

Single Record Testing Process

When looking to run testing for a new project, the following steps should be taken to ensure, that where possible, Automation and API testing are taken into consideration and implemented as soon as possible:

User Requirements Review

Ideally, prior to any backlog refinement meeting, the Lead Test Analysts, (along with some tech/dev support if possible), will look to review the requirements and decide upon the following factors:

- Does the requirement contain enough information to decipher the desired functionality, and where possible, suggest any alterations that would be required?
- Are there any technical elements that need to be investigated prior to testing, e.g. SQL, API, configurational (app files) testing, etc?
- Is the project suitable for automation, with regards to the scale of the project, efficiency gained from automation, is the code likely to change often, how complex is the functionality, etc?

Technical Elements Review

Once accepted into a sprint, Lead Test Analysts will look to identify areas within the requirements that can be tested efficiently using testing tools, such as SQL:

- SQL databases used within the requirement, e.g. calls to reference data, user account details, audit logging, environmental changes (Web Services, etc).
- Web Services/API endpoints used within the requirement, e.g. Can API testing against an endpoint be used as comparison against those objects within the requirement?
- Data Required to fully test the requirement.

Areas where this can be added, will be highlighted and any API/SQL projects and required data should be generated and shared with the Test Analyst for inclusion in any test cases.

Manual Testing Review

Once accepted into a sprint, the Test Analyst will also look to begin writing the test cases in Gherkin syntax, ensuring a review of these is undertaken by a Technical Analyst to ensure it conforms to the current Automation Framework. Even when there are areas within an automated project that will require manual testing, these areas should be fully reviewed from a technical aspect, as it may be possible these can be automated later in the project lifecycle.

There will be projects where, it will be impossible to automate testing straight away, due to many factors such as, work required in the background any application to successfully implement the requested functionality, meaning there is no physical UI layer with which to interact. Therefore, we need to ensure all projects are fully updated with manual test cases, written in Gherkin syntax, to enable a manual process in the event of automation failure/delay.

Automation Review

When looking at automation, the size/complexity of the project and the available resource within the Test Team may require a single resource to provide both roles. In these instances, we should look to ensure another member within the team is available to assist with any review, and all automation reviews performed should align with the following:

- At which stage will automation be possible?
- Can the UI be accessed prior to availability in SIT/Can we access the dev code?
- Does the project Parameters class require updating?
- Can SQL/API testing be brought into the automation project, e.g. API testing via JSON, SQL Testing via DbConnect)?

Final Project Review

Prior to delivery of any project to UAT, a full review should be conducted with all involved parties, to review the following:

- Complexity of project and if Automation assists with validation.
- Size of the project and where Automation provides efficiency compared to manual testing.
- Time taken to automate, and testing coverage provided within the project.
- Can the overall process be improved?