



Faculty of Engineering & Technology
Electrical & Computer Engineering Department

ENCS 3320 - Computer Networks

Report 1 – Project 1
“Socket Programming”

Prepared by:

Tala Dweikat 1191590

Rania Shahwan 1190585

Mazen Batrawi 1190102

Instructor: Dr. Muhammad Helal

Section: 2

Date: 30/4/2022

Abstract

This project consists of two parts. The first part is to run some commands on the terminal; ping a device in the same network, ping YouTube website, tracert and nslookup it. The second part is to implement a simple but a complete web server in python that is listening on port 900. Both html and css are used for the design of our website.

Table of Contents

Part1: Running Commands.....	4
1- Ping a device in the same network.....	4
2- Ping www.youtube.com	5
3- Traceroute www.youtube.com.....	6
4- nslookup www.youtube.com	7
Part2: Creating a web server.....	8
1- First page English version website.....	8
2- Second page English version website.....	11
3- First and second pages of Arabic version website	16
5- Request ends with .html.....	18
6- Request ends with .css	19
7- jpg request.....	20
8- png request.....	21
9- 307 Temporary redirect.....	22
10- Error page.....	23
11- Testing from another device	25
Appendix	26

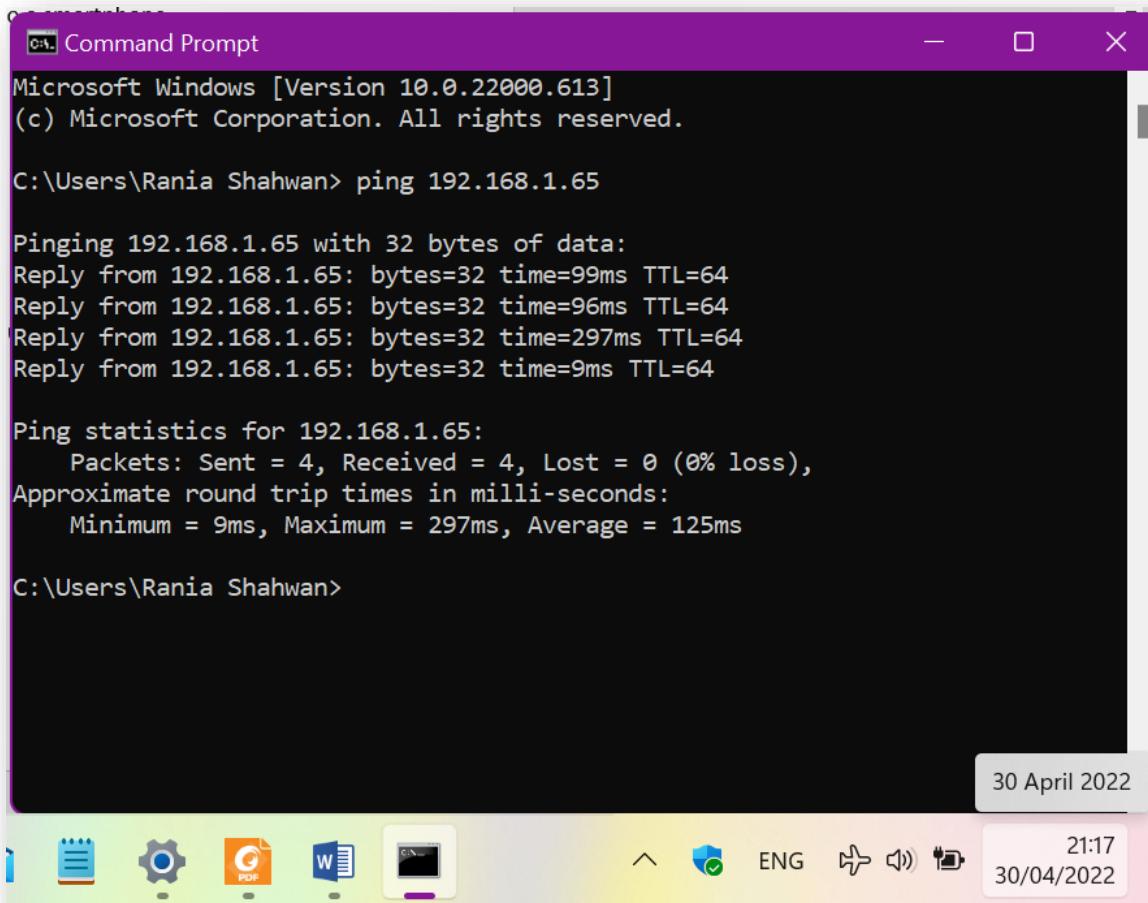
Table of Figures

Figure 1: Ping a host	4
Figure 2: Ping a website (yoututbe)	5
Figure 3: Traceroute command.....	6
Figure 4: Nslookup Command.....	7
Figure 5: Page 1 English version request	8
Figure 6: Page 1 English version website	8
Figure 7: Main_en.html code.....	9
Figure 8: Design.css code.....	10
Figure 9: Page 2 English version request	11
Figure 10: Page 2 English version website.....	11
Figure 11: Main2_en.html code.....	12
Figure 12: Design2.css code.....	13
Figure 13: Page 1 Arabic version request	14
Figure 14: Page 1 Arabic version website	14
Figure 15: Main_ar.html code	15
Figure 16: Page 2 Arabic version website	16
Figure 17: Main2_ar.html code	16
Figure 18: Design2_ar.css code	17
Figure 19: A .html request.....	18
Figure 20: Response of a .html request	18
Figure 21: A .css request	19
Figure 22: Response of a .css request	19
Figure 23: A .jpg request.....	20
Figure 24: Response of a .jpg request	20
Figure 25: A .png request	21
Figure 26: Response of a .png request	21
Figure 27: Requests of a 307 temporary redirect to CNN website.....	22
Figure 28: CNN Redirect	22
Figure 29: Code of error page.....	23
Figure 30: Error page	24
Figure 31: Testing project from a phone	25

Part1: Running Commands

1- Ping a device in the same network.

Ping is a command that can be used to troubleshoot network issues, such as network connectivity. The Ping utility uses the echo request, and echo reply messages within the Internet Control Message Protocol (ICMP). It sends 4 requests by default in Windows, it also contains information such as the bytes sent and received, time to live (TTL), how long the response took to receive, and statistics about packet loss and round trip times. Figure below shows pinging a host with received response from 192.168.1.65 with some information, different delays but an average round trip times of 125ms.



The screenshot shows a Microsoft Windows Command Prompt window titled "Command Prompt". The window displays the following text output:

```
Microsoft Windows [Version 10.0.22000.613]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Rania Shahwan> ping 192.168.1.65

Pinging 192.168.1.65 with 32 bytes of data:
Reply from 192.168.1.65: bytes=32 time=99ms TTL=64
Reply from 192.168.1.65: bytes=32 time=96ms TTL=64
Reply from 192.168.1.65: bytes=32 time=297ms TTL=64
Reply from 192.168.1.65: bytes=32 time=9ms TTL=64

Ping statistics for 192.168.1.65:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
        Minimum = 9ms, Maximum = 297ms, Average = 125ms

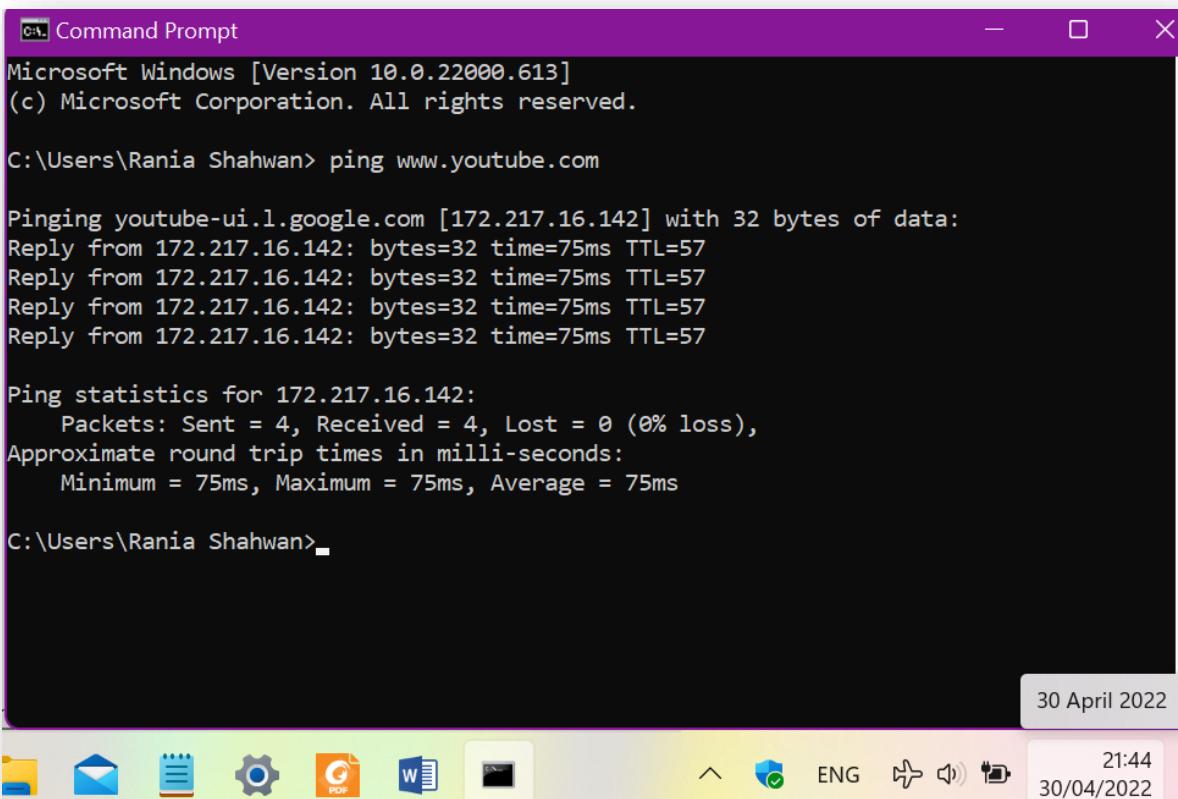
C:\Users\Rania Shahwan>
```

The taskbar at the bottom of the screen shows several pinned icons: OneDrive, Settings, Google PDF, Microsoft Word, and Microsoft Edge. The system tray icons include a battery icon, a signal strength icon, and a language switcher showing "ENG". The date and time are displayed as "30 April 2022" and "21:17" respectively.

Figure 1: Ping a host

2- Ping www.youtube.com

Pinging a website is actually pinging the domain name and not its' IP address. Because also name resolution issues related to DNS is tested, as the IP address is not memorized. Here a successful reply from www.youtube.com is received, so the device is connected to the internet, and all hardware such as cables, the network card, the router and the modem are all working correctly, including the internet service provider. Four packets response were received from 172.217.16.142 with their TTL, 75ms delay so also an average of 75ms.



The screenshot shows a Windows Command Prompt window titled "Command Prompt". The window displays the output of a "ping" command to the website "www.youtube.com". The output shows four successful replies from the IP address 172.217.16.142, each with a TTL of 57 and a 75ms round-trip time. It also provides statistics for the ping operation, indicating 0% loss and an average of 75ms. The taskbar at the bottom of the screen shows various pinned icons and the date and time as 30 April 2022, 21:44.

```
Microsoft Windows [Version 10.0.22000.613]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Rania Shahwan> ping www.youtube.com

Pinging youtube-ui.l.google.com [172.217.16.142] with 32 bytes of data:
Reply from 172.217.16.142: bytes=32 time=75ms TTL=57

Ping statistics for 172.217.16.142:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 75ms, Maximum = 75ms, Average = 75ms

C:\Users\Rania Shahwan>
```

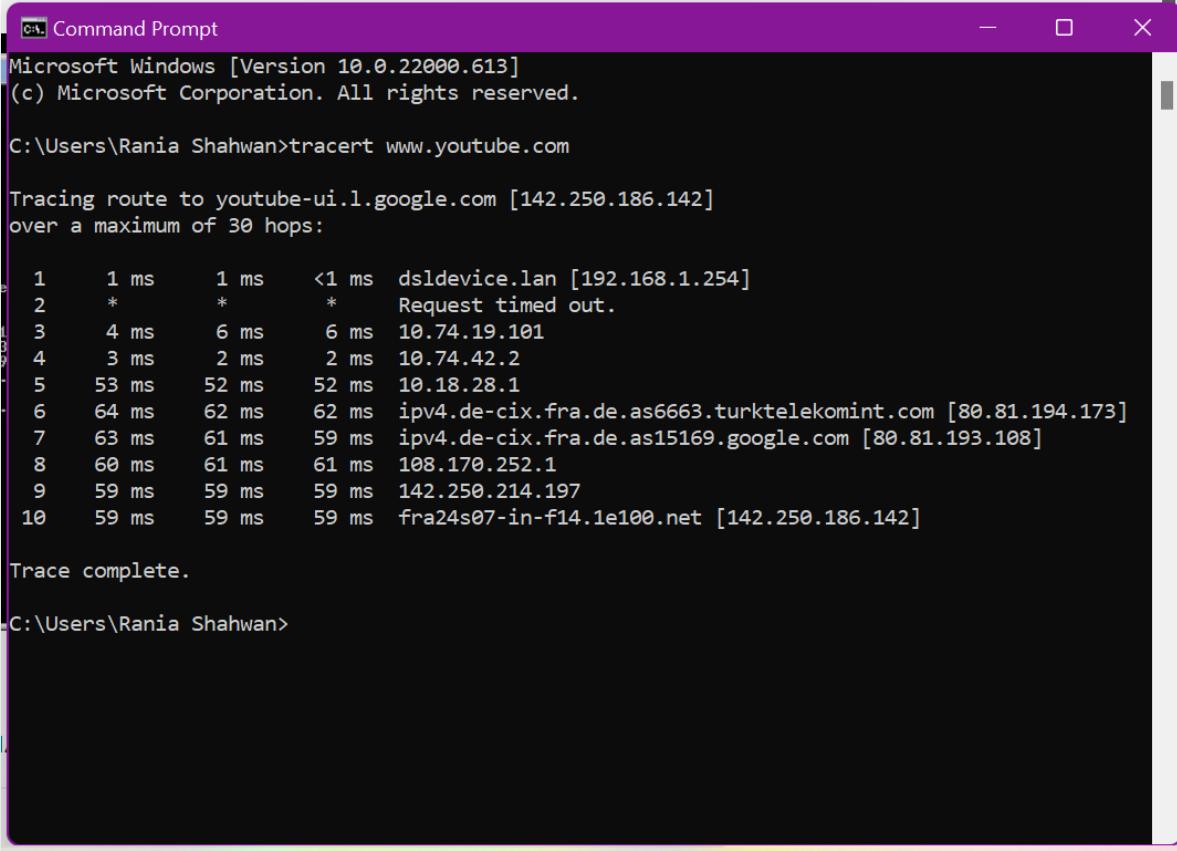
Figure 2: Ping a website (youtube)

3- Traceroute www.youtube.com

Traceroute command (tracert) is a utility designed can be used for troubleshooting of connection issue, displays the time it takes for a packet of information to travel between a local device and a destination of IP address or domain. After performing this command for www.youtube.com, the results displayed are the hops that data packets take along their path to destination desired. When the measurements are stars (*) in request timed out and indicates that the router at that hop doesn't respond to the traceroute requests.

The response structure is Hop RTT1 RTT2 RTT3 Name [IP Address].

Where hop number is the path from sender to receiver, and round trip time (RTT) time it takes for a packet to get to a hop and back, in milliseconds (ms).



```
Command Prompt
Microsoft Windows [Version 10.0.22000.613]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Rania Shahwan>tracert www.youtube.com

Tracing route to youtube-ui.l.google.com [142.250.186.142]
over a maximum of 30 hops:

 1    1 ms      1 ms      <1 ms  dsldevice.lan [192.168.1.254]
 2    *         *         * Request timed out.
 3    4 ms      6 ms      6 ms  10.74.19.101
 4    3 ms      2 ms      2 ms  10.74.42.2
 5    53 ms     52 ms     52 ms  10.18.28.1
 6    64 ms     62 ms     62 ms  ipv4.de-cix.fra.de.as6663.turktelekomint.com [80.81.194.173]
 7    63 ms     61 ms     59 ms  ipv4.de-cix.fra.de.as15169.google.com [80.81.193.108]
 8    60 ms     61 ms     61 ms  108.170.252.1
 9    59 ms     59 ms     59 ms  142.250.214.197
10    59 ms     59 ms     59 ms  fra24s07-in-f14.1e100.net [142.250.186.142]

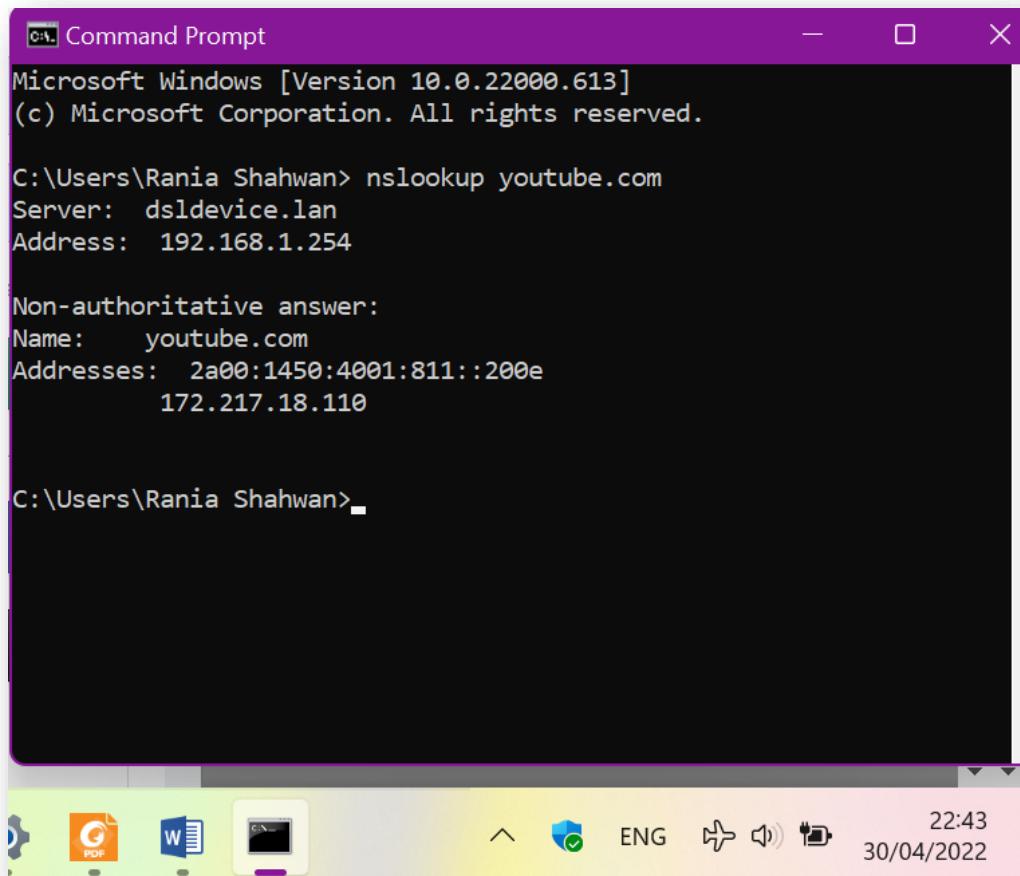
Trace complete.

C:\Users\Rania Shahwan>
```

Figure 3: Traceroute command

4- nslookup www.youtube.com

Command of nslookup followed by the domain name displays a record IP address of the domain. It queries to domain name and address servers and gets the details. This can be shown in figure below.



```
Windows [Version 10.0.22000.613]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Rania Shahwan> nslookup youtube.com
Server:  dsdevice.lan
Address:  192.168.1.254

Non-authoritative answer:
Name:      youtube.com
Addresses:  2a00:1450:4001:811::200e
           172.217.18.110

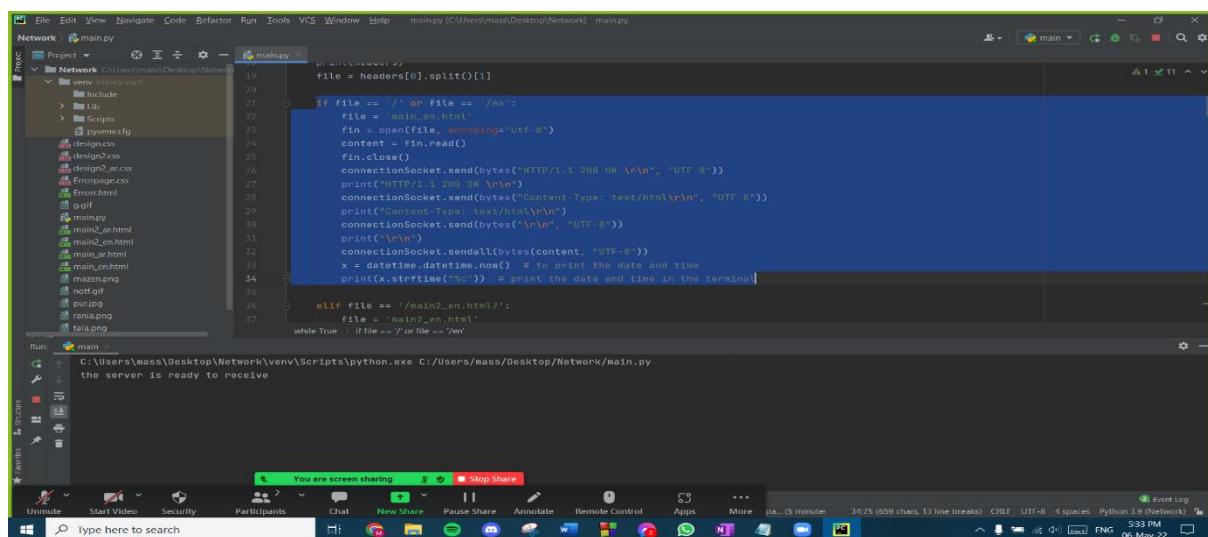
C:\Users\Rania Shahwan>
```

Figure 4: Nslookup Command

Part2: Creating a webserver

In this part of the project, the socket programming in Python language was applied to carry out a simple web server that serves eight different requests specified and shows the HTTP requests on the terminal window.

- [First Page English Version Website](#)
- If the request is sent with /, en, then the server sends main_en.html file with content-type: text/html.



The screenshot shows the PyCharm IDE interface. The left sidebar displays a project structure for 'Network' with files like 'main.py', 'main_en.html', 'main_ar.html', and various CSS and image files. The main editor window contains the following Python code:

```
if __name__ == '__main__':
    while True:
        headers = connectionSocket.recv(1024).decode('utf-8').split()
        file = headers[0].split()[1]
        if file == '/' or file == '/en':
            file = 'main_en.html'
            fin = open(file, encoding='utf-8')
            content = fin.read()
            fin.close()
            connectionSocket.sendall(bytes("HTTP/1.1 200 OK \r\n\r\n", "UTF-8"))
            print("HTTP/1.1 200 OK \r\n\r\n")
            connectionSocket.sendall(bytes("Content-Type: text/html\r\n\r\n", "UTF-8"))
            connectionSocket.sendall(bytes("\r\n\r\n", "UTF-8"))
            connectionSocket.sendall(bytes(content, "UTF-8"))
            x = datetime.datetime.now() # to print the date and time
            print(x.strftime("%s")) # print the date and time in the terminal
        elif file == '/main2_en.html':
            file = 'main2_en.html'
        else:
            file = 'index.html'
        while True:
            if file == '/' or file == '/en':
```

The terminal window at the bottom shows the command: C:\Users\mass\Desktop\Network\venv\Scripts\python.exe C:/Users/mass/Desktop/Network/main.py and the message: the server is ready to receive.

Figure 5: Page 1 English Version request

- The following website appears, it was styled using html and css

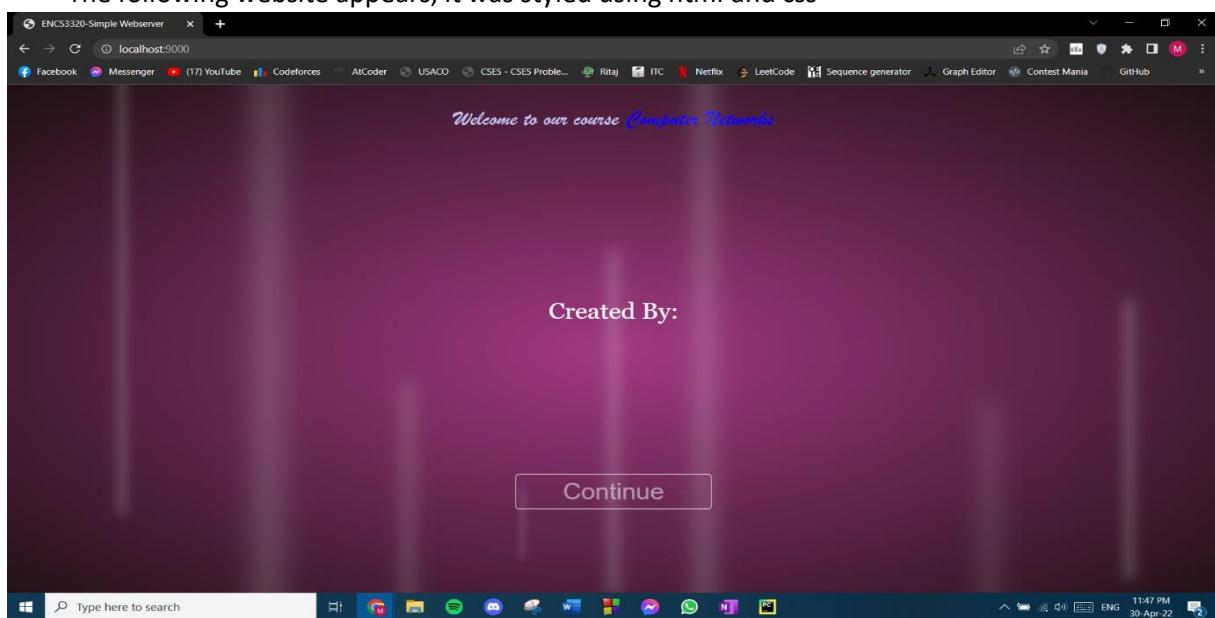


Figure 6: Page 1 English Version website

- HTML section for the first page of English version website

For the HTML code, “Welcome to our course computer networks” and “created by our team names and IDs, stay tuned” were printed on the first page of our website. And a “continue” button was added with a link to the second page of the website. This is the first page of the English version of the website. All these texts were styled in the CSS code. A link for the CSS code was added to the HTML code shown above.

```
1  <!DOCTYPE html>
2  <html>
3      <html>
4          <head>
5              <link rel="stylesheet" href="design.css">
6              <title>ENCS3320-Simple Webserver </title>
7          </head>
8          <body>
9
10         <p id='part1' class='main'> Welcome to our course <span class='part2'>Computer Networks</span></p>
11
12     <p id='head1' class='header'>Network Project </p>
13     <p id='head2' class='header'>Created By: </p>
14     <p id='head3' class='header'> Tala Dweikat 1191590 </p>
15     <p id='head4' class='header'> Rania Shahwan 1190585 </p>
16     <p id='heads' class='header'> Mazen Batrawi 1190102 </p>
17     <p id='head6' class='header'>Stay Tuned! </p>
18
19     <form action="main_en.html" method="get">
20         <button type="submit" formaction="main2_en.html">Continue</button>
21     </form>
22     <div class='light x1'></div>
23     <div class='light x2'></div>
24     <div class='light x3'></div>
25     <div class='light x4'></div>
26     <div class='light x5'></div>
27     <div class='light x6'></div>
28     <div class='light x7'></div>
29     <div class='light x8'></div>
30     <div class='light x9'></div>
31
32     </body>
</html>
```

Figure 7: Main_en.html code

o CSS section for the first page of English version website

The CSS code shown below is the one which styles the website and texts. For the background image, a GIF image was chosen with the purple colour, as shown in figure 6 above. “Welcome to our course” was written in white, “computer networks” was written in blue as required. A timer was set for each name and ID. When one of the names and numbers disappear the other appears after 4 seconds. When we press the “continue” button that was styled in white, we will get to the second page of the English version website.

```

body {
    margin: 0;
    height: 100vh;
    font-weight: 100;
    background: radial-gradient(■#a23982, ■#f1f013);
    -webkit-overflow-x: hidden;
    -moz-overflow-y: hidden;
    -o-overflow-y: hidden;
    overflow-y: hidden;
    -webkit-animation: fadeIn 1 s ease-out;
    -moz-animation: fadeIn 1 s ease-out;
    -o-animation: fadeIn 1 s ease-out;
    animation: fadeIn 1 s ease-out;
}
.main{
    color: ■rgb(196, 196, 240);
    text-align: center;
    font-style: italic;
    font-size: 30px;
    font-family: Brush Script MT;
}
.part1 {
    color: ■blue;
    text-align: center;
}
.nav{
    color: ■#FFF;
    float: right;
    margin: 30px 90px;
}
.nav ul{
    list-style: none;
}
.nav ul li {
    float: left;
    transition: .3s;
}
.nav ul li a{
    text-decoration: none;
    color: ■#591BC5;;
}
.nav ul li a:hover{
    text-decoration: none;
    color: ■#591BC5;;
}
.nav ul li a{
    height: 45px;
    padding-top: 30px;
    margin-top: -30px;
    background: ■#EFEFEF;
    text-decoration: none;
    transform: skew(15deg);
}
button{
    position: absolute;
    border: 2px solid ■white;
    background: transparent;
    font-family: 'Roboto', sans-serif;
    color: ■white;
    width: 250px;
    height: 50px;
    font-size: 2em;
    border-radius: 5px;
    opacity: .5;
    top: 60vh;
    bottom: 0px;
    left: 0px;
    right: 0px;
    margin: auto;
    transition: .3s;
}
button:hover{
    border: 2px solid ■#104F55;
    background-color: rgba(365,365,365,.5);
    cursor: pointer;
    color: ■#104F55;
    opacity: .8;
    transition: .3s;
    box-shadow: 0 8px 16px 0 ■rgba(0,0,0,.2);
}
.light{
    position: absolute;
    width: 0px;
    opacity: .5;
    background-color: ■white;
    box-shadow: ■#e9f1f1 0px 0px 20px 2px;
    opacity: 0;
    top: 100vh;
    bottom: 0px;
    left: 0px;
    right: 0px;
    margin: auto;
}
.x1{-webkit-animation: floatUp 4s infinite linear;
-moz-animation: floatUp 4s infinite linear;
-o-animation: floatUp 4s infinite linear;
animation: floatUp 4s infinite linear;
-webkit-transform: scale(1.0);
-moz-transform: scale(1.0);
-o-transform: scale(1.0);
transform: scale(1.0);
}
.x2{-webkit-animation: floatUp 7s infinite linear;
-moz-animation: floatUp 7s infinite linear;
-o-animation: floatUp 7s infinite linear;
animation: floatUp 7s infinite linear;
-webkit-transform: scale(1.6);
-moz-transform: scale(1.6);
-o-transform: scale(1.6);
transform: scale(1.6);
}
.x3{-webkit-animation: floatUp 2.5s infinite linear;
-moz-animation: floatUp 2.5s infinite linear;
-o-animation: floatUp 2.5s infinite linear;
animation: floatUp 2.5s infinite linear;
-webkit-transform: scale(.5);
-moz-transform: scale(.5);
-o-transform: scale(.5);
transform: scale(.5);
left: -15%;
}
.x4{-webkit-animation: floatUp 4.5s infinite linear;
-moz-animation: floatUp 4.5s infinite linear;
-o-animation: floatUp 4.5s infinite linear;
animation: floatUp 4.5s infinite linear;
-webkit-transform: scale(1.2);
-moz-transform: scale(1.2);
-o-transform: scale(1.2);
transform: scale(1.2);
left: -34%;}
.x5{-webkit-animation: floatUp 8s infinite linear;
-moz-animation: floatUp 8s infinite linear;
-o-animation: floatUp 8s infinite linear;
animation: floatUp 8s infinite linear;
-webkit-transform: scale(2.2);
-moz-transform: scale(2.2);
-o-transform: scale(2.2);
transform: scale(2.2);
}
.x6{-webkit-animation: floatUp 3s infinite linear;
-moz-animation: floatUp 3s infinite linear;
-o-animation: floatUp 3s infinite linear;
animation: floatUp 3s infinite linear;
-webkit-transform: scale(.8);
-moz-transform: scale(.8);
-o-transform: scale(.8);
transform: scale(.8);
left: -81%;}
.x7{-webkit-animation: floatUp 5.3s infinite linear;
-moz-animation: floatUp 5.3s infinite linear;
-o-animation: floatUp 5.3s infinite linear;
animation: floatUp 5.3s infinite linear;
-webkit-transform: scale(3.2);
-moz-transform: scale(3.2);
-o-transform: scale(3.2);
transform: scale(3.2);
left: 37%;}
.x8{-webkit-animation: floatUp 4.7s infinite linear;
-moz-animation: floatUp 4.7s infinite linear;
-o-animation: floatUp 4.7s infinite linear;
animation: floatUp 4.7s infinite linear;
-webkit-transform: scale(1.7);
-moz-transform: scale(1.7);
-o-transform: scale(1.7);
transform: scale(1.7);
left: 62%;}
.x9{-webkit-animation: floatUp 4.1s infinite linear;
-moz-animation: floatUp 4.1s infinite linear;
-o-animation: floatUp 4.1s infinite linear;
animation: floatUp 4.1s infinite linear;
-webkit-transform: scale(0.9);
}
.x10{-webkit-animation: fadeIn 1 3s ease-in;
-moz-animation: fadeIn 1 3s ease-in;
-o-animation: fadeIn 1 3s ease-in;
animation: fadeIn 1 3s ease-in;
-webkit-animation-delay: 4s;
-moz-animation-delay: 4s;
-o-animation-delay: 4s;
animation-delay: 4s;
}
.x11{-webkit-animation: fadeOut 1 3s ease-in;
-moz-animation: fadeOut 1 3s ease-in;
-o-animation: fadeOut 1 3s ease-in;
animation: fadeOut 1 3s ease-in;
-webkit-animation-delay: 8s;
-moz-animation-delay: 8s;
-o-animation-delay: 8s;
animation-delay: 8s;
}
.x12{-webkit-animation: fadeOut 1 3s ease-in;
-moz-animation: fadeOut 1 3s ease-in;
-o-animation: fadeOut 1 3s ease-in;
animation: fadeOut 1 3s ease-in;
-webkit-animation-delay: 20s;
-moz-animation-delay: 20s;
-o-animation-delay: 20s;
animation-delay: 20s;
}
.x13{-webkit-keyframes fadeIn{ from{opacity: 0; to{opacity: 1;}}
@webkit-keyframes fadeOut{ 0%{opacity: 0;} 30%{opacity: 1;} 50%{opacity: 0;} 75%{opacity: 1;} 100%{top: -100vh; opacity: 0;}}
@keyframes fadeIn{ from{opacity: 0; to{opacity: 1;}}
@keyframes fadeOut{ 0%{top: 100vh; opacity: 0;} 25%{opacity: 1;} 50%{top: 0vh; opacity: .8;} 75%{opacity: 1;} 100%{top: -100vh; opacity: 0;}}
@keyframes floatUp{ 0%{top: 100vh; opacity: 0;} 25%{opacity: 1;} 50%{top: 0vh; opacity: .8;} 75%{opacity: 1;} 100%{top: -100vh; opacity: 0;}}
@keyframes finalFade{ 0%{top: 100vh; opacity: 0;} 30%{opacity: 1;} 50%{top: 0vh; opacity: .8;} 75%{opacity: 1;} 100%{top: -100vh; opacity: 0;}}
}
.header{ position: absolute;
top: 40%; }
.footer{
font-family: 'Roboto', sans-serif;
font-size: 1.2em;
color: ■white;
position: fixed;
-webkit-transform: translate(95vw,90vh);
-moz-transform: translate(95vw,90vh);
transform: translate(95vw,90vh);
transform: translate(95vw,90vh);
}

```

Figure 8: Design.css code

- [Second page of English version](#)
 - When the ‘continue’ button on the first page on English version is pressed. Then a link to the second page will be open, and the server sends main2_en.html file with content-type: text/html.

The screenshot shows the PyCharm IDE interface with the following details:

- Project:** Network > main.py
- Code Editor:** The main.py file is open, displaying Python code for a network application. The code handles requests for files like 'main2_en.html', 'main_ar.html', and 'index'. It uses socket programming to send HTTP responses with content-type 'text/html' and includes date/time printing logic.
- Toolbars:** Standard PyCharm toolbars for navigation, search, and file operations.
- Status Bar:** Shows the current file path as 'main.py C:\Users\mas\Desktop\Network', line count (11), character count (49,659), line breaks (13), and encoding (UTF-8). It also indicates 4 spaces and Python 3.9 (Network).
- Bottom Bar:** Includes icons for Unmute, Start Video, Security, Participants, Chat, New Share, Pause Share, Annotate, Remote Control, Apps, More, and a search bar.

Figure 9: Page 2 English Version request

- The following website appears, it was styled using html and CSS

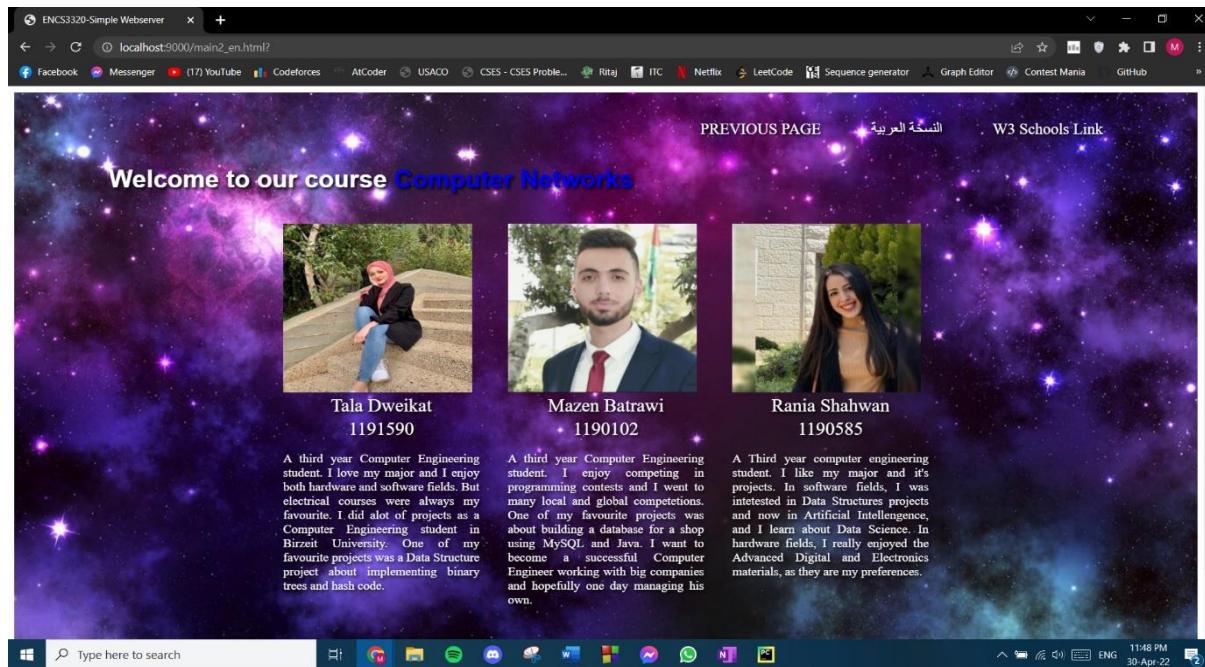


Figure 10: Page 2 English version website

- HTML section for the second page of English version website

After pressing the ‘continue’ button, the second page of the English version website appears. This page consists of three sections which have the names, information and pictures (with .png extension) of group members, each section has information about a single member of the team. Links of social media accounts will be associated with a hover property on the pictures of the group members. ‘Welcome to our course computer networks’ was written on the page. On the top of the page, some links were associated; a link to the previous page (leads to the first page on English version website), a link to the Arabic version website (leads to the first page of Arabic version website), and a link to WS schools website. A background image with the .jpg extension was chosen.

```
<!DOCTYPE html>
<html>
<html>
<title> ENCS3320-Simple Webserver </title>
</head>
<body>
    <div class="container">
        <div class="navbar">
            <nav>
                <ul>
                    <li><a href="main_en.html">PREVIOUS PAGE</a> </li>
                    <li><a href="main_ar.html">الصفحة السابقة</a> </li>
                    <li><a href="https://www.w3schools.com/tags/att_img_src.asp?fbclid=IwAR3KUUbX4L0iJeN43kmow_qOhjPaoc">WS Schools</a> </li>
                </ul>
            </nav>
        </div>
        <div class="row">
            <div class="col">
                <h1 id="part1" class='main'> Welcome to our course <span class='part2'>Computer Networks</span></h1>
            </div>
        </div>
        <div class="row2">
            <div class="col1">
                <div class="image1">
                    
                    <div class="img1_overlay img1_overlay_blur">
                        <div class="img1_title">Social Media Accounts</div>
                        <p class="img1_descrition">
                            <a href="https://www.facebook.com/tala.dweikat.39" target="_blank">
                                <ion-icon name="logo-facebook" size="large" color="blue"></ion-icon> </a>
                            <a href="https://www.instagram.com/tala_dweikat/" target="_blank">
                                <ion-icon name="logo-instagram" size="large" color="blue"></ion-icon></a>
                            </p>
                        </div>
                    </div>
                    <p class="names">Tala Dweikat</p>
                    <p class="numbers">1191590</p>
                    <p>A third year Computer Engineering student. I love my major and I enjoy both hardware and software fields. But electrical courses were always my favourite. I did a lot of projects as a Computer Engineering student in Birzeit University. One of my favourite projects was a Data Structure project about implementing binary trees and hash code.</p>
                </div>
            <div class="col2">
                <div class="image2">
                    
                    <div class="img2_overlay img2_overlay_blur">
                        <div class="img2_title">Social Media Accounts</div>
                        <p class="img2_descrition">
                            <a href="https://www.facebook.com/mazen.batrawi" target="_blank">
                                <ion-icon name="logo-facebook" size="large" color="blue"></ion-icon> </a>
                            <a href="https://www.instagram.com/mazenbatrawi/" target="_blank">
                                <ion-icon name="logo-instagram" size="large" color="blue"></ion-icon> </a>
                            </p>
                        </div>
                    </div>
                    <p class="names">Mazen Batrawi</p>
                    <p class="numbers">1190102</p>
                    <p>A third year Computer Engineering student. I enjoy competing in programming contests and I went to many local and global competitions. One of my favourite projects was about building a database for a shop using MySQL and Java. I want to become a successful Computer Engineer working with big companies and hopefully one day manage my own.</p>
                </div>
            <div class="col3">
                <div class="image3">
                    
                    <div class="img3_overlay img3_overlay_blur">
                        <div class="img3_title">Social Media Accounts</div>
                        <p class="img3_descrition">
                            <a href="https://www.facebook.com/rania185" target="_blank">
                                <ion-icon name="logo-facebook" size="large" color="blue"></ion-icon> </a>
                            <a href="https://www.instagram.com/rania_t_shahwan/" target="_blank">
                                <ion-icon name="logo-instagram" size="large" color="blue"></ion-icon> </a>
                            </p>
                        </div>
                    </div>
                    <p class="names">Rania Shahwan</p>
                    <p class="numbers">1190585</p>
                    <p>A Third year computer engineering student. I like my major and it's projects. In software fields, I was interested in Data Structures projects and now in Artificial Intelligence, and I learn about Data Science. In hardware fields, I really enjoyed the Advanced Digital and Electronics materials, as they are my preferences.</p>
                </div>
            </div>
        <script type="module" src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.esm.js"></script>
        <script nomodule src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.js"></script>
    </div>
</body>
</html>
```

Figure 11: main2_en.html code

- CSS section for the second page of English version website

This section is for styling the website. All information about the group members are separated in three boxes, box for each. They were written in white with pictures of the group members. When you get the mouse over the three pictures of the group members, the pictures will be hovered and the social media accounts for each of them will appear on the picture. When you press on the Facebook logo, a link will be associated with it and it will open a new website with the page of Facebook account, the same will happen when pressing on the Instagram logo. ‘Welcome to our course’ was written in white, ‘computer networks’ was written in blue as required. On the top of the page, some links were associated; when pressing on them, a new website will be open with the link associated in the HTML section. They were written in white.

```
main {
  color: #rgb(255, 255, 255);
  margin: 0;
  padding: 0;
  font-family: 'Roboto', sans-serif;
  text-shadow: 2px 2px 4px #000000;
  padding-bottom: 30px;
}

.part2 {
  color: #blue;
  text-align: left;
  padding-bottom: 30px;
}

body {
  margin: 0;
  padding: 0;
}

.row {
  width: 900px;
  color: #rgb(255, 255, 255);
  text-shadow: 2px 2px 4px #000000;
}

.container {
  width: 100%;
  height: 110vh;
  background-image: url(pur.jpg);
  background-color: black;
  background-position: center;
  background-size: cover;
  padding-left: 8%;
  padding-right: 8%;
  box-sizing: border-box;
}

.navbar {
  height: 12%;
  display: flex;
  align-items: center;
}

.navbar a {
  color: #white;
  font-family: 'Quicksand', sans-serif;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  opacity: 0;
  transition: opacity 0.5s;
}

.navbar a:hover {
  opacity: 1;
}

.navbar a * {
  transform: translateY(20px);
  transition: transform 0.5s;
}

.navbar a:hover * {
  transform: translateY(0);
}

.navbar .img1_title {
  font-size: 1.2em;
  font-weight: bold;
  color: #blue;
}

.navbar .img1_description {
  font-size: 1.25em;
  margin-top: 0.25em;
}

.navbar .img1_overlay_blur {
  backdrop-filter: blur(5px);
}

.navbar .image2 {
  position: relative;
  width: 240px;
  height: 220px;
}

.navbar .img2 {
  width: 240px;
  height: 220px;
}

.navbar .img2_overlay {
  position: absolute;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  background-color: #rgba(0,0,0,0.6);
  color: #white;
  font-family: 'Quicksand', sans-serif;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  opacity: 0;
  transition: opacity 0.5s;
}

.navbar .img2_overlay:hover {
  opacity: 1;
}

.navbar .img2_overlay * {
  transform: translateY(20px);
  transition: transform 0.5s;
}

.navbar .img2_overlay:hover * {
  transform: translateY(0);
}

.navbar .img2_title {
  font-size: 1.2em;
  font-weight: bold;
  color: #blue;
}

.navbar .img2_description {
  font-size: 1.25em;
  margin-top: 0.25em;
}

.navbar .img2_overlay_blur {
  backdrop-filter: blur(5px);
}

.navbar .image3 {
  position: relative;
  width: 240px;
  height: 220px;
}

.navbar .img3 {
  width: 240px;
  height: 220px;
}

.navbar .img3_overlay {
  position: absolute;
  top: 0;
  left: 0;
  width: 100%;
  height: 100%;
  background-color: #rgba(0,0,0,0.6);
  color: #white;
  font-family: 'Quicksand', sans-serif;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  opacity: 0;
  transition: opacity 0.5s;
}

.navbar .img3_overlay:hover {
  opacity: 1;
}

.navbar .img3_overlay * {
  transform: translateY(20px);
  transition: transform 0.5s;
}

.navbar .img3_overlay:hover * {
  transform: translateY(0);
}

.navbar .img3_title {
  font-size: 1.2em;
  font-weight: bold;
  color: #blue;
}

.navbar .img3_description {
  font-size: 1.25em;
  margin-top: 0.25em;
}

.navbar .img3_overlay_blur {
  backdrop-filter: blur(5px);
}
```

Figure 12: design2.css code

- First page of Arabic version
 - If the request is sent with /ar then the server response with main_ar.html which is an Arabic version of main_en.html

The screenshot shows the PyCharm IDE interface with a Python script named `main.py` open. The code implements a simple web server using sockets. It handles requests for files like `main_ar.html` and `main2_ar.html`, printing the date and time in the terminal. The PyCharm interface includes a Project tool window, a Run tool window, and a bottom toolbar with various icons.

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help main.py (C:\Users\masai\Desktop\Network) - main.py

Project Network C:\Users\masai\Desktop\Network
  -> Network
    -> venv library root
      -> Include
      -> Lib
      -> Scripts
        -> pyvenv.cfg
    design.css
    design2.css
    design2_ar.css
    Errpage.css
    Error.html
    gif.gif
    main.py
    main2_ar.html
    main2_en.html
    main_ar.html
    main_en.html
    mazen.png
    notif.gif
    pur.jpg
    rania.png
    talal.png

main.py
  x = datetime.datetime.now() # To print the date and time
  print(x.strftime("%c")) # print the date and time in the terminal

  if file == '/ar':
    file = 'main_ar.html'
    fin = open(file, encoding="utf-8")
    content = fin.read()
    fin.close()
    connectionSocket.send(bytes("HTTP/1.1 200 OK \r\n", "UTF-8"))
    print("HTTP/1.1 200 OK \r\n")
    connectionSocket.send(bytes("Content-Type: text/html\r\n", "UTF-8"))
    print("Content-Type: text/html\r\n")
    connectionSocket.send(bytes("\r\n", "UTF-8"))
    print("\r\n")
    connectionSocket.sendall(bytes(content, "UTF-8"))
    x = datetime.datetime.now() # to print the date and time
    print(x.strftime("%c")) # print the date and time in the terminal

  elif file == '/main2_ar.html?':
    file = 'main2_ar.html'
    while True:
      if file == '/ar'

Run: main
  [ "GET /pur.jpg HTTP/1.1\r\n", "Host: localhost:9080\r\n", "Connection: keep-alive\r\n", "sec-ch-ua: \" Not A;Brand\";v=99", "Chromium";v="101", "Google Chrome";v="101"\r\n", "sec-ch-ua-mobile: ?0\r\n", "sec-ch-ua-platform: \"Windows\"\r\n", "Upgrade-Insecure-Requests: 1\r\n", "User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.64 Safari/537.36\r\n", "Accept: image/jpeg\r\n", "Accept-Encoding: gzip, deflate\r\n", "Accept-Language: en-US,en;q=0.9\r\n" ]
  Content-Type: image/jpeg

  b''
  Date: Fri May 6 17:36:05 2022

  You are screen sharing
  Stop Share

Unmute Start Video Security Participants Chat New Share Pause Share Annotate Remote Control Apps More
  patty... 7 minute 51.5 (646 chars, 13 line breaks) CR LF UTF-8 4 spaces Python 3.9 (Network)
  Event Log
  5:36 PM 06-May-22
```

Figure 13: Page 1 Arabic version request

- The following website will appear, it was styled with HTML and CSS:



Figure 14: first page of Arabic version website

- HTML section for the first page of Arabic version website

باتمام الطلبة: مع اسماء والارقام الجامعية ”اهلا بكم في مساق شبكات الحاسوب“ and ”اعضاء الفريق“ were printed on the first page of our website. And ”استمرار“ button was added with a link to the second page of the website. This is the first page of the Arabic version of the website. All these texts were styled in the CSS code. A link for the CSS code was added to the HTML code shown above.

```
<!DOCTYPE html>
<html>
  <html>
    <head>
      <link rel="stylesheet" href="design.css">
    <title>
      ENCS3320- ويب سيرفر
    </title>
    </head>
    <body>
      <p id='part1' class='main'><span class='part2'> أهلا وسهلا بكم في مساق </span></p>
      <p id='head1' class='header'> مشروع شبكات الحاسوب </p>
      <p id='head2' class='header'> باتمام الطلبة : </p>
      <p id='head3' class='header'> 1191590 </p>
      <p id='head4' class='header'> 1190585 </p>
      <p id='heads' class='header'> 1190102 </p>
      <p id='head6' class='header'> مازن بطرولي </p>
      <p id='head6' class='header'> رانيا شهوان </p>
      <p id='head6' class='header'> ترقبوا المزيد ! </p>
    <form action="main_ar.html" method="get">
      <button type="submit" formaction="main2_ar.html">استمرار</button>
    </form>
    <div class='light x1'></div>
    <div class='light x2'></div>
    <div class='light x3'></div>
    <div class='light x4'></div>
    <div class='light x5'></div>
    <div class='light x6'></div>
    <div class='light x7'></div>
    <div class='light x8'></div>
    <div class='light x9'></div>
  </body>
</html>
```

Figure 15: main_ar.html code

- HTML section for the second page of Arabic version website



Figure 16: second page of Arabic version website

After pressing ‘استمرار’ button, the second page of the Arabic version website appears. This page consists of three sections which have the names, information and pictures (with .png extension) of group members all in Arabic, each section has information about a single member of the team. Links of social media accounts will be associated with a hover propriety on the pictures of the group members. ‘اهلا بكم في مساق شبكات الحاسوب’ was written on the page. On the top of the page, some links were associated; a link to the previous page (leads to the first page on English version website), a link to the English version website (leads to the first page of English version website), and a link to WS schools website. A background image with the .jpg extension was chosen.

```
<!DOCTYPE html>
<html><head>
<title>ENCS3320 - دين سيرفر</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=0" />
<link rel="stylesheet" href="design2_ar.css" />
<script type="module" src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.esm.js"></script>
<script nomodule src="https://unpkg.com/ionicons@5.5.2/dist/ionicons/ionicons.js"></script>
</head>
<body>
<div class="container">
<div class="nav_bar">
<ul>
<li><a href="#">الصفحة السابقة</li>
<li><a href="#">English Version</li>
<li><a href="#">W3 Schools Link</li>

```

Figure 17: main2_ar.html code

- CSS section for the first and second page of Arabic version website

The CSS code shown below is the one which styles the website and texts. For the background image, a GIF image was chosen with the purple colour, as shown in figure 13 above. “اهلا بكم في مساق” was written in white, “شبكات الحاسوب” was written in blue as required. A timer was set for each name and ID which were written in Arabic. When one of the names and numbers disappear the other appears after 4 seconds. When we press the “استمرار” button that was styled in white, we will get to the second page of the Arabic version website. In the second page: all information about the group members are separated in three boxes, box for each. They were written in white and arabic with pictures of the group members. When you get the mouse over the three pictures of the group members, the pictures with be hovered and “موقع التواصل الاجتماعي” for each of them will appear on the picture. When you press on the Facebook logo, a link will be associated with it and it will open a new website with the page of Facebook account, the same will happen when pressing on the Instagram logo. ‘اهلا بكم في مساق’ was written in white, ‘شبكات الحاسوب’ was written in blue as required. On the top of the page, some links were associated; when pressing on them, a new website will be open with the link associated in the HTML section. They were written in white.

```

.main { color: #rgb(255, 255, 255); nav ul li a { text-decoration: none; text-align: right; font-family: 'roboto', sans-serif; text-shadow: 2px 2px 4px #000000; padding-bottom: 30px; } .row2 { color: #blue; text-align: right; padding-bottom: 30px; } .body { margin: 0; padding: 0; } .row { width: 900px; color: #rgb(255, 255, 255); text-shadow: 2px 2px 4px #000000; } .content { width: 100%; height: 100vh; background-image: url(pur.jpg); background-color: #black; background-position: center; background-size: cover; padding-left: 8%; padding-right: 8%; box-sizing: border-box; } .navbar { height: 12%; display: flex; align-items: center; } nav { flex: 1; text-align: left; } nav li { list-style: none; display: inline-block; margin-left: 60px; } 
```

```

text-align-last: right; text-justify: inter-word; } .names { font-size: 1.5rem; font-weight: 500; text-align: center; margin-bottom: 0.2rem; } img { width: 240px; height: 240px; } .image1 { position: relative; width: 250px; height: 240px; } .img1_overlay{ position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #rgba(0,0,0,0.6); color: #white; font-family: 'quicksand',sans-serif; display: flex; flex-direction: column; align-items: center; justify-content: center; opacity: 0; transition: opacity 0.5s; } .img1_overlay:hover{ opacity: 1; } .img1_overlay > * { transform: translateY(20px); transition: transform 0.5s; } .img1_overlay:hover > * { transform: translateY(0); } .img1_title{ font-size: 1.2em; font-weight: bold; color: #blue; } .img1_description{ font-size: 1.25em; margin-top: 0.25em; } .img1_overlay_blur{ backdrop-filter: blur(5px); } 
```

```

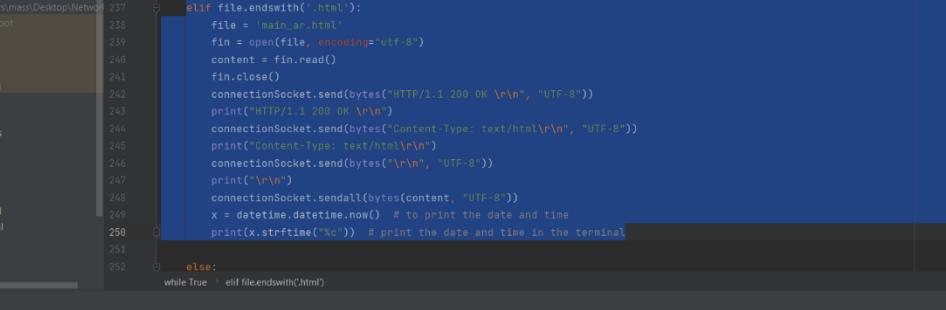
text-align-last: right; text-justify: inter-word; } .numbers { font-size: 1.5rem; font-weight: 500; text-align: center; margin-bottom: 0.2rem; } img { width: 240px; height: 240px; } .image2 { position: relative; width: 250px; height: 240px; } .img2_overlay{ position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #rgba(0,0,0,0.6); color: #white; font-family: 'quicksand',sans-serif; display: flex; flex-direction: column; align-items: center; justify-content: center; opacity: 0; transition: opacity 0.5s; } .img2_overlay:hover{ opacity: 1; } .img2_overlay > * { transform: translateY(20px); transition: transform 0.5s; } .img2_overlay:hover > * { transform: translateY(0); } .img2_title{ font-size: 1.2em; font-weight: bold; color: #blue; } .img2_description{ font-size: 1.25em; margin-top: 0.25em; } .img2_overlay_blur{ backdrop-filter: blur(5px); } 
```

```

text-align-last: right; text-justify: inter-word; } .names { font-size: 1.5rem; font-weight: 500; text-align: center; margin-bottom: 0.2rem; } img { width: 240px; height: 240px; } .image3 { position: relative; width: 250px; height: 240px; } .img3_overlay{ position: absolute; top: 0; left: 0; width: 100%; height: 100%; background-color: #rgba(0,0,0,0.6); color: #white; font-family: 'quicksand',sans-serif; display: flex; flex-direction: column; align-items: center; justify-content: center; opacity: 0; transition: opacity 0.5s; } .img3_overlay:hover{ opacity: 1; } .img3_overlay > * { transform: translateY(20px); transition: transform 0.5s; } .img3_overlay:hover > * { transform: translateY(0); } .img3_title{ font-size: 1.2em; font-weight: bold; color: #blue; } .img3_description{ font-size: 1.25em; margin-top: 0.25em; } .img3_overlay_blur{ backdrop-filter: blur(5px); } 
```

Figure 18: Design2_ar.css code

- Request ends with .html:



```
File Edit View Navigate Code Refactor Run Tools VCS Window Help main.py [C:\Users\mass\Desktop\Network] - main.py

Project Network main.py
main.py
Network C:\Users\mass\Desktop\Network 237
  +-- venv
    +-- bin
      +-- include
      +-- Lib
      +-- Scripts
        +-- pyenv.cfg
    +-- design.css
    +-- design2.css
    +-- design2_ar.css
    +-- Errpage.css
    +-- Error.html
    +-- ggf
    +-- main.py
    +-- main2_ar.html
    +-- main2_en.html
    +-- main_ar.html
    +-- main_en.html
    +-- mazzen.png
    +-- www.wtf

src/main.py:237
elif file.endswith('.html'):
    file = "main_ar.html"
    fin = open(file, encoding='utf-8')
    content = fin.read()
    fin.close()

    connectionSocket.send(bytes("HTTP/1.1 200 OK \r\n", "UTF-8"))
    print("HTTP/1.1 200 OK \r\n")
    connectionSocket.send(bytes("Content-Type: text/html\r\n", "UTF-8"))
    print("Content-type: text/html\r\n")
    connectionSocket.send(bytes("\r\n", "UTF-8"))
    print("\r\n")
    connectionSocket.sendall(bytes(content, "UTF-8"))
    x = datetime.datetime.now() # to print the date and time
    print(x.strftime("%c")) # print the date and time in the terminal

else:
    while True : elif file.endswith(html)

Run: main
Accept-Language: en-US,en;q=0.9
Cookie: Pycharm-73568b47=6a3f3d2a-9ff6-4014-a132-8bcea01c539b

[ 'GET /design.css HTTP/1.1\r', 'Host: localhost:9080\r', 'Connection: keep-alive\r', 'sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="101", "Google Chrome";v="101"\r', 'sec-ch-ua-m
HTTP/1.1 200 OK

Content-Type: text/css

Date: Fri May 6 17:45:55 2022

You are screen sharing Stop Share

Unmute Start Video Security Participants Chat New Share Pause Share Annotate Remote Control Apps More pattern to file type 'CSV' from plugin 'CSV Plugin' // Revert this ... (17 min) 545 PM 06-May-22 ENG Python 3.9 (Network) Event Log
```

Figure 19: A .html request

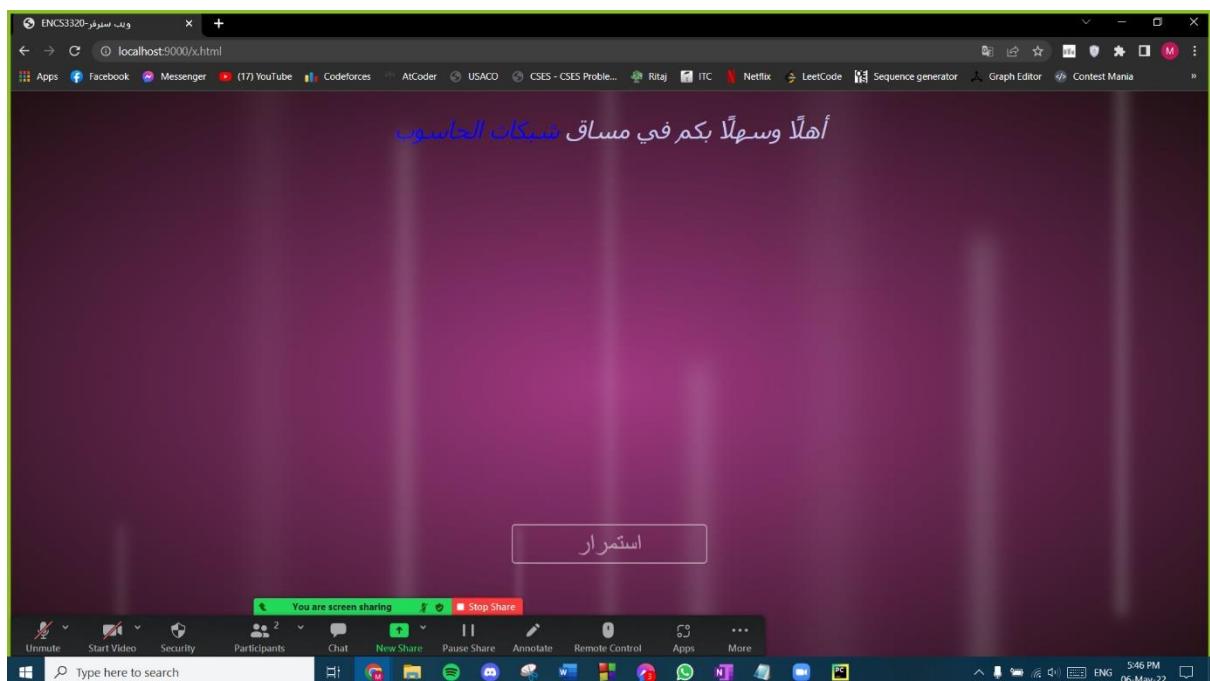
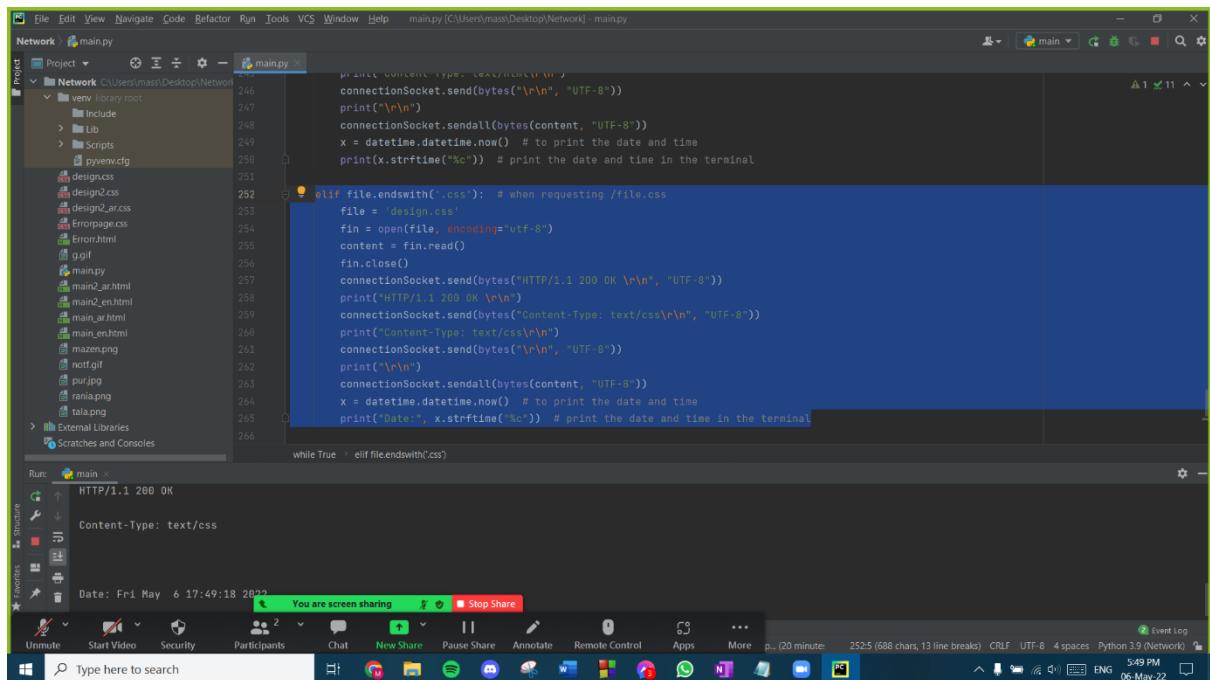


Figure 20: Response of a .html request

- Request ends with **.css**:



The screenshot shows a Python IDE interface with a code editor and a terminal window. The code editor displays a file named `main.py` containing Python code for a web server. The terminal window shows the output of the server's response to a CSS request.

```

PROJECT CONCERN: type: CALCULATE_DATE_TIME
connectionSocket.send(bytes("\r\n", "UTF-8"))
print("\r\n")
connectionSocket.sendall(bytes(content, "UTF-8"))
x = datetime.datetime.now() # to print the date and time
print(x.strftime("%c")) # print the date and time in the terminal

elif file.endswith('.css'): # when requesting /file.css
    file = 'design.css'
    fin = open(file, encoding='utf-8')
    content = fin.read()
    fin.close()
    connectionSocket.send(bytes("HTTP/1.1 200 OK \r\n", "UTF-8"))
    print("HTTP/1.1 200 OK \r\n")
    connectionSocket.send(bytes("Content-Type: text/css\r\n", "UTF-8"))
    print("Content-Type: text/css\r\n")
    connectionSocket.send(bytes("\r\n", "UTF-8"))
    print("\r\n")
    connectionSocket.sendall(bytes(content, "UTF-8"))
    x = datetime.datetime.now() # to print the date and time
    print("Date:", x.strftime("%c")) # print the date and time in the terminal

```

The terminal window shows the server's response:

```

while True : elif file.endswith('.css')
HTTP/1.1 200 OK
Content-Type: text/css
Date: Fri May 6 17:49:18 2022

```

Figure 21: A .css request

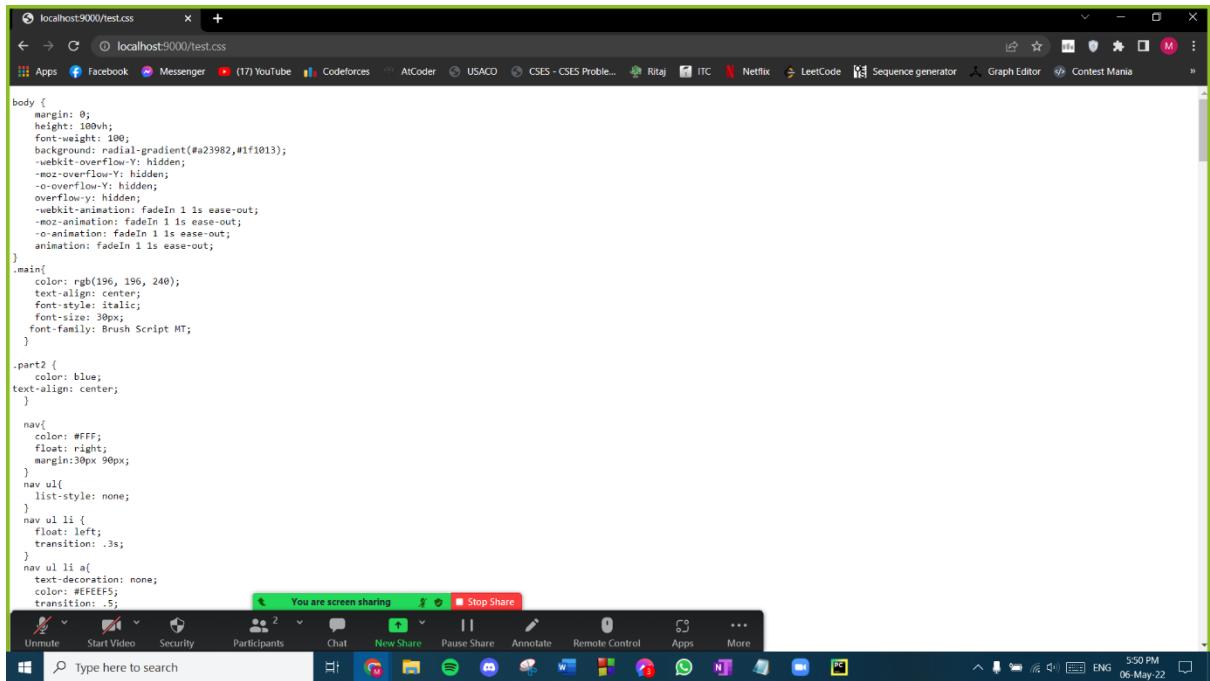
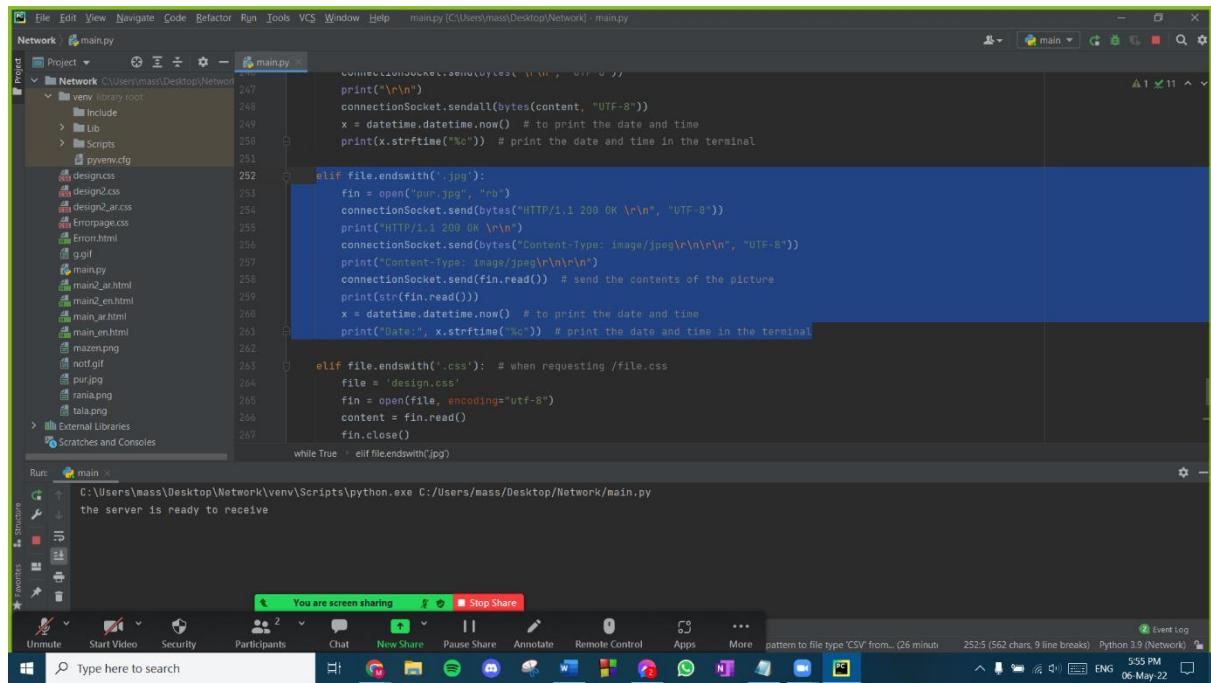


Figure 22: Response of a .css request

- **JPG request**



```

File Edit View Navigate Code Refactor Run Tools VCS Window Help main.py [C:\Users\mass\Desktop\Network] - main.py
Network Project Network C:\Users\mass\Desktop\Network\main.py
  venv library root
    include
      lib
    scripts
      pyvenv.cfg
    design.css
    design2.css
    design2_ar.css
    Errorpage.css
    Error.html
    g.gif
    main.py
    main_ar.html
    main_en.html
    main_en.html
    mazen.png
    notif.gif
    pur.jpg
    rania.png
    tala.png
  External Libraries
  Scratches and Consoles
  while True:
    if file.endswith('.jpg'):
        fin = open('pur.jpg', 'rb')
        connectionSocket.send(bytes("HTTP/1.1 200 OK \r\n\r\n", "UTF-8"))
        print("HTTP/1.1 200 OK \r\n\r\n")
        connectionSocket.send(bytes("Content-Type: image/jpeg\r\n\r\n", "UTF-8"))
        print("Content-Type: image/jpeg\r\n\r\n")
        connectionSocket.send(fin.read())
        print(str(fin.read()))
        fin.close()
    elif file.endswith('.css'): # when requesting /file.css
        file = 'design.css'
        fin = open(file, encoding="utf-8")
        content = fin.read()
        fin.close()
        connectionSocket.send(content)
    else:
        connectionSocket.send(bytes("HTTP/1.1 404 Not Found \r\n\r\n", "UTF-8"))
        print("HTTP/1.1 404 Not Found \r\n\r\n")
        connectionSocket.send(bytes("Content-Type: text/html\r\n\r\n", "UTF-8"))
        print("Content-Type: text/html\r\n\r\n")
        connectionSocket.send(main_ar.html)
        print(str(main_ar.html))
        connectionSocket.send(main_en.html)
        print(str(main_en.html))
        connectionSocket.send(main_en.html)
        print(str(main_en.html))
        connectionSocket.send(mazen.png)
        print(str(mazen.png))
        connectionSocket.send(notif.gif)
        print(str(notif.gif))
        connectionSocket.send(pur.jpg)
        print(str(pur.jpg))
        connectionSocket.send(ania.png)
        print(str(ania.png))
        connectionSocket.send(tala.png)
        print(str(tala.png))

  Run: main
  C:\Users\mass\Desktop\Network\venv\Scripts\python.exe C:/Users/mass/Desktop/Network/main.py
  the server is ready to receive
  You are screen sharing Stop Share
  Unmute Start Video Security Participants Chat New Share Pause Share Annotate Remote Control Apps More pattern to file type 'CSV' from... (26 minutes) 2525 (562 chars, 9 line breaks) Python 3.9 (Network) Event Log 5:55 PM 06-May-22
  Type here to search

```

Figure 23: A .jpg request

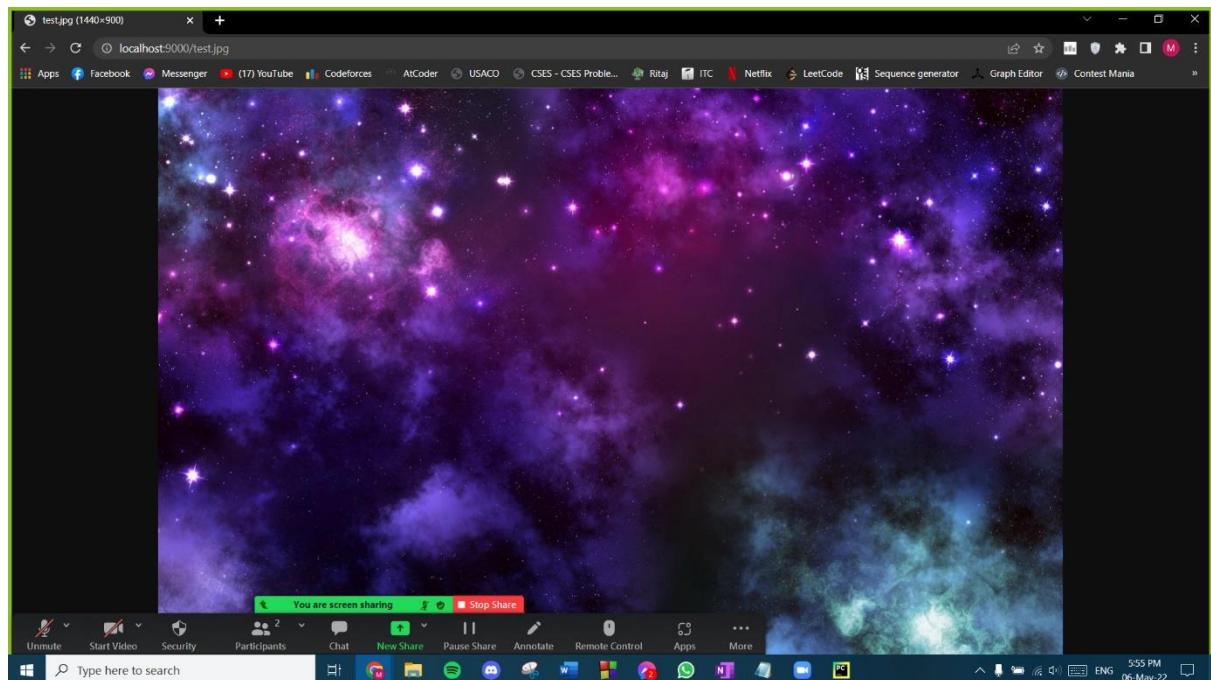


Figure 24: Response of a .jpg request

- Request ends with **PNG**:

The screenshot shows the PyCharm IDE interface with the following details:

- Project:** Network
- File:** main.py
- Code Editor:** The code is a Python script for a simple web server. It handles file requests and serves images from a local directory.

```
    (f".close()
connectionSocket.send(bytes("HTTP/1.1 200 OK \r\n", "UTF-8"))
print("HTTP/1.1 200 OK \r\n")
connectionSocket.send(bytes("Content-Type: text/css\r\n", "UTF-8"))
print("Content-Type: text/css\r\n")
connectionSocket.send(bytes("\r\n", "UTF-8"))
print("\r\n")
connectionSocket.sendall(bytes(content, "UTF-8"))
x = datetime.datetime.now() # to print the date and time
print("Date:", x.strftime("%c")) # print the date and time in the terminal

elif file.endswith('.png'):
fin = open("tala.png", "rb")
connectionSocket.send(bytes("HTTP/1.1 200 OK \r\n", "UTF-8"))
print("HTTP/1.1 200 OK \r\n")
connectionSocket.send(bytes("Content-Type: image/png\r\n\r\n", "UTF-8"))
print("Content-Type: image/png\r\n\r\n")
connectionSocket.send(fin.read()) # send the contents of the picture
print(str(fin.read()))
x = datetime.datetime.now() # to print the date and time
print("Date:", x.strftime("%c")) # print the date and time in the terminal

while True : elif file.endswith('.png')
```

- Run:** The run configuration is set to main.py.
- Output:** The terminal shows the server is ready to receive connections.
- Bottom Bar:** Shows system icons for Unmute, Start Video, Security, Participants, Chat, New Share, Pause Share, Annotate, Remote Control, Apps, More, pattern to file type 'CSV' from..., 2785 (561 chars, 9 line breaks) Python 3.9 (Network), and event log.

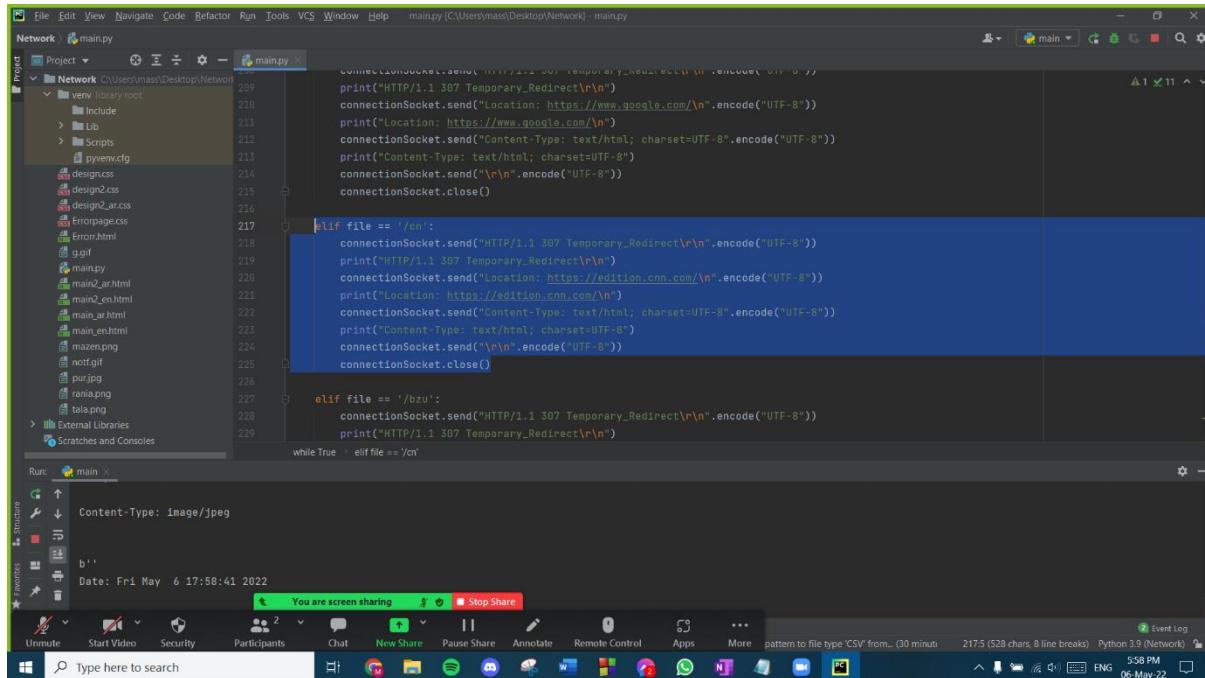
Figure 25: A .png request



Figure 26: Response of a .png request

- **307 Temporary redirect**

We send the server the status 307 and specify the redirect location and the content type as text/html and the encoding. Then, the server redirects to the specified location based on the given request.



```

File Edit View Navigate Code Refactor Run Tools VCS Window Help mainpy (C:\Users\mass\Desktop\Network) - main.py
Network > main.py
Project Network C:\Users\mass\Desktop\Network
  venv library root
    > Lib
    > Scripts
      pyvenv.cfg
    design.css
    design2.css
    design2_ar.css
    Erroneous.css
    Error.html
    g.gif
    main.py
    main_ar.html
    main_ar.html
    main_en.html
    main_en.html
    mazeng.png
    notif.gif
    purring
    rani.png
    talia.png
  > External Libraries
  Scratches and Consoles
Run: main
Content-Type: image/jpeg
b''
Date: Fri May 6 17:58:41 2022
You are screen sharing | Stop Share
Unmute Start Video Security Participants Chat New Share Pause Share Annotate Remote Control Apps More pattern to file type 'CSV' from... (30 min) 2175 (529 chars, 8 line breaks) Python 3.9 (Network) 5:58 PM 06-May-22 Event Log
Type here to search

```

```

File Edit View Navigate Code Refactor Run Tools VCS Window Help mainpy (C:\Users\mass\Desktop\Network) - main.py
Network > main.py
Project Network C:\Users\mass\Desktop\Network
  venv library root
    > Lib
    > Scripts
      pyvenv.cfg
    design.css
    design2.css
    design2_ar.css
    Erroneous.css
    Error.html
    g.gif
    main.py
    main_ar.html
    main_ar.html
    main_en.html
    main_en.html
    mazeng.png
    notif.gif
    purring
    rani.png
    talia.png
  > External Libraries
  Scratches and Consoles
Run: main
Content-Type: image/jpeg
b''
Date: Fri May 6 17:58:41 2022
You are screen sharing | Stop Share
Unmute Start Video Security Participants Chat New Share Pause Share Annotate Remote Control Apps More pattern to file type 'CSV' from... (30 min) 2175 (529 chars, 8 line breaks) Python 3.9 (Network) 5:58 PM 06-May-22 Event Log
Type here to search

```

```

connectionSocket.send("HTTP/1.1 307 Temporary_Redirect\r\n".encode("UTF-8"))
print("HTTP/1.1 307 Temporary_Redirect\r\n")
connectionSocket.send("Location: https://www.google.com/\r\n".encode("UTF-8"))
print("Location: https://www.google.com/\r\n")
connectionSocket.send("Content-Type: text/html; charset=UTF-8".encode("UTF-8"))
print("Content-Type: text/html; charset=UTF-8")
connectionSocket.send("\r\n".encode("UTF-8"))
connectionSocket.close()

elif file == '/cn':
    connectionSocket.send("HTTP/1.1 307 Temporary_Redirect\r\n".encode("UTF-8"))
    print("HTTP/1.1 307 Temporary_Redirect\r\n")
    connectionSocket.send("Location: https://edition.cnn.com/\r\n".encode("UTF-8"))
    print("Location: https://edition.cnn.com/\r\n")
    connectionSocket.send("Content-Type: text/html; charset=UTF-8".encode("UTF-8"))
    print("Content-Type: text/html; charset=UTF-8")
    connectionSocket.send("\r\n".encode("UTF-8"))
    connectionSocket.close()

elif file == '/bzu':
    connectionSocket.send("HTTP/1.1 307 Temporary.Redirect\r\n".encode("UTF-8"))
    print("HTTP/1.1 307 Temporary.Redirect\r\n")
    connectionSocket.close()

while True:
    elif file == '/cn'

```

Figure 27: Requests of a 307 temporary redirect to CNN website

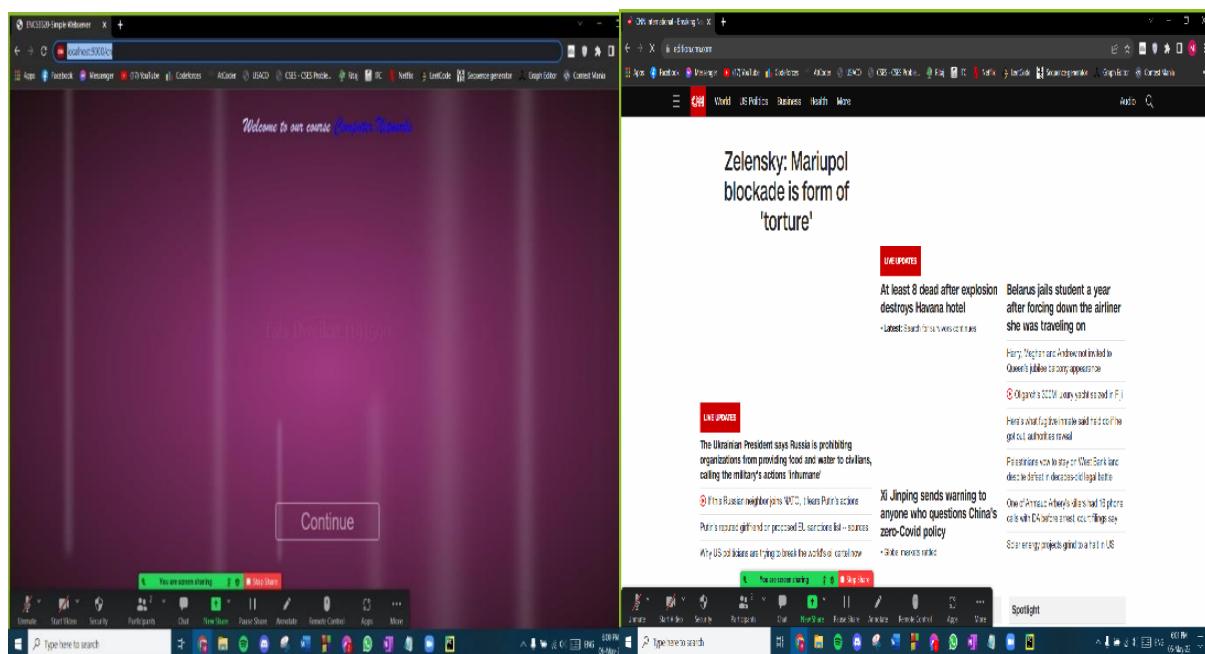


Figure 28: CNN Redirect

- Error page

If the request is not defined then we redirect to the error page with content type text/html and response status of HTTP/1.1 404 Not Found. It contains the details of the IP and port number of the client. The html code was written in the editor of the python to get the IP and the port.

The screenshot shows the PyCharm IDE interface with a Python project named "Network". The main editor window displays a file named "main.py" containing the following code:

```
A = datetime.datetime.now() # to print the date and time
print("Date:", x.strftime("%c")) # print the date and time in the terminal

if __name__ == "__main__":
    content = '<!DOCTYPE html><html><head><link rel="stylesheet" href="Errorpage.css"><title> Error</title></head><body><p class="code1"> Oops! </p> \
    <p id="msg" class="msg"> The file is not found </p><p id="names" class="Enome">We cant seem to find page you are looking for!</p><p class="code2"> Error Code: 404</p><p id="names" class="Enome"> <strong> Rania 1190585 Tala 1191590 Mezen 1190102 </strong></p> \
    <div class="ips">The IP is 648594 + ' + str(ip) + '</div><div class="port">The Port number is 648594 + ' + str(port) + '</div></body></html>'
connectionSocket.send(bytes("HTTP/1.1 404 Not Found \r\n", "UTF-8"))
print("HTTP/1.1 404 Not Found \r\n")
connectionSocket.send(bytes("Content-Type: text/html\r\n", "UTF-8"))
connectionSocket.send(bytes("\r\n", "UTF-8"))
print("Content-Type: text/html\r\n")
connectionSocket.sendall(bytes(content, "UTF-8"))
x = datetime.datetime.now()
print("Date:", x.strftime("%c"))

while True:
    pass
```

The "Run" tab at the bottom shows the command "main" and the output "Content-Type: image/gif". The status bar indicates the current date and time as "Fri May 6 18:02:29 2022". A "Stop Share" button is visible in the bottom center.

Figure 29: code of error page

- Here is the error page:

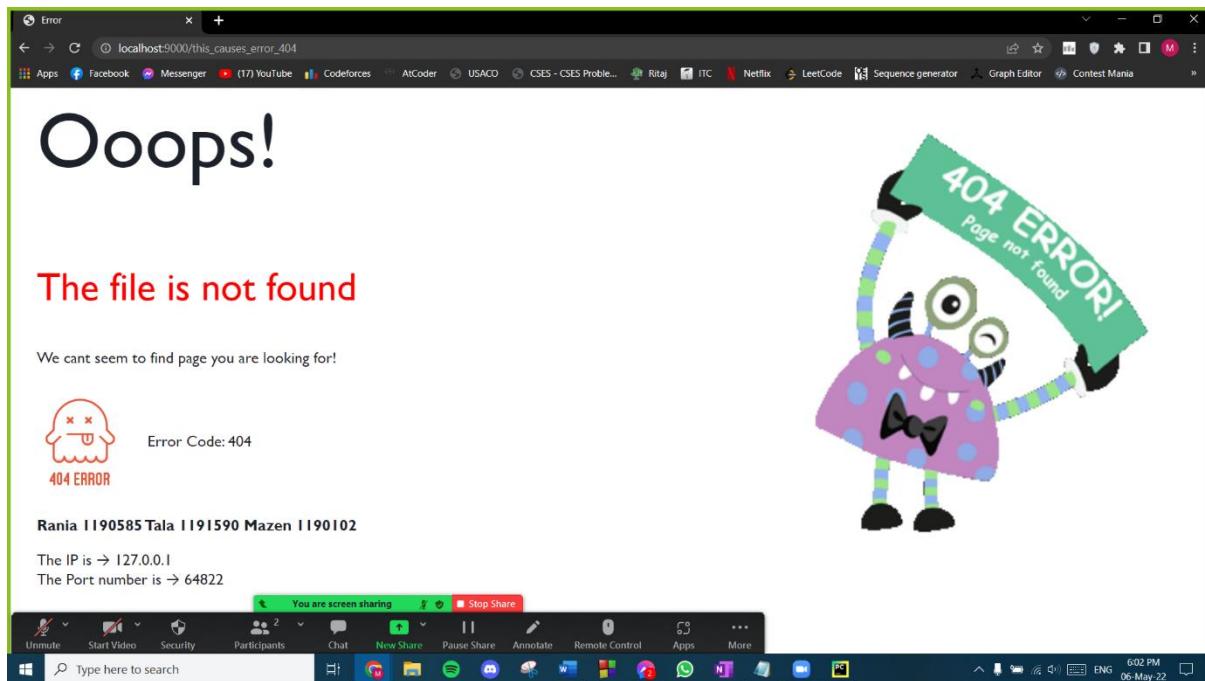


Figure 30: Error Page

As shown in the figures above, a request was sent to **localhost:9000/**. The message contents tell that the data was sent to the server using the **GET** method, the HTTP type is **HTTP/1.1**, the content type is specified like **text/html**, **text/css**, **image/png**, etc. The connection type for the request is **persistent** since that was told from the **keep-alive** statement. Similarly, for the other requests, the same operations happen but with different content types and destinations. Here are some requests for CSS files and images. The code and the response of every request was shown.

- Testing from another device

- We tested the project from a phone, everything was going well:

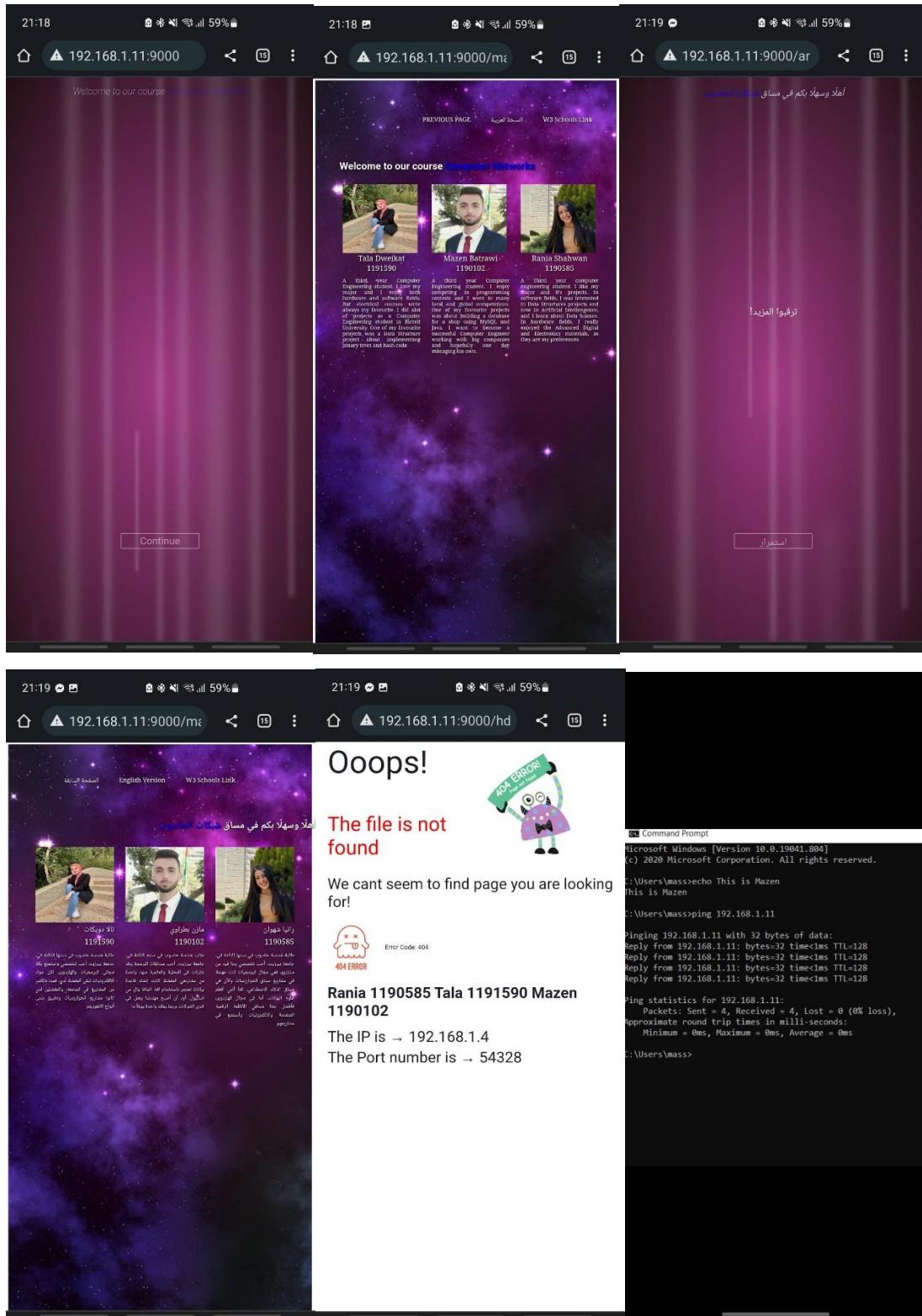


Figure 31: Testing project from a phone

- Appendix



code.txt