

**Final Year Project**

**Final Report**

**Department of Information Technology Network Monitoring System**

**Members:**

**IT114104-2A SIU Man Ho**

**IT114104-2B IU Hiu Pan**

**IT114104-2B TAM Siu Kei**

**IT114104-2B LAU Chun Yin**

Contents

[Introduction 3](#_Toc419982852)

[Program Background 3](#_Toc419982853)

[What is SNMP? 4](#_Toc419982854)

[MIB 5](#_Toc419982855)

[OID 5](#_Toc419982856)

[Agent 5](#_Toc419982857)

[How it works? 6](#_Toc419982858)

[Requirement Specification 7](#_Toc419982859)

[Testing environment’s network diagram 8](#_Toc419982860)

[Tools or techniques implemented 9](#_Toc419982861)

[Use Case Diagram 11](#_Toc419982862)

[Basic information of Use Case 12](#_Toc419982863)

[Fully Dressed Use Cases 15](#_Toc419982864)

[Use Case 1 : Device Status Capture 15](#_Toc419982865)

[Use Case 2 : Usage Trend Chart Generating 16](#_Toc419982866)

[Use Case 3 : Get Capture History 17](#_Toc419982867)

[Use Case 4 : Network Devices Auto Detection 18](#_Toc419982868)

[Use Case 5 : Database Configuration 19](#_Toc419982869)

[Use Case 6 : Database Auto Generating 20](#_Toc419982870)

[Use Case 7 : Delete Specific Device Record 21](#_Toc419982871)

[Use Case 8 : Usage Alert 22](#_Toc419982872)

[Use Case 9 : Get Logging records 23](#_Toc419982873)

[Database Design 24](#_Toc419982874)

[Data Dictionary 25](#_Toc419982875)

[Sequence Diagrams 29](#_Toc419982876)

[Device Status Capture 29](#_Toc419982877)

[Usage Trend Chart Generating 30](#_Toc419982878)

[Get Capture History 31](#_Toc419982879)

[Report Generating 32](#_Toc419982880)

[Network Device Auto Detection 33](#_Toc419982881)

[Database Configuration 34](#_Toc419982882)

[Database Auto Generating 35](#_Toc419982883)

[Delete Specific Record 36](#_Toc419982884)

[Usage Alert 37](#_Toc419982885)

[Get Logging Records 38](#_Toc419982886)

[GUI Design 39](#_Toc419982887)

[Structure 40](#_Toc419982888)

[Test Plan 43](#_Toc419982889)

[Discussion 47](#_Toc419982890)

[Expected Upcoming Progression 47](#_Toc419982891)

[Problems encountered and the corresponding solutions 48](#_Toc419982892)

[Project schedule 50](#_Toc419982893)

[Task 51](#_Toc419982894)

[Task and Role of Each Student 52](#_Toc419982895)

[Task Allocation of Final Report 53](#_Toc419982896)

[Conclusion 54](#_Toc419982897)

[Reference 55](#_Toc419982898)

# Introduction

Program Background

Nowadays, many company using internet to assist their business such as putting their services on internet to let people use their services anytime and anywhere. Therefore, network devices become a very important component, it is in charge of the communication within the company and the access from the outside of the company. If there are some network problem happened. For example, the server or router down because of it can't cope with the heavy workload. It will affect the normal function of the company and it may cause huge economic losses.

Therefore, it is important to keep acquire network devices’ status, determine the load of devices to remedy when problems occurred and to plan for upgrades or changes. However, there are many devices in the network, it is very hard for the administrator to check all these devices physically.

Thus, we are now developing a Network Monitoring System using a protocol called SNMP which is a service using for getting status from devices or other network devices and we are going to use version 2c for our system. This system will be implement in a Management PC (Windows 7) to collect the information of network deivecs centrally so that it can help the administrator to keep in touch with those devices. The system will also analyze those status and represent it in a simple way, such as graph, to help administrator understanding these information easily.

What is SNMP?

**Simple Network Management Protocol (SNMP)** is an "[Internet-standard protocol](http://en.wikipedia.org/wiki/internet_protocol_suite) for managing devices on [IP](http://en.wikipedia.org/wiki/internet_protocol) networks". Devices that typically support SNMP include routers, switches, access servers, hub, workstations, printers, modem racks and computer hosts.

SNMP is used mostly in [network management systems](http://en.wikipedia.org/wiki/network_management_systems) to [monitor](http://en.wikipedia.org/wiki/network_monitoring) network-attached devices for conditions that warrant administrative attention. SNMP is a component of the [Internet Protocol Suite](http://en.wikipedia.org/wiki/internet_protocol_suite) as defined by the [Internet Engineering Task Force](http://en.wikipedia.org/wiki/internet_engineering_task_force) (IETF). It consists of a set of [standards](http://en.wikipedia.org/wiki/technical_standard) for network management, including an [application layer](http://en.wikipedia.org/wiki/application_layer) [protocol](http://en.wikipedia.org/wiki/protocol_(computing))(Layer 7), a database [schema](http://en.wikipedia.org/wiki/logical_schema), and a set of [data objects](http://en.wikipedia.org/wiki/data_object). For the SNMP, the request will sent out through the network to the target. It agent receives requests on UDP port 161.

**Versions of SNMP**

**SNMPv1**

SNMPv1 is the first version of SNMP, it only require a string community for identifying. This version are supported by most device. However, there are some disadvantages, the community are sent in plaintext, so it is not secured enough. Moreover, SNMPv1 only supports 32bit counters, it is not enough for bandwidth monitoring with high loads.

**SNMPv2c**

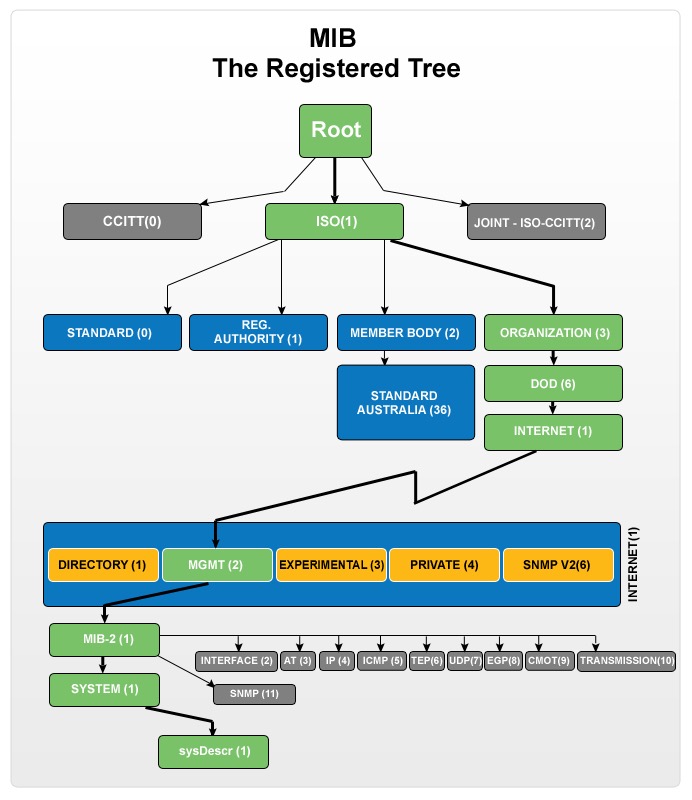
SNMPv2c is identical to version 1, expect it added a 64 bit counters. Therefore, it have the some situation with SNMPv1 which is limited security. Most devices support SNMPv2c nowadays and it is the most common one.

**SNMPv3**

SNMPv3 adds authentication and encryption. It need user account for authentication. Also, it can encrypt the data packet. Therefore, it is more secured than just defining a community string.

MIB

MIB is a collection of information used for managing devices in the network. Network devices have a database called ‘MIB’ with the set of ‘objects’. These object stores valuable information like memory status, hardware status, etc. within the network device. There ‘objects’ are identified by ‘OID’.



OID

OID is a set of number which used for identifying objects in MIB. For example, as you can see, every directory got a number, link up the numbers, we can get a OID or Object Identifier for example, .1.3.6.1.2.1.1.1 which represent sysDescr in the MIB. Agent will use it to find out appropriate object in MIB.

Agent

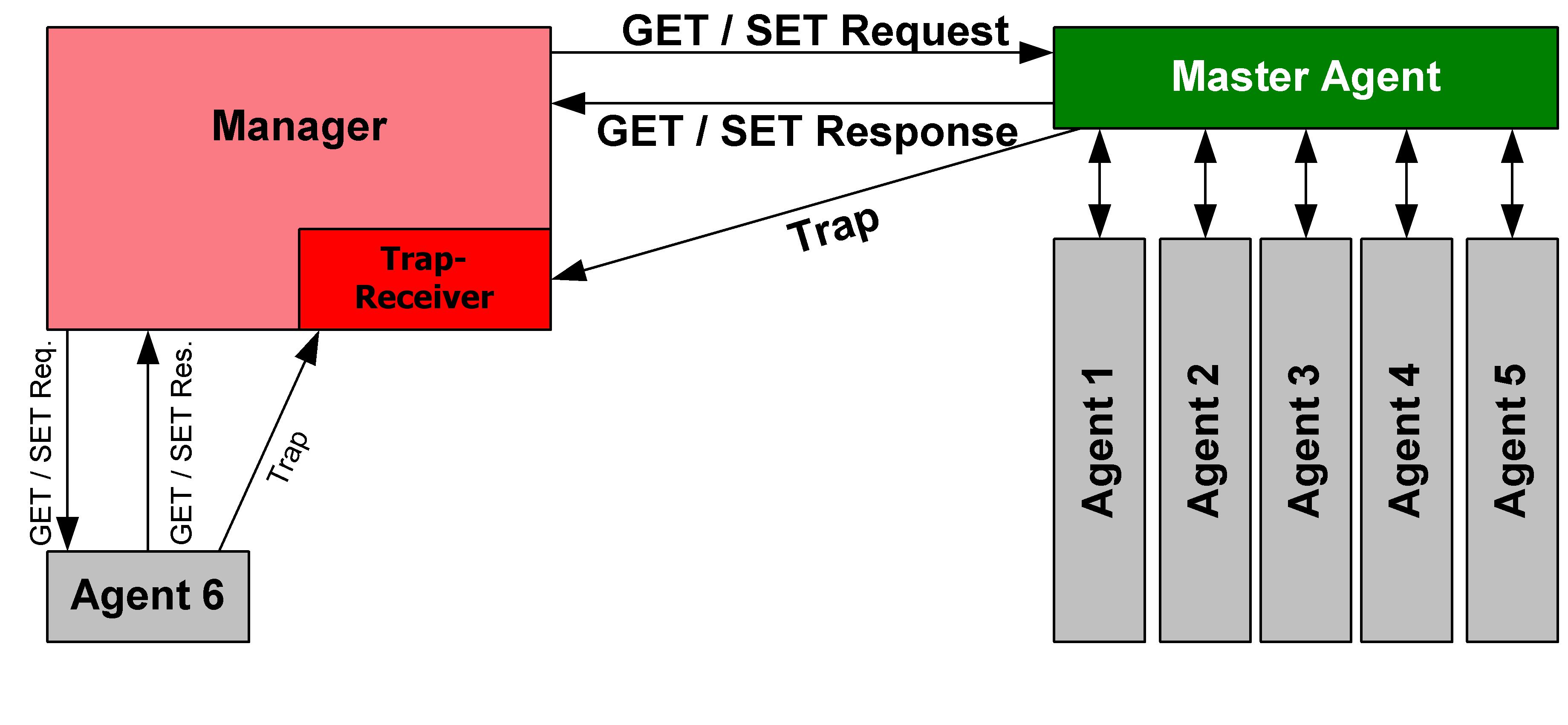
Agent is a program that installed in the device, it like a middleman helping us to communicate with device. After it received a request, it will collect the appropriate object in MIB with the OID and response it to the sender.

**Community**

Community is a set of string, it is like a user password that allow access to devices for getting status. Community will be configured in the device. Device will response the request if the community correct.

How it works?

For example, if the sender want to get the system name of your device, you can send a request to the device with the OID [1.3.6.1.2.1.1.5](http://www.alvestrand.no/objectid/1.3.6.1.2.1.1.5.html)(sysName) and a correct community. The agent will receive the request at UDP port 161. After that, agent will check the community. If the community is correct, it will use the OID to locate the target object. After locating, it retrieves that information from the MIB database and response it to sender.



# Requirement Specification

**Devices status capturing (Manually)**

This is the main feature of the program, we can get status form devices such as Linux server, Windows 7, Window Server 2008 R2, Cisco Router and Switch with the IP address.

**Devices status capturing (Automatically)**

The system will get the status from devices every 10 second automatically and show on the GUI. Record will insert to database every 5 minutes.

**Graphic User Interface**

Create a clearly user interface for user, the GUI will represent the current status that we captured with graphics. To let the user understand those status easily.

**Status Statistics**

Get status with specific period of time, and generate a line chart to represent those statuses.

**Record or Logging History**

Get all the status records or logging records of the device and output it into the text file or excel.

**Network Scanning**

User can input a target network or a range of IP addresses for auto detecting devices with SNMP service.

**Alerting**

User can set the cpu or ram usage limitation. For example, user input the number 70 for ram usage limitation. If the ram usage is over 70%, the system will alert the user.

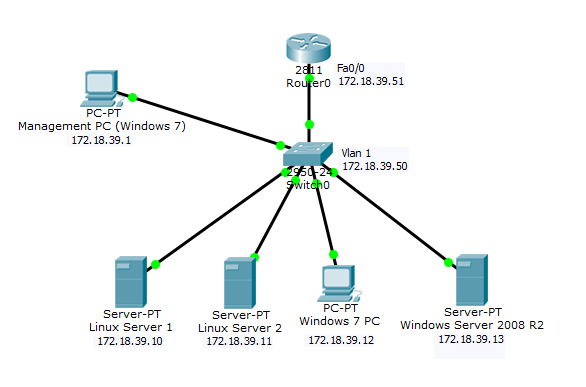
**Logging**

The system will log down the alert type and time to the database when cpu or ram usage is over the limitation that set by the user.

**DBMS for storage the status**

Every time the program capture status from devices, it will save the device status to DBMS for future use.

# Testing environment’s network diagram



**Manager:**

This computer’s role is a Manager. The OS of this computer is Window 7. We will install the network monitoring system and the database on it.

**Target devices:**

* Cisco Switch
* Cisco Router
* Linux Centos 7
* Windows Server 2008 R2
* Windows 7

These are the target devices.

We will install the SNMP client program to Linux and activate the SNMP service of Windows and Cisco devices.

# Tools or techniques implemented

**Eclipse**

Eclipse is an integrated development environment (IDE). It contains a base workspace and an extensible plug-in system for customizing the environment.

**MySQL Community Edition**

MySQL Community Edition is the freely downloadable version of the world's most popular open source database. It is available under the GPL license and is supported by a huge and active community of open source developers.

**JAVA SE**

Java SE can provide us to develop and deploy Java applications on desktop and servers.

**SNMP4J**

SNMP4J is an enterprise class free open source and state-of-the-art SNMP implementation for Java. In our program , we will use it to capture the status from the device.

**JFreeChart**

JFreeChart is a free 100% Java chart library that makes it easy for developers to display professional quality charts in their applications. In our program, it use to generate the chart from our data.

**JDBC**

JDBC is a free library provided by Oracle. It use to connect MySQL server with the

java Program.

**Apache-POI**

The Apache POI Project's provides pure Java libraries for reading and writing files in Microsoft Office formats, such as Word, PowerPoint and Excel. In our program, it used to generate report and output the capture record from the database.

**Net-Snmp(For Linux and agent software)**

Net-SNMP is a suite of applications used to implement SNMP v1, SNMP v2c and SNMP v3 using both IPv4 and IPv6.

**Linux Cent OS 7**

The client will install the CentOS7 as a server. We will install the Net-snmp and configure the SNMP service in here to make a part of testing environment.

**Windows 7**

The manager will install the windows 7. We will install the database and the Network Monitoring System in here to make a part of testing environment.

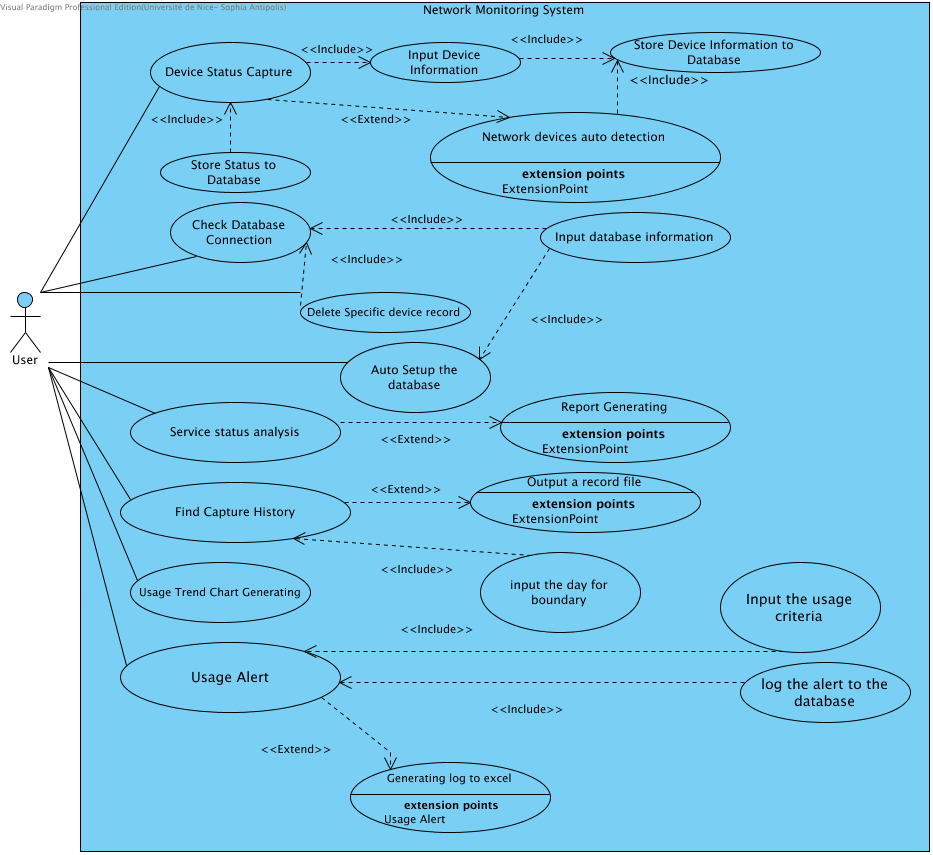
**Windows Server 2008 R2**

Target device for status capturing, a part of testing environment.

**Cisco Switch and Router**

Target device for status capturing, a part of testing environment.

# Use Case Diagram



Basic information of Use Case

Image1.png Auto Setup the database

Actors: User

Include by: Input database information

Image2.png Check Database Connection

Actors: User

Include by: Delete Specific device record,Input database information

Image3.png Delete Specific device record

Actors: User

Include: Check Database Connection

Image4.png Device Status Capture

Actors: User

Include: Device Status Capture,Input Device Information

Include by: Store Status to Database,Device Status Capture

Extend from: Network devices auto detection

Image5.png Find Capture History

Actors: User

Include by: input the day for boundary

Extend from: Output a record file

Image6.png Generating log to excel

Extend: Usage Alert

Image7.png Input Device Information

Like IP address

Include: Input Device Information,Store Device Information to Database

Include by: Device Status Capture,Input Device Information

Image8.png Input database information

Include: Auto Setup the database,Check Database Connection

Image9.png Input the usage criteria

Include: Usage Alert

Image10.png Network devices auto detection

Include: Store Device Information to Database

Extend: Device Status Capture

Image11.png Output a record file

Extend: Find Capture History

Image12.png Report Generating

Extend: Service status analysis

Image13.png Service status analysis

Actors: User

Extend from: Report Generating

Image14.png Store Device Information to Database

Include by: Network devices auto detection,Input Device Information

Image15.png Store Status to Database

Include: Device Status Capture

Image16.png Usage Alert

Actors: User

Include by: log the alert to the database,Input the usage criteria

Extend from: Generating log to excel

Image17.png Usage Trend Chart Generating

Actors: User

Image18.png input the day for boundary

Include: Find Capture History

Image19.png log the alert to the database

Include: Usage Alert

Image20.png User

Use Cases: Find Capture History,Usage Alert,Delete Specific device record,Usage Trend Chart Generating,Auto Setup the database,Check Database Connection,Service status analysis,Device Status Capture

# Fully Dressed Use Cases

Use Case 1 : Device Status Capture

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can capture the device status on the local network ,store data to database. |
| Preconditions | * MySQL had installed. * SNMP service had activated. * Complete “Database Configuration”. |
| Main Success Scenario: | 1. User press “New” to new a target device. 2. User input the target device IP and community.   3. System send request target for getting status.  4. Status will store into database.  5. Status will show on GUI. |
| Extensions: | 1. The IP address input dialog is blank 2. System prompts “Input Empty is invalid” to user that needs to enter the IP. 3. The community input dialog is blank 4. System prompts “Input Empty is invalid” to user that needs to enter the community. |
| Frequency of Occurrence: | Frequently happen |

Use Case 2 : Usage Trend Chart Generating

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can get status with specific period of time. |
| Preconditions | * MySQL had installed. * Device status exist in database. * Complete “Database Configuration”. |
| Main Success Scenario: | 1. User can press ”Usage Trend” button.  2. User choose the target type of status.  3. User choose the date.  4. User click “Ok” button.  5. System gets the target record from database.  6. User can see a line chart pop out. |
| Extensions: | 1. The user didn’t choose the target type 2. System prompts “Please choose the type” to user that needs to choose the type of record. 3. The user didn’t choose the target date 4. System prompts “Please choose the date” to user that needs to choose the target date. |
| Frequency of Occurrence: | Happen when user want to get records statistics. |

Use Case 3 : Get Capture History

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can get all record form database. |
| Preconditions | * MySQL had installed. * Complete “Database Configuration”. |
| Main Success Scenario: | 1. User can press ”History” button  2. User choose the target device.  3. User click “Ok” button.  4. System get all the record of target device.  5. System generates a text file or excel file with all  records of target device.  6. The file generate to the directory of the system. |
| Extensions: | 1. The user didn’t choose the target device 2. System prompts “Please choose the target device” to user that needs to choose the target. |
| Frequency of Occurrence: | Happen when user want to get all records |

Use Case 4 : Network Devices Auto Detection

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can scan out devices in the target network. |
| Preconditions | * SNMP service of device had activated |
| Main Success Scenario: | 1. User press ”Scan IP address” button.  2. User input the target network with subnet mask.  3. User input the community.  4. System scan the target network to find the devices which is available.  5. Available devices will be added to the list. |
| Extensions: | 1. The user need to input a valid network or range of IP.   System prompts “Please input a valid IP address” to user that needs to retype the IP address. |
| Frequency of Occurrence: | Frequently happen |

Use Case 5 : Database Configuration

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can connect to exist database. |
| Preconditions | * MySQL had installed. * Database are exist. |
| Main Success Scenario: | 1. User press ”Database Configuration” button.  2. User input the username and password of MySQL.  3. User input database’s URL  4. System prompts “This is connected” if above input data are corrected. |
| Extensions: | The user input incorrect database username, password or the target database are not exist.  System prompts “This is not connected”.  User need to reconfigure. |
| Frequency of Occurrence: | Frequently happen |

Use Case 6 : Database Auto Generating

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can create a database to MySQL Server |
| Preconditions | * MySQL had installed. * Complete “Database Configuration”. |
| Main Success Scenario: | 1. User press ”Database Auto Generating” button.  2. User input Database name.  3. A Message box will pop out to show it is success or fail |
| Extensions: | The user use the “Database Auto Generating” function without complete “Database Configuration” first, system will alert user “Please Complete the Database Connection first.” |
| Frequency of Occurrence: | Frequently happen |

Use Case 7 : Delete Specific Device Record

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can delete all record of target device. |
| Preconditions | * MySQL had installed. * Complete “Database Configuration”. |
| Main Success Scenario: | 1. User press ”Delete Specific Record” button.  2. User choose the target IP address.  3. A Message box will pop out to show it is success or fail |
| Extensions: | The user use the “Delete Specific Record” function without complete “Database Configuration” first, system will alert user “Please Complete the Database Connection first.” |
| Frequency of Occurrence: | Frequently happen |

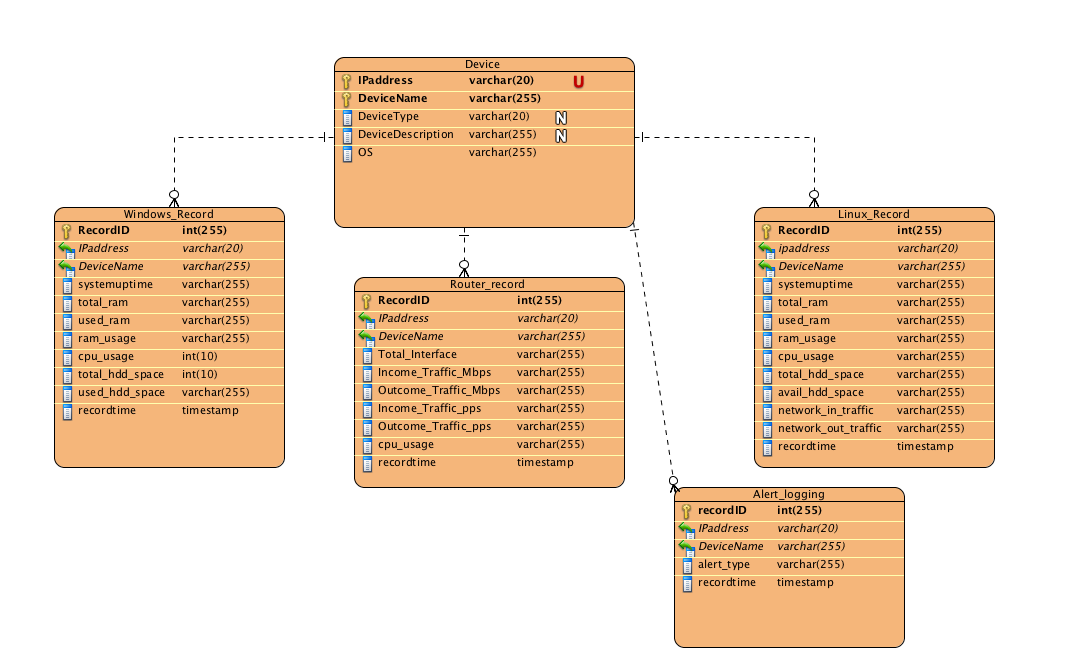
Use Case 8 : Usage Alert

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can set the cpu or ram usage limitation. If the usage is over the limitation, the system will alert the user. |
| Preconditions | * Capturing target is exist. * SNMP service had activated. |
| Main Success Scenario: | 1. User click on the tab of target device. 2. User press “Alert Setting” to set the limitation 3. User choose the type. 4. User input the percentage. 5. User click “OK” and the value will change. 6. When the usage is over the limitation, system will alert the user and log down the record to database. |
| Extensions: | 1. The input is over 100 or less then 0 2. System prompts “Input must between 0 and 100” to user that needs to enter the limitation. |
| Frequency of Occurrence: | Frequently happen |

Use Case 9 : Get Logging records

|  |  |
| --- | --- |
| Primary Actor: | User |
| Stakeholders and Interests: | User: User can get all logging records form database. |
| Preconditions | * MySQL had installed. * Complete “Database Configuration”. |
| Main Success Scenario: | 1. User can click ”Logging records” button.  2. User choose all the records or specific target..  3. User click “OK” button.  4. User choose the target device records.  5. User click “OK” button.  6. The file generated to the directory of the system. |
| Extensions: | 1. The user didn’t choose the target device 2. System prompts “Please choose the target” to user that needs to choose the target. |
| Frequency of Occurrence: | Happen when user want to get logging records |

# Database Design

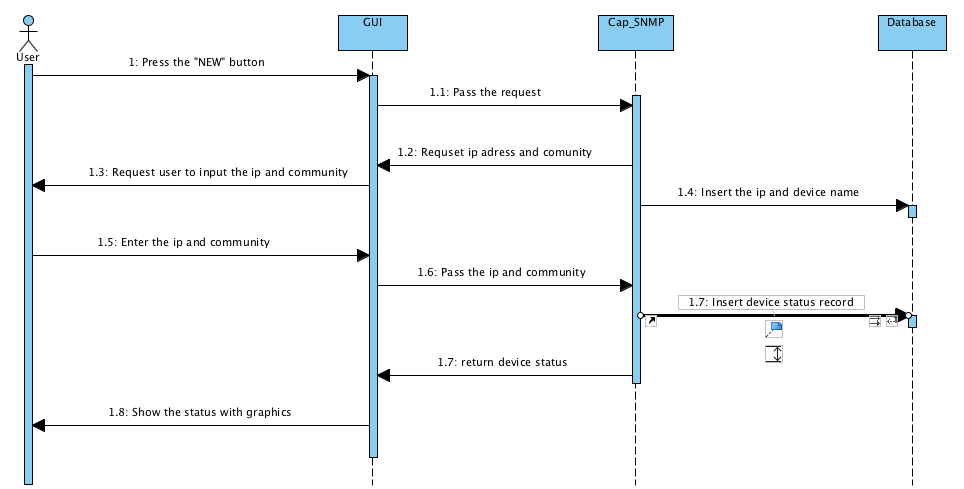


Data Dictionary

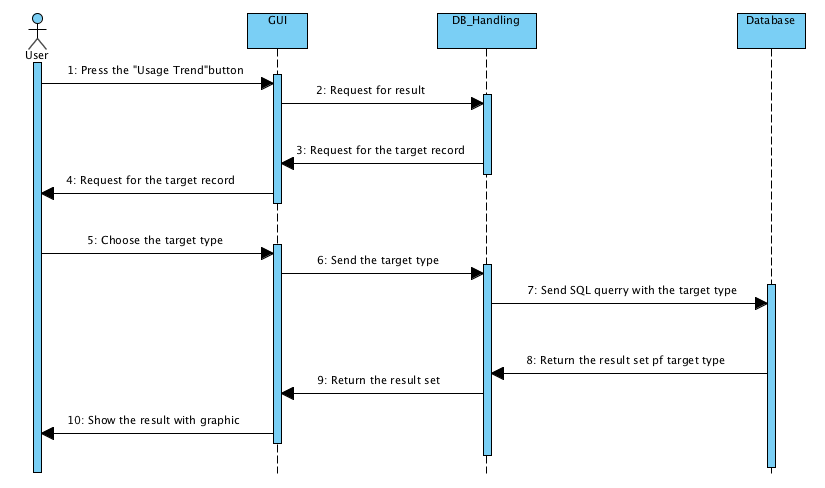
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Entity Name | | Entity Description | | | | | |
|  | Column Name | Column Description | Data Type | Length | Primary Key | Nullable | Unique |
| Alert\_logging | |  | | | | | |
|  | recordID |  | int | 255 | true | false | false |
|  | IPaddress |  | varchar | 20 | false | false | false |
|  | DeviceName |  | varchar | 255 | false | false | false |
|  | alert\_type |  | varchar | 255 | false | false | false |
|  | recordtime |  | timestamp | 0 | false | false | false |
| Router\_record | |  | | | | | |
|  | RecordID |  | int | 255 | true | false | false |
|  | IPaddress |  | varchar | 20 | false | false | false |
|  | DeviceName |  | varchar | 255 | false | false | false |
|  | Total\_Interface |  | varchar | 255 | false | false | false |
|  | Income\_Traffic\_Mbps |  | varchar | 255 | false | false | false |
|  | Outcome\_Traffic\_Mbps |  | varchar | 255 | false | false | false |
|  | Income\_Traffic\_pps |  | varchar | 255 | false | false | false |
|  | Outcome\_Traffic\_pps |  | varchar | 255 | false | false | false |
|  | cpu\_usage |  | varchar | 255 | false | false | false |
|  | recordtime |  | timestamp | 0 | false | false | false |
| Windows\_Record | |  | | | | | |
|  | RecordID |  | int | 255 | true | false | false |
|  | IPaddress |  | varchar | 20 | false | false | false |
|  | DeviceName |  | varchar | 255 | false | false | false |
|  | systemuptime |  | varchar | 255 | false | false | false |
|  | total\_ram |  | varchar | 255 | false | false | false |
|  | used\_ram |  | varchar | 255 | false | false | false |
|  | ram\_usage |  | varchar | 255 | false | false | false |
|  | cpu\_usage |  | int | 10 | false | false | false |
|  | total\_hdd\_space |  | int | 10 | false | false | false |
|  | used\_hdd\_space |  | varchar | 255 | false | false | false |
|  | recordtime |  | timestamp | 0 | false | false | false |
| Linux\_Record | |  | | | | | |
|  | RecordID |  | int | 255 | true | false | false |
|  | ipaddress |  | varchar | 20 | false | false | false |
|  | DeviceName |  | varchar | 255 | false | false | false |
|  | systemuptime |  | varchar | 255 | false | false | false |
|  | total\_ram |  | varchar | 255 | false | false | false |
|  | used\_ram |  | varchar | 255 | false | false | false |
|  | ram\_usage |  | varchar | 255 | false | false | false |
|  | cpu\_usage |  | varchar | 255 | false | false | false |
|  | total\_hdd\_space |  | varchar | 255 | false | false | false |
|  | avail\_hdd\_space |  | varchar | 255 | false | false | false |
|  | network\_in\_traffic |  | varchar | 255 | false | false | false |
|  | network\_out\_traffic |  | varchar | 255 | false | false | false |
|  | recordtime |  | timestamp | 0 | false | false | false |
| Device | |  | | | | | |
|  | IPaddress |  | varchar | 20 | true | false | true |
|  | DeviceName |  | varchar | 255 | true | false | false |
|  | DeviceType |  | varchar | 20 | false | true | false |
|  | DeviceDescription |  | varchar | 255 | false | true | false |
|  | OS |  | varchar | 255 | false | false | false |

# Sequence Diagrams

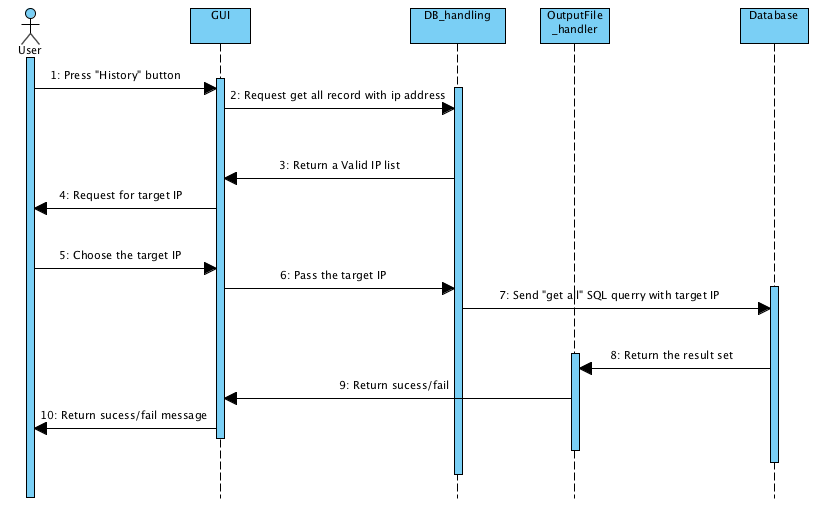
Device Status Capture



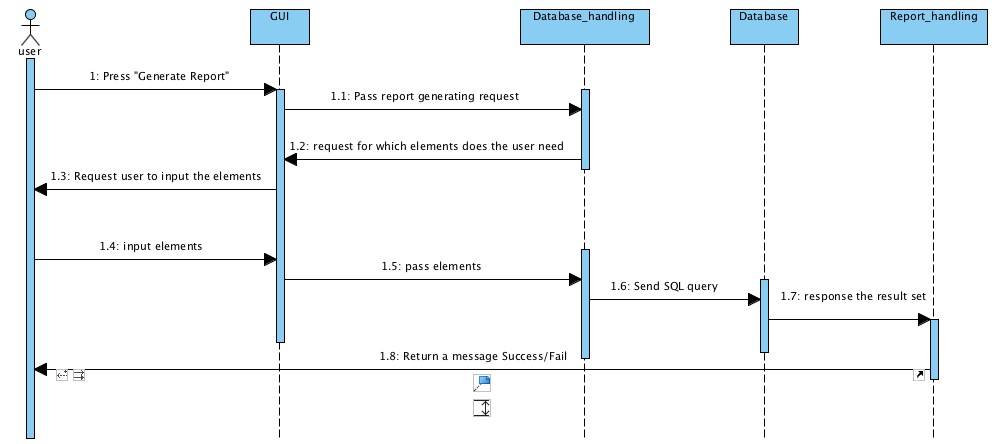
Usage Trend Chart Generating



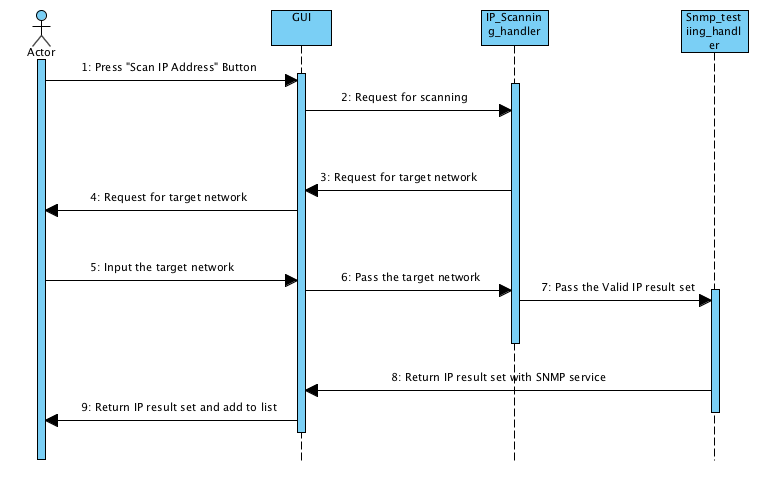
Get Capture History



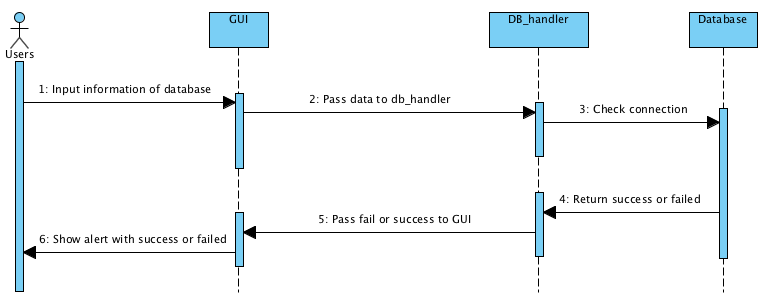
Report Generating



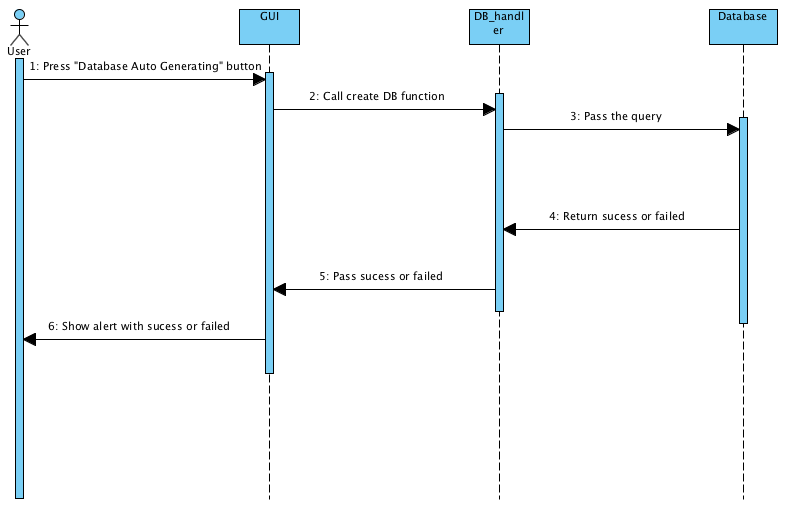
Network Device Auto Detection



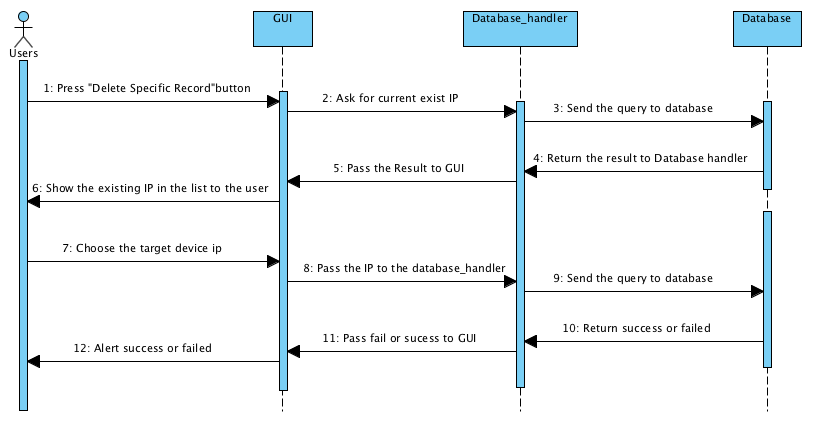
Database Configuration

****

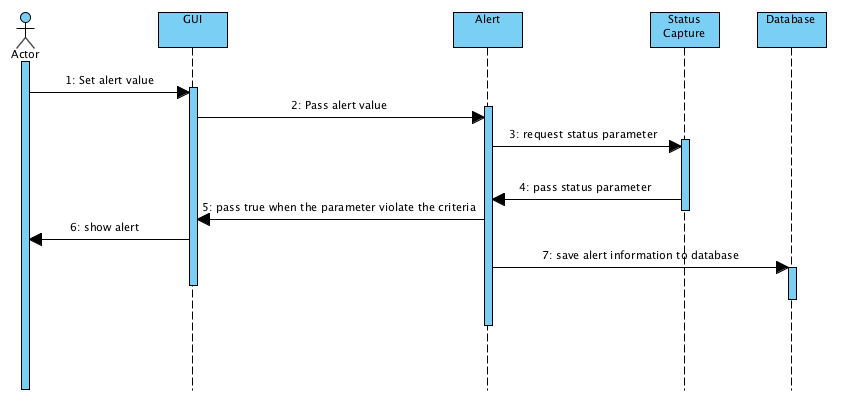
Database Auto Generating



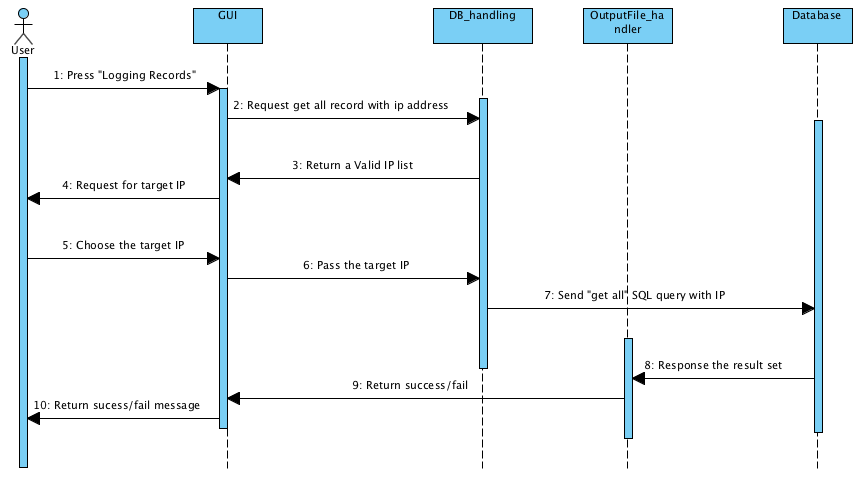
Delete Specific Record

****

Usage Alert

****

Get Logging Records



# GUI Design

Our Network Monitoring System will have the Graphical User Interface to the users. The aim is to make the end-users can control our system easily since end-users did not need to study the related command. Therefore, we design our system GUI before we have referred the related monitoring system interface such as SolarsWinds and Microsoft Network Monitor.

The GUI of Network Monitoring System will included the following components:

1. Menu Bar
2. Image Icon Button
3. Left Panel
4. Right Panel

For the Menu Bar, we were designed four menu item such as ‘File’, ‘Edit’, ’View’ and ‘Help’ which also included the sub menu item, such as following:

* File 🡪 New, Report and Exit
* Edit 🡪 Database Configuration, Database Auto Creation, Delete specific record and Clear
* View 🡪 Auto Capture 🡪 Start Capture and Stop Capture
* Help 🡪 About US and Webpage

Then, about the Image Icon Button, we provided Eight-image button. They are Start/Stop Capture, New, Refresh, Clear, History, Logging records and Alert setting button. Some button functions will be same with the menu bar item functions.

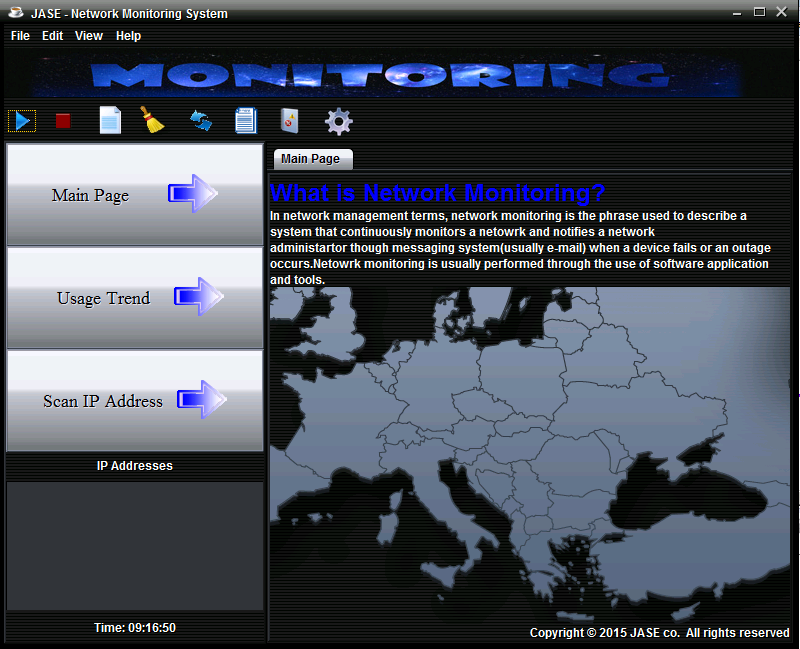
Regarding the Left Panel, we have three buttons. ‘Main Page’, ‘ Usage Trend’ and ‘Scan IP Address’. Also, in the left panel, we also have the device IP address and the real time timer.

For the right panel, the main purpose is to provide the display boxes to show the messages and chart.

Structure

Menu Bar

Image Icon Button



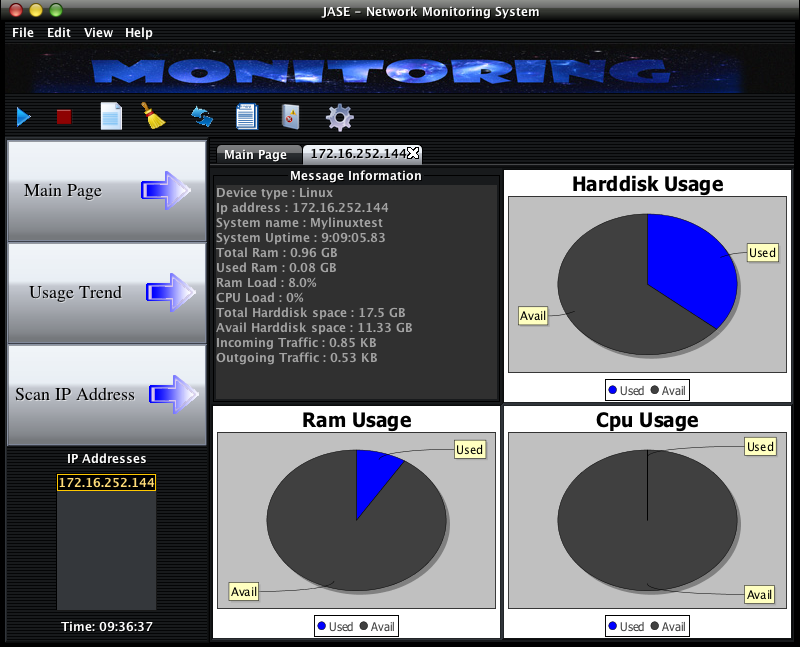
Button

Device List

Real time timer



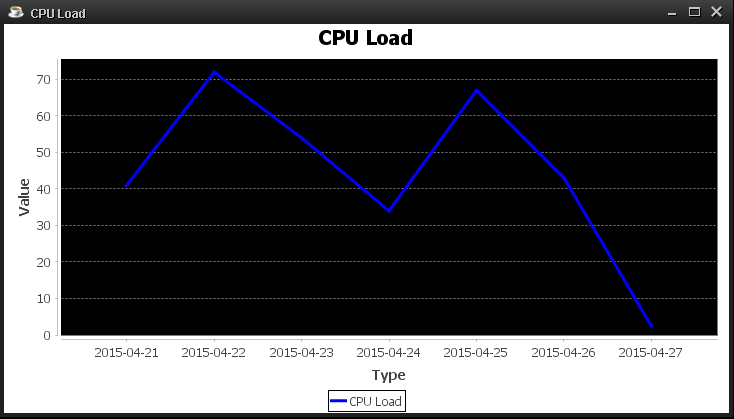
“Webpage”



Chart

Status

**Result**

****

Line Chart of CPU Load

You will see the IP address (172.16.252.144) on the device IP list and the related information regarding the IP address will be show on the right side. It contains the message information, Incoming Traffic, hard disk usage, ram usage and CPU usage,etc.

# Test Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test Description | Test Procedure | Expected Result | Pass/Fail |
| A01 | User can get status from device successfully | 1.Click “New” button  2.Enter the target IP and Community  3.Get the status | The status will show on the GUI | Pass |
| A02 | User can see the Chart | 1.Click “New” button  2.Enter the target IP and Community  3.Generate the Chart | The information will show with chart | Pass |
| A03 | User can start the automatic button | 1.Click” ”button | The status will be updated | Pass |
| A04 | User can stop the automatic button | 1.Click” "button | The status will stop updating | Pass |
| A05 | Website button function | 1.Click”Web Page” button in menu bar | the function will show the website in the default browser | Pass |
| A06 | About Us button function | 1.User click the “About Us” button in the menu bar | Pop up a Message Dialog and show the Contact information | Pass |
| A07 | Exit button function | 1.User click the “Exit” button in the menu bar  2. Click “Yes” in the confirm dialog. | The program will be cancelled | Pass |
| A08 | User can clear the current status that showed on the GUI | 1. User click the “Clear” button. | The status showed on the GUI will be cleared. | Pass |
| A09 | User can refresh the status of device | 1. User click the ”Refresh” button. | The status showed on the GUI will be changed | Pass |
| A10 | User can get status with specific period of time | 1.User click the “Usage Trend” button.  2. Choose the time and the target status.  3. Click “Ok” button. | A line chart with the status record will pop out. | Pass |
| A11 | User can get all the record from target device | 1.User click the  ”History” button  2.Choose the target IP  3.Choose the type of file  4.Click “OK” | A file with all records of the target device will generate to directory of the system | Pass |
| A12 | User leave the input dialog blank when they creating a target. | 1.User click the “New” button  2.Leave the input dialog blank  3.Click “OK” | A message dialog will pop out and request user to input the IP address again | Pass |
| A13 | User leave the target and date blank when they using the “Usage Trend” function | 1.User click the “History”.  2. Leave the target and date blank  3.Click “Ok” button | A message dialog pop out to inform the user that he need to choose the target and date | Pass |
| A14 | User can scan the network to discover devices. | 1.Click the “Scan IP Address” button  2.Input the IP Scanning boundary  3.Input the community | Show the IP which SNMP service has activated and connection has established to the GUI IP list. | Pass |
| A15 | Database Configuration  Button (Correct Input) | 1. Click Database Check Connection  2. Input the username  3. Input the password  4. Input the URL of database | Show alert with success  Message | Pass |
| A16 | Database Auto Generating Button  (Correct Input) | 1.Click Database Auto Generating button  2.Input the database name | Show alert with success message | Pass |
| A17 | Delete Specific Record button | 1.Click Delete Specific Record button  Select the target IP | 1.Show alert with success message of failed message  2.Database has remove target record | Pass |
| A18 | Database Configuration  Button (Incorrect Input) | 1. Click Database Check Connection  2. Input the username(incorrect)  3. Input the password(incorrect)  4. Input the URL of database(incorrect) | 1.Show alert with connection failed message | Pass |
| A19 | Database Auto Generating Button  (Empty Input) | 1.Click Database Auto Generating button  2.Input the database name | Show alert with invalid input message  Request user input again until it is valid | Pass |
| A20 | Database Configuration  Button (Empty Input) | 1. Click Database Check Connection  2. Input the username(Empty)  3. Input the password(Empty)  4. Input the URL of database(Empty) | Show alert with invalid input message  Request user input again until it is valid | Pass |
| A21 | Database Configuration haven’t finish | User press Database Auto Generation/ Delete Specific Record | Show alert to tell the user please configure the database first | Pass |
| A22 | Scan IP Address((Empty Input) | 1.User click “Scan IP Address” button  2.Input the target network(Empty)  3.Input the community(Empty) | Show alert with invalid input message  Request user input again until it is valid. | Pass |

# Discussion

Expected Upcoming Progression

**Email or Whatsapp Alerting**

In order to let the administrator can receive messages at any time when some problem happened on the server or router, we would like add a email or Whatsapp alert functions to alert the administrator when there are some server or router approaching the limit.

**Improvement of usage trend generating function**

In current stage, our program only provide 7 days average usage trend generating and the program require user to type in the date following the format 2015-04-28, it is inconvenient. Therefore, in the upcoming progression, we would like to improve the function to let the user choose date in graphic mode and the user can choose more than 7 days.

**Provide more types of chart**

Now, when our program finish the capture. The program will show the data in the pie chart and the network traffic is show in text format. We would like to add dynamic line chart for the network traffic, so the administrator can see the trend of network traffic more easily.

**Implement multi capture**

In terms of the current situation, the network monitoring system can not capture the status with multi device at the same. For the future, we will implement the multi-threading capture to the system for capture status with multi device and store it to the database.

**Data analysis with Report Generating**

In order to help the administrator to analysis those data easily. A report generating function is required. The report will show the averger usage of cpu, ram, network traffic with chart monthly and some important logging.

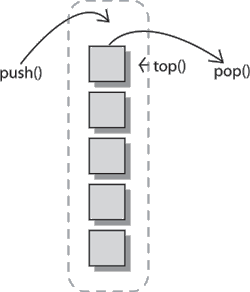
Problems encountered and the corresponding solutions

**Problem1**

When we implement the part of the device auto detection, we face a problem we need to implement something like array to store the IP address from the device we have detected but we don’t know how many IP we will detected. Array need to specify the capacity when we define it in the program. We need to find the other solution to store these data.

**Solution1**

We decided to use stack to store IP address because when we define a stack, we don’t need to specify the capacity of the stack and implement stack is easier than implement the dynamic array.



**Problem 2**

When we implement the part of device auto detection, we face a problem is how to detect the device on the network which is enabled the SNMP service.

**Solution2**

For the flow of this function, we decided to divide to two parts.

First part, we will ask the user input a range of IP with subnet mask. Then we will import the ping -n 1 -w 100 ipaddress to command prompt by using **java.lang.*getRuntime*().exec().** This command is use to check the connection with the target IP. If there is a response form target IP, the IP address will store into the stack and pass it to the auto detection handler.

The Second part, when the handler receive the stack of IP, it will **pop()** out every IP to the SNMP connection checking to find out which IP has activate the SNMP service. If the program detected the target IP ‘s connection had established and SNMP service had activated, the program will pass the IP address to the GUI and show the valid IP in the list.

**Problem 3**

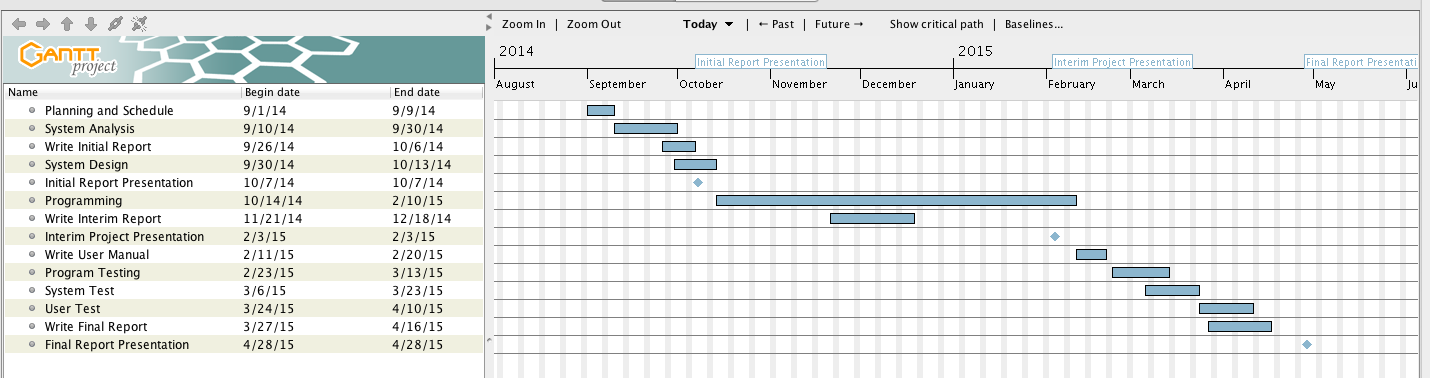
When we want to capture the network traffic of the target device, we found that there are no such OID showing the network loading, for example, showing how many KB per second.

**Solution 3**

For the network traffic monitoring, we found that there is an OID showing the total size of packet that device received and sent. This is number that keeps superimposed. Therefore, we identify the difference value between current sizes and previous size to find out traffic that going through the interface.

However, the system can capture two values in a flash, less than a second, the value (total size of packet) could not immediately change so we can only find out very small difference (only about 3 KB). To solve this, we added a break between two statements, so that we can see a big difference between the current total size and previous total size. Therefore we can find out the inbound and outbound traffic.

# Project schedule



# Task

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Name | Start | End | Milestone |  |  |  |
| Planning and Schedule | 9/1/14 | 9/9/14 |  |  |  |  |
| System Analysis | 9/10/14 | 9/30/14 |  |  |  |  |
| Write Initial Report | 9/26/14 | 10/6/14 |  |  |  |  |
| System Design | 9/30/14 | 10/13/14 |  |  |  |  |
| Initial Report Presentation | 10/7/14 | 10/7/14 | \* |  |  |  |
| Programming | 10/14/14 | 2/10/15 |  |  |  |  |
| Write Interim Report | 11/21/14 | 12/18/14 |  |  |  |  |
| Interim Project Presentation | 2/3/15 | 2/3/15 | \* |  |  |  |
| Write User Manual | 2/11/15 | 2/20/15 |  |  |  |  |
| Program Testing | 2/23/15 | 3/13/15 |  |  |  |  |
| System Test | 3/6/15 | 3/23/15 |  |  |  |  |
| User Test | 3/24/15 | 4/10/15 |  |  |  |  |
| Write Final Report | 3/27/15 | 4/16/15 |  |  |  |  |
| Final Report Presentation | 4/28/15 | 4/28/15 | \* |  |  |  |

Task and Role of Each Student

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Student ID | E-mail | Contribution to Group (%) |
| IU Hiu Pan | 130351532 | [130351532@stu.vtc.edu.hk](mailto:130351532@stu.vtc.edu.hk) | 25% |
| TAM Siu Kei | 130548064 | [130548064@stu.vtc.edu.hk](mailto:130548064@stu.vtc.edu.hk) | 25% |
| LAU Chun Yin | 130226235 | [130226235@stu.vtc.edu.hk](mailto:130226235@stu.vtc.edu.hk) | 25% |
| SIU Man Ho | 130297142 | [130297142@stu.vtc.edu.hk](mailto:130297142@stu.vtc.edu.hk) | 25% |

Task Allocation of Final Report

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | IU Hui Pan | TAM Siu Kei | LAU Chun Yin | SIU Man Ho |
| Mission Researching and Analysis | √ | √ | √ | √ |
| System Discussion and Planning | √ | √ | √ | √ |
| Database Design |  | √ |  |  |
| Database Implementation | √ | √ |  |  |
| Report Generating Function |  | √ |  |  |
| OID Research |  |  | √ |  |
| SNMP Configuration | √ |  |  |  |
| Device Auto Detection | √ | √ |  |  |
| Auto Capture | √ | √ |  |  |
| GUI Design |  |  | √ | √ |
| GUI Implementation | √ |  |  | √ |
| Graphic Design |  |  | √ | √ |
| Website Design |  |  | √ |  |
| Chart Generating | √ |  |  |  |
| Setup testing environment | √ | √ |  | √ |
| Report | √ | √ | √ | √ |

# Conclusion

Nowadays, there are more and more company putting their service on the internet. Therefore, people can use these services anytime and anywhere. For some large scale company, there is a huge number of users using their service at a same time. It may cause a heavy workload to their network devices, if the device crash without any notification for a time, it may cause a huge economic loss. For administrator, it is very hard to keep checking those device physically. Therefore, we would like to develop a Network Monitoring System to help then keep monitor devices through network.

Our software aims to provide a clearly GUI for the administrator to get status from devices easily. Therefore, administrator can manage those status efficiently because the system can collect device status centrally. Our system provided some function that help the administrator to evaluate the status. First, our system represent the status with pie chart and line chart. Second, the system can output records from the database with a excel file, so that the administrator can use the Excel as a tool for analyzing status. Third, system can alert the administrator when device approaching the limitation.

Obviously, our system is not perfect, there are many parts that we can improve. First, we should add more type of chart. For example, the dynamic chart to help the administrator understanding the status. Second, we can add the email or Whatsapp alerting function to our system. Therefore, administrator can receive the alert any time when there are some problem occurred on the device. Third, improve the usage trend generating function to make it more flexible.

These are the function that we completed, and further improvement of our Network Monitoring System.

# Reference

-SNMP4J Official Website

Retrieved from <http://www.snmp4j.org/>

-JFreeChart Official Website

Retrieved from <http://www.jfree.org/jfreechart/>

-Doc4j Official Website

Retrieved from <http://www.docx4java.org/trac/docx4>j

-oracle website about JDBC

Retrieved from <http://www.oracle.com/technetwork/java/javase/jdbc/index.html>

-Apache POI official website

Retrieved from <https://poi.apache.org/>

- [系統開發生命週期]Use Case Diagram概述

Retrieved from [http://www.dotblogs.com.tw..../use-case-diagram.aspx](http://www.dotblogs.com.tw/jimmyyu/archive/2010/01/18/use-case-diagram.aspx)

-Functional requirement on Wikipedia

Retrieved from [http://en.wikipedia.org/wiki/Functional\_requirement](http://en.wikipedia.org/wiki/functional_requirement)

- {UML2.0} Sequence Diagram 簡單範例與說明

Retrieved from [http://www.kenming.idv.tw...\_e\_eo](http://www.kenming.idv.tw/uml2_0_sequence_diagram_cdia_rcm_af_e_eo)

O'Reilly & Associates. (2001). JTree and TreeModel (Java Foundation Classes).

Retrieved from <http://docstore.mik.ua/orelly/java-ent/jfc/ch03_19.htm>

Oracle / affiliates. (1995). How to Write a Mouse Listener

Retrieved from <http://docs.oracle.com/....events/mouselistener.html>

Tutorialspoint. (2014). Java Simple GUI - Programming Examples

Retrieved from <http://www.tutorialspoint.com/javaexamples/gui_piechart.htm>

Tutorialspoint. (2012). SWING Event Handling.

Retrieved from <http://www.tutorialspoint.com/swing/swing_event_handling.htm>

Ouali Idrissi Med Amine. (2013, January 6). JAVA Swing Tutorial 5

Retrieved from [https://www.youtube.com/watch?v=92H0AN6egFI](https://www.youtube.com/watch?v=92h0an6egfi)

Java - Multithreading Programming Tutorial - TutorialsPoint

Retrieved from <http://www.tutorialspoint.com/java/java_multithreading.htm>

JDBC Tutorial - TutorialsPoint

Retrieved from <http://www.tutorialspoint.com/jdbc/>

Java - The Stack Class - TutorialsPoint

Retruieved from <http://www.tutorialspoint.com/java/java_stack_class.htm>

[Java.io package tutorial - TutorialsPoint](http://www.tutorialspoint.com/java/io/)

Retrieved from <http://www.tutorialspoint.com/java/io/index.htm>