VSS Semester Project

Student Name: Haroon Allahdad

Reg: 652-FBAS4/BSIT4/F21

Date of Submission: Expected 30th, October 2024

Submitted To: Mr. Majid Bashir

University: International Islamic University, Islamabad

Revision History

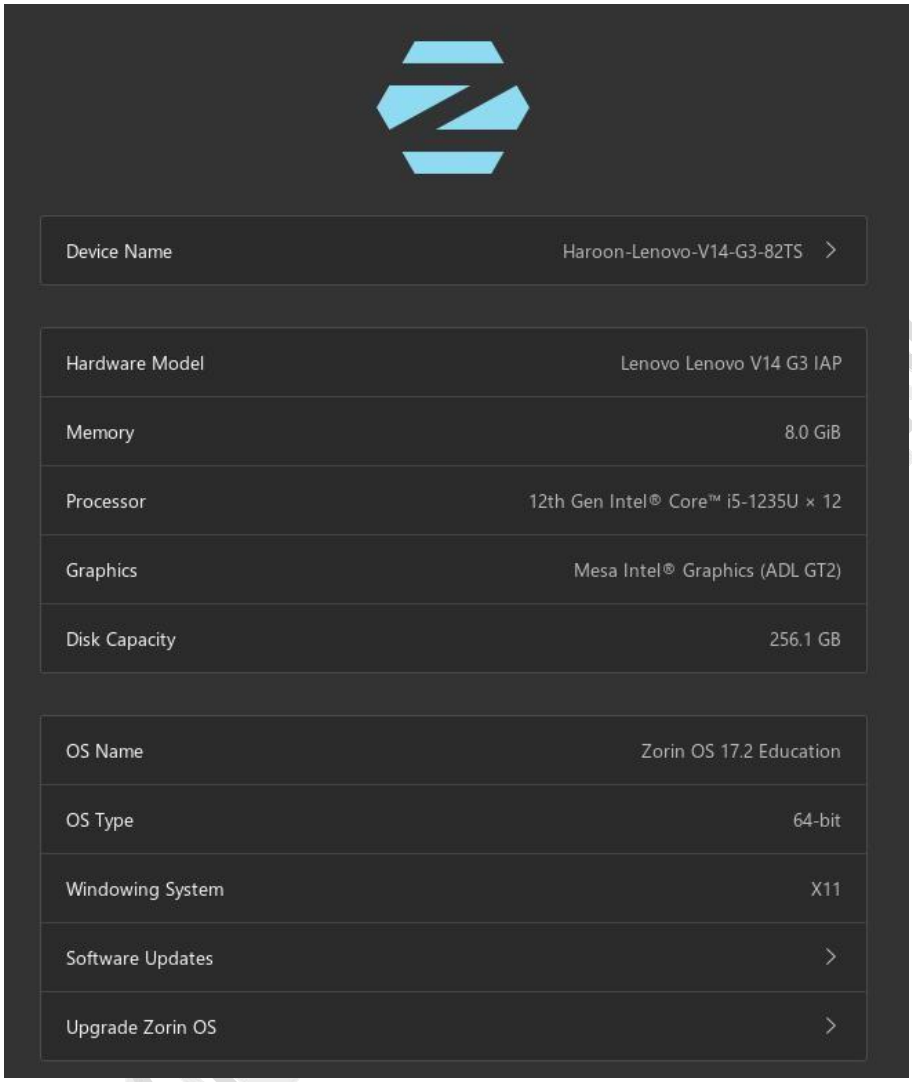
Name	Date	Project Description	Version
Haroon Allahdad	26th, Oct 24	Performed through a local Hypervisor, on Local Machine VM's and Webhosting was done through IIS	1.0
Haroon Allahdad	26th, Dec 24	Performed through Microsoft Azure, through Online Virtualization services, and same task as the version 1,0 was done, additionally My hosted website was also available for online view all over the world wide web.	2.0

Major task 1: Host a Website on a Local VM and access it through your own host OS and a local Client VM through it's name. Perform the step by step tasks.

Task:

Install an environment like Virtual Box which may help in creation of VM's for the virtualization task. Default VM's like Hyper-V Manager are appreciated. Next create at least 2 VM's with one acting as a server VM and the other acting as a host VM. After that show that both the VM's are working and functional. After that enable the IIS feature on the Server VM and host a website on it. Next create a connection between both the VM's and access the website from the Client OS to the hosting Server VM.

Desktop Specifications:


Device Name
Haroon-Lenovo-V14-G3-82TS
Hardware Model
Lenovo Lenovo V14 G3 IAP
Memory
8.0 GiB
Processor
12th Gen Intel® Core™ i5-1235U × 12
Graphics
Mesa Intel® Graphics (ADL GT2)
Disk Capacity
256.1 GB
OS Name
Zorin OS 17.2 Education
OS Type
64-bit
Windowing System
X11
Software Updates
Upgrade Zorin OS

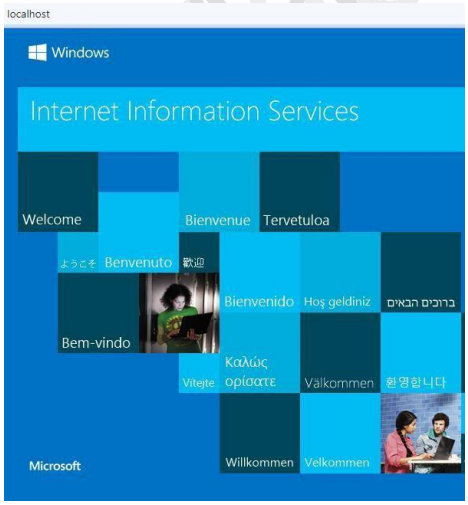
Steps performed to clear the task:

1) As a Zorin 17 (Linux Debian Based Distribution), I have a default hypervisor known as Virtual Machine Manager (https://ubuntu.com/server/docs/virtual-machine-manager). I used it as a substitute for the Hyper-V Manager of Windows.

2) I installed two ISO files, from the internet one for Windows 10 (https://www.microsoft.com/en-in/software-download/windows10ISO?msockid=0e6ee6c59f376fbd159ef3d99e656e04) and one for Ubuntu 24.04 (https://ubuntu.com/download/desktop) and I created a connection on my VMM and added three VM's on it through the connection established on QEMU/KVM Hypervisor connection. I had to take YouTube's assistance for this as it was a new interface for me being a Linux user.

3) I opened the VM's separately and performed the installation. I used one Windows 10 VM as a Server VM and another Windows 10 as a Client. I have also used the Ubuntu one as a Client, but here I will simply show the Windows Client and Server functionalilties.

4) I logged into my Server Windows and went to **Control Panel** > P**rograms and Features** > **Turn Windows Feature On or Off** > Click on **Internet Information Services** and enable it. To check it further I had opened IIS and seen the interface and further went to a browser (Microsoft Edge) and typed localhost in the search bar.



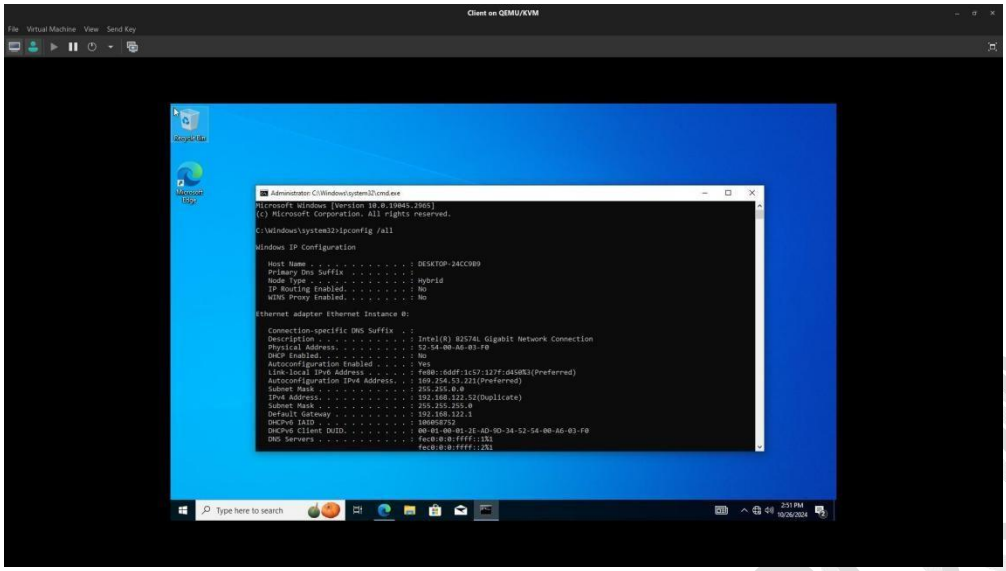
5) Initially I easily found the IP addresses of the devices through ipconfig and performed the ping command, but I noticed that after I turn off my devices (VM's), and reboot or start them again, the connection was no longer maintained, so by the help of copilot, and online searching, I found the solution which was fairly easy and suitable. It was to change the IP addresses from automatic to manual and hard-code the IP addresses to maintain a permanent connection. So I performed it by following these steps.

First go to the administrator command prompt and turn off the firewall from both the Server and Client Devices. Use this code: `netsh advfirewall set allprofiles state off` to verify that the changes you made have been done, `type netsh advfirewall show allprofiles` and confirm.

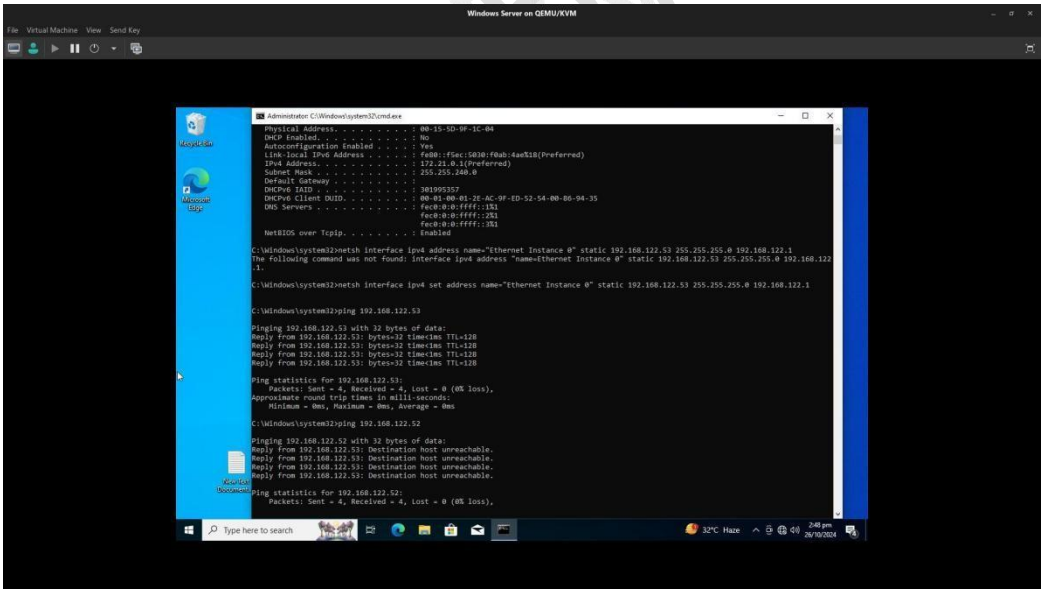
Now I typed `ipconfig /all` in the server VM and noted the interface_name, the actual IP address, the subnet_mask and the gateway. And then I typed this command I got from a website:

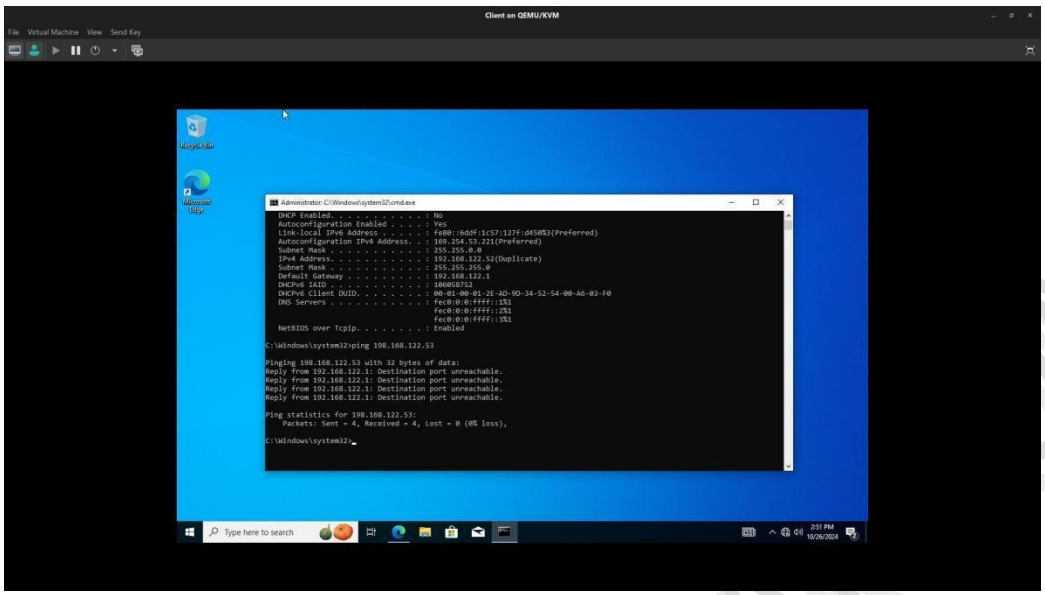
`netsh interface ipv4 set address name="<interface_name>" static <new_ip_address> <subnet_mask> <gateway>` while replacing the <info> with my actual values, which was `netsh interface ipv4 set address name="Ethernet Instance 0" static 192.168.122.53 255.255.255.0 192.168.122.1`

Then I performed the same steps in my client VM, inputting this command: `netsh interface ipv4 set address name="Ethernet Instance 0" static 192.168.122.52 255.255.255.0 192.168.122.1` and hence I got the Server IP as a static 192.168.122.53 and the Client VM's IP as a static 192.168.122.52

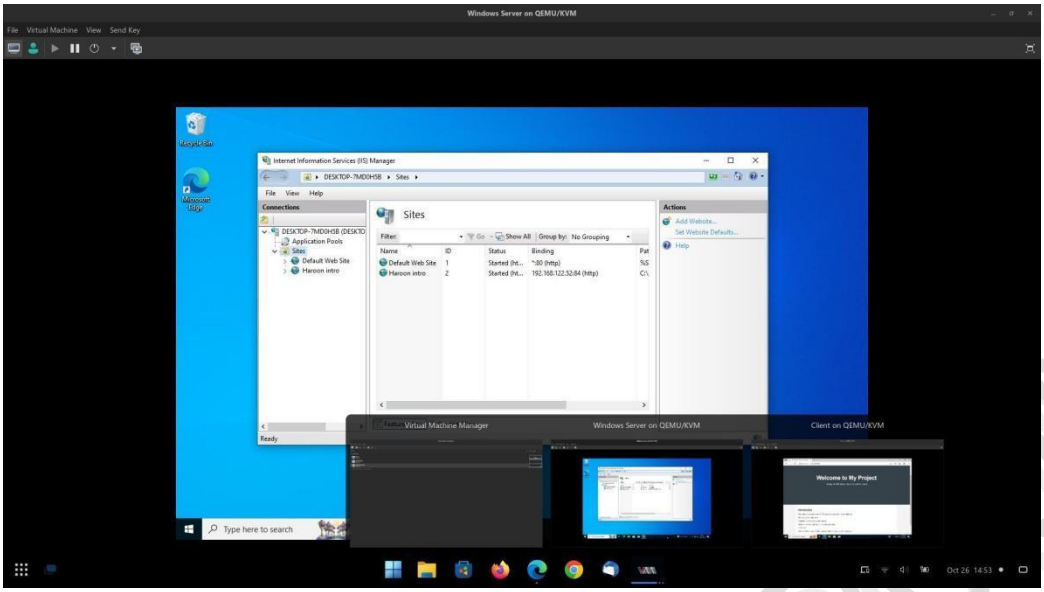


Next I had to ping them together for that I went to my Server Command Prompt and typed `ping 192.168.122.52` and similarly in my Client VM's Command Prompt typed `ping 192.168.122.53`. This was how I permanently connected my VM's together to **two-way Ping connection** for hosting of websites. The commands are also same for ping in Ubuntu.

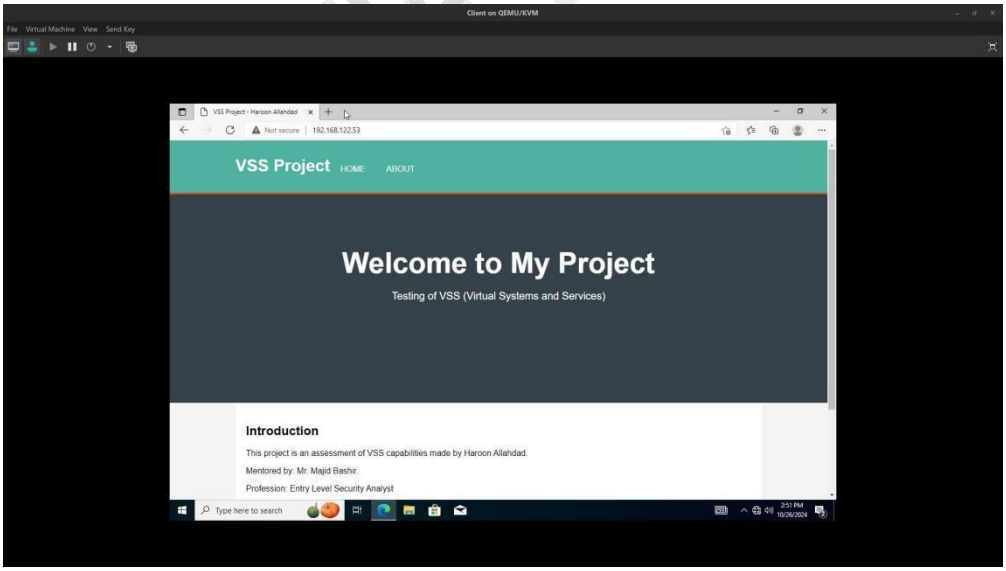


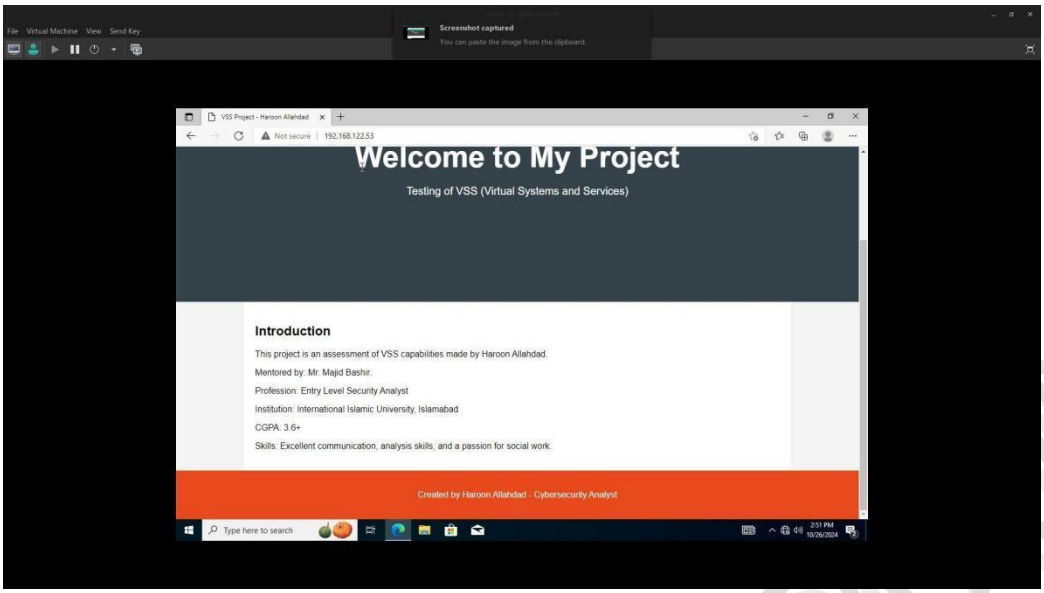


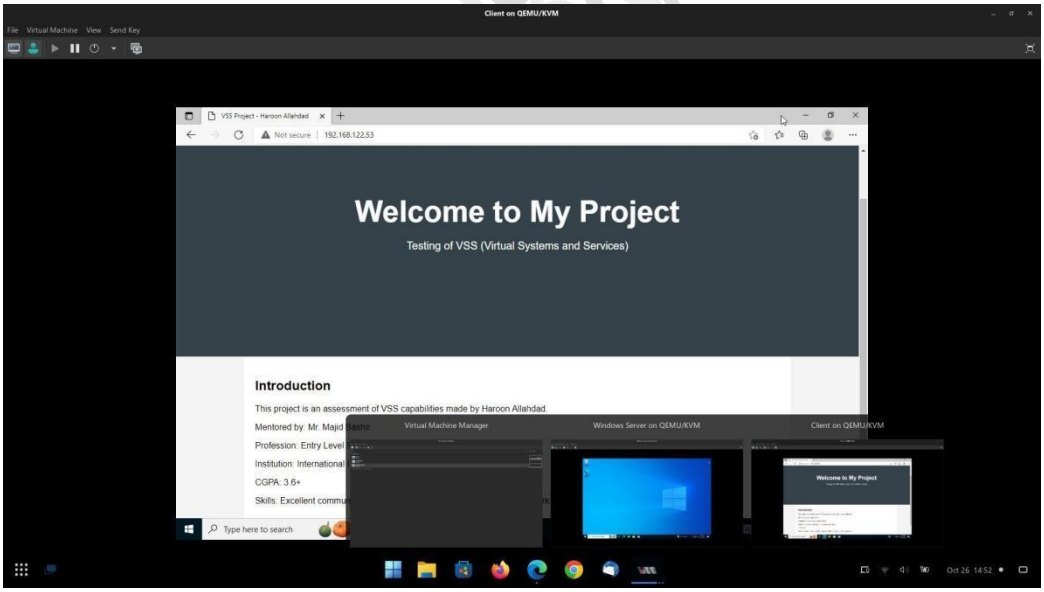
6) I have created a file on the Server VM in the following path: C:\inetpub\wwwroot and named it index and changed it's file type to .html from .txt and then copied my sample website's code to it. Next I went to the IIS and clicked on Add new Website in the Sites option on the left menu panel of the IIS of Server VM and named it Haroon intro and assigned it port 84. It was showing some, error's so I did a bit of settings in the Directory Browsing and Error Pages. Also I set the Default Document for easy access.



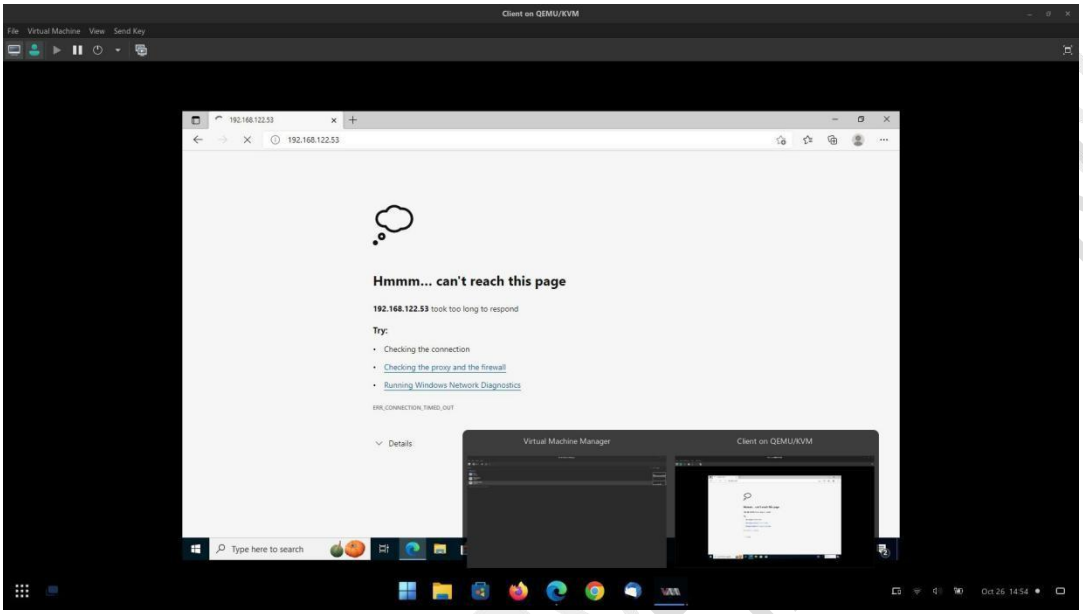
7) After that I confirmed the website by accessing it through my client VM, and it was showing it against the Server VM's IP address (192.168.130.53). Here are the pictures as proof.




Screenshot captured
You can paste the image from the clipboard.
VSS Project - Haroon Allahdad
Not secure | 192.168.122.53
Welcome to My Project
Testing of VSS (Virtual Systems and Services)
Introduction
This project is an assessment of VSS capabilities made by Haroon Allahdad.
Mentored by: Mr. Majid Bashir.
Profession: Entry Level Security Analyst
Institution: International Islamic University, Islamabad
CGPA: 3.6+
Skills: Excellent communication, analysis skills, and a passion for social work.
Created by Haroon Allahdad - Cybersecurity Analyst
Type here to search


Client on QEMU/KVM
VSS Project - Haroon Allahdad
Not secure | 192.168.122.53
Welcome to My Project
Testing of VSS (Virtual Systems and Services)
Introduction
This project is an assessment of VSS capabilities made by Haroon Allahdad.
Virtual Machine Manager
Windows Server on QEMU/KVM
Client on QEMU/KVM
CGPA: 3.6+
Type here to search

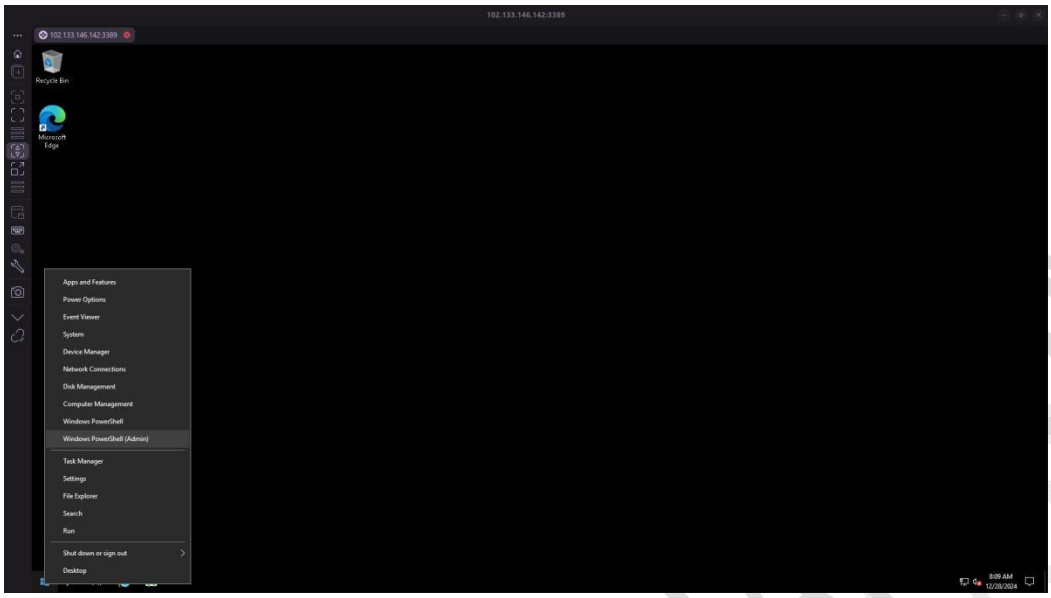
8) To further solidify my claim, I closed my Server VM, and then again opened the same link (192.168.122.53), but it returned the can't reach the page error, which showed that the Server VM



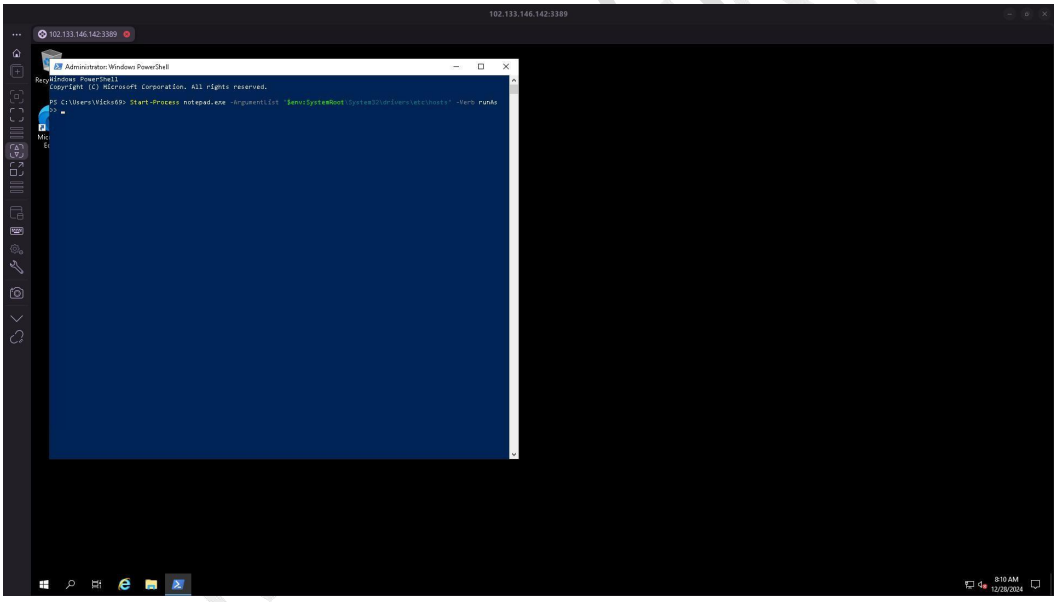
was hosting my website.

9) If we want to view our website through our host OS the same is the process we just have to use this command: `ifconfig` (in my case of Ubuntu/ Zorin based OS). Then again use same ping command with ip address of the server device.

10)Now lastly if we want to access our websites through a website name instead of ip then we can simply do this in the windows server:

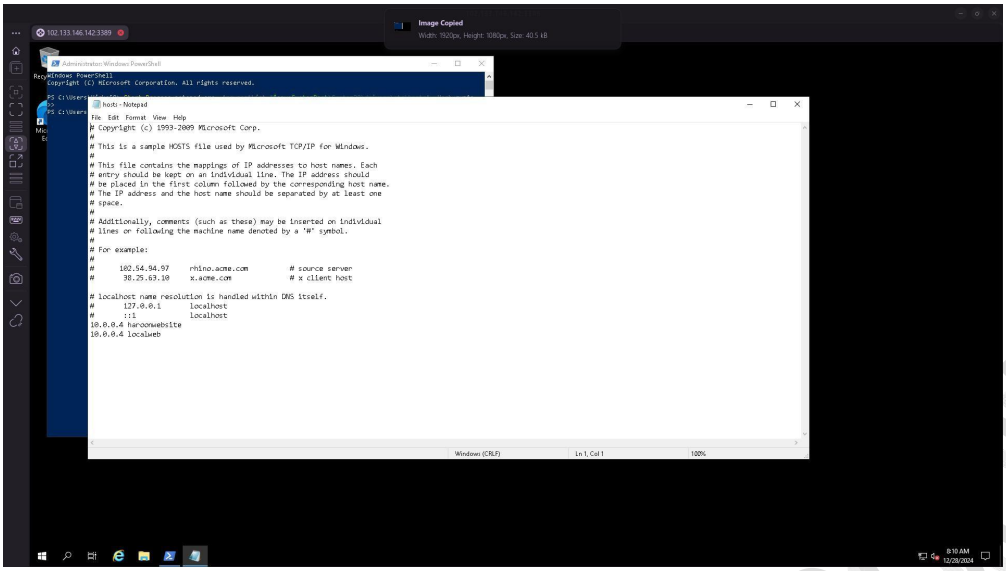


Then type this command in the powershell (admin)



Command: `Start-Process notepad.exe -ArgumentList "$env:SystemRoot\\System32\\drivers\\etc\\hosts" -Verb runAs`

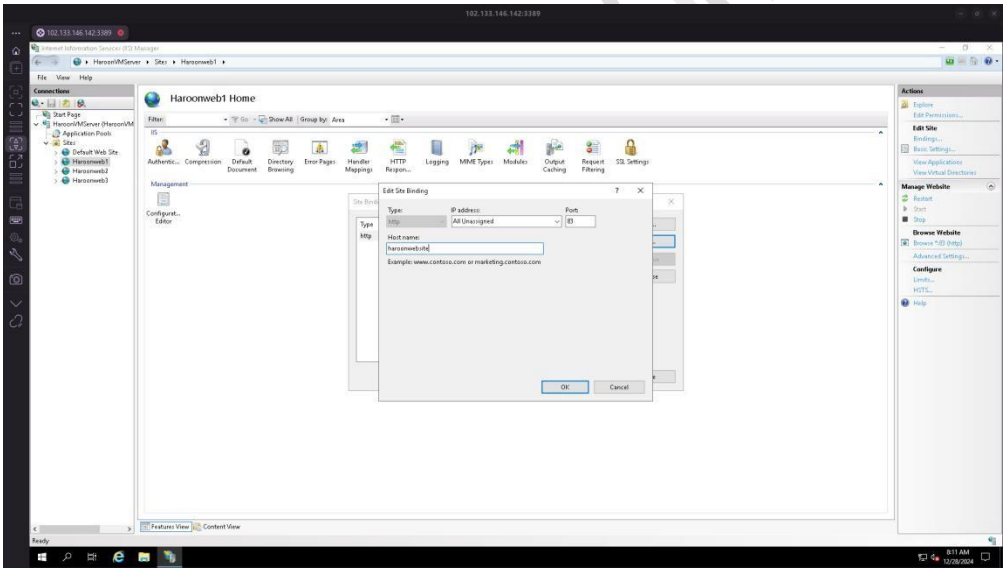
Then hosts file would open up, here you would have to add your server's ip address through which you are able to view your website with a **space** and the website's name after it.



Like 10.0.0.4 haroonwebsite

Then go to the server VM's IIS and then click on the Site which you want to access through it's name, then you would see a button at the right pane named Binding, click on it and edit the bindings.

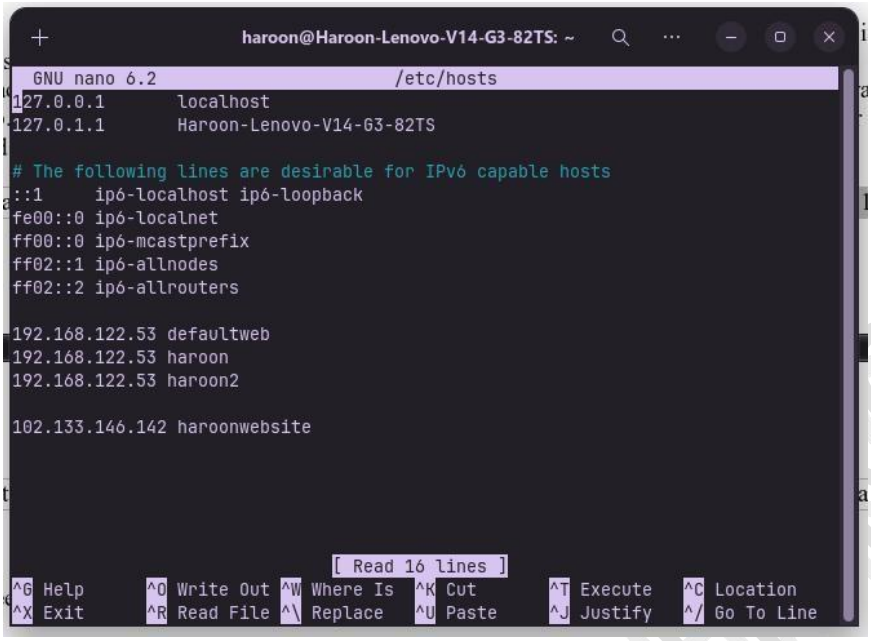
Then in the host name enter the website name you entered same as in the hosts file.



11) If the step 10 was too hard for you then you can instead perform only this step, as a shortcut, but even if you followed step 10 then too you must folllow this step and go to your client VM or Host Machine or whichever machine you want to access thiswebsite from, remember that the device must be pinged two-way befor this step. Then for windows go to the powershell (admin) and follow the same instructions and enter the ip address and website name as in step 10. In case of Ubuntu/ Zorin I typed this command:

```
sudo nano /etc/hosts
```

Then this screen opened:



Here I have entered some of my ip addresses which can be accessed through names. This same concept is used in DNS now-a-days in Internet termonology.

Here our Major Task 1 comes to a final ending.

Acronyms and Abbreviations:

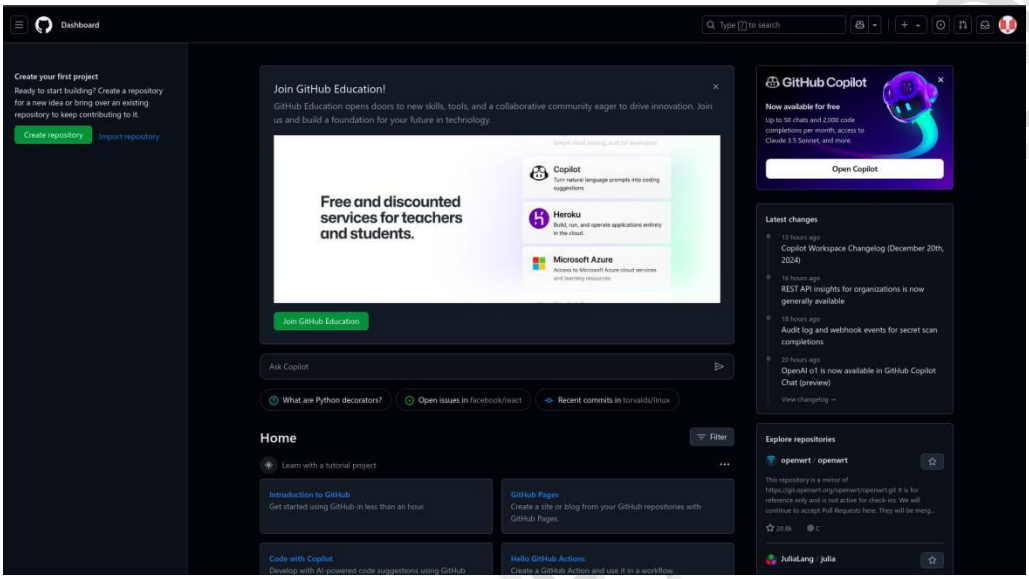
VM:	Virtual Machine
VMM:	Virtual Machine Manager
OS:	Operating System
IP:	Internet Protocol
IIS:	Internet Information Services
VSS:	Virtual Systems & Services

Major Task 2: Create an Azure Account and perform the same tasks as step 1 and make website available for viewers online.

Note: To start with I must tell you that I don't have a bank account, otherwise you can simply go to Microsoft Azure and signup with your credit card details and get a free 200$ trail version credit for this task. So I used the second approach which I would advise you to follow through too.

1) Create a GitHub Account with your university mail.

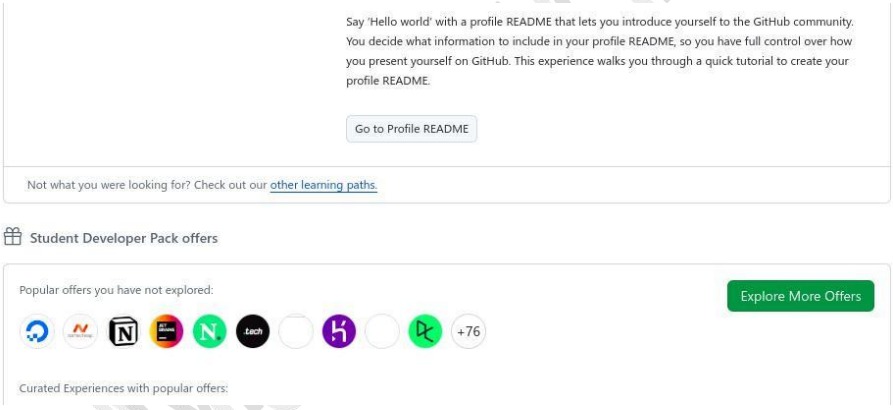
2) Then I clicked on this link to get GitHub student pack.



3) Then you will get a form where you must enter the location of your university, select your university, add your card picture and add your transcript (permission form) image. You can change size/ quality of image with online size/ pixel converters online.

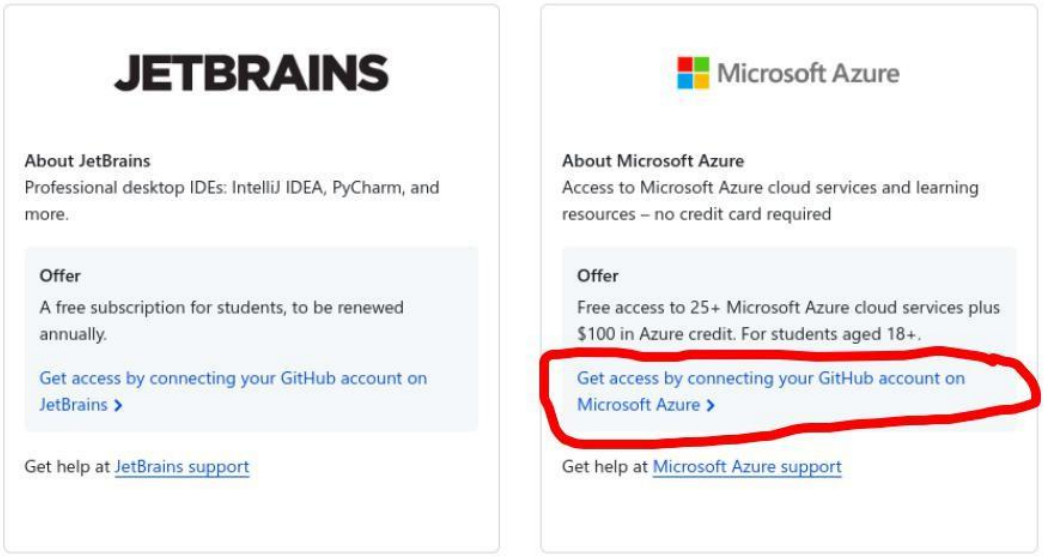
4) Once you submit this form correctly, you will get it's confirmation email from 1 to 4 days later.

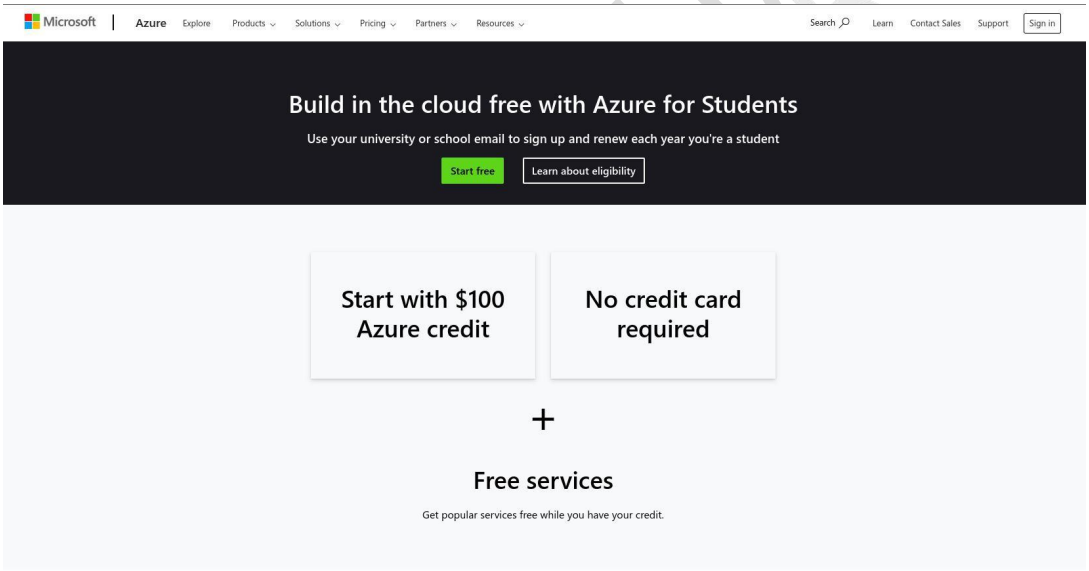
Here instead of the black line you will get a link, click on that link. And it will take you to the github student pack then you shall scroll down and get this screen:



Click on Explore More Offers.

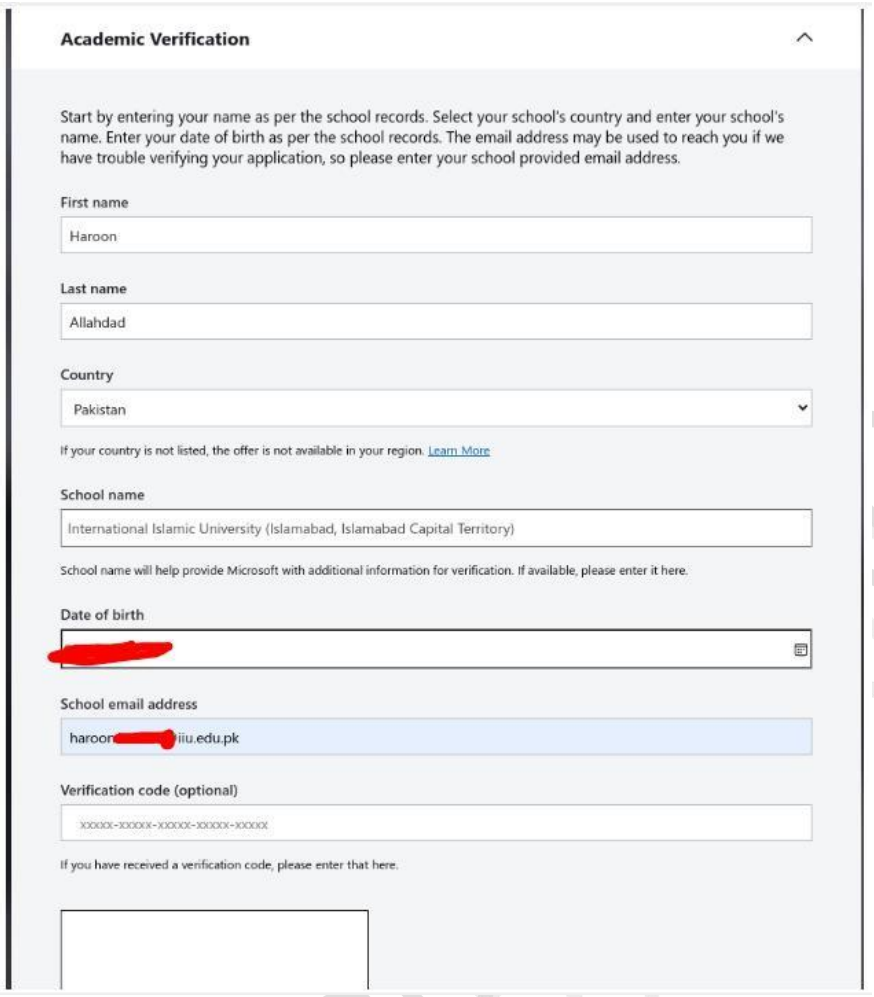
4) Then you will further scroll down and in the all offers section you will find this Microsoft Azure Offer for GitHub Student Pack:



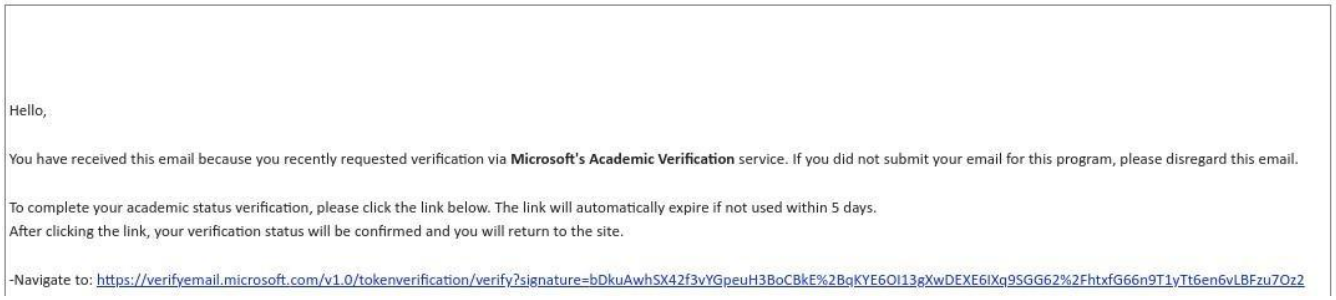


Click on Start free.

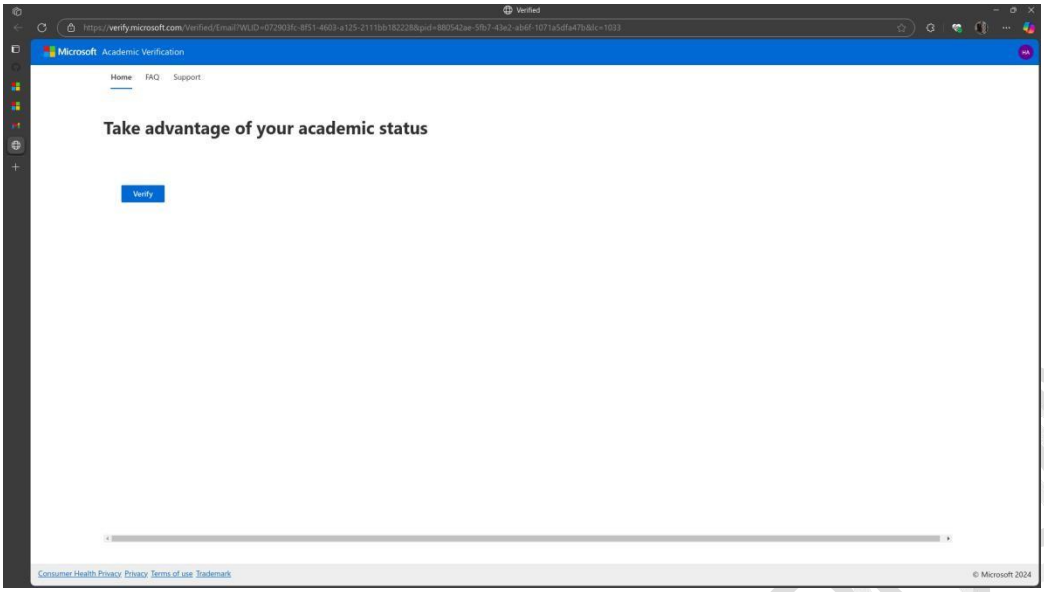
5) Then a form would showup where you must enter your info in this form:

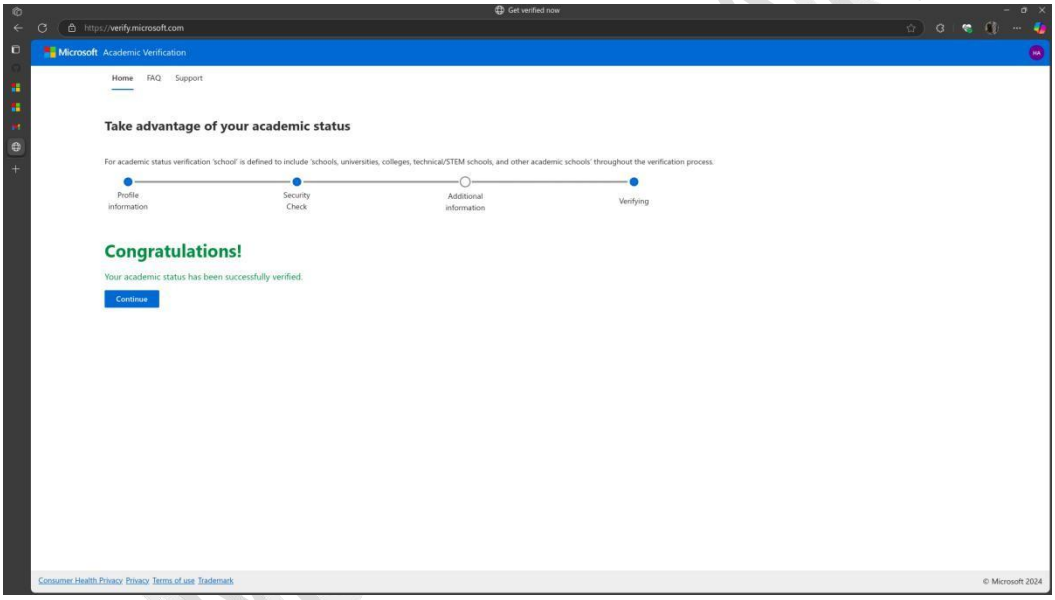


Once done with that a verification code would be sent to your university email.

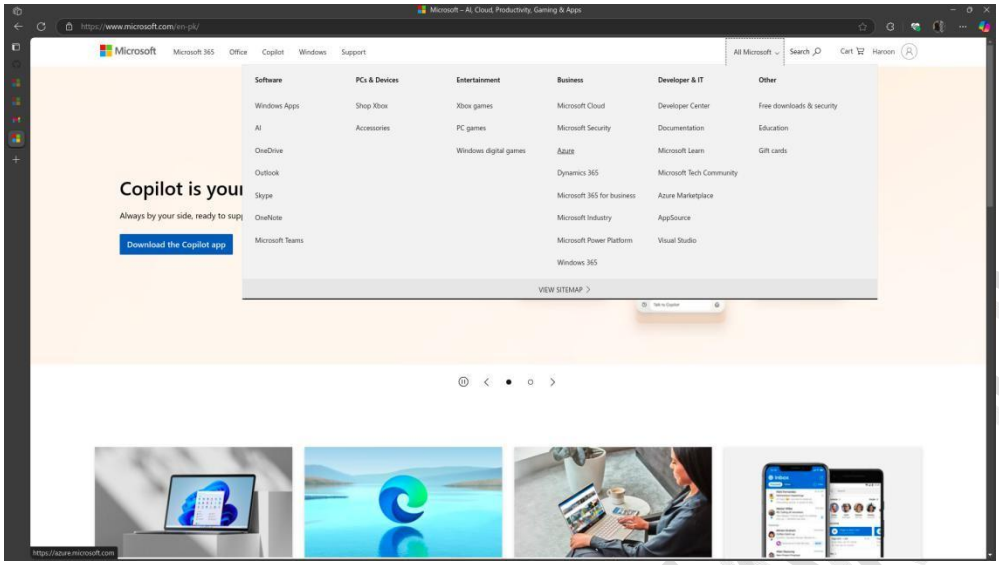


Click on the link at the bottom of this picture.


Verified
Microsoft Academic Verification
Home FAQ Support
Take advantage of your academic status
Verify
Consumer Health Privacy Privacy Terms of use Trademark
© Microsoft 2024

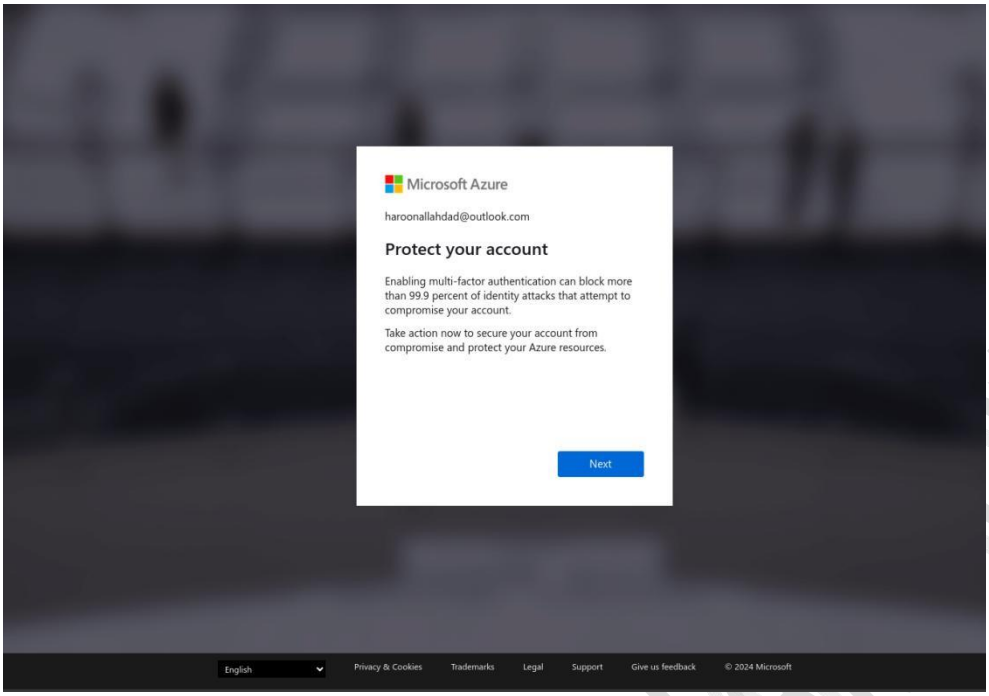

Get verified now
https://verify.microsoft.com
Microsoft Academic Verification
Home FAQ Support
Take advantage of your academic status
For academic status verification 'school' is defined to include 'schools, universities, colleges, technical/STEM schools, and other academic schools' throughout the verification process.
Profile information
Security Check
Additional information
Verifying
Congratulations!
Your academic status has been successfully verified.
Continue
Consumer Health Privacy Privacy Terms of use Trademark
© Microsoft 2024

6) Then you will be lead to this screen:



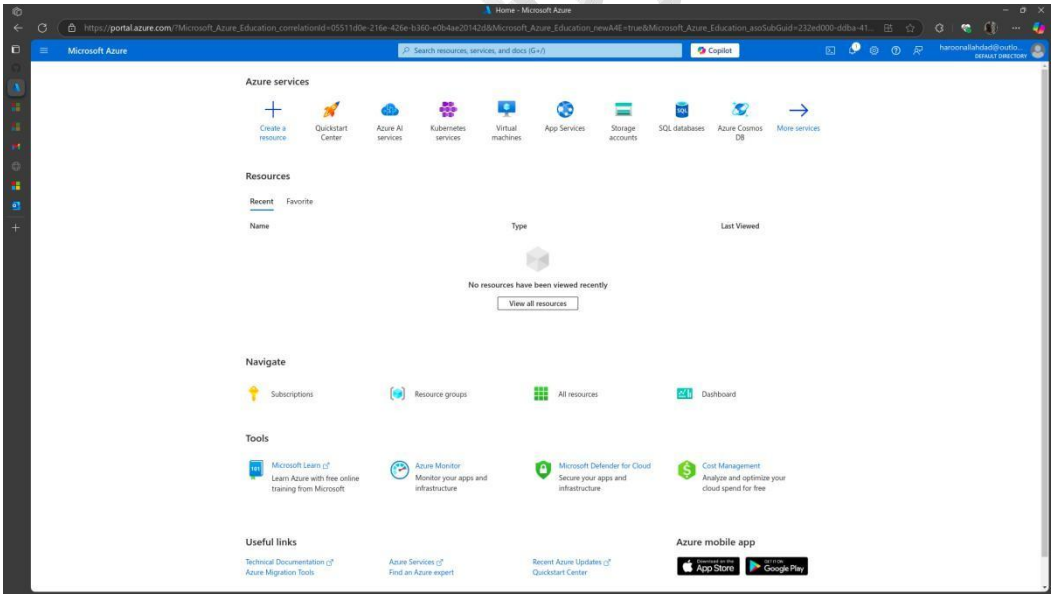
Once on this screen hover over the All Microsoft Button and a menu like this would appear and then click on Azure.

7) Now once again a form would show now look at the right side of the form if 200$ free credit is written there then go to the first screen again which said start free and then you will directly be taken to the students form again and if not then simple add your work email (Other email then your university and your phone number in that form which doesn't have +92 in it meaning something like this: 322-------). Now click on submit form, and if this form works and it goes on and shows you this screen:

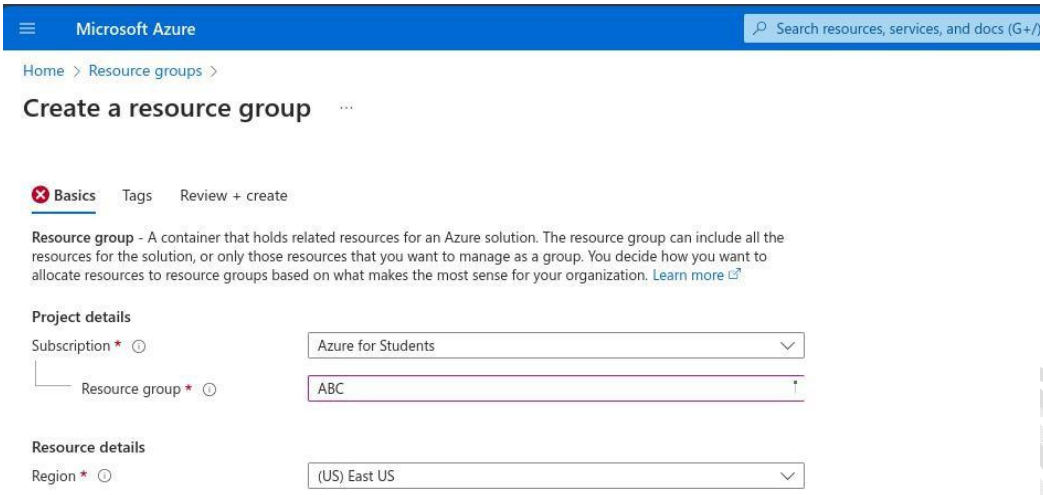


Then congrats! Your Account has been created on MS Azure.

8) Now you must come across a screen like this:



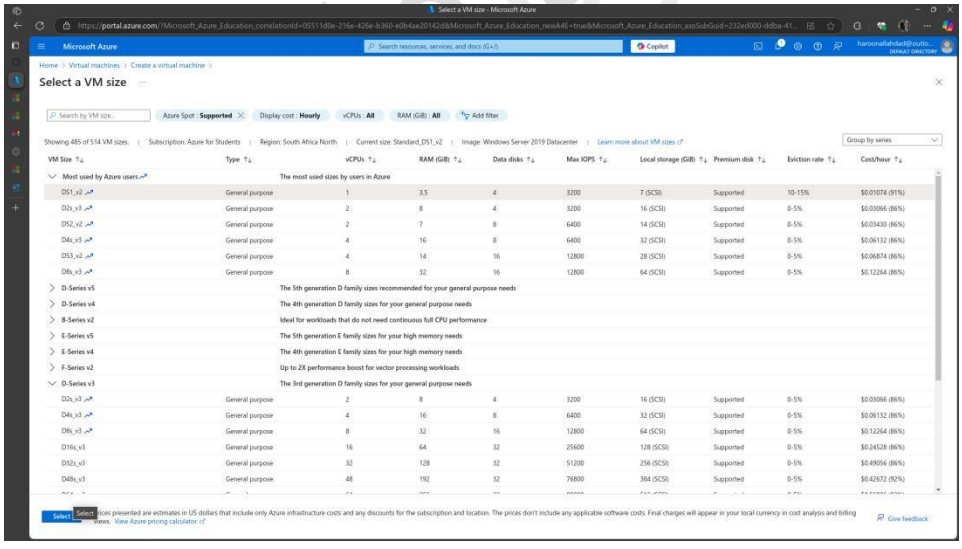
9) Click on Resource Groups and then click on create and enter any name you want.



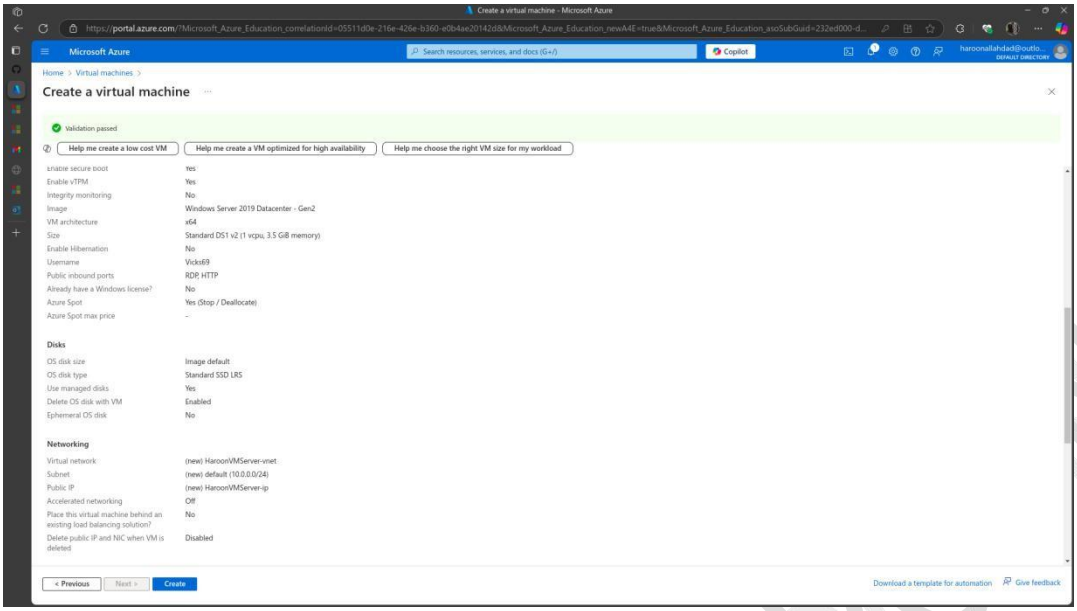
Then click on create, remember to have the Subscription at Azure for Students.

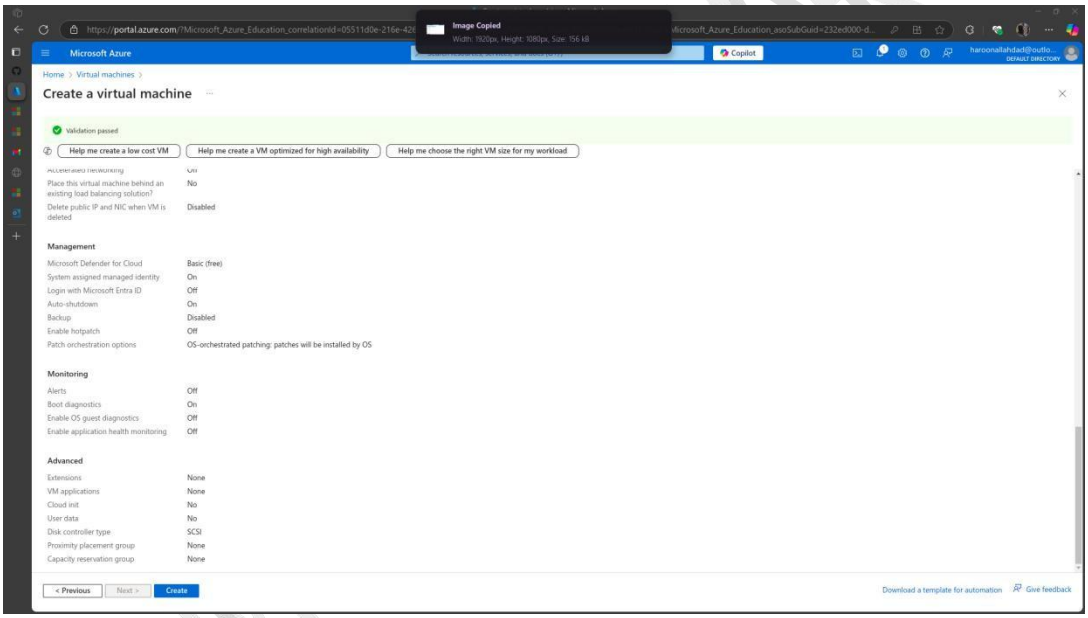
10) Again go to Microsoft Azure button at top left corner and go to the home and then click on Virtual Machines, then click on Create, Create Azure Virtual Machine.

11) Then click on the Resource group and select the one you created before, next go to size and click on more sizes and select the most basic one which suits your resources.



12) Select your username, password and in the Select inbound ports, select HTTP along with RDP.

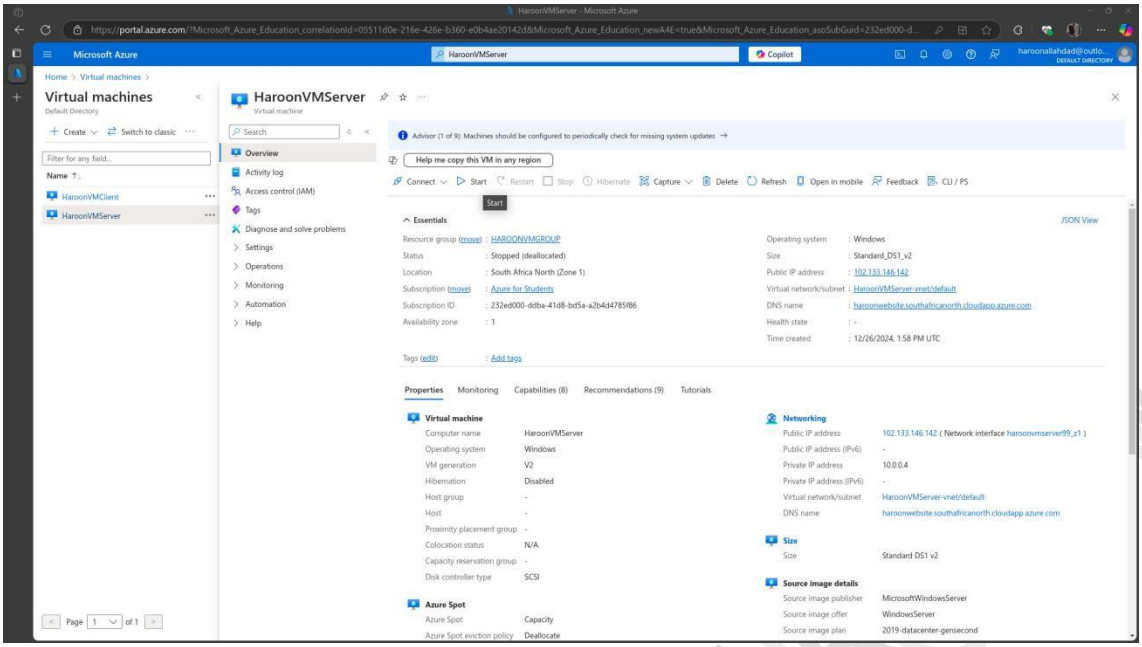




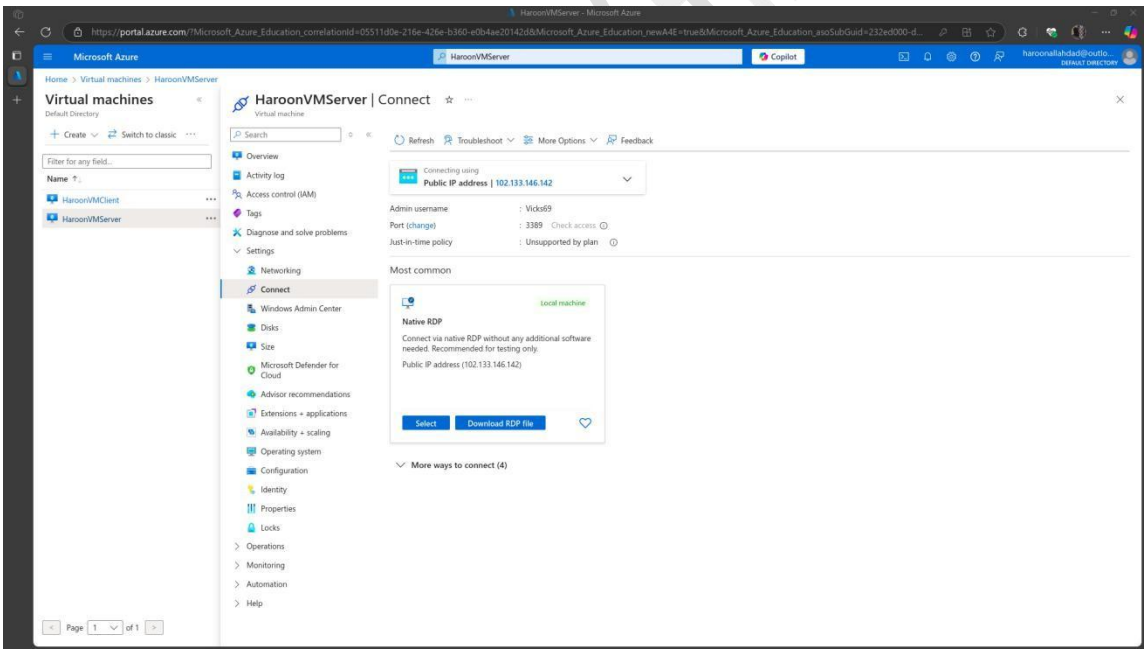
Then click on create.

14) Similarly create a Clint VM.

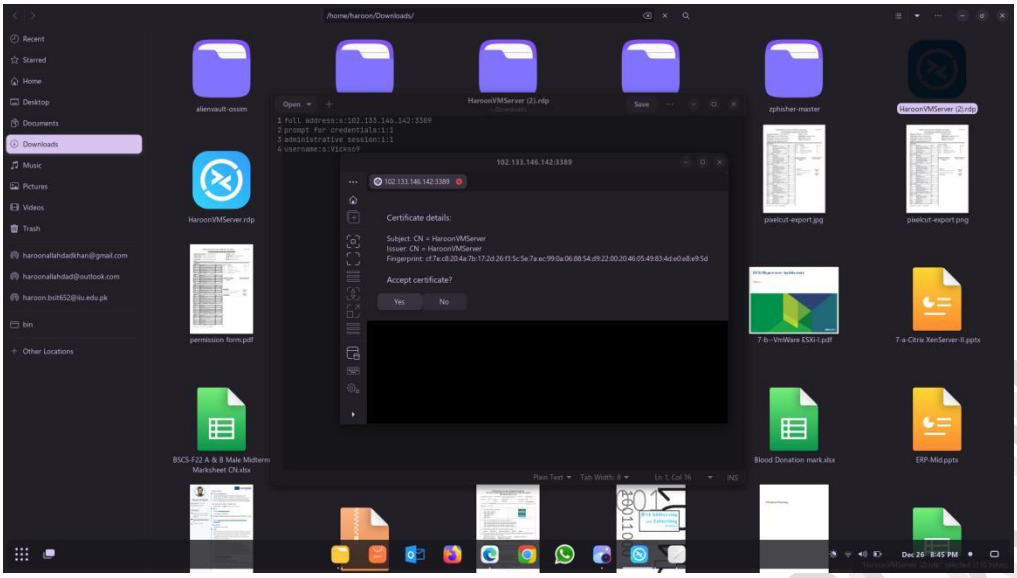
15) Now again go to Virtual Machines and select Server VM, and click on start:



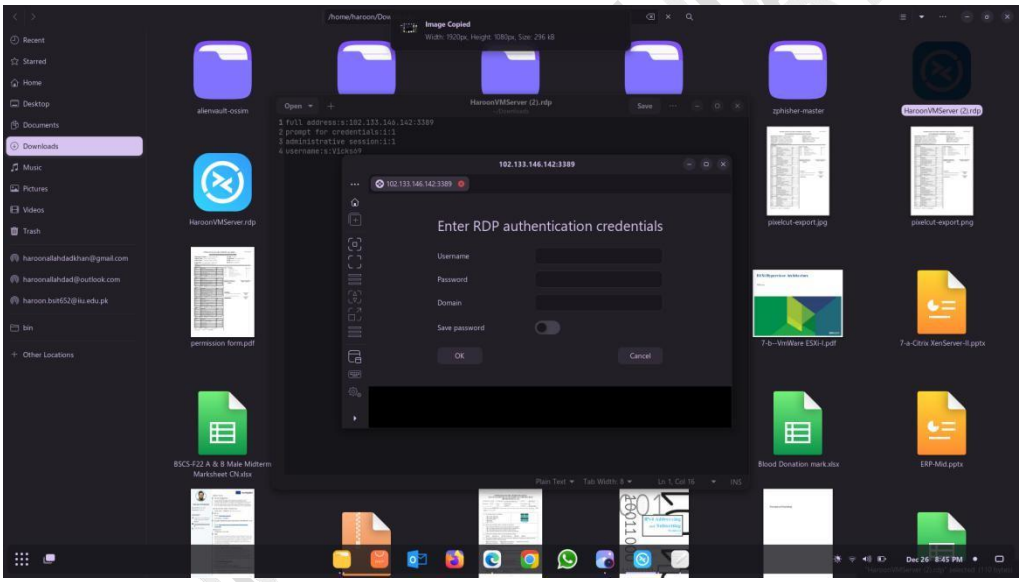
16) Then click on Connect > connect. And then you will see this page:



Download this RDP file and then click on it and it will open, but in case of Ubuntu or Zorin Open Remmina Application and open this downloaded file in text editor. Next copy the full address and paste here on Remmina App:



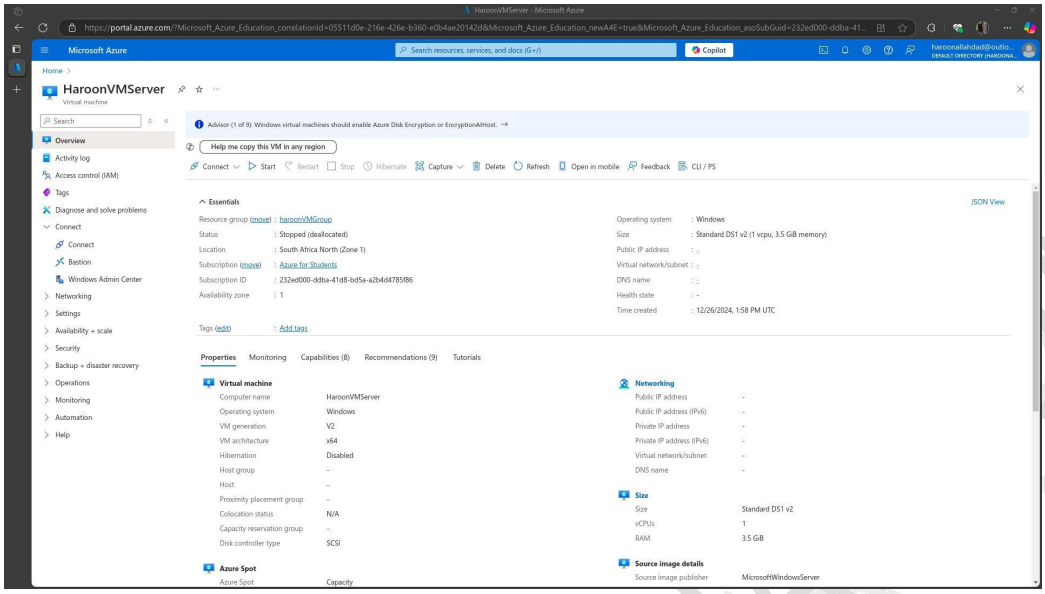
Next Click Yes and enter your username and password here:



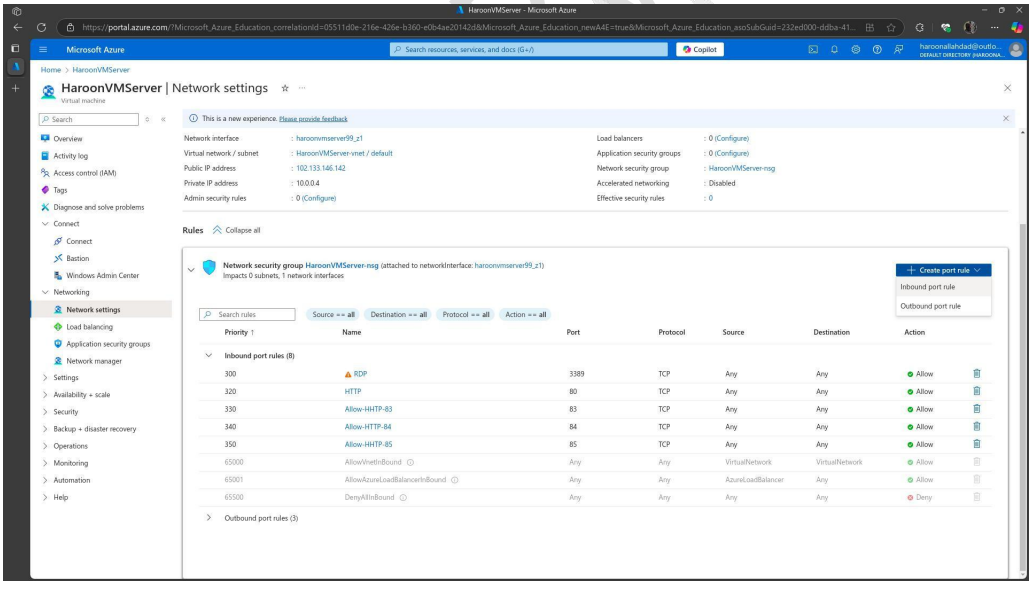
Further on you can open your client VM the same way and then follow the same steps as earlier to Create and host a website through IIS on your server and then access it through your Client Machine too.

17) Now you can even access these websites through your mobile or other devices without pinging with the Azure services, you just have to turn the Server VM on.

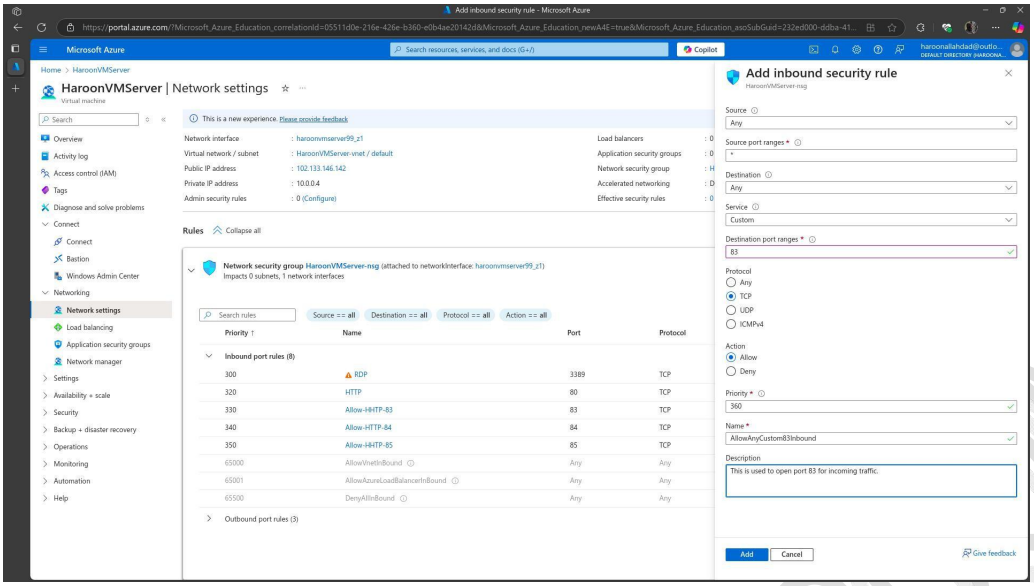
18) For that you must go to the Microsoft Azure Home placed on the top left and then click on the Virtual Machines options, next turn on the Server VM and Click on the Server VM's Name to go to details. Then click on the Networking heading for details.



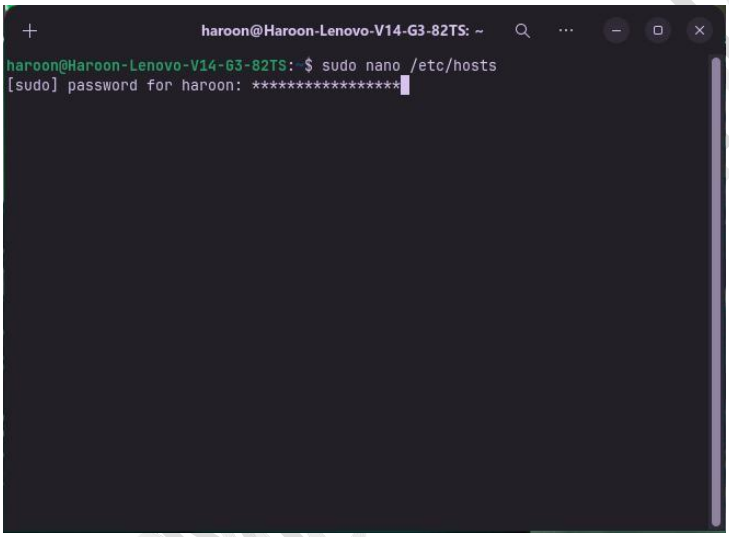
19) Then scroll down and you would see the rules panel, in it click on the add new rules option and then click on inbound rules:

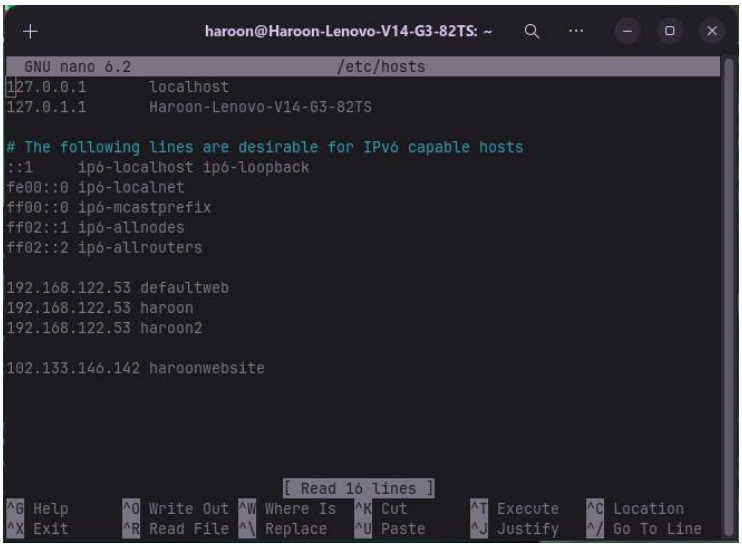


20) Then add the following details as in the screenshot, changing the port number to the one through which you want your audience to view your server hosted website. In my case it is port 83, 84, 85 and port 80 which is default was already enabled.



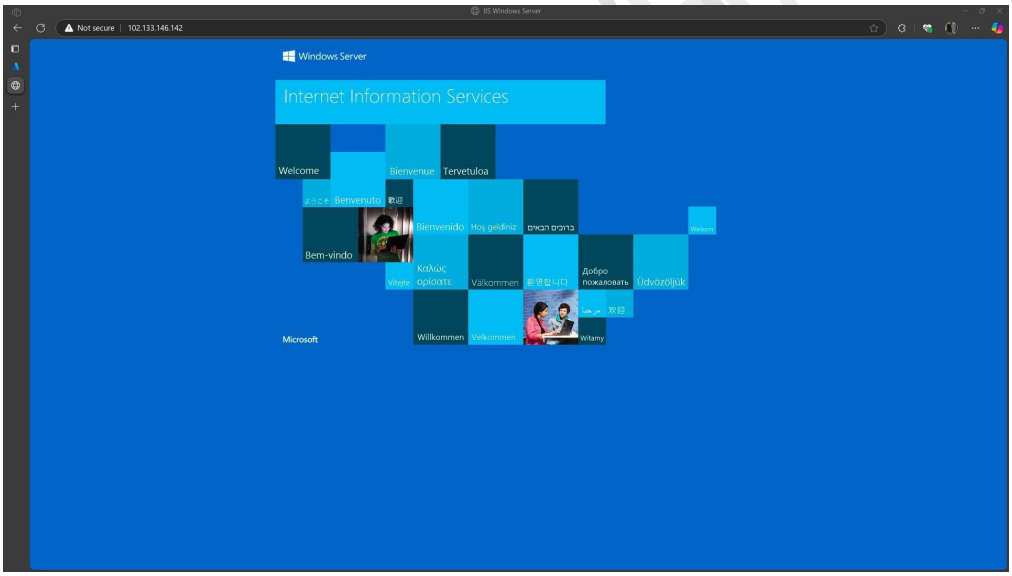
21) Then again if you want to view your website on any device through webname instead of an IP, you must again follow the steps to input your IP and webname in any host device's hosts file:

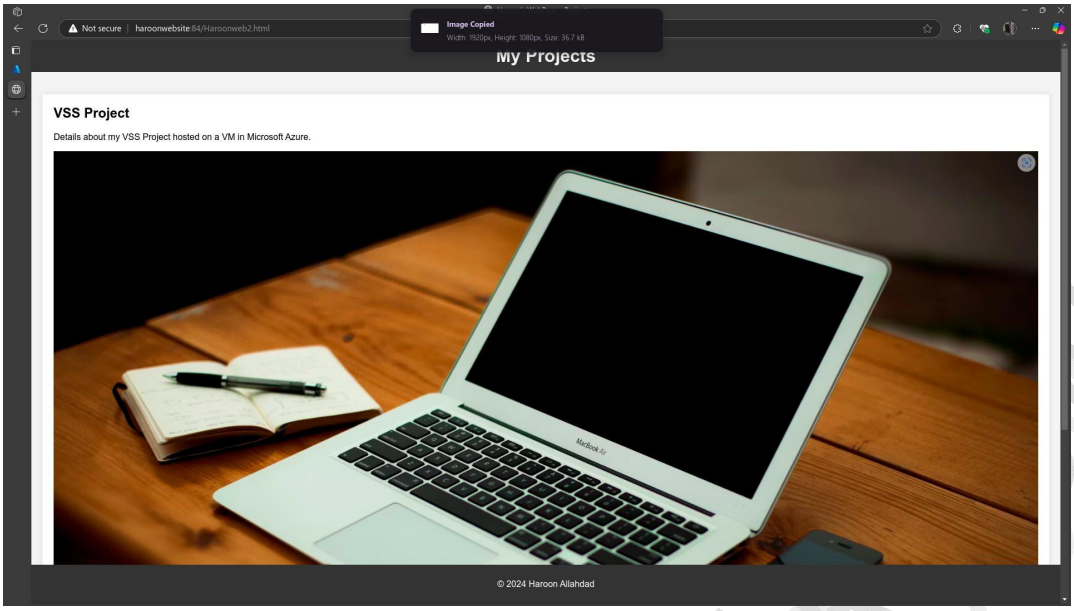




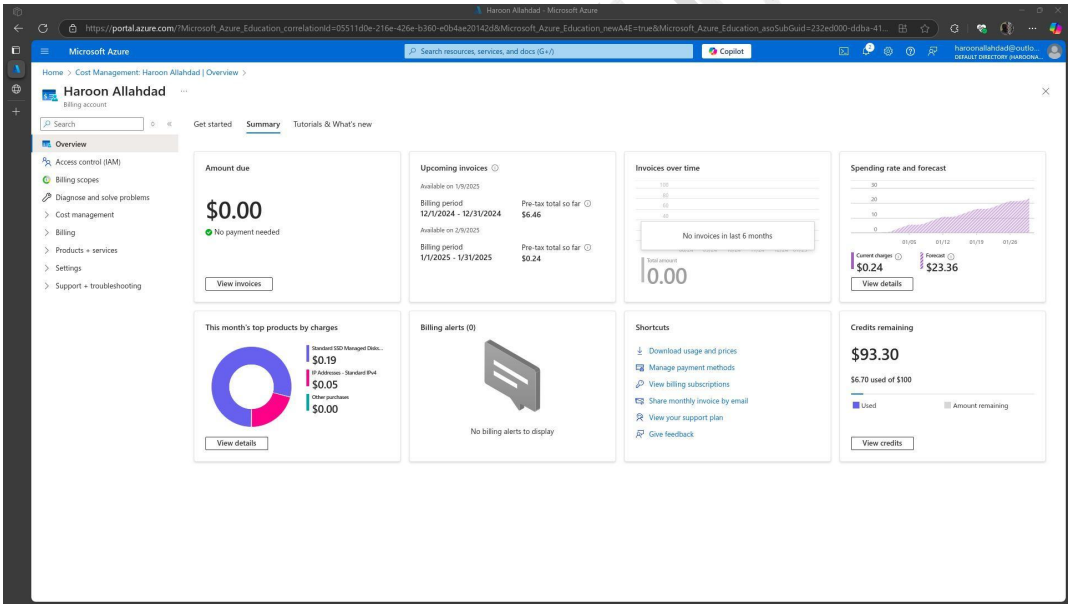
22) Next you can now access your website through the IP from any device, or through the webname from the devices your IP and webname is stored in hosts file.

Here are some photo's (My localhost website is on Port 80, And other three websites is on Port 83, 84 and 85 and my public IP is 102.133.146.142)





23) Now even after performing all the works I only spent roundabout 0.70$ in credit, but after extensive testing and checking things out, I still had this much credit left:



Your Major Task 2 is also Done after this! Congrats!

For Further details on these steps if you are facing issues related to them or having troubleshooting problems, ask for assistance from Gemini or Copilot as they may assist you better than any video or web tutorials or you may also mail me directly on haroonallahdad@outlook.com.