Test Summary Report for Unacademy Application

**Contents**

1. [Purpose 2](#_bookmark0)
2. [Application Overview 2](#_bookmark1)
3. [Testing Scope 2](#_bookmark2)
4. [Metrics 3](#_bookmark3)
5. [Types of testing performed 5](#_bookmark4)
6. [Test Environment & Tools 5](#_bookmark5)
7. [Lessons Learnt 6](#_bookmark6)
8. [Recommendations 6](#_bookmark7)
9. [Best Practices 6](#_bookmark8)
10. [Exit Criteria 7](#_bookmark9)
11. [Conclusion/Sign Off 7](#_bookmark10)

# Purpose

This document explains the various activities performed as part of Testing of Unacademy application.

# Application Overview

This system allows the user’s to register and login to the application, The user can select the Exams which they want to prepare from Manage goals section The user can choose their preference. There are more than thirteen courses available .Then they have to select their level, that is Beginner, Intermediate, and Advanced. Then the user get tests according their interests. The courses available for free and subscription. If they need subscription course the user have to pay using Net banking or Upi method payments.

# Testing Scope

* + 1. **In Scope**

Functional Testing for the following modules are in Scope of Testing

* + - * Registration
      * Login
      * Manage Goals
      * Syllabus
      * Tests
      * Payment
    1. **Out of Scope**

Recovery testing is not done for this application. This can be tested after any failure of data.

* + 1. **Items not tested**

Verification of connectivity with the third party system ‘Central repository system’ was not tested, as the connectivity could not be established due to some technical limitations. This can be verified during UAT (User Acceptance Testing) where the connectivity is available or can be established.

# Metrics

**a.No. of test cases planned vs executed b.No. of test cases passed/failed**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test cases**  **planned** | **Test cases**  **executed** | **TCs**  **Pass** | **TCs**  **Failed** |
| 15 | 13 | 13 | 0 |

# Types of testing performed

* + 1. **SMOKE TESTING**
       - This testing was done whenever a Build is received (deployed into Test environment) for Testing to make sure the major functionalities are working fine, Build can be accepted and Testing can start.
    2. **SYSTEM INTEGRATION TESTING**
       - This is the Testing performed on the Application under test, to verify the entire application works as per the requirements.
       - Critical Business scenarios were tested to make sure important functionalities in the application works as intended without any errors.
    3. **RETESTING TESTING**
       - Re-testing is executing a previously failed test against new software to check if the problem is resolved. After a defect has been fixed, re-testing is performed to check the scenario under the same environmental conditions.
       - Retesting ensures that the issue has been fixed and is working as expected.
       - In some cases the entire module is required to be re-tested to ensure the quality of the module.
    4. **SANITY TESTING**
       - Sanity testing is done to check the bugs have been fixed after the build.
       - Sanity tests helps to avoid wasting time and cost involved in testing if the build is failed. Tester should reject the build upon build failure.
    5. **UNIT TESTING**
       - Unit testing is a software testing method where “units”—the individual components of software—are tested. Developers write unit tests for their code to make sure that the code works correctly. This helps to detect and protect against bugs in the future.

# Test Environment & Tools

**Software Environment**

* Operating System: Windows7 Ultimate which supports networking.
* JAVA development toolkit.

**Hardware requirements**

* Processor:1 GHz, intel i3
* RAM:100 Mb
* Hard Disk:1 GB
  1. **Lessons Learnt**

|  |  |  |
| --- | --- | --- |
| **S. No** | **Issues faced** | **Solutions** |
| 1 | Smoke testing test cases required to be executed manually each  time. | Smoke test cases were automated and the scripts were run, which ran fast and saved time. |

* 1. **Recommendations**

While doing and executing the test cases it requires more time. For time saving we can use some automation tools.

* 1. **Best Practices**

A repetitive task done manually every time was time consuming. This task was automated by creating scripts and run each time, which saved time and resources.

* Smoke test cases were automated and the scripts were run, which ran fast and saved time.
* Automation scripts were prepared to create new users, where lot of records need to be created for Testing.
* Business critical scenarios are separately tested on the entire application which are vital to certify they works fine.
  1. **Exit Criteria**
     1. All test cases should be executed – **Yes**
  2. **Conclusion/Sign Off**

As the Exit criteria was met and satisfied as mentioned in Section 10, this application is suggested to ‘Go Live’ by the Testing team. Appropriate User/Business acceptance testing should be performed before ‘Go Live’.

**\*\*\*\*\*\*\*\*\*\***