



Quality Assurance Test Strategy

Upstart13: BCO

Prepared by:
Marcelo Garcia

Company Confidential



Table of Contents

1. Document Properties	3
2. Revision History	3
3. Related Documents	3
4. Key Terms and Definitions	4
5. Introduction	5
6. Test Process	6
7. Defect Management	8
8. Item Pass/Fail Criteria	11
9. Tools & Resources	12
10. QA Responsibilities	1
11. Documentation	2
12. Test Case Management	3
13. QA Reporting	3



1. Document Properties

Item	Details
Document Title	BCO Test plan
Author(s)	Marcelo Garcia
Creation Date	27/07/2022

2. Revision History

Date	Author(s)	Version	Change Reference

3. Related Documents

REFERENCE #	DOCUMENT NAME
01	
02	

Company Confidential

4. Key Terms and Definitions

TERM	DEFINITION
UP13	Upstart13.
Bug	See Defect.
Defect	An error which impairs or prevents the function of the product.
Task	A user story or job to be completed by a developer.
Enhancement	An improvement to an existing feature or task.
Issue	A generic name given to an item that requires resolution or attention such as Defect, Enhancement or Task.
Issue Workflow	The way an issue transitions on its way to a resolution across different steps, statuses, and team members.
Issue Owner	The person who discovered or created the defect.
SDLC	Software Development Life Cycle.
Testing Levels	Test Levels are used to split a testing phase into logical phases and set clear boundaries for each level of testing.
Testing Types	The Development process involves various types of testing. Each test type addresses a specific testing requirement or section within the product.
Smoke Testing	A quick test across major functions of the product. We ensure the build is stable enough for testing, key feature implementations are functioning as expected, and issues previously discovered have been fully resolved.
Regression Testing	A more detailed type of testing that focuses on ensuring updates have not broken any existing functionality. Its purpose is to catch defects that may have been introduced into the updated version of the application, and to ensure that previously eradicated defects remain resolved.
Usability Testing	This testing covers application flow, system navigation and other user interaction checks. The goal is to not only provide feedback on the application flow, but recommendations to make navigation and transitions easy to understand to the average User.
User Interface Testing	User Interface testing is a type of testing that focuses on the graphical mechanics of the application. It involves checking the screens and the controls such as menus, buttons, icons, toolbars, dialog boxes, windows, and others.

Company Confidential

Acceptance Testing	This type of testing is done to ensure the product meets specified requirements and quality standards. Testing is done by QA, Users and or Clients to determine product acceptance.
Functionality Testing	This type of testing concentrates on the output or front-end piece of the product. All testing is important and necessary, but ensuring all Functions work as intended or required by Client/Customer is top priority.
Integration Testing	The most important type of testing after Functionality Testing for Red Queen, since our main focus is ensuring integration between APIs goes smoothly. This type of testing verifies all API and functions transfer correctly during communication between the application and the sensors.

5. Introduction

PURPOSE

This document is a high-level presentation of the test approach to be taken in relation to the BCO-Upstart13 initiative. This document will be used as reference by QA for all testing activities.

This document will provide a good understanding of the following:

- The objectives for BCO test plan
- Test activities in different stages of the project life cycle
- The different test levels and test types
- Key testing processes and procedures
- Entry and exit criteria including reporting

TESTING OBJECTIVES

- Verify BCO's product integrity
- Validate and verify application functionality works to specified requirements.
- Provide confidence that test results meet quality standards, client requirements, needs and user expectations.

Company Confidential

6. Test Process

QA will work remarkably close with Development by running a series of Test Levels and Test Types to ensure a release candidate is packed with the core features and, most importantly, defect free.

The focus for QA is to be involved at an early stage of Development, implementing a strategy based on project requirements, timeframe, and available documentation. Defects are tracked and managed internally.

The aim is for QA and DEV to work together as a team. QA's role is to validate and verify functionality throughout development and integration; and provide feedback on results. Quality is a conjoint effort of all parties involved in the process, not one area alone.

- **Planning:** QA participates in planning meetings gathering information preparing Test Cases based on project requirements and all deliverables.
- **Backlog:** QA enters defects in the selected tracking tool(s) for proper resolution based on priorities and available resources.
- **Daily Stand-ups:** QA participates in daily stand-ups (if needed) providing support where appropriate, updating the team on test results, and updating test documentation as needed.
- **Reporting:** QA initially will circulate weekly results, and as Development and QA become more involved in the project, possibly daily updates. Results will be based on Test Cases run through the application based on the QA process.

TEST LEVELS & TEST TYPES

This section describes the relationship between different test levels and test types which need to be included as part of the QA process.

- Test Levels

Test Levels are used to split a testing phase into logical phases and set clear boundaries for each level of testing. The following Test Levels are:

- Smoke Testing

A quick test across major functions of the product. The goal here is to ensure the product is stable enough for testing, key feature implementations are functioning as expected, and issues previously discovered have been fully resolved.

- Regression Testing

A more detailed type of testing that focuses on ensuring updates or changes have not broken any existing functionality. Its purpose is to catch defects that may have been introduced into a new build or release candidate, and to ensure that previously eradicated defects remain resolved.

- Test Types

Test Type is about applying the appropriate type of testing method to verify specific requirements. We will concentrate on the following Test Types:

- Usability Testing

This testing covers application flow, system navigation and other user interaction checks. The goal is to ensure there are no progression stoppers throughout the product, and provide feedback not only on issues, but recommendations to make navigation and transitions easy to understand to the average User.

- Functionality Testing

This type of testing concentrates on the output or front-end piece of the product. All testing is important and necessary, but ensuring all Functions work as intended or required by Client/Customer is top priority.

- Acceptance Testing

This type of testing is done to ensure the product meets specified requirements and quality standards. Testing is done by QA, Users and or Clients to determine product acceptance.

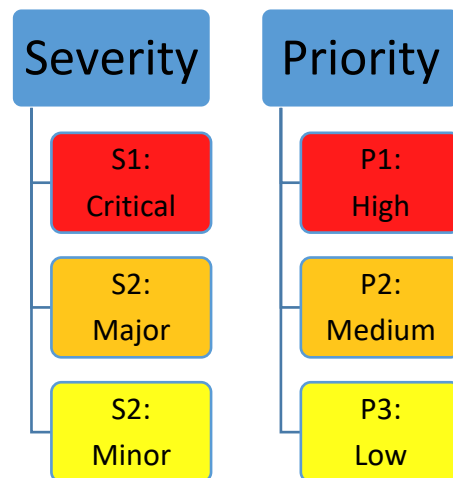
- Integration Testing

Perhaps the most important type of testing after Functionality Testing for BCO, since our main focus is ensuring Integration goes smoothly. This type of testing verifies all items and functions transfer correctly during integration.

7. Defect Management

Management and resolution of defects will be properly recorded and progressed using Jira. Concentrating all tickets in one place should grant the team a bird's eye view of the efforts and 24/7 knowledge of the current backlog of items

All Defects will be assessed to determine severity and priority based on many factors.



- Severity

Defect severity determines the defect's effect on the application. Severity is based on how important functionality is in the area affected and it relates to the quality standard.

- Priority

Defect priority signifies how important and urgent it is to fix this defect. The priority is typically set by the DEV Team based on the project priorities, capacity and velocity. Depending on the priority set, the defect may or may not be fixed with the next build release.

- **Severity Categories**

Severities are classified into various categories, and they can vary depending on company standards or project needs. We will use the following:

- **S1: Critical**

A defect that affects major components of the application, critical flows with no workaround, either functional or graphical components in which the application cannot recover from. Testing cannot be started until these types of defects are resolved. Critical issues need to be resolved as soon as possible.

- **S2: Major**

Major issues with main components, but there are workarounds and or areas of the application that can be tested. Major issues need to be fixed by the next day/build or sooner.

- **S3: Minor**

A small defect that does not affect major components of the application. Typically, small graphical glitches, minor functionality issues, or missing incorrect text/fonts and others.

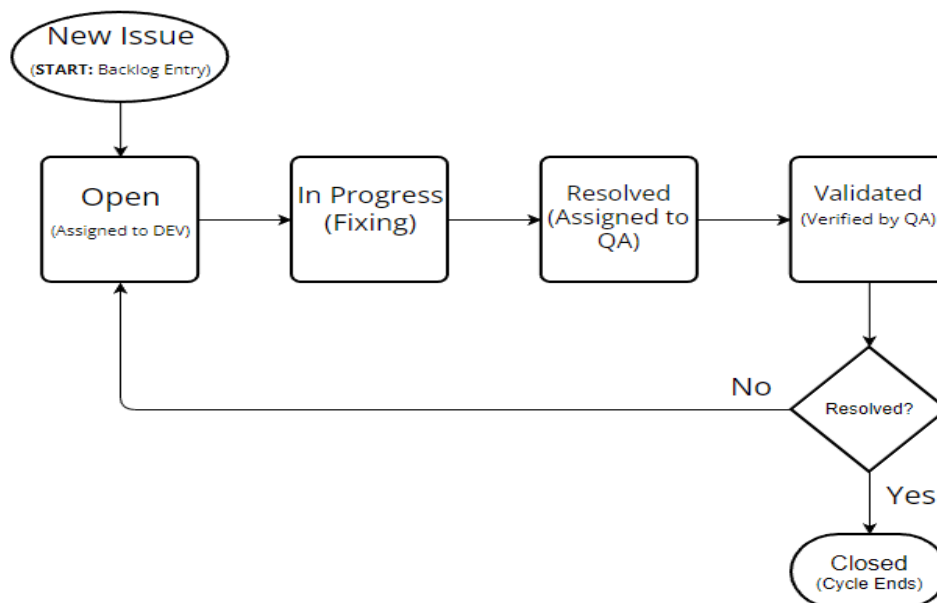
- **Priority Categories**

Company Confidential

As we mentioned earlier, a Priority category defines how soon an issue needs fixing. We will use the following priorities:

- **P1: High**
A defect that is assigned a high priority needs to be fixed right away, before anything else. A build cannot be released without this defect fixed first.
- **P2: Medium**
A defect assigned a medium priority is usually a major or minor severity defect that must be resolved with the next build release.
- **P3: Low**
When assigning a low priority to a defect, we are implying a fix can wait until all other higher priority defects/issues have been resolved. It can be resolved with the next build release if needed.

- Issue Life-Span Cycle



Company Confidential



A few important notes to keep in mind:

- *Defects are not the only items in the backlog that could be assigned a Severity or Priority. User Stories, Tasks, and Enhancements can also be assigned a level of resolution and importance.*
- *It is important to know that a defect, user story, task or enhancement at any point can change its Severity and Priority if the Project needs change.*

8. Item Pass/Fail Criteria

- Suspended
 - Any situation which impedes the ability to continue testing or value in performing testing leading to the suspension of testing activities.
- Resumed
 - Once the problem that caused the suspension has been resolved, testing activities can be resumed.
- Approved
 - An item will be considered approved once it meets the expected outcome defined in the corresponding Test Case.

Company Confidential



9. Tools & Resources

- JIRA

Web based bug tracking and task management tool that will be used to track Defects reported against a build version. It contains a workflow which transitions an issue from open to Resolved. Reports on progress and results will be provided to the customer.

- Test Environment

US13 will test all dev efforts in lower environments (Test/QA and STG/Staging) before giving a sign off for the code to be deployed to production

- Automation

US13 will create (as needed and as per their consideration) Automated tests (Scripts) to help ease out the testing effort for the BCO platform. US13 will implement the best tech stack suited for each task.

Company Confidential



- Test Data

Test Data will be created by Dev, Data Scientist or QA as needed to accurately test all areas of the application. Test Data will be addressed on a scenario-based need (if any).

US13 QA Responsibilities

- QA develops a Test Strategy (**this document**) based on application requirements, company needs and expectations and solution design documentation.
- QA attends Daily Standup Meetings to provide feedback to the team (as needed).
- QA creates and maintains documentation to be utilized in testing for each sprint.
 - Test Strategy (**this document**)
 - Test cases (**Jira**)
 - Test Results Report (**email**)
- QA participates in development & walkthrough meetings providing QA analysis when needed and gaining information needed for testing.
- QA tests features as they are implemented and made available in a new build.
- QA enters defects into the bug tracking system as they are discovered.
- QA runs through all Test Cases with each Test Level and provides feedback back to the team on results.
- QA verifies and validates tasks when marked resolved and updates statuses.
- QA ensures the necessary resources & tools are in place for proper testing and to avoid risks on deadlines.
- QA provides Test Results internally and creates Test Results reports
- QA provides a full Status Report at the end of each week with the following:
 - Defects resolved
 - Defects discovered
 - Test Levels and areas tested.
 - Outstanding or unresolved items.

Company Confidential



10. Documentation

QA will create QA documentation for the process as needed, that outlines processes, tools, resources, and others.

Test Strategy (**this document**) will only receive updates if major changes to deadlines, development cycle, or feature updates require it.

Test Checklist will constantly be updated with changes or addition of Test Cases with each development cycle. All documentation will be submitted to the Project Manager for validation and proper distribution.

The Test Results Report will be put together at the end of each week to update everyone on the testing status. New Issues, Issues Resolved, Blockers, and Process changes/updates are some of the items that will be covered in the report.

The following are documents that will be created and maintained as part of this project:

- Test Strategy

Test Strategy will contain definitions, introduction to the project, test process, defect management, pass/fail criteria, tools & resources, documentation, reporting and others. Test Strategy will only receive updates when major changes in development, requirements or process occur.

11. Test Case Management

There are two types of test cases:

1. Test Case to verify the given requirement (positive test case)
2. Test Case to break the requirement (negative test case)

A Test Case should have the following:

- Test Case ID
- Summary and Preconditions (to execute the test)
- Steps (unitary instructions to test the scenario)
- Expected result (what you are expecting out of your execution on each step)
- Actual result (what the output of the followed steps was)
- Section for Comments

A good test case should have the above points in a simple language that anybody can understand even without prior knowledge of under test.

12. QA Reporting

QA will issue a Test Results Report which will include the following:

- Testing status on build
- Test documentation updates/changes
- Testing Roadblocks
- Bugs
 - New Bugs
 - Closed Bugs
 - High priority open Bugs
- Others (Feature Coverage, Tech Debt)