Class: Third Year (Computer Science and Engineering)

Year: 2020-21   **Semester:** 1

Course: Programming Lab-3 (4CS354)

Practical No. 4
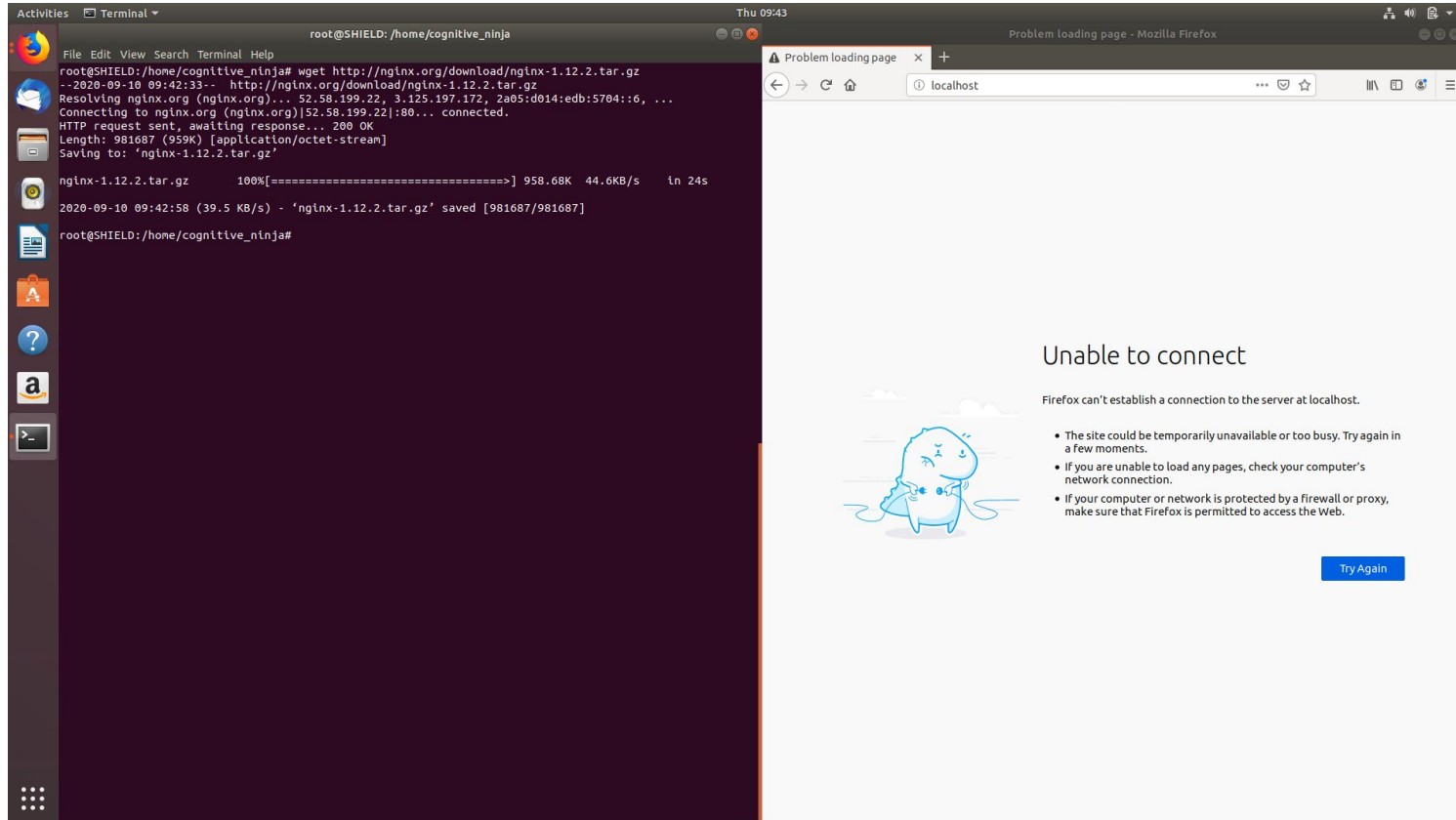
Team Members Exam Seat No:

1. 2018BTECS00050 – Rushikesh Shelke
2. 2018BTECS00063 – Aryan Mali
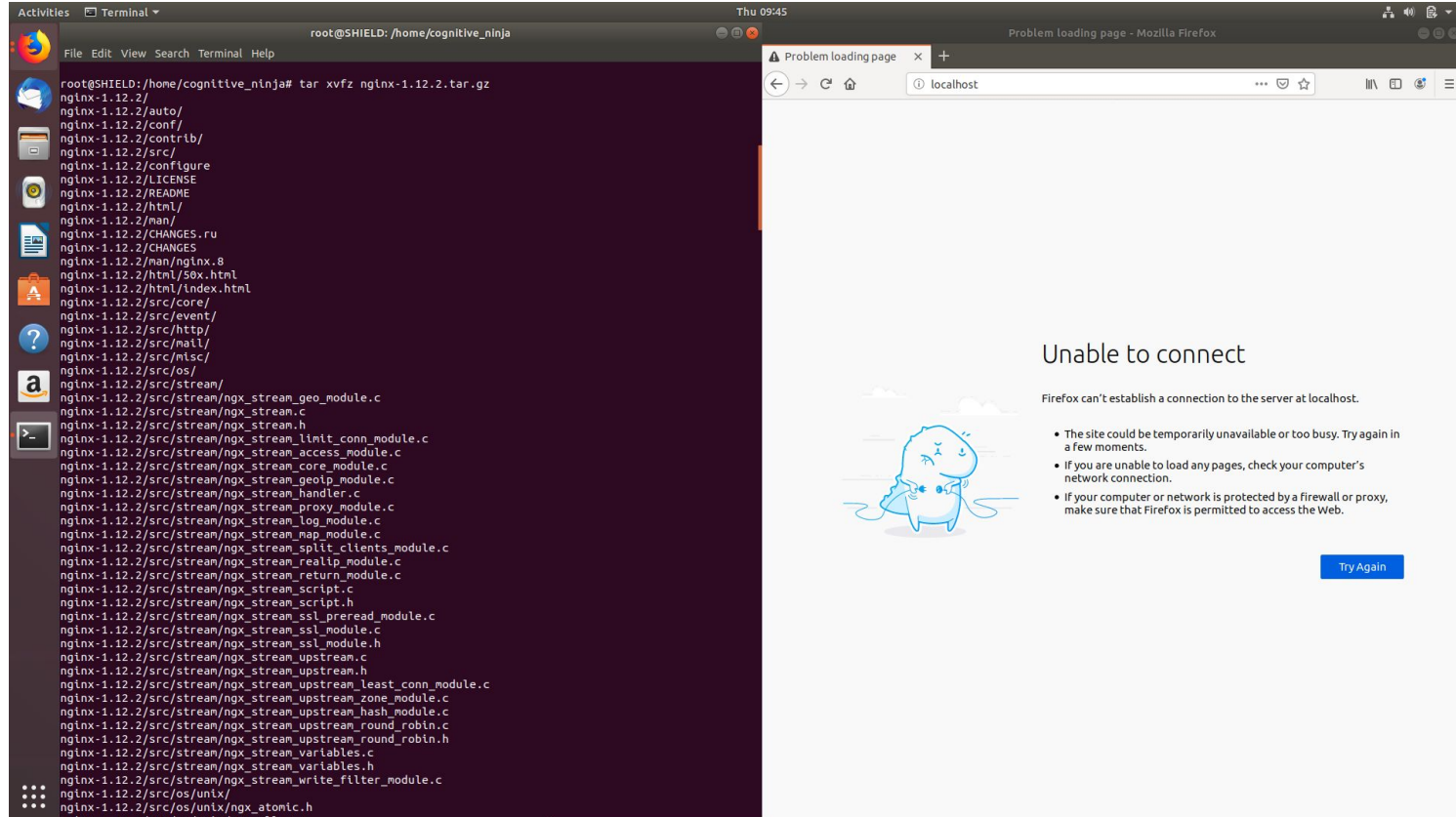3. 2018BTECS00064 – Saurabh Hirugade

Practical No 4

Problem Statement : To install and configure web server.

We decided to install and configure nginx server. For that, we created a ubuntu instance on Oracle VirtualBox to perform this task. The step wise process that we performed is as following:
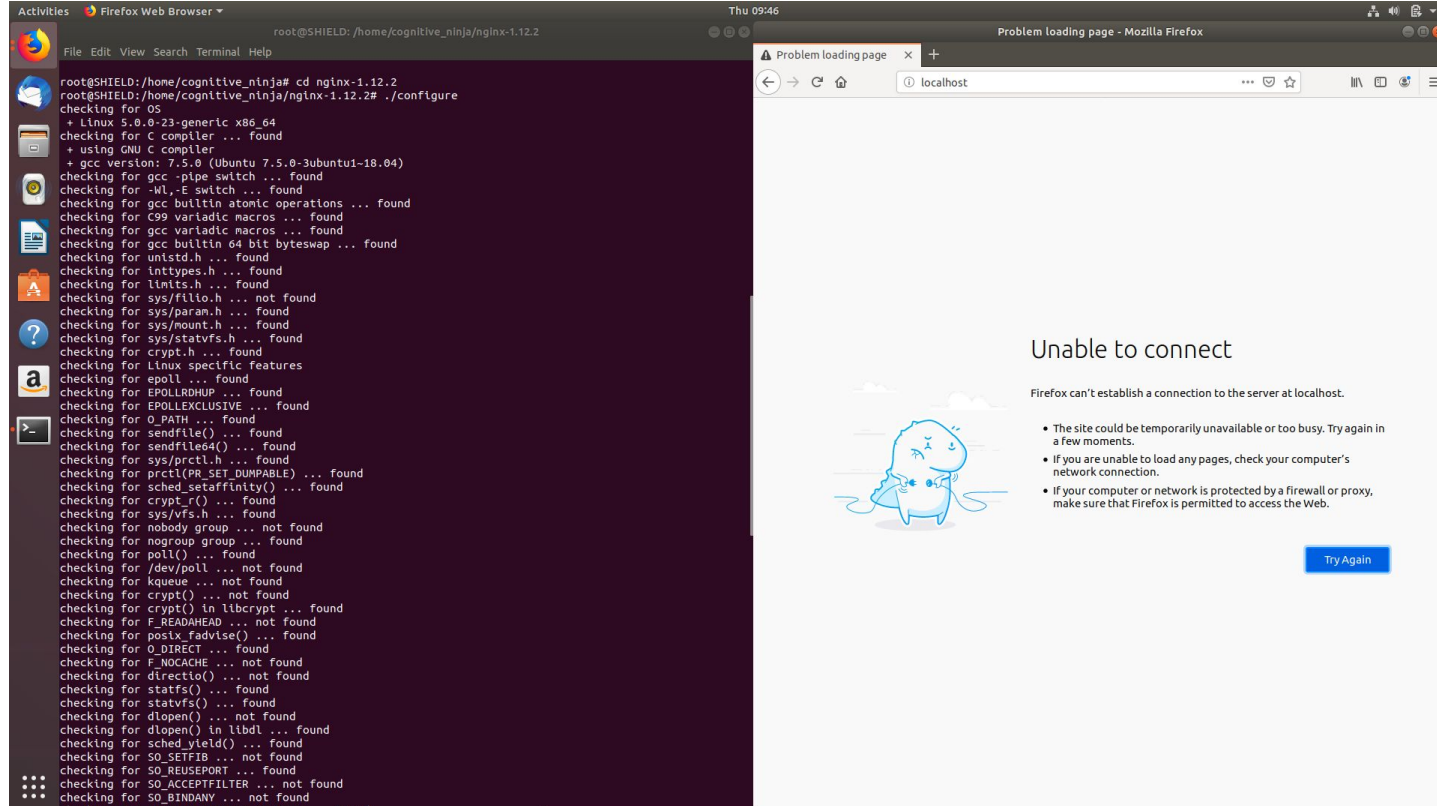
1.Download Nginx' s latest version from the official site using the wget command from the terminal.
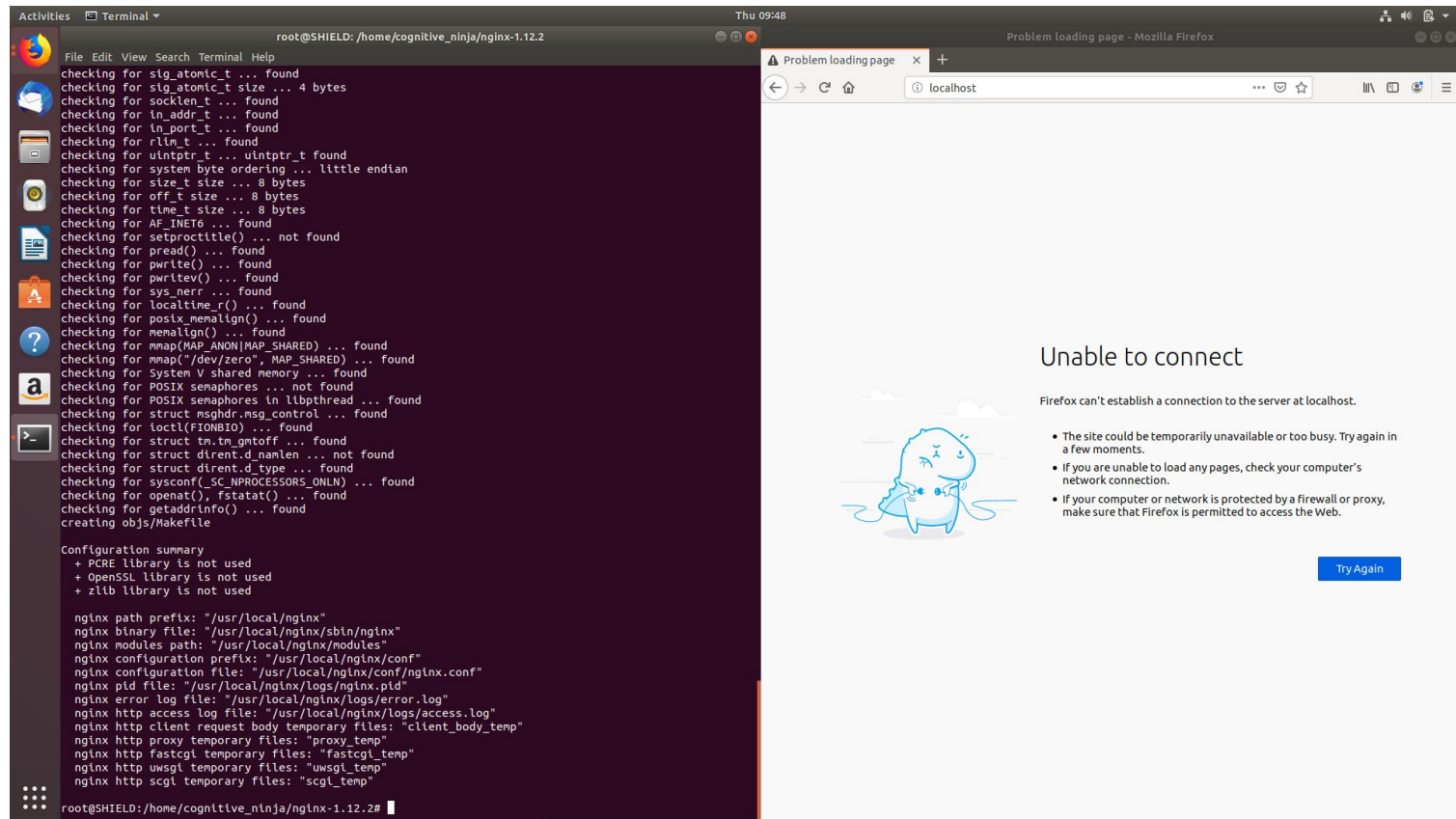
2.After the download completes, extract the package using the following command.
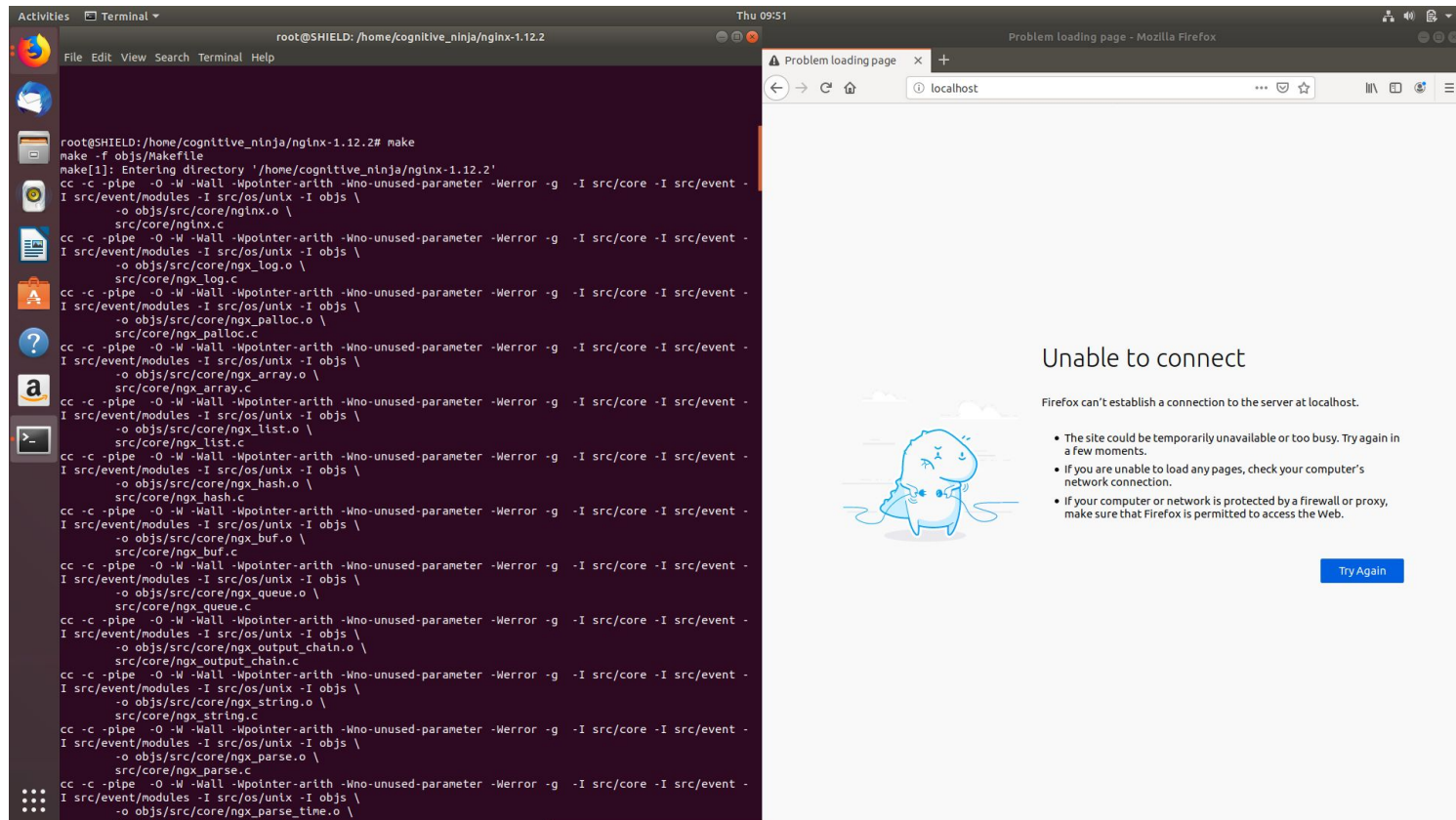
3.Navigate to the extracted directory and list all the files in it and you have to run the configuration file using the below mentioned command.
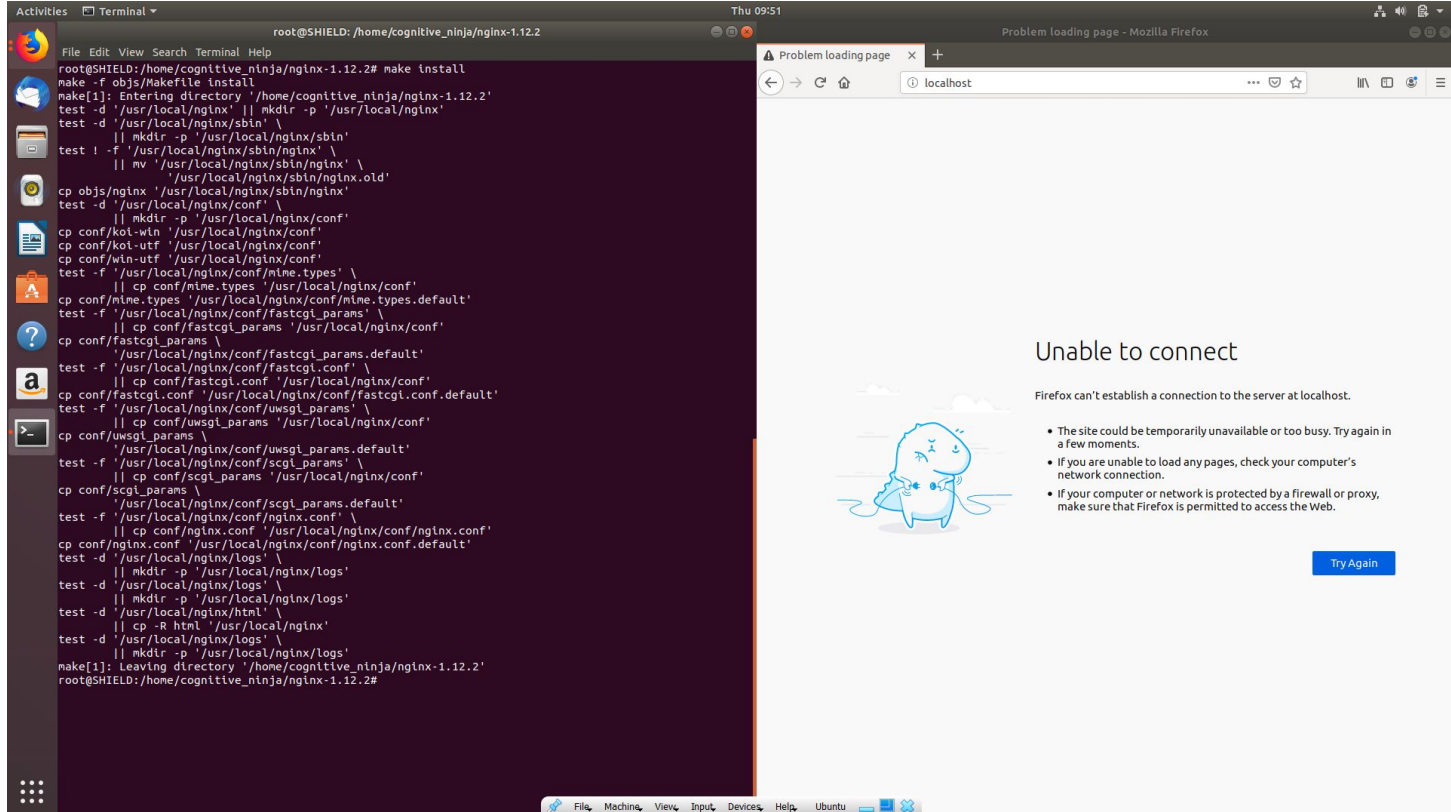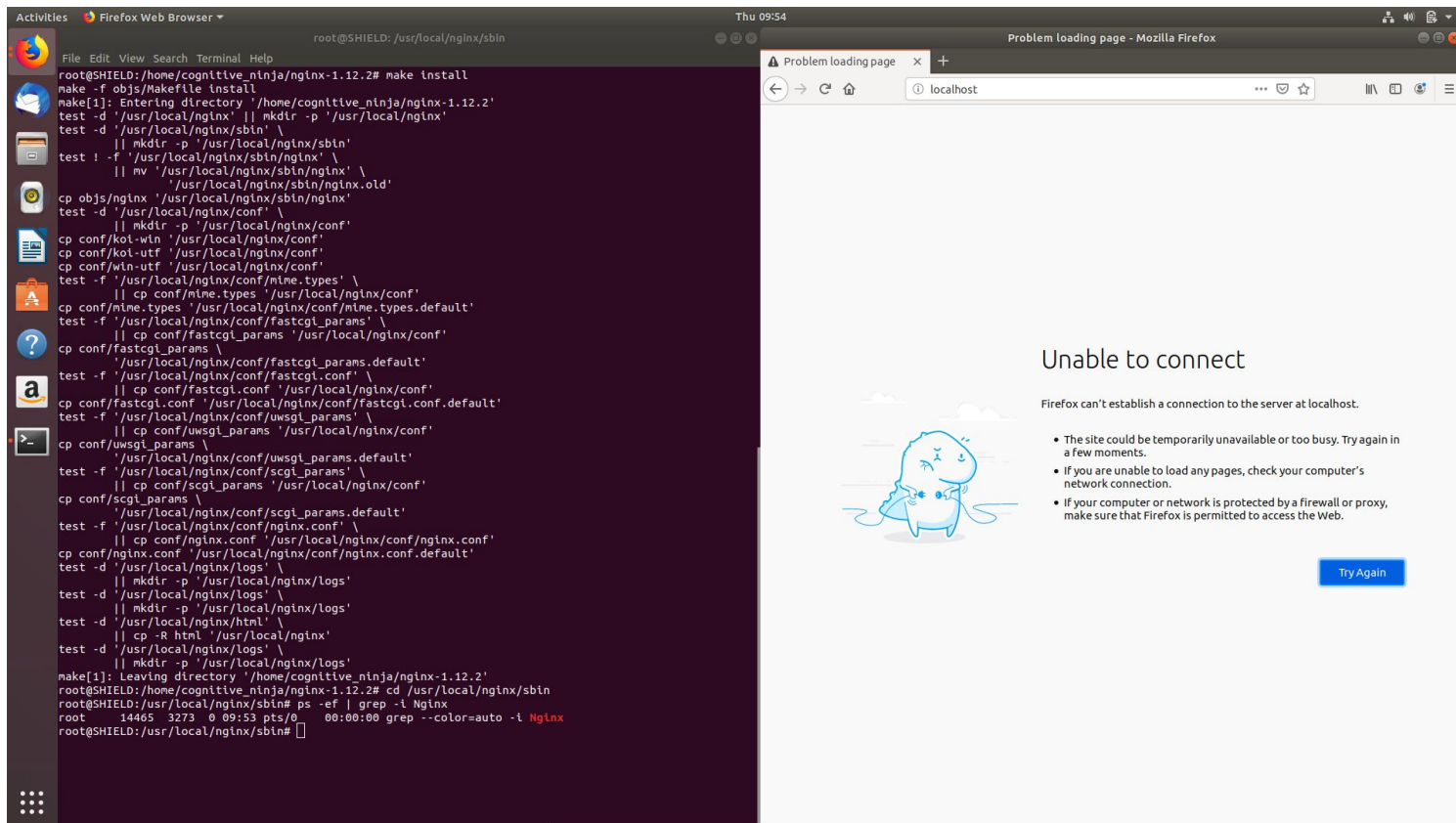
root@SHIELD: /home/cognitive_ninja/nginx-1.12.2

File Edit View Search Terminal Help

```
checking for sig_atomic_t ... found
checking for sig_atomic_t size ... 4 bytes
checking for socklen_t ... found
checking for in_addr_t ... found
checking for in_port_t ... found
checking for rlim_t ... found
checking for uintptr_t ... uintptr_t found
checking for system byte ordering ... little endian
checking for size_t size ... 8 bytes
checking for off_t size ... 8 bytes
checking for time_t size ... 8 bytes
checking for AF_INET6 ... found
checking for setproctitle() ... not found
checking for pread() ... found
checking for pwrite() ... found
checking for pwritev() ... found
checking for sys_nerr ... found
checking for localtime_r() ... found
checking for posix_memalign() ... found
checking for memalign() ... found
checking for mmap(MAP_ANON|MAP_SHARED) ... found
checking for mmap("/dev/zero", MAP_SHARED) ... found
checking for System V shared memory ... found
checking for POSIX semaphores ... not found
checking for POSIX semaphores in libpthread ... found
checking for struct msghdr.msg_control ... found
checking for ioctl(FIONBIO) ... found
checking for struct tm.tm_gmtoff ... found
checking for struct dirent.d_namlen ... not found
checking for struct dirent.d_type ... found
checking for sysconf(_SC_NPROCESSORS_ONLN) ... found
checking for openat(), fstatat() ... found
checking for getaddrinfo() ... found
creating objs/Makefile

Configuration summary
  + PCRE library is not used
  + OpenSSL library is not used
  + zlib library is not used

  nginx path prefix: "/usr/local/nginx"
  nginx binary file: "/usr/local/nginx/sbin/nginx"
  nginx modules path: "/usr/local/nginx/modules"
  nginx configuration prefix: "/usr/local/nginx/conf"
  nginx configuration file: "/usr/local/nginx/conf/nginx.conf"
  nginx pid file: "/usr/local/nginx/logs/nginx.pid"
  nginx error log file: "/usr/local/nginx/logs/error.log"
  nginx http access log file: "/usr/local/nginx/logs/access.log"
  nginx http client request body temporary files: "client_body_temp"
  nginx http proxy temporary files: "proxy_temp"
  nginx http fastcgi temporary files: "fastcgi_temp"
  nginx http uwsgi temporary files: "uwsgi_temp"
  nginx http scgi temporary files: "scgi_temp"
root@SHIELD:/home/cognitive_ninja/nginx-1.12.2#
```

Problem loading page - Mozilla Firefox

Problem loading page

localhost

Unable to connect

Firefox can't establish a connection to the server at localhost.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

Try Again

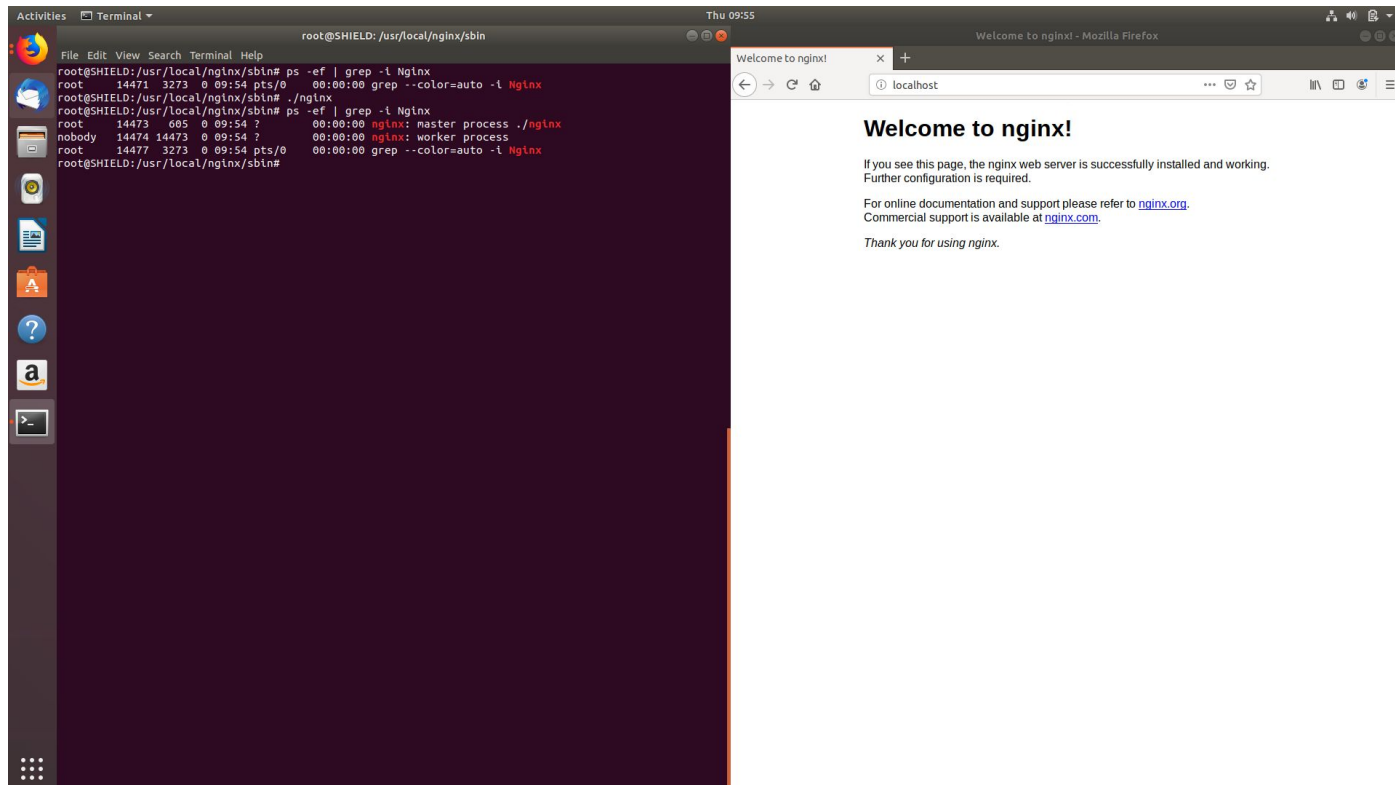**4.After the configuration, we have to make install using the below command.**

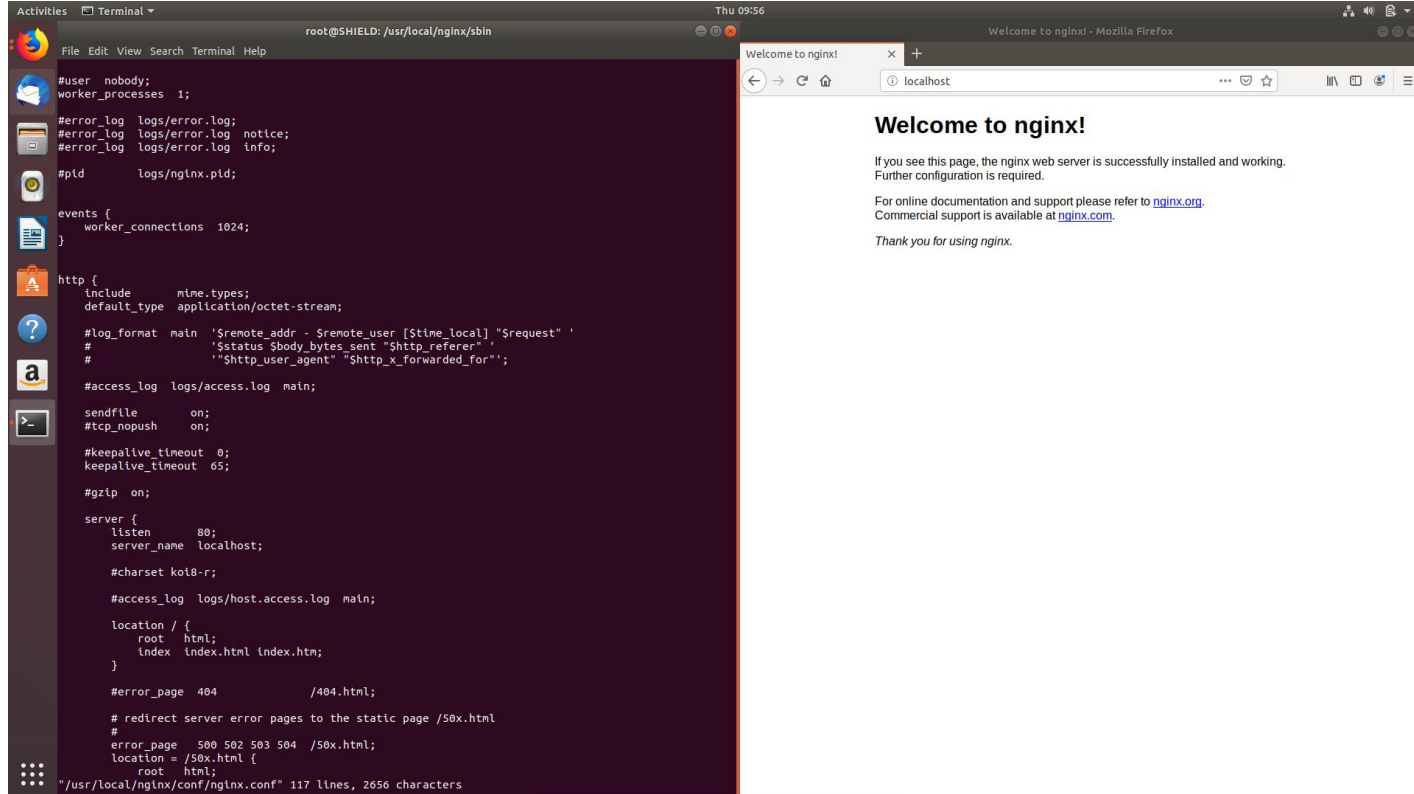**5.After making configuration enter the following command to install the nginx.**

6.Now navigate to the /usr/local/nginx/sbin/ to start the nginx.

7.You can verify the status by using the following command

8.By default, it runs on port 80 so here we changed the listen port by editing in this file.

root@SHIELD: /usr/local/nginx/sbin

File   Edit   View   Search   Terminal   Help

```
#user  nobody;
worker_processes  1;

#error_log  logs/error.log;
#error_log  logs/error.log  notice;
#error_log  logs/error.log  info;

#pid        logs/nginx.pid;


events {
    worker_connections  1024;
}


http {
    include       mime.types;
    default_type  application/octet-stream;

    #log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
    #                  '$status $body_bytes_sent "$http_referer" '
    #                  '"$http_user_agent" "$http_x_forwarded_for"';

    #access_log  logs/access.log  main;

    sendfile        on;
    #tcp_nopush     on;

    #keepalive_timeout  0;
    keepalive_timeout  65;

    #gzip  on;

    server {
        listen       8081;
        server_name  localhost;

        #charset koi8-r;

        #access_log  logs/host.access.log  main;

        location / {
            root   html;
            index  index.html index.htm;
        }

        #error_page  404              /404.html;

        # redirect server error pages to the static page /50x.html
        #
        error_page   500 502 503 504  /50x.html;
        location = /50x.html {
            root   html;
        }
"/usr/local/nginx/conf/nginx.conf" 117 lines, 2656 characters
```

Welcome to nginx! — Mozilla Firefox

Welcome to nginx!   ✕   +

localhost

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working.
Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

root@SHIELD: /usr/local/nginx/sbin

File Edit View Search Terminal Help

```
root@SHIELD:/usr/local/nginx/sbin# vi /usr/local/nginx/conf/nginx.conf
root@SHIELD:/usr/local/nginx/sbin# ./nginx
```

Welcome to nginx! - Mozilla Firefox

Welcome to nginx!   +

localhost:8081

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working.
Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.
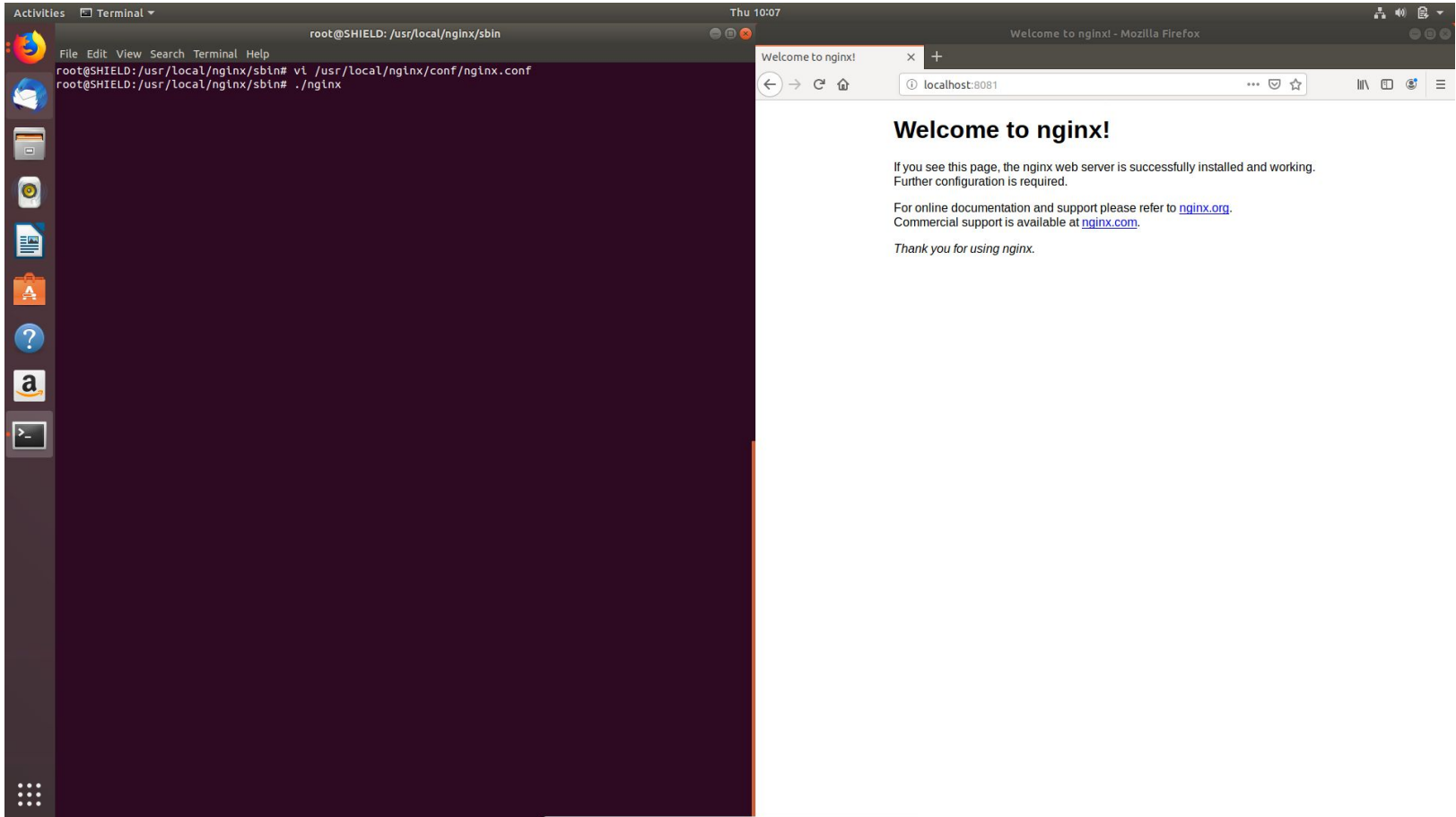
Since the Nginx server was running on a virtualbox, it wasn't directly accessible on a mobile device even if in the same subnet.
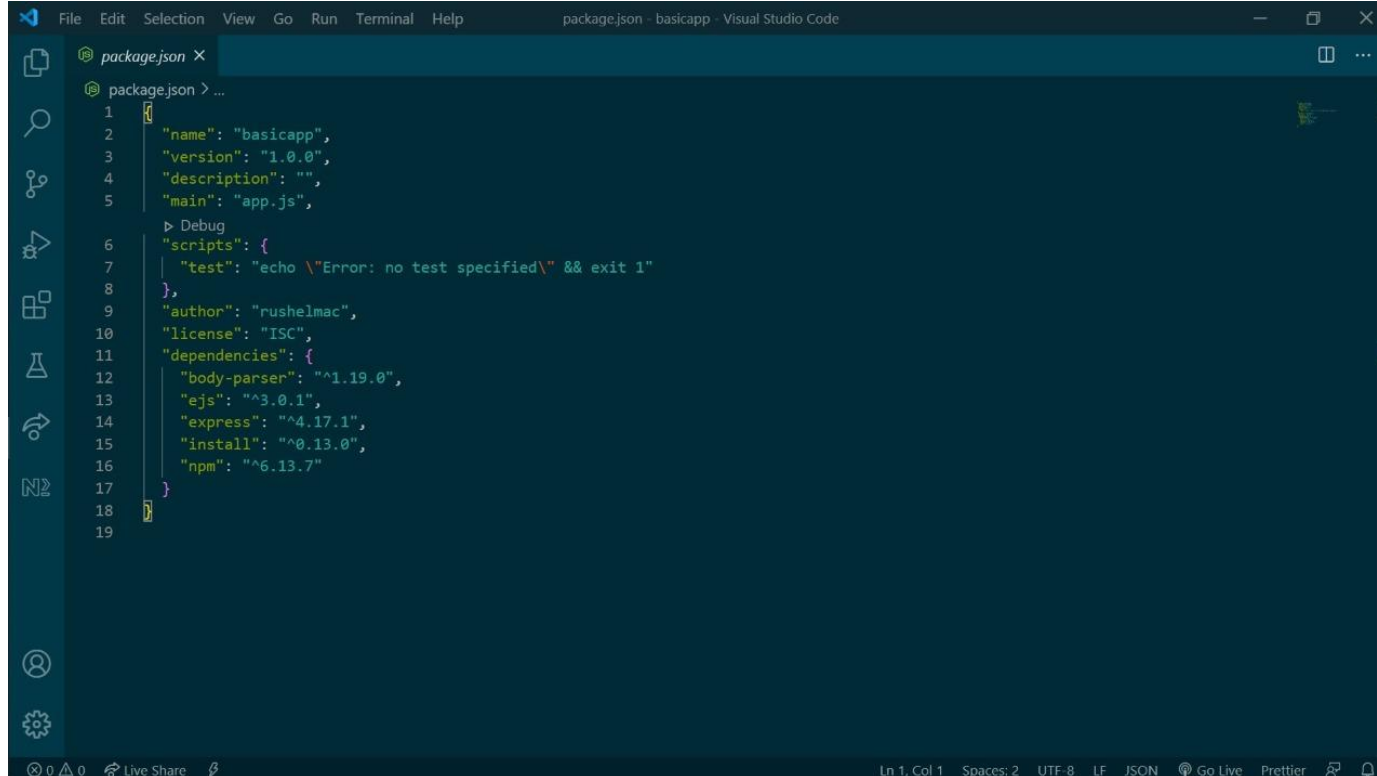
So we created a node server just for the sake of completeness in the practical.

Question asked in the presentation:

Que: Can we use other virtual instances of operating systems and treat them as different devices in a single subnet and access the server?

Ans: Yes. It is possible. The virtual instance in our case had a private IP(Non routable) address (due to NAT), hence it was not accessible. It was like two private networks with a router between them. So it can be done by adding the VM as an additional node in the local network by attaching VM's Network adapter to "Bridged adapter". It will basically share the internet connection between them.

This is the configuration file of node.js server :

# Starting the server on localhost and changing ports.

This is from where we changed the default IP address of localhost in Windows OS.
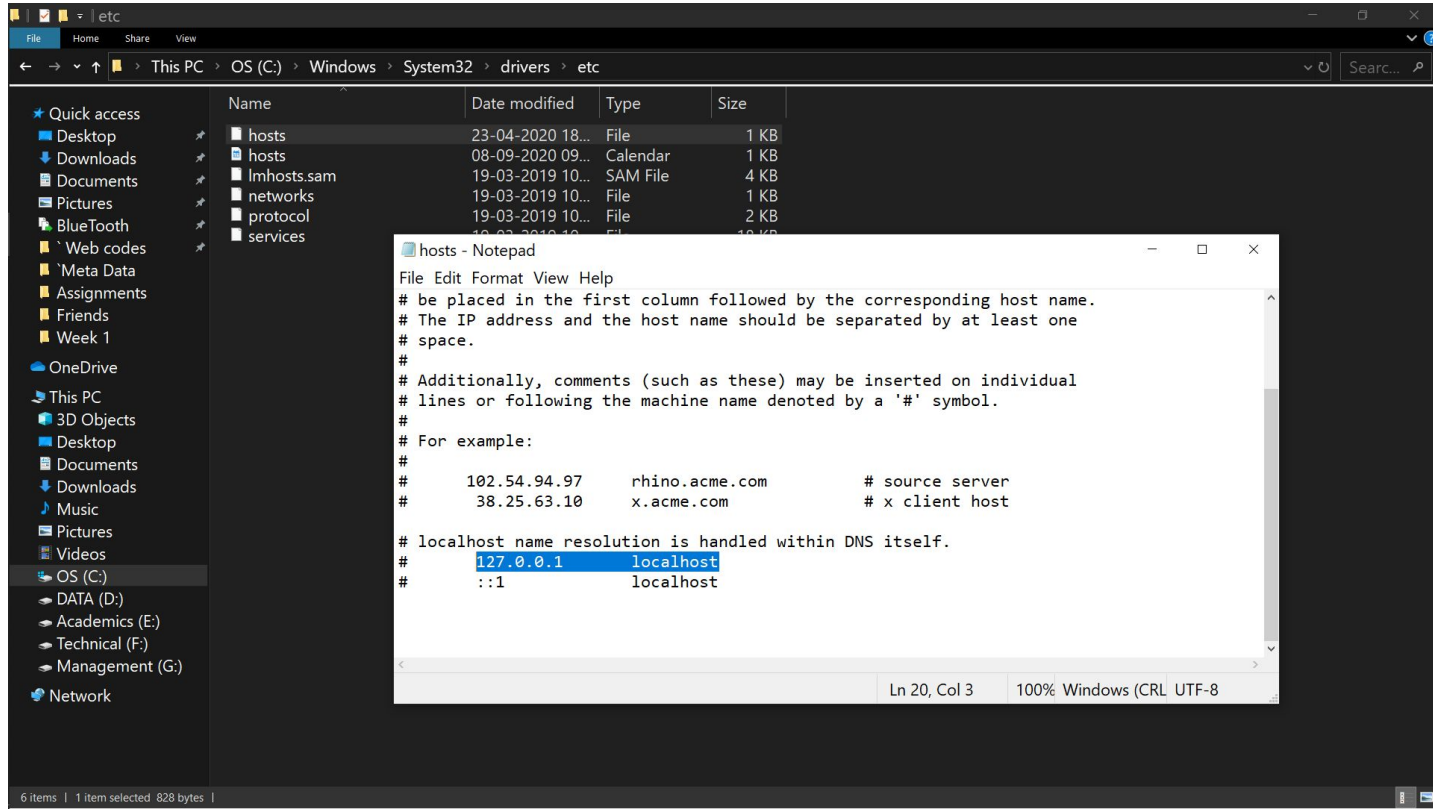
Then we accessed the server on a mobile device. Screenshot ->

We got the IP address assigned to the server machine by using the command 'ipconfig'.

It was 192.168.43.48

9:54

0.0KB/s

192.168.43.48:5600

Hello There ..!! You are connected to my local server
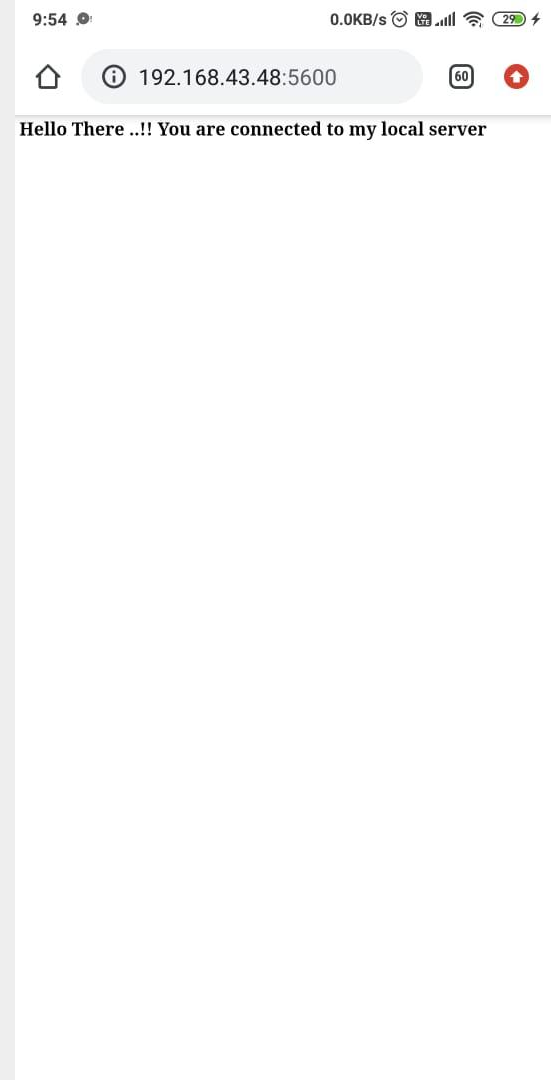
Contribution of team:

Actions performed:

1. Installation of Nginx server and execution of further instructions. (Linux OS)
2. Installation of Node server and execution of further instructions. (Windows OS)

All of the above actions were performed together by the team using online meeting platform.