Software Requirements Specification

Remote Monitoring of Power Consumption in a Cloud Server Network

Version SRS 1.5 - May 27, 2014

Customer: Patrik Arlos

Chief Executive Officer: Dragos Ilie

Developer Team Zenoss		
Student 1	Name	Sarat Chandra Pasumarthy
	E-Mail	sapa14@student.bth.se
	Social Security Number	921231-0297
Student 2	Name	Manish Reddy Guduru
	E-Mail	magu14@student.bth.se
	Social Security Number	930706-0518
Student 3	Name	Sukesh Kumar Tedla
	E-Mail	sute14@student.bth.se
	Social Security Number	940224-5352
Student 4	Name	Prakhyath Neelagiri
	E-Mail	prne14@student.bth.se
	Social Security Number	930326-1359
Student 5	Name	Hadassah Pearlyn Nagathota
	E-Mail	hana14@student.bth.se
	Social Security Number	930719-9548
Student 6	Name	Preetham Kalakuntla
	E-Mail	prka14@student.bth.se
	Social Security Number	920523-3779

Preface

The document is the sixth release of the Software Requirements Specification document concerned with this project. This document is primarily aimed at the parties involved in this project, that is, the customer and the company involved in providing solution to the customer's needs. The company comprises of Chief Executive Officer (CEO) and the developer team Zenoss. The document starts with glossary and abbreviations which define the technical terms and abbreviations used in the document. Secondly, the section of system architecture gives a high-level view of the system components and their interactions along with further descriptions of the major components or modules of the system. Thirdly, we have the requirements section which comprises subsections of user requirements and system requirements. Lastly, the references section gives a list of the references cited in the document.

This is version SRS1.5 of the document and the version history is as follows:

Released v SRS1.5 on 2014-05-27

- Made changes to Figure 2, Figure 3, Figure 4 for better illustration of the modules.
- Made changes in sections 2.1, 2.2 and 2.3. This is to clear the ambiguity between the modules and have coherence between them.
- Added more information about UFR 3.

Released v SRS1.4 on 2014-05-26

- Made changes in Section 2.2. The change includes more detailed description of the data retrieval.
- Changed Figure3 for better illustration of the module.

Released v SRS1.3 on 2014-05-19

- Included SNMP as one of the data retrieval methods in Section 2.2.
- SFR 3 has been modified by explaining its purpose in a more detailed way and modified regarding tests as well in Section 3.2.1.
- Corresponding changes for SFR 3 and related tests have been made in Table 1 of Section 3.

Released v SRS1.2 on 2014-05-12

- Changed specific terminology in the document according to RFC 2119 in sections 3.1 and 3.2.
- Included the test string numbers which are associated with the software requirements in Table 1. These tests are described in detail in the design document.
- Made change in Section 2.2. It is adding more information about data retrieval.
- Changes made in UFR1, UFR2, UFR4 and UFR6.

Released v SRS1.1 on 2014-05-05

- Modified preface by adding snapshot of the previous versions of the document.
- Included customer-specified probe rate (1 minute).
- Changed figures for modules. Modified description of module 3.
- Added user requirements UFR4 - UFR7. Modified UFR1 - UFR3.
- Removed UNR3 and merged UFR1 with SFR5.

Released v SRS1.0 on 2014-04-28

- Initial release

1. Glossary and abbreviations

API- *Application Program Interface*

An interface, generally specified as a set of operations, that allows access to an application program's functionality.

Cloud Computing

The provision of computing and/or application services over the Internet using a large number of commodity computers and virtualization technology to make effective use of these systems.

CN- *Computer Node*

HTTP- *Hypertext Transfer Protocol*

It is an application protocol for distributed, collaborative and hypermedia information systems.

PDU- *Power Distribution Unit*

It is a device designed to distribute electric power with multiple outputs.

SNMP- *Simple Network Management Protocol*

It is an internet-standard protocol for managing devices on IP networks.

SSH- *Secure Shell*

It is a cryptographic network protocol for secure network services between two networked computers.

UPS- *Uninterruptable Power Supply*

It is an electrical device that provides emergency power when the input power supply fails.

RFC- *Request for comments*

A publication by the Internet Engineering Task Force (IETF) and the Internet Society.

Test Identification string

Used to uniquely identify a test for modules. Follows the following format:

M <Module no.> T <Test no.>

2. System architecture

This section presents a high-level view of the system architecture. The section is further divided into subsections describing and illustrating the major components or modules of the system. Figure 1 is a block diagram gives a brief illustration about the working of the product to be developed. The main aim of the product is to enable the user to monitor the power consumption of a remote device present in the network. The product software runs on the server. The remote devices include but are not limited to UPS and PDUs. The architecture described in this document gives a glimpse of the product that is being developed.

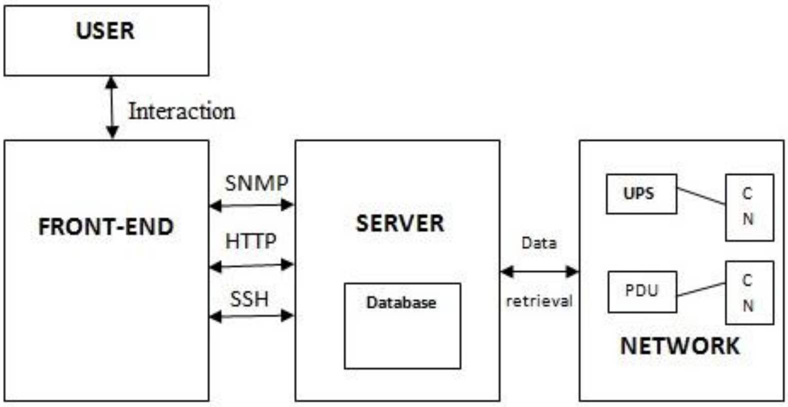

Figure 1: Basic system architecture

2.1 Module 1 : Data Retrieval

This module illustrates the process in which the data related to power is retrieved from the devices present in the network. The product which runs on the server, after getting a request from the user, polls the concerned devices using SNMP, to retrieve the data about the power consumption of the devices. This process is repeated multiple times to get the data every 1 minute in real time scenario. A brief description of the same is illustrated in Figure 2.

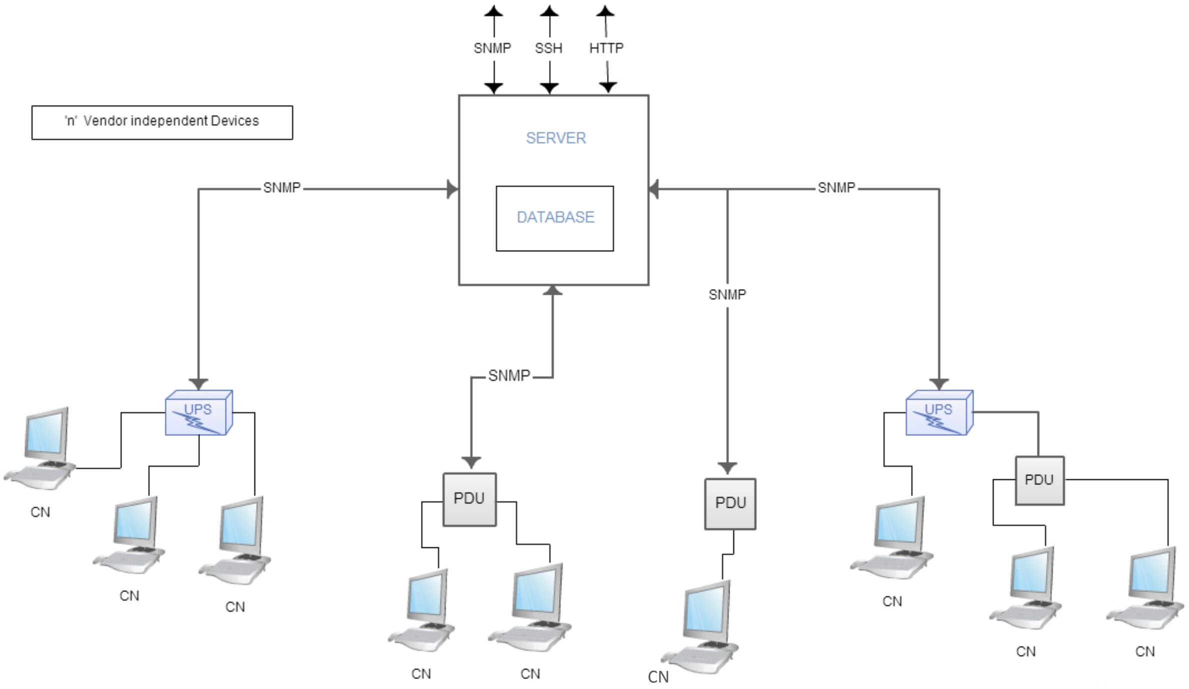

Figure 2: Data retrieval

2.2 Module 2 : Database

This module illustrates the architecture associated with the database which is present in the server. The network devices' information is provided by the customer in the front end. This device information is stored in the database present in the server. The product software which runs on the server, sends SNMP requests to the devices, by considering the credentials from the database. The data, received from the devices as SNMP responses to the product software, is stored in the database. This data is retrievable either through SNMP or SSH or HTTP. This module is described in Figure 3.

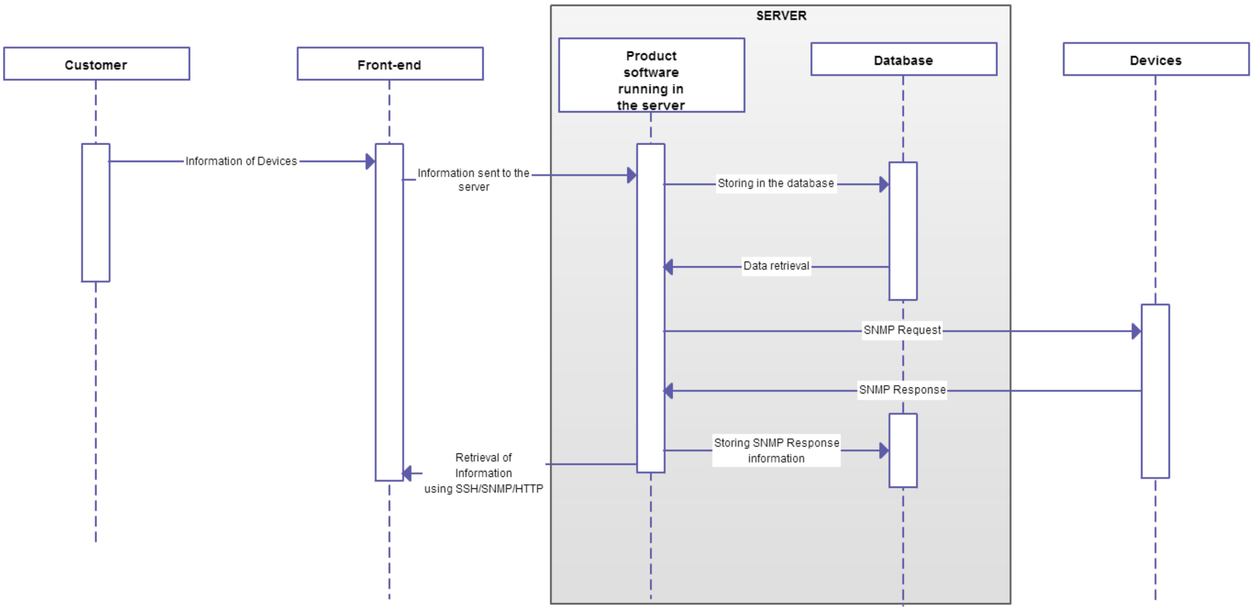
SERVER
Customer
Front-end
Product software running in the server
Database
Devices
Information of Devices
Information sent to the server
Storing in the database
Data retrieval
SNMP Request
SNMP Response
Storing SNMP Response information
Retrieval of Information using SSH/SNMP/HTTP

Figure 3: Database

2.3 Module 3 : Front-end

This module illustrates about the front end details of the product software that is to be developed. The module is depicted as a block diagram in Figure 4. The user can add, edit, select and remove any particular device for which the power consumption is to be monitored. Open source tools are used to create orderly graphs in real time. The graphs are plotted between power consumption and time on different timescales. The product also notifies the user if there are any important developments in the network such as a power failure in any of the device.

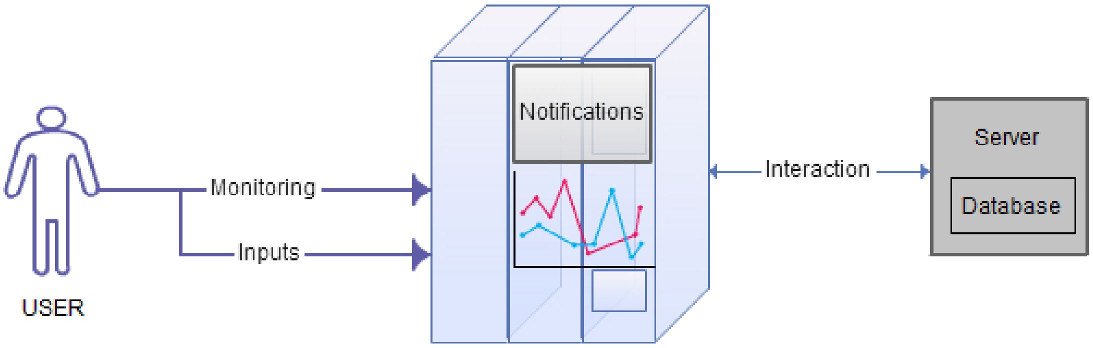

Figure 4: Front End

3. Requirements

The requirements for the project specify both user requirements and system requirements. These are discussed in this section as a list of requirements record. Moreover, they are given a requirement record format for easier and better identification of the requirement. This format mainly consists of an identification string which is based on a classification scheme which is:

Identifier <integer>

Identifier A word or abbreviation that identifies that type of requirement whether it user or system requirement and functional or non-functional requirement.

<integer> For identification of the number of the requirement in the requirement record.

3.1 User requirements

This section describes the services provided for the user by the product. This section is further divided into functional and non-functional requirements subsections.

3.1.1 Functional Requirements:

UFR 1: Functions associated with the device to monitor power

The foremost function is that the user is provided a service to monitor power. Other functions include allowing the user to add, edit, select and remove a device to be monitored. While adding and editing a device to be monitored, the user is required to give the details of the IP address, community names, port details and details about the manufacturer of the device which shall include the name of the vendor. The devices are probed every 1 minute.

Monitoring of power consumption of a device attached to a particular UPS or PDU is possible only if the attached device is either UPS or a PDU. For other device which is neither a UPS nor a PDU, monitoring of power consumption is possible if and only if the power monitoring of output ports of the parent device is possible. Here the parent device is one to which other devices are attached.

UFR 2: Availability of graphs to analyze the power consumption of the devices

The user after selecting a particular device must be able to get graphs indicating the power consumption with respect to time and would be presented on different timescales. The graphs timescales would vary from an hour to a month to a year. The power consumption graphs will vary with the data that is obtained from the devices. The data associated with the graphs would be retained for a year after which the data would be overridden with the new data collected.

UFR 3: Notifications before power failure of a particular device

A notification would be given before a particular device fails. The level of notifications would be varied depending on the severity of the issue. The severity here is based on the battery capacity remaining. The notifications would be displayed on the front end.

UFR 4: Monitoring and representation of other associated parameters:
The other associated parameters of the UPS/PDUs that will be monitored and represented on the front-end will include the following:

- Device status.
- Power usage.
- Battery state.
- Remaining battery.
- Temperature of the battery of the device.

UFR 5: Grouping of devices:
The grouping of the devices is based upon a tree structure. The user must enter the UPS or PDU that is to be monitored. Then the user shall mention the devices attached to that particular UPS or PDU. Thus, one group of devices are formed. The user shall also be able to add, delete and edit the devices which are present in each group. This process is continued for other UPS and PDUs. The notifications of the devices associated with the device present in the group would the same as mentioned in UFR 3.

UFR 6: Handling devices of different vendors:
The product must be able to retrieve data regarding power from devices of various vendors. For this, we request the customer to provide a list of vendors he would be using for the network. This information would help us develop the product in the manner of retrieving the data from the devices. The product shall be able to handle new vendors. This is subject to the information provided by the user while adding new device.

UFR 7: Interaction with Zenoss:
The customer is using Zenoss as the front-end and would like the product to provide a user interface for using the services provided by the product.

3.1.2 Non Functional Requirements:

UNR 1: Easy usability of the product
The product would be in such a manner that it will have an easy-to-use graphical user interface. The customer will be provided with the related user documentation to facilitate the usage of the product.

UNR 2: Maintainability

The product will be provided with service support. The team will take up responsibility to rectify any possible faults and correct them in an effective manner.

3.2 System requirements

This section discusses about the system requirements. These are technical requirements that complement the user requirements and provide information for design and implementation of the system.[1] In the same manner as in user requirements, this section is also divided into subsections of functional and non-functional requirements.

3.2.1 Functional Requirements:

SFR 1: Operating System

The required operating system as part of the system requirements is Ubuntu 12.04 LTS. The desired RAM for the operating system is 2GB.

SFR 2: Web Server

An open source web server will be used to develop the product.

SFR 3: Programming Language

Multiple open source object oriented programming languages will be used to meet the product requirements. The programming languages will be used to design the software according to accepted project proposal.

SFR 4: Database

An open source data management system will be used to meet the module requirements.

3.2.2 Non Functional Requirements:

SNR 1: Testability

The product developed should meet the requirements of the acceptance test plan.

[1] Ian Sommerville, *Software Engineering*.

Requirement	Creation Date	Change Date	Module	Associated test string number	Type
UFR 1	27-04-2014	02-05-2014	3	M3T1, M3T2, M2T1, M2T2	Functional
UFR 2	27-04-2014	02-05-2014	3	M3T5, M3T6, M3T7	Functional
UFR 3	27-04-2014	02-05-2014	3	M3T6, M3T7	Functional
UFR 4	02-05-2014	-	3	M1T1, M2T2, M3T3	Functional
UFR 5	02-05-2014	-	3	M3T5, M3T6, M3T7	Functional
UFR 6	02-05-2014	-	1,2	M2T1,M2T2	Functional
UFR 7	02-05-2014	-	3	M3T1 - M3T7	Functional
UNR 1	27-04-2014	-	3	M3T1 - M3T5	Non-Functional
UNR 2	02-05-2014	-	1, 2, 3		Non-Functional
SFR 1	27-04-2014	-	1, 2	M3T1-M3T7	Functional
SFR 2	27-04-2014	-	1, 2, 3	M1T1, M2T1, M2T2	Functional
SFR 3	27-04-2014	19-05-2014	1, 2, 3		Functional
SFR 4	27-04-2014	-	1, 2	M1T2, M2T1, M2T2	Functional
SNR 1	27-04-2014	-	1,2, 3	M1T1, M2T2	Non-Functional

Table 1 : List of Requirements

4. References

[1]Ian Sommerville. *Software Engineering*. 8th ed., n.d.