Text

Description automatically generated

“GET EMPLOYED”

SYSTEM TEST PLAN

Author: Athauda Dias (Dayan)

Project Manager: Sanket Bunage

Date: 11/10/2020

Table of Contents

[INTRODUCTION 3](#_Toc485964763)

[1. TESTING SCOPE 3](#_Toc485964764)

[2. TESTING OBJECTIVES 3](#_Toc485964765)

[2.1 Core Features to be Tested 4](#_Toc485964766)

[2.2 Non-Functional Features to be Tested 4](#_Toc485964767)

[2.3 Features not to be Tested 5](#_Toc485964768)

[3. TEST PROCESS DEFINITION 5](#_Toc485964769)

[3.1 Test Process Phases 5](#_Toc485964770)

[3.2 Deliverables 6](#_Toc485964771)

[4. APPROACH TO SYSTEM TESTING 7](#_Toc485964772)

[4.1 Approach to Functional Testing 7](#_Toc485964773)

[4.2 Approach to Non-Functional Testing 7](#_Toc485964774)

[5. ENTRY/EXIT CRITERIA 7](#_Toc485964775)

[6. SYSTEM TEST ENVIRONMENT 8](#_Toc485964776)

[7. ROLES AND RESPONSIBILITIES 8](#_Toc485964777)

[8. TEST CYCLES AND SCHEDULE 9](#_Toc485964778)

[9. RISKS AND CONTINGENCIES 9](#_Toc485964779)

# INTRODUCTION

This document describes the plan for testing the architecture of “JOBSTER” web application and to establish a common understanding among the core stakeholders about the scope, objectives and approach to performing system testing.

Further this document will support objectives such as, identify existing project information and the software that should be tested, list the recommended test requirements, recommend and describe the testing strategies to be employed, identify the required resources and provide an estimate of the test efforts and list the deliverable elements of the test activities.

# 1. TESTING SCOPE

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

The functional scope includes the following modules in the system:

* User Experience
* User Login/Signup
* Partner Login/Signup.

The technical scope includes the following architectural components:

* Web browser
* Web Server
* Application server
* Database server

# 2. TESTING OBJECTIVES

The main objective of system testing is to validate the implementation of the system features for compliance with their functional and non-functional requirements. The system test cases should include negative, i.e., challenging testing conditions in order to be effective in finding software defects.

This section describes the features to be tested and the features that will be out of testing scope.

The list of project documents that will be used as a basis for designing test cases includes:

* Business Requirement Document (BRD)
* Functional Requirements
* Requirements Composition Table (RCT)
* Database Design

## 2.1 Core Features to be Tested

Home Module

* User portal
* Test whether a user can successfully create an account by signing up and whether current users can login-in with their credentials.
* Partner portal
* Test whether a partner can successfully create an account by signing up and whether current partner can login-in with their credentials.
* Home search bar
* Test whether both input fields in search bar contains text input before submitting.

Help Desk

* Contact form
* Test whether a contact form can generate a ticket and display the created ticket.
* FAQ page
* Whether FAQs are displayed when requested by users.

In addition to the above core features, testing will cover crosscutting concerns applicable to the context of individual core features (Please refer RCT)

## 2.2 Non-Functional Features to be Tested

The System Test includes the following objectives to test non-functional requirements:

* **Portability testing** is used to validate that browsers, Mozilla Firefox and Google Chrome, can be equally used by customers;
* **Layout testing** validates that changing window display settings do not affect the system’s usability. Eg: High or low resolution | Changing screen size (Zoom in/out).

## 2.3 Features not to be Tested

Search bar does not have its full functionality at this stage of development because it’s dependencies are not yet fulfilled in this semester.

# 3. TEST PROCESS DEFINITION

## 3.1 Test Process Phases

The test process consists of five phases,

* Test Planning
  + Define scope and objectives of testing
  + Define roles and responsibilities
  + Define testing approach
* Test Design
  + Identify test ideas, define an approach to designing test cases
  + Develop test case specifications
  + Measure test coverage
  + Determine requirements for test data
* Test Preparation
  + Setup a test environment
  + Provision test data
  + Install the software in the test environment
* Test Execution
  + Execute all test cases
  + Find and report software defects
  + Evaluate the system stability
  + Validate all target features
* Test Reporting
  + Summarize and report the test execution results
  + Report defect metrics
  + Evaluate the test exit criteria
  + Create a test completion report, submit for stakeholder approval
  + Obtain stakeholder signoff on system testing

## 3.2 Deliverables

On this project, the test process deliverables include:

* System Test Plan document
* Test Design specifications
* Test Case specifications
* Software Defects
* Test Execution Logs
* Test Completion Report

# 4. APPROACH TO SYSTEM TESTING

## 4.1 Approach to Functional Testing

The System Test will be performed based on the black-box techniques. This means, first, that the external functional specifications or business rules will be used as a primary source to design test conditions. Secondly, testing will be executed from the user perspective, i.e., considering the system as a black box and entering input data and evaluating results via the user interface.

The system features identified above can be classified by the following types of business logic – GUI, Field Edits, Field Dependencies, and General Business Rules. Each type can have its own test logic that can be reused across the system. Test conditions can be designed using conventional techniques, such as boundary analysis, equivalence partitioning, decision tables, etc. The detailed test logic for each pattern of business rules will be described in the test design specification.

## 4.2 Approach to Non-Functional Testing

All non-functional test objectives specified above can be tested using the black-box approach, i.e. from the user perspective.

Platform compatibility will also be taken into consideration as to see whether there are any layout mismatches when viewed on different browsers.

The portability and extreme layout tests should cover all functions (menu options) of the system and validate that each function works under the specified test conditions.

# 5. ENTRY/EXIT CRITERIA

The **Test Entry criteria** is used to formally evaluate the conditions necessary to begin test execution, it includes the following conditions:

* The application build is produced and deployed to the test environment
* System Test Plan document has been approved
* QA environment is ready
* QA team members have access to the QA environment
* Test case specifications have been completed and reviewed
* Release Notes document has been sent to the QA team

The **Test Exit criteria** is used to evaluate the conditions 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:

* All QA test cases have been executed
* Zero defects of Critical and Hi-severity remain open
* Open defects of Medium and Low severity have known work-around
* A Test Completion Report has been produced and communicated to stakeholders
* QA testing sign-off has been provided

# 6. SYSTEM TEST ENVIRONMENT

The Test Environment should be available to start test execution. It includes a laptop with virtual machine running the web server and database, and internet browsers (Chrome and Firefox) to access the application. (Please refer Jobster Architecture Diagram)

# 7. ROLES AND RESPONSIBILITIES

The project roles involved in system testing include the following:

|  |  |
| --- | --- |
| **Project Role** | **Role Responsibilities** |
| Project Manager | Responsible for the overall project timelines, review and approval of the System Test Plan, escalation of issues. |
| Product Owner and QA Environment Manager | Responsible for support of the QA environment.  Contributing to the test plan and test case specifications. Reviewing test results. |
| Lead QA Analyst | Designing a test plan, establishing a test repository, developing test case specifications, executing testing and reporting defects. |
| Lead Developer | Responsible for producing a working software build, build migration to the QA environment, communicating release notes, investigating and fixing software defects. |
| Lead DBA | Establishing and maintaining the test environment.  Assisting Lead QA Analyst throughout the testing process. |
| Lead Business Analyst | Contributing to the test plan and test case specifications. Reviewing test results. |

# 8. TEST CYCLES AND SCHEDULE

System testing will be executed in three cycles:

* Cycle 1

Focuses on Home module. Login and sign-up validation and Home Search Bar field validation.

* Cycle 2

Focuses on Help Desk module. Creating a Contact Form and accessing FAQ page.

# 9. RISKS AND CONTINGENCIES

* A lack of testing resources can result in more time needed to complete test case specifications.
* Changes to the implementation scope or existing functional requirements can impact the test execution schedule.
* Too many defects can delay the completion of test execution.
* Instability of the test environment can impact the test execution schedule.