

# Sistemas Distribuidos – Práctica 1

Matteo Simone Gagliano – AY6962165 – Estudiante en Erasmus

Componentes desarrollados:

- **WebServer:** en esto package hay dos componentes, el server *HTTPwebServer.py* y *controllerHTTPwebServer.py* que es un módulo llamado desde *HTTPwebServer.py*
- **GatewayServer:** en esto package hay dos componentes, el server *GatewayServer.py* y *controllerGatewayServer.py* que es un módulo llamado desde *GatewayServer.py*
- **Procesador1:** en esto package hay dos componentes, el procesador1 *Processor.py* y *controllerProcessor.py* que es un módulo llamado desde *Processor.py*
- **Procesador2:** en esto package hay dos componentes, el procesador2 *Processor.py* y *controllerProcessor.py* que es un módulo llamado desde *Processor.py*
- **Procesador3:** en esto package hay dos componentes, el procesador3 *Processor.py* y *controllerProcessor.py* que es un módulo llamado desde *Processor.py*
- **utils:** en esto package hay *Exception.py* que sirve para manejar las excepciones, *inputOutput.py* que sirve para hacer operaciones sobre files y manejar conexiones, y *stringManager.py* que sirve para hacer operaciones sobre las cadenas. Todos los controllers lo utilizan.

Además, el WebServer tiene dos páginas web estáticas: *index.html* y *error.html*

Cuando un error ocurre entre cliente y web server, esto devuelve *error.html* modificada oportunamente con el código de respuesta HTTP.

En el WebServer hay una carpeta '*Frontend\_Jquery\_Bootstrap*' donde están un archivo de *Jquery* y los modulos de *Bootstrap*. *Jquery* es utilizado para enviar una *Ajax request* cuando el cliente pulsa el botón 'PAY'; mientras *Bootstrap* es utilizado para el desarrollo de la interfaz (CSS).

El Gateway tiene también cuatro páginas HTML que va a modificar con nuevo contenido HTML cada vez que tiene que devolverlas al web server (el contenido depende desde las operaciones

efectuadas). Las paginas web son: *authResponse.html*, *error.html*, *info.html*, y *parametersResponse.html*.

El Gateway también tiene una carpeta (*processorsMappingAndAddresses*) con dos archivos, *Bines.txt* y *Procesadores.txt*.

Cada procesador tiene un archivo *config.txt* donde se guardan los parámetros del procesador. Esto archivo se modifica cuando se hacen peticiones con parámetro 'set'.

Nota sobre el GatewayServer: el Gateway se conecte a los procesadores solo si recibe una petición que necesita que se llaman los procesadores; entonces, cuando se lanza el programa *GatewayServer.py*, el Gateway no se conecte instantáneamente a los procesadores, se conecte solo cuando recibe un mensaje desde el web server. Además, para que la conexión con los procesadores funcione correctamente, hay que poner en el archivo *Procesadores.txt* las direcciones de los procesadores segundo el formato *numero#ip#puerto*, por ejemplo:

1#127.0.0.1#20001

2#127.0.0.1#20002

3#127.0.0.1#20003

y dejando una línea (pulsar enter en el teclado) después que se ha escrito la última dirección.

Nota sobre los procesadores: en el archivo *Bines.txt* está el mapping entre el primer número de la tarjeta y el procesador que puede evaluar la autorización de pago. Entonces, para consentir que todas las tarjetas pueden solicitar autorizaciones hay que lanzar todos los procesadores (1, 2 y 3). Sin embargo, si el procesador delineado para esa tarjeta está lanzado y tiene *status=off*, la autorización no se puede evaluar porque el procesador está 'apagado' (y al cliente se devuelve la información).

Nota sobre los archivos *\_\_init\_\_.py*: estos archivos sirven para manejar las carpetas como packages, así que se pueden llamar componentes guardados en diferentes carpetas.

## Guía de despliegue

El proyecto se ha desarrollado en **Python 3.8**, por eso cada ordenador debe tener instalado esta versión de Python.

En <https://www.python.org/downloads/release/python-380/> puedes descargar Python 3.8 y instalarlo como prefieres. La instalación a través de lo file executable en Windows es muy sencilla.

Para lanzar el código del web server tenemos que abrir una console y ponernos en la carpeta *WebServer*. Si estas utilizando Linux tienes que escribir *python3 HTTPwebServer.py*. Sin embargo, si estas utilizando Windows tienes que escribir *python HTTPwebServer.py*. El programa va a preguntar el puerto de escucha, el IP del Gateway y su puerto de escucha.

Para lanzar el código del Gateway tenemos que abrir una console y ponernos en la carpeta *GatewayServer*. Si estas utilizando Linux tienes que escribir *python3 GatewayServer.py*. Sin

embargo, si estas utilizando Windows tienes que escribir *python GatewayServer.py*. El programa va a preguntar el puerto de escucha.

Para lanzar el código de cada procesador tenemos que abrir una console y ponernos en la carpeta ProcesadorX (por ejemplo, Procesador1). Si estas utilizando Linux tienes que escribir *python3 Processor.py*. Sin embargo, si estas utilizando Windows tienes que escribir *python Processor.py*. El programa va a preguntar el puerto de escucha.

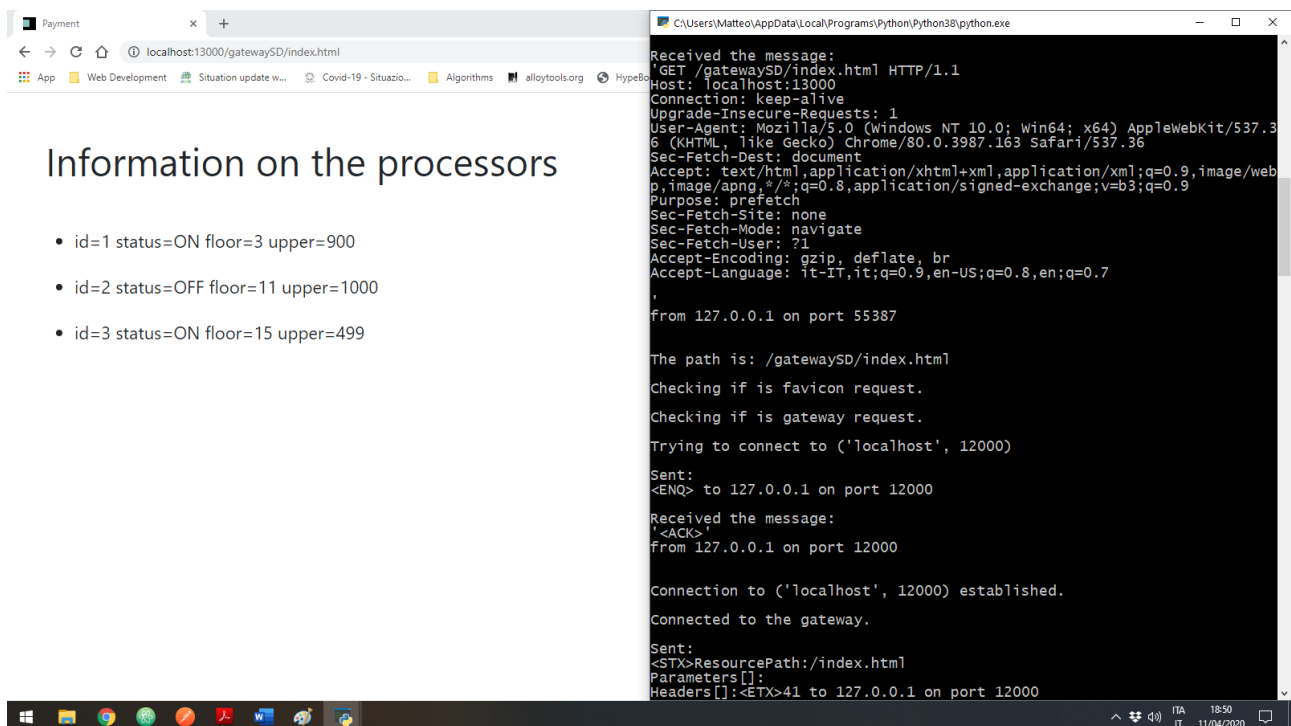
Nota sobre el nombre de la directory: No cambiar el nombre de la carpeta principal, *distributedSystemProject*. Esto porque si lo hace el programa no puede encontrar los modulos que yo he implementado y que se necesitan para que el programa funcione correctamente.

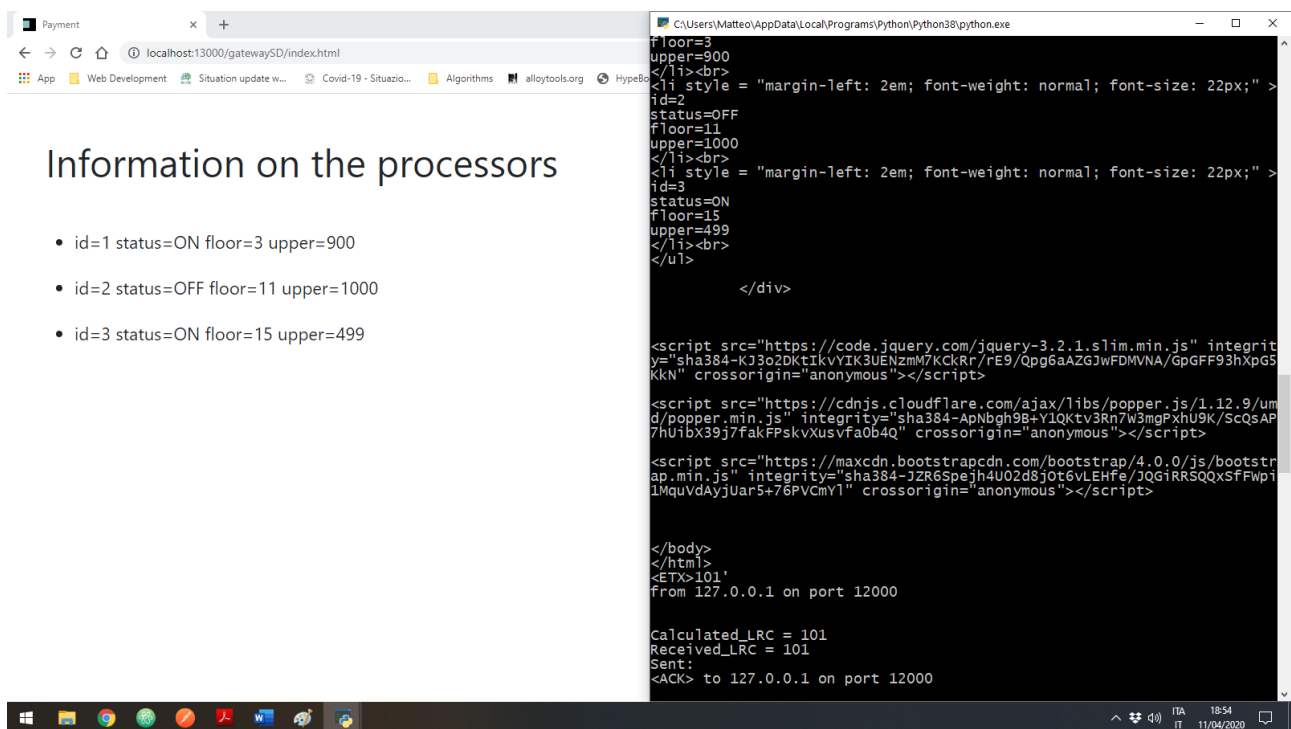
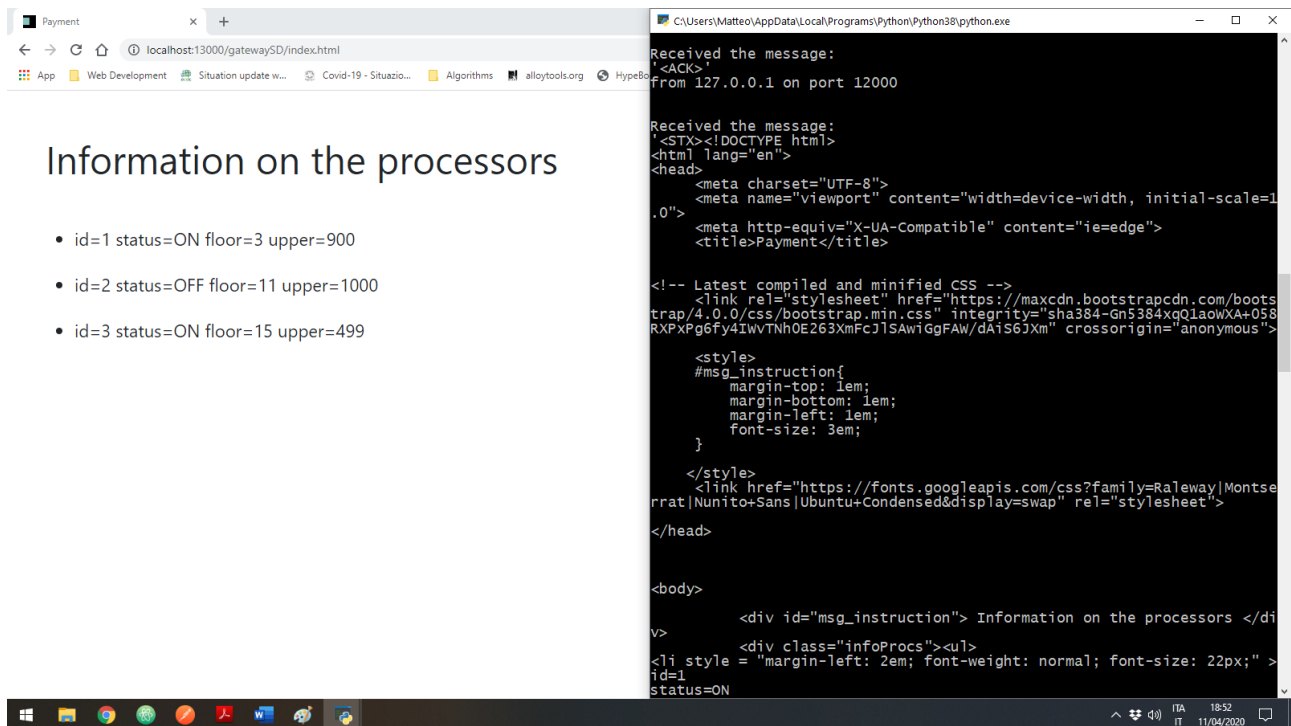
## Ejemplos de peticiones

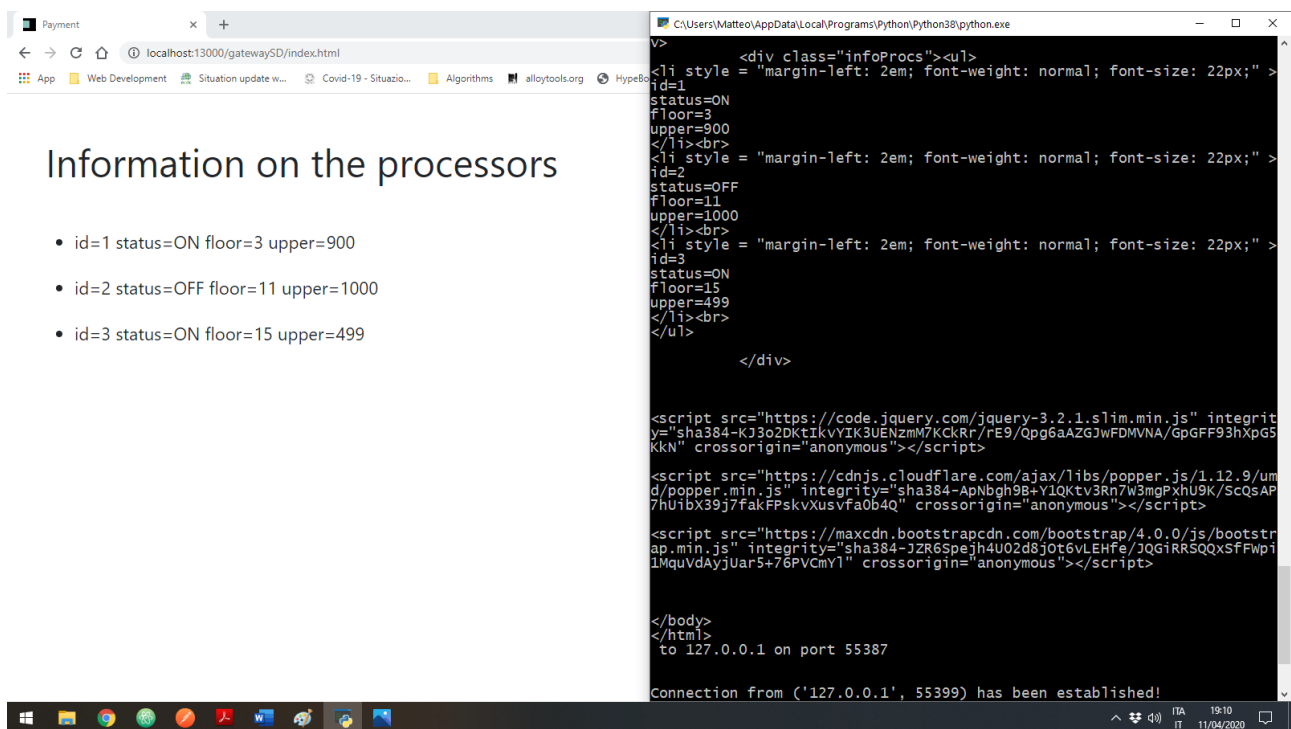
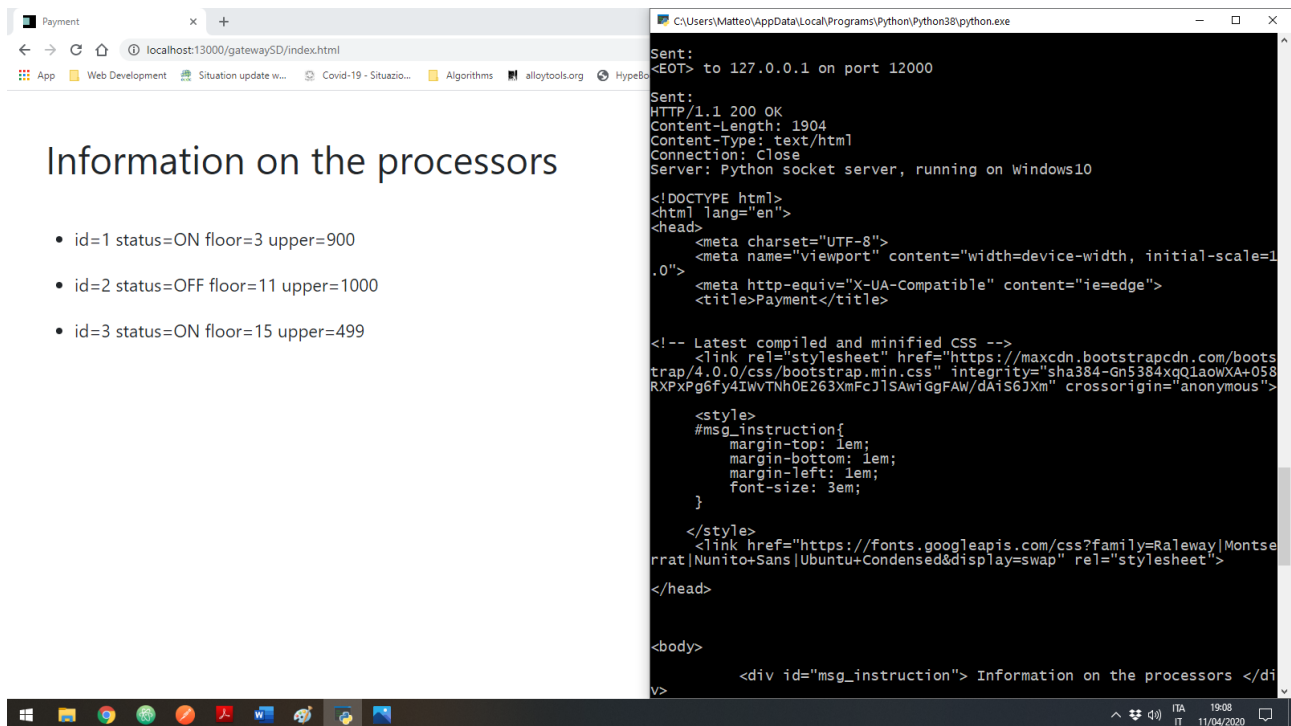
- <http://localhost:13000>
- <http://localhost:13000/gatewaySD/index.html>
- <http://localhost:13000/gatewaySD/ul?proc=2>
- <http://localhost:13000/gatewaySD/fl?proc=3>
- <http://localhost:13000/gatewaySD/ul?proc=3&set=1600>
- <http://localhost:13000/gatewaySD/fl?proc=1&set=8>
- <http://localhost:13000/gatewaySD/status?proc=2>
- <http://localhost:13000/gatewaySD/status?proc=1&set=OFF>

## Screenshot

- Screenshots que hacen ver lo que pasa en el WebServer cuando llega una petición con el url <http://localhost:13000/gatewaySD/index.html>







- Abajo hay screenshots que muestran lo que pasa en el GatewayServer cuando el url en el browser es `http://localhost:13000/gatewaySD/fl?proc=3`

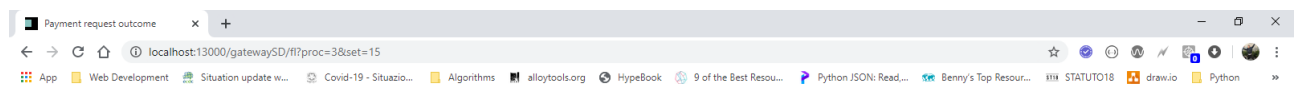
```
C:\Users\Matteo\AppData\Local\Programs\Python\Python38\python.exe
Received the message:
{<STX>ResourcePath:/f1
Parameters[]:proc=3
Headers[]:<ETX>110'
from 127.0.0.1 on port 54447

lrcCalculated = 110
lrcReceived = 110
Sent:
<ACK> to 127.0.0.1 on port 54447
dictionary is {'proc': '3'}
getting address of processor: 3
Read file:
1#127.0.0.1#20001
2#127.0.0.1#20002
3#127.0.0.1#20003

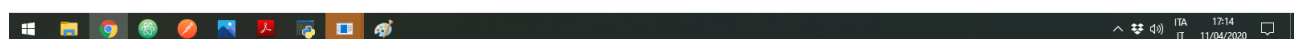
Address: 3#127.0.0.1#20003
Trying to connect to ('127.0.0.1', 20003)
Sent:
<ENQ> to 127.0.0.1 on port 20003
Received the message:
{<ACK>
from 127.0.0.1 on port 20003

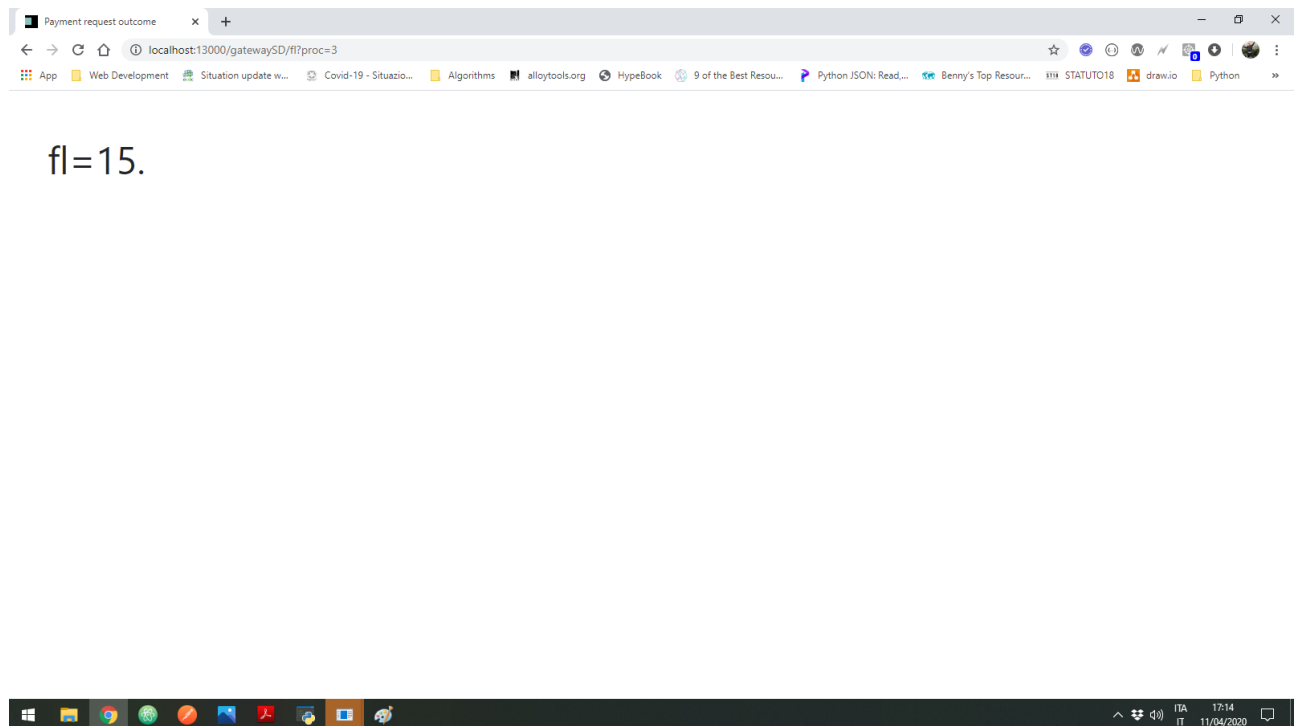
Connection to ('127.0.0.1', 20003) established.
Sent:
<STX>ResourcePath:/f1
Parameters[]:
Headers[]:<ETX>110 to 127.0.0.1 on port 20003
Received the message:
{<ACK>
from 127.0.0.1 on port 20003
```

- Otros screenshots que muestran lo que se devuelve al cliente cuando el quiere obtener un parámetro o cambiar el valor de eso

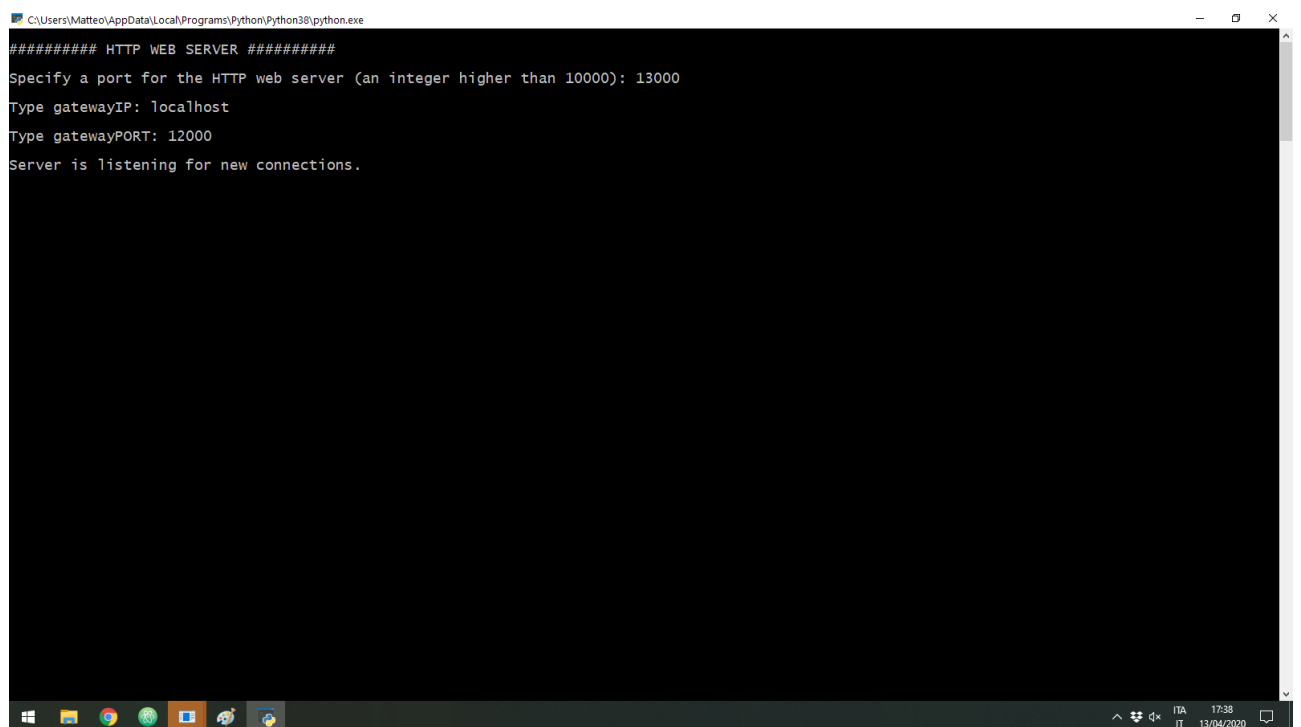


Parameter floor has been updated with the value 15.

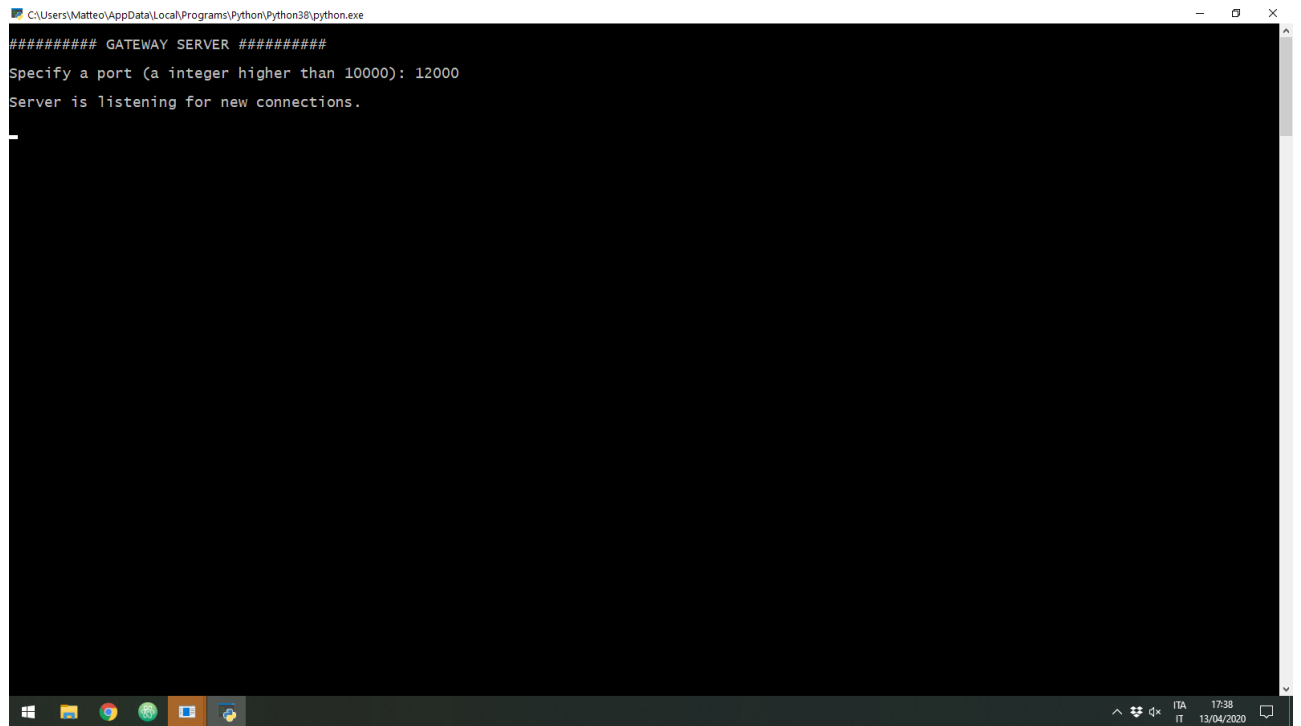




- WebServer start up screenshot



- GatewayServer start up screenshot:

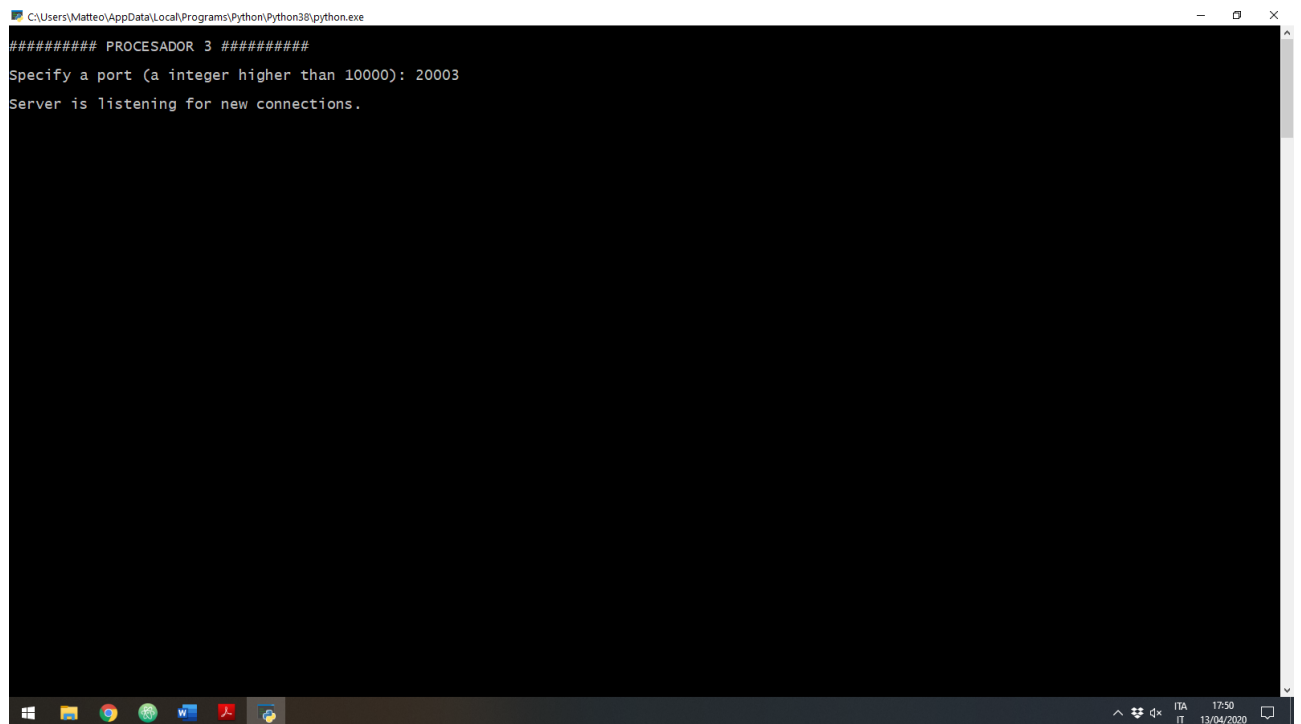


A screenshot of a Windows command prompt window. The title bar shows the file path: C:\Users\Matteo\AppData\Local\Programs\Python\Python38\python.exe. The window has a black background with white text. The text displayed is:   
##### GATEWAY SERVER #####  
Specify a port (a integer higher than 10000): 12000  
Server is listening for new connections.  
The window is open on a Windows 10 desktop. The taskbar at the bottom shows icons for the Start menu, File Explorer, Google Chrome, Microsoft Edge, and a few other applications. The system tray on the right shows the date and time: 17:38 on 13/04/2020.

```
C:\Users\Matteo\AppData\Local\Programs\Python\Python38\python.exe

##### GATEWAY SERVER #####
Specify a port (a integer higher than 10000): 12000
Server is listening for new connections.
```

- Processor start up screenshot:



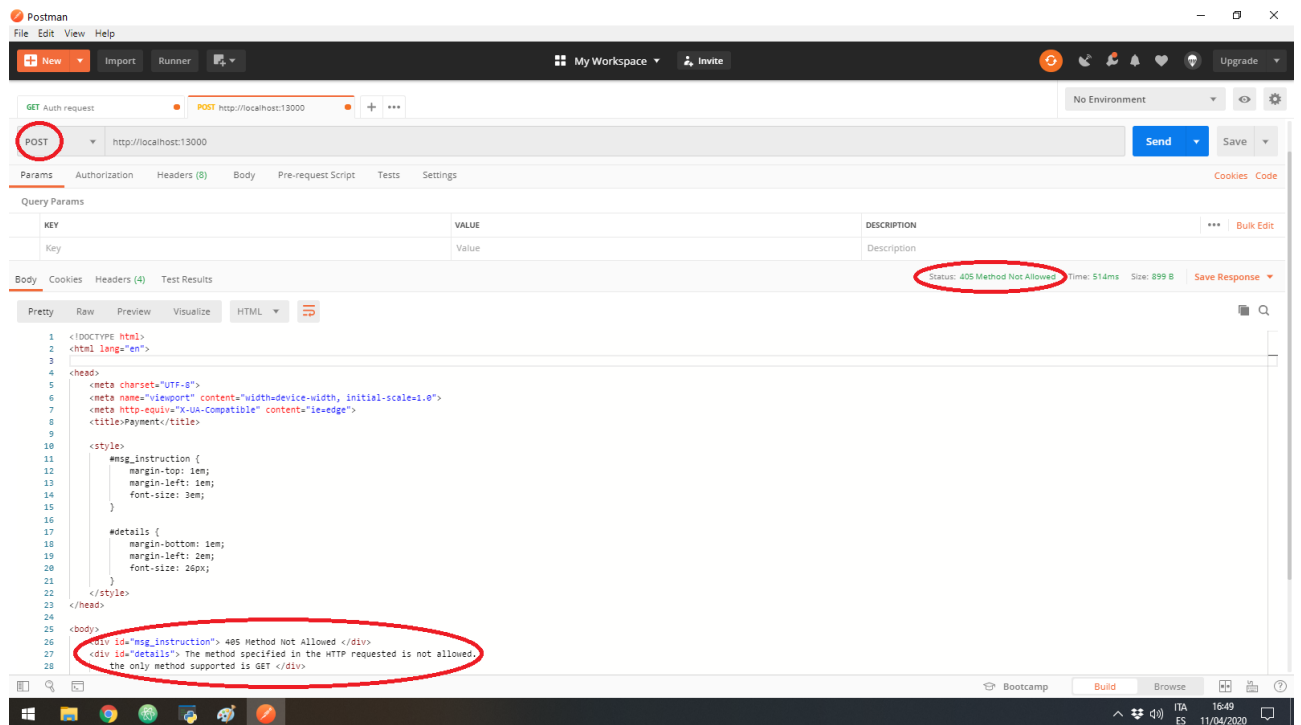
A screenshot of a Windows command prompt window, similar to the one above. The title bar shows the file path: C:\Users\Matteo\AppData\Local\Programs\Python\Python38\python.exe. The window has a black background with white text. The text displayed is:   
##### PROCESADOR 3 #####  
Specify a port (a integer higher than 10000): 20003  
Server is listening for new connections.  
The window is open on a Windows 10 desktop. The taskbar at the bottom shows icons for the Start menu, File Explorer, Google Chrome, Microsoft Edge, Word, and a few other applications. The system tray on the right shows the date and time: 17:50 on 13/04/2020.

```
C:\Users\Matteo\AppData\Local\Programs\Python\Python38\python.exe

##### PROCESADOR 3 #####
Specify a port (a integer higher than 10000): 20003
Server is listening for new connections.
```

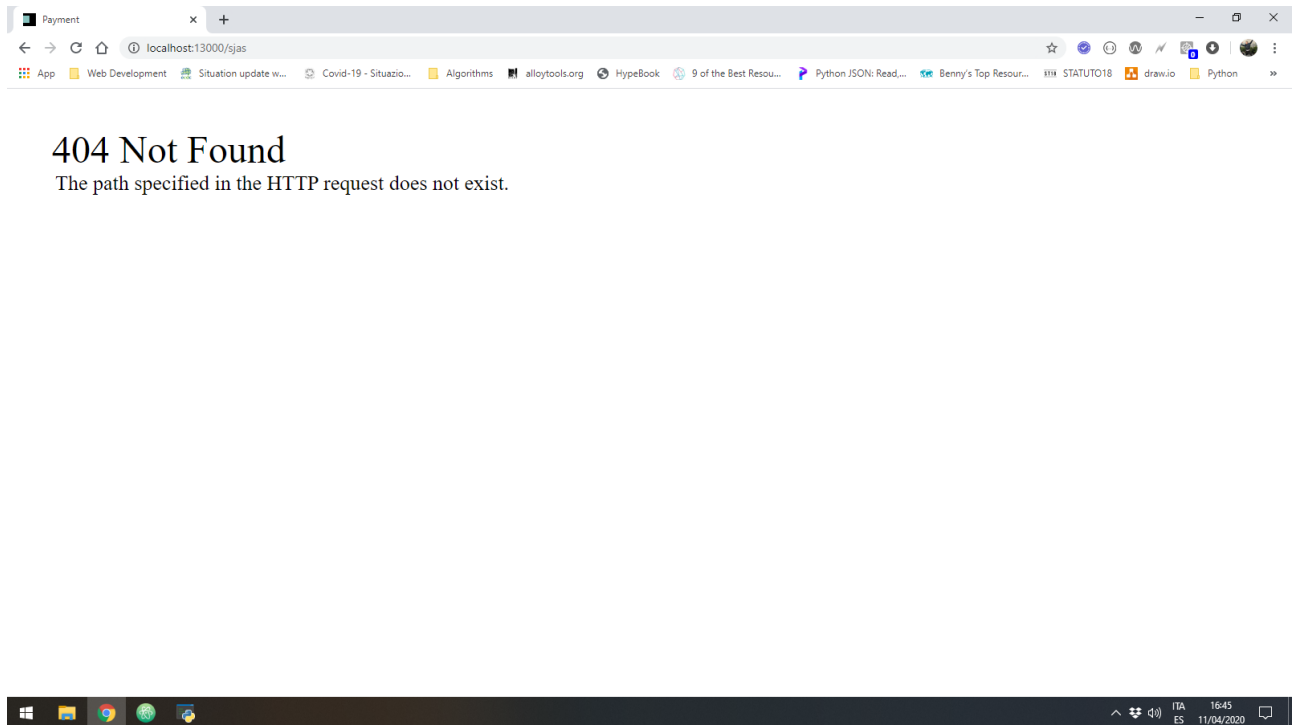


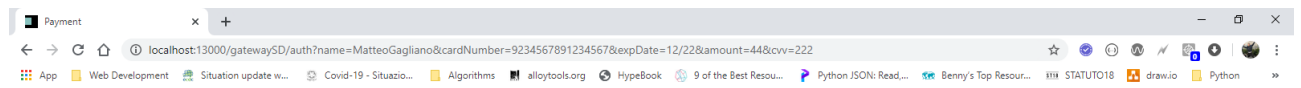
- Páginas de errores



## 404 Not Found

The path specified in the HTTP request does not exist.

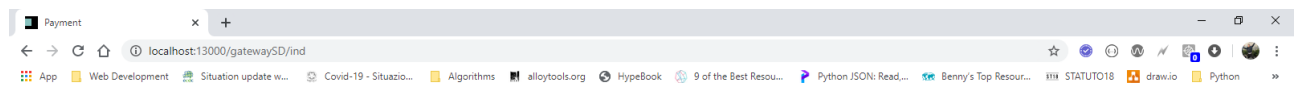




## 409 Conflict

The gateway server cannot connect to the processor properly. Try again later.

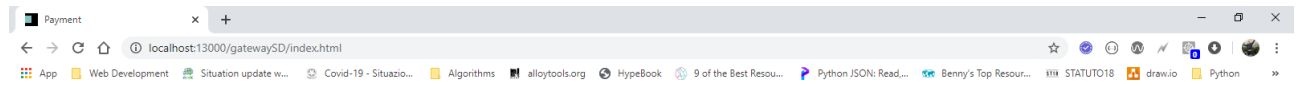
Error details: impossible to connect through the socket system to the address: 127.0.0.1 on port: 20003



## 404 Not Found

The path specified in the HTTP request does not exist.

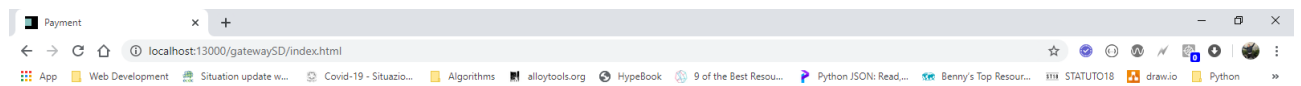
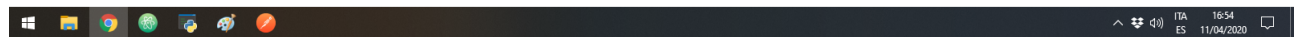




# 409 Conflict

The gateway server cannot connect to the processor properly. Try again later.

Error details: All the processors cannot be reached.

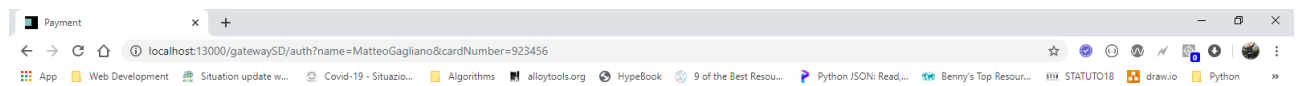
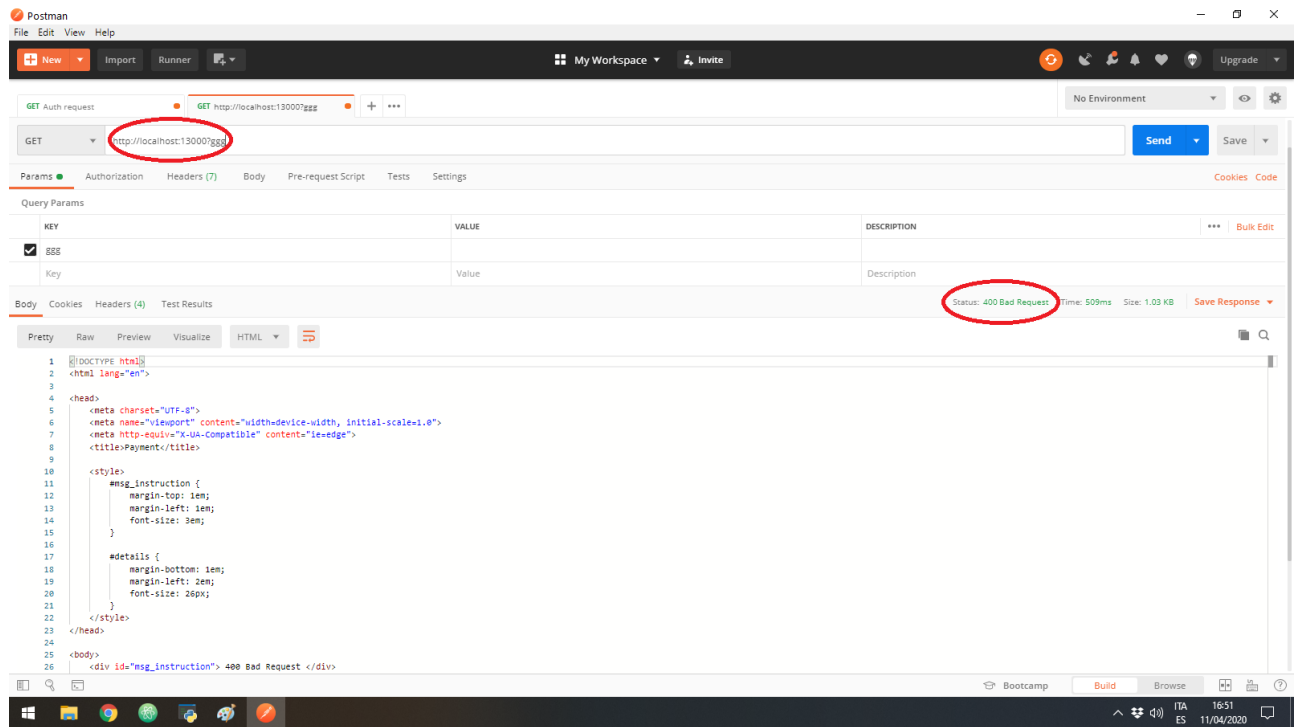


## Information on the processors

- id=1 status=ON floor=3 upper=900
- id=3 status=ON floor=11 upper=499

The following processors could not be reached: [('127.0.0.1', 20002)]





## 400 Bad Request

The server cannot process your request. If it's an auth request, check that all the parameters: name, cardNumber, cvv, expDate and amount are present in the url.

