System Description (SysD) – Arrowhead Zwave System Demonstrator

**Abstract**

This document provides the main template for the System Description of Arrowhead compliant Systems. It should be used to define the main services and interfaces of a system, without describing its internal implementation.

All Arrowhead systems should be specified using this template and stored on a common repository (available on the SVN server), in order to document and formalize the pilot demonstrators and the common Arrowhead framework.

Table of Contents

[Table of Contents 2](#_Toc375649363)

[1. System Description Overview 2](#_Toc375649364)

[2. Diagrams 3](#_Toc375649366)

[3. Application services 3](#_Toc375649367)

[3.1. Produced Services 3](#_Toc375649368)

[3.2. Consumed Services 3](#_Toc375649369)

[4. Security 3](#_Toc375649370)

[5. References 3](#_Toc375649371)

[6. Revision history 3](#_Toc375649372)

[6.1. Amendments 3](#_Toc375649373)

[6.2. Quality Assurance 3](#_Toc375649374)

1. System Description Overview:

The Arrowhead Zwave System Demonstrator is a SOS with a Z-wave Controller, a Z-wave compliant Dasnfoss Thermostat Valve and a Z-wave compliant Fibaro Wall Plug which all behave as service providers in the arrowhead framework and the consumer connect to these devices and consumes their services.

* The Consumer will first send the orchestration request for getting the z-wave device list service to get all the available devices connected to z-wave controller.
* After the list of devices with unique device ids and their types is received from zwave controller service provider, it will use the device ids and device types to request for different kind of services.
* The consumer request get-setpoint-thermo from Thermostat provider after getting the orchestration response for get-setpoint-thermo service to receive the current setpoint value from Thermostat Valve while the Thermostat provider also stores the value with the timestamp into its inherent DataManager.
* The consumer then requests get-set-point-history from Thermostat provider after getting the orchestration response for get-setpoint-history service to receive the records of setpoint value history stored in the Thermostat Provider inherent DataManager.
* Then the consumer requests switch-plug-state from Plug provider after getting the orchestration response for it to First Turn OFF the Switch then Turn it ON after 5 seconds.

Following is the Hardware used for this Project

* + - Raspberry Pi 4.0
    - Z-Wave hat
    - Fibaro Z-Wave electrical Outlet PLUG
    - Danfoss Z-Wave radiator valve Thermostat

1. Behaviour Diagrams

Timeline

Description automatically generated

1. Application services

This system provides four services:

# Produced Services

Table 2 Pointers to IDD documents

|  |  |
| --- | --- |
| Service | IDD Document Reference |
| get-zwave-devices |  |
| get-setpoint-thermo |  |
| get-setpoint-history |  |
| switch-plug-state |  |
|  |  |

# Consumed Services

Table 3 Pointers to IDD documents

|  |  |
| --- | --- |
| Service | IDD Document Reference |
| get-zwave-devices | n/a |
| get-setpoint-thermo | n/a |
| get-setpoint-history | n/a |
| switch-plug-state | n/a |
|  |  |

1. Security

The system is using the HTTPS-SECURE-JSON security interface. Each system both consumer and providers are using their corresponding client certificate for secure communication. Authorization core system is responsible for the access verification and token generation during the provider and consumer interaction.

1. References

<https://github.com/arrowhead-f/core-java-spring>

<https://github.com/arrowhead-f/sos-examples-spring>

1. Revision history

# Amendments

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Date | Version | Subject of Amendments | Author |
| 1 | 2021-01-6 | 0.1 | First draft | Salman Javed |

# Quality Assurance

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Date | Version | Approved by |
| 1 |  |  |  |
| 2 |  |  |  |