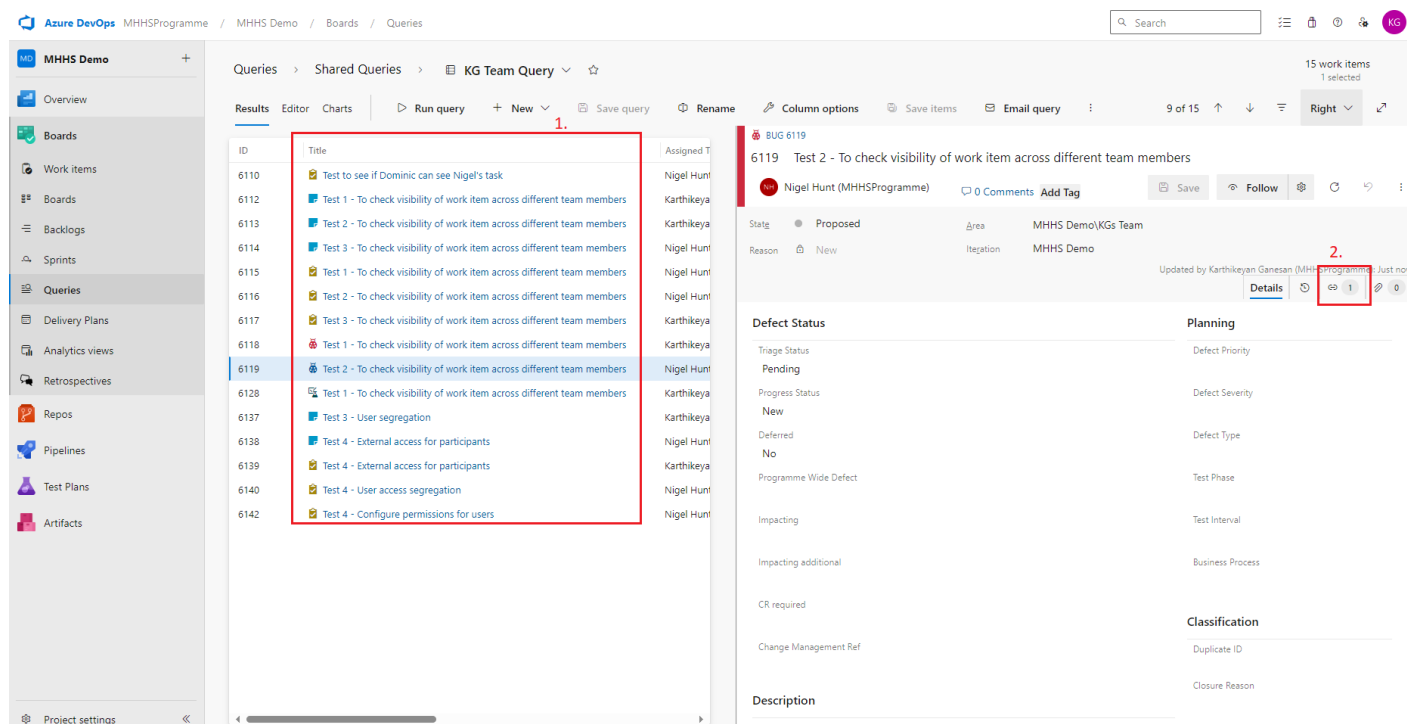
The requirements and test cases with the Defects can be visualised on the Dashboard and can be drilled down by clicking on them:

1. Choose the report and select to view the work items



The screenshot of dashboard is for illustration purpose only

2. After clicking on the chart, the drilled down version of the report pops up
3. This can also be viewed along with the related work items (requirements)



5 User: Tester

This role allows Programme Participants to run test cases utilising the Test runner feature of ADO. Test Runner allows users to execute test cases, record test results and identify any failed tests. During the testing process, for any failed test, the tester can create a Defect using the Defect feature. The Defect feature allows users to create a detailed description of the issue, including steps to reproduce the defect, screenshot, and other relevant information. Once created, the Defect is assigned to the relevant team member who can investigate the issue and resolve it. Once the Defect has been solved, it needs to be retested to ensure it has been fixed. Finally, users can generate reports about defects using the Defect Reporting feature. These reports provide insights into defect trends, such as the number of Defects found and resolved over time, as well as the severity of the defects. These Defect can also be linked to test cases using the Development field to keep track of them.

The Tester in ADO can:

- Execute test in the test runner
- Raise Defects
- Re-test when the Defects are fixed
- Track the progress of Defects
- Re-test Defects
- Report at project level on defects and test executions

5.1 Execute tests in the test runner

Follow these steps to run a test in a project:

1. Click on Test Plans
2. Click on Test Suite
3. Click on the Execute tab
4. Select the test you want to run
5. Click on Run for Web Application to execute the test

1. Test plans

2. SITF Cycle 3 - All Cohorts

3. Execute

4. Test Points (11 items)

5. Run for web application

Title	Outcome	Order	Test Case Id	Configuration	Tester
SITFTS-0050 TC01 - Smart Metered	Passed	1	38267	Windows 10	CohortA.ADS
SITFTS-0050 TC03 - Advanced	Failed	2	38269	Windows 10	CohortA.ADS
SITFTS-0050 TC02 - Traditional Metered	Passed	3	38268	Windows 10	CohortA.ADS
SITFTS-0050 TC04 - Unmetered	Passed	4	38270	Windows 10	CohortA.SUP
SITFTS-0090 TC01 - Smart Metered	Paused	5	38271	Windows 10	CohortA.SUP
SITFTS-0090 TC02 - Traditional Metered	Blocked	6	38272	Windows 10	CohortA.SUP
SITFTS-0090 TC03 - Unmetered	Active	7	38273	Windows 10	CohortA.ADS
SITFTS-0100 TC05 - Advanced Reject	Not Applicable	8	38276	Windows 10	CohortA.ADS
SITFTS-0090 TC04 - Unmetered Add CDC	Paused	9	38274	Windows 10	CohortA.ADS
SITFTS-0100 TC03 - Unmetered Add CDC	Failed	10	38275	Windows 10	CohortA.MSS
SITFTS-0135 TC01 - Smart	Blocked	11	38277	Windows 10	CohortA.ADS

6. Depending on the result, click on pass or fail for the steps
7. Add comments to the steps that have failed
8. Use the camera icon to capture any screenshot
9. Create a Defect for any steps that have failed
10. Click on Save and close the Runner window

10. Save and close

9. Create bug

8. Camera

7. COMMENT

6. Pass/Fail buttons

23688*: SIT Test Case Demo - SITFTS-0040 TC01 - SITFTS-0020 - Smart Metered

- BP001 - 60
Sender- SUPI
Receiver - CSS
Evidence - Yes
SUPI sends Switch Request to CSS
- BP001 - 65
Sender- SUPI
Receiver - CSS
Evidence - Yes
CSS receives Switch Request from Incoming Supplier and responds to REGS
- BP001 - 70
Sender- CSS
Receiver - REGS
Evidence - Yes
CSS sends CSS02800 (pending Supplier request via CSS flow) to the REGS.

5.2 Raise Defects

Follow these steps to create a Defect:

1. Enter a title for the Defect
2. Complete all the required fields, including MPAN reference if appropriate, step where issue occurred etc.
3. Attach relevant evidence e.g including but not limited to payloads, asynchronous / synchronous messages, Portal routing screenshots
4. Click on Save and Close to create the Defect

1. Enter title

4. Save & Close

2.

5.3 Re-test when the Defects are fixed

When the defect needs re-testing, the state changes as follows from active to resolved:

1. Status dropdown

2. Save button

5.4 Track the progress of defects

Follow these steps to keep track of any changes in Defect

1. Click the respective Defect
2. Click on Follow to receive all notifications for the Defect
3. Click on the setting to pick the type of notification on the Defect
4. Select the desired notification setting for the Defect
5. Click on OK to start receiving notifications

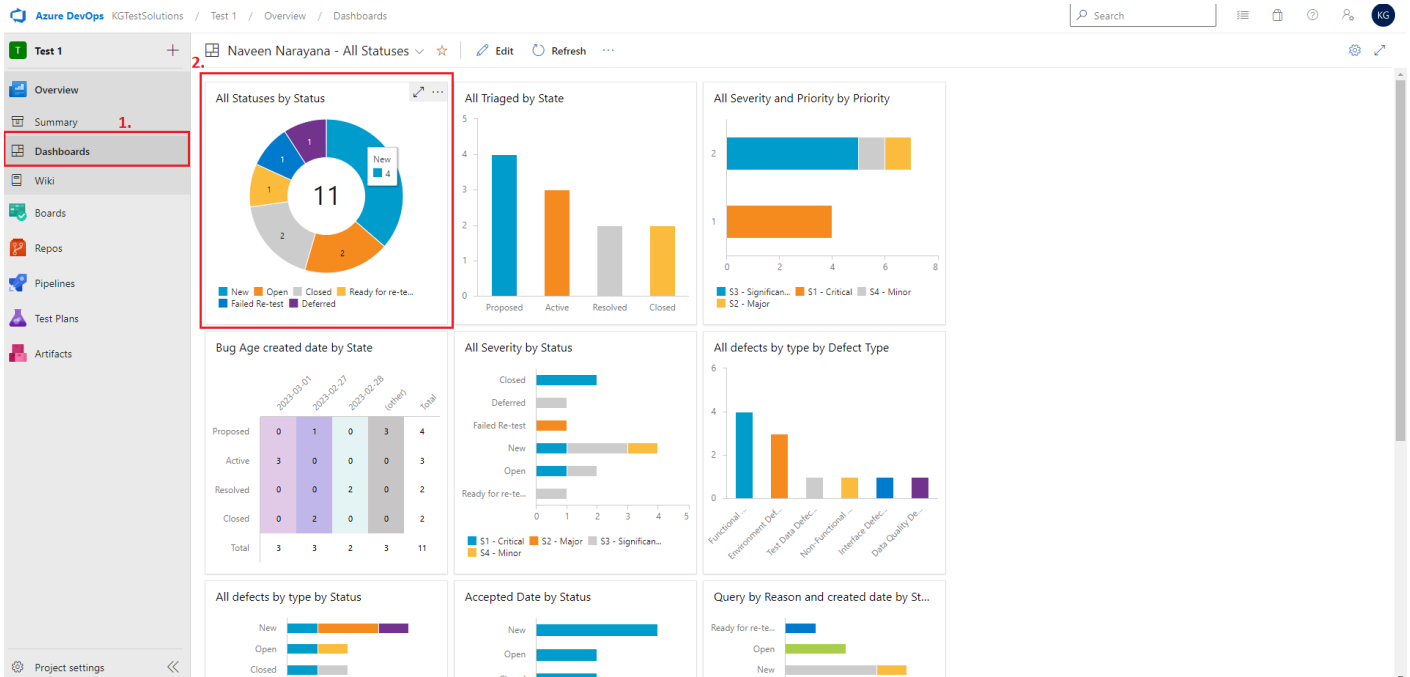
The screenshot displays a defect tracking system interface. At the top, a defect titled 'BUG 24641*' is shown with the number '24641' and the word 'demmo'. Below this, there are tabs for 'Defects', '0 comments', and 'Add tag'. On the right side, there are buttons for 'Save', 'Follow', and a settings gear icon. The 'Follow' button is highlighted with a red box and labeled '2.'. The settings gear icon is highlighted with a red box and labeled '3.'. A 'Notification Settings' dialog box is open in the center, with a red box around it labeled '4.'. The dialog box has three options: 'Not Subscribed' (selected), 'Subscribed', and 'Custom'. The 'Not Subscribed' option is described as 'Only receive notifications from this work item when you have been @mentioned.' The 'Subscribed' option is described as 'Receive all notifications from this work item.' The 'Custom' option is described as 'You will only be notified for the events selected from the list below or when you are @mentioned in this work item.' Below the 'Custom' option, there are three checkboxes: 'State Changed', 'Assigned To Changed', and 'Iteration Changed'. At the bottom of the dialog box, there are 'OK' and 'Cancel' buttons. The 'OK' button is highlighted with a red box and labeled '5.'. The background interface shows various sections: 'Defect Status' (Created By: Karthikeyan Ganesan, Assigned to Group: Karthikeyan Ganesan, Resolver Group, Triage Status, Resolver received? False, Progress Status: New, Deferred, No, Impact Notes), 'References' (Design Document Ref, Swagger Version, Programme Subject Matter Expert), 'Environment' (Defect Found Environment: SIT A, Found in Build Version: 2), 'Fix Release' (Fix ETA, Expected Release Date, Expected Release Number, Actual Release Date, Release Version), and 'Planning' (Cohort: SITF Cohort B, Participant Organisation Name: Edgware, Test Phase: SIT - Migration, Test Cycle).

5.5 Report at project level on defects and test executions

Follow these steps to drill down from consolidated view:

1. Click on the dashboard and go under the respective dashboard

2. Click on the desired chart to view in detail



Note: The screenshot of dashboard is for illustration purpose only

Azure DevOps KGTestSolutions / Test 1 / Boards / Queries

Queries > Shared Queries > All Statuses

Filter by keyword

ID	Work Item...	Title	Assigned To	State	Tags	Status	Created Date
47	Bug	Test 4 - Naveen mic issue	Naveen Narayana	Resolved		Ready for...	28/02/2023 10:42
50	Bug	Test case 3 - Naveen login field issue	Naveen Narayana	Active		Open	01/03/2023 10:21
51	Bug	Test case 5 user not able to click download button	Naveen Narayana	Active		Open	01/03/2023 10:42
37	Bug	pop-up page not visible	Darren Pyefinch	Propos...		New	24/02/2023 13:16
42	Bug	Not able to view items	Karthikeyan Ganes...	Propos...		New	24/02/2023 14:28
46	Bug	Test 3 Naveen mic issue	Naveen Narayana	Propos...		New	27/02/2023 17:29
53	Bug	Unable to select the drop down	Naveen Narayana	Propos...		New	13/03/2023 16:54
49	Bug	Test case 2 - Naveen login issue	Naveen Narayana	Active		Failed Re...	01/03/2023 10:08
48	Bug	Test 5 - Naveen mic issue	Naveen Narayana	Resolved		Deferred	28/02/2023 10:42
44	Bug	Test 1 Naveen mic issue	Naveen Narayana	Closed		Closed	27/02/2023 14:12
45	Bug	Test 2 Naveen mic issue	Naveen Narayana	Closed		Closed	27/02/2023 17:17

11 work items
1 selected

BUG 47
47 Test 4 - Naveen mic issue
Naveen Naraya... 0 comments

State: Resolved
Reason: Fixed
Iteration: Iteration

Repro Steps
Click to add Repro Steps

Symptom
Click to add Symptom

System Info
Click to add System Info

Fix
Click to add Proposed Fix

Planning
Priority: 2
Severity: S3 - Significant

Dev
+ A

6 User: Defect Manager

The Programme Defect Manager in ADO raises and manages Defects that have been identified during different Test Phases.

A Defect Manager can:

- Analyse the defects
- Report on defect status

6.1 Analyse the Defects

The Defect Manager ensures that the fields are kept up to date.

The screenshot displays the ADO Defect Manager form. At the top, there's a 'NEW BUG' alert and a 'Field Title cannot be empty' error message. Below this is a 'Enter title' input field. The form is divided into several sections: 'Defects' (with '0 comments' and 'Add tag'), 'Status' (with 'Proposed' and 'New' tabs), 'Area' (with 'MHHS Demo' and 'MHHS Demo' tabs), and 'Reason' (with 'MHHS Demo' and 'MHHS Demo' tabs). The main form area is divided into three columns: 'Theme', 'Planning', and 'Ownership'. The 'Theme' column includes fields for 'Defect Status' (Created By, Changed Date, Assigned to Group, Resolver Group, Triage Status, Progress Status, Deferred, Programme Wide Defect, Impacting, Impacting additional, CR required, Change Management Ref) and 'Description'. The 'Planning' column includes fields for 'Defect Priority', 'Defect Severity', 'Defect Type', 'Test Phase', 'Test Interval', 'Business Process', 'Classification' (IF, PUB, IF / PUB Messages), 'Duplicate ID', 'Closure Reason', 'Defect Root Cause', 'Rejected Reason', and 'Mapped on SpecMap'. The 'Ownership' column includes fields for 'System or Service', 'Programme Subject Matter Expert', 'Deployment' (with a link to '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'), 'Environment' (Cohort, Participant Organisation Name, Defect Found Environment, Market Role, Found in Build Version), and 'Build' (Fix ETA, Expected Release Date, Expected Release Number, Actual Release Date, Release Version). At the bottom, there are sections for 'Repro Steps' and 'Cohorts Impacted'.

6.2 Report on defect status

Follow these steps to generate reports:

1. Click on Work Items under overview
2. Filter the defects to the build the report

3. Click on Open filtered view in Queries

The screenshot shows the Azure DevOps interface for the 'Test 1' project. The left sidebar has 'Work items' highlighted with a red box and the number '1.'. The main area shows a list of work items. At the top of this list, there is a link 'Open filtered view in Queries' with a red box and the number '3.'. Above the list, there is a 'Bug' dropdown menu with a red box and the number '2.'. The table below contains the following data:

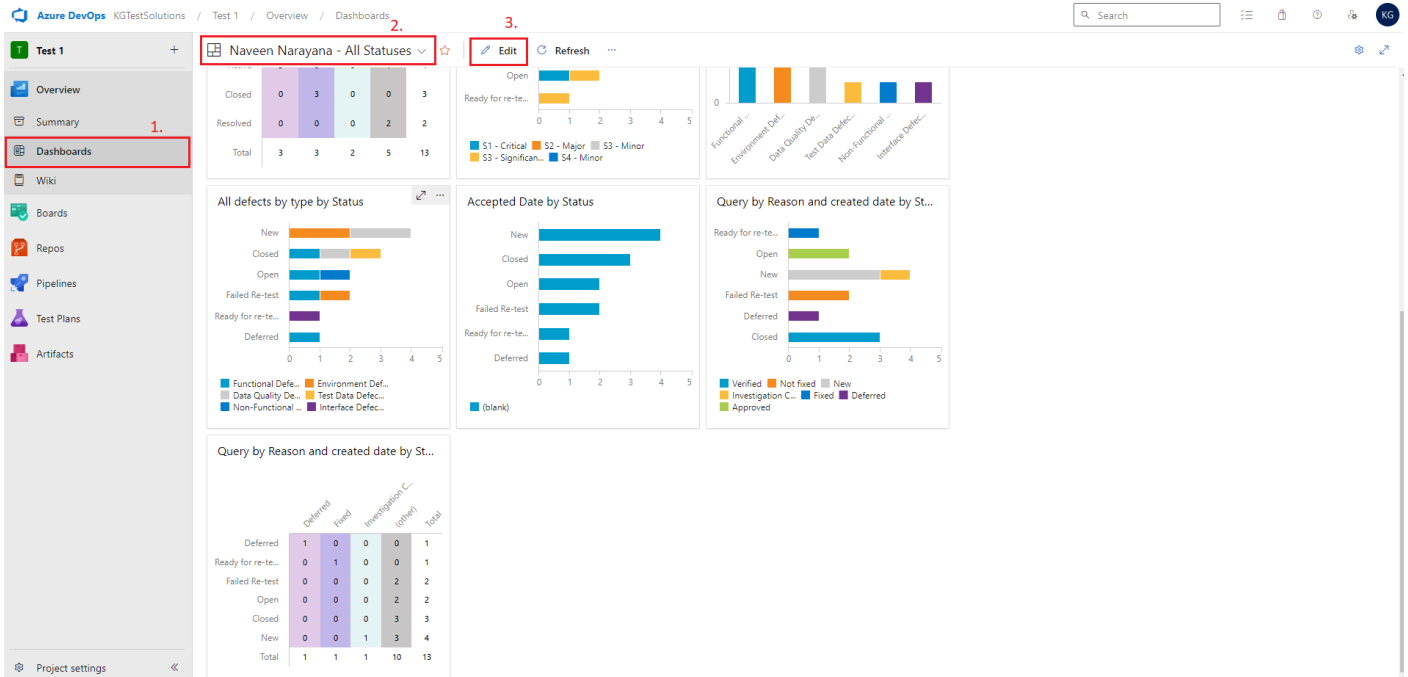
ID	Title	Assigned To	State	Area Path	Tags	Comments	Activity Date
53	Unable to select the drop down	Karthikeyan Ganesan	Active	Test 1			18/05/2023 13:39:41
67	Test 1	Karthikeyan Ganesan	Proposed	Test 1			11/05/2023 17:14:18
64	Test	Karthikeyan Ganesan	Proposed	Test 1			20/04/2023 10:39:20
48	Test 5 - Naveen mic issue	Naveen Narayana	Resolved	Test 1			20/03/2023 14:36:54
49	Test case 2 - Naveen login issue	Naveen Narayana	Active	Test 1			13/03/2023 15:56:32
50	Test case 3 - Naveen login field issue	Naveen Narayana	Active	Test 1			09/03/2023 15:37:10
47	Test 4 - Naveen mic issue	Naveen Narayana	Resolved	Test 1			09/03/2023 15:36:31
42	Not able to view items	Karthikeyan Ganesan	Proposed	Test 1			08/03/2023 14:05:11
37	pop-up page not visible	Darren Pyefinch	Proposed	Test 1			07/03/2023 11:43:28
51	Test case 5 user not able to click download button	Naveen Narayana	Active	Test 1			01/03/2023 16:10:13

1. Click on Save query
2. Give a name to the Query
3. Save the Query under shared Queries for reporting
4. Click on OK to save the Query

The screenshot shows the Azure DevOps 'Queries' page. The 'Save query' button is highlighted with a red box and the number '1.'. A 'New query' dialog box is open in the center. The 'Name' field is highlighted with a red box and the number '2.'. The 'Folder' dropdown menu is set to 'Shared Queries' and is highlighted with a red box and the number '3.'. The 'OK' button is highlighted with a red box and the number '4.'. The background shows a list of queries and a detailed view of a query titled '53 Unable to select the drop down'.

1. After saving the Queries, go to Dashboard (under Overview)
2. Create the Dashboard to create the charts on Defects

3. Click on Edit



The screenshot of dashboard is for illustration purpose only

1. Select Chart for Work Items widget to create the report
2. Click Add
3. Click on Setting to define the chart

The screenshot shows the Azure DevOps Test 1 Dashboard in 'Done Editing' mode. The dashboard displays the same charts and table. A red box highlights the 'Chart for Work Items' widget. The 'Add Widget' panel on the right is open, showing various widget options. The 'Chart for Work Items' widget is selected, and the 'Add' button is highlighted.

The screenshot of dashboard is for illustration purpose only

From the configuration pop-up screen:

1. Enter the Title of the chart, select the Query that was saved earlier and complete the required fields
2. Click on Save

The screenshot displays the Azure DevOps dashboard for 'Test 1'. The dashboard includes several charts and tables. A 'Configuration' pop-up window is open on the right side, showing the configuration for a chart titled 'Defects in Test 1 by State'. The configuration includes fields for Title, Width, Height, Query, Chart type, Group by, Aggregation, Sort, and Series. The 'Save' button is highlighted with a red box and labeled '2.'.

Configuration Pop-up Screen:

- Title: Defects in Test 1 by State
- Width: 2
- Height: 2
- Query*: Defects in Test 1
- Chart type: Pie
- Group by*: State
- Aggregation: Count of work items
- Sort: Value Descending
- Series: Proposed, Design, Active
- Buttons: Save, Close

Dashboard Charts:

- All defects by type by Status:** A horizontal bar chart showing the count of defects by status (New, Closed, Open, Failed Re-test, Ready for re-te..., Deferred) and type (Functional Defec..., Environment Defec..., Data Quality Defec..., Test Data Defec..., Non-Functional..., Interface Defec...).
- Accepted Date by Status:** A horizontal bar chart showing the count of defects by status (New, Closed, Open, Failed Re-test, Ready for re-te..., Deferred) and type (Functional Defec..., Environment Defec..., Data Quality Defec..., Test Data Defec..., Non-Functional..., Interface Defec...).
- Query by Reason and created date by St...:** A horizontal bar chart showing the count of defects by status (Ready for re-te..., Open, New, Failed Re-test, Deferred, Closed) and type (Verified, Not fixed, New, Investigation C..., Fixed, Deferred, Approved).
- Defects in Test 1 by State:** A pie chart showing the distribution of defects by state (Proposed, Design, Active, Resolved) with a total count of 15.

The screenshot of dashboard is for illustration purpose only

7 Report writer

The Report Writer supports building queries and reports for the dashboard.

Note: This is an optional feature, and the default reports have already been built on the programme participants' instance.

7.1 Building queries and reports for the dashboard

Follow these steps:

1. Click on Work Items under overview
2. Filter the defects to build the report
3. Click on Open filtered view in Queries

ID	Title	Assigned To	State	Area Path	Tags	Comments	Activity Date
53	Unable to select the drop down	Karthikeyan Ganesan	Active	Test 1			18/05/2023 13:39:41
67	Test 1	Karthikeyan Ganesan	Proposed	Test 1			11/05/2023 17:14:18
64	Test	Karthikeyan Ganesan	Proposed	Test 1			20/04/2023 10:39:20
48	Test 5 - Naveen mic issue	Naveen Narayana	Resolved	Test 1			20/03/2023 14:36:54
49	Test case 2 - Naveen login issue	Naveen Narayana	Active	Test 1			13/03/2023 15:56:32
50	Test case 3 - Naveen login field issue	Naveen Narayana	Active	Test 1			09/03/2023 15:37:10
47	Test 4 - Naveen mic issue	Naveen Narayana	Resolved	Test 1			09/03/2023 15:36:31
42	Not able to view items	Karthikeyan Ganesan	Proposed	Test 1			08/03/2023 14:05:11
37	pop-up page not visible	Darren Pyefinch	Proposed	Test 1			07/03/2023 11:43:28
51	Test case 5 user not able to click download button	Naveen Narayana	Active	Test 1			01/03/2023 16:10:13

1. Click on Save Query
2. Give a name to the Query
3. Save the Query under shared Queries for reporting

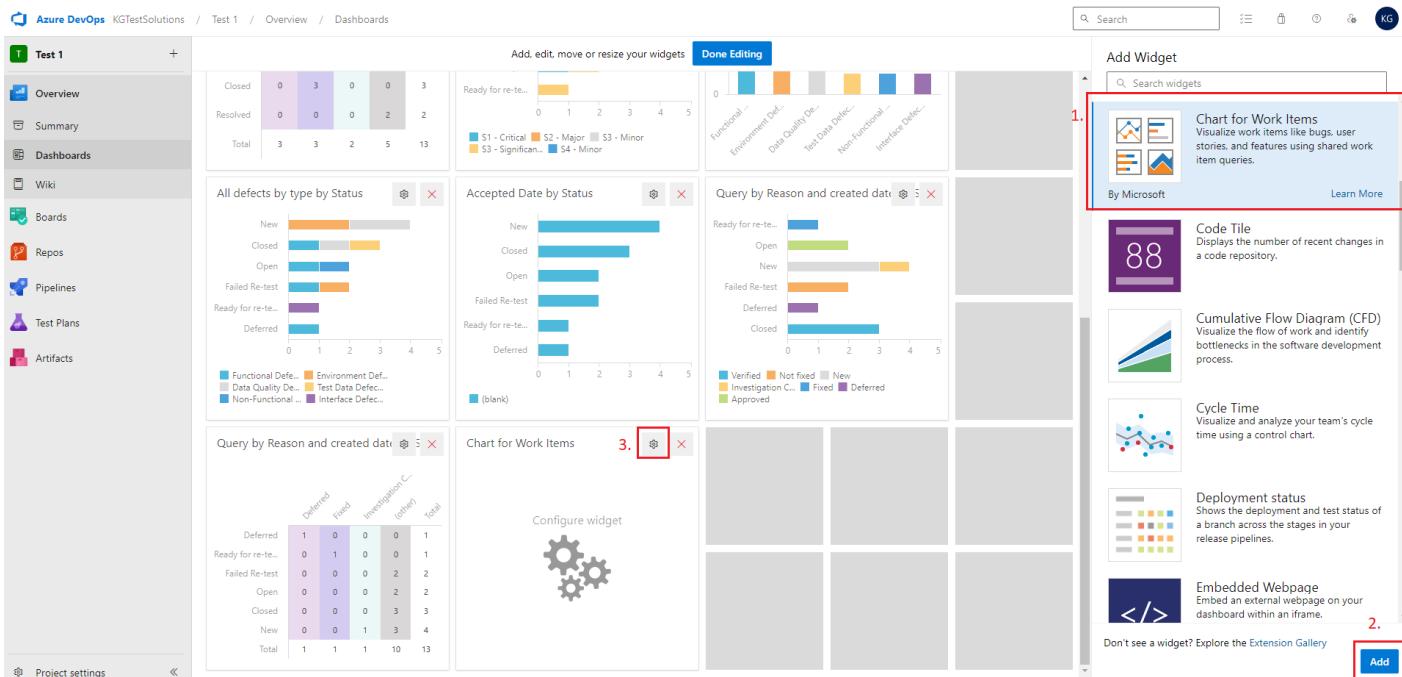
4. Click OK to save the Query

The screenshot shows the Azure DevOps interface. In the center, a 'New query' dialog box is open. It has a 'Name' field (labeled 2), a 'Folder' dropdown menu (labeled 3, currently set to 'Shared Queries'), and 'OK' (labeled 4) and 'Cancel' buttons. The background shows a list of queries under 'Test 1' and a detailed view of a specific query titled 'Unable to select the drop down'.

1. After saving the Query go to Dashboard under Overview
2. Select the Dashboard to create the charts on Defects
3. Click on Edit

The screenshot shows the Azure DevOps interface. In the left sidebar, the 'Dashboards' tab is selected (labeled 1). The main area shows a dashboard titled 'Naveen Narayana - All Statuses' (labeled 2). The 'Edit' button is highlighted (labeled 3). The dashboard contains several charts: 'All defects by type by Status', 'Accepted Date by Status', 'Query by Reason and created date by St...', and 'Query by Reason and created date by St...'.

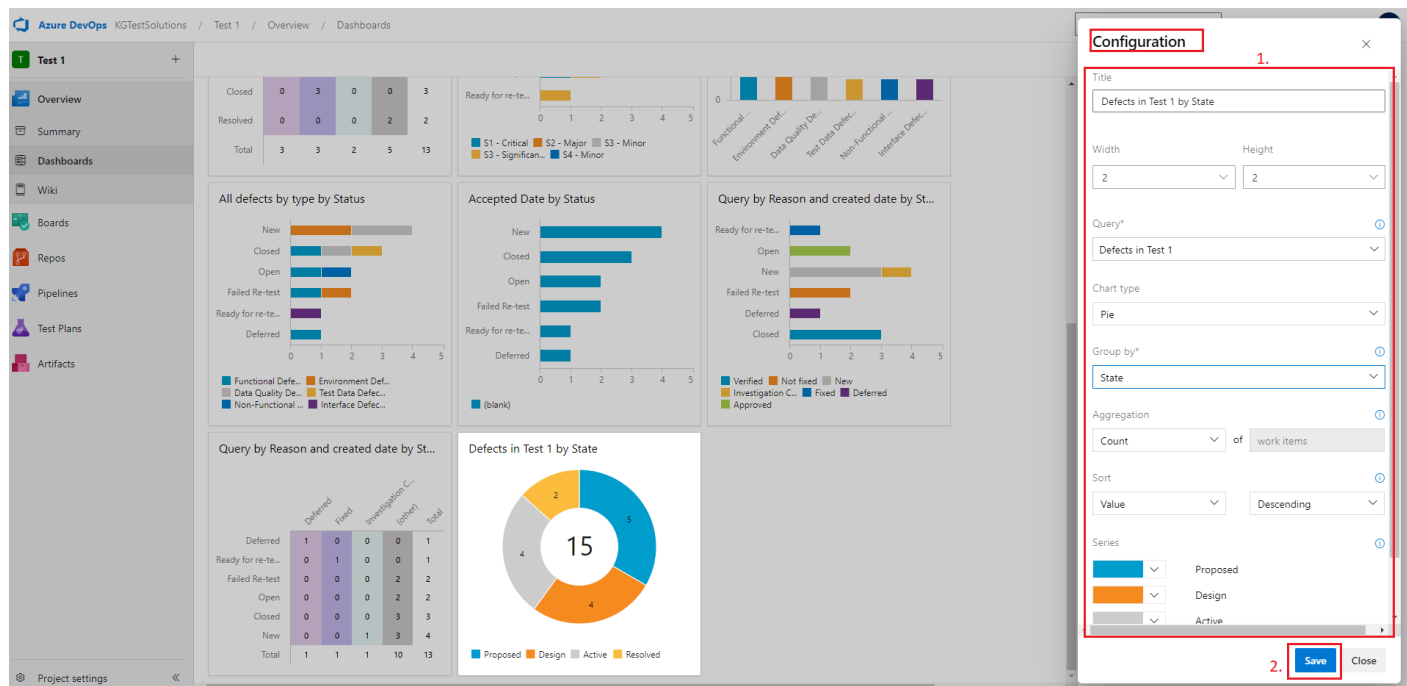
1. Select Chart for Work Items widget to create the report
2. Click Add
3. Click on Setting to define the chart



The screenshot of dashboard is for illustration purpose only

From the configuration pop-up screen:

1. Enter the Title of the chart, select the Query that was saved earlier and complete the required fields
2. Click on Save



The screenshot of dashboard is for illustration purpose only