Craving! Web Application

System Test Plan Document

Project: Craving! Web Application

Author(s): Akshay Pesari, Irina Sachovska, Midhuna Manchi, Onyekachi Ohiaeri, Shubham Trivedi, Tejashri Parurkar

Date: 11/04/2020

Table of Contents

[**Introduction**](#_heading=h.4d34og8) **3**

[**Testing Scope**](#_heading=h.2s8eyo1) **3**

[**Testing Objectives**](#_heading=h.3rdcrjn) **3**

[**Test Process Definition**](#_heading=h.lnxbz9) **5**

[**Approach To System Testing**](#_heading=h.35nkun2) **6**

[**Entry/Exit Criteria**](#_heading=h.x2oiyqrcc0c0) **6**

[**System Test Environment**](#_heading=h.xvyw9zsjce9x) **8**

[**Roles & Responsibilities**](#_heading=h.y9qe6iytpanz) **8**

[**Test Cycles & Schedule**](#_heading=h.f1a19ptfcgpd) **9**

[**Risks & Contingencies**](#_heading=h.hc3lg1vdbbt) **9**

# **Introduction**

The purpose of this system test plan document is to provide details on how the testing process will be conducted for a Craving! application project and establish the common understanding among the project stakeholders about the objectives, project scope and an approach to perform the system testing. Furthermore, the document also provides the details about the environmental needs, testing entry/exit criteria, test schedule and cycles, roles and responsibilities, and risks and contingencies.

# **Testing Scope**

The testing scope includes two perspectives - the functional scope and technical scope.

The *functional scope* includes the following modules of the “Craving!” system:

* 01. Authentication
* 03. Subscription Management

The Authentication module consists of the features such as:

* 01.01. Login
* 01.02. SignUp
* 01.03. Restore Credentials

The Subscription Management module consists of the following features:

* 03.01. Select/Pause Plan
* 03.03. Remove/Add products in box

The *technical scope* includes the following architectural components:

* Web Browser
* Web server
* Application server
* Database server

# **Testing Objectives**

The objective of the testing is finding as many software defects as possible; ensure that the software under test is bug free before release. The overall goal and achievement of the test execution is to define the test objectives, such as list all the software features (functionality, performance, GUI…) which may need to be tested and define the target or the goal of the test based on those features. The system test cases should include negative and challenging testing conditions in order to be effective in finding software defects.

Following list of documents will be serving as the basis for developing the functional and non-functional tests and evaluating the system functionality.

* Business Requirements Document (BRD)
* User Stories (Functional Requirements)
* Requirements Traceability Matrix (RTM)
* Requirements Composition Table (RCT)

**Core Features to be tested**

**Authentication Module** -

* Login feature
* Sign up Feature

**Subscription Management Module** -

* Remove/add products in the box
* Select/Pause plan - partial test (required for the remove/add products in the box functionality test.

In addition to the above core features being tested, the testing will also cover the crosscutting concerns applicable to the context of individual core features, for more details please refer to the requirements composition table.

**Non-Functional Features to be tested**

* **Portability testing** - validate that different most popular browsers, can support the functionalities being tested.
* **Extreme layout testing** - validate that any changes done to the computer system display settings, such as resolution changes, font updates, has no negative affect the application’s usability;

**Features not to be tested**

System performance will not be tested due to the lack of required tools. In addition the volume testing, such as a large number of customers accessing the system at the same time will not be tested. Usability and security will not be tested.

# **Test Process Definition**

Conventional test process lifecycle will be followed, there are **five phases** included in the process, including:

* **Test Planning** - the primary purpose of this phase is to specify the scope and objectives of testing, the roles and the responsibilities as well as determine the testing approach to the system testing.
* **Test Design** - the primary purpose of this phase is to define the test design logic, define the test specifications, and specify the requirements of the test data.
* **Test Preparation** - the primary purpose of this phase is to set up the test environment, install all the necessary software to perform the testing in the UAT environment.
* **Test Execution** - the primary purpose of this phase is to execute all the defined test cases, discover all the application defects, and report the bugs, as well as evaluate the system stability by validating all the features defined in the system test plan document.
* **Test Reporting** - the primary purpose of this phase is for testers to report the defect metrics, generate test execution status reports, evaluate the test exit criterion in the test completion report, and provide the stakeholders with the updates and visibility into the testing execution progress and results. Test completion report approval will serve as the basis for the system testing sign off.

Each test process phase includes specific **tasks** -

* In the test planning phase the tasks are to define the scope, objectives and the approach to the system testing, as well as assign the roles and responsibilities.
* In the test design phase the tasks are to set up a test management system, develop the test case specifications, specify the required test data and define the approach to designing test cases.
* In the test preparation phase the tasks include the set up of the UAT environment, install the necessary software in the QA environment, set up the defect tracking system and provision the test data.
* In the test execution phase the tasks are to execute and validate all the defined test cases, discover the defects if any, and report the uncovered defects.
* In the test reporting phase the tasks are to summarize and report the test execution progress, report on the defect metrics, evaluate the test exit criterion, create and submit to stakeholders the test completion report for approval, and receive the signoff on the system testing from the appropriate stakeholders.

Each test phase has the the following **deliverables** specified below:

* System Test Plan document is produced during the planning phase
* Test design specifications and test case specifications are produced during the test design phase.
* Readiness of the UAT environment with loaded test data and readiness of the defect tracking system are delivered during the test preparation phase.
* Software defects report and test execution log report are produced during the test execution phase.
* Test execution status report, defect report and completion report, - all are produced during the test reporting phase.

# **Approach To System Testing**

**Approach to Functional Testing**

The overall approach to functional testing of the Craving! application will be performed on the Black-box techniques. The functional specifications and business rules will be used as a base to design test conditions. Test execution will be done manually, from the user perspective and based on formal test case specifications, considering the system as a black box, inputting the data and evaluating the results via the user interface. The test execution results will be captured and reported in test execution logs.

**Approach to Non-Functional Testing**

Non-functional test objectives specified above will be tested using the black-box approach, from the user’s perspective. The portability testing will be performed by testing all the features using different browsers. Extreme layout testing will cover all implemented application’s functions and validate that functionalities are working properly under selected conditions.

# **Entry/Exit Criteria**

**Entry Criterion** - used to determine when a given test activity should start. It also includes the beginning of a level of testing, when test design or when test execution is ready to start, it includes the following conditions:

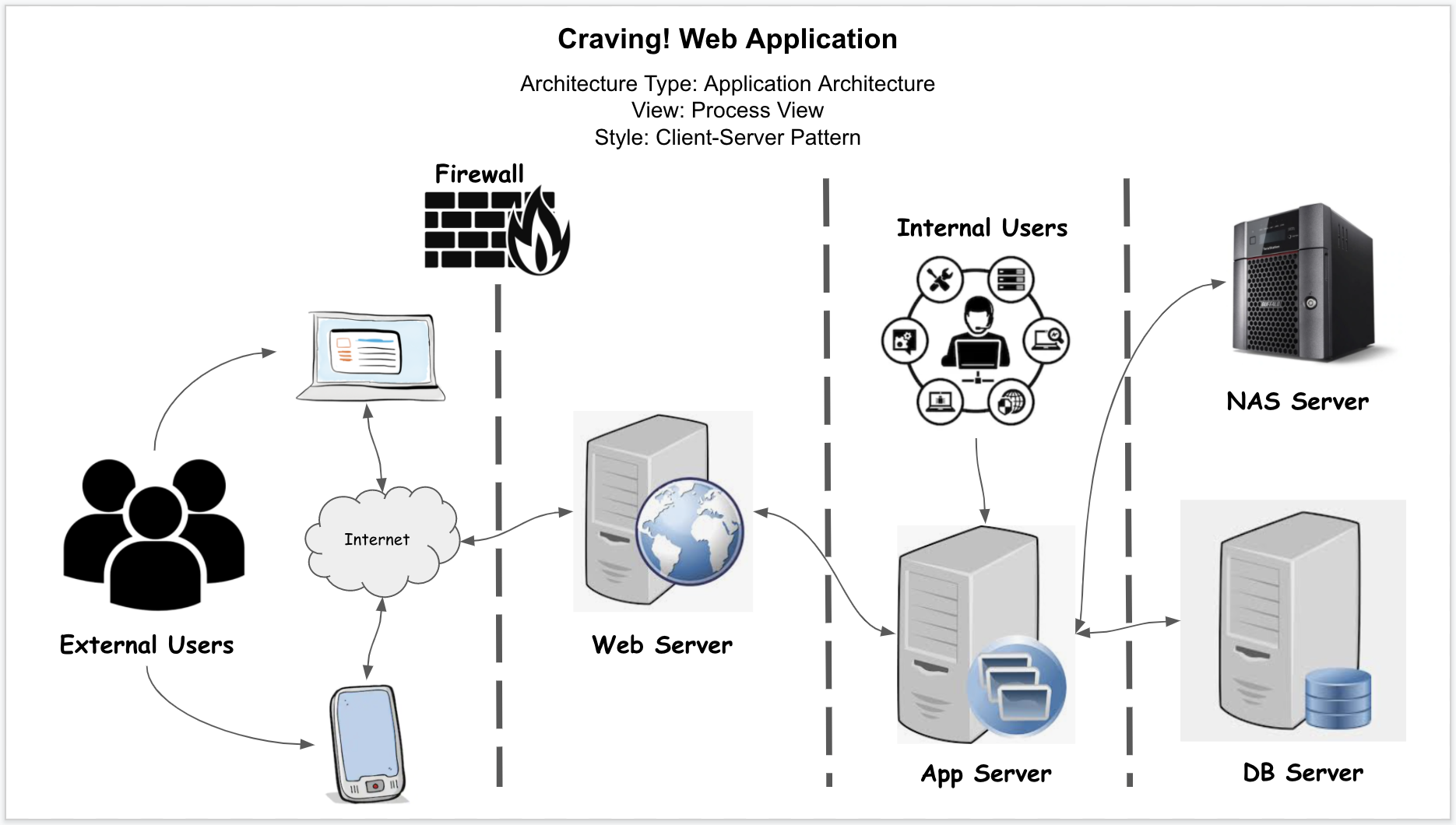
* Verify if the Test environment is available and ready for use.
* Verify if test tools installed in the environment are ready for use.
* Verify the testers have the access to the test environment.
* Verify Test Designs and test case specifications are completed.
* Verify if Testable code is available.
* Verify if Test Data is available and validated for correctness of Data.

**Exit Criterion** - evaluates the conditions that are necessary to conclude that testers can stop test execution and the system is ready for the final user acceptance testing, it includes the following conditions:

* Verify if All tests planned have been run.
* Verify if the level of requirement coverage has been met.
* Verify if there are NO Critical or high severity defects that are left outstanding.
* Verify if all high risk areas are completely tested.
* Verify open defects of medium and low severity have known work-arounds.
* Verify if software development activities are completed within the projected timelines.
* Verify QA/Testing sign offs have been provided.

# **System Test Environment**

The Test Environment must be available to start test execution. It includes a laptop with a virtual machine running the web, app servers and database, and internet browsers (Chrome, Firefox, Internet Explorer and Safari) to access the application. The architecture of the test environment is shown below.



# **Roles & Responsibilities**

The project team has six members that are assigned to the various project roles and their responsibilities are defined in the below table:

|  |  |  |
| --- | --- | --- |
| Role | Name | Responsibilities |
| Project Manager | Irina Sachovska | Responsible for the overall project timeline by tracking the testing schedule and results, review and approvals of the system test plan and test design specifications, overseeing the test environment preparation, and escalation of the issues. |
| Product Owner | Shubham Trivedi | Contributing to the test plan and test case specifications. Reviewing the test results. |
| Lead Business Analyst | Onyekachi Ohiaeri | Contributing to the test plan and test case specifications. Reviewing the test results. |
| Lead Developer | Midhuna Manchi | Responsible for producing a working software build, establishing and maintaining the test environment, communicating release notes, investigating and fixing software defects and assisting a QA throughout the testing process. |
| DBA | Tejashri Parurkar | Assisting the lead developer and QA in establishing and maintaining the test environment(s). |
| QA/Tester | Akshay Pesari | Responsible for procurement and support of the QA environment. Designing a test plan, establishing a test repository, developing test case specifications, overseeing the test execution, reporting defects, conducting the defect calls, providing the metrics and reports. |

# **Test Cycles & Schedule**

There will be three test cycles executed, with the first two aligned with the implemented application modules and the third one concentrating on the re-tests.

* **Cycle I** - will be primarily focused on the testing of the implemented parts of the Authentication module;
* **Cycle II** - will be primarily focused on the testing of the implemented parts of the Subscription Management Module
* **Cycle III** - will be focused on the regression testing and re-testing of the fixes of the uncovered bugs and issues.

Refer to the project plan for a detailed schedule of the test execution cycles.

# **Risks & Contingencies**

Potential risks and contingencies that might be encountered during the system testing include the following:

* Any changes to the implementation scope or last minute changes to the functional requirements can cause the significant delays to the implementation schedule.
* Insufficient testing resources can cause a delay as a result of an additional time required to complete the test case specifications.
* Unpredicted number of discovered defects may lead to extra time needed to rectify found issues and lead to the delays.
* Unforeseen issues in the test environment can cause consequential delays and impact the test execution schedule