Report On

Software Test Plan of Superdry Website

**Software Testing and Quality Assurance**

Prepared By:

***Taras Kolesnichnko***

**Contents**

1. Test Plan Identifier: ......................................................................................................................... 4

2. References: ..................................................................................................................................... 4

3. Introduction .................................................................................................................................... 5

3.1 Objectives...................................................................................................................................... 5

3.2 Scope ............................................................................................................................................. 5

3.3 Definitions and Acronyms ............................................................................................................. 6

4. Test Items ........................................................................................................................................ 7

5. Software Risk Issues ........................................................................................................................ 8

6. Features to be tested .................................................................................................................... 10

7. Features not to be tested ............................................................................................................. 13

8. Process Overview .......................................................................................................................... 13

9. Testing Process.............................................................................................................................. 14

10. Test Strategy ............................................................................................................................... 15

Unit Testing: ...................................................................................................................................... 16

Integration Testing ............................................................................................................................ 17

System Testing .................................................................................................................................. 17

Functional test .................................................................................................................................. 17

User Interface Testing ....................................................................................................................... 17

11. Item Pass/Fail Criteria ................................................................................................................. 18

12. Suspension Criteria ..................................................................................................................... 18

13. Test Deliverable .......................................................................................................................... 19

14. Remaining Test Tasks .................................................................................................................. 20

15. Environmental Needs .................................................................................................................. 20

16. Staffing and Training Needs ........................................................................................................ 21

17. Responsibilities ........................................................................................................................... 22

18. Schedule ...................................................................................................................................... 23

Test Plan Identifier

**1. Test Plan Identifier:**

This is the test plan on **Superdry Website**. This site is a online store for selling of clothing and accessories for both men and women. This test plan will ensure the functional test of this system and it will support the whole system that is being developed accurately. This test plan contains the test approach, how different types of testing will take place, the test scenarios and expected result.

References

**2. References:**

This test plan for **Superdry Website** is developed on some supporting documents. Refer to the actual version of the document as stored in the configuration management system. Here is the list for those reference documents.

Project Plan

Superdry Website Software Requirements Specification

Detail design document

Test plan template IEEE-829.

International Standard Testing.

For this test plan, we followed IEEE-829 format and the Software Requirements Specification document.

Introduction of the Test Plan

**3. Introduction**

This document is intended to give a complete planning of a systematic strategy for software testing of Superdry Website. Superdry Website is composed of numerous features. This test plan is actually designed to ensure those features work up to the mark. Both directly and indirectly affected elements will be addressed here.

In following paragraphs we will discuss the purpose of the plan, its scope and define the acronyms used in this document.

3.1 Objectives

This document supports the following objectives:

Identify existing project information

Identify the approach that should be followed.

Identify the features 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 estimation of the test efforts.

Fix the schedule of intended testing activities.

Identify the risks associated with the test strategy.

List the deliverable elements of the test activities.

3.2 Scope

Testing will begin at the component level and work toward the integration of the entire system. This document will mainly provide the blue print of high level testing approaches of Superdry Website. It will provide a guidance to the test engineers and a set of milestones for the manager. It will validate major system functions of Superdry Website against the customer requirements. So the whole software specification will be strictly followed in every steps. As the system has a large set of features, we have prioritized the features and testing will take place according to this priority.

3.3 Definitions and Acronyms

|  |  |
| --- | --- |
| Name | Description |
| IEEE | The Institute of Electrical and Electronic Engineers, Inc. Publisher of engineering standards. |
| Integration Testing | A level of test undertaken to validate the interface between internal components of a system. Typically based upon the system architecture. |
| Black-Box Testing | A type of testing where the internal workings of the system are unknown or ignored (i.e., functional or behavioral testing). Testing to see if the system does what it's supposed to do. |
| Milestone | A major checkpoint or a sub-goal identified on the project or testing schedule. |
| Smoke Test | A test runs to demonstrate that the basic functionality of a system exists and that a certain level of stability has been achieved. Frequently used as part of the entrance criteria to a level of test |
| Software Risk Analysis | An analysis undertaken to identify and prioritize features and attributes for testing. |
| Strategy | A description of how testing will be conducted. Includes any issues that affect the effectiveness or efficiency of testing. |
| Suspension Criteria | Metrics that describe a situation in which testing will be completely or partially halted (temporarily). |
| System Testing | A (relatively) comprehensive test undertaken to validate an entire system and its characteristics. Typically based upon the requirements and design of the system. |
| Test Item | A programmatic measure of something that will be tested (i.e. a program, requirement specification, version of an application, etc.). |
| Test Deliverable | Any document, procedure, or other artifact created during the course of testing that's intended to be used and maintained. |
| Unit Testing | A level of test undertaken to validate a single unit of code. Typically conducted by the programmer who wrote the code. |
| White-Box Testing | Testing based upon knowledge of the internal (structure) of the system. Testing not only what the system does, but also how it does it (i.e. Structural Testing) |

Test Items

**4. Test Items**

In this section we will provide a list of all those components that has been identified as test items. It is assumed that unit testing will be done thorough black box testing and testing of all module interfaces will be ensured.

The interfaces between the following subsystems will be tested:

|  |  |
| --- | --- |
| MENS  WOMENS  NEW IN  IDRIS  MY ACCOUNT  WISH LIST  SHOPPING BAG | Social networks  Footer items |

The external interfaces to the following browser will be tested:

Mozilla Firefox

Google Chrome

Internet Explorer

The basic performance test:

Add product to the basket

Delete product from the basket

Add product to the wish list

Delete product from the wish list

User registration

User authentication

The most critical performance measures to test are:

1. Response time for remote login.

2. Response time for updating information.

3. Response time for requested services.

Software Risk Issues

**5. Software Risk Issues**

There are several parts of the project that are not within the control of the application but have direct impacts on the process and must be checked as well. Our main goal is to design a test strategy that utilizes a balance of testing techniques to cover a representative sample of the system in order to minimize risk. When conducting risk analysis, two major components are taken into consideration:

The probability that the negative event will occur.

The potential loss or impact associated with the event.

We identify following risk issues associated with the test approaches of Superdry Website.

**Not Enough Training/Lack of Test Competency**: As the system will be tested by the tester who lacks competency and experience, there is a possibility of misunderstanding and misapplication of testing techniques.

**Us versus Them Mentality:** This common problem arises when developers and testers are on opposite sides of the testing issue. As the developers and testers have lack of experience, there may arise this kind of situations.

**Lack of Test Tools:** As the system is a voluntary task, management may have the attitude that test tools are a luxury. Manual testing can be an overwhelming task. Trying to test effectively without tools is like trying to dig a trench with a spoon.

**Lack of User Involvement:** Users play one of the most critical roles in testing. They make sure the software works from their perspective**.** It will be really hard to involve the users of IIT official website in the testing process.

**Not Enough Schedule or Budget for Testing:** As this project is being developed as a semester project, there is not provided enough time to conduct the test processes. It will be a challenge to prioritize the plan to test the right things in the given time**.**

**Not Enough Budget for Testing:** As this is not a commercial project, there lacks budget for testing which may affect the testing process.

**Rapid Change:** There are continuously coming some changes in requirements of the system which may affect in testing process.

Features to be tested

**6. Features to be tested**

The following is a list of the areas to be focused on during testing of the application. **Features Priority Description**

|  |  |
| --- | --- |
| **Features Priority Description** | **Features Priority Description** |
| Add product to the basket | 3 |
| Edit product in the basket | 3 |
| Delete product from the basket | 3 |
| Test User registration | 3 |
| Ensure the Email is sent to the expected receiver | 3 |
| Test User authentication | 3 |
| Change User password | 3 |
| Add product to the wish list | 3 |
| Delete product from the wish list | 3 |
| Search options | 2 |
| Section Social Networks Access | 2 |
| Section WHY SHOP AT SUPERDRY.COM | 2 |
| Section CUSTOMER SERVICES | 2 |
| Section SUPERDRY THE BRAND | 2 |
| Section INFORMATION | 2 |
| Section FIND YOUR NEAREST STORE | 1 |

Features not to be tested

**7. Features not to be tested**

The following is a list of the areas that will not be specifically addressed.

|  |  |
| --- | --- |
| **Features** | **Description** |
| Network Security | Testing network security is out of our scope. |

Process Overview

**8. Process Overview**

The following represents the overall flow of the testing process:

1. Identify the requirements to be tested. All test cases shall be derived using the current Program Specification.

2. Identify which particular test(s) will be used to test each module.

3. Review the test data and test cases to ensure that the unit has been thoroughly verified and that the test data and test cases are adequate to verify proper operation of the unit.

4. Identify the expected results for each test.

5. Document the test case configuration, test data, and expected results.

6. Perform the test(s).

7. Document the test data, test cases, and test configuration used during the testing process. This information shall be submitted via the Unit/System Test Report (STR).

8. Successful unit testing is required before the unit is eligible for component integration/system testing.

9. Unsuccessful testing requires a Bug Report Form to be generated. This document shall describe the test case, the problem encountered, its possible cause, and the sequence of events that led to the problem. It shall be used as a basis for later technical analysis.

10. Test documents and reports shall be submitted. Any specifications to be reviewed, revised, or updated shall be handled immediately.

Testing Process

**9. Testing Process**

This is a generic diagram for following the testing process.

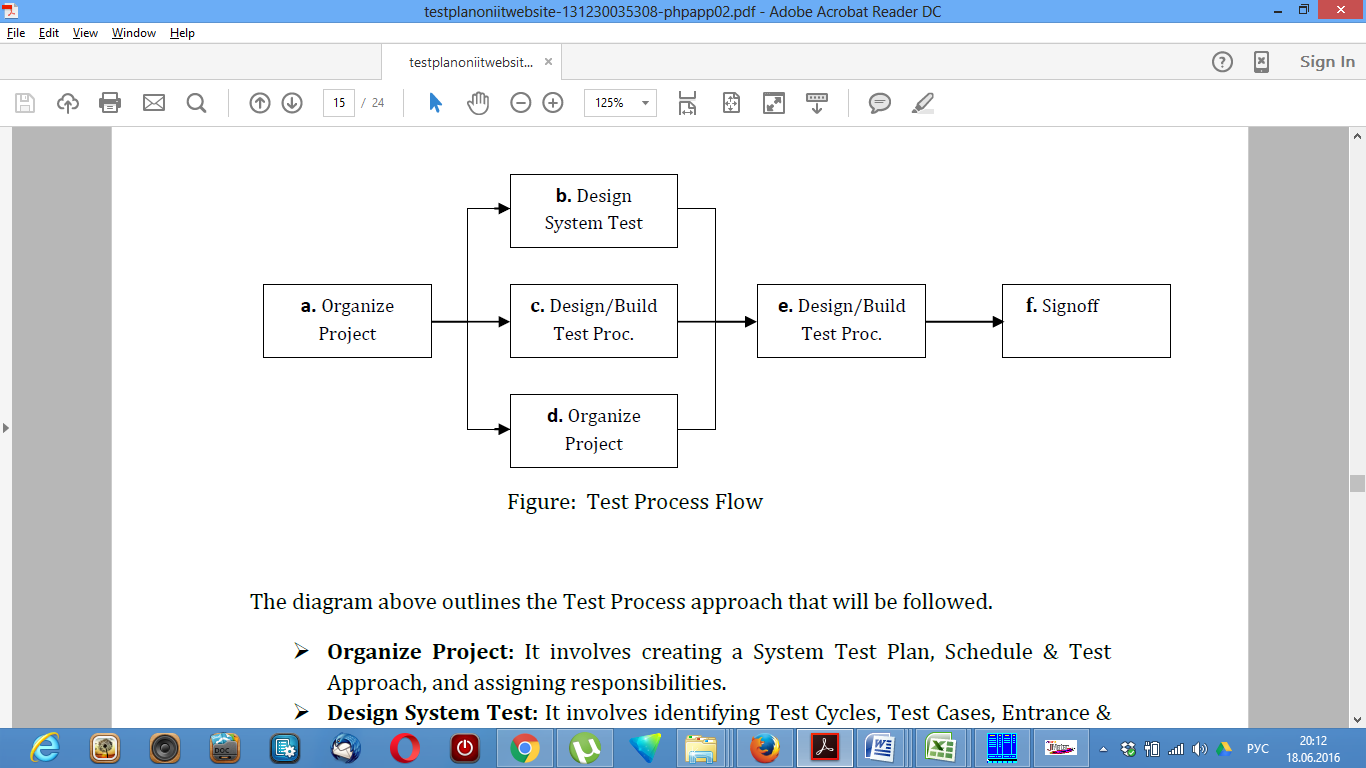


Figure: Test Process Flow

The diagram above outlines the Test Process approach that will be followed.

**Organize Project:** It involves creating a System Test Plan, Schedule & Test Approach, and assigning responsibilities.

**Design System Test:** It involves identifying Test Cycles, Test Cases, Entrance & Exit Criteria, Expected Results, etc. In general, test conditions, expected results will be identified by the Test Team in conjunction with the Development Team. The Test Team will then identify Test Cases and the Data required. The Test conditions are derived from the Program Specifications Document.

**Design Test Procedure:** It includes setting up procedures such as Error Management systems and Status reporting.

**Build Test Environment:** It includes requesting, building hardware, software and data set-ups.

**Execute System Tests:** The tests identified in the Design Test Procedures will be executed. All results will be documented and Bug Report Forms filled out and given to the Development Team as necessary

**Signoff:** Signoff happens when all pre-defined exit criteria have been achieved.

Test Strategy

**10. Test Strategy**

The testing will be done manually until the site is sufficiently stable to begin developing automatic tests. The testing will cover the requirements for all of the different roles participating in the site: guests, members, students, teachers, and administrators.

The following outlines the types of testing that will be done for unit, integration, and system testing. While it includes what will be tested, the specific use cases that determine how the testing is done will be detailed in the Test Design Document. The template that will be used for designing use cases is shown in the following figure.

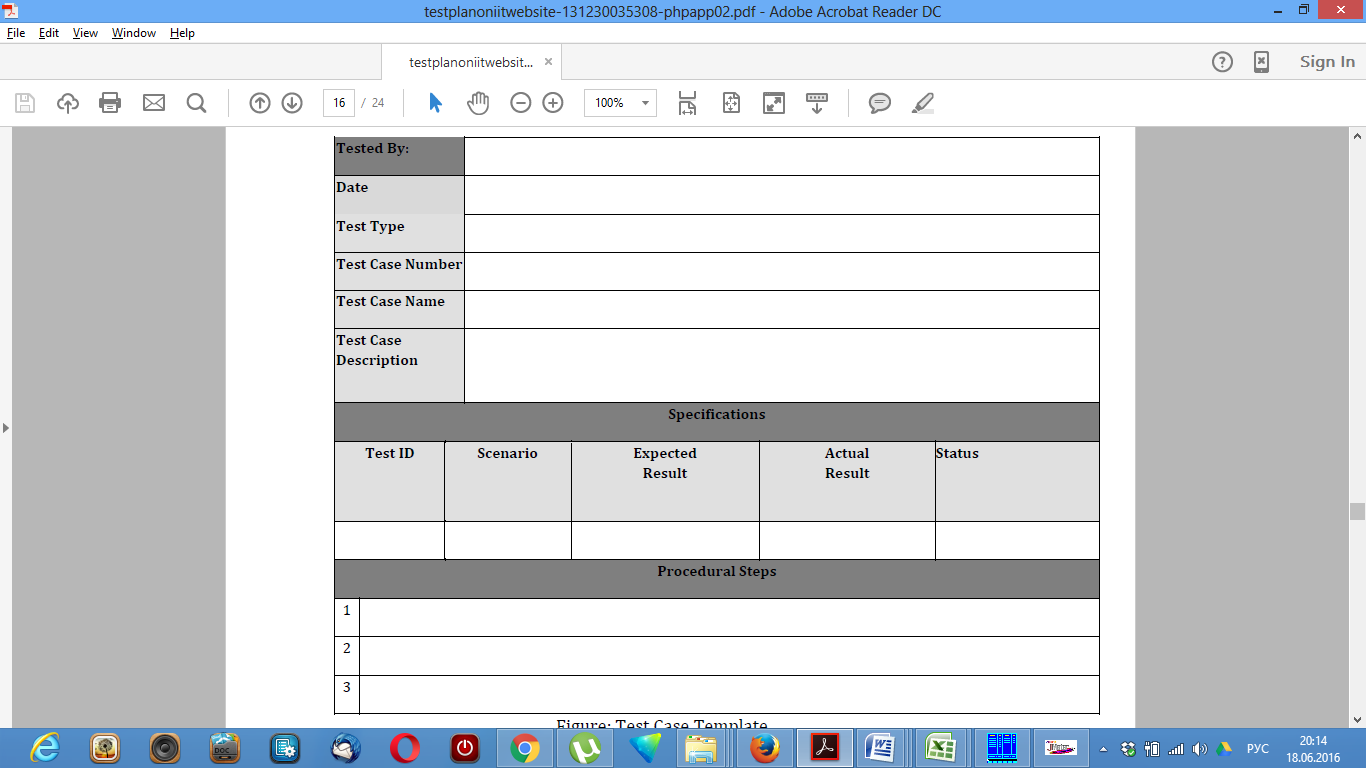


Figure: Test Case Template

An effective testing strategy includes automated, manual, and exploratory tests to efficiently reduce risk and tighten release cycles. Lacking of automated test software enforces the test manually. So the Tests come in several flavors:

Unit Testing:

**Black Box Testing**

Black box testing typically involves running through every possible input to verify that it results in the right outputs using the software as an end-user would. We have decided to perform Equivalence Partitioning and Boundary Value Analysis testing for the website.

(1) Equivalence Class Partitioning

In considering the inputs for our equivalence testing, the following types will be used:

Legal input values (Valid Input) – Test values within boundaries of the specification equivalence classes. This shall be input data the program expects and is programmed to transform into usable values.

Illegal input values (Invalid Input) – Test equivalence classes outside the boundaries of the specification. This shall be input data the program may be presented, but that will not produce any meaningful output.

Using these Valid and Invalid classes test engineer will generate test cases.

(2) Boundary Value Analysis

The acceptable range of values for this application was set by the development team. At the time of testing developer will define the boundary value and generate test case for performing the boundary value analysis.

Integration Testing

Integration tests exercise an entire subsystem and ensure that a set of components play nicely together.

System Testing

The goals of system testing are to detect faults that can only be exposed by testing the entire integrated system or some major part of it. Generally, system testing is mainly concerned with areas such as performance, security, validation, load, and configuration sensitivity. But in our case well focus only on Performance and Load testing.

**(a) Performance Testing**

This test will be conducted to evaluate the fulfillment of a system with specified performance requirements. It will be done using black-box testing method. And this will be performed by:

Storing the maximum data in the file and trying to insert, and observe how the application will perform when it is out of boundary.

Deleting data and check if it follows the right sorting algorithm to sort the resulting data or output.

Trying to store new data and check if it over writes the existing once.

Trying to load the data while they are already loaded.

**(b) Load Testing**

The test will perform after executing a performance test. This will help to develop the loading quality of the website. Tester will perform the test and generate a performance graph.

Functional test

Functional tests verify end-to-end scenarios that your users will engage in.

User Interface Testing

User Interface (UI) testing verifies a user’s interaction with the software. The goal of UI testing is to ensure that the UI provides the user with the appropriate access and navigation through the functions of the target-of-test. In addition, UI testing ensures that the objects within the UI function as expected and conform to corporate or industry standards.

For the website, we will perform unit testing first. Because unit testing will ensure that the all components of the system is working properly or not. If a single unit not work properly the integration test is not necessary to perform because in the smallest component is not working well.

If the unit testing works properly then we will perform integration test. Total system will be divided in some subsystem. And we will test that sub system work properly or not.

If the integration test perform well then we will move to functional test as well as all other tests for total system is it work properly or not.

Item Pass/Fail Criteria

**11. Item Pass/Fail Criteria**

The entrance criteria's for each phase of testing must be met before the next phase can commence. Now the criteria’s for pass and fail are given below. These criteria are set with the help of software requirements specification version 1.0

1. According to the given scenario the expected result need to take place then the scenario will be considered as pass otherwise that criteria should be failed.

2. If an item tested 10 times, 9 times perfectly worked and single time do not work properly then it will consider as fail case.

3. System crash will be considered as fail case.

4. After submitting a query in the system, if expected page won’t appear then it will be considered as fail case.

Suspension Criteria

**12. Suspension Criteria**

Testing work is performed by the test team to ensure that all the functional activities are working well in the system properly. If some of these functions are not working properly in the system the further test procedure should not be continued to the next level. So here are some criteria’s for which we will be paused the test work for the website.

If sanity check failed.

If smoke test failed.

If interdepended modules are not working well.

Testing will only stop if the Web site Under Test (WUT) becomes unavailable.

If the number or type of defects reaches a point where the follow on testing has no value, it makes no sense to continue the test; it just wasting resources.

Certain individual test cases may be suspended, skipped or reduced if prerequisite tests have previously failed e.g. usability testing may be skipped if a significant number of Web page navigational tests fail.

These criteria’s are to be used to suspend the testing activity resolved these criteria specifying testing activities which must be redone when testing is resumed.

Test Deliverable

**13. Test Deliverable**

The following documents will be generated as a result of these testing activities:

Master test plan (this document)

Individual test plans for each phase of the testing cycle

Test Design Specifications

Test log for each phase

Acceptance Test plan.

Unit test plan.

Screen Prototypes.

Test report.

Test scenario and expected result in an excel sheet.

System manual.

Remaining Test Tasks

**14. Remaining Test Tasks**

|  |  |
| --- | --- |
| **Task** | **Assigned To** |
| Create Acceptance Test Plan | Project Manager, Clients |
| Create Integration Test Plan | Developers, Project Manager |
| Define Unit Test rules and Procedures | Developers, Project Manager |
| Define Turnover procedures for each level | Developers |
| Verify prototypes of Screens | Clients, Project Manager |
| Verify prototypes of Reports | Clients, Project Manager |

Environmental Needs

**15. Environmental Needs**

The following elements are required to support the overall testing effort at all levels within the IIT official website project:

Access to the Superdry Website

Access to the database

Access to the environments used by the Superdry Website users

Systems functional structure created by Mind map

Staffing and Training needs

**16. Staffing and Training Needs**

This section outlines how to approach staffing and training the test roles for the project. Staffing is fixed for the duration of this project. It is likely most of the staff will assume some testing role.

The following roles are identified:

**Project Manager:** Responsible for managing the total implementation of the Web site. This includes creating requirements, managing the vendor relationship, overseeing the testing process.

**Test Manager:** Responsible for developing the master test plan, reviewing the test deliverables, managing the test cycles, collecting metrics and reporting status to the Project Manager, and recommending when testing is complete.

**Test Engineer:** Responsible for designing the tests, creating the test procedures, creating the test data, executing tests, preparing incident reports, analyzing incidents, writing automated test procedures, and reporting metrics to the test manager.

The test manager and test engineers should be familiar with the Website development life cycle methodology. This is a generic description of Staffing and Training needs because this project is being developed by following a generic methodology. So, the name of responsible persons for each task is not defined.

Responsibilities

**17. Responsibilities**

Identify groups responsible for managing, designing, preparing, executing, witnessing, checking and resolving that will help the whole team to deliver a quality full website.

Schedule

**18. Schedule**

This section defines the following tasks.

Specify test milestones

Specify all item transmittal events

Estimate time required to do each testing task

Schedule all testing tasks and test milestones

For each testing resource, specify its periods of use.

Test Schedule:

|  |  |
| --- | --- |
| **Test Phase** | **Time** |
| Test Plan Creation | 1 Week |
| Test Specification Creation | 2 Weeks |
| Test Spec. Team Review | 1 Week |
| Unit Testing | Developing time |
| Component Testing | 1 Week |
| Integration Testing | 1 Week |
| Use Case Validation | 1 Week |
| User Interface Testing | 1 Week |
| Load Testing | 1 Week |
| Performance Testing | 1 Week |
| Release to Production | 1 Week |

This is the actual model for testing the whole project before delivering the project.