

Additional questions – test strategy



**ASSIGN
BUSTER**

Additional Questions – Test Strategy Table of Contents

- 1 Purpose
- 2 About Additional Questions
- 3 Requirements
- 3.1 As-Is and New AQ Features
- 4 AQ As-Is Feature from V14 AQ Additional / Enhanced Feature for V24 ? Create AQ functional Component ? AQ FCA linked to Scoring FCA ? Create AQ Event by copying an existing event (closed, open, draft & archived) ? Invite Supplier
- 3.2 AQ High Level Functionality and List of User Stories
- 5 Table below gives the mapping of User Stories against the As-Is AQ Functionality
- 5 Table below gives the mapping of User Stories against the New AQ Functionality
- 4 Additional Questions Development
- 6.1 AQ Feature Development in Releases
- 6.2 In Sprint Testing (Blue are new AQ features)
- 4.3 System Testing
- 4.4 In Sprint Testing Dependencies
- 4.5 System Testing Dependencies

1 Purpose The purpose of this document is to give the overview of the Test Strategy (Approach) for Additional Questions Feature which will be developed and Implemented for V2. This document briefs about AQ functionality and its requirements, the development approach, this document also discuss about the various levels of testing AQ as its being developed & the dependencies for testing. About Additional Questions Additional Question is an existing functionality of V1 Accelerate application, hence it's known as As-Is feature for V2. And also have some additional/enhanced features for V2. This Additional Questions (AQ) is a buyer centric functionality and in V1 AQ is not a common feature which is available for all Buyers. It's a Bolt-On feature, where a Buyer organisation can opt for this feature by making additional payment. AQ functionality has been developed and implemented in V1 in a way it can be configured for any Buyer Organisation.

AQ provides an additional edge to Buyer Organisation where they can ask their intended Questions to a specific set of suppliers of their own interest and short list them upon their response. Using this feature member of a Buyer Organisation, who is having permissions to AQ can create an AQ Event which comprises of a template with questions, and the member can search for suppliers and invite them all or a specific set of suppliers to answer the questions, on or before a dead line date set for the particular AQ event by the member.

Buyer user can create AQ Events and save it for future purpose, existing AQ event can be used by some other Buyer Organisation member and the member can use it as it is or can do some changes before inviting the suppliers. Even the invited AQ events can be reused. Buyer User will be able to add or remove questions to an AQ event. A. Q summary is the last stage of draft event in which the user can preview the whole event and also review it before sending it to the supplier users. Supplier needs to respond to AQ and buyer will rate supplier according.

Scoring engine will help AQ to rate Supplier. AQ product needs to be configured by C3 user with all required features. AQ will be added as a component which will be inherited by child product (community). * 3 Requirements Requirements developed as User Stories and reviewed by Products & Services team and approved. Both the existing features (As-Is) and the new V2 specific additional / enhanced features are also covered in user stories. The below table will give the bifurcation of Existing and New features of Additional Questions. . 1 As-Is and New AQ Features * AQ As-Is Feature from V1| * AQ Additional / Enhanced Feature for V2| * Create AQ

Template | * Create AQ functional Component| * Create AQ Draft event & review of AQ| * AQ FCA linked to Scoring FCA| * Create AQ Event using AQ Template| * Permission to C3 User for AQ| * Create AQ Event by copying an existing event (closed, open, draft & archived)| * Create & use AQ Library * | * Invite Supplier| * Language Support to AQ| AQ view for GBO | * Buyer assigns scoring to AQ Event| * AQ view for Buyer| * Scoring can be manual or automatic * | * AQ view for Supplier| * AQ alert to buyer when Current Date + 7 >= End Date| * Modification to AQ Open Event | * Reminder email to supplier who has not responded| * Categorisation of AQ Open Event (Responded, Not Responded, Not Interested)| * Clarification asked by Buyer on AQ supplier response | * AQ Reports| * AQ scoring for each supplier| * | * Comparison of AQ with respect to supplier| * | * Buyer rate & email to supplier on AQ Close Event| * 4. 2 AQ High Level Functionality and List of User Stories * Table below gives the mapping of User Stories against the As-Is AQ Functionality Area| Functionality| User Story| GBO| | | | | | | | Buyer| | | | | | | | | | Supplier| | | | | | | | * Table below gives the mapping of User Stories against the New AQ Functionality Area| Functionality| User Story| GBO | | | | | | | | Buyer| | | | | | | | | | Supplier| | | | | | | | * 4 V2 Development Strategy
 5. 3 Development Approach V2 development is a mix of both Waterfall and Agile development framework.

The Development of requirements follows waterfall, where as the actual code development will happen in agile methodology. First all the User Stories will be written, reviewed and signed off by the stake holders. Development will follow the high level milestone plan, which comprises of internal releases and Demo Release. Date| 1/Dec/12| 1/Jan/13| 15/Jan/13| 29/Jan/13| 1/Feb/13|

1/Mar/13| 15/Mar/13| 29/Mar/13| Release| Alpha 0. 1| Alpha 0. 2| Alpha 0. 3|
Alpha 1. 0| Alpha 1. 1| Alpha 1. 2| Alpha 1. 3| Alpha 2. 0| Purpose| Internal|
Internal| Internal| Board| Internal| Internal| Internal| Board|

As mentioned above each and every internal release has multiple Iterations for development and the user stories will be allocated to these iterations for development. Within these iterations all the allocated user stories will be developed and tested In-Sprint Testing. 5. 4 AQ Feature Development in Releases As explained above, AQ as feature to be developed for V2 will also follow the same development methodology, All User Stories belongs to AQ will first written, reviewed and signed off, and then developed in multiple releases in multiple iterations.

The table below will give us the picture of AQ Development in Release and Iteration wise. The Main purpose of the below table is to give clear picture of When AQ feature development will be started and in which release it will developed and delivered 100%. Table below will not give the Start and End dates of either each and every release or Iterations, these dates are available with the development team and ion their plan.

Release Iteration	User Story Functionality / Area
Internal Release Alpha 0. 1	Iteration 1
Iteration 2	Iteration 3
Iteration 4	Internal Release Alpha 0. 2
Iteration 1	Iteration 2
Iteration 3	Iteration 4
Internal Release Alpha 0. 3	Iteration 1
Iteration 2	Iteration 3
Iteration 4	Demo Release 1. 0
Iteration 1	Iteration 2
Iteration 3	Iteration 4
Internal Release Alpha 1. 1	Iteration 1
Iteration 2	Iteration 3
Iteration 4	Internal Release Alpha 1. 2
Iteration 1	Iteration 2
Iteration 3	Iteration 4
Internal Release Alpha 1. 3	Iteration 1

Iteration 2| | | Iteration 3| | | Iteration 4| | | Demo Release 2. 0| Iteration 1| | | Iteration 2| | | Iteration 3| | | Iteration 4| | | 5 AQ Test Approach This section describes the test approach for Additional Question by explaining the following. * Testing Scope of Additional Questions * Dependency with other critical V2 functionalities * Integration with Third Party Application Features * Levels of Testing * Test Design & Execution * Test Data Requirements * Functional Automation Testing * Non Functional Testing Scope 6. 5 Scope of Additional Questions

Additional Question as feature it spread across all the major areas of V2 application, though it's Buyer centric feature it's having scope in C3 (Configuration Control Center), GBO (Global Back Office), SCC (Standard Company Concept) Buyer & Supplier. End to End Perspective From an end to end perspective the AQ functionality testing needs to be started from C3 (may be configuration of AQ - need additional information to complete it) and then SCC-Buyer can create AQ Events and invite Suppliers and then SCC- Suppliers can respond for AQ events.

From GBO-User perspective AQ needs to be tested for AQ Dashboard, AQ Reports, etc... (need additional information to complete this section) Non Functional Perspective - Performance Few AQ Features needs to be tested for Performance * AQ template creation with more Questions * Inviting multiple suppliers * Response time of viewing AQ responses * AQ Report Generation for SCC-Buyer & GBO User Multilingual Perspective As Additional Question can be configured for any scheme / community, and we have communities which support more than one language, AQ needs to be tested in all applicable languages.

Cross Browser Perspective As Buyer organisation members and Suppliers can use any browser to access the application, Additional Questions features needs to be tested in multiple browser combinations. Community Specific Additional Questions can be configured for specific communities and as we already have few communities effectively using AQ, all such communities needs to be tested thoroughly AQ feature. Supplier Perspective AQ needs to be thoroughly tested with * Newly Registered Suppliers * Migrated Suppliers

6. 6 Additional Questions Dependency with other Features. Additional

Questions is one of the feature in V2 application, this feature has some integration / dependency with some other features of V2 application. This section describes the AQ dependency with such other features of V2. This dependency may play a critical role in AQ testing at functional level and at end to end testing, we may have a work around to bypass the dependency, where as some feature must be available to test AQ. AQ Feature| Dependent Feature| Dependency level| When This Feature will be Ready| Do we have Work Around| Work Around| Effectiveness of Work Around| Invite Supplier| Search| Very High| ? ? | | ? | | Email Generation| Very High| ? | ? | | ? | | Existing / Migrated Suppliers| High| ? | ? | | ? | All such dependencies for all AQ features needs to be captured in a separate excel sheet and attached to this document. And the Dependencies needs to be discussed with Development team, as we may need their help for some work around or the feature needs to be developed in a priority. This dependency is very critical for test execution. 6. 7 Integration with Third Party Application Features This section needs to be filled in 6. 8 Levels of Testing

Additional Questions will be tested at In-Sprint Testing and System Testing, this section describes what will be covered in In-Sprint Testing and System Testing. In-Sprint Testing In-Sprint Testing is part of development, In-Sprint testing team will work along with Development team for Release and Iterations, user stories assigned to iterations is the scope for development and testing, while development teams starts coding In-Sprint testing team starts test design, when the feature is developed and ready for testing, In-Sprint test team will test the feature and give the result.

In-Sprint test team covers Unit, Integration, Continuous Integration and Regression Testing. In-Sprint Unit Testing In unit testing, team checks for the following using Checklists * Field Level Validation of all controls * Boundary Value, Equivalence Partitioning * Page Navigation on Links * Messages (information on control validation, Tool Tips, etc...) * Page Templates, Company Logo, T&C's, Copy Right, etc... * Cross Browser – All UI design needs to be validated with all applicable browser combinations

In-Sprint Integration Testing Team test the features developed integrated with preceded and following features of a feature belongs to same module (Group of requirements, belongs to one user story) within the iteration as the features being developed in iterations. And also the integration of group of requirements developed (one User story) with another group of requirements (another User Story) as iteration is having multiple User stories. In-Sprint test team will write functional test cases to test this integration of features.

In-Sprint Continuous Integration Testing Team tests the integration of features developed in multiple iterations, as the features developed across multiple iterations of any release are being continuously integrated together.

Team may write separate set of integration test cases for this else they will enhance their existing integration test cases to test this, in an another approach team can group set of individual integration test cases and execute them in an order which covers this continuous integration.

In-Sprint Regression Testing Test team executes all the test cases which belongs to earlier release for any successive release, this is to ensure that the new release features are not hampering the existing features already developed and tested and also ensures the integration of features in between two successive release is working fine. Table below gives a picture of Unit, Integration, Continuous Integration & Regression testing for Releases and Iterations.

Release	Iteration	User Story	Feature	Testing	Unit	Integration	Continuous Integration	Regression
Release 1	Iteration 1	User Story 1	Feature 1	Y				
			Integration of feature 1+2+3+4					
			Continuous Integration of User Story 1 + 2+ 3					
			Release 1 TC's Regression Suite for Release 2					
			Feature 2	Y				
			Feature 3	Y				
			Feature 4	Y				
			User Story 2	Y				
			Feature 5	Y				
			Integration of feature 5+6+7+8					
			Feature 6	Y				
			Feature 7	Y				
			Feature 8	Y				
			User Story 3	Y				
			Feature 9	Y				
			Integration of feature 9+10					
			Feature 10	Y				
			Iteration 2					
			User Story 4	Y				
			Feature 11	Y				
			Integration of feature 11+12+13+14					
			Continuous Integration of User Story 1 + 2+ 3+4+5+6					
			Feature 12	Y				
			Feature 13	Y				
			Feature 14	Y				
			User Story 5	Y				
			Feature 15	Y				
			Integration of feature 15+16+17+18					
			Feature 16	Y				
			Feature 17	Y				
			Feature 18	Y				
			User Story 6	Y				
			Feature 19	Y				
			Integration of feature 19+20					
			Feature 20	Y				

Release 2| Iteration 3| User Story 7| Feature 21| Y| Y| Y| Release 1 + Release 2 TC's Regression Suite for Release 3| | | | Feature 22| Y| | | | | Feature 23| Y| | | | | Feature 24| Y| | | | | User Story 8| Feature 25| Y| Y| | | | | Feature 26| Y| | | | | Feature 27| Y| | | | | Feature 28| Y| | | | | User Story 9| Feature 29| Y| Y| | | | | Feature 30| Y| | | | | Iteration 4| User Story 10| Feature 31| Y| Y| Y| | | | | Feature 32| Y| | | | | | Feature 33| Y| | | | | | Feature 34| Y| | | | | User Story 11| Feature 35| Y| Y| | | | | Feature 36| Y| | | | | | Feature 37| Y| | | | | | Feature 38| Y| | | | | | User Story 12| Feature 39| Y| Y| | | | | Feature 40| Y| | | | System Testing System testing will be executed by System Testing team, System test design and execution will be done as End to End level. In System test design we will write Test Scenarios and Prepare Test Data to execute the System Testing. Test Scenarios will cover multiple functionalities with all permutation and combination of functionalities to test all possible real time end to end scenarios.

System testing will also cover the end to end scenarios for maximum number of suppliers, for example System testing will be executed for inviting 500 suppliers for an AQ event, creating an AQ event with 100 Questions. And to execute the end to end testing for such high volume of suppliers we need support from Automation, because as the V2 application being developed right from the scratch, existing suppliers not available in the system, hence we need to first register for 100 of suppliers, registering 100 of suppliers will require high resource count and it's practically not possible.

And also respond to an AQ event sent for 100 of suppliers it take time and resource, if it needs to be done manually, in such scenario we need Automation script to complete the task. 6. 9 Test Design 6. 10 Test Data

Requirements 6. 11 Functional Automation Testing 6. 12 Non Functional
Testing Scope