Table of Contents

[Document Control 2](#_Toc75091458)

[Version History 2](#_Toc75091459)

[Document Review 3](#_Toc75091460)

[Distribution List 3](#_Toc75091461)

[Test Plan Sign Off 3](#_Toc75091462)

[Introduction 3](#_Toc75091463)

[Purpose: 3](#_Toc75091464)

[Test Overview 3](#_Toc75091465)

[Testing Scope (What will be tested): 4](#_Toc75091466)

[Testing out of scope (What won’t be tested): 4](#_Toc75091483)

[Assumptions: 4](#_Toc75091490)

[Test Approach (How it is tested): 4](#_Toc75091494)

[Story/Sprint Level: 4](#_Toc75091495)

[Done: Definition of done to be agreed with all the relevant stakeholders 5](#_Toc75091496)

[Plan: 5](#_Toc75091497)

[Environments: 5](#_Toc75091498)

[Entry and Exit Criteria: 5](#_Toc75091499)

[Risks 6](#_Toc75091500)

[Risks and mitigation plan: 6](#_Toc75091501)

[Reporting: 6](#_Toc75091502)

[Testing tools: 6](#_Toc75091503)

[Defects and Defect Tracking: 6](#_Toc75091504)

[Defect lifecycle and assigning process: 6](#_Toc75091505)

# Document Control

|  |  |
| --- | --- |
| Date | 20 June 2021 |
| Author | Ujjwal Kumar |
| Version Status | Draft |
| Document Location | G:\ |

## Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date Issued** | **Document Location** | **Comments** |
| Draft | *20/06/2021* | G:\ |  |
| V0.1 |  | G:\ |  |
|  |  |  |  |

## 

## Document Review

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date Reviewed** | **Department** | **Reviewed By** |
| Draft |  | *QA* | *QA Team* |
|  |  |  |  |

## 

## Distribution List

|  |  |  |
| --- | --- | --- |
| **Name** | **Title / Responsibility** | **Action** |
| Bunning Team | *QA team* | *Reviewer* |

## 

## Test Plan Sign Off

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Title / Position** | **Version** | **Date** |
| Bunning Team | *QA team* |  |  |
|  |  |  |  |

Introduction

## Purpose:

The purpose of this document is to outline the test plan/strategy and various testing activities that will be performed in the testing phase. The testing activity will be broadly categorised as

1. Manualtesting
2. Automation testing

Test Overview

Testing Scope (What will be tested):

The scope of the testing is the search functionality on the Bunnings Homepage.

**Manual testing**

As a part of manualtesting this individual search component needs to be test for below scenarios.

1. Verify search field has a text field and a search icon.
2. Verify user can provide an input to the search field.
3. Verify when user starts typing any search keyword then a drop-down area appears.
4. Verify based on user input the search results should be shown
5. Verify search field has a predictive search feature.
6. Verify user can select a suggestive search text on drop down.
7. Verify user can close the suggestive drop-down section.
8. Verify search functionality on cross browser and mobile devices

**Automation testing**

As a part of UI automation QA member will identify and automate scenario based on priority and business needs.

Below mentioned scenarios will be part of automation testing.

1. Verify when user starts typing the any search keyword then a drop-down area appears.
2. Verify based on user input the search results should be shown
3. Verify user can close the suggestive drop-down section.

Testing out of scope (What won’t be tested):

The following items are out of scope for testing:

1. Backend testing of search functionality
2. Any other component on the Bunnings Homepage
3. Automation testing for cross browser
4. Automation testing on mobile devices
5. Performance testing of search functionality

Assumptions:

1. No authentication is needed to test search functionality.
2. Automation testing to be done on chrome browser.
3. Random dummy test data to be used for searching.

# Test Approach (How it is tested):

## Story/Sprint Level:

The whole feature to be split into different stories spanning across different sprint. The Story will have an acceptance criterion which will act as exit criteria for the development and QA team for that story. Each story will have test cases based on the acceptance criteria. QA team to prepare test cases and test data for each story and then execute the test case. Based on test case result, defect will be raised to track. (Refer to Defect Lifecycle in section [Defect Tracking](#_Defects_and_Defect))

## Done: Definition of done to be agreed with all the relevant stakeholders

## Plan:

The QA Plan will consist of following activity in three Phases:

Phase 1:

1. Test Plan Document Preparation and review: A test plan is a detailed document that outlines the test strategy, testing objectives, resources required for testing and test deliverables.
2. Test Scenarios, Test Data and Test Case Preparation: Once the test plan has been finalized and approved, the second part of phase 1 will be Identifying test scenarios and creating test cases and test data based on the scenarios.

Phase 2:

1. Test Cases execution: As a part of second phase QA team will create a Test Cycle and execute the test cases relevant to the test cycle.
2. Defect Management: QA team to identify and report defects whenever there is any deviation from the acceptance criteria. QA team will follow the below mentioned defect lifecycle process for handling the defects. The severity of the defect will be set by the QA team and the prioritization will be set based on discussion with product owners.

Phase 3:

1. Test Reporting: QA team to prepare and share the test case execution and defect Summary report with the relevant stakeholders.
2. QA sign off: Once all the acceptance criteria are met and no outstanding issues are Open/pending, QA team will provide a final sign off on the product.

## Environments:

QA activities will be performed live site.

## Entry and Exit Criteria:

**The entry criteria for QA activity will be as below**

1. Requirement document approved and signed off.
2. Acceptance criteria approved by the BA.
3. Test Plan document ready and approved
4. Test Cases ready and approved
5. Unit test coverage of at least 80% of the code.
6. Test Data prepared and ready
7. Environments are configured and set-up for QA
8. Testing notes to be provided by Dev team.

**The following would be exit criteria for QA**

1. All testing activities mentioned in this document to be completed.
2. All test cases executed and passed.
3. All defects tested and verified.

# Risks

## Risks and mitigation plan:

Testing on live site without any notice to the team.

Mitigation: Since this is a dummy test no real users will be impacted.

# Reporting:

1. **Test Summary report:** Automation test suite to generate a html report of all the test scenarios

# Testing tools:

1. **Selenium:** Automation testing to be done on selenium testing framework
2. **Maven:** To build and manage project QA to use maven build tool
3. **Java:** Programming language to be java
4. **IDE:** Eclipse Ide to be used.
5. **Cucumber:** Automated testing to be done in BDD. So cucumber to be used.

# Defects and Defect Tracking:

Defects will be created as a separate ticket and will be linked to the story in which the defect is found. The story which has a defect in open state will be marked as blocked. The priority and severity of the defect will be discussed along with the scrum master or producer.

## Defect lifecycle and assigning process:

The defect lifecycle will be as below.

* New**:** When a new defect is logged and posted for the first time. It is assigned a status as NEW.
* Assigned**:** Once the bug is posted by the tester, the test lead or producer approves the bug and assigns the bug to the developer team.
* Open: The developer starts analysing and works on the defect fix
* Fixed: When a developer makes a necessary code change and verifies the change, he or she can make bug status as "Fixed."
* Retest: Tester does the retesting of the code at this stage to check whether the defect is fixed by the developer or not and changes the status to "Re-test."
* Reopen: If the bug persists even after the developer has fixed the bug, the tester changes the status to "reopened". Once again, the bug goes through the life cycle.
* Closed: If the bug is no longer exists then tester assigns the status "Closed."
* Duplicate: If the defect is repeated twice or the defect corresponds to the same concept of the bug, the status is changed to "duplicate."
* Not a bug: If it does not affect the functionality of the application then the status assigned to a bug is "Not a bug".

Diagram

Description automatically generated