



MIDDLE EAST TECHNICAL UNIVERSITY
COMPUTER ENGINEERING DEPARTMENT



CENG 492 – Computer Engineering Design – II

SMART HOME PROJECT

SOFTWARE TEST DOCUMENT

v1.0

FRANQUILLUM
FRANQUILLUM

Abdullah HasanTaherBayrakdar

Anıl Ulutürk

ŞerafettinÖztürk

ZeynepMavuş

Sponsored by



Preface

This document contains the system test information for SMARTHOME Project.

The document is prepared according to the “IEEE Standard for Software and System Test Documentation – Software Test Descriptions – IEEE Std 829 – 2008”. This Software Test Documentation provides a complete description of test plan of the Smart Home Project. The first section of this document includes the document purpose, scope, major constraints, some definitions and references used though out the document.

The second section contains the software to be tested and describes the testing strategies that will apply though the testing process.

The Third section contains the software test cases in details. It includes the main types of testing that will be performed on the system and their inputs, expected outcomes, objectives, the preconditions and the environmental needs for each test case.

The fourth section introduces the resources to be used for testing.

The fifth section explains the path to be followed whenever a problem observed while applying the below mentioned steps.

The sixth section gives the detailed information about who tests which part while the last section gives a schedule which is designed based on which part will be tested when.

The seventh part contains the table for the Test Schedule.

Table of Contents

1. Introduction	5
1.1 Goals and Objectives	5
1.2 Statement of Scope	5
1.3 Major Constraints	6
1.4 Definitions, Acronyms and Abbreviations	7
1.5 References.....	7
2. Test Plan	8
2.1 Software to be tested.....	8
2.2 Testing Strategy	8
2.2.1 Functional Testing	8
2.2.2 Unit Testing	8
2.2.3 Security Testing.....	8
2.2.4 Usability Testing	9
2.2.5 Performance Testing.....	9
2.2.6 Integration Testing	9
2.2.7 Validation Testing	10
2.2.8 High-Order Testing	10
2.3 Test Metrics	10
2.4 Testing Tools and Environment.....	10
3. Test Procedure	12
3.1 Functional Use Cases.....	12
3.1.1 Remote User Interface.....	12
3.1.1.1 Sign Up	12
3.1.1.2 Sign In	12
3.1.1.3 Sign Out	13
3.1.1.4 Forgot Password.....	13
3.1.1.5 Change Password	14
3.1.1.6 Choose a Device.....	14
3.1.1.7 Turn on a Device.....	15
3.1.1.8 Turn off a Device	15
3.1.1.9 Adjust.....	16
3.1.1.10 Update Sensor Info.....	16
3.1.1.11 Add a New Device	17
3.1.1.12 View Statistics.....	17
3.2 Unit Test Cases	18
3.2.1 Components and Database(s) Interaction.....	18
3.2.2 Control Box Data Flow	19
3.2.3 Home Appliance(s) Simulation	20
3.3 Integration Testing.....	21
3.4 Validation Testing	21
3.5 High-Order Testing.....	21
4. Testing Resources and Staffing.....	21

5. Test Work Products and Test Log 22

6. Organization and Responsibilities 22

7. Test Schedule..... 23

1. Introduction

1.1 Goals and Objectives

The purpose of this document is to define and outline the strategies and procedures for testing management for the final release of SmartHome System. It covers the general methods made use of in the tests conducted for the project. Since SmartHome is a product for the average home owners and various groups of age and technical knowledge, each unit and the whole system needs further considerations rather than just testing single functionalities just after they are developed. Quality assurance, verification and validation, reliability estimation, a generic metric, and trade-off between time and quality imply the importance of testing. The goal is to end up with a robust, bug-free and high-performance product after being finished all the testing process.

1.2 Statement of Scope

This document was prepared by the developer team of tranquillum. It covers descriptions of the methods used in the testing process of SmartHome, plan of the testing phase, procedures to be followed and techniques of keeping records during testing. During the development of the project SmartHome, each component of every module is supposed to be tested by the responsible developer(s) right after its implementation is finished. However, since many modules including many components work interactive in SmartHome, further separate testing of modules and testing of the whole system is absolutely necessary. To achieve this, we will elaborate on;

- What we will test
- What are our constraints while testing
- How will we handle severe, user-facing bugs
- How will we find internal bugs
- How our testing relates to our schedule
- How team members are responsible for different tests

1.3 Major Constraints

Time: SmartHome is designed to control several home appliances concurrently namely reading the data from the an appliance via Zigbee interface, showing it graphically and keeping it for further use while the user is playing the game. Furthermore, user –from remote interface- will be able to issue commands for each appliance. Therefore, optimization of all modules for better performance and any possible speed up is in the scope the time constraint.

Data Optimality: Since SmartHome is composed of many modules running concurrently and passing data to each other during the run, minimizing the amount of data being transferred between modules is a goal in the sense of data constraint.

Data Synchronization: Local controlbox and remote server databases will have certain copies of commands and statistical data available for themselves, but synchronization of these data whenever possible in a well-defined way in order not to fail users' demands or devices' information flow is crucial and is an important part of the test phase.

Security: Zigbee devices and remote interfaces require preliminary security constraints to control any unauthorized or unauthentic commands/statistics to be sent/received within the system.

1.4 Definitions, Acronyms and Abbreviations

Term	Definition
STD	Software Test Document
User	The person(s) who use the SMARTHOME application
XBee	Is the brand name from Digi International for a family of form factor compatible radio modules.
Raspberry PI	Is a credit-card sized computer that plugs into your TV and a keyboard. It's a capable little PC which can be used for many of the things.
Zigbee	A specification for a suite of high level communication protocols using small, low-power digital radios based on an IEEE 802 standard
IEEE	Institute of Electrical and Electronics Engineers
GNU	Unix-like computer operating system developed by the GNU Project
GUI	Graphical User interface

1.5 References

- [1] IEEE Std 829-2008 IEEE Standard for Software and System Test Documentation. Computer Society, 2009.

2. Test Plan

2.1 Software to be tested

This part is aimed at identifying the items to be tested by the test cases. The items are given in following:

- Software Requirement Specifications
- Initial Design Documentation

2.2 Testing Strategy

All of the components mentioned above will be tested thoroughly. Then, each integration will also be tested to ensure their interaction is as desired. Next, validation testing will be conducted to evaluate the component during the development process to determine whether it satisfies specified requirements. Lastly, system testing will be performed to ensure the robustness of the system as a whole.

2.2.1 Functional Testing

After identifying the all functional properties specified in the software requirement specifications and initial design reports, input data will be created according to specifications given in the documentations. Each use case will be evaluated as a test case and after the execution of each test cases (specified in the use cases), the outputs will be compared with the expected ones. In other words,for each home appliance, all of the related configuration types will be tested.

2.2.2 Unit Testing

Remote server, Control Box and Embedded Simulation modules are needed to be tested separately. Since each module cannot be tested as individual parts for our project, other modules are going to be ignored during unit testing a module. The purpose of this phase is to ensure that each component does not have any internal errors.

2.2.3 Security Testing

Security testing will be considered in 2 different levels. First one is network &Internet and the second one is Zigbee communication.

Zigbee devices have certain security measures implemented within, but for the sake of better security, every device in network will be required to provide its dedicated id whenever it communicates with Control Box to send/receive frames.

Network may provide certain security problems since an unauthorized user may try to access remote interface somehow. Because of that, security of remote server database and PHP interface must be tested.

2.2.4 Usability Testing

System will be implementing few abilities to ease first hand use of the system by homeowners. These abilities include automatic device registration & recognition, sensor type detection and local administration of Control Box. Such features need to be tested by homeowners of families and this test will be covered with a similar version of Alpha/Beta testing.

2.2.5 Performance Testing

In order to be able to determine how our system performs in terms of responsiveness and stability under special workloads, load testing will be applied. Load testing includes multiple users logging in the system and multiple devices connected to the system. Additionally, after adding new devices to the system, the performance observation should be done for each device. Request-response time intervals will be observed in detail to decide to what extent the system suffers from delays.

2.2.6 Integration Testing

After all the modules pass the unit testing, they get to be tested whether they work correctly when they are running concurrently and communicating to each other. This specification is vital since although each module might be working individually, different implementations coming from different modules may conflict while running together which is to be located during integration testing and to be recorded for fixing.

Each of the other modules is connected to the Control Box module, so after the unit testing of each created module, it will be integrated to the Control Box module and the integration of them will be tested. Apart from this testing, connection of other modules will be tested with the following procedure. Because there is a strong dependency between the Control Box and Zigbee devices, their integration will be tested first. Second, the integration of Remote Server and

Control Box will be tested because charts are drawn according to the game process. The third integration test process will be done with the integration of the databases on these separate devices. Next, user interface will be integrated and tested considering the interaction with the home appliances and their statistics and commands. Lastly, Embedded Simulation devices will be integrated and tested based on the related modules.

2.2.7 Validation Testing

Result of this test is going to show whether expectations in design is met or not. Our Software Requirements Specification document defines the functional and non-functional requirements of the SmartHome Project. Every single requirement in this document must be considered one-by-one and the software must be guaranteed to satisfy all these requirements. Test cases and scenarios are defined and will be run to ensure the correct working of the project as a whole. The order of validation is the same as integration.

2.2.8 High-Order Testing

Various system tests are going to be held such as:

- Performance Test: Performance of the system will be tested based on Zigbee communication and network speed.
- Alpha/Beta Test: The system is going to be tested by the families and home owners and their adaptation as well as ease of use will be observed.

2.3 Test Metrics

We will use the following metrics:

- Number of Test Cases Executed
- Number of Bugs Detected
- Number of Bugs Fixed
- Number of Priority Bugs Fixed

2.4 Testing Tools and Environment

There are no additional testing tools other than the SmartHome implementation code itself. For embedded device debugging, IAR Embedded Workbench certainly will be used during the development and unit testing; however, different testing environments are required.

First environment type is internet connection. While low bandwidth would be causing lower availability, lesser synchronization with careful analysis can solve this problem. Thus, system must be tested in different internet connection speeds.

Second environment is MySQL Administrator tool, which is a graphical user interface for the project's database management system MySQL. We can see and check the database content after each insert, delete or update operation tests.

3. Test Procedure

3.1 Functional Use Cases

3.1.1 Remote User Interface

3.1.1.1 Sign Up

The aim of this function to enable the user to be registered to the system.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
SU-01	Sign Up for the System	Functional Test	Sign Up for the System first time	1) The user fills related information and clicks submit.	A conformation mail is sent to the e-mail address user provided.
SU-02	Sign Up Duplication	Functional Test	Try to sign up with a duplicate e-mail address.	1) The user tries to register with an e-mail address, which is already registered to the system.	System directs user to forgot password section.

3.1.1.2 Sign In

The aim of this function to enable the user gets related information about his/her devices.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
SI-01	Sign in	Functional Test	Sign in with e-mail and password	1) The user enters related information and clicks enter.	The system directs user to the appliances page.
SI-02	Sign in with wrong information	Functional Test	Sign in with wrong e-mail or password	1) The user tries to enter the system with wrong e-mail or password.	System gives an error message and asks user whether try again or direct to forgot password page.

3.1.1.3 Sign Out

The aim of this function to enable the user gets out of the system in a secure way.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
SO-01	Sign out	Functional Test	Sign out of the system	1) The user clicks log out in order to close the connection.	The system directs user to sign in page.
SO-02	Sign out while the user is not signed in	Functional Test	Sign out while the user is already signed out	1) The user tries to sign out from the system by clicking the sign out button.	The system redirect the user to the home page

3.1.1.4 Forgot Password

The aim of this function to enable the user creates a new password that he/she uses to enter the system.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
FP-01	Forgot Password	Functional Test	Create a new password	1) The user enters his/her e-mail address and click submit.	The system sends an email to the user with a link that helps the user to create a new password for his/her account.
FP-02	Wrong information	Functional Test	Provided e-mail address is not registered	1) The user enters an e-mail, which does not registered to the system.	System gives an error message and asks user whether he/she remembers the correct e-mail or want to contact system administrators.

3.1.1.5 Change Password

The aim of this function to enable the user changes his/her password that he/she uses to enter the system.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
CP-01	Change password	Functional Test	Replace the old password	1) The user enters a new password and clicks change password.	The system shows a verification page about the password changed.
CP-02	Enter different characters	Functional Test	Enter a non-accepted character	1) The user tries to enter a not English character or not a number.	System shows the appropriate character message and asks user to create a new password.

3.1.1.6 Choose a Device

The aim of this function to enable the user to choose a device from the context.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
CD-01	Select a device from menu	Functional Test	Select the device that you want to control	1) The user selects the device and sees its current condition.	The user can manipulate the appliance with given commands.
CD-02	Choose a device without sign in	Functional Test	Try to choose a device before sign in	1) The user tries to select a device before signing in.	System directs user to sign in page automatically.

3.1.1.7 Turn on a Device

The aim of this function to enable the user to turn on the chosen device.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
TON-01	Turn on a device	Functional Test	Select the device that you want to turn on	1) The user clicks on the turn on button and waits for the system to start it.	The system turns on the appliance with the default adjustments and update user panel with new information.
TON-02	Duplicate commands.	Functional Test	Change status of device the same as it is configured	1) The user tries to turn on a device, which is already on.	The system gives a warning about the status of the device.

3.1.1.8 Turn off a Device

The aim of this function to enable the user to turn off the chosen device.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
TOF-01	Turn off a device	Functional Test	Select the device that you want to turn off	1) The user clicks on the turn off button and waits for the system to stop it.	The system turns off the appliance and update user panel with new information.
TOF-02	Duplicate commands.	Functional Test	Change status of device the same as it is configured	1) The user tries to turn off a device which is already off.	The system gives a warning about the status of the device.

3.1.1.9 Adjust

The aim of this function is to enable the user to make the adjustments according to the device capabilities.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
A-01	Adjust the Device	Functional Test	Enter the values that you want to change	1) The user should enter the adjustment type and content into the adjustment panel and click to adjust.	The status of device is updated after update.
A-02	Type and wait	Functional Test	Type content without clicking adjust button.	1) The user types the adjustments and waits without clicking adjust button.	The system does nothing.

3.1.1.10 Update Sensor Info

The aim of this function is to enable the user to update the sensor info anytime s/he wants.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
USI-01	Update Info	Functional Test	Update Device Info	1) The user clicks on the update button.	The system updates the sensor info beside the periodic updating times of the system.
USI-02	Conflict with planned time.	Functional Test	Conflict with periodic time.	1) This command can conflict with the periodic update times.	The system continues with the periodic updating without making extra function call.

3.1.1.11 Add a New Device

The aim of this function is to connect a new home appliance to the system and start controlling it from the remote interface.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
AND-01	Add a Device	Functional Test	Connect a new device to the system	1) The user should connect the device to the system via Xbee and Enter device information to the new device page	The system will save the information of the new device and show the device in the appliances page
AND-02	Add a Device but connection failed	Functional Test	Plug a not working Xbee to the device	1) The user connects the device to the system via Xbee but the wireless connection is not obtained	The system will show a message that the device is not connected yet.

3.1.1.12 View Statistics

The aim of this function is to show the power consumption of the devices over the time.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
VS-01	Show statistics	Functional Test	View statistics page	1) The user clicks on the statistics page and select a period.	The system will show the statistics of the power consumption for the device(s) for the selected period.
VS-02	Show statistics without signing in	Functional Test	Try to see the statistics before sign in	1) The user tries to view the statistics before signing in.	System directs user to sign in page automatically.

3.2 Unit Test Cases

3.2.1 Components and Database(s) Interaction

The aim of this test case is controlling whether current statistic/status data is inserted correctly to remote server and new device info is currently stored & prompted by remote server.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
RES-01	Statistic/Status info received from control box	Unit Test	N/A	1) MySQL client on Control Box (Raspberry) issues an SQL insert operation into Statistics table.	1) Database on MySQL server gets updated. 2) PHP interface of related home appliance displays new status/statistic of device 3) Statistics graph of related device gets updated.
RES-02	New device detection info received from Control Box	Unit Test	N/A	1) MySQL client on Control Box (Raspberry) issues an SQL insert operation into Devices table.	1) The system gives a warning about the new device detection. 2) System prompts user to define or accept device data to the available devices list.
RES-03	New device detection approved by user	Unit Test	N/A	N/A	1) System adds new device to the available devices list in PHP interface as well as Devices table. 2) PHP interface issues a statistic request from Control Box regarding new device.

3.2.2 Control Box Data Flow

The aim of this test case is controlling whether Control Box acts properly as a mediate data flow agent.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
COD-01	Statistic/Status info received from a device	Unit Test	N/A	1) A Zigbee Frame containing statistic/status info received from a valid device.	1) MySQL client on Control Box (Raspberry) issues an SQL insert operation into Statistics table. 2) Statistics synchronization with Remote Server is issued as MySQL client action.
COD-02	Command received from Remote Server	Unit Test	N/A	1) MySQL client on Remote Server issues an SQL insert operation into Commands table.	1) New Command is added to the database. 2) Zigbee Command Sender (python module) prepares required frame and sends it to serial port.
COD-03	New device tried to register itself to Zigbee network	Unit Test	N/A	1) A Zigbee Frame containing device register info received from a previously unregistered device.	1) Control Box checks new device info from the available devices models/types. 2) An SQL insert operation on Raspberry MySQL server is issued to enter a new device to Devices table. 3) Devices synchronization with Remote Server is issued as MySQL client action.

3.2.3 Home Appliance(s) Simulation

The aim of these test cases are to be sure on whether the simulations of the end devices (home appliances) are working properly.

Test Case ID	Test Case Name	Test Type	Test Case Description	Test Steps	
				Step	Expected
SIM-01	Turn on	Unit Test	N/A	1) The frame including the turn on information is given as input to the simulation device which is in close mode currently.	1) The device status should become ON and displayed on the simulation screen.
SIM-02	Turn off	Unit Test	N/A	1) The frame including the turn off information is given as input to the simulation device which is in open mode currently.	1) The device status should become OFF and displayed on the simulation screen.
SIM-03	Adjust	Unit Test	N/A	1) By using the device specific panel screen as simulator, device related adjustments are applied.	1) The simulation truly applies the adjustments, displays them on the simulation screen and send the truly formatted frame as output.
SIM-04	Calculate consumption	Unit Test	N/A	N/A	1) The current consumption should be reflected on the screen and created as a frame which carries consumption information as output.

3.3 Integration Testing

Because tight communication interfaces exist between the modules through the whole system, it is essential to do integration testing while integrating modules to each other. As soon as a new interaction or a new object is defined to be subscribed or published between modules, an integration test is applied to check correctness the communication. Therefore, this procedure lasts as long as the implementation continues.

3.4 Validation Testing

Validation testing is the testing procedure that is enabling compatibility between implementation and the other previous reports i.e. Software Requirement Analysis; Initial and Detailed Design Reports. Therefore, we divided this developer side testing procedure into two main parts that are requirements Validation and Design Validation.

3.5 High-Order Testing

Performance Test: We will test SmartHome on different environments to test the whole system performance. According to active internet upload/download rate, we will try to optimize the data transfer between local controlbox and remote server.

Alpha/Beta Test: When we will release alpha version of our project, namely a home automation system with few available device types, we will test it with our group and an average family and see how well family adapts to the system. Then we will fix any design problems or bugs, and after that we will release the beta version. Our main aim is to release beta version is to share it with the people with low technical background such that they can use it with ease. Thanks to this feedback we will be able to improve the features and fix functional defects in our system.

4. Testing Resources and Staffing

Each module will be tested firstly by its developer, so unit tests will be conducted by the developers of each module. Integration testing will be conducted by the developers of the integrated modules on each integration phase.

Validation testing will be conducted by the developer(s) of the related module(s) with a member who is not a developer of that module for getting the benefit of outside perspectives.

System testing will be conducted by the whole team.

5. Test Work Products and Test Log

The result of the testing process is the identification of bugs. Whenever one of the team members found a bug, he/she will assign the correction to the corresponding developer using TRAC. This provides keeping records of the bugs. Moreover, we will keep the outputs of the testing methods in a log file.

6. Organization and Responsibilities

Main testing resources are the team members. Since they are responsible for both developing and testing, there is a time constraint on testing. Thus, time spent on testing by group members has to be arranged in an efficient way. Although each developer has to test his/her own module during unit testing, another member should be included during the validation-testing phase.

Staff responsibilities for testing different modules are seen below:

Task	Responsible Member(s)
Functional Testing	All Team Members
Unit Testing	All Team Members
Security Testing	Anıl Ulutürk, Abdullah H. T. Bayrakdar
Usability Testing	All Team Members
Performance Testing	ZeynepMavuş, Anıl Ulutürk
Integration Testing	All Team Members
Validation Testing	Abdullah H. T. Bayrakdar, Anıl Ulutürk
High-Order Testing	Anıl Ulutürk, ŞerafettinÖztürk

7. Test Schedule

Task	Start Date	Due Date
Functional Testing	15.04.2013	15.05.2013
Unit Testing	15.04.2013	15.05.2013
Security Testing	01.05.2013	20.05.2013
Usability Testing	25.04.2013	10.06.2013
Performance Testing	10.05.2013	10.06.2013
Integration Testing	10.05.2013	30.05.2013
Validation Testing	10.05.2013	10.06.2013
High-Order Testing	20.05.2013	10.06.2013