Team A

Date <March 21, 2012>

[Note: Text enclosed in square brackets and displayed in blue italics is included to provide guidance to the author and should be deleted before publishing the document.]

Project Participants:

# Table of Contents

**Executive Summary**

*- Introduction*

*- Summary of the report*

# Requirements Specification

## Requirements Summary

- This section of the Software Requirements Specification should contain all the software requirements to a level of detail sufficient to enable designers to design a system to satisfy those requirements and testers to test that the system satisfies those requirements.

- Include the entire team’s requirements here.

## Project Scope

## Team plan and individual responsibilities

* State both planned and the extent to which you adhered or deviated from the plan.
* Meeting dates, attendees, and primary purpose of meeting (as planned and as executed)

# Database

**2.1 ERD**

**2.2 List of tables and attributes**

# User Interface

*- Describe the activities related to screens/forms. Provide a brief explanation on the screens/forms that were created, changed, and/or deleted. Make it cumulative – include the screens/forms you created earlier and highlight the addition/modification/deletion.*

*- At the iteration, create an integrated screen flow diagram at the group level. This should include all the screens for the system.*

|  |  |
| --- | --- |
| **Screen name** | **Description** |
| Screen1 |  |
| Screen1 |  |
| Etc |  |

Screenshots should be included as an appendix.

# Coding

*- Describe the activities related to coding. Explain the source code files, and classes/objects that were created, changed, and/or deleted. Make it cumulative – include the codes you created earlier and highlight the addition/modification/deletion.*

|  |  |
| --- | --- |
| **Program name** | **Description** |
| Program1 |  |
| Program2 |  |
| etc |  |

Revision History

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Report #** | **Date** | **Version** | **Description** | **Author** |
| 1 | <mm/dd/yyyy> | <x.x> | <details> | <name> |
| 2 |  |  |  |  |
| 3 |  |  |  |  |
| 4 |  |  |  |  |
| 5 |  |  |  |  |
| 6 |  |  |  |  |
| 7 |  |  |  |  |

- The link for our system is as follows:

*(Put the link for your system here.)*

# Testing

## Web Testing Functionality[[1]](#footnote-1)

The following elements should be tested in your web testing.

• Links

- Internal Links

- External Links

- Mail Links

- Broken Links

• Forms

- Field validation

- Error message for wrong input

- Optional and Mandatory fields

• Database

- Testing will be done on the database integrity

• Cookies

- Testing will be done on the client system side, on the temporary Internet files.

• Usability

Usability testing is the process by which the human-computer interaction characteristics of a system are measured, and weaknesses are identified for correction.

- Ease of learning

- Navigation

- Subjective user satisfaction

- General appearance

• Security

The primary reason for testing the security of a web is to identify potential vulnerabilities and subsequently repair them.

- Network Scanning

- Vulnerability Scanning

- Password Cracking

- Log Review

- Integrity Checkers

- Virus Detection

## Unit Testing

Unit testing is conducted to verify the implementation of the design for one software element (e.g., unit, module) or a collection of software elements. The purpose of unit testing is to ensure that the program logic is complete and correct and ensuring that the component works as designed.

### Test Results

*- Write test cases and the code that pass them. The test cases should be sophisticated enough to find failures in the modules. The failure result does mean you have something to improve in the next iteration.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID # | Test Condition | Expected Result | Procedure | Pass/ Fail |
| 1 | A job seeker applies to a job. | Generate the confirmation id of the job application. | 1. A job seeker logs in.  2.  3.  4.  5. |  |

### Screen Shots

*- Provide the screen shots for each test case. Include the screens that show the input and the output of each test case, respectively.*

## Integration Testing

- Integration Testing is to ensure that design objectives are met and ensures that the software, as a complete entity, complies with operational requirements. Testing is conducted in which software elements, hardware elements, or both are combined and tested until the entire system has been integrated.

*- Provide the test results and screen shots.*

## User Acceptance Testing

- User acceptance testing is conducted to determine whether or not a system satisfies the acceptance criteria and to enable the customer to determine whether or not to accept the system. This ensures that customer requirements' objectives are met and that all components are correctly included in a customer package.

- This testing will be conducted by the instructor team and the clients.

# Issues and Risks

*- Describe issues, challenges, and risks that affect your implementation. Also, let the instructor team know technical issues that require additional support.*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Description** | **Resolution Strategy** | **Target Resolution Date** |
| 1 |  |  | MM/DD/YYYY |
| 2 |  |  | MM/DD/YYYY |
| 3 |  |  | MM/DD/YYYY |
| 4 |  |  | MM/DD/YYYY |
| 5 |  |  | MM/DD/YYYY |

# Issues faced and lessons learned

# What issues did you face - both technically and in terms of teamwork?

# Likewise, what did you learn from this project and process?

# If you were to do this project again, what would you do differently?

# User Manual

# Source code

- Compress your source code and submit the single file via the Blackboard

1. http://www.softwaretestinghelp.com/web-testing-example-test-cases/ [↑](#footnote-ref-1)