Template-Test Plan Standards

Contents

Flight Profile Optimization (FPO)

IFT Test Plan

Test Plan Type: Integrated Functional Testing & Regression

Status: Draft

Version: 1.01

Prepared by: Md Zaman

Project Sponsor: Collins

Project Manager: MD Zaman

Project ID: Collins FPO

Alph Release #: 17 May 2020

Date Created: 18 Mar 2020

Date Last Modified: 31 Mar 2020

Document Purpose

The purpose of this document is to describe the detailed specifications for the test scenarios, cases and scripts required to complete the specific type of testing (Integrated Functional Testing and/ or Regression and/or API). It will fully define the testing approach to validate that the agreed-upon application solution will perform as documented in the BRD and FSD.

NOTE: Integrated Functional Testing (IFT) and Regression may be documented separately or combined into one plan

Customer Vision & Problem Statement

For pilots and airlines who need to improve the comfort and convenience of their passengers' travel experience, Collins FPO is a sophisticated, NASA-tested route optimization tool that offers efficient traffic- and weather-aware in-flight reroute solutions for a faster, more fuel-efficient flight.

Domestic airlines in the U.S. spend an estimated combined $2 to $5 billion on jet fuel every month. JetBlue and other airlines are looking for an Electronic Flight Bag (EFB) solution that helps them optimize fuel and time on their flights to reduce operational costs and improve the passenger experience.

Approvals

Documentation of approval is required from those designated as an approver in the table below.

Project Role & Responsibility	Approver's name	Approval Date	Approval Method
Project Manager	MD Zaman		
Product Manager	MD Zaman		
Product Owner	MD Zaman		
Full Stack SQAE Lead	MD Zaman		

Contributors

The list of individuals who provided content to this document

Name	Project Role
MD ZAMAN	Full Stack QE Lead

Reviewers

The list of individuals had the opportunity to review this document and provide input prior to submission for approval.

Name	Project Role
MD Zaman	Project Manager
MD Zaman	Full Stack QE Lead
MD Zaman	Product Manager
MD Zaman	Tech Lead
MD Zaman	Developer

Version History

List of all revisions to the original artifact, starting with the publication of the original artifact with the author listed.

Version #	Revision Date	Revised by	Description of change
01	18 Mar 2020	Product Owner	N/A
02	01 Apr 2020	Product Owner	N/A
03	15 Apr 2020	Product Owner	N/A

Glossary of Terms

An inventory of terms and acronyms within this document that may be unfamiliar to readers

Terms/Acronym	Definition
AT	Authentication Tracker
BRD	Business Requirement Definition document
BPP	Business Process & Procedures
CR	Change Request
E2E	End to End Testing
End to End Testing	Testing of a complete application environment in a situation that mimics real-world use.
FSD	Functional System Design document
Functional Testing	Testing geared to functional requirements of an application
GUI	Graphical user interface
LOB	Line of business

PDD	Project Definition document
SQA/QE	Software Quality Assurance/Quality Engineer
Regression Testing	Testing after fixes or modifications of the software or its environment to ensure existing functionality is not impacted
SLA	Service Level Agreement
UAT	User Acceptance Test
Unit Test	The most ‘micro’ scale of testing; to test particular functions or code modules.
T.C	Test Cases

1. 1. Testing Approach – required

This project will be done by using Agile methodology as well as the iterative approach to development and testing. The test plan will be updated after each iteration or Sprint cycle with more specific information related to what occurred. This includes updating eh detailed testing scope as well as after each iteration cycle. Testing will be performed in the SQA testing Environment.

The following testing may be included depending on the needs, user stories developed & tested, and the extent of the iteration/Sprint:

- Cross-Functional and Integration Manual & Automation Testing for frontend (UI)
- Integration Manual & Automation Testing for backend (API)
- Automated Regression Testing
- Performance Testing

Note: Test Strategy and Package Test Plan contains the Testing Details that will be covered for the scope of this project.

1. 2. Impacted Areas

Required if Test Strategy or package Test Plan not completed

List the areas that will be impacted by testing including a high-level description of their participation.

Test Team or Team support testing	Role & Responsibilities
FPO SQA	Cross-Functional testing Integration, Regression testing, Automaton E2E testing
FPO Dev	New Development/Unit Test
Web Services	RESTful/API’s testing using Traffic Parrot & RestAssured

1. 3. Testing Assumptions, Constraints, Dependencies, and Risks – required

List of Assumptions, Constraints, and dependencies that will impact the specific type of testing detailed in this plan. This section should include descriptions of any dependencies on external resources such as support teams or other packages to successfully complete the planned testing, as well as any other dependencies that other projects or packages have on the testing being done in this plan. This section should also include a process for communicating delays in testing and define a remediation process to position the package to continue testing.

3.1. Assumptions

- User stories in iteration/sprint complete and approved.
- Acceptance criteria reviewed and approved.
- **Test environments available (Native App) (Change based on the project scope)**
- Adequate resources will be available to complete end to end testing
- Adequate resources will be available to resolve issues and/or defects.
- The code is in process or has been unit tested by the development team and there are no high issues impacting SQA testing.
- SQA is expecting unit test results for each iteration.
- Applications will complete their cross-functional integration/regression testing and end to end testing by the defined completion date for each iteration.
- During test script development, the business analysts will be available to the SQA testers to answer any application documentation or functionality questions.
- Prolonged issues/outages with test environments and/or applications will be escalated for an immediate resolution to limit any negative impact on testing efforts.
- SQA will have adequate time to complete planned cross-functional integration testing.
- SQA and all applicable project team members will participate in all test meetings and project meetings to ensure the end to end test requirement is met.
- Required test systems will be available.
- The root cause (what is causing the defect to occur), resolution (what exact changes were made to resolve the defect) and impacted functionality will be provided for every defect fix in the defect database being used to track defects.

- Test data requirements have been identified in a timely manner. Test data is available within enough time to start testing each iteration.
- The Development teams are able to run around any defect fixes in a timely manner, and these fixes are unit tested by the development team before moving them into SQA Environments.
- Test scripts will be written the first few days of week 1 of the iteration/Sprint while Development codes.
- The development will code, and SQA will test during the remainder of the 1st week and all of the 2nd week of the iteration/Sprint.
- Ideally, SQA will test during week 3 of the iteration without any builds, providing the most stable code possible to test again. It is understood this may change based on workload within each iteration and defects found.
- Test results summary will be provided with results for each iteration/Sprint.

3.2. Constraints

- Testing will be completed while development is coding
- Timely communication of all internal and external defects by all applications
- Timely resolutions of identified issues/defects

3.3. Dependencies

- Extended team member identified, contacted and committed to iteration
- **Middleware mapping, coding, schema complete for iteration (Change based on the project scope)**
- Changes to system architecture, test environment, or other major components occur during the course of the iteration.
- Inability to resolve internal or external defects
- Primary team members and extended team members are available to resolve problems that will affect the completion of testing activities.
- Deploy of code within the timeframe allotted.
- Code available to test stories successfully
- Completion of user stories in each iteration.
- Completion of end to end test scenarios and requirements.
- Quick resolution of identified issues and/or defects.
- **Daily builds to occur every commit on github. (Change based on the project scope)**
- **Overnight builds to occur every morning at 8 am. SQA has the right to request additional builds/deploys as needed. (Change based on the project scope)**
- SQA environment will have code available for current & previous iteration for testing.
- UAT Environment is available to use as needed. Usage may change from iteration to iteration/sprint.

3.4. Risk

- Incomplete environment to support testing.
- Delays in delivery of user stories or coding to support testing by the start date.
- Dedicated resources unavailable
- Delay in the completion of the test planning phase.
- Unstable test environment.

1. **4. Testing Details – required**

Documents relevant details for the specific type of testing to be conducted

Iteration/Sprint 1:

4.1. Integrated Functional Testing

4.1.1. Description

Provided a detailed description of the testing activities that will be performed.

Integrated Functional (UI) & Integration (API) Testing

- The following User Stories will be included in

Iteration/Sprint 1: 18 Mar 2020 - 31 Mar 2020 Story point - 27

1. CF-89 Display information for lateral route suggestion (unsettled) - 8
2. CF-90 Display information for vertical route suggestion (unsettled) - 2
3. CF-91 Display information for combo route suggestion (unsettled) - 2
4. CF-308 Style lateral route suggestion - 8

5.  CF-310 Style vertical route suggestion - 2
6.  CF-309 Style combo route suggestion - 2
7.  CF-291 Toggle route map to north-up view -3

Iteration/Sprint 2: 01 Apr 2020 - 14 Apr 2020 Story point - 38 (27+11)

1.  CF-89 Show lateral route suggestions in the application component - 8
2.  CF-90 Show vertical route suggestions in the application component - 2
3.  CF-91 Show combo route suggestions in the application component - 2
4.  CF-308 Style lateral route suggestion - 8
5.  CF-310 Style vertical route suggestion - 2
6.  CF-309 Style combo route suggestion - 2
7.  CF-292 Show ownship position on the active route - 8
8.  CF-291 Toggle route map to north-up view - 3
9.  CF-430 Select an optimized route - 3

Iteration/Sprint 3: 15 Apr 2020 - 28 Apr 2020 Story point - 56

1.  CF-89 Show lateral route suggestions in the application component - 8
2.  CF-90 Show vertical route suggestions in the application component - 2
3.  CF-91 Show combo route suggestions in the application component - 2
4.  CF-308 Style lateral route suggestion - 8
5.  CF-310 Style vertical route suggestion - 2
6.  CF-309 Style combo route suggestion - 2
7.  CF-292 Show ownship position on the active route - 8
8.  CF-291 Toggle route map to north-up view - 3
9.  CF-430 Select an optimized route - 3
10.  CF-165 See current active route (track up) - 5
11.  CF-165 See current active route (track up) - 5
12.  CF-165 See current active route (track up) - 8

4.1.1.1 Story vs Test vs Defect vs Automation density

a. Total Story vs Test vs Defect vs Automation on the project

Total User story	Total StoryPoint	Total Defect	Total Manual UI T.C	Total Manual API T.C	Total UI Automation	Total API Automation	Total AutomationTechDebt(Story)

b. Story vs Test vs Defect vs Automation on the sprint

Sprints	UserStory	StoryPoint	Defect	Manual UI T.C	Manual API T.C	Automation UI Story	Automation API Story	NoNeedQA	AutomationTechDebt (Story)
Sprint 1									
Sprint 2									
Sprint 3									
Sprint 4									
Sprint 5									
Sprint 6									
Sprint 7									

4.1.2. Test Environment

Provide details related to the test environment that will be used (i.e. URLs, Schedule, etc.…)

For iteration/Sprint 1, we will be provided with URLs from STM Dev to be able to access and test the application for this iteration/sprint.

Test Environment: QA or Staging

4.1.3. Target Test Schedule (Deliverables/Milestone)

Milestone	Strat	End	Resource
Test Plan Iteration/sprint	18 Mar 2020	31 Mar 2020	MD Zaman
QA Environment	TBD	TBD	MD ZAMAN
Test Summary Report	TBD	TBD	MD ZAMAN

4.1.4. Source of Test Data

- Yes/No - Test data needed for Iteration (if yes test data will be provided by P.O)

4.1.5. Type of Test Data

Describe the type of test data that will be used during testing (i.e. TBD)

- Yes/No test data needed for Iteration (if yes test data will be provided by P.O)

4.1.6. Entrance Criteria (Defination of Ready)

List the prerequisites to start testing

The information below applies to all iterations:

- All user stories and user story acceptance criteria for the current iteration are approved by the project team.
- SQA Test plans are complete and approved for the current iteration
- SQA Test script s have been created for each iteration
- The developer(s) has performed or is performing Unit and Integrated Unit Testing and created supporting documentation for each iteration.
- Unit Test cases executed with no high severity (sev 1 or 2) defects
- Service virtualization and test data creation completed
- The test environment(s) are available and correctly configured for SQA testing
- Interfacing applications are ready for Software Quality Assurance (SQA) testing to begin
- Test data has been obtained for each iteration/sprint
- The development has discussed with SQA what coding will be completed for each iteration.

4.1.7. Test Scenarios, Cases, and Scripts

If the test scenarios, cases, and scripts are located in JIRA Plugin then provide the JIRA Plugin domain, project and other relevant information that will allow an individual to locate the information within JIRA. Otherwise, use this section to document the test scenarios, cases, and scripts required for testing

JIRA URLs - Jira Backlog List

4.1.8. Special Test Requirements

Document additional testing requirements if any

4.1.9. Exit Criteria (Defenation of Done)

List the prerequisites for completing testing and providing the final approval

The information below applies to all iterations:

- All acceptance criteria have been validated, if unable to validate, these items have been identified to be added to the backlog and the discussion with the business has been completed on those items
- All open defects have been discussed with the Business/user
    - No Open Critical or High Defects
    - Business User acceptance of any open medium/low defects.
    - Workarounds needed for defects not closed (deferred).
- SQA test results will be discussed with the business users (100% test cases executed and/or passed, with remaining test cases that didn't pass being moved to backlog) upon business acceptance.

4.2. Automated Regression

Description	Regression testing provides a mechanism to ensure that any modified code has not caused defects or deviations from the previously existing functionality. This testing phase encompasses both manual and automated test suites. Regression normally supports two basic types of test runs: 1. Full Regression – All tests in the regression suite are run and analyzed for any possible regressions in the code. 2. Targeted Regression – Specific parts of the regression suite can be run in order to provide coverage for a particular change that is introduced and/or configuration changes to the environment (i.e. Flag configuration changes).
Test Environment	QA Environment (FPO Native App)
Target Test Schedule (Deliverables / Milestones)	Iteration/sprint 1: 18 Mar 2020 – 31 Mar 2020 Iteration/sprint 2: 01 Apr 2020 – 14 Apr 2020 Iteration/sprint 3: 15 Apr 2020 – 28 Apr 2020
Participants / Applications	SQA and Automation Team
Source of Test Data	Provided by PO
Entrance Criteria	• The code has entered SQA. • Release testing is complete and code freeze in place
Links to: Test script / Results / Test Cases	JIRA & GitHub
Exit Criteria	• All executed regression tests have been analyzed • All severity one and two defects have been resolved and closed. • All other defects (below severity two) have either been closed or added to the backlog and deferred to a later iteration/sprint.
Defect density	
Automation density	

4.3. Manual Regression (optional)

1. ## 5. Impacted Application – required

5.1. System Changed

Iteration/sprint 1:

App ID	FPO
App Name	Collins FPO
Type of testing	API, Functional and Integrated
App point of Contact	Lisa Mitchell & Anya Palkowski
Test Phases	QA
Gets & Gives Source of Data, Destination of Data, File Transmissions	Provided by PO/Dev

1. ## 6. Traceability Matrix

Source	**ID #**	**Source**	**ID #**
Name and Version number of the document where the solution design is established	*Individual Architecture identifier from the source document*	*Name and Version number of the document where the solution design is established*	*Individual Architecture identifier from the source document*
JIRA & Xray	Xray	JIRA URL Domain - Native App (Collins FPO) Project - GitHub Repo	Same as the left side column

1. ## 7. Vendor Requirements

Used as appropriate when Vendor(s) involved, and Package Test Plan not completed. This section may include the engagement agreement, acceptance criteria, process, and/or other relevant information for each vendor application.

N/A

1. ## 8. Testing Control Procedures

Required if not documented in the Test Strategy or Package Test Plan

Provide a description of the Testing Control Procedures. This may include.

- Planned communication/meetings – Daily stand up meetings to discuss iteration/sprint testing progress and other meetings to verify the progress of the project as well as when issues occur.
- Test status reporting – Weekly summary report to PM and Business product manager of iteration/sprint testing progress.
- Defect management (including prioritization scheme for defect tracking and resolution) – Create defects in JIRA using established SQA criteria for prioritization to resolution
- Procedures and tools for identifying, recording, and tracking test results through closure – SQA test plan will be updated after each iteration has been completed.
- Classification codes – N/A
- The approval process to be followed – approvals will be gathered for all iterations delivered for release code delivery and the same will be true once the project has been completed.
- Testing risks and issues management – These items will be documented at the project level based on the discussion with PM and Business product manager as to what issues are written in JIRA.

Reference(s) to documented procedure(s) in type-specific plans and other locations

1. ## 9. Test Tools and Repository

This section is required if a Test Strategy document or Package Test plan has not been completed. List the testing tools required for test execution and defect reporting; identify the test documentation repository.

9.1. Testing Tools

- JIRA for application defects management
- Collins FPO Native App for testing
- JIRA for user stories, sub-tasks, and issues

- Xray for test case management
- Appium Selenium API's Open source for Automation Testing
- ResAssured for backend testing
- Confluence, Swagger for Documentation
- Git/GitHub for Automation Repository
- Jenkins for CI Pipeline
- Traffic Parrot for gRPC Protocol Testing

9.2 Documentation Repository

- All user stories with Acceptance Criteria and result validation materials will be compiled and posted in the projects JIRA site: URL TBD
- All testing requirement documents, test scripts, and result validation materials will be compiled and posted in the Application Lifecycle Management tool – JIRA – plugin -TBD
- Document artifacts for all iterations and project will be posted – Confluence & Swagger Link: TBD

1. 10. Logical Day Processing Details

Required if testing involves Logical Day processing

Provide the logical day calendar if testing is dependent on the calendar. Identify the logical days in which the package will participate when applicable. Identify independent application processing schedule when applicable.

Logical day information will be added for each iteration/sprint when it requires logical day testing.

1. 11. Project Document Reference

DocType	Name & Location	Author	Version	Approved Date
Sprint 1	Link above	Anya Palkowski	Alpha	18 Mar 2020
Sprint 2	Link above	Anya Palkowski	Alpha	01 Apr 2020
Sprint 3	Link above	Anya Palkowski	Alpha	15 Apr 2020

Labels to be added into the User Story

- **Need_QA:** During the Refinement/Sprint Planning meeting, the Product Owner will decide the story needs to be testing or not? based on that decision product owner/Scrum master/TPM will add the Label on the JIRA user story.
- **NeedToAutomation:** During the Refinement/Sprint Planning meeting, QA Engineer should decide the user story will be automated or not? based on that decision product owner/Scrum master/TPM will add the Label on the JIRA user story.
- **NotToAutomate:** QA Engineer should decide the user story will be automated or not? If the Automation criteria don't meet the checklist based on that QA Engineer will add the Label on the JIRA user story.

Onboarding Checklist for SQA

PW Project (X)	Collins Corps Projects	DX Projects	Links
• PW ID • PW(X) Aero Laptop (MAC & Windows) • PW(X) JIRA ACCESS (Including Xray) • PW(X) Confluence access • PW(X)Staging & Production application access • PW(X) Swagger Docs access for API Automation • PW(X) GitHub access • PW(X) Database access • Liferay (cms) Framework access • Dash-A account (TPAM access)	• DX Corp Laptop (MAC & Windows) • DX Corp iPad/iPhone/Android • DX Corp JIRA ACCESS (Including Xray) • DX Corp Confluence access • DX Corp Staging & Production application access • DX Corp Swagger Docs access for API Automation • DX Corp GitHub access • DX Corp Database access	• DX Laptop (MAC & Windows) • DX iPad/iPhone/Android • DX JIRA ACCESS (Including Xray) • DX Confluence access • DX Staging & Production application access • DX Swagger Docs access for API Automation • DX GitHub access • DX Database access	TBD