

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS

COMPUTER NETWORKING

Paola Andrea Cuellar Beltran

Workshop No. 2 – Sockets and Services

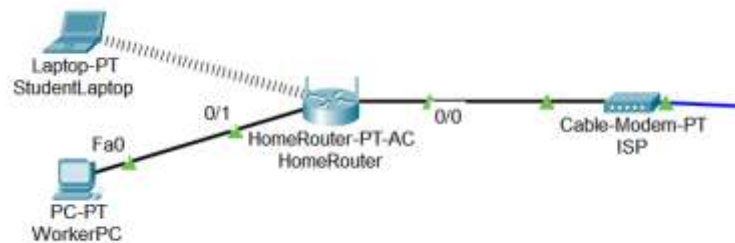
Eng. Carlos Andrés Sierra, M. Sc.

Computer Engineering

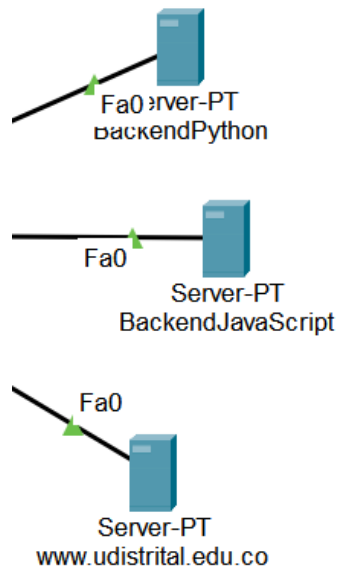
2025

For this workshop, I used Workshop 1 as a basis. I followed the step-by-step instructions as mentioned in the workshop and found myself in trouble many times. To explain things in a more organized way, I will go step by step:

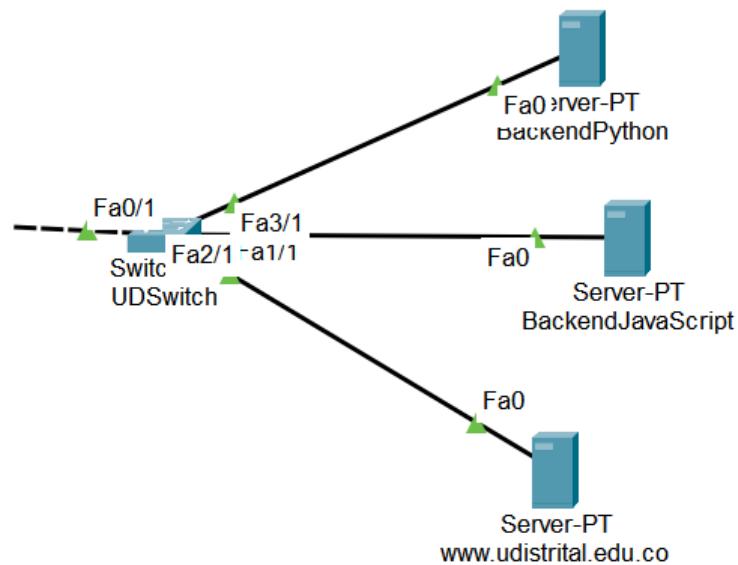
1. I used the StudentLaptop, the WorkerPC, the ISP, and the Internet again.



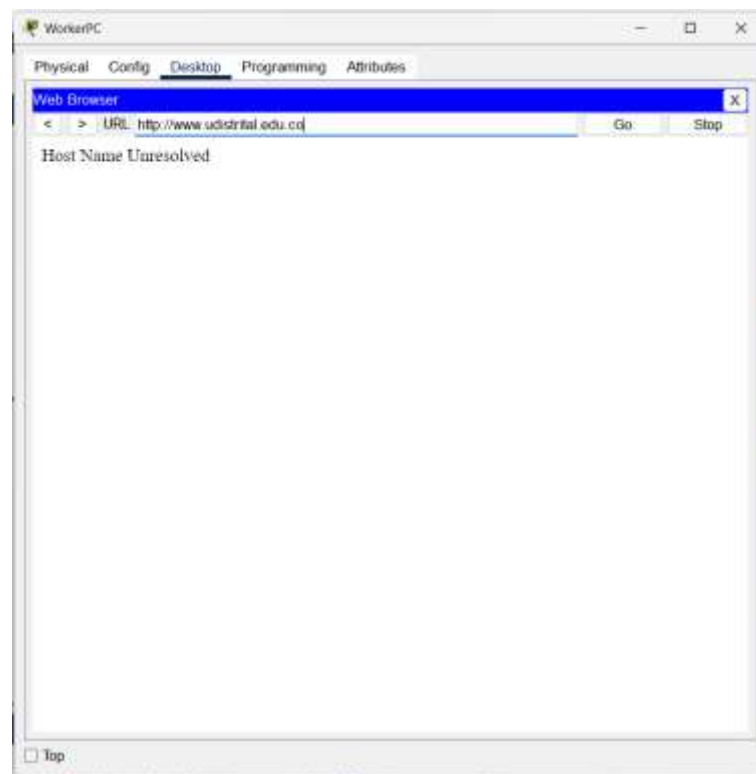
2. I added and configured the BackendPython, BackendJavaScript, and the FrontendServer or www.udistrital.edu.co.



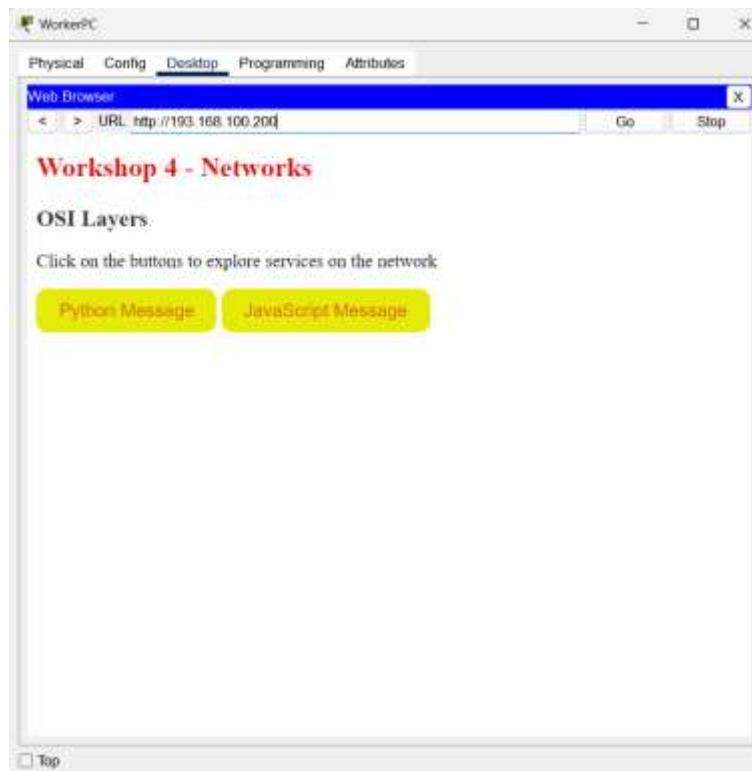
3. To establish the connection between the servers, I used a switch, which I named UDSwitch.



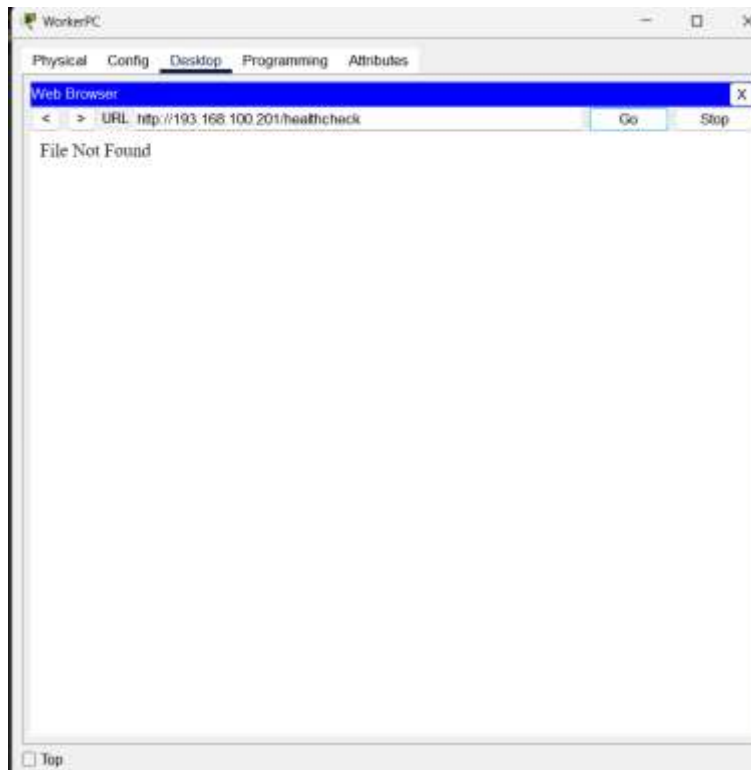
4. One of the initial problems I faced was that the waiting time was too long. So, I decided to set a checkpoint and work from there, always verifying that everything continued to function properly. One of the problems I encountered was the following:



5. And I was able to solve it as follows:



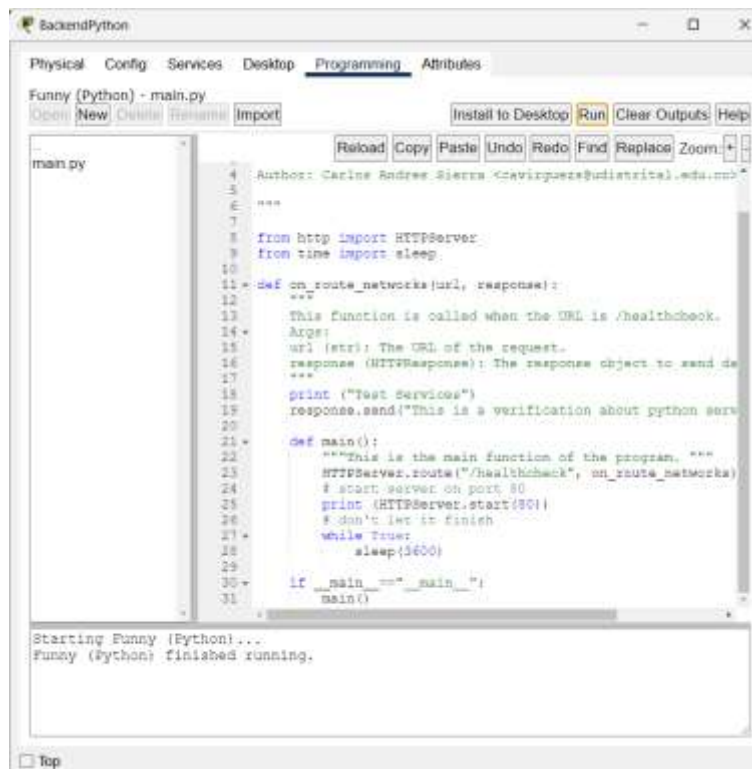
6. For some reason, the buttons are not working, and when I search for the URL, I get the following message:



7. For some reason, the buttons are not working, and when I search for the URL, I get the following message:



7. And at least the Python code is working correctly.



The screenshot shows the BackendPython IDE interface. The main window displays a Python script named `main.py` with the following code:

```
4 Author: Carlos Andres Sierra <cavirpaz@udianrital.edu.ec>
5
6 """
7
8 from http import HTTPServer
9 from time import sleep
10
11 def on_route_networks(url, response):
12     """
13     This function is called when the URL is /healthcheck.
14     Args:
15     url (str): The URL of the request.
16     response (HTTPResponse): The response object to send de
17     """
18     print ("Test Services")
19     response.send("This is a verification about python serv
20
21 def main():
22     """This is the main function of the program. """
23     HTTPServer.route("/healthcheck", on_route_networks)
24     # start server on port 80
25     print (HTTPServer.start(80))
26     # don't let it finish
27     while True:
28         sleep(3600)
29
30 if __name__ == "__main__":
31     main()
```

The output window at the bottom shows the execution results:

```
Starting Funny (Python)...
Funny (Python) finished running.
```

The interface includes a menu bar with options like Physical, Config, Services, Desktop, Programming, and Attributes. Below the menu is a toolbar with buttons for Open, New, Delete, Rename, Import, Install to Desktop, Run, Clear Outputs, and Help. A secondary toolbar contains Reload, Copy, Paste, Undo, Redo, Find, Replace, and Zoom. The status bar at the bottom left has a "Top" button.