

<SmartHome+> Supplementary Specification and Glossary

Revision History

Date	Revision #	Description	Author(s)
2020-08-05	1.0	Supplementary Specification and Glossary	Jiuxiang Chen, Yilin Li, Yilun Sun, Junwei Zhang, Xianqi Zhang
2020-08-06	2.0	Passive voice, weak phrases optimization	Jiuxiang Chen, Yilin Li, Yilun Sun, Junwei Zhang, Xianqi Zhang

Table of Content

1. Introduction.....	4
1.1 Purpose.....	4
1.2 Scope.....	4
1.3 Definitions, Acronyms and Abbreviations.....	4
1.4 References.....	4
1.5 Overview.....	4
1. Functionality.....	4
2.1 System Error Logging.....	5
2.2 Remote Access.....	5
2.3 Energy saving system.....	5
2.4 Housekeeping system.....	5
2.5 Media and entertainment system.....	5
2.6 Multi-device adaptation.....	5
2.7 Environmental Control system.....	5
2. Usability.....	5
3.1 Training.....	5
3.2 User Interface.....	5
3.3. Daily Usage.....	5
3.4 Mean Time between Failures.....	6
3.5 Training.....	6
3.6 User Interface.....	6
5.1 Response time.....	6
3. Reliability.....	6
4.1 Availability.....	6
4.2 Mean Time Between Failures (MTBF).....	6
4.3 Mean Time To Repair (MTTR).....	6
4.4 Accuracy.....	6
4.5 Maximum bugs or defect rate.....	6
4.6 Bugs or defect rate.....	6
4. Performance.....	6
5.2 Throughput.....	6
5. Reference.....	7
5.3 Degradation modes.....	7
5.4 Resource utilization.....	7
6. Supportability.....	7
6.1 Coding standards.....	7
6.2 Naming conventions.....	7
7. Design Constraints.....	7

<i>7.1. Software Language Used.....</i>	<i>7</i>
<i>7.2 Development tools.....</i>	<i>7</i>
8. Online User Documentation and Help System Requirements.....	7
9. Purchased Components.....	8
10. Interfaces.....	8
<i>10.1 User Interfaces.....</i>	<i>8</i>
<i>10.2 Hardware Interfaces.....</i>	<i>8</i>
<i>10.3 Software Interfaces.....</i>	<i>8</i>
<i>10.4 Communications Interfaces.....</i>	<i>8</i>
11. Licensing Requirements.....	8
12. Legal, Copyright and Other Notices.....	8
13. Applicable Standards.....	8
15. Reference.....	10

1. Introduction

Supplementary systems specifications for the requirements are described along the following document. In essence this document helps better identify and elaborate supplementary requirements that are not obvious or properly described in the use cases. It also elaborates on system requirements previously described in the use cases. The supplementary specifications focuses on standards and quality attributes of non-functional requirements and also on context and environment requirements. It will also facilitate legal and regulatory requirements and discuss other system environment related issues.

1.1 Purpose

The supplementary specification defines the system requirements that are not easily defined in the use case model of the SmartHome+ system.

The following roles and role sets use the Supplementary Specifications:

- **Analysts:** create and maintain the Supplementary Specifications, which serve as a communication medium between the analyst, the customer, and developers.
- **Developers:** use the Supplementary Specifications as a reference when defining responsibilities, operations, and attributes on classes, and when adjusting classes to the implementation environment.
- **Implementers:** refer to the Supplementary Specifications for input when implementing classes.
- **Managers:** refer to the Supplementary Specifications for input when planning iterations.
- **Testers:** use the Supplementary Specifications to verify system compliance.

1.2 Scope

- The scope of this supplementary specification is to capture, analyze and elaborate the following requirements, including:
- Functional requirements that were not elaborated in the use case model;
- Non-functional requirements include system quality attributes such as usability, reliability, performance, supportability, as well as legal, regulatory and document requirements;
- Design constraints, including restrictions on design choices, conditions imposed on the development process, as well as prescribed and imposed standards;
- Interfaces include user interfaces, hardware interfaces, software interfaces, communication interfaces.

1.3 Definitions, Acronyms and Abbreviations

Please consult Chapter 14 - Glossary.

1.4 References

Please consult the References Section at the end of the document.

1.5 Overview

The following sections contain information about the corresponding non-functional and functional requirements of the SmartHome+ system. These describe how requirements map to system functions, what the system should achieve, and how to develop methods that meet established non-functional requirements.

1. Functionality

The section will provide details about system features (organized by feature) as per use case specifications. In addition, several functionalities may need a more detailed analysis regarding their usage and may be not covered entirely in use cases.

2.1 System Error Logging

All system errors shall be logged. Fatal system errors shall result in an orderly shutdown of the system.

The System error messages shall include a text description of the error, the operating system error code (if applicable), the module detecting the error condition, a data stamp, and a time stamp. All system errors shall be retained in the Error Log Database.

2.2 Remote Access

All functionality shall be available remotely through an internet connection. This may require applications or controllers running on the personal devices (PC, laptop, Mobile).

2.3 Energy saving system

This system allows the SmartHome+ to control curtains and so on automatically to save electricity and so on.

2.4 Housekeeping system

This system allows the SmartHome+ to help homeowners to feed their pets and do other houseworks automatically .

2.5 Media and entertainment system

This system allows users to have karaoke mode and family cinema mode at home.

2.6 Multi-device adaptation

Android and ios system users can access the smart home system through the app. At the same time, users can operate via mobile phones, tablets, and computers.

2.7 Environmental Control system

The system allows the SmartHome+ to adjust the environmental condition like inside temperature, air quality, User comfort and so on to provide the most suitable living environment.

2. Usability

In this section we will discuss system usability. The system should be easy to use and understandable by the stakeholders.

3.1 Training

System users should be able to use the system as fast as they can. The training process shall be comprehensive and users shall be provided with the instruction book with training prior to using the system and as they go. Initial training will increase usability standards of the system.

3.2 User Interface

System interface shall be compliant to standard UI accessibility features and follow a cognitive design which allows users for better interaction with the system. It also shall follow library inherited UI standards if any.

3.3. Daily Usage

System shall be available and reliable once running. Its daily performance shall remain steady and never degrade over time.

3.4 Mean Time between Failures

System shall be reliable. In event of a fault, error or misconfiguration, fixing the issue in addition to a simple system restart shall resolve the problem. Time between failures shall be no longer than system restart time (application re-launches).

3.5 Training

System users should be able to use the system as soon as possible. The training process should be comprehensive, and training should be provided to users before and during the use of the system. Initial training will improve the usability standards of the system.

3.6 User Interface

The system interface should conform to the standard UI auxiliary functions and follow the cognitive design to enable users to better interact with the system. If any, it should also follow the UI standards inherited by the library.

5.1 Response time

As per Use case the user may alter via graphical controls so the system answer shall be in a short time to the user.

3. Reliability

4.1 Availability

The system is available 100% of the time for the users. The system shall be operational 24 hours a day for 365 days a year less the downtime for maintenance preferably on weekends.

4.2 Mean Time Between Failures (MTBF)

The system used to perform real-time simulation and the mean time between Failures shall exceed 1 year.

4.3 Mean Time To Repair (MTTR)

No specification is in here, but we can presume that the system cannot stay offline for more than 8 hours and any system error shall be addressed immediately by the Smart-Home System report back service. The system is repaired in 7 days after the system fails.

4.4 Accuracy

System weighs the things and converts it in the units like kilograms (kg), pounds. The system should accurately calculate a weight of an object in kgs and pounds. Also, system should accurately count the time in seconds and minutes.

4.5 Maximum bugs or defect rate

The bug rate is being estimated as 600/KLOC

4.6 Bugs or defect rate

There are critical errors such as the database crash due to corruption in the sql. The other errors include any runtime errors or logical errors and syntax errors and bugs.

4. Performance

5.2 Throughput

Researchers use the system for simulation and they can alter it at run-time. That implies a possible high number of transactions per second.

5. Reference

- [1] Mashery, "Driving Real-World Enterprise & B2B Results With APIs", Retrieved 28 December 2012.
- [2] Operating system, https://en.wikipedia.org/wiki/Operating_system, last edited on 2 August 2020.
- [3] Merriam Webster Dictionary, "Performance definition", <http://www.merriam-webster.com>, Retrieved 03 February 2013.
- [4] Linux Information Project, "Source Code Definition"
- [5] The International Organization for Standardization (ISO)/The International Electrotechnical commission (IEC), http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=35733
- [6] The International Organization for Standardization (ISO)/The International Electrotechnical commission (IEC), http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=35786
- [7] The International Organization for Standardization (ISO)/The International Electrotechnical commission (IEC), http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=43073
- [8] The International Organization for Standardization (ISO)/The International Electrotechnical commission (IEC), http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=35867

5.3 Degradation modes

People can press the button directly when it comes to degradation modes.

5.4 Resource utilization

As per use case, machines shall select various algorithms at runtime for comparative studies on various issues. Enough memory must be in place to accommodate that.

6. Supportability

6.1 Coding standards

The system compiles and runs on the Microsoft Windows platform and accommodates other platforms, such as Linux and Mac OS X. The system is written in C++. A Plan shall be there for deployment and building under different platforms and build systems. E.g. Linux with Makefiles and autoconf, or Mac OS X with Xcode or also Makefiles. The standard for each language is not specified.

6.2 Naming conventions

The exported API and global variables should be restricted to the system's own namespace to avoid conflicts with external applications during linking.

7. Design Constraints

7.1. Software Language Used

Portability of the source code (at minimum) and plan for deployment and building under different platforms and build systems, e.g. Linux with Makefiles and autoconf, or Mac.7.2.

7.2 Development tools

There are not enough appropriate algorithms to use.

8. Online User Documentation and Help System Requirements

A plan for academic value of teaching and learning images and physical-based simulations, by structuring the code, comments, and documentation per consistent naming and coding convention and the APU.

9. Purchased Components

No licensing has been specified. No required purchased components to be used.

10. Interfaces

10.1 User Interfaces

The Smarthome+ system provides a user interface on the UI display which is fixed on the wall. The User could operate on User interface and System will automatically connect to each subsection's interface according to their category.(environment consideration, security, ect)

The remote system also provides a user interface application installed on a user's personal device(PC, mobile, laptop) and will connect to the system's user interface by network.

10.2 Hardware Interfaces

Users usually interact with the system through touch screen control, mouse keyboard, or voice. These interactions can be enhanced with reactive controls (ie haptic devices) and voice recognition controls. These devices run under the support of the device driver (provided by the manufacturer to the corresponding OS) and have the correct API hooks for the required functions.

10.3 Software Interfaces

External software interfaces may operate the system using the same API hooks provided for external hardware interfaces. These interactions are restricted to the same functionalities of user I/O for manipulating parameters.

10.4 Communications Interfaces

The SmartHome+ system and remote system will communicate through network cloud. The database component must connect to a cloud server provided by the Company.

11. Licensing Requirements

This system is a collaborative open-source project for academic study purposes. It complies with Concordia University's Code of Conduct and it's under the provisions of a Software Assignment submission.

12. Legal, Copyright and Other Notices

The SmartHome+ system copyright their respective owners. All rights reserved. Permission to use all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation. Copyright content must be submitted to respective owners prior to commercial use. Organizations involved in this project may be consulted.

The SmartHome+ system is intended for academic study only. Researchers involved on this project are not liable for any losses or injuries caused by misuse of this system or the results of this system. All users of this system hereby indemnify any and all responsibilities from the researchers as a result of using the system.

13. Applicable Standards

The SmartHome+ System must follow, meet, and compliant with all ISO/IEC/IEEE standards for Information Technology and Software Engineering, more specifically the following standards: ISO/IEC 25010:2011 Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models [5]

ISO/IEC TR 25060:2010 Systems and software Quality Requirements and Evaluation (SQuaRE) - Common Industry Format (CIF) for usability: General framework for usability-related information [6]

ISO/IEC 26514:2008 Requirements for designers and developers of user documentation [7]

ISO/IEC 90003:2004 Guidelines for the application of ISO 9001:2000 to computer software [8]

14. Glossary

Actor - In UML, someone or something outside the system that interacts with the system.

Alternate flow - The part of a use case that describes its alternative implementations. It's also used to describe error conditions, since errors can be considered a kind of alternative. Also called alternate path. association - In UML, a relationship between an actor and a use case that indicates that the actor interacts with the system by means of the use case.

API – Application Program Interface is a protocol intended to be used as an interface by software components to communicate with each other. [1]

Basic flow - The part of a use case that describes its most common implementation. The basic flow is written assuming that no errors or alternatives exist. Also called basic path or happy day scenario. constraint - A semantic condition or restriction that describes a limitation or state. For example, a constraint might be a limitation on some data's range of values or on some behavior of the application, or it could be a description of a required state of the system at a particular point in time.

Extend - In UML, a relationship from an extending use case to a base use case specifying how the behavior defined for the extending use case can be optionally inserted into the behavior defined for the base use case. include - In UML, a relationship from a base use case to an included use case specifying how the behavior defined for the included use case can be inserted into the behavior defined for the base use case.

Model - A semantically closed abstraction of a system or a complete description of a system from a particular perspective. Examples include use case, architecture, and domain models and code.

Object Management Group (OMG) - An international standards organization that owns and maintains CORBA and UML standards.

Open Source – Is a software development or broader philosophy which promotes free distribution of software design and implementation.

OS - The operating system is system software that manages computer hardware, software resources, and provides common services for computer programs. [2]

Performance - the execution of an action [3]

Postcondition - A constraint that must be true when a use case has ended.

Precondition - A constraint that must be true when a use case is invoked.

Relationship - A semantic connection between model elements. Examples include associations, dependencies, and generalizations. Relationships to use cases include association, extend, and include.

Requirement - A condition or capability to which a system must conform. Requirements are either derived from user needs or stated in a contract, standard, specification, or other formally imposed document.

Scenario - A description of a specific sequence of actions. In use case scenarios, specific persons or actor instances replace actors, and only one path is taken through the use case's possible basic and alternate flows. Also called use case instance.

System - A conceptual entity defined by its boundaries. Examples include companies, divisions, sets of software applications, components, machines, and devices.

Source Code – In computer science, source code is any collection of computer instructions (possibly with

comments) written using some human-readable computer language, usually as text. The source code of a program is specially designed to facilitate the work of computer programmers [4]

Unified Modeling Language (UML) - A graphical language for visualizing, specifying, constructing, and documenting an object-oriented software-intensive system's artifacts.

Use case - In UML, a complete task of a system that provides a measurable result of value for an actor. More formally, a use case defines a set of use case instances or scenarios.

Use case diagram - A UML diagram that shows actors, use cases, and their relationships.

Use case model - A model that describes a system's functional requirements in terms of use cases.

Use case specification - A document that describes a use case. A use

UC – Use Case

UI – User Interface

15. Reference

[1] Mashery, "Driving Real-World Enterprise & B2B Results With APIs", Retrieved 28 December 2012.

[2] Operating system, https://en.wikipedia.org/wiki/Operating_system, last edited on 2 August 2020.

[3] Merriam Webster Dictionary, "Performance definition", <http://www.merriam-webster.com>, Retrieved 03

February 2013.

[4] Linux Information Project, "Source Code Definition"

[5] The International Organization for Standardization (ISO)/The International Electrotechnical commission

(IEC), http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=35733

[6] The International Organization for Standardization (ISO)/The International Electrotechnical commission

(IEC), http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=35786

[7] The International Organization for Standardization (ISO)/The International Electrotechnical commission

(IEC), http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=43073

[8] The International Organization for Standardization(ISO)/The International Electrotechnical commission

(IEC), http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=35867