CS691 – Computer Science, Spring 2021

Pace University



SYSTEM TEST PLAN

SERVICE GENERAL

# 

# TABLE OF CONTENTS

[**TABLE OF CONTENTS**](#_heading=h.feitme7v1209) **2**

[**INTRODUCTION**](#_heading=h.9z7sgmqz2zim) **3**

[**TESTING SCOPE**](#_heading=h.3znysh7) **3**

[**Testing Objectives**](#_heading=h.6jv9oluc7cyj) **3**

[Core Features to be Tested](#_heading=h.5iwwp7s413p8) 4

[Features not to be Tested](#_heading=h.2s8eyo1) 6

[**TEST PROCESS DEFINITION**](#_heading=h.e85a5hbsse2q) **6**

[Test Process Phases and Tasks](#_heading=h.26in1rg) 6

[Deliverables](#_heading=h.35nkun2) 7

[**APPROACH TO SYSTEM TESTING**](#_heading=h.1ksv4uv) **7**

[Approach to Functional Testing](#_heading=h.2jxsxqh) 7

[**ENTRY/EXIT CRITERIA**](#_heading=h.lht0jls1ln0d) **8**

[**ENVIRONMENTAL NEEDS**](#_heading=h.w5h1uftcf22a) **8**

[**ROLES AND RESPONSIBILITIES**](#_heading=h.iv0sjy6v0wxs) **10**

[**TEST CYCLES AND SCHEDULE**](#_heading=h.3as4poj) **11**

[Cycle 1 - User Experience](#_heading=h.i6sbzp7u9jgz) 11

[Cycle 2 - User Register/Login](#_heading=h.6lrs24sk3ogp) 11

[Cycle 3 - User Register Job](#_heading=h.si5h4d22on09) 11

[Cycle 4 - User Feedback](#_heading=h.79cflnvpvotu) 11

[Cycle 5 - Full (Rest) User Functionality](#_heading=h.uwxt9rys8xp7) 11

[**RISKS AND CONTINGENCIES**](#_heading=h.u1fdqqy48ad) **11**

# 

# INTRODUCTION

This document describes the System Test Plan that provides a common understanding among the “Service General” project stakeholders on the scope, objectives, and approach to performing the system testing. Also, the document explains the features to be tested, testing entry/exit criteria, resource and responsibilities, and testing schedule.

# TESTING SCOPE

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

The functional scope includes the following modules of the “Service General” system:

* Create an Account - 1.1
* Sign into Account - 1.2
* Register Job Service - 4.1
* Create Feedback - 7.1

The technical scope includes the following modules of the “Service General” system:

* Database
* Servers (Deployment/QA/Demo)
* Internet Browsers
* Mobile Devices

# Testing Objectives

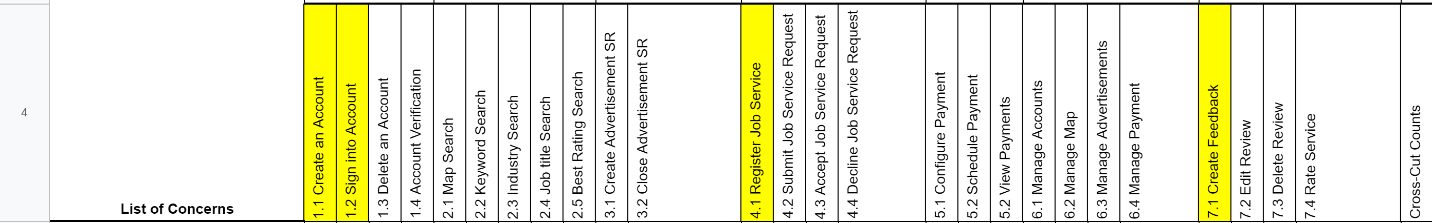
The main objective of system testing is to validate the implementation of the system features for compliance with their functional and non-functional requirements. The system test cases should include negative, i.e., challenging testing conditions to be effective in finding software defects.  This section describes the features to be tested and the features that will be out of testing scope. The focus of this Test Plan helps in finalizing the software application or product against business and user requirements.

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

* Business Requirements Document (BRD)
* "User Stories" functional requirements
* Requirements Composition Table (RCT)
* Database design and data dictionary specifications

## Core Features to be Tested

This section lists all core features that will be tested:



User Account module

* Create an account

To test whether the user can create a new account to use Service General.

* Sign into an account

To test whether a user can login into their existing account.

* Register for Job Service

To test if a user can register for a job service posted.

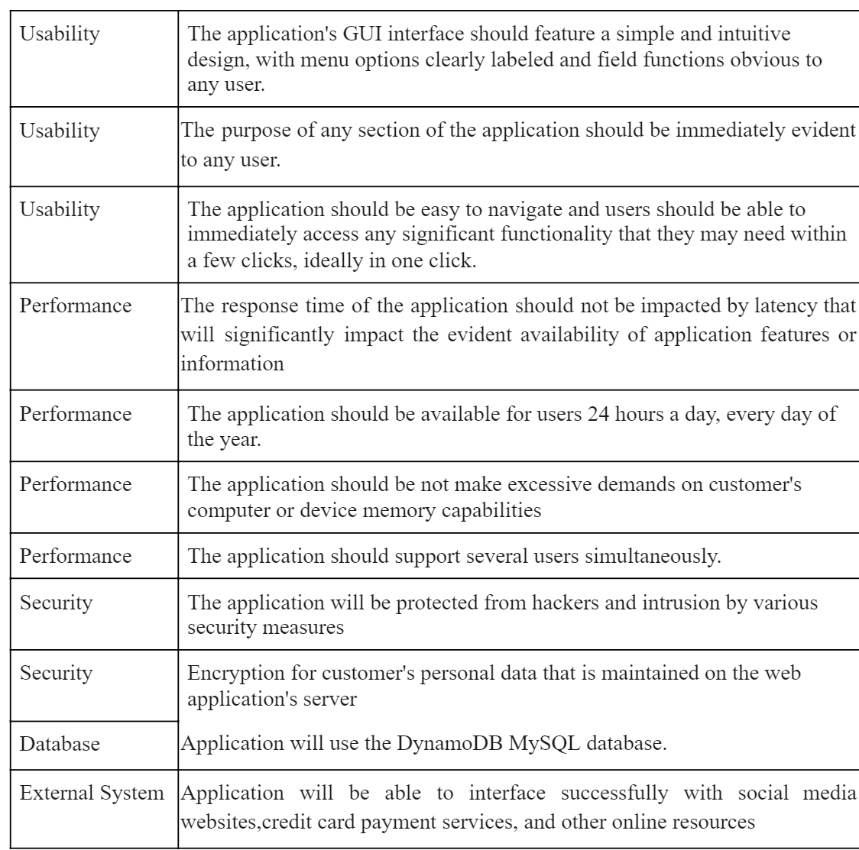
* Create a feedback

To test if a user can upload their feedback.

In addition to the above core features, testing will cover crosscutting concerns applicable to the context of individual core features, see the Requirements Composition Table for reference.

*Non-Functional Features to be Tested*

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



* Usability
  + To test whether the application has a clear interface.
  + To test whether users can use the application without a high level of computer experience.
  + To test whether the application requires users to take less than 5 minutes to figure out a feature.
* Performance
  + To test whether the response time of the application is not exceeding 2 seconds depending on the user's connection condition.
  + To test whether the response is fast enough to avoid users’ response collisions.
  + To test whether the application is available for users 24 hours a day, 365 days per year.
  + To test whether the application can simultaneously support several users.
* Space
  + To test whether the system has easy scalability.
  + To test whether the application can be backed up nightly and be able to be restored on one hour’s notice.
* Security
  + To test whether the system is safe from hackers and malicious intrusions.
  + To test the level of encryption maintained on the customer’s personal data.
* Database
  + To test the database functions are running smoothly.
* External System
  + To test if the application is properly interacting with the external platforms linked to it.

## Features not to be Tested

As mentioned above, system performance will not be tested for the lack of required tools. Also, space, External system and security will not be tested as well.

# TEST PROCESS DEFINITION

## Test Process Phases and Tasks

The test process consists of 5 phases and each phase consists of a list of several tasks as defined below.

* 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

## Deliverables

The test process deliverables for the Service General project consists of documents below.

* Test Plan Document (System)
* RCT (shows testing scope)
* Test Design Document
* Test Case Specifications
* Test Execution Logs
* Application Demo (GitHub review)

# APPROACH TO SYSTEM TESTING

## Approach to Functional Testing

The main approach for functional testing is black-box method.

* Test cases design focuses on black-box techniques such as
  + boundary-value analysis
  + equivalent-class partitioning
  + cause-effect graphing
  + decision tables
  + state-transition testing
* Test execution will be conducted manually based on
  + user perspective
  + formal test case specifications

The test execution results will be captured and reported in test execution logs.

# ENTRY/EXIT CRITERIA

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

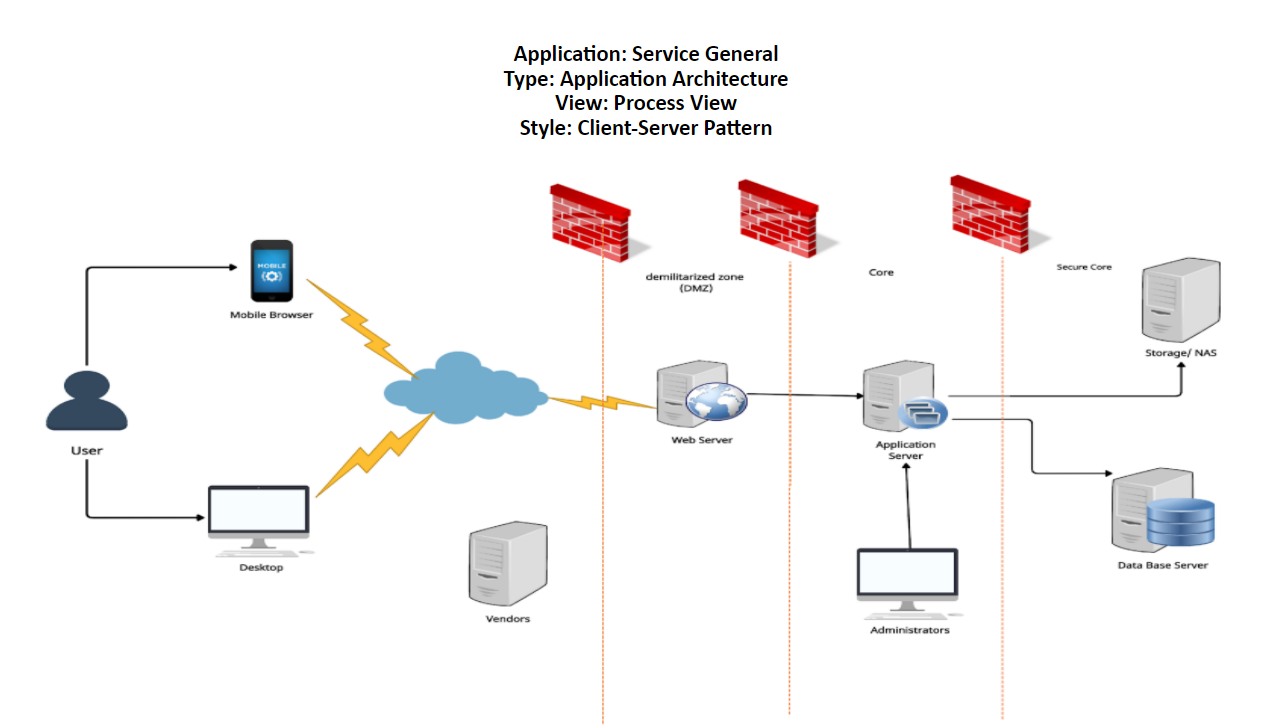
* Development tasks and integration testing have been completed
* System Test Plan document has been approved
* QA environment is ready
* QA team members have access to the QA environment
* Test case specifications have been completed and reviewed
* Release Notes document has been sent to the QA team

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

# ENVIRONMENTAL NEEDS

The Test Environment should be available to start test execution. It includes a computer running the web server 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 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 | Contributing to the test plan and test case specifications. Reviewing test results. |
| Developers | Responsible for producing a working software build, build migration to the QA environment, communicating release notes, investigating and fixing software defects. |
| Test Manager | Responsible for developing a System Test Plan document, planning the testing tasks, maintaining test specifications and test cases, coordinating test execution, producing a Test Completion Report. |
| QA Lead | Responsible for developing test cases, overseeing test execution, conducting defect review calls, providing test execution metrics and reports. |
| Testers | Responsible for developing and executing test cases, reporting defects and re-testing defect fixes. |

# 

# TEST CYCLES AND SCHEDULE

The system test execution suggested to be conducted as 5 test cycles that are aligned with 3 application modules for 1st semester:

## Cycle 1 - User Experience

* The cycle focuses on testing the first part User Experience frontend: forms input and validation
* Scheduled for Monday each week

## Cycle 2 - User Register/Login

* The cycle focuses on testing backend of user Sign In / Up
* Scheduled for Tuesday each week

## Cycle 3 - User Register Job

* The cycle focuses on testing backend of registering and viewing jobs by User
* Scheduled for Wednesday each week

## Cycle 4 - User Feedback

* The cycle focuses on testing backend of users Feedback
* Scheduled for Thursday each week

## Cycle 5 - Full (Rest) User Functionality

* The cycle focuses on the rest part of users functionality which is not highlighted in RCT for the first semester in yellow.
* Scheduled for Friday each week

# 

# 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.
* Many defects require a longer time to fix defects and complete testing.
* Instability of the test environment can impact the test execution schedule.