

RELAZIONE

Progetto di Programmazione di Reti a.a. 2020-21

Traccia 2: Python Web Server

Marsild Spahiu

0000916048

marsild.spahiu@studio.unibo.it

| | |
|--------------------------------|----------|
| Introduzione | 3 |
| Descrizione | 3 |
| Dettagli implementativi | 5 |
| Multithreading | 6 |
| Servizi | 6 |
| Interruzione da tastiera | 6 |
| Download PDF | 7 |
| Autenticazione | 7 |
| Guida utente | 8 |
| Download | 9 |

Introduzione

Il progetto consiste nella realizzazione di un Web Server in Python per una azienda ospedaliera.

I requisiti da rispettare (definiti dalla traccia) sono i seguenti:

1. Il web server deve consentire l'accesso a più utenti in contemporanea.
2. La pagina iniziale deve consentire di visualizzare la lista dei servizi erogati dall'azienda ospedaliera e per ogni servizio avere un link di riferimento ad una pagina dedicata.
3. L'interruzione da tastiera (o da console) dell'esecuzione del web server deve essere opportunamente gestita in modo da liberare la risorsa socket.
4. Nella pagina principale dovrà anche essere presente un link per il download di un file pdf da parte del browser.
5. Come requisito facoltativo si chiede di autenticare gli utenti nella fase iniziale della connessione.

Descrizione

Dopo aver avviato il server da linea di comando, essersi autenticati ed aver raggiunto il Web Server dal proprio motore di ricerca (vedere [Guida utente](#) per informazioni aggiuntive), il sito si presenta nel seguente modo:

Servizi Ospedalieri Spahiu

[Home](#) [118](#) [Pronto soccorso](#) [Medici e Pediatri di famiglia](#) [Guardia Medica \(ex\)](#) [Farmacie di turno](#) [FSE](#) [COVID-19](#) [Download info](#) [Aggiorna contenuti](#)



Servizi Ospedalieri
Servizi AUSL della Romagna



L'Azienda Unità Sanitaria Locale della Romagna ([AUSL Romagna](#)), istituita con Legge regionale n. 22 del 21 novembre 2013, è l'ente strumentale attraverso il quale la Regione assicura i livelli essenziali ed uniformi di assistenza dell'ambito territoriale della Romagna. L'Azienda è dotata di personalità giuridica pubblica e di autonomia imprenditoriale ai sensi delle disposizioni legislative nazionali e regionali di regolamentazione del Servizio Sanitario Regionale.

Attraverso la barra di navigazione sovrastante, oppure attraverso i collegamenti riportati qui di seguito, potrai accedere ai principali servizi offerti. È anche possibile scaricare l'elenco di questi ultimi da "Download info" nella barra di navigazione.

[Passa al servizio 118](#)
[Passa al servizio Pronto Soccorso](#)
[Passa al servizio Medici e Pediatri di famiglia](#)
[Passa al servizio Continuità Assistenziale \(Guardia Medica\)](#)
[Passa al servizio Farmacie di turno](#)
[Passa al servizio FSE](#)

Per qualsiasi informazione, dubbio o segnalazione riguardante i Servizi Ospedalieri Spahiu [contattaci qui](#).

Figura 1: homepage del Web Server

La homepage dei Servizi Ospedalieri Spahiu ([figura 1](#), corrispondente all'index.html) mostra una breve descrizione del sito e degli argomenti trattati e, successivamente, come richiesto dal punto 2 della traccia (*"La pagina iniziale deve consentire di visualizzare la lista dei servizi erogati dall'azienda ospedaliera e per ogni servizio avere un link di riferimento ad una pagina dedicata"*) elenca in successione tutti i servizi offerti con collegamento alla propria pagina di riferimento.

Questi servizi, in ordine, sono:

1. 118
2. Pronto Soccorso
3. Medici e Pediatri di famiglia
4. Continuità Assistenziale
5. Farmacie di turno
6. FSE

Data l'emergenza sanitaria in corso, è stata inserita nella barra di navigazione anche un collegamento alla pagina dell'AUSL riguardante il 'Covid-19' e tutte le informazioni e servizi ad esso collegato (vaccini, tamponi, ecc...).

La struttura della homepage viene poi ripresa similmente da tutte le altre pagine dei singoli servizi i quali, oltre alla descrizione, presentano anche i collegamenti esterni necessari agli utenti per potersi informare maggiormente.

In fondo alla barra di navigazione sono anche presenti due pulsanti aggiuntivi, "Download info" e "Aggiorna contenuti" i quali, come preannuncia il nome, permettono rispettivamente di scaricare il file "info.pdf" (vedere la sezione [Download](#) per informazioni aggiuntive) e di aggiornare i contenuti delle pagine di servizio, qualora fosse necessario.

Dettagli implementativi

Per la realizzazione del Web Server si è deciso di svolgere il progetto in linguaggio Python (attraverso l'IDE Spyder, all'interno della distribuzione open source Anaconda), utilizzando anche il linguaggio HTML per la realizzazione delle pagine e dei loro contenuti.

Durante lo svolgimento del progetto è stato trovato necessario l'utilizzo delle seguenti librerie:

```
import sys
import signal
import http.server
import socketserver
import threading
```

sys: Questo modulo contiene una serie di funzioni e parametri che risultano molto utili ogni volta che il nostro programma deve interagire con il sistema operativo su cui si sta lavorando, oltre all'accesso ad alcune variabili usate o mantenute dall'interprete, e a funzioni che interagiscono fortemente con l'interprete stesso, come ad esempio la lettura degli argomenti forniti da linea di comando (`sys.argv[1]`) oppure la chiusura del programma (`sys.exit(0)`).

signal: Questo modulo fornisce funzioni, variabili e meccanismi atti ad usare gestori di segnali in Python. È stato utilizzato per la gestione dell'interruzione del programma da tastiera (vedere successiva [sottosezione](#)).

http.server: Questo modulo definisce le classi per implementare i server HTTP (server web).

socketserver: Il modulo socketserver semplifica il compito di scrivere server di rete: è stato utilizzato per la gestione di più richieste (vedere il suo utilizzo nella successiva sottosezione [Multithreading](#)).

threading: Questo modulo realizza un'interfaccia ad alto livello per i thread sulla base del modulo a basso livello thread. È stato utilizzato anche per gestire l'attesa senza busy waiting e per realizzare un thread per aggiornare periodicamente i contenuti delle pagine.

Segue una breve descrizione delle principali problematiche trattate (al fine di rispettare i requisiti della traccia) e delle loro soluzioni:

Multithreading

Come accennato in precedenza, per soddisfare il requisito di “consentire l’accesso a più utenti in contemporanea” si è ricorsi all’utilizzo del modulo `socketserver`, impiegato nel seguente modo:

```
# ThreadingTCPServer per gestire più richieste
server = socketserver.ThreadingTCPServer(('127.0.0.1',port), ServerHandler)
```

dove ‘port’ rappresenta il valore di porta inserito da linea di comando.

Servizi

La creazione delle singole pagine di servizio è stata gestita dalla definizione del seguente metodo:

```
#metodo lanciato per la creazione delle pagine servizi
def create_page_servizio(title,file_html, end_page):
    f = open(file_html,'w', encoding="utf-8")
    try:
        message = header_html + title + navigation_bar + end_page + footer_html
    except:
        pass
    f.write(message)
    f.close()
```

In questo modo, chiamando questa funzione diverse volte quante sono le pagine da creare (e fornendo i parametri necessari), la loro definizione si semplifica notevolmente, considerando che “header_html”, “navigation_bar” e “footer_html” sono comuni a tutti i servizi.

Interruzione da tastiera

Per permettere l’uscita dal processo attraverso la sequenza Ctrl+C è stata definita la seguente funzione:

```
def signal_handler(signal, frame):
    print('Combinazione di interruzione(Ctrl+C)premuta:uscita dal server http.')
    try:
        if(server):
            server.server_close()
    finally:
        # fermo il thread del refresh senza busy waiting
        waiting_refresh.set()
        sys.exit(0)
```

Questa funzione viene chiamata nel main dalla seguente linea di codice:

```
#interrompe l'esecuzione se da tastiera arriva la sequenza(CTRL+C)  
signal.signal(signal.SIGINT, signal_handler)
```

dove signal.SIGINT rappresenta appunto la sequenza Ctrl + C.

In questo modo si garantisce l'interruzione dell'esecuzione del web server e la corretta liberazione delle risorse socket utilizzate.

Download PDF

Il download del file "info.pdf" è stato gestito direttamente all'interno del codice HTML della barra di navigazione nel seguente modo:

```
<a href="http://127.0.0.1:{port}/info.pdf" download="info.pdf"  
style="float: right">Download info</a>
```

Autenticazione

L'autenticazione, richiesta dalla traccia al punto 5 dei requisiti, avviene da linea di comando nella parte iniziale di connessione ed è stata gestita attraverso le seguenti linee di codice:

```
#variabile utilizzata come controllo del while. Finchè è true, continua  
#richiedere username e password  
cond = True  
while cond:  
    try:  
        username = input("Inserire l'username: ") #richiesto l'usr da tastiera  
        password = input("Inserire la password: ") #richiesta la password  
    except:  
#Try-Except se Ctrl+C premuto durante l'inserimento delle credenziali  
        print("\nErrore nell'inserimento delle credenziali.")  
        print("Chiusura del programma.")  
        server.server_close() #Per evitare errori al prossimo avvio  
        sys.exit(0)  
    if username != 'admin' or password != 'admin':  
        print("\nUsername e/o password incorretta. Ritenta")  
    else:  
        cond = False #usr e psw corrette, cond=False per uscire dal while  
print("\nAccesso eseguito correttamente.\n")
```

Come espresso anche dai commenti accanto al codice, all'utente viene chiesto di identificarsi attraverso un username ed una password. Finché non vengono inseriti correttamente ('admin' e 'admin') essi continuano ad essere richiesti (ciclo while). Nel caso l'interruzione da tastiera avvenga durante questa fase, è stato inserito un Try ed Except che permette la corretta chiusura del programma e delle risorse utilizzate.

Guida utente

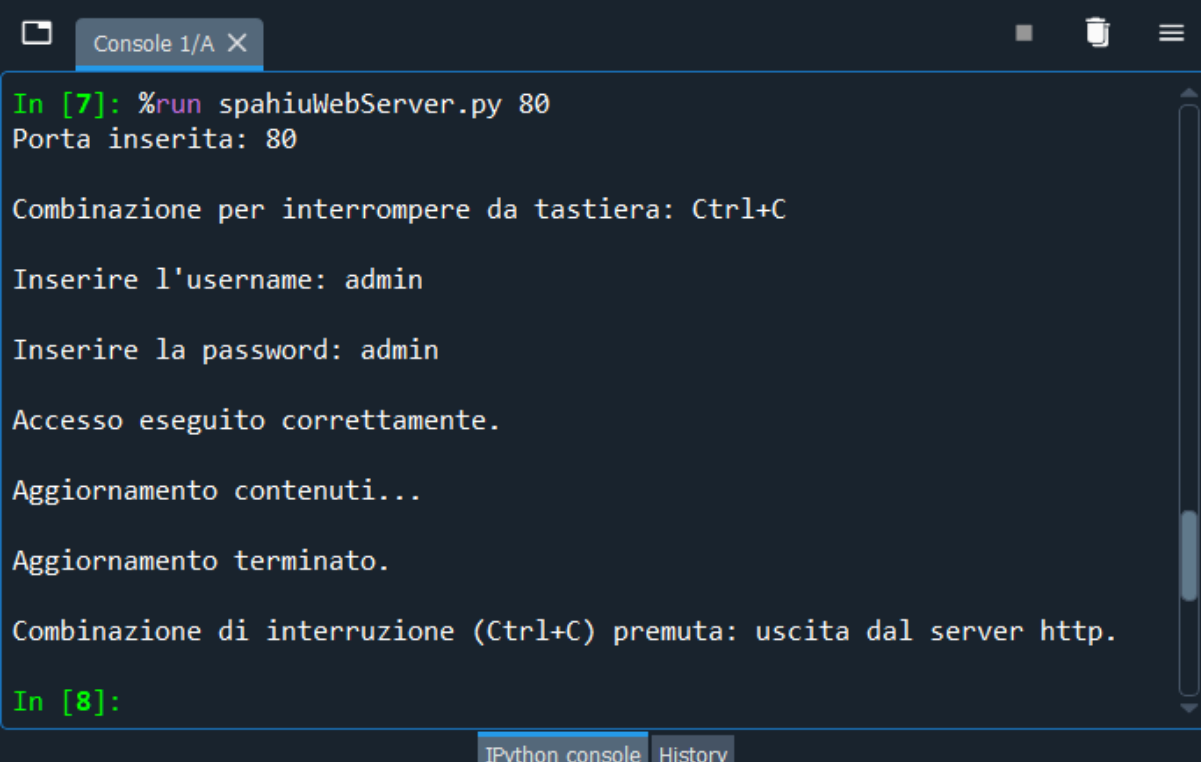
Una volta clonato il progetto oppure scaricata la directory, posizionarsi all'interno di quest'ultima e lanciare, da linea di comando (raccomandato l'utilizzo di Anaconda Powershell Prompt"), il comando:

```
python .\spahiuWebServer.py port
```

dove 'port' specifica il valore della porta da utilizzare, ad esempio 80. Nel caso non venga specificato questo valore, viene utilizzato il valore 8080 di default (in maniera analoga, se si vuole avere il codice a portata di mano, si può utilizzare direttamente la console di Spyder ricordandosi di utilizzare %run invece che python).

Successivamente, sempre da linea di comando, bisogna autenticarsi inserendo 'admin' sia come username che come password ed infine, dal proprio motore di ricerca preferito, è sufficiente recarsi all'indirizzo `http://localhost:port/index.html` (dove 'port' è il valore inserito in precedenza) per poter navigare nel Web Server.

Per chiudere il server da linea di comando bisogna premere contemporaneamente Ctrl + C (combinazione di chiusura), come viene riportato anche da console subito dopo l'autenticazione ([figura 2](#)).

The image shows a screenshot of the Spyder IPython console window. The title bar at the top reads "Console 1/A X". The console output is as follows:
In [7]: %run spahiuWebServer.py 80
Porta inserita: 80

Combinazione per interrompere da tastiera: Ctrl+C

Inserire l'username: admin

Inserire la password: admin

Accesso eseguito correttamente.

Aggiornamento contenuti...

Aggiornamento terminato.

Combinazione di interruzione (Ctrl+C) premuta: uscita dal server http.

In [8]:
At the bottom of the console window, there are two tabs: "IPython console" and "History".

Figura 2: snapshot della console di Spyder all'avvio (e chiusura) del Web Server

Download

Come richiesto dai requisiti della traccia al punto 4 (*"Nella pagina principale dovrà anche essere presente un link per il download di un file pdf da parte del browser"*), all'interno del Web-Server è presente, in fondo alla barra di navigazione, un pulsante etichettato come "Download info" il quale, una volta premuto, farà partire il download di un file PDF "info.pdf" contenente un'introduzione al Web Server ed ai relativi servizi offerti ([figura 3](#)).

vizi Ospedalieri Spahiu

ca (ex) Farmacie di turno FSE

COVID-19 ↗

Download info

Aggiorna contenuti

Servizi Ospedalieri

Servizi AUSL della Romagna



info.pdf ^



Servizi Ospedalieri Spahiu

Servizi AUSL della Romagna

L'Azienda Unità Sanitaria Locale della Romagna (AUSL Romagna), istituita con Legge regionale n. 22 del 21 novembre 2013, è l'ente strumentale attraverso il quale la Regione assicura i livelli essenziali ed uniformi di assistenza dell'ambito territoriale della Romagna. L'Azienda è dotata di personalità giuridica pubblica e di autonomia imprenditoriale ai sensi delle disposizioni legislative nazionali e regionali di regolamentazione del Servizio Sanitario Regionale.

Principali servizi offerti:

1. 118
2. Pronto Soccorso
3. Medici e Pediatri di famiglia
4. Continuità Assistenziale (ex Guardia Medica)
5. Farmacie di turno
6. FSE - Fascicolo Sanitario Elettronico

Figura 3: come scaricare ed aprire il file 'info.pdf'