

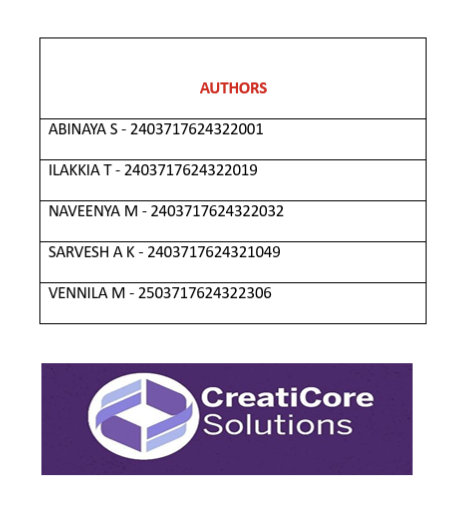
CIT Tech

Department of Artificial Intelligence and Data Science

**Home Appliance Control System (HACS)**

**Software Test Document**

**Version: *1.0d***



Document Id: HACS-STD-*1.0d*

Reviewed By: Abinaya S – Tester CreatiCore Solutions

Date: 07-10-2025

**Modification History**

**Table 1 Modification History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Modifications** | **Reason** | **Version** |
| 19/9/2025 | A draft version of the document created. | Adapted from various IEEE standards and MIL-498 standards on software test documentation. | 0.5d |
| 26/9/2025 | Add guidelines and faked examples as italics | For the team to move forward | 0.6d |
| 2/10/2025 | Minor amendments | As recommended by reviewer | 0.7d |
| 4/10/2025 | Add User Acceptance Test plan and mapping of critical features | To provide high-level UAT reference for client demonstration | 0.8d |
| *7/10/2025* | Add all sections to match current HACS STD including Functional, Navigation, UI/UX, and Look & Feel test cases | To reflect the latest HACS software version and IoT modules | 0.9d |
| 7/10/2025 | Added Definitions, Acronyms, and Abbreviations with IoT, UI/UX, and security terms | To ensure consistent look and feel across HACS app | v1.0d |

**Contents**

[1. Introduction 1](#_Toc210842572)

[1.1 Document Purpose 1](#_Toc210842573)

[1.2 About the Project 1](#_Toc210842574)

[1.3 Document Scope 1](#_Toc210842575)

[1.4 System Overview 1](#_Toc210842576)

[2. Software Test Plan 2](#_Toc210842577)

[2.1 User Acceptance Test Plan 2](#_Toc210842578)

[2.2 Functional Test Plan 3](#_Toc210842579)

[3. Software User Acceptance Test Plan 4](#_Toc210842580)

[3.1 Test Case Design for “HACS Authentication” 4](#_Toc210842581)

[3.1.1 Test Case Specification and Design for “Authentication based upon the valid username and password” 4](#_Toc210842582)

[3.2 Test Case Design for “HACS APPLIANCE CONTROL - GENERAL” 6](#_Toc210842583)

[3.2.1 Test Cases Specification and Design for Appliance Controller - General 6](#_Toc210842584)

[3.3 Test Case Design for “Energy Consumption Monitoring” 7](#_Toc210842585)

[3.3.1Test Case Specification and Design for Energy Consumption Monitoring 7](#_Toc210842586)

[3.4 Test Case Design for “UI / UX design” 8](#_Toc210842587)

[4 Software Functional Test Plan 9](#_Toc210842588)

[4.1 Test Case Design for “HACS Appliance Controller List - Operations” 9](#_Toc210842589)

[4.1.1 Test Case Specification and Design for “HACS Appliance controller list“. 10](#_Toc210842590)

[4.1.2 Test Case Specification and Design for “Adding Appliance operation 11](#_Toc210842591)

[4.1.3 Test Case Specification and Design for deleting appliance operation 11](#_Toc210842592)

[5 Requirements Traceability 12](#_Toc210842593)

[References 13](#_Toc210842594)

[Appendix A: Definitions, Acronyms, and Abbreviations 14](#_Toc210842595)

[Appendix B: Contributions 15](#_Toc210842596)

# Introduction

## Document Purpose

This document presents the software test plan prepared by CreatiCore Solutions Consulting (henceforth referred to as “HACS”) developed for the Electronic concern (henceforth referred to as “Client”). Specifically, the document details the user acceptance test plans and the functional test plans for the computer software to be developed by CreatiCore Solutions for the Client.

This document will be subject to formal and informal reviews by the CreatiCore Solution’s development team and the Client, and will form the basis for ongoing testing of the software being developed by CreatiCore Solutions , to meet the requirements of the Client.

This document forms part of the main deliverable from the software testing design activity. It is intended to be a detailed documentation for the CreatiCore Solutions’ development team to continue the quality assurance tasks such as testing as required.

Sections 2.1 and 3 constitute the user acceptance test plans,

whereas Sections 2.2 and 4 constitute the functional test plans of testing the HACS.

## About the Project

The aim of this project is to perform the ‘Software Test Design’ activity only. All tasks associated with this activity shall be performed based on the theoretical support provided as part of the ‘Software Engineering Principles and Practices ’ course. It is assumed that other related or required activities involved in the development of the system .

## Document Scope

This document defines the test plans, test case specifications, and quality assurance procedures for the Home Automation Control System (HACS). It covers functional testing, UI/UX validation, navigation, appliance control, energy monitoring, and security features. Performance and user acceptance testing guidelines are included. It does not cover software development, hardware design details, or deployment procedures. The document serves as a reference for testers to ensure system reliability and usability.

Note: In an actual test plan document, you need different levels of test plans such as unit test plans and integration test plans. However, for assignment purposes, this document only details the testing of the HACS at the user acceptance level and at the functionality level.

## System Overview

The “Home Control Appliance System” (HACS) is a system which provides various services to remote systems such as a mobile phone, desktop and entertainment system.

Document Overview

This document, the Software Test Document (STD), identifies the software test plan and the details of the testing to be carried out for the HACS.

This document has *5* major sections and *2* appendices :

* **Introduction** (Section 1)provides an overview of the entire HACS system and the HACS Software Test Document.
* **Software Test Plan** (Section 2) documents the approaches used in different stages of testing from functional testing to user acceptance testing.
* **Software User Acceptance Test Plan** (Section 3) describes the details of the user acceptance testing of the software.
* **Software Functional Test Plan** (Section 4) describes the details of testing the functionalities of the system to ensure the software behaves as described in SRS.
* **Traceability of Requirements**(Section 5) documents the related requirements per test cases in this document.
* *All other major sections and appendices must be added as dot points with a brief description of the contents in each of the sections or appendices.*

# 2. Software Test Plan

This section describes the testing plans for testing the HACS software. In general, this includes the user acceptance test (UAT) and the functional testing. Moreover, it is assumed that the next level of testing is performed only when every test plan in the previous testing level has been carried out without any error.

## User Acceptance Test Plan

This section provides a high-level view of the User Acceptance Test (UAT) plan for HACS, mapping critical functionalities under test to the SRS. The UAT focuses only on essential features that need to be demonstrated to the client, ensuring the system meets user expectations. Critical functionalities include:

**Appliance Control:** Turning ON/OFF appliances, scheduling, manual overrides, and multiple appliance management.  
**Energy Monitoring:** Displaying real-time energy consumption and historical reports.  
**Authentication:** Secure login/signup, role-based access, and password recovery.  
**UI/UX and Navigation:** Consistent layout, branding, and responsive design.  
**System Connectivity:** Verification of device-controller link and error handling for disconnections.

**Note:** Functional testing must be completed prior to UAT to ensure all features work correctly. References to functional test sections are sufficient when testing the same functionalities in UAT.

**Table 2 User acceptance test plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Functionality** | **Requirements in SRS** | **Description/Main Responsibilities** | **Feature to be tested** | **Test Spec. and Design** |
| Sign up to the Application | Section 3.1 | To create an account | Create account is in database | Section 3.1.1 |
| Sign in to HACS System | Section 3.1 | To access the account | Account existing ,moving to dashboard | Section 3.1.6 |
| Appliance Control(On/Off/Mode) | Section 3.2 | Enable users to control appliances from dashboard | Appliance status update and command execution | Section 4.1.2 |
| Energy Consumption Monitoring | Section 3.3 | Display usage reports and suggest savinngs | Graphical / Tabular report generation | Section 4.2.1 |
| Kid Mode Interface | Section 2.3.1.3 | Restrict access for child users | Limited appliance visibility and control | Section 4.2.2 |
| Remote Access | Section 2.2.10 | Allow control of appliances outside home | Cloud-based command execution | Section 4.2.3 |
| Scheduling and Automation | Section 2.3.1.6 | Enable users to automate appliance operations | Timer-based control and conflict resolution | Section 4.2.4 |
| Security and Alerts | Section 2.3.1.7 | Notify users of suspicious activity or device misuse | Push notifications and alert logs | Section 4.2.5 |
| Sign Out the Application | Section 3.1 | Ends current session | Redirection to Login Page | Section 3.1 |

## 2.2 Functional Test Plan

This section documents a high level view of the functional test plan of the software. Moreover, it traces the functionalities under test against the SRS of the software.

**Table 3 System function test plan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Functionality** | **Requirements in SRS** | **Description** | **Feature to be tested** | **Test Spec. and Design** |
| HACS Authentication System | Section 3.1 | Validates login/signup credentials and session creation | Input validation, error handling, session management | Section 4.1.1 |
| Appliance Controller | Section 3.2 | Executes appliance commands and updates status | ON/OFF toggle, mode change, command logging | Section 4.1.2 |
| *Energy Consumption Monitoring* | *Section 3.3* | Tracks and displays energy usage per appliance | *Data logging, report generation, cost estimation* | Section 4.2.1 |
| Group Control | Section 3.2.7 | Allows batch control of appliances in a room | Simultaneous command execution | Section 4.2.2 |
| Scheduling and Automation | Section 2.3.1 | Enables timer-based appliance operations | Conflict detection, schedule execution | Section 4.2.3 |
| Notification and Alerts | Section 2.2.11 | Sends alerts for unusual activity or prolonged usage | Push notifications, alert triggers | Section 4.2.4 |
| Kid Mode Interface | Section 2.3.1 | Restricts access to high-risk appliances for child users | Role-based UI filtering | Section 4.2.5 |
| Remote Access | Section 2.2.10 | Allows control from outside the home via cloud | Internet-based command relay | Section 4.2.6 |
| Security Features (TLS/SSL) | Section 2.5 | Ensures encrypted communication between app and hardware | TLS/SSL handshake, data encryption | Section 4.2.7 |
| Exception Handling | Section 3.1.8, 3.2.9 | invalid inputs, offline devices, and system errors | Error messages, retry prompts, fallback mechanisms | Section 4.2.8 |

# 3. Software User Acceptance Test Plan

This section describes the details of the plan for the user acceptance test (UAT). Some representatives from the Client are required for this testing session. The number of representative test cases should be subject to the mutual agreement between the Client and CreatiCore Solutions Tech The CreatiCore Solutions Tech development team will demonstrate the functionalities of the HACS using the following test specifications.

## Test Case Design for “HACS Authentication”

This section describes the specifications of the test cases for testing how well the software handles the request of creating an account.

### 3.1.1 Test Case Specification and Design for “Authentication based upon the valid username and password”

**Features to be tested**: Login and signup validation, password strength enforcement, duplicate username check, session creation and termination, error handling for invalid credentials, forgotten password and reset flow, biometric and third-party (Google) login, role-based access verification, and secure TLS/SSL data transmission.

**Requirements**: *Section 3.1 in HACS\_SRS-V 1.0 d*

**Expected Behaviour**: System should allow valid users to log in and create sessions securely, reject invalid or duplicate credentials with clear error messages, enforce strong password rules, enable signup for new users, support forgotten password recovery, ensure session logout works properly, allow biometric or Google login when configured, restrict access based on user roles, and protect all communications with TLS/SSL encryption.

**Test Case Design**:

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Id.** | **Inputs required** | **Environment needs** | **Output expected** |
| 3.1.1.1 | Username: validUser Password: validPass | Mobile app installed Internet connection Database with registered user | User successfully logged in Redirected to Home Dashboard |
| 3.1.1.2 | Username: wrongUser Password: wrongPass | Mobile app installed Internet connection Database with registered user | Error message: “Re-enter your username and password” |
| 3.1.1.3 | Username: notRegistered Password: any | Mobile app installed Internet connection Database with registered user | Prompt: “Username not found. Would you like to sign up?” |
| 3.1.1.4 | Username: validUser Password: forgotten | Mobile app installed Internet connection Database with registered user | Redirected to “Forgot Password” flow |
| 3.1.1.5 | Login via Google/Gmail | App with Google Auth enabled Internet connection | Successful login via OAuth Session created |
| 3.1.1.6 | Login via Biometric (fingerprint or face) | Device with biometric sensor App permissions granted | Successful login Redirected to Home Dashboard |

## Test Case Design for “HACS APPLIANCE CONTROL - GENERAL”

This section describes the specifications of the test cases for testing how well the software handles the request of creating an account.

### 3.2.1 Test Cases Specification and Design for Appliance Controller - General

**Features to be tested**: Appliance ON/OFF and mode control, real-time status updates, scheduling and timer accuracy, group control functionality, command validation, error handling for offline devices, data logging of actions, and synchronization with the hardware controller

**Requirements**: *Section 3.2 in HACS\_SRS-V 1.0 d*

**Expected Behaviour**: *The system should allow authenticated users to select and control appliances (turn ON/OFF or change mode) with instant status updates, ensure each command is validated before execution, prevent overlapping manual and scheduled actions, log every control event with date and time, display real-time appliance state on the dashboard, handle offline or unresponsive devices with appropriate error messages, support group control for multiple devices, and synchronize all actions with the hardware controller.*

**Test Case Design**:

|  |  |  |  |
| --- | --- | --- | --- |
| **Test ID** | **Inputs Required** | **Environment Needs** | **Output Expected** |
| 3.2.2.1 | Command: ON | HACS system running, appliance connected | Appliance turns ON, status shows ON |
| 3.2.2.2 | Command: OFF | HACS system running, appliance connected | Appliance turns OFF, status shows OFF |
| 3.2.2.3 | Schedule.Time (07:00AM), Command: ON | Scheduler enabled, HACS system online | Appliance turns ON at 07:00 AM |
| 3.2.2.4 | Schedule.Time (10:00PM), Command: OFF | Scheduler enabled, HACS system online | Appliance turns OFF at 10:00 PM |
| 3.2.2.1.2 | Delete / Remove Appliance | HACS system enables removing of a device in database | Appliance is removed from the HACS System |
| 3.2.2.5 | Remote app command (ON/OFF) | HACS system online, network connectivity | Appliance responds to remote command, status updates in app |
| 3.2.2.6 | Manual command (OFF) while scheduled ON | HACS system running, appliance connected | Appliance turns OFF immediately, scheduler updates next action |
| 3.2.2.7 | Invalid Appliance ID Command: ON | HACS system running | Error message: "Invalid appliance" |
| 3.2.2.8 | Power cut and restore | HACS system running, appliance connected, simulate power outage | Appliance returns to previous state or default OFF |
| 3.2.2.9 | Multiple Appliance Command: ON/OFF | HACS system running, all appliances connected | All selected appliances respond correctly, status updates for all |
| 3.2.2.10 | HACS UI | HACS UI panel accessible | Status panel shows correct ON/OFF state for all appliances in real time |

## Test Case Design for “Energy Consumption Monitoring”

This section describes the specifications of the test cases for testing how well the software handles the request of creating an account.

### 3.3.1Test Case Specification and Design for Energy Consumption Monitoring

**Features to be tested**: Accurate logging and visualization of energy consumption data for individual appliances.

**Requirements**: Refer to Section 3.3 of the SRS – Energy Consumption Monitoring.

**Expected Behaviour**:The system should fetch energy usage data, generate daily/weekly/monthly reports, and display them in graphical/tabular format. It should also

estimate cost and suggest optimizations.

**Test Case Design**:

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Id.** | **Inputs required** | **Environment needs** | **Output Expected** |
| 3.3.1.1 | Particular Date Range = 01/10/2025 to 07/10/2025 | User logged in, appliance connected, energy data available | Graphical report showing daily usage in kWh and estimated cost |
| 3.3.1.2 | Interval based  Interval = hourly | Appliance active, controller online, database logging enabled | Tabular report with hourly usage and total consumption |
| 3.3.1.3 | Particular Date based  Date = 08/10/2025 | Appliance offline, user requests report | Error message: “No usage detected” |
| 3.3.1.4 | ID and Date Range  Appliance ID = FRIDGE004, Date Range = 01/09/2025 to 30/09/2025 | User logged in, appliance connected, historical data available | Monthly report with peak usage times and energy-saving suggestions |
| 3.3.1.5 | Child Mode  Appliance ID = HEATER005, Date Range = 01/10/2025 to 08/10/2025 | User in child mode | Restricted access message or simplified report view |

## Test Case Design for “UI / UX design”

**3.4.1 Test Case Specification and Design for “Look and Feel Requirements – UI / UX ”**

**Features to be tested:** HACS slogan placement, team logo watermark, connection status icon, bottom navigation bar functionality, layout consistency, readability of text/icons, highlighting of critical actions, overall modern and intuitive interface.

**Requirements:** Section 3.3 in HACS\_SRS-V 1.0 d

**Expected Behaviour:** Every screen should consistently display the slogan and team logo, show connection status correctly, maintain consistent bottom navigation, keep layout uniform across pages, ensure text/icons are readable, highlight critical actions appropriately, and provide a modern, intuitive interface.

**Test Case Design :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Id** | **Inputs Required** | **Environment Needs** | **Output Expected** |
| 3.4.1.1 | Open any screen/page in the app | Mobile app installed, Internet connection | HACS slogan visible in top-right corner on all screens |
| 3.4.1.2 | Open any screen/page in the app | Mobile app installed, Internet connection | Team name and logo displayed in top-left corner as a low-opacity watermark |
| 3.4.1.3 | Check connection status icon | Mobile app installed, Device connected/disconnected | Green link icon when connected; Red link with cross when disconnected |
| 3.4.1.4 | Navigate to Home via bottom nav bar | Mobile app installed | Home icon highlighted, user redirected to Home Dashboard |
| 3.4.1.5 | Navigate to Appliances via bottom nav bar | Mobile app installed | Appliance section opens; Interior and Exterior categories visible and accessible |
| 3.4.1.6 | Navigate to Energy Report via bottom nav bar | Mobile app installed | Energy Report page opens, statistics displayed correctly |
| 3.4.1.7 | Switch between screens | Mobile app installed | Layout consistent; elements in same position; navigation intuitive |
| 3.4.1.8 | Verify readability of text and icons | Mobile app installed, Desktop browser | Text/icons are clear, minimalistic, and readable across devices |
| 3.4.1.9 | Perform critical action (e.g., Switch OFF all appliances) | Mobile app installed | Action highlighted with distinct color or warning prompt displayed |
| 3.4.1.10 | Overall UI evaluation | Mobile app installed, Desktop browser | Interface is modern, simple, intuitive, and visually appealing |

# 4 Software Functional Test Plan

This section describes the details of the test plan for the system testing which includes functional testing and performance testing.

## Test Case Design for “HACS Appliance Controller List - Operations”

This section describes the specifications of the test cases for testing the functionality of the Home Appliance Control System

### 4.1.1 Test Case Specification and Design for “HACS Appliance controller list“.

**Features to be tested**: User need to create a HACS appliance list. and command its operations with specified

**Requirements**: As listed in Section 3.1 of SRS HACS\_SRS\_v1.0d

**Expected Behaviour**: The system create the account details and store the details in the database along with timestamp accordingly

**Test Case Design**:

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Id.** | **Inputs required** | **Environment needs** | **Output expected** |
| 4.1.1.1 | Date = 30/04/20036Start time = 12:00  Duration = 1 | User log on as a system.  Create a account for entering to HACS system | System accepts the accounts details and stores the account details including user name and password |
| 4.1.1.2 | Date = 30/04/2003  Start time = 13:00  Duration = 2 | User log on to the system and it displays the appliance menu | Same as 4.1.1.1 |
| 4.1.1.3 | Date = 30/04/2003 Start Time = 14:00 Duration = 1 | User logged in Appliances registered and connected | System displays appliance list with ON/OFF status indicators |
| 4.1.1.4 | Date = 30/04/2003 Start Time = 15:00 Duration = 1 | User logged in One appliance disconnected | System shows “Device Offline” message for that appliance |
| 4.1.1.5 | Date = 30/04/2003 Start Time = 16:00 Duration = 1 | User logged in No appliances registered | System displays message: “No appliances available” |
| 4.1.1.6 | Date = 30/04/2003 Start Time = 17:00 Duration = 1 | User logged in Multiple appliances registered | System displays full appliance list with room-wise grouping |
| 4.1.1.7 | Date = 30/04/2003 Start Time = 18:00 Duration = 1 | User logged in Appliance list includes exterior devices | System displays both interior and exterior appliances under respective categories |

*Note: You need many more test cases here as compared with the UAT plan in Section 3.1.1.*

### 4.1.2 Test Case Specification and Design for “Adding Appliance operation

**Feature.to.be.tested:** Adding a new appliance into the Home Appliance Control System (HACS).

**Requirements:** As per SRS Section 3.1, the system shall allow users to add a new appliance with parameters such as Appliance Name, Type, and Unique ID. The details must be stored in the database for later control and monitoring.

**Expected.Behaviour:** When the user inputs valid appliance details and clicks “Add,” the system should save the information, display a success message (“Appliance added successfully”), and update the appliance list in the dashboard.

**Test Case Design**:

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Id.** | **Inputs required** | **Environment needs** | **Output Expected** |
| 4.1.2.1 | appliance name = “fan”,operation = “turn ON” | user logged into HACS system | system displays “appliance operation added successfully”and updates list. |
| 4.1.2.2 | appliance name = “light”,operation = “turn OFF” | same as 4.1.2.1 | system adds new operation and displays confirmation message. |
| 4.1.2.3 | appliance name = “AC”,operation = “set timer=2 hrs” | user logged into system with active session | appliance added and system displays “TIMER SET FOR 2 HOURS” |
| 4.1.2.4 | appliance name = “heater”,operation = “turn ON” | database disconnected or offline | system displays error message:”database connection failed. Try again” |

### 4.1.3 Test Case Specification and Design for deleting appliance operation

**Features.to.be.tested:** Deleting a registered appliance from the HACS system.

**Requirements:** As listed in Section 3.2 of SRS — the system shall allow the user to delete an appliance previously added to the appliance list and update the database accordingly.

**Expected.Behaviour:** The system should successfully remove the selected appliance from the list and confirm deletion with a message such as “Appliance Removed Successfully.” The updated list should no longer contain the deleted appliance.

**Test Case Design**:

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Id.** | **Inputs required** | **Environment needs** | **Output Expected** |
| 4.1.3.1 | user selects a registered appliance and clicks delete | user logged into HACS system with valid credentials | system prompts confirmation and removes the selected appliamce from the list |
| 4.1.3.2 | User attempts to delete an appliance that doesn’t exist. | user logged into HACS system. | System displays error message “Appliance not found.” |
| 4.1.3.3 | User deletes multiple appliances sequentially. | user logged into HACS with atleast 3 appliances registered | System removes all selected appliances one by one and updates the list each time. |
| 4.1.3.4 | User cancels the deletion when confirmation message appears. | user logged into HACS system. | deletion cancelled;appliance remains in the list. |

# 5 Requirements Traceability

This section documents the relationship between the requirements specifications in the SRS of the software and the test cases identified in this document. *(Refer to Table 9.5 in Pfleeger, 2nd ed. 2001.)*

**Table 4 Traceability of Software Requirements from test case *(Students to complete)***

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Id | Requirements  in SRS\* | 3.1  Authentication | 3.2  Appliance Control | 3.3  Energy Monitoring | 3.4  Scheduling | 3.5  Alerts | 3.6  Mode | 3.7  Remote Access |
| 3.1.1.1 | Section 3.1 | ☑ |  |  |  |  |  |  |
| 3.1.1.2 | Section 3.1 | ☑ |  |  |  |  |  |  |
| 3.1.2.1 | Section 3.1 | ☑ |  |  |  |  |  |  |
| 3.2.1.1 | Section 3.2 |  | ☑ |  |  |  |  |  |
| 3.2.1.2 | Section 3.2 |  | ☑ |  |  |  |  |  |
| 3.3.1.1 | Section 3.3 |  |  | ☑ |  |  |  |  |
| 3.3.2.1 | Section 3.3 |  |  | ☑ |  |  |  |  |
| 3.4.1.1 | Section 3.4 |  |  |  | ☑ |  |  |  |
| 3.5.1.1 | Section 3.5 |  |  |  |  | ☑ |  |  |
| 3.6.1.1 | Section 3.6 |  |  |  |  |  | ☑ |  |
| 3.7.1.1 | Section 3.7 |  |  |  |  |  |  | ☑ |
| 4.1.1.1 | Section 3.2 |  | ☑ |  |  |  |  |  |
| 4.1.2.1 | Section 3.2 |  | ☑ |  |  |  |  |  |

\*Note: The Requirements in SRS may refer to the functional requirements or non-functional requirements.

# References

1. CreatiCore Solutions, “The SRS document of the”HACS”, 2025

CIT Tech Engineering Team , Department of AI & DS , CIT Semester 3 2025

[2] IEEE, “ANSI/IEEE Standards 829-1983”, 1983

[3] IEEE, “ANSI/IEEE Standards 1008-1987”, 1987

[4] IEEE, “ANSI/IEEE Standards 1012-1986”, 1986

[5] IEEE, “ANSI/IEEE Standards 1059-1993”, 1993

[6] “MIL-498 Standards on Software Test Description”, 1994

[7] “MIL-498 Standards on Software Test Plan”, 1994

[8] Pfleeger, S. L., “Software Engineering: Theory and Practice”, Prentice Hall, Second Edition 2001

[9] Survey Report on Smart Home Systems, Department of AI & DS, Coimbatore

Institute of Technology, 2025.

[10] Notes – Software Engineering Principles and practices Lecture Notes , Dr Valliappan Raman , 2025

# Appendix A: Definitions, Acronyms, and Abbreviations

This section describes any general information that helps to understand this software test document. This section contains an alphabetical listing of all acronyms, abbreviations, and their meanings as used in this document and a list of any terms and definitions needed to understand this document.

**Table 5 Definitions, Acronyms, and Abbreviations**

|  |  |
| --- | --- |
| **Term / Acronym** | **Definition / Description** |
| Functional Testing | The process of testing software to ensure that it performs according to its functional requirements specified in the SRS. It verifies features, user interactions, and workflows. |
| Performance Testing | The activity of testing the software as a whole to verify that its performance (e.g., response time, throughput, and scalability) matches its SRS. |
| Requirement | A condition or capability needed by a user to solve a problem or achieve an objective. |
| Specification | A document that prescribes, in a complete, precise, verifiable manner, the requirements, design, behaviors, or other characteristics of a system or system components. |
| Software Requirements Specification (SRS) | A technical document that details the functional and non-functional requirements of the system. |
| Software Test Document (STD) | A document that describes test plans, test case specifications, test strategies, and other testing-related information for the software. |
| System Testing | The testing of the software application as a whole to verify that it works according to its specification. It includes functional and performance testing. |
| User Acceptance Testing (UAT) | Testing performed by the end users or clients to verify that the software meets their needs and requirements and is ready for deployment. |
| Regression Testing | Testing done after changes or updates to ensure that existing functionalities continue to work correctly. |
| Smoke Testing | A preliminary test to check the basic functionality of the software before conducting more detailed testing. |
| Test Case | A set of conditions, inputs, and expected results developed to verify a specific functionality of the system. |

# Appendix B: Contributions

This section shall contain the names of all the contributors to this document. The sections that each contributor has worked on shall be described in the table below.

**Table 6 Document Contributions**

|  |  |
| --- | --- |
| **Contributor Name** | **Sections Worked On** |
| ABINAYA.S | Recommending Test Cases, Requirement Traceability and Reviewing STD |
| ILAKKIA.T | Software UAT and contributed in adding Test Cases |
| NAVEENYA.M | Functional UAT Updation and Test Case Building |
| SARVESH A K | User Case Test plan Updation , Test Case Collection |
| VENNILA M | User Case Test Plan Section and Appendices Building |

***Note:*** *The full name of the section worked on is expected rather than just the section number.*