Test Plan

Job Search System

Prepared by:

1. Deepak Kadam (F16111035)
2. Arisha Chamadiya (F16111049)
3. Swati Patra (F16111065)

Problem Statement:

To test Job Search System Modules using Black-box and White-box testing covering Unit and Integration test by using JUnit testing tool.

System Modules:

1. Insert Company
2. Show Company Details
3. Search Candidate
4. Insert Candidate
5. Search Company

TABLE OF CONTENTS

1.0 INTRODUCTION

The application “Job Search” provides an easy and convenient search application for job seekers to find their desired jobs and for the recruiters to find the right candidate. Job seekers can register with the application and update their details and skill set. Employer can register with the application and post their current openings. They can view the Job applications and can screen them according to the best fit.

2.0 OBJECTIVES AND TASKS

2.1 Objectives of Test Plan

- Finding defects which may get created by the programmer while developing the software.
- Gaining confidence in and providing information about the level of quality.
- To prevent defects.
- To make sure that the end result meets the business and user requirements.

2.2 Tasks of Test Plan

- Identifying and describing appropriate test techniques.
- Verify and assess the Test Approach.
- Execute the tests, Log results, Report the defects.

Sr. No.	#Business Requirements (BR)	#Functional Requirements (FR)	#Test Scenarios (TS)	#Testing Approaches /Strategies (TA)
01	Provide a way to the company to display their skills and marks required.	Candidates must be displayed after validating the marks and skill set.	Insert company, Show details, Search candidates	1.Functional Testing (Standard Testing)
02	Provide a way for the candidates to search for the best company that matches their skill set and marks required.	Best company must be displayed after validating the marks and skill set of the company.	Insert candidate, Search Company	

3.0 SCOPE

The Job Search System will act as a bridge of communication between company and the candidates. It will help to reach a wide range of audience. This system will provide searching of eligible candidates and search for company that matches skills and marks required without any manual checking.

4.0 TESTING STRATEGY

Approach	Type of Testing	Manual Testing		Automated Testing on Device	Tools/APIs/Libraries
		Using Device	Using Emulator		
Standard Testing (Functional Testing)	Unit Testing	Yes	Yes	Yes	Junit (Unit Testing Framework)
	Integration Testing	Yes	Yes	Yes	
	Acceptance Testing	Yes	Yes	No	
Special Type of Testing to Address Specific Challenge	Compatibility Testing	Yes	Yes	Yes	

4.1 UNIT TESTING

Definition:
Each module that is developed by group members will to be tested individually to verify proper operation so that any faulty module can be fixed immediately rather than let it exist and then cause some major issue in the integration phase.

Participants:
Deepak Kadam, Arisha Chamadiya, Swati Patra

Methodology:
During this testing, the test team brainstorms the scope of testing, test strategy and drafts a detailed test plan.

MODULE/FUNCTIONALITY NAME:	Test database connection
UNIT/CLASS:	Database
CREATED BY:	Deepak Kadam
DATE OF CREATION:	06/08/2019
DATE OF REVIEW:	10/08/2019

TEST CASE ID	TEST UNIT/CLASS	TEST CASE	PRE-CONDITION	EXPECTED RESULT	ACTUAL RESULT	STATUS (PASS/FAIL)
1	Database	Validate DBMS Connection	JDBC driver should be present	Not null	Not null	Pass

Execution Status: Completed

MODULE/FUNCTIONALITY NAME:	Insert Company test
UNIT/CLASS:	InsertCompany
CREATED BY:	Swati Patra
DATE OF CREATION:	06/08/2019
DATE OF REVIEW:	10/08/2019

TEST CASE ID	TEST UNIT/CLASS	TEST CASE	PRE-CONDITION	EXPECTED RESULT	ACTUAL RESULT	STATUS (PASS/FAIL)
1	InsertCompany	Insert company details in the database	Database connection should be established	Company details inserted into the database	Company details inserted into the database	Pass

Execution Status: Completed

MODULE/FUNCTIONALITY NAME:	Display entries from database
UNIT/CLASS:	Display
CREATED BY:	Arisha Chamadiya
DATE OF CREATION:	06/08/2019
DATE OF REVIEW:	10/08/2019

TEST CASE ID	TEST UNIT/CLASS	TEST CASE	PRE-CONDITION	EXPECTED RESULT	ACTUAL RESULT	STATUS (PASS/FAIL)
1	Display	Display company entries	Company details should be present in the database	Company list	Company list	Pass

Execution Status: Completed

MODULE/FUNCTIONALITY NAME:	Search for candidates
UNIT/CLASS:	SearchCandidate
CREATED BY:	Arisha Chamadiya
DATE OF CREATION:	06/08/2019
DATE OF REVIEW:	10/08/2019

TEST CASE ID	TEST UNIT/CLASS	TEST CASE	PRE-CONDITION	EXPECTED RESULT	ACTUAL RESULT	STATUS (PASS/FAIL)
1	SearchCandidate	Display candidates	Candidate details should be present in the database	Arisha Deepak	Arisha Deepak	Pass

Execution Status: Completed

MODULE/FUNCTIONALITY NAME:	Insert Candidate details
UNIT/CLASS:	InsertCandidate
CREATED BY:	Swati Patra
DATE OF CREATION:	06/08/2019
DATE OF REVIEW:	10/08/2019

TEST CASE ID	TEST UNIT/CLASS	TEST CASE	PRE-CONDITION	EXPECTED RESULT	ACTUAL RESULT	STATUS (PASS/FAIL)
1	InsertCandidate	Insert candidate details in the database	Database connection should be established	Candidate details inserted into the database	Candidate details inserted into the database	Pass

Execution Status: Completed

MODULE/FUNCTIONALITY NAME:	Search for company
UNIT/CLASS:	SearchCompany
CREATED BY:	Deepak Kadam
DATE OF CREATION:	06/08/2019
DATE OF REVIEW:	10/08/2019

TEST CASE ID	TEST UNIT/CLASS	TEST CASE	PRE-CONDITION	EXPECTED RESULT	ACTUAL RESULT	STATUS (PASS/FAIL)
1	SearchCompany	Searching for companies in the database based on candidate's constraints	Company Details should be present	TCS	TCS	Pass

Execution Status: Completed

4.2 INTEGRATION TESTING

Definition:
Integration 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.

Participants:
Deepak Kadam, Arisha Chamadiya, Swati Patra

Methodology:
Initially we are integrating all resource files then we are testing all integrated modules with JUnit framework

PROJECT NAME:	Job Search System
MODULE/FUNCTIONALITY:	Linked Resources Testing
CREATED BY:	Deepak Kadam
DATE OF CREATION:	11/08/2019
DATE OF REVIEW:	14/08/2019

4.3 USER ACCEPTANCE TESTING

Definition:
The purpose of acceptance test is to confirm that the system is ready for operational use. During acceptance test, end-users (customers) of the system compare the system to its initial requirements.

Participants:
Users

Methodology:
This testing done with the help of users of the system. User is the one who will test the project.

PROJECT NAME:	Job Search System
MODULE/FUNCTIONALITY:	Validate Input Fields
CREATED BY:	Deepak Kadam, Swati Patra, Arisha Chamadiya
DATE OF CREATION:	15/08/2019
DATE OF REVIEW:	19/08/2019

ID	Test Description	Step #	Test Steps	Exp. Results	#Business Req. Covered	#Functional Req. Covered
1	Validating all the entries	-	1) If one of the fields is empty, it will give an alert message that all entries are compulsory 2) Input type will be validated	Text field constraints and user credentials are validated	Username and password are being verified using Java String functions	Login credentials has been validated through database

5.0 HARDWARE REQUIREMENTS

- RAM 1024MB+
- Intel Core i3 Processor

6.0 ENVIRONMENT REQUIREMENTS

6.1 Main Frame

- Operating Platform (Any Operating System)
- Java Technology
- JUnit Testing Framework libraries

7.0 TEST SCHEDULE

Task Name	Start Date	Finish Date	Effort Estimation	Comments
Test Planning	16/07/2019	04/08/2019		Test Cases Formed
Review Requirements documents	16/07/2017	20/07/2019	05 d	
Create initial test estimates	21/08/2019	4/08/2019	15 d	
Unit Testing	06/08/2019	10/08/2019	6d	Modules Tested
Integration testing	11/08/2019	14/08/2019	4d	Successful
User Acceptance Testing	15/08/2019	19/08/2019	5d	Successful
Resolution of final defects and final build testing	20/08/2019	23/08/2019	4d	Defects Found

8.0 FEATURES TO BE TESTED

- Login text fields validation
- Database fetching
- User credentials should be tested

9.0 RESOURCES/ROLES & RESPONSIBILITIES

1. Deepak Kadam – Unit Testing
2. Arisha Chamadiya – Integration Testing
3. Swati Patra – User Acceptance Testing

10.0 DEPENDENCIES

- JDBC Libraries

11.0 TOOLS

- JUnit testing framework
- Eclipse IDE

12.0 APPROVALS

Ms. R. M. WAHUL

Defect Report

ID	1
Project	Job Search System
Product	Job Search System
Release Version	1
Module	Database
Detected Build Version	1
Summary	Missing JDBC MySQL component
Description	Connection to the MySQL database fails due to absence of important JDBC component
Steps to Replicate	Try to establish a connection to the database using DriverManager without including the JDBC connector JAR file
Actual Result	ClassNotFoundException thrown
Expected Results	Database connection established successfully
Defect Priority	**HIGH**
Reported By	Group Members
Assigned To	Group Members
Status	Solved
Fixed Build Version	2