

Taller de Proyecto 2

Práctica 1

Integrantes:

- Arcuri, Antonio 538/8
- Blanco Regojo, Mauricio 492/0
- Santarcangelo Zazzetta, Marco 652/8

Indice

Página

Enunciado 1	1
Resolución ejercicio 1	1
Enunciado 2	2
Resolución ejercicio 2	2
Enunciado 3	19
Resolución ejercicio 3	20
Enunciado 4	20
Resolución ejercicio 4	21
Enunciado 5	21
Resolución ejercicio 5	21
Enunciado 6	22
Resolución ejercicio 6	22

Enunciado 1)

Generar el archivo 'requirements.txt' con las dependencias necesarias para poder levantar un servidor con Flask. Explicar un ejemplo de uso con la secuencia de acciones y procesos involucrados desde el inicio de la interacción con un usuario hasta que el usuario recibe la respuesta.

Resolución ejercicio 1:

En primera medida se deberá instalar tanto Python como Pip, siendo Python un lenguaje de programación y Pip un sistema de gestión de paquetes utilizado para instalar y administrar paquetes de software escritos en lenguaje Python. Para esto, utilizando el sistema operativo Ubuntu, en la terminal se ingresarán los siguientes comandos:

```
sudo apt-get install Python
```

```
sudo apt-get install Python-pip
```

Pip permite la instalación de un conjunto de paquetes, en este caso imprescindibles para levantar un servidor con Flask, a partir de la creación de un documento "requirements.txt" en el cual se encuentran detalladas las dependencias necesarias. Dentro del repositorio de GitHub del proyecto se encontrará este documento, el cual solo tiene como ítem:

Flask

Una vez generado el archivo "requirements.txt", este tiene que ser ejecutado ejecutando el comando:

```
pip install -r requirements.txt
```

El proceso se compone de dos pasos:

1. El usuario solicita acceder a la página escribiendo la dirección en el navegador, lo cual genera un método GET de HTTP.
2. El servidor responde enviando el documento HTML mediante HTTP (método GET) que es visto por el usuario en el navegador.

La Fig. 1. demuestra la metodología de conexión entre usuarios y el servidor, siendo la red de comunicaciones el navegador web.

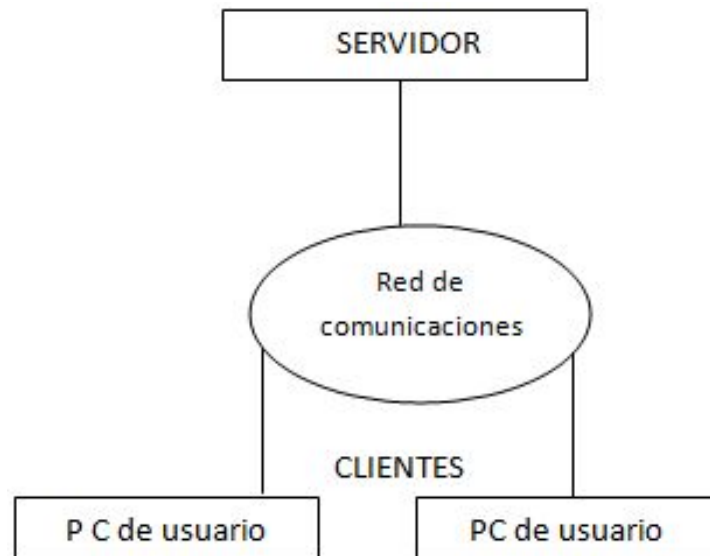


Fig. 1. Gráfico de interacción entre los usuarios y el servidor.

Enunciado 2)

Desarrollar un experimento que muestre si el servidor HTTP agrega o quita información a la genera un programa Python. Nota: debería programar o utilizar un programa Python para conocer exactamente lo que genera y debería mostrar la información que llega del lado del cliente, a nivel de HTTP o, al menos, a nivel de HTML (preferentemente HTTP).

Resolución ejercicio 2:

El servidor agrega información con el objetivo de poder enviar los datos generados originalmente desde el cliente al servidor y viceversa.

El programa utilizado es el ejemplo otorgado por la cátedra, python.ej, el cual genera información sobre el nombre, el email y el sexo del usuario.

A continuación se muestra el tráfico HTTP correspondiente a la aplicación python capturado mediante Wireshark. Los datos generados en python están marcados en negrita.

Frame 74: 360 bytes on wire (2880 bits), 360 bytes captured (2880 bits) on interface 0

Interface id: 0 (any)

Encapsulation type: Linux cooked-mode capture (25)

Arrival Time: Sep 5, 2017 19:23:01.636410766 ART

[Time shift for this packet: 0.000000000 seconds]

Epoch Time: 1504650181.636410766 seconds

[Time delta from previous captured frame: 0.000100344 seconds]

[Time delta from previous displayed frame: 0.000000000 seconds]

[Time since reference or first frame: 12.104625604 seconds]

Frame Number: 74

Frame Length: 360 bytes (2880 bits)

Capture Length: 360 bytes (2880 bits)

[Frame is marked: False]

[Frame is ignored: False]

[Protocols in frame: sll:ethertype:ip:tcp:http]

[Coloring Rule Name: HTTP]

[Coloring Rule String: http || tcp.port == 80 || http2]

Linux cooked capture

Packet type: Unicast to us (0)

Link-layer address type: 772

Link-layer address length: 6

Source: 00:00:00_00:00:00 (00:00:00:00:00:00)

Protocol: IPv4 (0x0800)

Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1

0100 = Version: 4

.... 0101 = Header Length: 20 bytes (5)

Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

0000 00.. = Differentiated Services Codepoint: Default (0)

.... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)

Total Length: 344

Identification: 0xe62e (58926)

Flags: 0x02 (Don't Fragment)

0... = Reserved bit: Not set

.1.. = Don't fragment: Set

..0. = More fragments: Not set

Fragment offset: 0

Time to live: 64

Protocol: TCP (6)

Header checksum: 0x556f [validation disabled]

[Header checksum status: Unverified]

Source: 127.0.0.1

Destination: 127.0.0.1

[Source GeoIP: Unknown]

[Destination GeoIP: Unknown]

Transmission Control Protocol, Src Port: 42910, Dst Port: 8080, Seq: 1, Ack: 1,
Len: 292

Source Port: 42910

Destination Port: 8080

[Stream index: 2]

[TCP Segment Len: 292]

Sequence number: 1 (relative sequence number)

[Next sequence number: 293 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
Header Length: 32 bytes
Flags: 0x018 (PSH, ACK)
000. = Reserved: Not set
...0 = Nonce: Not set
.... 0... = Congestion Window Reduced (CWR): Not set
.... .0.. = ECN-Echo: Not set
.... ..0. = Urgent: Not set
.... ...1 = Acknowledgment: Set
.... 1... = Push: Set
.... 0.. = Reset: Not set
....0. = Syn: Not set
....0 = Fin: Not set
[TCP Flags:AP...]
Window size value: 342
[Calculated window size: 43776]
[Window size scaling factor: 128]
Checksum: 0xff4c [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP),

Timestamps

No-Operation (NOP)

Type: 1

0... = Copy on fragmentation: No
..00. = Class: Control (0)
...0 0001 = Number: No-Operation (NOP) (1)

No-Operation (NOP)

Type: 1

0... = Copy on fragmentation: No
..00. = Class: Control (0)
...0 0001 = Number: No-Operation (NOP) (1)

Timestamps: TSval 179248, TSecr 179248

Kind: Time Stamp Option (8)

Length: 10

Timestamp value: 179248

Timestamp echo reply: 179248

[SEQ/ACK analysis]

[iRTT: 0.000026374 seconds]

[Bytes in flight: 292]

[Bytes sent since last PSH flag: 292]

Hypertext Transfer Protocol

GET / HTTP/1.1\r\n

[Expert Info (Chat/Sequence): GET / HTTP/1.1\r\n]
[GET / HTTP/1.1\r\n]
[Severity level: Chat]
[Group: Sequence]
Request Method: GET
Request URI: /
Request Version: HTTP/1.1
Host: localhost:8080\r\n
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:45.0)
Gecko/20100101 Firefox/45.0\r\n
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n
Accept-Language: en-US,en;q=0.5\r\n
Accept-Encoding: gzip, deflate\r\n
Connection: keep-alive\r\n
\r\n
[Full request URI: http://localhost:8080/]
[HTTP request 1/1]
[Response in frame: 88]

Frame 88: 5402 bytes on wire (43216 bits), 5402 bytes captured (43216 bits) on interface 0

Interface id: 0 (any)
Encapsulation type: Linux cooked-mode capture (25)
Arrival Time: Sep 5, 2017 19:23:01.638112424 ART
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1504650181.638112424 seconds
[Time delta from previous captured frame: 0.000010371 seconds]
[Time delta from previous displayed frame: 0.001701658 seconds]
[Time since reference or first frame: 12.106327262 seconds]
Frame Number: 88
Frame Length: 5402 bytes (43216 bits)
Capture Length: 5402 bytes (43216 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: sll:ethertype:ip:tcp:http:data-text-lines]
[Coloring Rule Name: HTTP]
[Coloring Rule String: http || tcp.port == 80 || http2]
Linux cooked capture
Packet type: Unicast to us (0)
Link-layer address type: 772
Link-layer address length: 6

Source: 00:00:00_00:00:00 (00:00:00:00:00:00)
 Protocol: IPv4 (0x0800)
 Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
 0100 = Version: 4
 0101 = Header Length: 20 bytes (5)
 Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
 0000 00.. = Differentiated Services Codepoint: Default (0)
00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
 Total Length: 5386
 Identification: 0x6bb7 (27575)
 Flags: 0x02 (Don't Fragment)
 0... = Reserved bit: Not set
 .1.. = Don't fragment: Set
 ..0. = More fragments: Not set
 Fragment offset: 0
 Time to live: 64
 Protocol: TCP (6)
 Header checksum: 0xbc34 [validation disabled]
 [Header checksum status: Unverified]
 Source: 127.0.0.1
 Destination: 127.0.0.1
 [Source GeoIP: Unknown]
 [Destination GeoIP: Unknown]
 Transmission Control Protocol, Src Port: 8080, Dst Port: 42910, Seq: 158, Ack: 293, Len: 5334
 Source Port: 8080
 Destination Port: 42910
 [Stream index: 2]
 [TCP Segment Len: 5334]
 Sequence number: 158 (relative sequence number)
 [Next sequence number: 5492 (relative sequence number)]
 Acknowledgment number: 293 (relative ack number)
 Header Length: 32 bytes
 Flags: 0x018 (PSH, ACK)
 000. = Reserved: Not set
 ...0 = Nonce: Not set
 0... = Congestion Window Reduced (CWR): Not set
0.. = ECN-Echo: Not set
0. = Urgent: Not set
1 = Acknowledgment: Set
 1... = Push: Set
0.. = Reset: Not set
0. = Syn: Not set
0 = Fin: Not set

[TCP Flags:AP...]

Window size value: 350

[Calculated window size: 44800]

[Window size scaling factor: 128]

Checksum: 0x12ff [unverified]

[Checksum Status: Unverified]

Urgent pointer: 0

Options: (12 bytes), No-Operation (NOP), No-Operation (NOP),

Timestamps

No-Operation (NOP)

Type: 1

0... = Copy on fragmentation: No

.00. = Class: Control (0)

...0 0001 = Number: No-Operation (NOP) (1)

No-Operation (NOP)

Type: 1

0... = Copy on fragmentation: No

.00. = Class: Control (0)

...0 0001 = Number: No-Operation (NOP) (1)

Timestamps: TSval 179249, TSecr 179249

Kind: Time Stamp Option (8)

Length: 10

Timestamp value: 179249

Timestamp echo reply: 179249

[SEQ/ACK analysis]

[iRTT: 0.000026374 seconds]

[Bytes in flight: 5334]

[Bytes sent since last PSH flag: 5334]

TCP segment data (5334 bytes)

[7 Reassembled TCP Segments (5491 bytes): #76(17), #78(40), #80(22), #82(39),
#84(37), #86(2), #88(5334)]

[Frame: 76, payload: 0-16 (17 bytes)]

[Frame: 78, payload: 17-56 (40 bytes)]

[Frame: 80, payload: 57-78 (22 bytes)]

[Frame: 82, payload: 79-117 (39 bytes)]

[Frame: 84, payload: 118-154 (37 bytes)]

[Frame: 86, payload: 155-156 (2 bytes)]

[Frame: 88, payload: 157-5490 (5334 bytes)]

[Segment count: 7]

[Reassembled TCP length: 5491]

[Reassembled TCP Data:

485454502f312e3020323030204f4b0d0a436f6e74656e74...]

Hypertext Transfer Protocol

HTTP/1.0 200 OK\r\n

[Expert Info (Chat/Sequence): HTTP/1.0 200 OK\r\n]

[HTTP/1.0 200 OK\r\n]

[Severity level: Chat]

[Group: Sequence]

Request Version: HTTP/1.0

Status Code: 200

Response Phrase: OK

Content-Type: text/html; charset=utf-8\r\n

Content-Length: 5334\r\n

[Content length: 5334]

Server: Werkzeug/0.12.2 Python/2.7.12\r\n

Date: Tue, 05 Sep 2017 22:23:01 GMT\r\n

\r\n

[HTTP response 1/1]

[Time since request: 0.001701658 seconds]

[Request in frame: 74]

File Data: 5334 bytes

Line-based text data: text/html

<html>\n

<head>\n

\n

<title>TP2</title>\n

<meta name="viewport" content="width=device-width, initial-scale=1">\n

<!-- JQuery -->\n

<script src="/static/jquery/jquery-3.2.1.min.js"></script> \n

<!-- Bootstrap CSS -->\n

<link rel="stylesheet" href="/static/bootstrap/css/bootstrap.min.css" />\n

<!-- Bootstrap JS -->\n

<script src="/static/bootstrap/js/popper.min.js"></script>\n

<script src="/static/bootstrap/js/bootstrap.min.js"></script>\n

\n

</head>\n

<body> \n

\n

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">\n

TP2\n

<button class="navbar-toggler" type="button"

data-toggle="collapse" data-target="#navbarNavDropdown"

aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle

navigation">\n

\n

</button>\n

<div class="collapse navbar-collapse"

id="navbarNavDropdown">\n

```

        <ul class="navbar-nav">\n
        <li class="nav-item active">\n
            <a class="nav-link" href="#">Formulario <span
class="sr-only">(current)</span></a>\n
        </li>\n
        <li class="nav-item">\n
            <a class="nav-link" href="http://info.unlp.edu.ar/">Fac.
Inform\303\241tica</a>\n
        </li>\n
        <li class="nav-item">\n
            <a class="nav-link" href="http://www.ing.unlp.edu.ar/">Fac.
Ingenier\303\255a</a>\n
        </li>\n
        <li class="nav-item dropdown">\n
            <a class="nav-link dropdown-toggle"
href="http://example.com" id="navbarDropdownMenuLink"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">\n
                Links de inter\303\251s\n
            </a>\n
            <div class="dropdown-menu"
aria-labelledby="navbarDropdownMenuLink">\n
                <a class="dropdown-item"
href="https://ideas.info.unlp.edu.ar/">IdeasUNLP</a>\n
                <a class="dropdown-item" href="#">Otro..</a>
\n
            </div>\n
        </li>\n
    </ul>\n
</div>
\n
</nav>\n
\n
<div class="jumbotron jumbotron-fluid">\n
    <div class="container">\n
        <h3 class="display-4">Taller de Proyecto II</h3>\n
        <p class="lead">Ejemplo de teor\303\255a de Python</p>\n
    </div>\n
</div>\n
\n
<div class="container">\n
\n
\n
    <h1>Formulario de registro</h1>\n
    <form action="/form" method="POST">\n
\n

```

```

<div class="row">\n
  <div class="col-md-6 col-xs-12">\n
    <div class="form-group">\n
      <label for="nombre">Nombre</label>\n
      <input type="text" class="form-control" id="nombre"
name="nombre" aria-describedby="nombreHelp" placeholder="Ingrese su nombre"
required/>      \n
      <small id="nombreHelp" class="form-text
text-muted">Aqu\303\255 debe ingresar su nombre.</small>\n
    </div>\n
  </div>\n
  <div class="col-md-6 col-xs-12">\n
    <div class="form-group">\n
      <label for="nombre">Apellido</label>\n
      <input type="text" class="form-control" id="apellido"
name="apellido" aria-describedby="apellidoHelp" placeholder="Ingrese su
apellido" required/>      \n
      <small id="apellidoHelp" class="form-text
text-muted">Aqu\303\255 debe ingresar su apellido.</small>\n
    </div>\n
  </div>\n
</div>      \n
\n
  <div class="row">\n
    <div class="col-md-12">\n
      <div class="form-group">\n
        <label for="email">Email</label>\n
        <input type="text" class="form-control" id="email"
name="email" aria-describedby="emailHelp" placeholder="email@example.com"
required>\n
        <small id="emailHelp" class="form-text
text-muted">Aqu\303\255 debe ingresar su email.</small>\n
      </div> \n
    </div>\n
  </div>\n
\n
  <div class="row">\n
    <div class="col-md-12">\n
      <div class="form-group">\n
        <label for="sexo">Example select</label>\n
        <select class="form-control" id="sexo" name="sexo"
aria-describedby="sexoHelp" required>\n
          <option value="" disabled hidden
selected>Selecione...</option>          \n

```

```

        <option value="masc">Masculino</option>\n
        <option value="fem">Femenino</option>\n
    </select>\n
    <small id="sexoHelp" class="form-text
text-muted">Seleccione su sexo.</small>\n
    </div>\n
</div>\n
</div>\n
\n
        <button type="submit" class="btn btn-primary" style="float:
right;">Registrarme!</button>\n
    </form>\n
</div>\n
</body>\n
</html>

```

Frame 265: 528 bytes on wire (4224 bits), 528 bytes captured (4224 bits) on interface 0

```

    Interface id: 0 (any)
    Encapsulation type: Linux cooked-mode capture (25)
    Arrival Time: Sep  5, 2017 19:23:33.071715124 ART
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1504650213.071715124 seconds
    [Time delta from previous captured frame: 0.000209759 seconds]
    [Time delta from previous displayed frame: 31.433602700 seconds]
    [Time since reference or first frame: 43.539929962 seconds]
    Frame Number: 265
    Frame Length: 528 bytes (4224 bits)
    Capture Length: 528 bytes (4224 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: sll:ethertype:ip:tcp:http:urlencoded-form]
    [Coloring Rule Name: HTTP]
    [Coloring Rule String: http || tcp.port == 80 || http2]

```

Linux cooked capture

```

    Packet type: Unicast to us (0)
    Link-layer address type: 772
    Link-layer address length: 6
    Source: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Protocol: IPv4 (0x0800)

```

```

Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1
0100 .... = Version: 4

```

.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
0000 00.. = Differentiated Services Codepoint: Default (0)
.... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
Total Length: 512
Identification: 0xfadf (64223)
Flags: 0x02 (Don't Fragment)
0... = Reserved bit: Not set
.1... = Don't fragment: Set
..0. = More fragments: Not set
Fragment offset: 0
Time to live: 64
Protocol: TCP (6)
Header checksum: 0x4016 [validation disabled]
[Header checksum status: Unverified]
Source: 127.0.0.1
Destination: 127.0.0.1
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]

Transmission Control Protocol, Src Port: 42916, Dst Port: 8080, Seq: 1, Ack: 1,
Len: 460

Source Port: 42916
Destination Port: 8080
[Stream index: 6]
[TCP Segment Len: 460]
Sequence number: 1 (relative sequence number)
[Next sequence number: 461 (relative sequence number)]
Acknowledgment number: 1 (relative ack number)
Header Length: 32 bytes
Flags: 0x018 (PSH, ACK)
000. = Reserved: Not set
...0 = Nonce: Not set
.... 0... = Congestion Window Reduced (CWR): Not set
.... .0.. = ECN-Echo: Not set
.... ..0. = Urgent: Not set
.... ...1 = Acknowledgment: Set
.... 1... = Push: Set
....0.. = Reset: Not set
....0. = Syn: Not set
....0 = Fin: Not set
[TCP Flags:AP...]
Window size value: 342
[Calculated window size: 43776]
[Window size scaling factor: 128]

Checksum: 0xffff4 [unverified]
[Checksum Status: Unverified]
Urgent pointer: 0
Options: (12 bytes), No-Operation (NOP), No-Operation (NOP),

Timestamps

No-Operation (NOP)

Type: 1

0... = Copy on fragmentation: No
.00. = Class: Control (0)
...0 0001 = Number: No-Operation (NOP) (1)

No-Operation (NOP)

Type: 1

0... = Copy on fragmentation: No
.00. = Class: Control (0)
...0 0001 = Number: No-Operation (NOP) (1)

Timestamps: TSval 187107, TSecr 187107

Kind: Time Stamp Option (8)

Length: 10

Timestamp value: 187107

Timestamp echo reply: 187107

[SEQ/ACK analysis]

[iRTT: 0.000025252 seconds]

[Bytes in flight: 460]

[Bytes sent since last PSH flag: 460]

Hypertext Transfer Protocol

POST /form HTTP/1.1\r\n

[Expert Info (Chat/Sequence): POST /form HTTP/1.1\r\n]

[POST /form HTTP/1.1\r\n]

[Severity level: Chat]

[Group: Sequence]

Request Method: POST

Request URI: /form

Request Version: HTTP/1.1

Host: localhost:8080\r\n

User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:45.0)

Gecko/20100101 Firefox/45.0\r\n

Accept:

text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8\r\n

Accept-Language: en-US,en;q=0.5\r\n

Accept-Encoding: gzip, deflate\r\n

Referer: http://localhost:8080\r\n

Connection: keep-alive\r\n

Content-Type: application/x-www-form-urlencoded\r\n

Content-Length: 61\r\n

[Content length: 61]
\r\n
[Full request URI: http://localhost:8080/form]
[HTTP request 1/1]
[Response in frame: 279]
File Data: 61 bytes
HTML Form URL Encoded: application/x-www-form-urlencoded
Form item: "nombre" = "marco"
Key: nombre
Value: marco
Form item: "apellido" = "marco"
Key: apellido
Value: marco
Form item: "email" = "marco@marco.com"
Key: email
Value: marco@marco.com
Form item: "sexo" = "masc"
Key: sexo
Value: masc

Frame 279: 3254 bytes on wire (26032 bits), 3254 bytes captured (26032 bits) on interface 0

Interface id: 0 (any)
Encapsulation type: Linux cooked-mode capture (25)
Arrival Time: Sep 5, 2017 19:23:33.073771931 ART
[Time shift for this packet: 0.000000000 seconds]
Epoch Time: 1504650213.073771931 seconds
[Time delta from previous captured frame: 0.000015230 seconds]
[Time delta from previous displayed frame: 0.002056807 seconds]
[Time since reference or first frame: 43.541986769 seconds]
Frame Number: 279
Frame Length: 3254 bytes (26032 bits)
Capture Length: 3254 bytes (26032 bits)
[Frame is marked: False]
[Frame is ignored: False]
[Protocols in frame: sll:ethertype:ip:tcp:http:data-text-lines]
[Coloring Rule Name: HTTP]
[Coloring Rule String: http || tcp.port == 80 || http2]
Linux cooked capture
Packet type: Unicast to us (0)
Link-layer address type: 772
Link-layer address length: 6

Source: 00:00:00_00:00:00 (00:00:00:00:00:00)

Protocol: IPv4 (0x0800)

Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.1

0100 = Version: 4

.... 0101 = Header Length: 20 bytes (5)

Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

0000 00.. = Differentiated Services Codepoint: Default (0)

.... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)

Total Length: 3238

Identification: 0xdb1d (56093)

Flags: 0x02 (Don't Fragment)

0... = Reserved bit: Not set

.1.. = Don't fragment: Set

..0. = More fragments: Not set

Fragment offset: 0

Time to live: 64

Protocol: TCP (6)

Header checksum: 0x5532 [validation disabled]

[Header checksum status: Unverified]

Source: 127.0.0.1

Destination: 127.0.0.1

[Source GeoIP: Unknown]

[Destination GeoIP: Unknown]

Transmission Control Protocol, Src Port: 8080, Dst Port: 42916, Seq: 158, Ack: 461, Len: 3186

Source Port: 8080

Destination Port: 42916

[Stream index: 6]

[TCP Segment Len: 3186]

Sequence number: 158 (relative sequence number)

[Next sequence number: 3344 (relative sequence number)]

Acknowledgment number: 461 (relative ack number)

Header Length: 32 bytes

Flags: 0x018 (PSH, ACK)

000. = Reserved: Not set

...0 = Nonce: Not set

.... 0... = Congestion Window Reduced (CWR): Not set

.... .0.. = ECN-Echo: Not set

.... ..0. = Urgent: Not set

.... ...1 = Acknowledgment: Set

.... 1... = Push: Set

....0.. = Reset: Not set

....0. = Syn: Not set

....0 = Fin: Not set

[TCP Flags:AP...]

Window size value: 350

[Calculated window size: 44800]

[Window size scaling factor: 128]

Checksum: 0x0a9b [unverified]

[Checksum Status: Unverified]

Urgent pointer: 0

Options: (12 bytes), No-Operation (NOP), No-Operation (NOP),

Timestamps

No-Operation (NOP)

Type: 1

0... = Copy on fragmentation: No

.00. = Class: Control (0)

...0 0001 = Number: No-Operation (NOP) (1)

No-Operation (NOP)

Type: 1

0... = Copy on fragmentation: No

.00. = Class: Control (0)

...0 0001 = Number: No-Operation (NOP) (1)

Timestamps: TSval 187108, TSecr 187108

Kind: Time Stamp Option (8)

Length: 10

Timestamp value: 187108

Timestamp echo reply: 187108

[SEQ/ACK analysis]

[iRTT: 0.000025252 seconds]

[Bytes in flight: 3186]

[Bytes sent since last PSH flag: 3186]

TCP segment data (3186 bytes)

[7 Reassembled TCP Segments (3343 bytes): #267(17), #269(40), #271(22), #273(39), #275(37), #277(2), #279(3186)]

[Frame: 267, payload: 0-16 (17 bytes)]

[Frame: 269, payload: 17-56 (40 bytes)]

[Frame: 271, payload: 57-78 (22 bytes)]

[Frame: 273, payload: 79-117 (39 bytes)]

[Frame: 275, payload: 118-154 (37 bytes)]

[Frame: 277, payload: 155-156 (2 bytes)]

[Frame: 279, payload: 157-3342 (3186 bytes)]

[Segment count: 7]

[Reassembled TCP length: 3343]

[Reassembled TCP Data:

485454502f312e3020323030204f4b0d0a436f6e74656e74...]

Hypertext Transfer Protocol

HTTP/1.0 200 OK\r\n

[Expert Info (Chat/Sequence): HTTP/1.0 200 OK\r\n]

[HTTP/1.0 200 OK\r\n]

[Severity level: Chat]

[Group: Sequence]

Request Version: HTTP/1.0

Status Code: 200

Response Phrase: OK

Content-Type: text/html; charset=utf-8\r\n

Content-Length: 3186\r\n

[Content length: 3186]

Server: Werkzeug/0.12.2 Python/2.7.12\r\n

Date: Tue, 05 Sep 2017 22:23:33 GMT\r\n

\r\n

[HTTP response 1/1]

[Time since request: 0.002056807 seconds]

[Request in frame: 265]

File Data: 3186 bytes

Line-based text data: text/html

<html>\n

<head>\n

\n

<title>TP2</title>\n

<meta name="viewport" content="width=device-width, initial-scale=1">\n

<!-- JQuery -->\n

<script src="/static/jquery/jquery-3.2.1.min.js"></script> \n

<!-- Bootstrap CSS -->\n

<link rel="stylesheet" href="/static/bootstrap/css/bootstrap.min.css" />\n

<!-- Bootstrap JS -->\n

<script src="/static/bootstrap/js/popper.min.js"></script>\n

<script src="/static/bootstrap/js/bootstrap.min.js"></script>\n

\n

<style type="text/css">\n

\n

\t.page-header{\n

\t\tborder-bottom: 1px solid;\n

\t\tmargin-bottom: 2%;\n

\t}\n

\n

</style>\n

\n

</head>\n

\n

<body>\n

\n

```

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">\n
  <a class="navbar-brand" href="#">TP2</a>\n
  <button class="navbar-toggler" type="button"
data-toggle="collapse" data-target="#navbarNavDropdown"
aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle
navigation">\n
    <span class="navbar-toggler-icon"></span>\n
  </button>\n
  <div class="collapse navbar-collapse"
id="navbarNavDropdown">\n
    <ul class="navbar-nav">\n
      <li class="nav-item active">\n
        <a class="nav-link" href="#">Formulario <span
class="sr-only">(current)</span></a>\n
      </li>\n
      <li class="nav-item">\n
        <a class="nav-link" href="http://info.unlp.edu.ar/">Fac.
Inform\303\241tica</a>\n
      </li>\n
      <li class="nav-item">\n
        <a class="nav-link" href="http://www.ing.unlp.edu.ar/">Fac.
Ingenier\303\255a</a>\n
      </li>\n
      <li class="nav-item dropdown">\n
        <a class="nav-link dropdown-toggle"
href="http://example.com" id="navbarDropdownMenuLink"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">\n
          Links de inter\303\251s\n
        </a>\n
        <div class="dropdown-menu"
aria-labelledby="navbarDropdownMenuLink">\n
          <a class="dropdown-item"
href="https://ideas.info.unlp.edu.ar/">IdeasUNLP</a>\n
          <a class="dropdown-item" href="#">Otro..</a>\n
        </div>\n
      </li>\n
    </ul>\n
  </div>
</nav>\n
\n
<div class="jumbotron jumbotron-fluid">\n
  <div class="container">\n
    <h3 class="display-4">Taller de Proyecto II</h3>\n

```

```

        <p class="lead">Ejemplo de teor\303\255a de Python</p>\n
    </div>\n
    \n
    <div class="container">\n
    \t\n
    \t<div class="page-header">\n
    \t\t\t <h1>\302\241Bienvenido a TP2!</h1>        \n
    \t\t\t</div>    \t\n
    \n
    \t\t\t<ul class="list-group">\n
    \t\t\t\t<li class="list-group-item"><b>Nombre:</b> marco </li>\n
    \t\t\t\t<li class="list-group-item"><b>Apellido:</b> marco </li>\n
    \t\t\t\t<li class="list-group-item"><b>Email:</b> marco@marco.com
</li>\n
    \t\t\t\t<li class="list-group-item"><b>Sexo:</b> Masculino </li>\n
    \t\t\t</ul>    \t\n
    \n
    </div>        \n
    \n
    </body>\n
    </html>

```

Enunciado 3)

Generar un proyecto de simulación de acceso a valores de temperatura, humedad, presión atmosférica y velocidad del viento.

- a) Un proceso simulará una placa con microcontrolador y sus correspondientes sensor/es o directamente una estación meteorológica proveyendo los valores almacenados en un archivo o en una base de datos. Los valores se generan periódicamente (frecuencia de muestreo).
- b) Un proceso generará un documento HTML conteniendo:
 - i) Frecuencia de muestreo
 - ii) Promedio de las últimas 10 muestras
 - iii) La última muestra
- c) El documento HTML generado debe ser accesible y responsivo.

Aclaración: Se deberá detallar todo el proceso de adquisición de datos, cómo se ejecutan ambos procesos (ya sea threads o procesos separados), el esquema general, las decisiones tomadas en el desarrollo de cada proceso y la interacción del usuario.

Resolución ejercicio 3:

Los archivos de la resolución de este ejercicio se encuentran en el repositorio GitHub del proyecto.

Se crearon dos aplicaciones **sim.py** y **app.py**, de las cuales la primera actúa como un microcontrolador que obtiene los valores correspondientes a la temperatura, humedad, presión atmosférica y velocidad del viento, y la segunda aplicación tomará las muestras y las exhibirá en una página web. Ambas se comunican almacenando y retirando la información de un archivo de texto.

La Fig. 2. muestra como el microcontrolador añade las nuevas mediciones a un archivo, el cual posteriormente será leído por la aplicación que presentará en la página web la información.

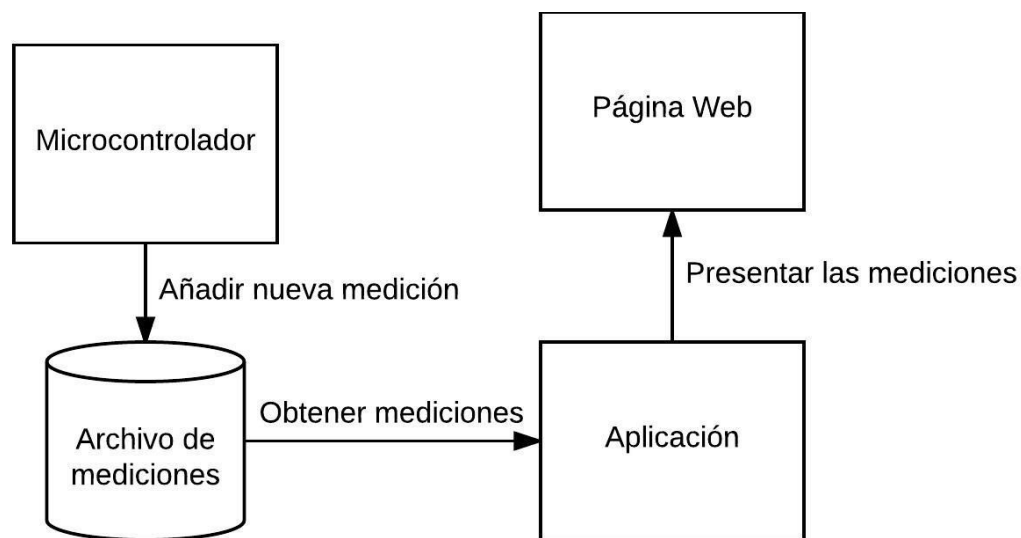


Fig. 2. Diagrama de flujo de la interacción entre el microcontrolador y la aplicación.

En el simulador, los datos se generan mediante un `while` infinito que introduce números aleatorios en el archivo de texto y espera una cantidad de segundos determinada mediante la secuencia `sleep()`.

Enunciado 4)

Agregar a la simulación anterior la posibilidad de que el usuario elija entre un conjunto predefinido de períodos de muestreo (ej: 1, 2, 5, 10, 30, 60 segundos). Identifique los cambios a nivel de HTML, de HTTP y de la simulación misma.

Resolución ejercicio 4:

Al inciso anterior se le añadió un formulario en el cual se deberán ingresar valores entre 1 y 60. Este campo modifica la frecuencia de obtención de datos por parte de los sensores y de actualización de la interfaz web.

La Fig. 3. permite observar que, luego de los cambios realizados al inciso anterior, la página web puede indicarle a la aplicación cual es la nueva frecuencia, y esta la almacenará en un archivo para que el microcontrolador pueda obtener la nueva frecuencia de muestreo.

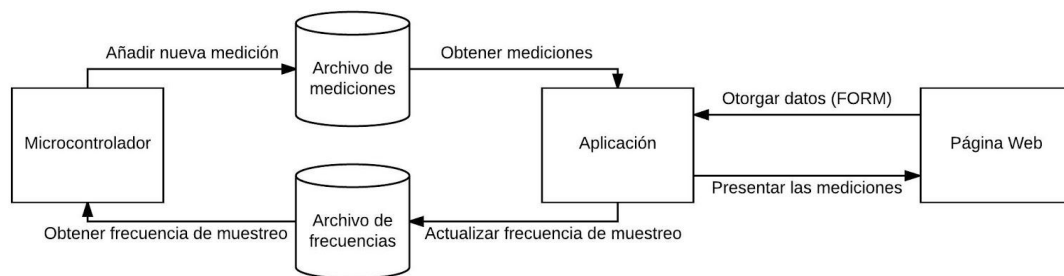


Fig. 3. Diagrama de flujo de la interacción entre el microcontrolador, la aplicación y la página web.

Enunciado 5)

Comente la problemática de la concurrencia de la simulación y específicamente al agregar la posibilidad de cambiar el período de muestreo. Comente lo que estima que podría suceder en el ambiente real ¿Podrían producirse problemas de concurrencia muy difíciles o imposibles de simular? Comente brevemente los posibles problemas de tiempo real que podrían producirse en general.

Resolución ejercicio 5:

En esencia, el problema del productor-consumidor es un ejemplo clásico de problema de sincronización de multiprocesos. El programa describe dos procesos, productor y consumidor, ambos comparten un buffer de tamaño finito. La tarea del productor es generar un producto, almacenarlo y comenzar nuevamente; mientras que el consumidor toma (simultáneamente) productos uno a uno. El problema consiste en que el productor no añada más productos que la capacidad del buffer y que el consumidor no intente tomar un producto si el buffer está vacío.

En este caso, los problemas de concurrencia son similares. Uno de los problemas es que el consumidor intente obtener un dato antes de que se haya generado uno nuevo, y el otro, que el productor produzca un dato luego de que el consumidor haya tomado uno.

Para solucionar estos problemas se utilizan mecanismos para sincronizar los procesos.

Enunciado 6)

¿Qué diferencias supone que habrá entre la simulación planteada y el sistema real? Es importante para planificar un conjunto de experimentos que sean significativos a la hora de incluir los elementos reales del sistema completo.

Resolución ejercicio 6:

Existen demasiadas variables que tienen que ser contempladas en un sistema real, tales como alimentación de los dispositivos, medios de comunicación, problemas de ruido, interferencia y distorsión, delay de la comunicación, latencia, etc. La simulación planteada es una aproximación al sistema real pero considerando que esté fuera perfecto. A la hora de crear un sistema completo real se deben considerar soluciones a los distintos tipos de problemas que puedan afectar su ejecución. Deben planificarse experimentos para comprobar que el sistema responderá correctamente antes cualquier problema, ya que si se plantea, por ejemplo, un delay en la comunicación, se deberá esperar que los datos estén listos antes de que sean leídos.