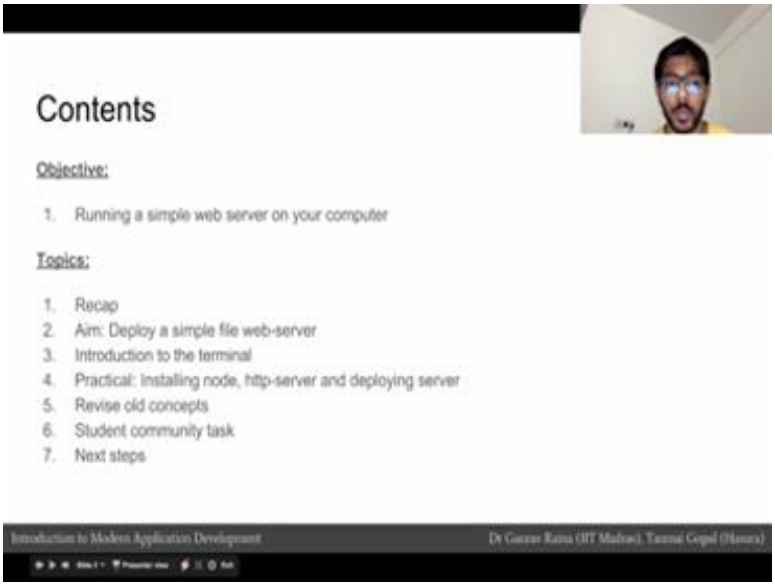
Introduction to Modern Application Development
Prof. Tanmai Gopal
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module - P1
Lecture – 06
Practical: Running your own webserver

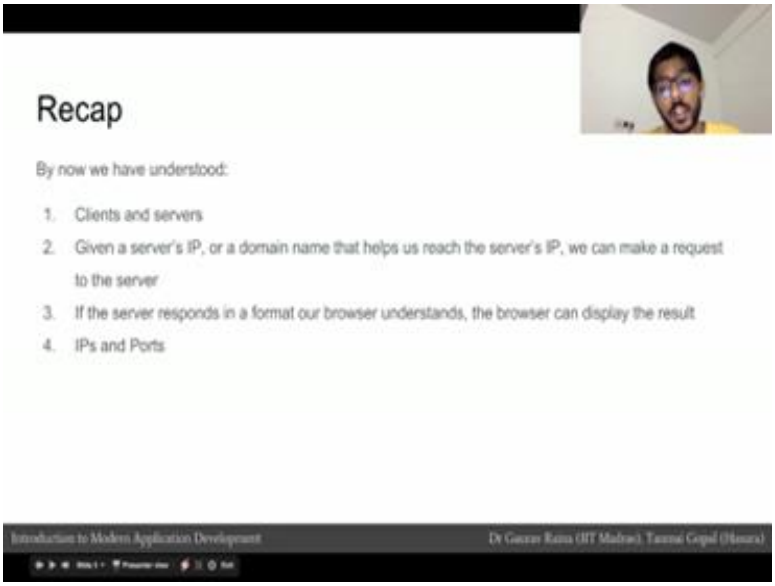
Hi all. Welcome to module P1.

(Refer Slide Time: 00:03)



In this module, which is the first practical module, will be understanding how we can run our own simple web server and this will be running on our computer.

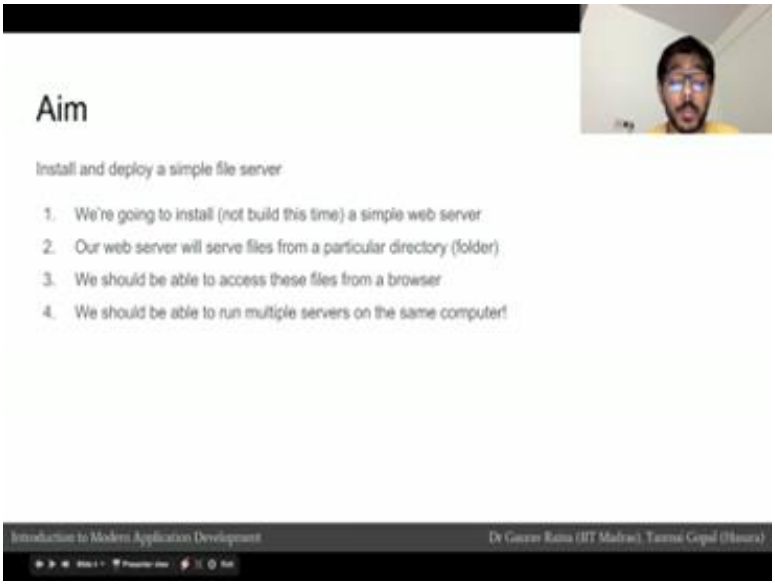
(Refer Slide Time: 00:12)



So, let us quickly recap what we learnt. So, far through module 1, 2, 3 and 4

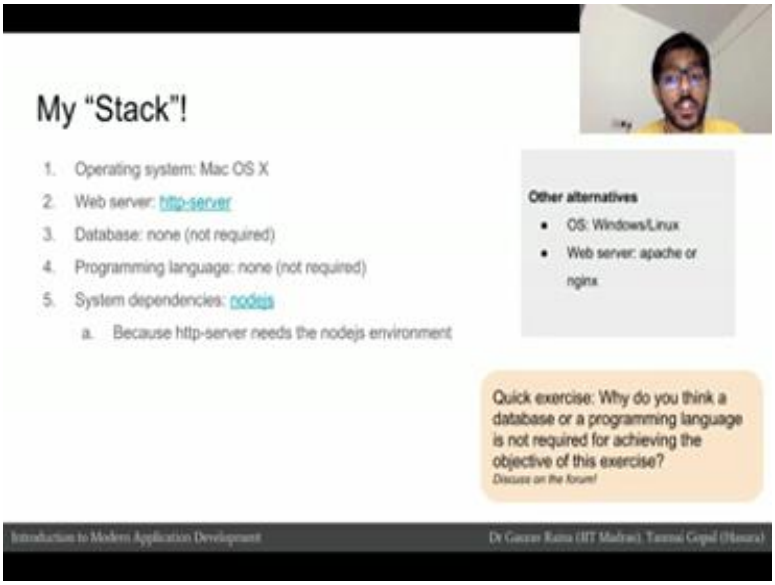
We have sort of understood what clients and servers are we have also understand that if we have a servers IP or if we have domain name that translates into the servers IP we can make a request to the server. Then if the server makes the response in a format that a browser can understand which our browser was in html format or a png image or a text file then can display the result. We have also had a brief introduction into what ips and ports are.

(Refer Slide Time: 00:49)



Our aim in this module is to install and deploy a simple file web server and we are going to do this on our own computer. We will use the web server to serve files from a particular directory a folder is often technically called directory. We will then see if we can access these files on a browser, and once we set up a basic server on our own computer, we will see if we can run multiple servers on the same computers.

(Refer Slide Time: 01:12)



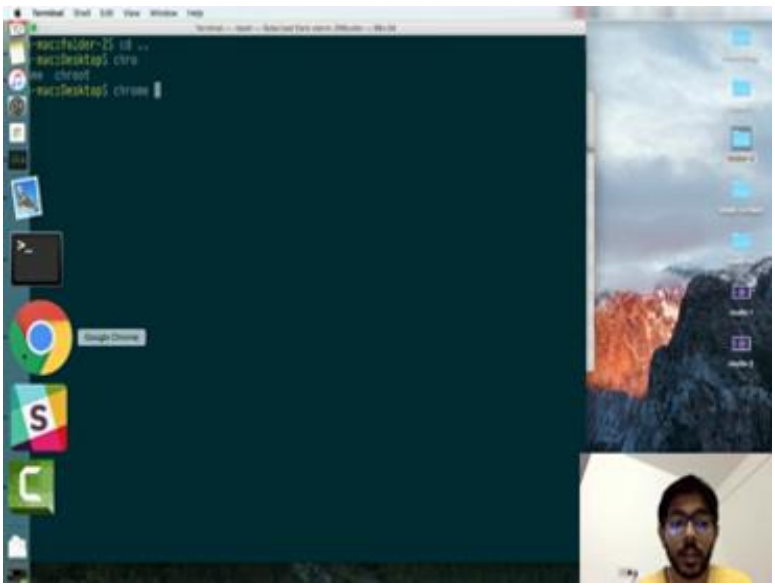
Since I am going to be running this experiment on my own computer, this is what my stack looks like, my operating system is MacOS x because my laptop is a Macbook, the web server I will be using is http server. Http server is the name of the software package if you click on the slides it will take you to the name of the package; I will not be using the database or a programming language for this simple experiment.

I do have to install something on my system which is nodejs, and the reason I have install this is because http server which is the package the software it needs the nodejs environment. To be very clear this is just an example configuration, I can do the exact same experiment with a completely different set up for example, I can use a different operating system say windows or Linux I can have a completely different web server like apache or nginx, a quick exercise for you for guys is to think about why I do not need a database or programming language for this simple experiment.

(Refer Slide Time: 02:13)

Now, during this experiment you see that I use the command line tool or the terminal application quite often, the terminal is something is seems quite scary to a lot beginners when they start, it kind of a reminds you of metrics with a black screen a lot of green numbers going up and down, but it is important to understand that the terminal or the command line interface it is just another interface to do what we normally do with the mouse and keyboard let me show you an example of what using the terminal is like. So, this is my desktop.

(Refer Slide Time: 02:44)



As you can see on my desktop I have 3 folders and 2 photographs. I can look at these exact same files through a different interface which is a terminal, I open the terminal application I am going to make it little larger and increase the font size. So, I can go to the desktop by doing CD desktop which reads change directory desktop. Now, I am in the desktop and I can do an ls, ls is a command to say list files.

So, now, I use my keyboard to type the command ls and I see a list files and you can see that its saying that I have 3 directories which are on different color and 2 files which are in different color which are png files, this is the exact same as looking at the interface here for example, I can say. So, I can see now these files in more detail because I passed a more specific command and this shows me the same files, but what the different sizes are. So, this for example, says that this png image is 215000 bytes this 1 is about 215000 bytes which about 212 k b.

Now, in case I for example, create a new folder and I call it folder1. I can do the same thing from the terminal m k d I r folder 2and you see that folder 2 has appear I can double click and see the contents of folder 2, but when I double clicked the file browser opened up and showed me the contents of the folder. I can do the same thing I can also open the file browser because a file browser is another program, I can open the file browser from my terminal.

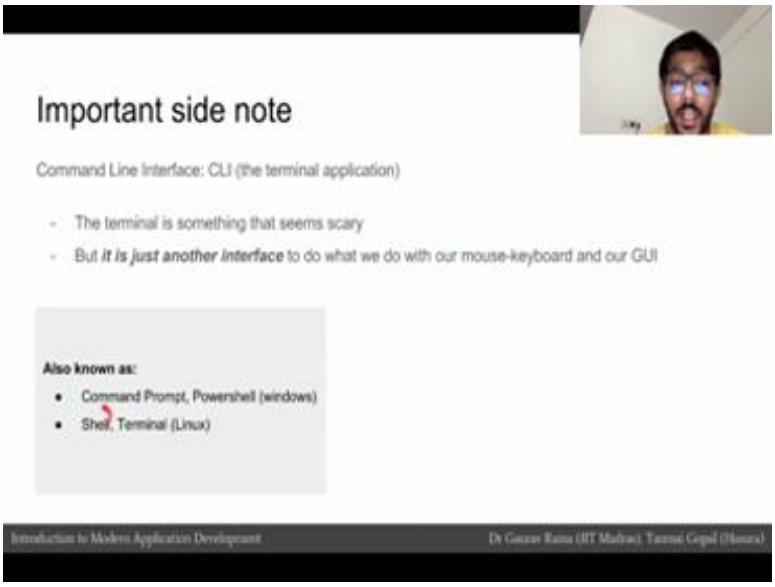
So, I go inside folder 2 and I say open, and then you see that the file browser is open. So, this act of me executing this command from here was prevalent to me double clicking on folder 2 and opening the file browser here. Let us say I create a new file I will use a text editor called vin.

So, I have created a new file in this folder, I go back and I look I can see that the file is in this folder I can double click on this file and edit it from more friendly text editor, which is a editor like notepad it is called text edit in Mac and I can edit the content and say this is a new line and I can save it I can exit it and I can continue editing from the terminal, through my terminal text editor the 1 that I use is called vim vin, and I can say new line I can paste it a bunch of times save it I can look at the file size, I can look at the file size in a more friendly format and size is about 4 k b in size.

So, the terminal is the place where I can browse files and folders, I can execute command for example, I can even open up chrome, and this is list and executing this

command would be relevant to clicking on the chrome browser here and opening it now the terminal software that I used on my Mac to do this is available in other operating systems also.

(Refer Slide Time: 06:04)



There is something called the command prompt or power shell on windows which is equivalent to the terminal which is sort of equivalent to the terminal app on Mac, it is also called shell or terminal on Linux.

(Refer Slide Time: 06:19)

Now, let us move on to actually deploying our web server. So, the first thing I need to do is to install nodejs and then install http server. Let us go these links and how we do that. So, once you go the nodejs page it will prompt you for the different operating systems and you install the right version for your operating system, once you install node which I already has will then install http server once you go the http server page you can scroll down and it will give you a command for installing http server globally, now this command is a command that you execute in your command line. So, I can copy this command go to my terminal and paste this command.

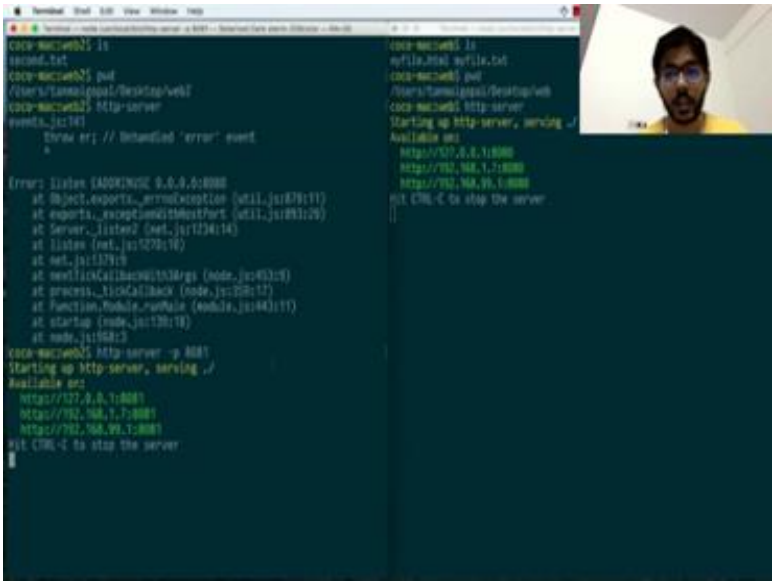
The next step is to create directory is. So, now, I am going make a directory called web I go inside a web I add content that I want serve inside that directory, and so I am going to sayandso, my file is added and then going run http server in this directory. So, type for the command, and I once runs http server it says serving available on.

Let us go our browser and let us go to this URL; I copy it and its shows me my file dot txtand if I click on it I can now open my file. So, we have a web server that serving a file. Let us stop the server if I press control c I can stop the serverand let us add and let us add some more files. So, let us add an html file let us start let us start HTTP server again if I (Refer Time: 08:44) paste it still here if I change this to HTML I can see my html file.

(Refer Slide Time: 08:25)

The interesting thing to note that, when I connected to my Wi-Fi network on my LAN my IP is this.

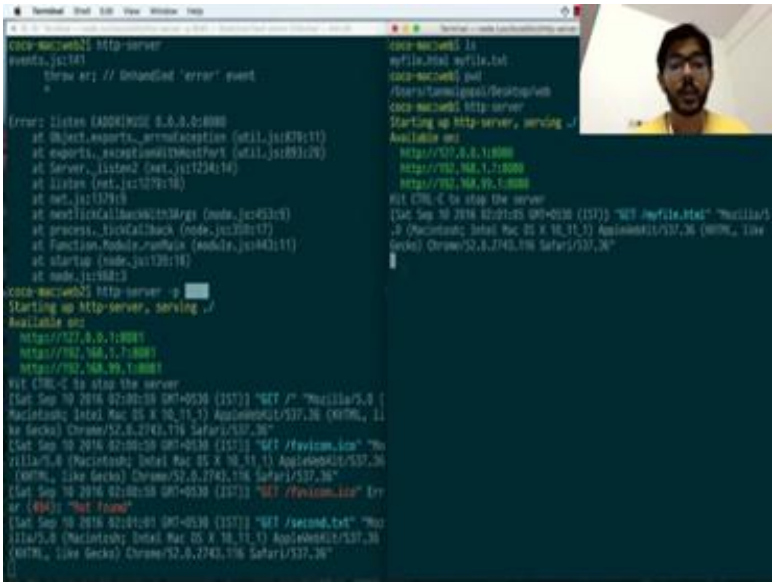
(Refer Slide Time: 09:11)



So, this means that I can also access my server on this IP right is to use the host name local host now I am getting the which says that the I cannot be reached which is because I switched my server off, and switch it on again and my file is and my server is (Refer Time: 09:31). So, all of these host names or ips are interact pointing to the same server the next thing that you want to do is create another directory.

Let us go back control c and stop our serverlet us add another folder m k d I r web 2and in this folder I am going to have second dot txt this is the second web server, serving a text file and I am going to open another instance of terminal you see that I have 2 terminals open I am going zoom that little bit and. These are 2 different directories when I move to the desktop. So, as you can see here I am in a directory web and here I am in the directory web 2, I can run the http server here now if I try to run the http server here, the error thrown is address already in use. So, this means that port 80, 80 already being used by another server. So, I can change the port and use port 80, 81.

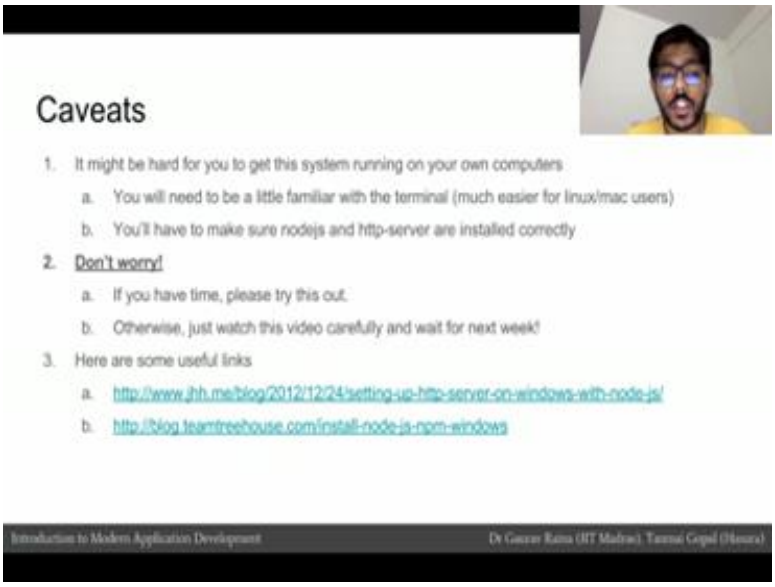
(Refer Slide Time: 10:57)



Let us now go back and check this out, let us go to local host 80 81 and you can see second dot txt if you click on second dot txt you can see a second web server.

So, simultaneously we have 2 servers running on the same IP, and the only where is can happen is because of the concept of ports. Ports allow me to run multiple servers on the same computer.

(Refer Slide Time: 11:27)

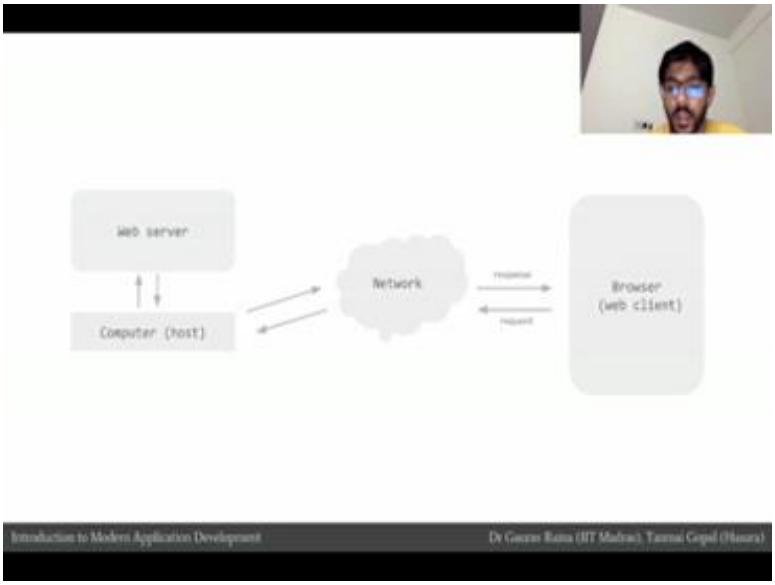


So, I just installed nodejs http server and then got 2 web servers running on my computer it might be hard for you to get the system running on your computers, because you will

be need you need to be a little familiar with the terminal which is usually much easier for Linux or Mac users.

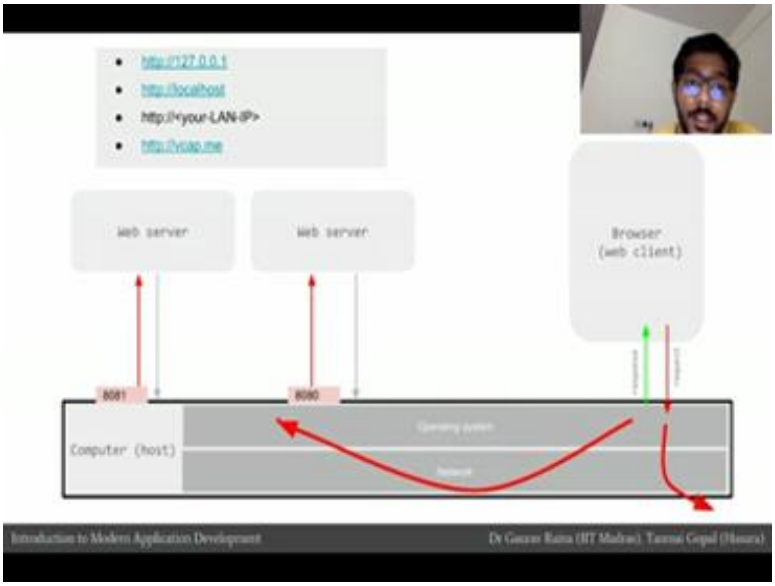
You will also make sure that nodejs and http server are installed correctly if you have time please do try this exercise out, but if you are running out time and you are not familiar with the system enough or you are not getting the instructions working correctly, make sure you watch this video correctly and understand what I was doing, when I was running the commands and what I was doing, when I was accessing the files after creating the web servers. There are a few useful links that are found that I have mentioned on the slides here that you can try for installing nodejs and npm and http server on windows.

(Refer Slide Time: 12:16)



So, let us try to quickly see how what we did fix into our understanding from what we have understood in the previous modules. We understood that a browser is web client which makes the request through the network to the computer to the web server and the web server responds back on the same connection following the same route.

(Refer Slide Time: 12:34)



What we did here however; was that the browser and the server who sitting on the same machine our browser made a request and somehow it went to the web server sitting on our own computer and the web server responded and the response came back on to our server this is what really happens. When the browser tries to request, a server the browser mentions the IP the browser send up the connection via an operating system call the operating system detects the IP is not an external IP, but it is a special internal IP that belong to the system itself.

So, the operating system looks the request back into the system then the port number is used to figure out which process the system needs to request in our case we made request to IP colon 8080 and it went to the process that was listening on the port number 8080 the request goes up to the web browser come back down through the same connection.

Similarly, when we made a request to our IP 80 81 the connection again look back came to the process that was on the port 80 81 and click the web server these special ips that redirect back to our own system our 127 001localhost is the host name v cap dot me it is also a host name that often redirects back to the same system your LAN IP will also look back to your own system.

(Refer Slide Time: 14:11)

If we go through the steps that we discussed about building a web app in this case we built a very simple web app they just serves files, but this web app cannot be used by the people other people cannot use our web app that we just made a this this simple file server because our IP is not a static IP our IP is not public to the world if our IP was public to the world we could have attached domain name to our IP and then people could have use our web app by using the domain name because we do not have a static IP other people can't use our app, but others on our own LAN can use our app because our IP is known to everybody on the same LAN second step that we discussed, when we wanted to build up a web app was to set up a computer with an IP to serve the web app.

This process was done because you were using our computer which had our own IP we will not have to link the domain name to the servers IP address because you were not using the domain name, in our case we were using a special host name which was local host and this process is automatically done.

The forth step that we discussed was to write application code or to write a web server, which we did not need to do in this case because http server the software that we used automatically serves static files, the 5th step which was deploying our application code or a web server on to web server host also need not be done because in our case our server was our computer.

(Refer Slide Time: 15:42)

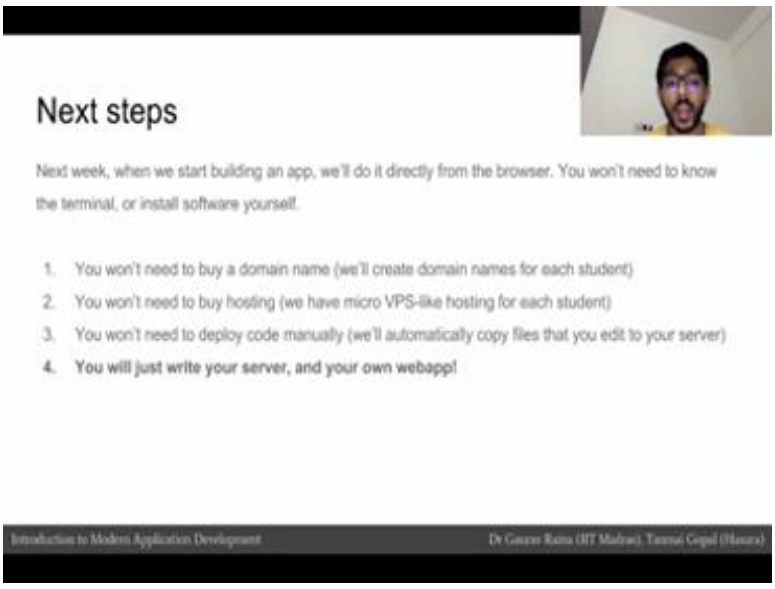
We also discussed how we would write application code to deploy our web server in this case. We are not writing any application code we are simply installing pre configured web server we discussed that the steps involved to write the application code would be to listen on the network accept a request, process the request, output response and send the response back.

We did not have to write any code and we simply used free configured server because http server the software similar to apache or nginx is preconfigured to request in a particular way and respond in a particular way. It processes request by assuming that URL path are actually file locations and it responds with the contents of the file that it finds.

(Refer Slide Time: 16:25)

I did this experiment on my computer which is running the operating system OS x I encourage all of you to try this experiment on windows Linux and even Mac, record videos of how you did this experiment upload them to YouTube and post them on the forum. So, the other people can also learn from these videos.

(Refer Slide Time: 16:43)



The next week when we build our webapp we will directly do from the browser you will not need to learn how to use the terminal or install software by yourself. When we build

webapp you will not need to buy domain name because we will have created domain name for each student.

You will not need to buy hosting alsobecause we have micro VPS like hosting for each student you will also not need to learn how to deploy a code using SSH or FTP because we will automatically copy files for you that you edit, and will automatically copy them for you onto your server. You will just write your server and write your web app and it will automatically work on the domain name that we give to you.