

Wiki for describing test practises we follow at TCC as well as knowledge transfer for new testers getting on boarded.

The screenshot shows the Azure DevOps Wiki interface. The left sidebar contains navigation options: Knowledge Center, Overview, Summary, Dashboards, Analytics view, Wiki (selected), and Search. The main content area is titled 'Test DoR and DoD' and is authored by Rodney Colwin on Jul 18. It is organized into three sections: DOR (Definition Of Ready), Entry criteria for Test Execution, and DOD (Definition Of Done). Each section contains a list of bullet points detailing the requirements for test readiness and completion. The 'Exit criteria for Test Execution' section is partially visible at the bottom.

Test DoR and DoD
Rodney Colwin · Jul 18

DOR (Definition Of Ready)

- Are any required Test Environments available and are they reflective of what is in production? (i.e. Version/Data)
- Do we have enough access levels for the environment?
- Are the User story's detailed enough to derive test case scenarios from?
- Do we need sample data?
- Are any technical resources required such as a SME if so are they available?
- Do we have enough time to test to go to CAB/release?
- Do we need to write Prod verification tests?
- Is cross browser/device testing required? If so do we have required devices/tools available?
- Is automation needed?
- Security stories have been discussed

Entry criteria for Test Execution

- Business accepts the Test Plan/ Cases
- All requirements are covered
- Test cases designed and reviewed
- Unit tests completed and application available to test in Test environment
- Any Known/existing issues are communicated

DOD (Definition Of Done)

- All Tests Cases Run and Passed or if failed have a valid reason/ Workaround and accepted by PO
- No outstanding bugs without an action that is accepted by the business
- No P1 bugs without workaround or action plan with dates
- Test evidence captured for all test cases
- Report created for CAB and attached to Change Request with evidence of testing and approvals if required (Azure Dashboard)
- Any automated tests written have be stored in Azure Repo
- Security Fix review performed on User Story

Exit criteria for Test Execution

Boards showing details on user story's Number of Tests against it and their state as well as any bugs raised and their state also

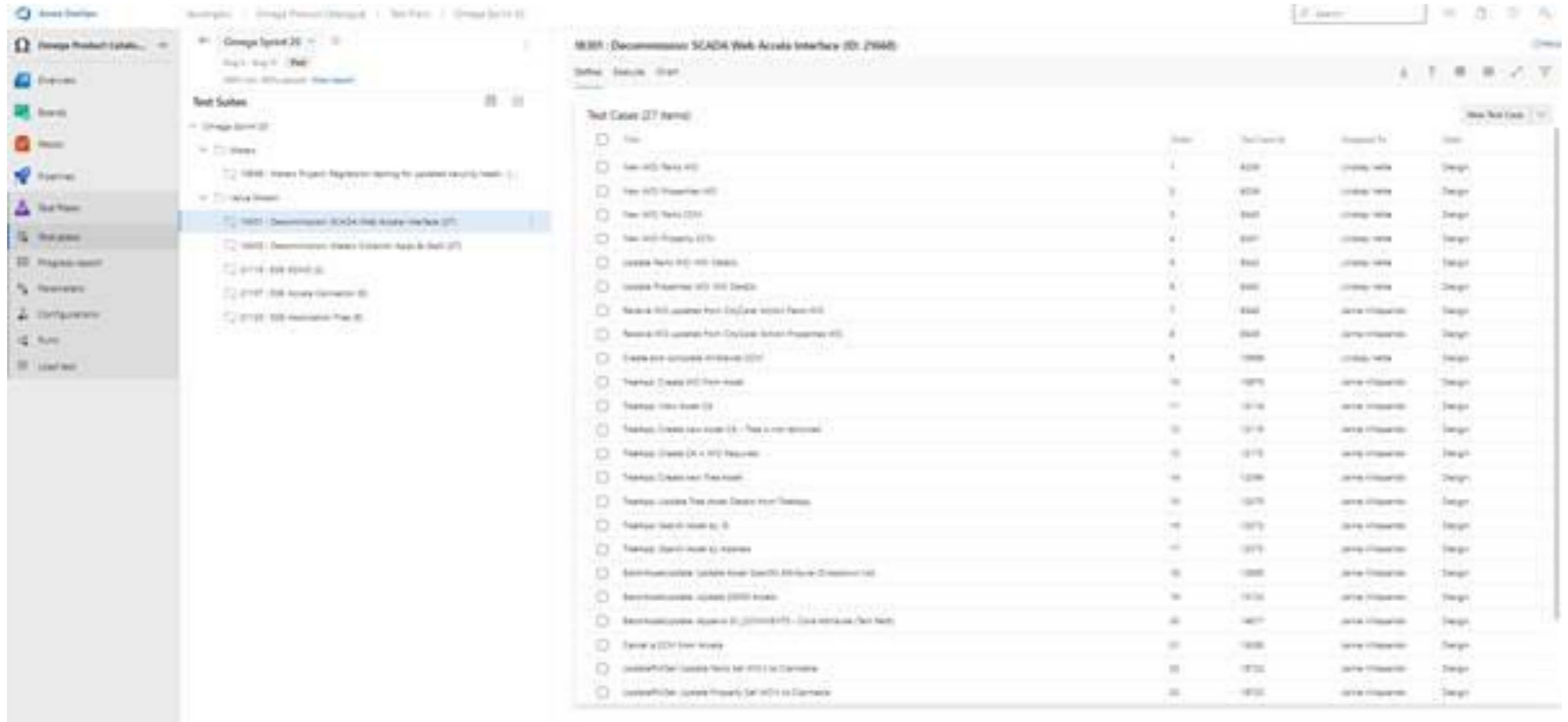
The screenshot displays the Azure DevOps interface for the 'Omega Scrum Team'. The main view is a Kanban board with the following columns and counts:

- New: 4
- Ready: 18
- Reviewed: 0
- On Hold: 2
- In Development: 26
- Testing: 10
- Closed: 4

The 'Closed' column contains the following user stories:

- 18385** Update Reporting - 1 test, 0 bugs
- 18386** Decommission Azure Container Apps & SaaS - 1 test, 0 bugs
- 18387** Decommission SCADA Web Access Interface - 1 test, 0 bugs
- 18388** Water Project Regression testing for updated security headers - 1 test, 0 bugs

Test plans where the test suites are created from existing user story's which link directly to test cases and bugs raised against it from here we are also to view execution history of the test cases to view previous run results and open the test case to see the steps involved



Execution of test cases is also done via Azure Test Runner where the assigned tester can run their tests and pass/fail each step add comments raise bugs insert supporting screen shots etc

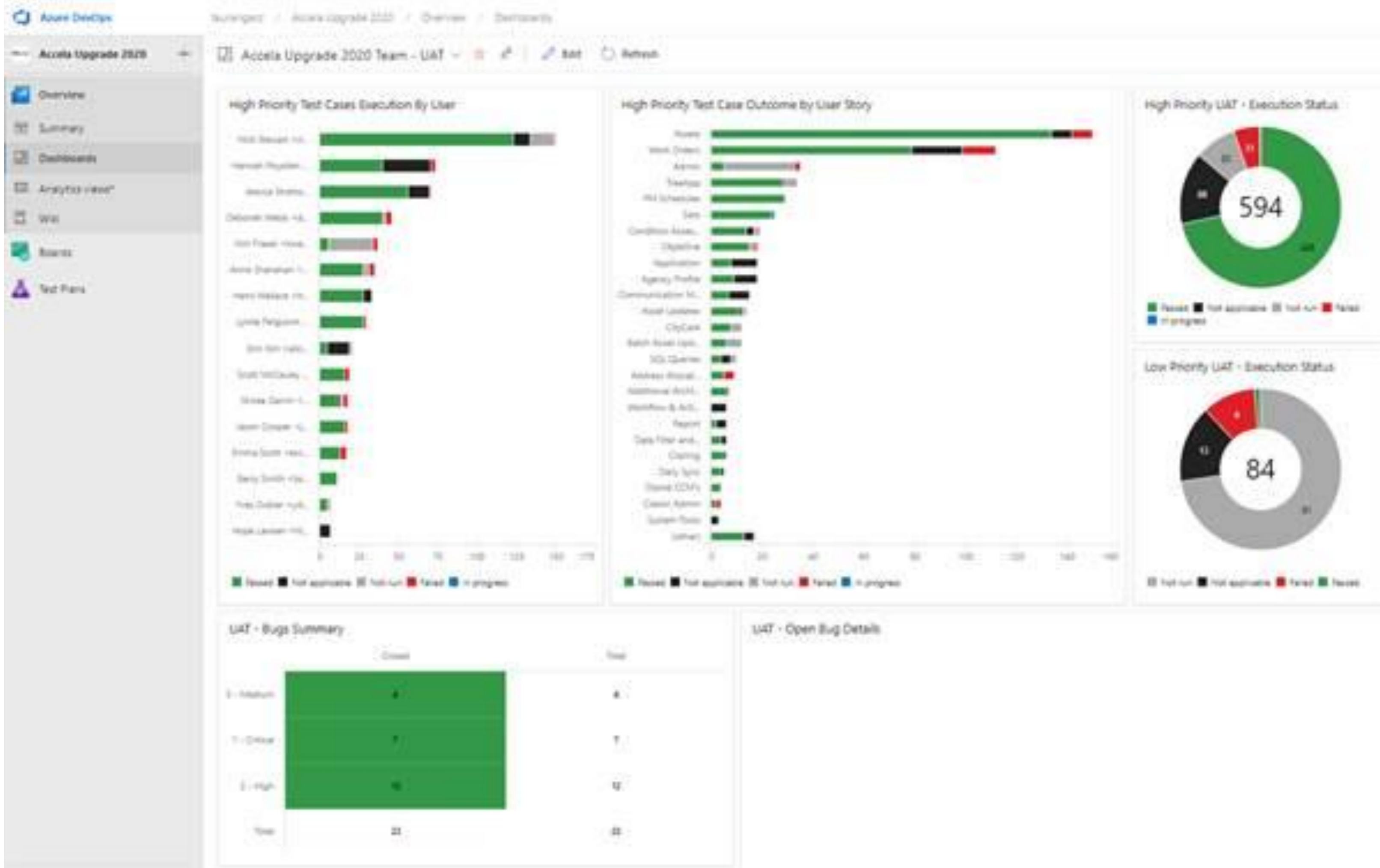
Runner - Test Plans - Google Chrome
dev.azure.com/taurangacc/Omega%20Product%20Ca...

9238: New WO: Parks WO

1. Login to Accela
EXPECTED RESULT
Successfully logged in
2. Go to Work Orders
EXPECTED RESULT
Work Orders screen is displayed
3. At the WO list, click New button
EXPECTED RESULT
New WO popup is displayed
4. Click the icon to the right of Record type field
EXPECTED RESULT
Record type list is displayed
5. Select a Record Type
EXPECTED RESULT
New WO form is displayed
6. Populate the following fields:
-Detailed Description
-Priority (A/B/C/D)
-Assigned to Department (CityC2)
-Contact
EXPECTED RESULT
Fields are populated
7. At the Assets section, click Search button
EXPECTED RESULT
Assets search popup is displayed
8. Enter filter/s to the form
Note: Asset Group/Asset Type fields will only return values applicable to the selected WO Record Type
EXPECTED RESULT
Form is filled up
9. Click Submit button
EXPECTED RESULT
Asset list is displayed. Only the applicable Assets returned in the list
10. Tick the checkbox of the Asset ID/s from the search results
EXPECTED RESULT
Checkbox/es is/are ticked
11. Click Select button
EXPECTED RESULT
Asset search popup is closed. Selected Asset/s is/are added to the Assets list of the New WO popup
12. At the Assets list, tick the checkbox of an associated Asset
EXPECTED RESULT
Checkbox is ticked
13. At the Assets list, click Get Address button
EXPECTED RESULT
Address section is populated with the linked Address of the selected Asset
14. At the Contact section, click Search button
EXPECTED RESULT
Contact list popup is displayed
15. At the Contact list popup, click Search button
EXPECTED RESULT
The popup is refreshed to display Contact search fields
16. Enter filter/s to the form, then click Submit button

Windows 10

Dashboards for displaying testing metric's in real-time as the project/sprint is progressing



Azure Repo for Version control of Automated Test Scripts shared Functions and Repository's

The screenshot displays the Azure DevOps interface for a repository. The left sidebar contains navigation options: Testing, Overview, Boards, Repos, Files, Commits, Pushes, Branches, Tags, Pull requests, My Hub, Pipelines, and Test Plans. The main area shows the 'Files' view for the 'Automation Scripts' repository. A table lists the files and their commit history.

Name	Last change	Commits
Automation Scripts	Wednesday	5102f284 UFT commit Akhil David
Shared API Tests	Aug 9	70727718 UFT commit Akhil David
Shared Functions	Wednesday	1ee2c5f8 UFT commit Akhil David
Shared GUI Tests	Jul 27	3143e561 Added File Rodney Cowin
Shared Repository	Jun 25	40888612 Updated Structure Akhil David
Ms README.md	Jun 25	4c82161f Added README.md Rodney Cowin

Below the file list, there is a template for a README file with sections: Introduction, Getting Started, Build and Test, and Contribute. Each section contains a 'TODO' placeholder and a list of items to be included in the documentation.

Introduction

TODO: Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.

Getting Started

TODO: Guide users through getting your code up and running on their own system. In this section you can talk about:

1. Installation process
2. Software dependencies
3. Latest releases
4. API references

Build and Test

TODO: Describe and show how to build your code and run the tests.

Contribute

TODO: Explain how other users and developers can contribute to make your code better.

If you want to learn more about creating good readme files then refer the following [guidelines](#) !! You can also seek inspiration from the below readme files:

- [ASP.NET Core](#) !!
- [Visual Studio Code](#) !!
- [Chakra Core](#) !!

Azure Pipeline used for triggering automated tests downloading the latest version of the scripts from version control then determining if a dataset was supplied to run the test with or to just use the default data supplied in the test itself.

Then executing the test on our on premise test server with your desired test Tool.

Then emailing the results of the test to the person who triggered the Release as well as a designated Microsoft Teams channel.

Then publishing the results back into azure to Pass/Fail the release /Build.

The screenshot displays the Azure DevOps web interface for configuring a pipeline. The breadcrumb navigation at the top reads: Home > Sigma Product Catalog > Pipelines > Releases > UFT_Sunday_Invoices. The main view shows the pipeline configuration for 'UFT_Sunday_Invoices' under the 'SAP' group. The pipeline is divided into 'Stage 1' with several tasks:

- Deployment group job
- Git Repository Clone UFT
- Copy Local Dataset if set to "Yes" at runtime
- Run UFT Test (highlighted)
- Send Test Results
- Publish Test Results "Yes" and

The right-hand pane shows the configuration for the 'Run UFT Test' task, titled 'UFT File System Evaluation'. It includes a dropdown for 'Run version' set to '1.1', a 'Display name' field containing 'Run UFT Test', a 'Script' field with the path 'C:\UFT\src\Repo\OT Automation Scripts\SAP\Releases\Def\UFT\runTest.ps1', and a 'Timeout' field set to '60'. There are also sections for 'Control Options' and 'Output Variables'.