ALEKSANDRA TUMANENKO

Canny Store

Master Test Plan

# Contents

[Contents 1](#_Toc9269287)

[1. Introduction 3](#_Toc9269288)

[2. References 4](#_Toc9269289)

[3. Testing Areas 5](#_Toc9269290)

[3.1. Features to be tested 5](#_Toc9269291)

[3.2. Features Not to be tested 5](#_Toc9269292)

[4. Test Strategy/Test Approach 6](#_Toc9269293)

[4.1. Test Levels 6](#_Toc9269294)

[4.2. Testing in development process 6](#_Toc9269295)

[5. Quality metrics 8](#_Toc9269296)

[5.1. Priority definition 8](#_Toc9269297)

[5.2. Item Pass/Fail Criteria 8](#_Toc9269298)

[6. Test Environment and Tools 10](#_Toc9269299)

[7. Testing exit criteria 11](#_Toc9269300)

[8. Deliverables 12](#_Toc9269301)

**Revision history**

| Date | Version | Description | Author of changes |
| --- | --- | --- | --- |
| 11/04/2019 | v1.0 | Initial document | Aleksandra Tumanenko |
| 15/02/2021 | V2.0 | References  Testing Areas  Testing exit criteria Test Environment and Tools | Nadezhda Kuzmina |
|  |  |  |  |
|  |  |  |  |

# Introduction

The purpose of this document is to establish a plan for the activities that will verify Canny Storeunder test as a high-quality product. The document will serve as a guide for all testing activities and will highlight the following:

* Test Strategy/Test Approach
* Test Methodology
* Test environment
* Acceptance criteria
* Roles and Responsibilities

We believe that the quality assurance team possesses all the qualities required for successful project development and implementation.

# References

**Table 1 - List of referenced documents**

| Code | Document | Location |
| --- | --- | --- |
| 0101 | specification | <https://docs.google.com/document/d/1ARIVJB8cegXgbr_J18XoByc_8k-jcc8v/edit?rtpof=true> |

# Testing Areas

## Features to be tested

Main menu:

1. functionality should be tested be able to reach the website <https://kungfuqa.com/store_fixed>
2. the presence of sections: about us, furniture, accessories, social networks, exhibition

Furniture and accessories:

1. information about items
2. change the number of items in the basket or remove them
3. buy items with a sale

Process with paments:

1. the required fields for registration
2. choose a payment method

Exhbition:

1. general information about the exhibition
2. registration for the exhibition and designation of the location on the map

Newsletter:

1. Newsletter subscription form

Social media links:

1. following a link to social networks

## Features Not to be tested

* LigPay payment and WayForPay
* Work of social link Facebook, Twitter, Pintere

# Test Strategy/Test Approach

Agile approach, takes QA team and development team as one team. Thus, they all have the same goal that is developing a top quality product that fulfils its users’ needs. Tasks carried out in a highly collaborative manner by teams that embrace and adapt changes to address customer's needs. This approach allows QA team efficiently collaborate with the development team and produce great results.

## Test Levels

There will be four levels of tests applied: unit testing, integration testing, system testing and acceptance testing.

**Unit testing**is a level of software testing where individual units/ components of a software are tested. The purpose is to validate that each unit of the software performs as designed.

**Integration testing**is a level of software testing where individual units are combined and tested as a group. The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

**System testing**, or end-to-end testing, verifies that a completely integrated system meets the requirements.

**Acceptance testing** performed by the Client to certify the system with respect to the requirements that were agreed upon. The main purpose of this testing is to validate the End-to-End business flow.

## Testing in development process

**Smoke testing** – type of software testing which ensures that the major functionalities of the application are working fine. It is also known as Build verification testing where the build is verified by testing the important features of the application and then declaring it as good to go for further detailed testing.

**New feature testing** – testing process exercised to check whether the new feature is worked correct like mentioned in specification for this feature. This phase goes in parallel with bug fix verification.

**Regression testing**Regression Testing is defined as a type of software testing to confirm that a recent program or code change has not adversely affected existing features. Regression testing performed before each release to production.

# Quality metrics

## Priority definition

|  |  |
| --- | --- |
| **Priority** | **Description** |
| **P1** | Critical business issue.  This is a failure that is preventing either large or core areas of the Product from functioning. This defect needs to be fixed and deployed as soon as possible. |
| **P2** | Significant business issue.  This is a serious failure that must be fixed before going live. Defects badly affect core areas, causing important or highly visible areas to fail. |
| **P3** | Limited business issue  A failure that is causing an error in the application’s functionality. It is of lower impact or in a less visible area than a P2 failure. Defects should be fixed before going live. Normally has a workaround. They usually impact only a few test cases and will not stop QA testing, and staying on schedule. |
| **P4** | Minimal business issue.  A failure with minimal implications for the users of the applications. Most users will never notice it. May be minor cosmetic issues and may not prevent any test case from completing. It may be acceptable to mark a test case as having passed even with trivial defects against it |

## Item Pass/Fail Criteria

This set of criteria will be used to determine whether test item has passed or failed the test:

| Test results | Description | Action |
| --- | --- | --- |
| Passed | Test case has met the “Expected result” | Pass record in TestRail |
| Failed | Test case has not met “Expected result” | Failed record in TestRail |
| Blocked | Test case cannot be executed at the moment | Blocked record in TestRail |
| Not tested | Test case wasn’t tested yet. | Not tested record in TestRail |

# Test Environment and Tools

This section describes an environment, which will be used on the project, tools required for testing and list of browsers under the test.

**Table 3 - List of tools to be used**

| Tool | Tool name | Note |
| --- | --- | --- |
| Bug Tracking System | Trello | https://trello.com/b/IWCyEWS5/team-2 |
| Project document repository | Google Docs |  |
| Database administration tool | CRM | https://crm.tilda.cc/ |
| Test Management Tool | Test Rail | <https://alicekirsa.testrail.io/index.php?/suites/view/1&group_by=cases:estimate&group_order=asc> |
| Browsers | Chrome | Версия 88.0.4324.146 (Официальная сборка), (64 бит) |
| API manual tests tools | Postman,  Fiddler. |  |

# Testing exit criteria

**Exit criteriais as follows:**

* **P1 shouldn't be bugs**
* **P2** **shouldn't be bugs**
* **P3 up to 5 bugs allowed**
* **P4** **up to 10 bugs allowed**

# Deliverables

This section will describe the following deliverables:

* Test Plan (this document itself)
* Test Cases
* Bug reports
* Release notes
* QA weekly reports