**COMSATS University Islamabad,   
 Park Road, Chak Shahzad, Islamabad Pakistan**

#### Department of Computer Science

**Assignment-05**

**CLO-4**

**Software Test Plan Document**

For

Project Tittle

*(Valid Title reflecting scope and objectives)*

**Submitted By:**

**Student Name 1** **SP21-BCS-xxx**

**Student Name 2 SP21-BCS-xxx**

**Supervised By:**

**Mr. Tehseen Riaz Abbasi**

Submission Date: (Day-Month-Year)

Original Version 1.0

(Mention the appropriate project TP Document version)

*Bachelor of Science in Computer Science (2021-2025)*

Submission Date: (Day-Month-Year)

Table of Content

[1. Introduction 3](#_Toc71659496)

[1.1. Document Purpose 3](#_Toc71659497)

[1.2. Product Scope 3](#_Toc71659498)

[1.3. Intended Audience 3](#_Toc71659499)

[1.4. Definitions, Acronyms and Abbreviations 3](#_Toc71659500)

[1.5. References and Acknowledgments 3](#_Toc71659501)

[2. Executive Summary 4](#_Toc71659502)

[2.1. Test Items 4](#_Toc71659503)

[2.2. Features To Be Tested 4](#_Toc71659504)

[2.3. Features Not To Be Tested 4](#_Toc71659505)

[2.4. Item Pass/Fail Criteria 5](#_Toc71659506)

[2.5. Item Pass/Fail Criteria 5](#_Toc71659507)

[3. Testing and Evaluation 5](#_Toc71659508)

[3.1. Verification 6](#_Toc71659509)

[3.2. Validation 6](#_Toc71659510)

[3.3. Usability Testing 6](#_Toc71659511)

[3.4. Module / Unit Testing 6](#_Toc71659512)

[3.5. Integration Testing 6](#_Toc71659513)

[3.6. System Testing 6](#_Toc71659514)

[3.7. Acceptance Testing 6](#_Toc71659515)

[4. Test Cases. 6](#_Toc71659516)

[5. Test Deliverables 7](#_Toc71659517)

[What is to be delivered as part of this plan? 7](#_Toc71659518)

[5.1. Test Tasks 7](#_Toc71659519)

[5.2. Environmental Needs 8](#_Toc71659520)

[5.3. Responsibilities 8](#_Toc71659521)

[6. Conclusion 8](#_Toc71659522)

[7. Appendices 8](#_Toc71659523)

[8. References 8](#_Toc71659524)

[9. Plaragism Report 8](#_Toc71659525)

## Introduction

Specify the introduction of the **Software Test Plan Document (STP),** should provide an overview of the entire Test Plan. Please provide a brief introduction to your project and a brief overview of what the reader will find in this section.

(**Usually in 1-2 paragraphs)**

## Document Purpose

Specify the purpose of this **Test Plan.** Describe the scope of the product that is covered by this Test Plan, particularly if this Test Pan describes only part of the system or a single subsystem.

(**Usually in 1-2 paragraphs** describing the purpose of this document as explained above)

## Product Scope

Write down the scope of your project in a paragraph. Briefly define the main functionalities of the proposed project.

**This subsection should:**

* + 1. Provide a brief description of the software system to be produced by name along with its purpose, **For example,** Host DBMS, Report Generator, etc
    2. Explain what the software product(s) will, the feature or other subsystem grouping and, if necessary, will not do
    3. **Describe the application of the software being specified.** As a portion of this, it should:

Describe all relevant benefits, objectives, and goals as precisely as possible. **For example**, to say that one goal is to provide effective reporting capabilities is not as good as saying parameter-driven, user-definable reports with a 2 h turnaround and on-line entry of user parameters.

Be consistent with similar statements in higher-level specifications (**For example**, the System Requirement Specification), if they exist.

**(Usually in 1-2 paragraphs** describing the scope of the product. Make sure to describe the benefits associated with the product.)

## Intended Audience

Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers (In your case it would probably be the “client” and the professor, project committee members.

**(Usually in 1 paragraph**.)

## Definitions, Acronyms and Abbreviations

This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the **Test Plan**.  This information may be provided by reference to the project Glossary. Please provide a list of all abbreviations and acronyms used in this document sorted in alphabetical order.

## References and Acknowledgments

This subsection should provide a complete list of all documents referenced elsewhere in the TP or in a separate, specified document. Each document should be identified by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.

## Executive Summary

Identify the Scope of the plan in relation to the Software Project plan that it relates to. Other items may include, resource and budget constraints, scope of the testing effort, how testing relates to other evaluation activities (Analysis & Reviews), and possible the process to be used for change control and communication and coordination of key activities. As this is the “Executive Summary” keep information brief and to the point.

## Test Items

These are things you intend to test within the scope of this test plan. Essentially a list of what is to be tested. This can be developed from the software application test objectives inventories as well as other sources of documentation and information such as:

* Requirements Specifications
* Design Specifications
* Users Guides
* Operations Manuals or Guides
* Installation Manuals or Procedures

This can be controlled and defined by your local Configuration Management (CM) process if you have one. This information includes version numbers, configuration requirements where needed, (especially if multiple versions of the product are supported). It may also include key delivery schedule issues for critical elements.

Identify any critical steps required before testing can begin as well, such as how to obtain the required item.

This section can be oriented to the level of the test plan. For higher levels it may be by application or functional area, for lower levels it may be by program, unit, module or build.

References to existing incident reports or enhancement requests should also be included.

This section can also indicate items that will be excluded from testing

## Features To Be Tested

This is a listing of what is to be tested from the **USERS viewpoint of what the system does**. This is not a technical description of the software but a USERS view of the functions. It is recommended to identify the test design specification associated with each feature or set of features.

Set the level of risk for each feature. Use a simple rating scale such as (H, M, L); High, Medium and Low. These types of levels are understandable to a User. You should be prepared to discuss why a level was chosen.

This is another place where the test objectives inventories can to use to help identify the sets of objectives to be tested together, (this takes advantage of the hierarchy of test objectives). Depending on the level of test plan, specific attributes (objectives) of a feature or set of features may be identified.

## Features Not To Be Tested

This is a listing of what is NOT to be tested from both the Users viewpoint of what the system does and a configuration management/version control view. This is not a technical description of the software but a USERS view of the functions.

* Identify WHY the feature is not to be tested, there can be any number of reasons.
* Not to be included in this release of the Software.
* Low risk has been used before and is considered stable.
* Will be released but not tested or documented as a functional part of the release of this version of the software.

## Item Pass/Fail Criteria

What are the Completion criteria for this plan? This is a critical aspect of any test plan and should be appropriate to the level of the plan. The goal is to identify whether a test item has passed the test process.

* At the Unit test level this could be items such as:
  + All test cases completed.
    - A specified percentage of cases completed with a percentage containing some number of minor defects.
    - Code coverage tool indicates all code covered.
    - At the Master test plan level this could be items such as:
  + All lower level plans completed.
    - A specified number of plans completed without errors and a percentage with minor defects.
* This could be an individual test case level criterion or a unit level plan or it can be general

functional requirements for higher level plans.

* What is the number and severity of defects located?
* Is it possible to compare this to the total number of defects? This may be impossible, as some

defects are never detected.

* A defect is something that **may** cause a failure, and may be acceptable to leave in the

application.

* A failure is the result of a defect as seen by the User, the system crashes, etc.

## Item Pass/Fail Criteria

* <list of items to determine if overall testing effort succeeds or fails>
* **Application**
* The application will receive a Pass status if it succeeds in producing an accurate project estimate total.
* The application will receive a Fail status if it fails in producing an accurate project total estimate.
* **Data Conversion**
* Success criteria for data conversion are outlined in the Data Conversion Plan.]

## Testing and Evaluation

Fill the following heading as per your project criteria.

# Verification

# Validation

# Usability Testing

# Module / Unit Testing

# Integration Testing

# System Testing

# Acceptance Testing

.

## Test Cases.

Write the test cases of your project as per module wise. A sample test cases with format is mentioned.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Id:** | TC-01 | **Test Case Designed by:** | Shamim |
| **Test Case Title:** | Open Alphabets image slider | **Test Case Executed by:** | Ali |
| **Module Name:** | Teaching Pakistani sign language (PSL). | **Test Case Execution Date:** | 13-11-2019 |
| **Test Data:** | Text and Images of alphabets in PSL | **Priority:** | High |
| **Precondition:** | | System has images of alphabets of PSL. | |
| **Steps /Action** | | **System Response** | |
| 1. Click on SLV .apk  2. System display “Home” Screen  3. Click on “Navigation Menu”  4. Click on “Learning Mode”  5. Click on “Learning Alphabets”  6. Click on “Learning Alphabets by images” | | 1. System start working  2. System display “Home” screen successfully  3. System display “Navigation Menu” successfully  4. System display “Learning Mode” screen successfully  5. System display “Learning Alphabets” screen successfully  6. System will load and display PSL alphabets image slider successfully. | |
| **Expected Result:** | | After execution, SLV should load text along with images of PSL alphabets and display image slider. | |
| **Actual Result:** | | After execution, SLV has loaded text along with images of PSL alphabets and displayed image slider. | |
| **Status:** | | Pass | |

## Test Deliverables

What is to be delivered as part of this plan?

* Test plan
* Test design specifications.
* Test case specifications
* Test procedure specifications
* Test item transmittal reports
* Test logs
* Test Incident Reports
* Test Summary reports
* Test Incident reports

Test data can also be considered a deliverable as well as possible test tools to aid in the testing process

One thing that is not a test deliverable is the software; that is listed under test items and is delivered by development.

These items need to be identified in the overall project plan as deliverables (milestones) and should have the appropriate resources assigned to them in the project tracking system. This will ensure that the test process has visibility within the overall project tracking process and that the test tasks to create these deliverables are started at the appropriate time. Any dependencies between these deliverables and their related software deliverable should be identified. If the predecessor document is incomplete or unstable the test products will suffer as well.

* 1. **Test Tasks**

There should be tasks identified for each test deliverable. Include all inter-task dependencies, skill levels, etc. These tasks should also have corresponding tasks and milestones in the overall project tracking process (tool).

If this is a multi-phase process or if the application is to be released in increments, there may be parts of the application that this plan does not address. These areas need to be identified to avoid any confusion should defects be reported back on those future functions.

This will also allow the users and testers to avoid incomplete functions and prevent waste of resources chasing Non-defects.

If the project is being developed as a multi-party process this plan may only cover a portion of the total functions/features. This needs to be identified so that those other areas have plans developed for them and to avoid wasting resources tracking defects that do not relate to this plan.

When a third party is developing the software, this section may contain descriptions of those test tasks belonging to both the internal groups and the external groups.

* 1. **Environmental Needs**

Are there any special requirements for this test plan, such as:

* Special hardware such as simulators, static generators etc.
* How will test data be provided? Are there special collection requirements or specific ranges of

data that must be provided?

* How much testing will be done on each component of a multi-part feature?
* Special power requirements.
* Specific versions of other supporting software.
* Restricted use of the system during testing.
* Tools (both purchased and created).
* Communications
* Web
* Client/Server
* Network
* Topology
* External
* Internal
* Bridges/Routers
* Security
  1. **Responsibilities**

Who is in charge? There should be a responsible person for each aspect of the testing and the test process. Each test task identified should also have a responsible person assigned.

## Conclusion

Conclude this document.

(Usually 4-5 sentences)

## Appendices

Make sure to include the complete list of all constants, state variables (and their possible states), inputs and outputs in a table. In the table, include the description of these items as well as all related operations and requirements.

## References

Mention the books, research papers, web links etc.

(Usually 3-5 References)

## Plaragism Report

Attach the Plaragism report of your project requirement document from library staff of turnitin tool (<http://turnitin.com>)