Bahria University

Karachi Campus

**COURSE: Data Communication and Networking**

COURSE CODE: **CEN-222**

**PROJECT REPORT**

**SUBMITTED BY:**

Name of group members and registration number:

**MOEEZ AHMED SHAH\_ (02-131182-045)**

**SUBMITTED TO:**

Engr. Mahawish Fatima/ Engr. Fareeha Dilawar

**SIGNED REMARKS: SCORE:**

Table of Contents

[**BACKGROUND:** 3](#_Toc61470015)

[**INTRODUCTION:** 4](#_Toc61470016)

[**PROJECT PARADIGM** 5](#_Toc61470017)

[**MECHANISM AND WORKING:** 14](#_Toc61470018)

[**FUNCTIONALITIES:** 39](#_Toc61470019)

[**FUTURE WORK:** 42](#_Toc61470020)

[**CONCLUSION:** 43](#_Toc61470021)

[**REFERENCES:** 44](#_Toc61470022)

# **BACKGROUND:**

**EARLY HISTORY OF RIP PROTOCOL:**

RIP (Routing Information Protocol) was first described in RFC1058 and revised to RIPv2 in RFC1388 and then in RFC1723 and RFC2453. RIP-enabled devices share their routing tables sending unacknowledged messages over UDP port 520. The processing of RIP requests (command code 0x1) specified in the RFC states that requests made to any RIP interface (event “silent” ones) must be responded, without any restriction. This leads to a situation in which more devices or users than expected would be gaining certain information about the network. This situation also makes to device serving routes use CPU resources for purposes other than originally intended. This might potentially result in denial-of-service attacks (although this are rarely considered in routers and will not be considered anymore in the rest of the paper). If we consider that the RFC does not limit the TTL values of the messages involved, potentially any host in the whole network is a potential attacker. These problems can be avoided with the use of filters that deny any traffic directed to infrastructure IP addresses (but perhaps a few controlled packets, such as ICMP requests and responses or some high UDP ports used by applications such as traceroute).

**ORIGIN OF SMART HOME:**

With the 1975 release of X10, a communication protocol for home automation, the smart home, once a pipe dream a la The Jetsons, came to life. X10 sends 120 kHz radio frequency (RF) bursts of digital information onto a home's existing electric wiring to programmable outlets or switches. These signals convey commands to corresponding devices, controlling how and when the devices operate. A transmitter could, for example, send a signal along the house's electric wiring, telling a device to turn on at a specific time. However, as electrical wiring isn't designed to be free from radio-band "noise," X10 was not always fully reliable. Signals would be lost and, in some cases, signals wouldn't cross circuits that were wired on different polarities, created when 220-volt service is split into a pair of 100-volt feeds, as is common in the U.S. Additionally, X10 was initially a one-way technology, so while smart devices can take commands, they cannot send data back to a central network. Later, however, two-way X10 devices became available, albeit at a higher cost.

When home automation company Insteon came on the scene in 2005, it introduced technology that combined electric wiring with wireless signals. Other protocols, including ZigBee and Z-Wave, have since emerged to counter the problems prone to X10, though X10 remains a widely installed communications protocol to this day. Nest Labs was founded in 2010 and released its first smart product, the Nest Learning Thermostat, in 2011. The company also created smart smoke/carbon monoxide detectors and security cameras. After being acquired by Google in 2015, it became a subsidiary of Alphabet Inc. in the same year.

(Routing Information was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

RIP (Routing Information Protocol) was first

described in RFC1058 [19] and revised to RIPv2 in

RFC1388 [20] and then in RFC1723 [21] and

RFC2453. RIP-enabled devices share their routing

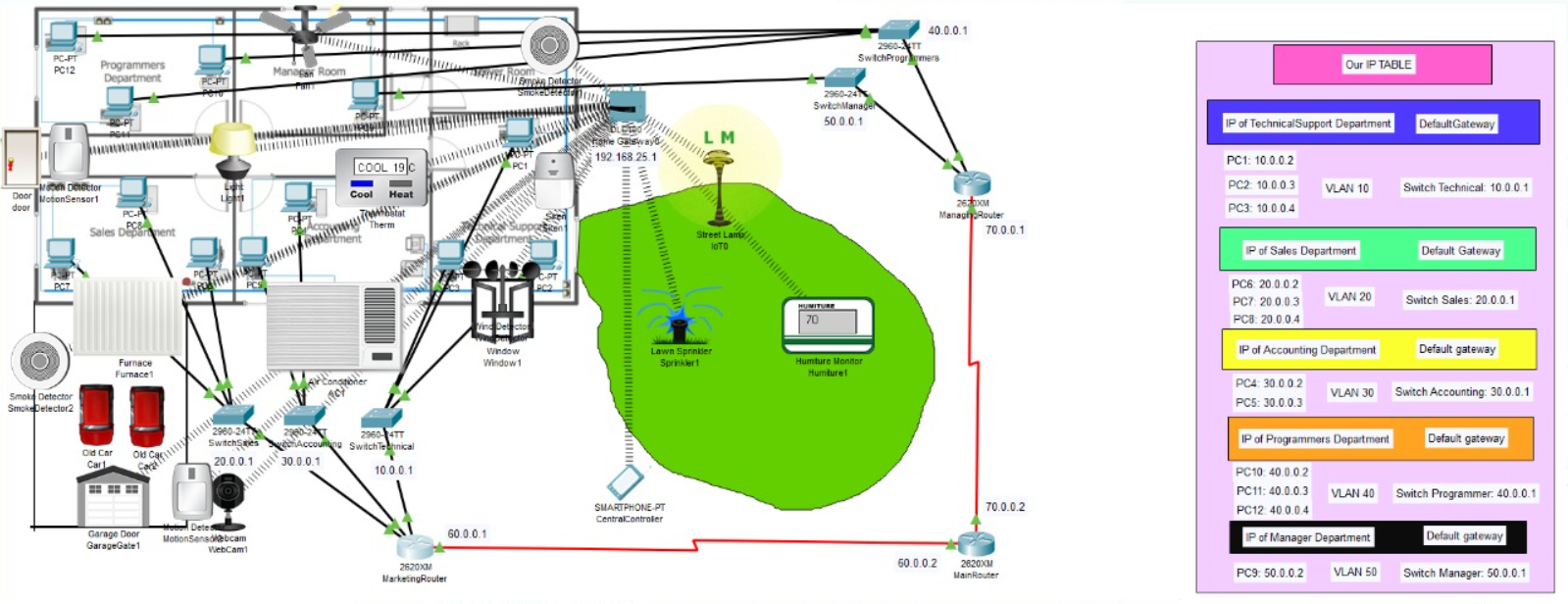
tables sending unacknowledged messages over UDP

port 520. RIP (RIPv1 from now on), has a packet

# **INTRODUCTION:**

Our project is a smart software house with many departments like, a manager, technical support department, accounting department, program department and we connected them through pc’s, switches and routers. In the routers we used the concept of protocol of RIP, and also we created vlans and enabled trunking in the switches. For making it smart we use connect different things (like wind detector, furnace, a lamp, motion detector, smoke detector for opening of garage, webcam, lawn sprinkler, thermos state. And a lot of other things as well). In this smart software house project with many departments like, a manager, technical support department, accounting department, program department and we connected them through pc’s, switches and routers. In the routers we used the concept of protocol of RIP, and also we created vlans and enabled trunking in the switches. In our project the network will allow the entire software house to communicate with each other whenever it is required. File sharing can also be done from one department to the other department of the network. Communication speed the whole process maintained efficiently. Our network has been completely secured and reliable. As we make sure that the communication and other important data of the company is not leaked or tampered by any outsider In this network the entire organization is connected to a single hence different entities of the network can share hardware and software which includes servers, clients, transmission media, shared data, shared printers and other hardware and software resources, network interface card (NIC), local operating system (LOS), and the network operating system (NOS). In the departments and all the things in our software house will be operated by the smartphone.

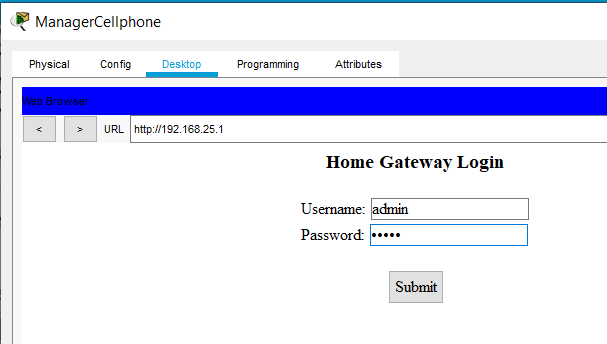
# **PROJECT PARADIGM**



**DETAILS:**

**Manager Cellphone:** The cellphone of the manager would have everything that is smart. We used a home gateway from which each of the smart device and detectors are connected. The smartphone of manager is connected to the home gateway and provides a very effective UI in order for everything to work smoothly. Ip for home gateway = **192.168.25.1**

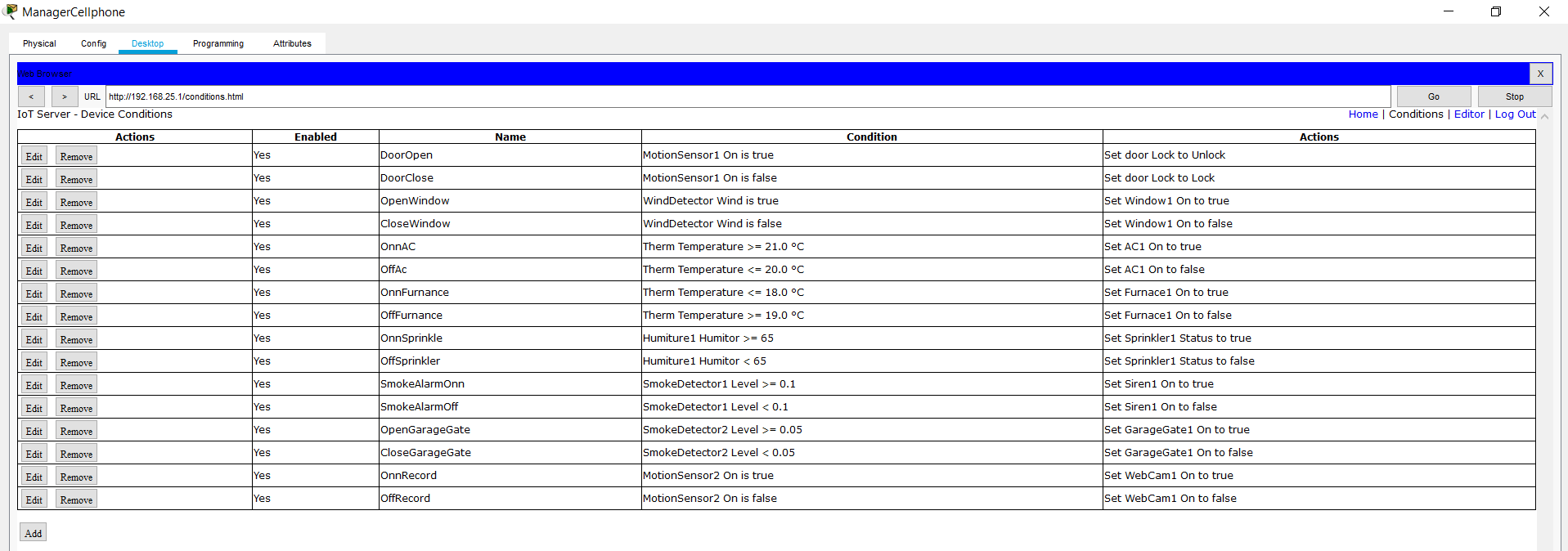
UI for cellphone:



Username and password = admin

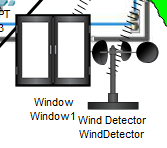


**Conditions:**



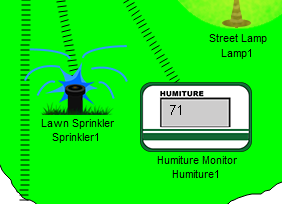
Conditions are self-explanatory.

**Wind detector:** we connected our software house window with this detector, so whenever there will be a fresh air outside it will be opened automatically.



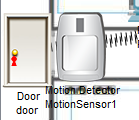
**Humiture:**

When humiture less than 70, the sprinkler will stop watering garden.

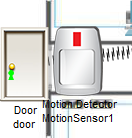


**Motion detector:** we connected our software house door with this detector so when someone is near to the software house door, it will open automatically.

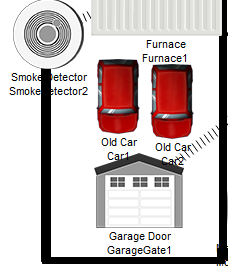
Locked when not detected:



Unlocked when detected:

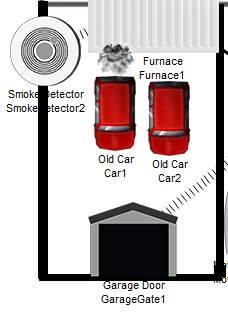


**Smoke detector:** we use smoke detector for the opening of garage as it will detect the smoke of cars and open automatically. Also, the smoke detector is inside the server room of our software house, as soon as it`ll detect a significance amount of smoke, the alarm would sound guiding everyone about the situation of fire.

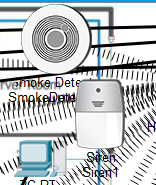


**When smoke is detected (Car is turned on):**

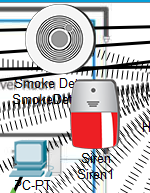
Garage opens:



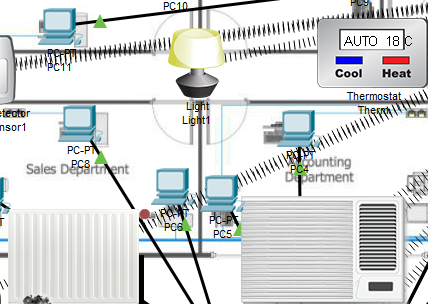
**Siren (initially):**



**Siren (After certain amount of smoke is detected):**



**Thermostat**: we connected thermostat to both AC and Furnace so whenever the temperature is below then 19 degree the furnace will be opened automatically and provide the heat and make us warm and also when temperature up to 19 degree it will automatically be off. This same will apply to the AC but in vice versa. The AC would open if the temperature exceeds 20 and would stop when it`s below 20. In simple words, a temperature of 20 would be maintained all the time.



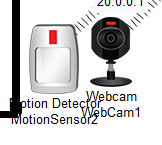
Furnace is working as temperature is less than 19.

**Human Monitor Detector**: we also connected webcam to human monitor detector in our house so when someone is near it will detect the person and webcam will atomically capture him/her.

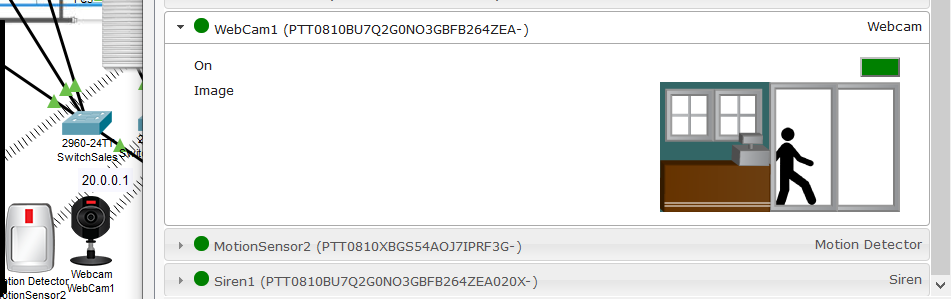
Initially:



If a person is detected:



In manager`s cellphone, image would be displayed:



We also used concepts **vlans** and **vtp** in our switches, we created **5 vlans**, and they are as follow:

**VLAN 10:** we connected our technical support department to vlan 10.

**VLAN 20:** we connected our sales department to vlan 20.

**VLAN 30:** we connected our accounting department to vlan 30.

**VLAN 40:** we connected our program department to vlan 40.

**VLAN 50:** we connected our manager department to vlan 50.

And in there we used the concepts of **vtp:**

**Vtp mode server:** We used vtp in every department, we make our **manager** department is in server mode as he is the head of all departments. Manager is like an administrator for our software house.

**Vtp mode client:** we make rest of all departments in our client mode

**Why we use vlan?**

A VLAN allows different computers and devices to be connected virtually to each other as if they were in a LAN sharing a single broadcast domain. Another reason to use that, it can help reduce IT cost, improve network security and performance, provide easier management, as well as ensuring network flexibility

**Why we used RIP protocol?**

The reason to used rip in this project because our project is based on very small network and our hop count is also less than 16, another reason to use rip is that, rip prevents routing loops, And it also in-corporate with the distance-vector router, which calculate the best path on direction and distance between routers.

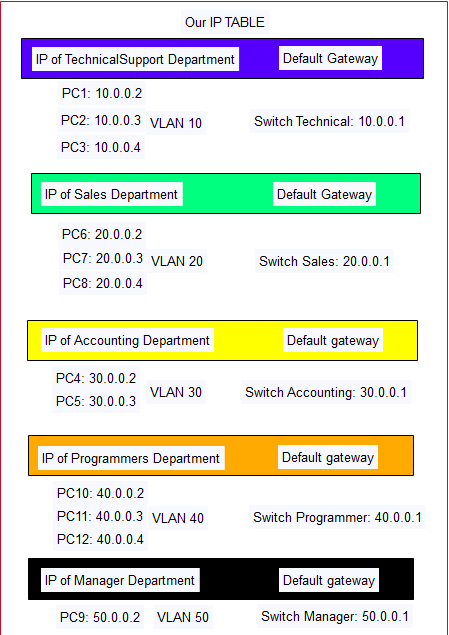
**Why we used Trunking in switches?**

As there are many departments in our projects, so we create vlans for each departments. So after vlan we can’t communicate (ping) to another departments, so for resolving this we used trunking in each vlans and enable the trunking to make the communication possible in our software house. It also enables switched network solutions to scale to large sizes by reducing the network's manual configuration needs.

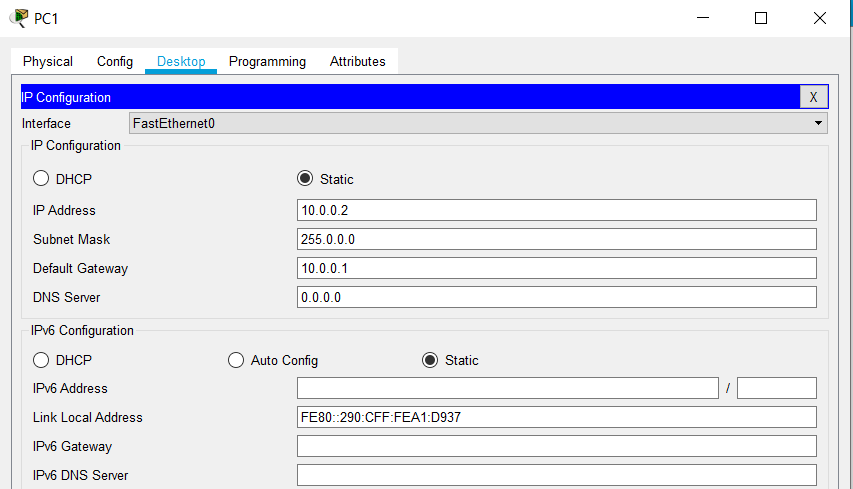
**Why we used VTP in switches?**

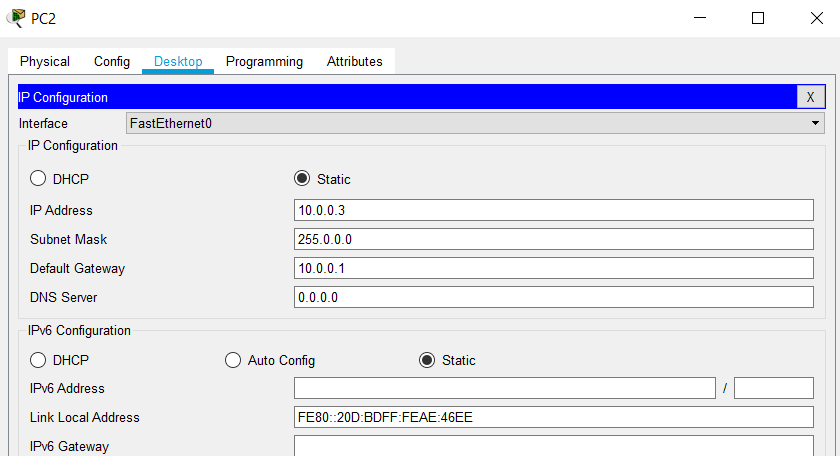
Also, we put our manager department in server mode of VTP where as other departments in client mode. This way our manager is the administrator where as other departments act as client and cannot really add more VLANs.

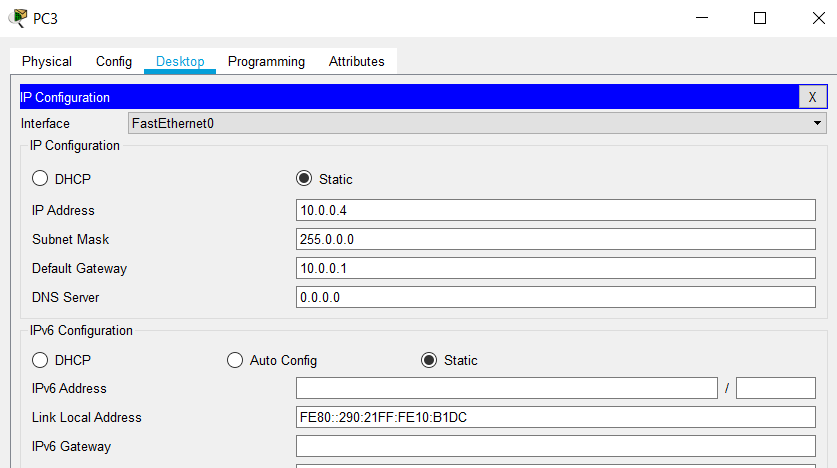
# **MECHANISM AND WORKING:**

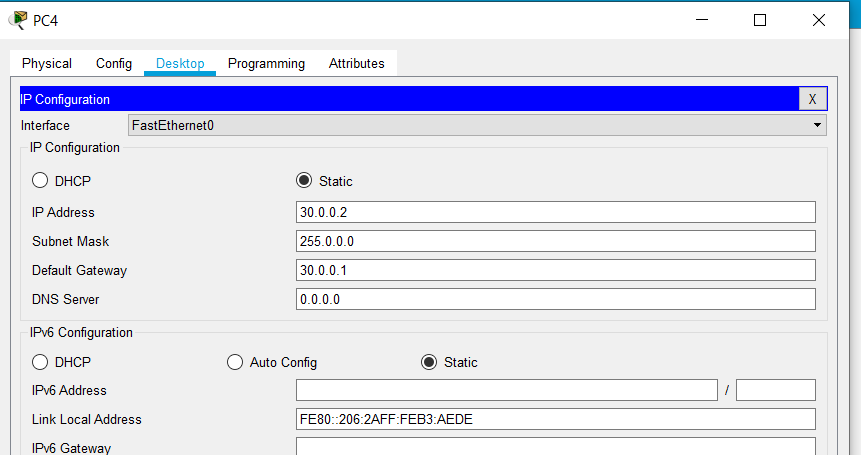


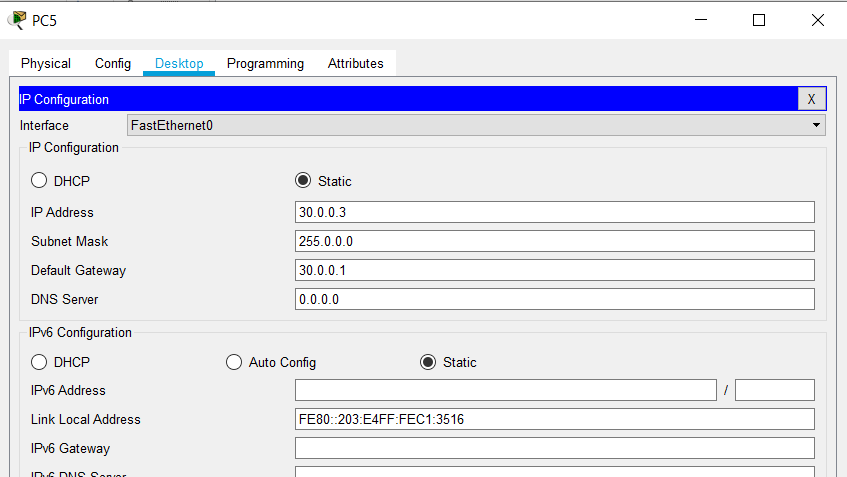
**PC’S COINFIGURATION:**

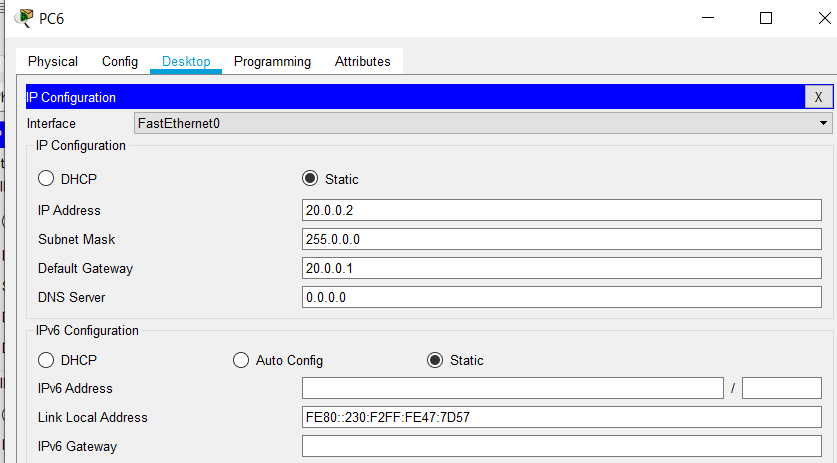


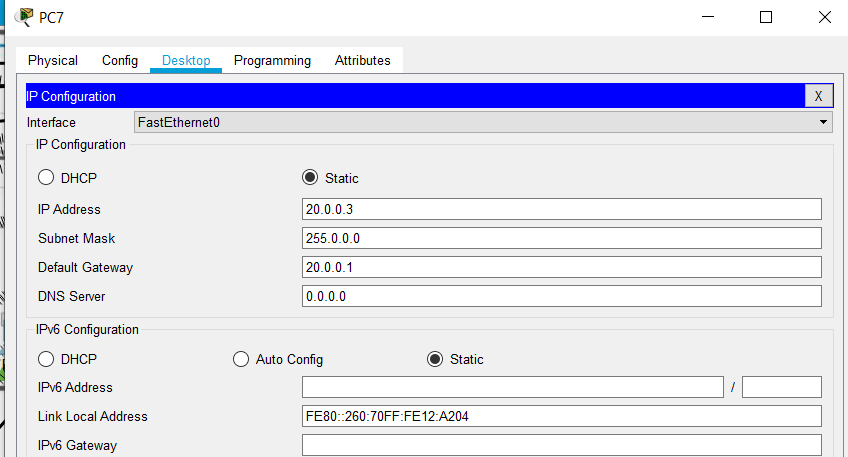


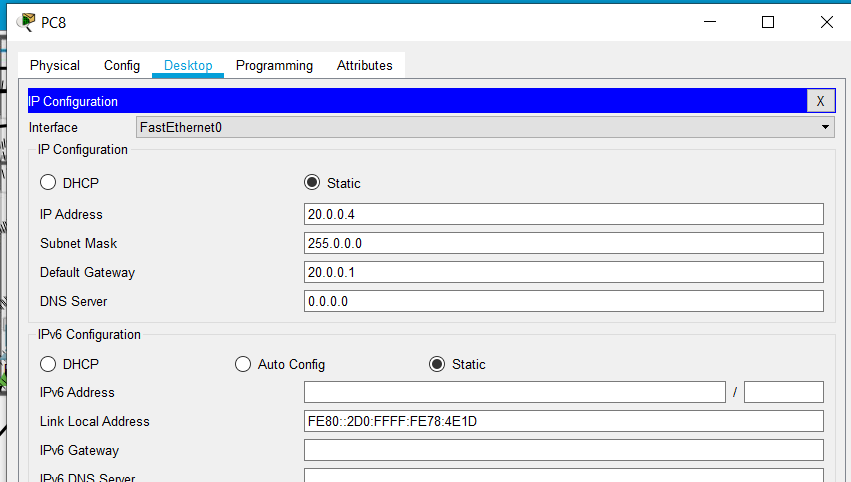


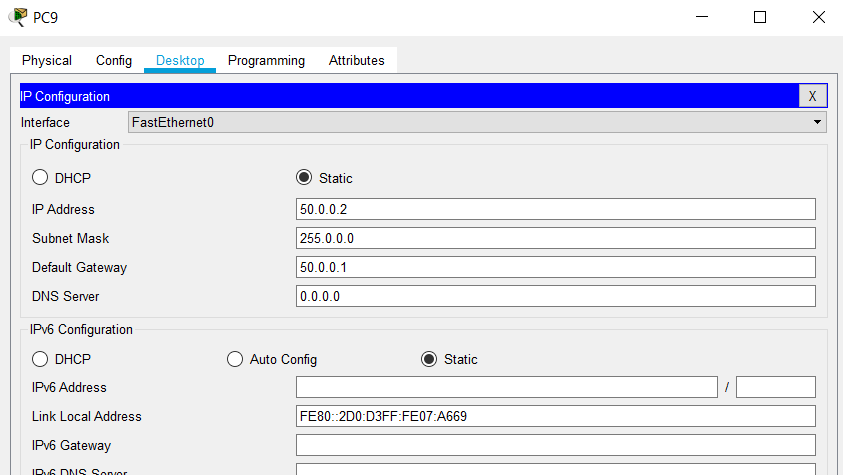


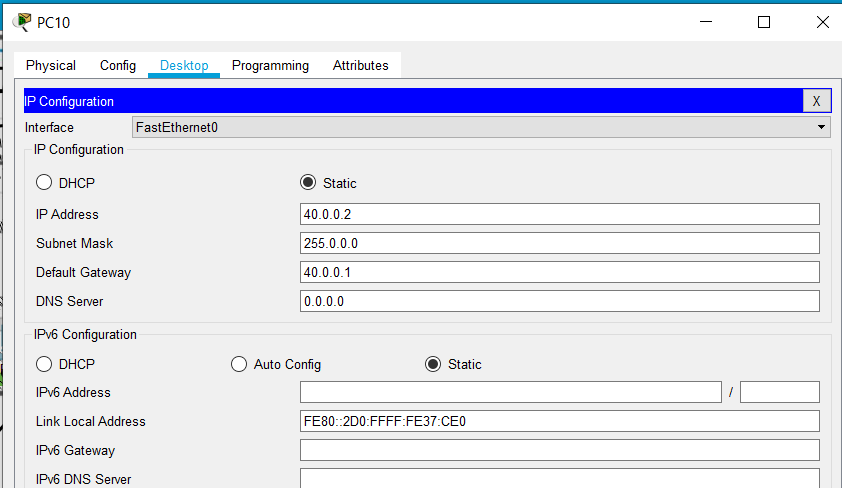


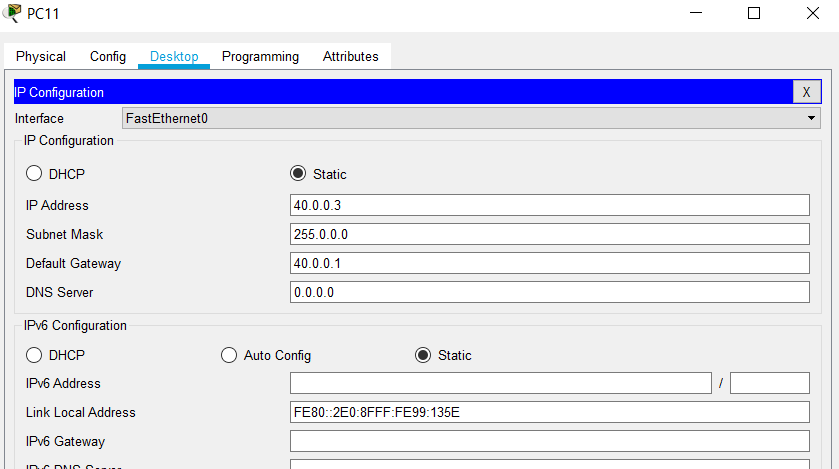


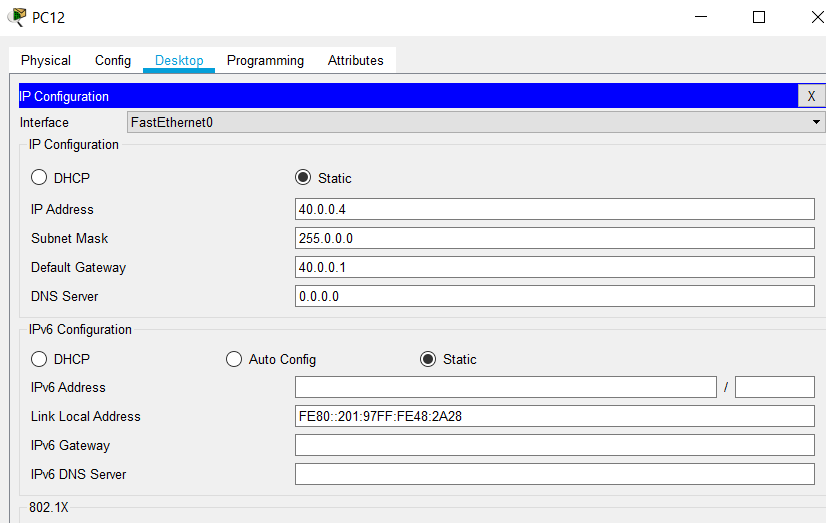








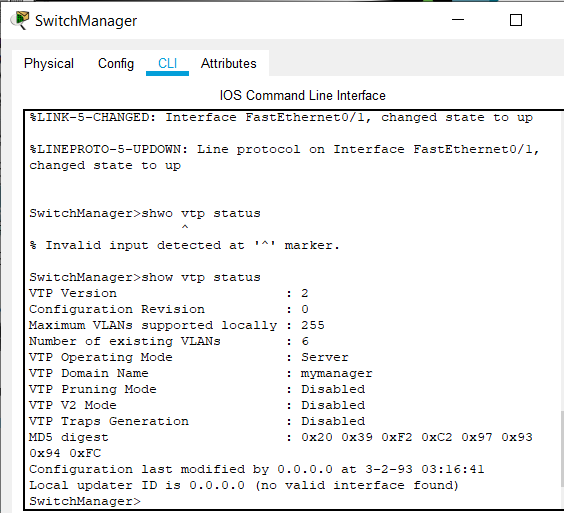


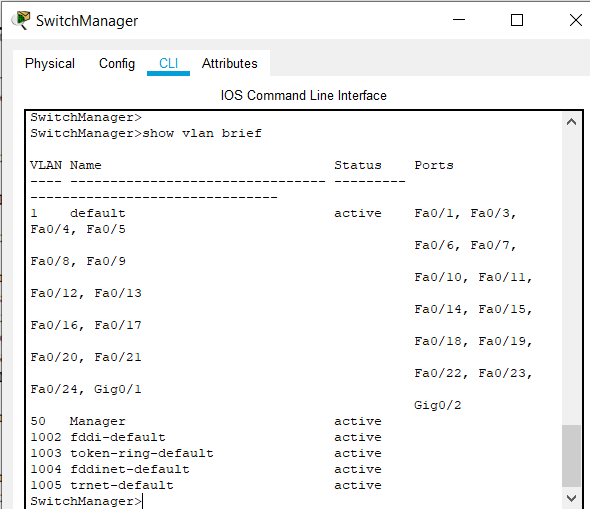


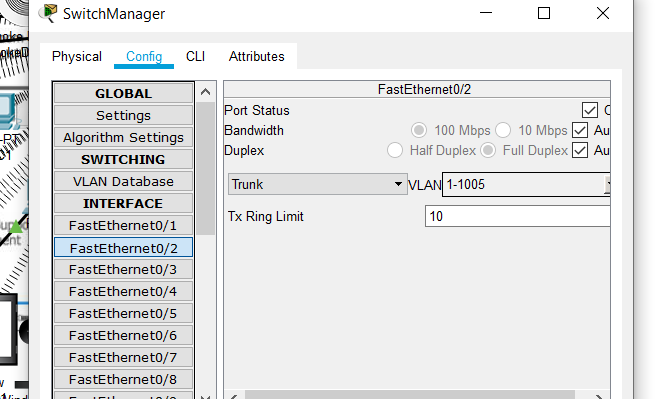
**SWITCHES CONFIGURATION:**

**MANAGER SWITCH:**

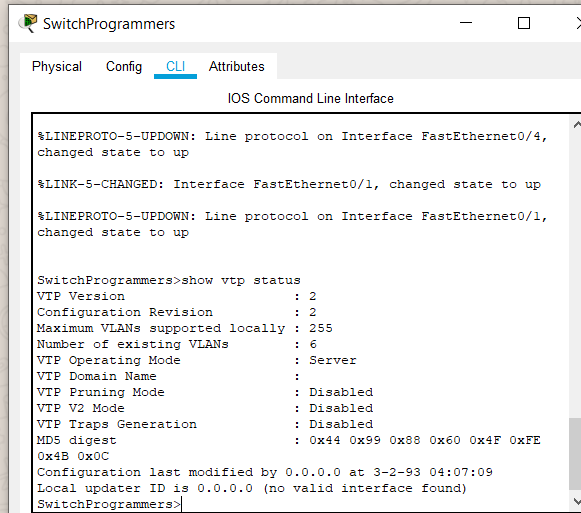
Switch manager is basically in Server mode of VTP as it`s acting as the administrator in the network, the software house.

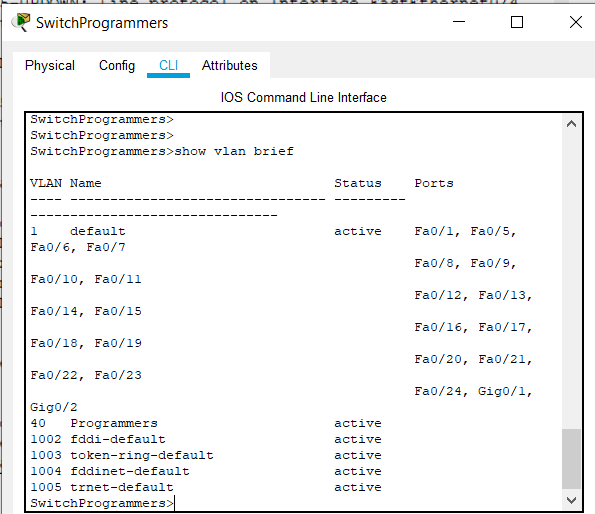


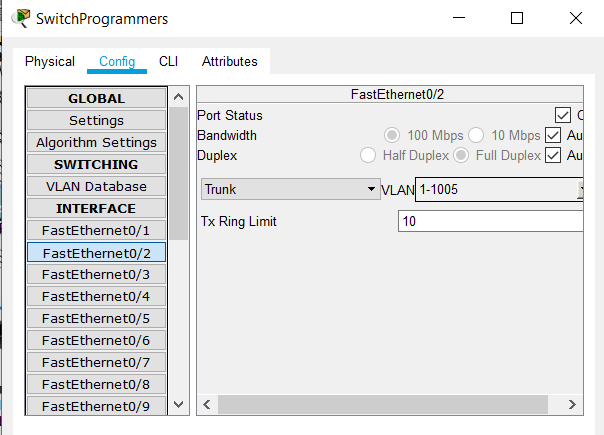
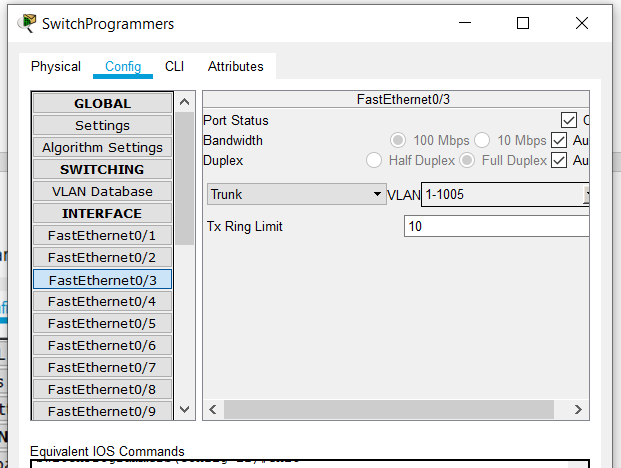


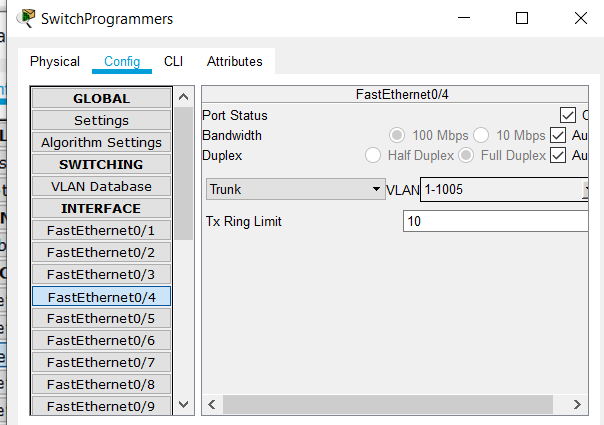


**SWITCH PROGRAMMER:**

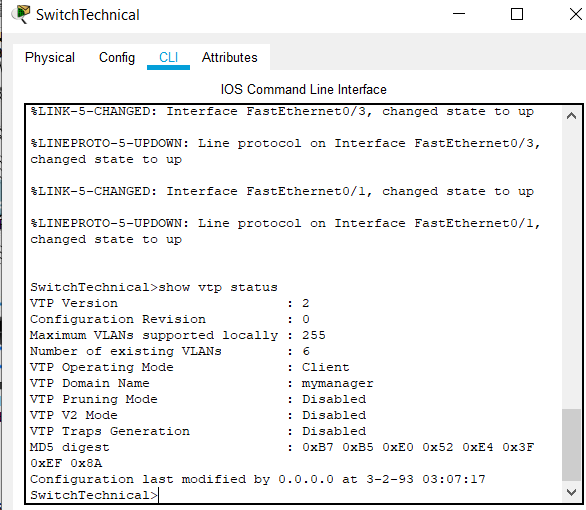


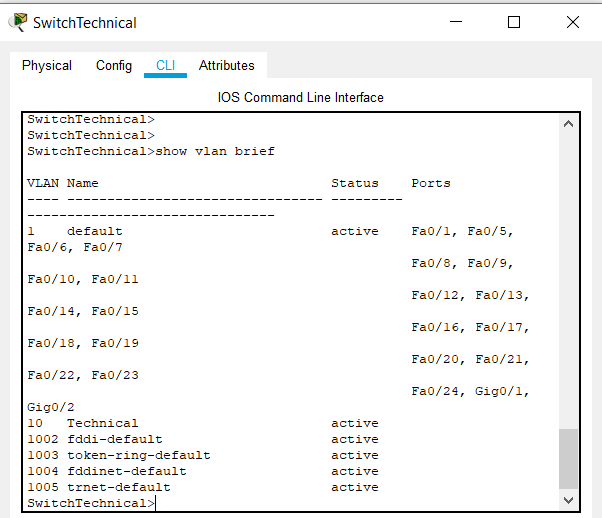


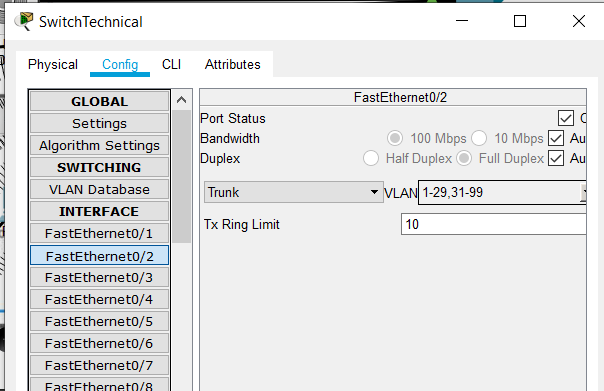
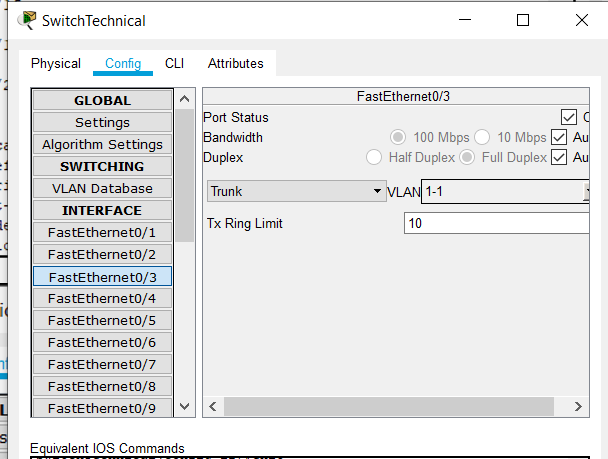
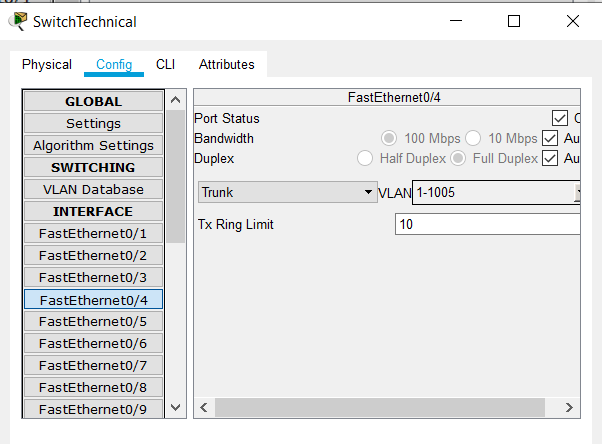
 



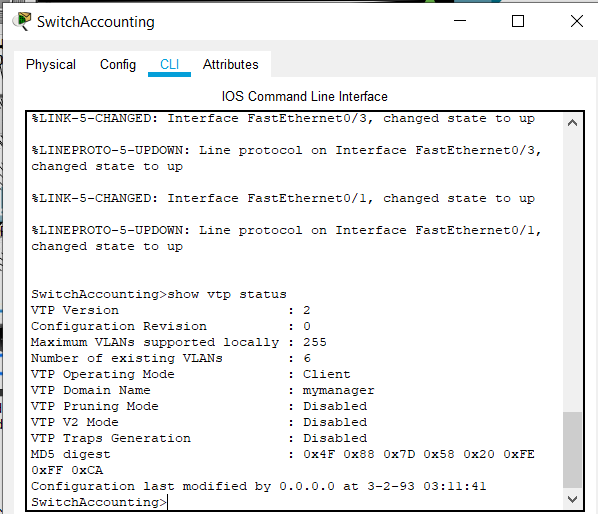
**SWITCH TECHNICAL:**

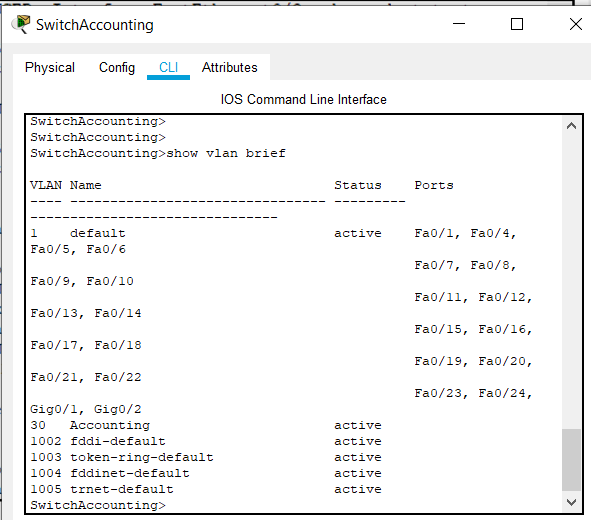


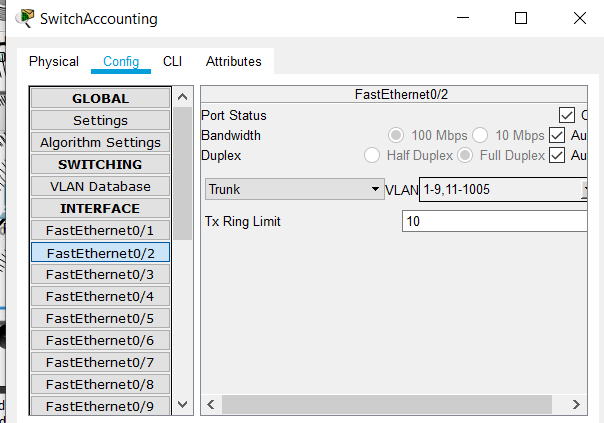
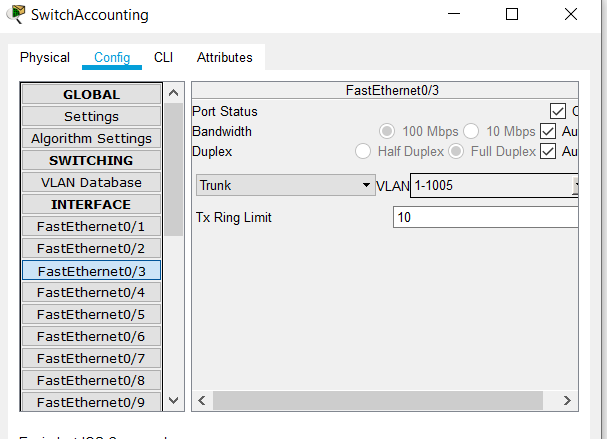


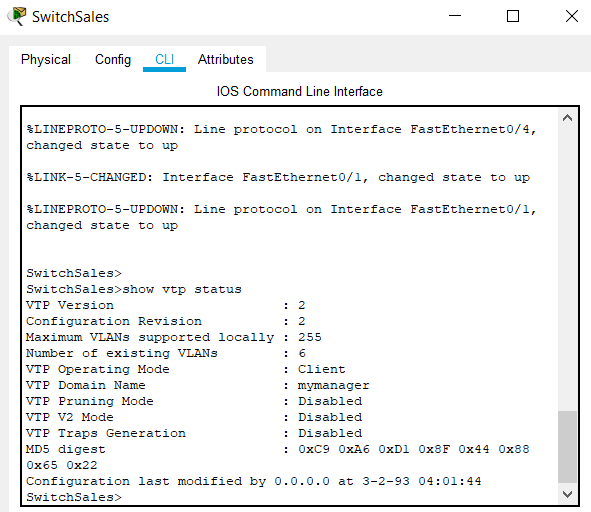
**SWITCH ACCOUNTING:**

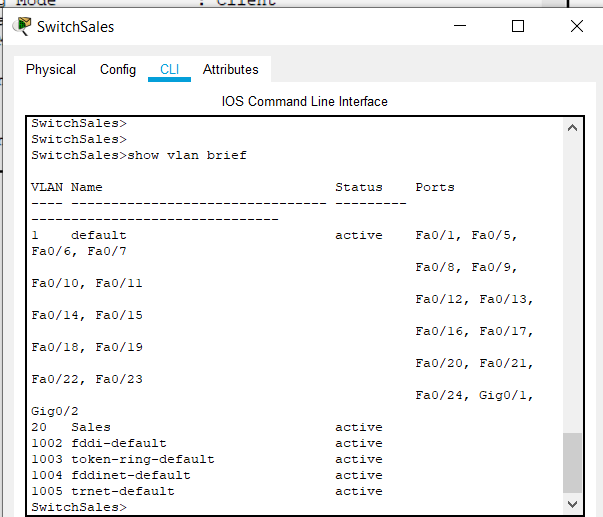


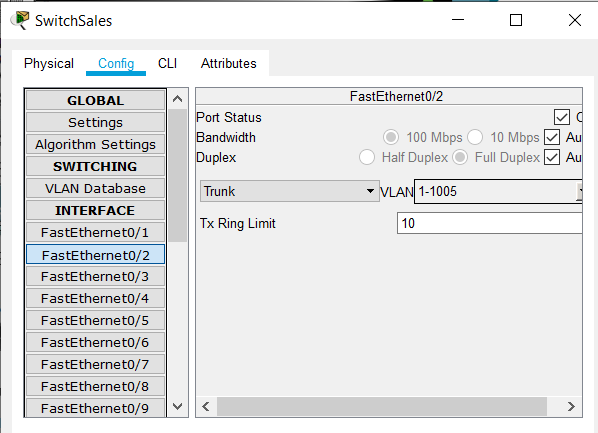
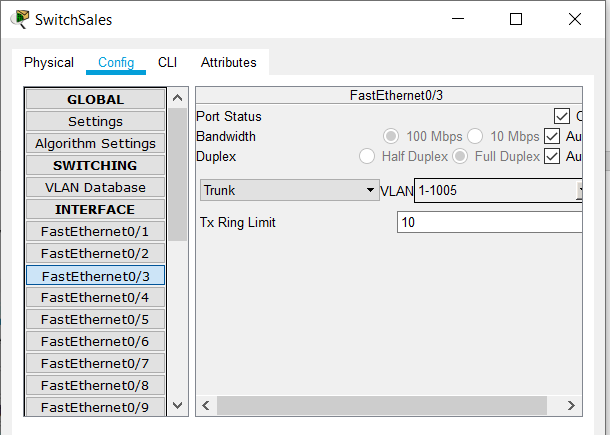
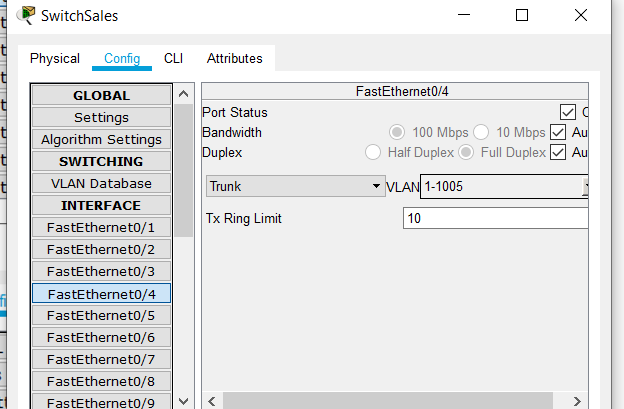


**SWITCH SALES:**



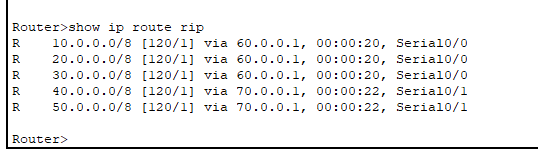


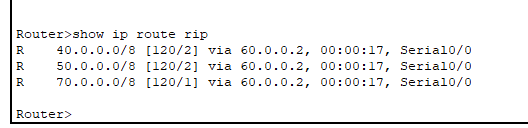
**THREE ROUTERS CONFIGURATION:**

**Main Router:**

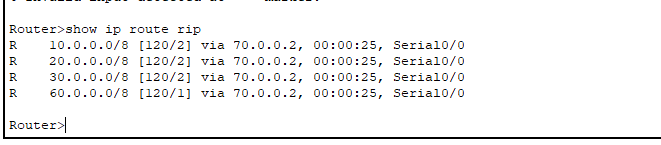
We used RIP protocol because our software house has limited hop counts of 3 and it`s not such a huge network. RIP is smart when it comes to routing and updates it`s routing table automatically.



**Marketing Router:**

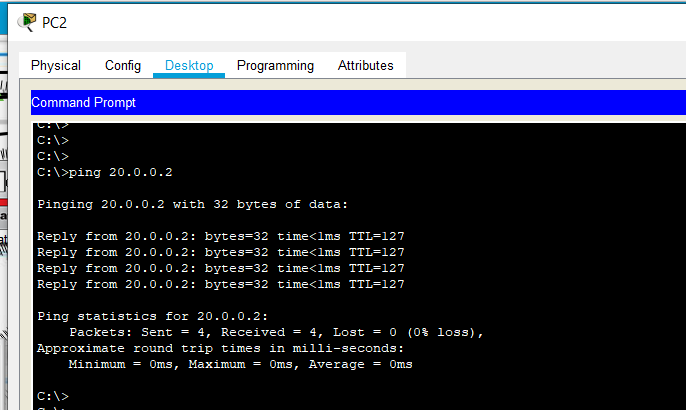


**Managing Router:**

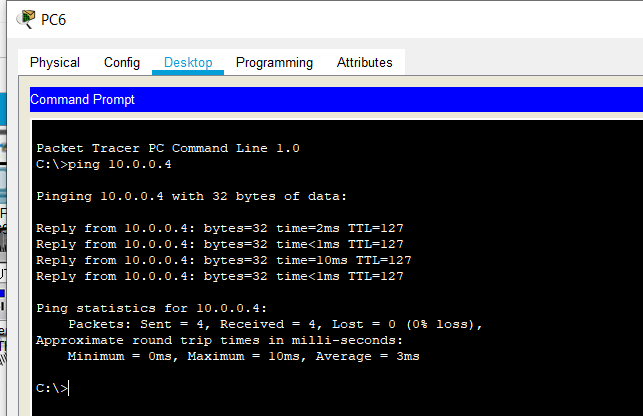


**PINGING BETWEEN DIFFERRENT DEPARTMENTS:**

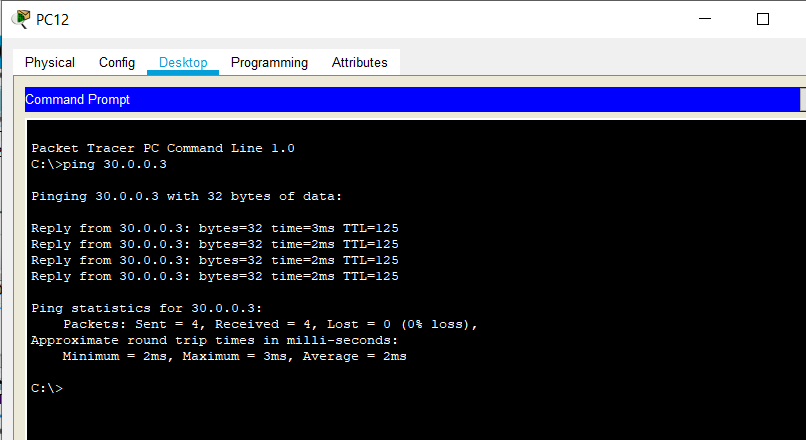
**Ping sales department from technical department:**



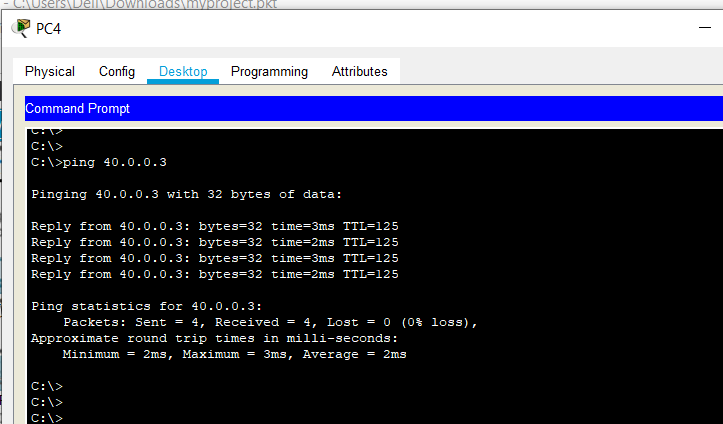
**Ping technical department from sales department:**



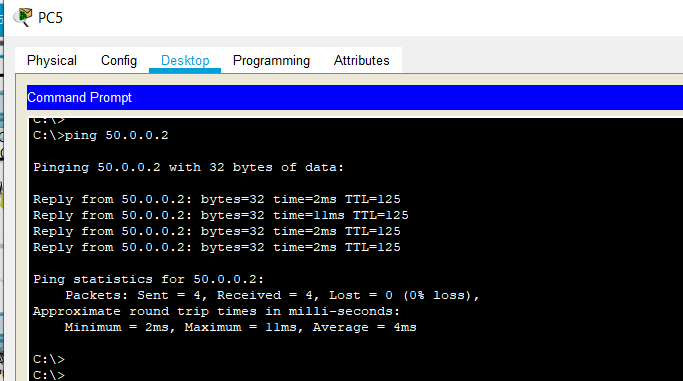
**Ping accounting department from programmer department:**



**Ping programming department from accounting department:**

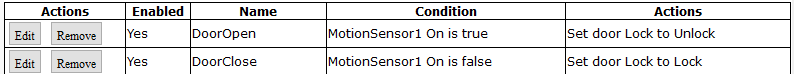


**Ping manager department from accounting department:**



**Conditions for IoT:**

**Door:**



**Window:**



**Central Air Conditioner:**



**Furnace:**



**Sprinkle:**



**Smoke Alarm:**



**Garage Gate:**



**Camera (Recording):**



# **FUNCTIONALITIES:**

**Rip protocol:**

RIP is a protocol used by routers to exchange routing information on a network. Its primary functions are to 1) determine the most efficient way to route data on a network and 2) prevent routing loops.

RIP maintains a routing table, which lists all routers reachable within a network. Each router uses this table to determine the most efficient way to route data. RIP incorporates distance-vector routing, which calculates the best path based on the direction and distance between routers. Each packet is forwarded to the appropriate routers until the packet reaches its destination.

**Vlan:**

A Virtual Local Area Network (VLAN) is a logical division of computer systems in a LAN (Local Area Network) that are connected to a switch, based on their functionalities. VLANs are implemented for enhancing security between various departments and also for easy configuration.

**Vtp:**

VTP is a protocol used to distribute and synchronize identifying information about VLANs configured throughout a switched network. Configurations made to a single VTP server are propagated across trunk links to all connected switches in the network. VTP enables switched network solutions to scale to large sizes by reducing the network's manual configuration needs.

**Switches:**

We used switch 2960 because it provide desktop Fast Ethernet and Gigabit Ethernet connectivity, enabling enhanced LAN services for entry-level enterprise, mid-market, and branch office networks.

**Routers:**

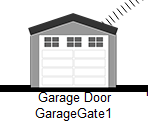
We use 2620XM router, because it offers flexibility for LAN and WAN configurations, voice and data solutions, and much more. This Cisco fast Ethernet router features a single network module slot, AIMs slot, two WICs slots, and dual Ethernet ports.

**Door:**

Main door for entering our software house, can be used using mobile device or the motion detector.

**Garage:**

Opens as soon as a car is turned on and enough smoke is produced. However, can also be opened using mobile device.



**Siren:**

Alarms as soon as enough smoke is detected and there is a caution of fire.



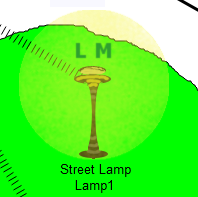
**Fan:**

Fan can be controlled using cellphone device.



**Street lamp:**

Can be controlled using mobile device, goes on and off depending on day and night itself (is smart).



**Sprinkler:**

Water the plants and grass when detects humiture below a certain amount.



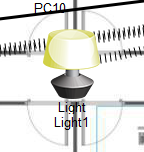
**Camera:**

For security purposes. Manager can open it manually or it can detect motion smartly and itself start recording.



**Lamp (Light):**

Can be controlled using cellphone of managers, is also smart enough to open itself when detects people inside software house.





**Home Gateway:**

Used for IoT purposes.

# **FUTURE WORK:**

**In future we will apply these steps to upgrade the smart houses.**

**Assistive personal devices:**

These devices can monitor your vital functions, including heart rate and body temperature. Others can track you during exercise to monitor effort and calorie count. For people with health problems, smart devices, possibly connected to implants, can monitor oxygen or glucose in the blood so treatment can be fine-tuned.

**Smart beds and sleep monitors:**

Smart beds aim to help you get a healthy night’s sleep. Such beds could offer customized mattress firmness that could sense and react to pressure points and body temperature. They could track your breathing, heart rate and restfulness. They could even help you wake up and smell the coffee, starting your brew and opening your blinds.

**Smart toilets:**

By collecting waste samples on a regular basis, smart toilets could detect infections or serious diseases like cancer early. They could help people with chronic diseases like diabetes better manage their conditions. Connected to the internet, this new smart home technology will take telemedicine to the next level.

**Emotional health:**

A class of technologies in the field of emotion AI have promise for people suffering from emotional stresses. Apps using artificial intelligence could sense tone of voice, facial expressions, speech patterns and biometrics to help people with their own self-awareness. For those with autism, these apps could help people better interact with others.

**Saving Energy:**

We will make our home much smarter, son it will harness computing power and data to optimize energy use in a comprehensive way. From closing blinds when the sun hits certain windows to zoned cooling and heating, smart systems can minimize energy use while increasing your comfort. They save you money without sacrifice.

# **CONCLUSION:**

One of the recent researches and experiment on the networking is discussed below. A study carried out by Shiv-Yadav and Ashraf (2016), concentrated on the Enterprise Network Design and Implementation for Airports. In their study they looked at three major areas: quality, safety, and security. Utilities for good network security was presented in their research. The utilities configured to enhance security for the entire network were: proxy server, domain server, hardware firewalls, Mac address port security, and IP access control list. This was to prevent internal and external attacks from gaining access to sensitive departments like the service providers and flight management departments. Cabling system, failover firewalls utility Dynamic Host Configuration Protocol, Pre-boot Execution Environment, and Domain Name System servers were the technological services used in the air port’s network. The performance of the network was improved with these technical tools. Two different internet service vendors’ services was deployed for the Air Traffic Control System. The role of the internet service providers was to enable the confirmation of the Air Traffic Control Complex backup operation outside the local network. The "iSCSI initiators" and "iSCSI target" of Windows server backup was used to accomplish this task.

Now our network is better from the previous existing software because it has all the required functionalities and features of a network. In addition to this, we are working to implement an AI system on our network. As we know that we have different software available in the market which has these AI functionalities which includes Amazon Alexa, Google Home etc. But till date we don’t have any network which has AI functionalities. Our network is one of them.

# **REFERENCES:**

Useful links involved in making the projects. List all of them here…

* <https://blog.constellation.com/2019/09/06/the-future-of-smart-homes-and-connected-technology/#:~:text=Connected%20technology%20is%20driving%20the,of%20the%20future%20are%20many.&text=Some%204%2C000%20new%20smart%20home,door%20locks%20and%20kitchen%20appliances>.