



Laurea Triennale in Informatica - Università di Salerno  
Corso di *Ingegneria del Software* - Prof.ssa F. Ferrucci, Prof. F. Palomba

**CHEMO SMART**  
**SCHEDULER FOR CHEMOTERAPY**



# Manuale di Installazione ChemoSmart

Versione	1.0
Data	17/02/2023
Destinatario	Prof.ssa F. Ferrucci, Prof. F. Palomba
Presentato da	C. Troiano, M. Purice, L. Miranda, A. Nappi, G. Basile, C. De Palma
Approvato da	A. Bergamo, F. P. Ianuzziello



Laurea Triennale in Informatica - Università di Salerno  
Corso di *Ingegneria del Software* - Prof.ssa F. Ferrucci, Prof. F. Palomba

## Revision History

Data	Versione	Descrizione	Autori
17/02/2023	1.0	Prima stesura	LM, CT, CDP, GB, AN, MP



## Team members

Nome	Ruolo nel progetto	Acronimo	Informazioni di contatto
Alessandro Bergamo	Project Manager	AB	<a href="mailto:a.bergamo2@studenti.unisa.it">a.bergamo2@studenti.unisa.it</a>
Francesco Pio Iannuzziello	Project Manager	FPI	<a href="mailto:f.ianuzziello1@studenti.unisa.it">f.ianuzziello1@studenti.unisa.it</a>
Luigi Miranda	Team Member	LM	<a href="mailto:l.miranda11@studenti.unisa.it">l.miranda11@studenti.unisa.it</a>
Ciro Troiano	Team Member	CT	<a href="mailto:c.troiano17@studenti.unisa.it">c.troiano17@studenti.unisa.it</a>
Antonio Nappi	Team Member	AN	<a href="mailto:a.nappi47@studenti.unisa.it">a.nappi47@studenti.unisa.it</a>
Giuseppe Basile	Team Member	GB	<a href="mailto:g.basile36@studenti.unisa.it">g.basile36@studenti.unisa.it</a>
Mihail Purice	Team Member	MP	<a href="mailto:m.purice@studenti.unisa.it">m.purice@studenti.unisa.it</a>
Claudio De Palma	Team Member	CDP	<a href="mailto:c.depalma5@studenti.unisa.it">c.depalma5@studenti.unisa.it</a>



## Sommario

Revision History.....	2
Team members.....	3
Sommario .....	4
1 Introduzione.....	5
2 Avvio applicazione web.....	5
3 Avvio applicazione in locale.....	5
4 Prima dell'installazione .....	6
5 Installazione su Windows tramite script Python.....	8
5.1 Scaricare il progetto .....	8
5.2 Installazione dipendenze .....	11
5.3 Eseguire i microservizi .....	14
5.4 Terminare i microservizi.....	16
6 Installazione manuale per tutti i sistemi.....	16



# 1 Introduzione

---

Lo scopo di questo documento è di fornire una spiegazione dettagliata delle operazioni necessarie all'installazione e all'avvio del sistema "Chemosmart".

## 2 Avvio applicazione web

---

I requisiti essenziali per l'avvio del sistema Chemosmart sono l'utilizzo di un browser ed analogamente l'accesso alla rete internet.

## 3 Avvio applicazione in locale

---

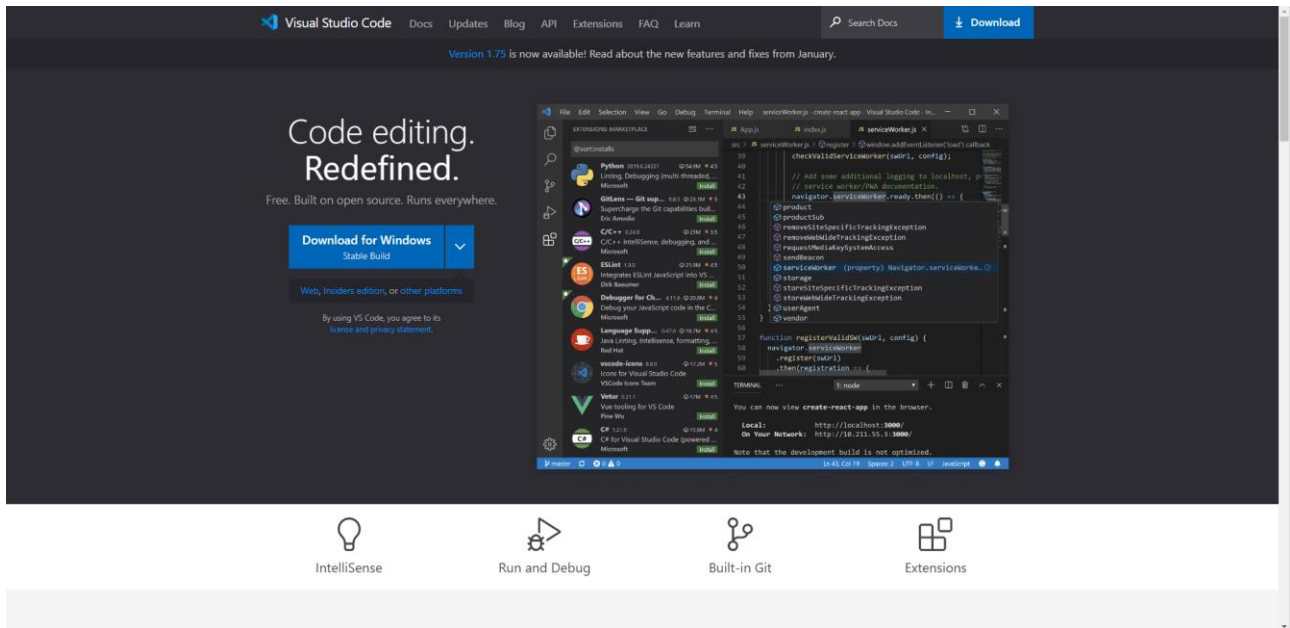
I requisiti per l'avvio dell'applicazione in locale sono:

- Node.js alla versione v18.12.0 o superiori
- Python alla versione più recente
- Visual studio code

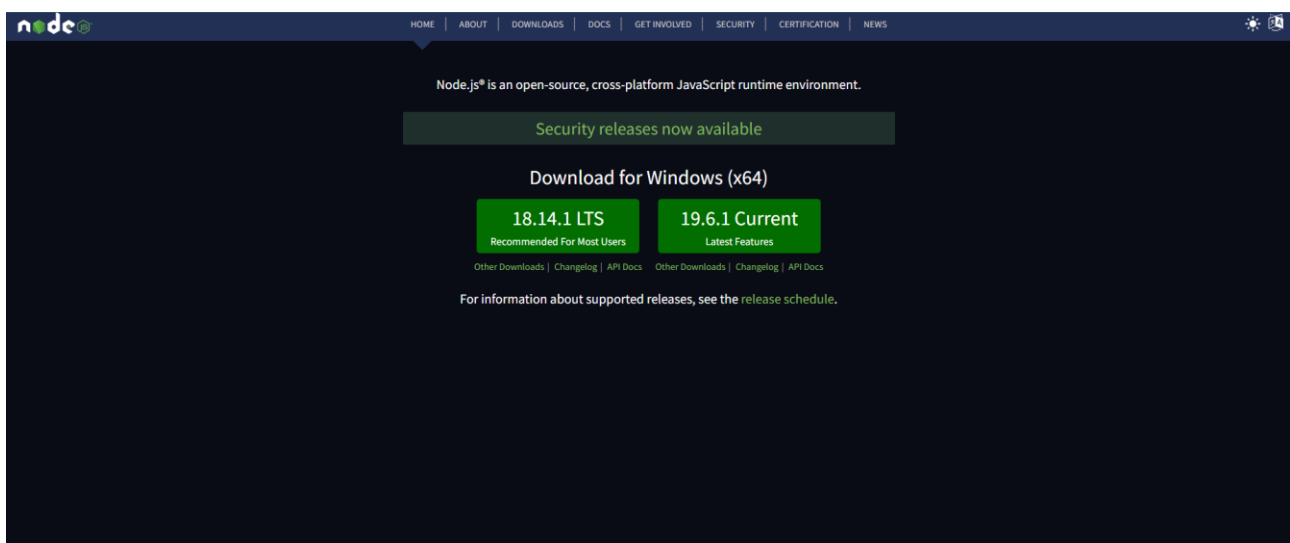


## 4 Prima dell'installazione

Link di installazione di Visual Studio Code: <https://code.visualstudio.com/>



Link per il download di node.js: <https://nodejs.org/en/>





Laurea Triennale in Informatica - Università di Salerno  
Corso di *Ingegneria del Software* - Prof.ssa F. Ferrucci, Prof. F. Palomba

Link per il download di python: <https://www.python.org/>

Python

PSF

Docs

PyPI

Jobs

Community

python™

Donate

Search

GO

Socialize

About Downloads Documentation Community Success Stories News Events

```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

**Functions Defined**

The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. [More about defining functions in Python 3](#)

1 2 3 4 5

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)

**Get Started**

Whether you're new to programming or an experienced developer, it's easy to learn and use Python.

Start with our [Beginner's Guide](#)

**Download**

Python source code and installers are available for download for all versions!

Latest: Python 3.11.2

**Docs**

Documentation for Python's standard library, along with tutorials and guides, are available online.

[docs.python.org](https://docs.python.org)

**Jobs**

Looking for work or have a Python related position that you're trying to hire for? Our [relaunched community-run job board](#) is the place to go.

[jobs.python.org](https://jobs.python.org)



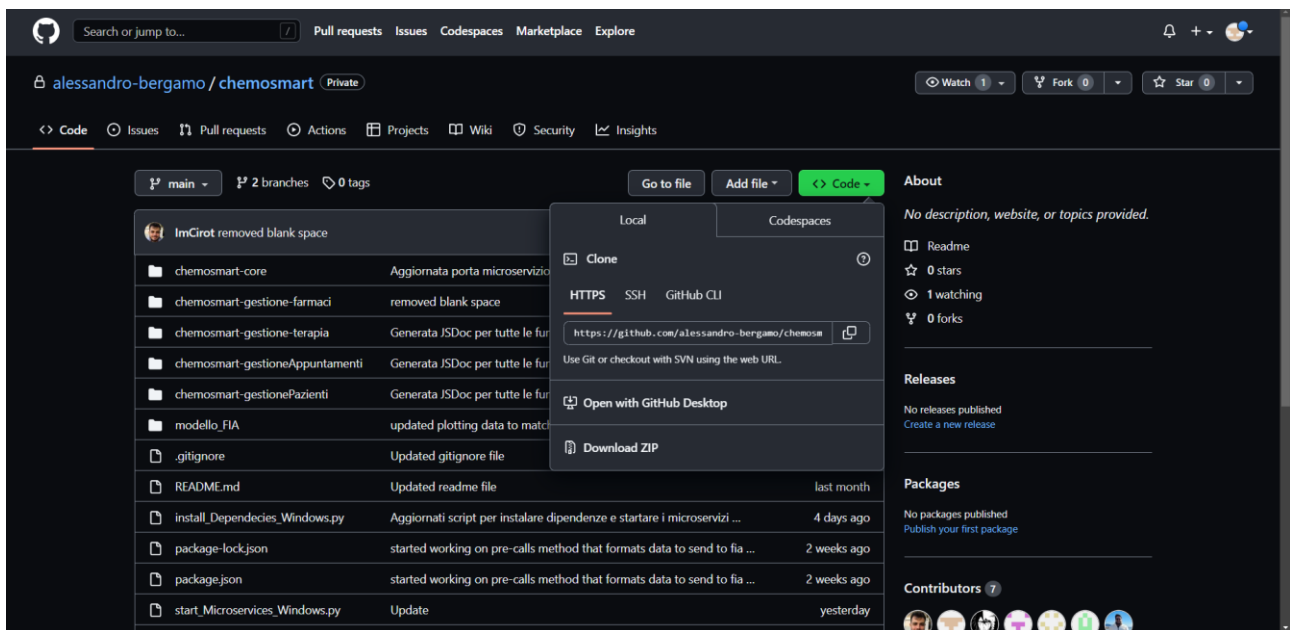
## 5 Installazione su Windows tramite script Python

### 5.1 Scaricare il progetto

Come prima operazione bisogna scaricare il progetto seguendo il link del repository su github:

<https://github.com/alessandro-bergamo/chemosmart>

Cliccare “Code” e nel menù a tendina cliccare “Download ZIP”.

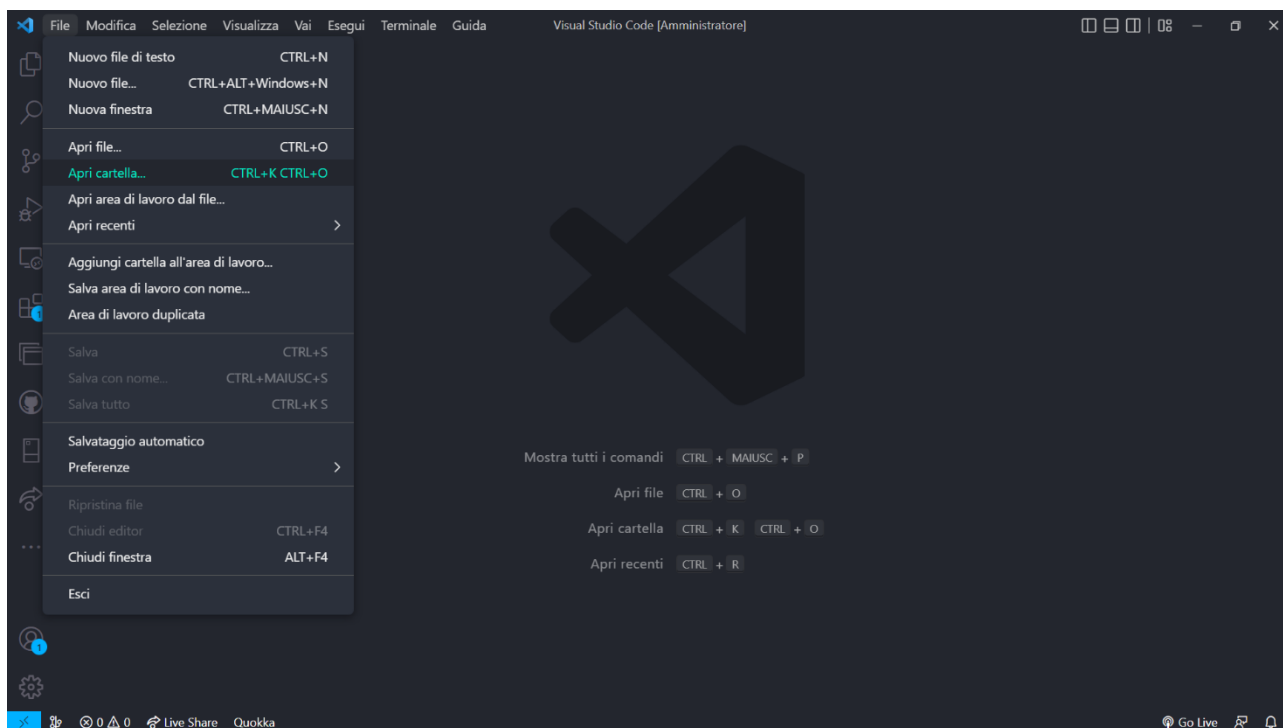


Estrarre il progetto in una qualsiasi directory del vostro sistema.



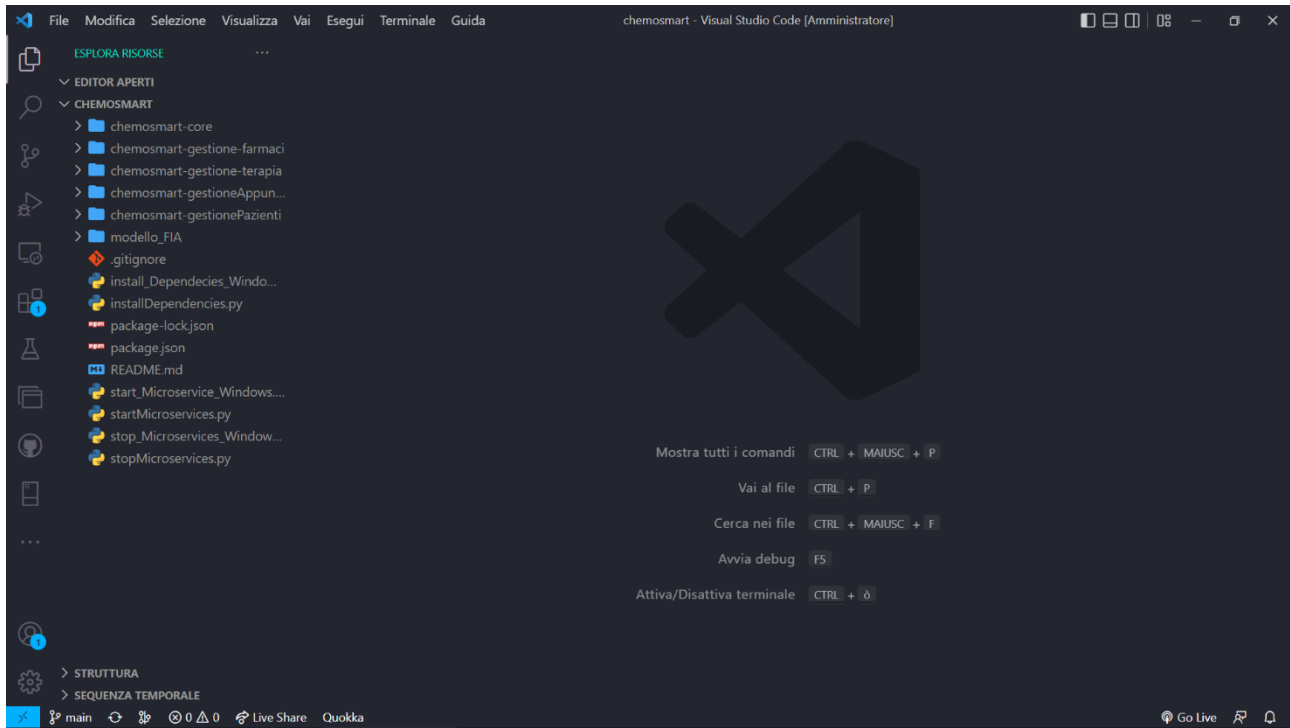


Eseguire Visual Studio Code, ed importare il progetto andando nella sezione “File”





Cliccare “Apri cartella...” e selezionare la cartella del repository scaricato.





## 5.2 Installazione dipendenze

Prima di eseguire il progetto vanno scaricate tutte le dipendenze necessarie ad ogni microservizio.

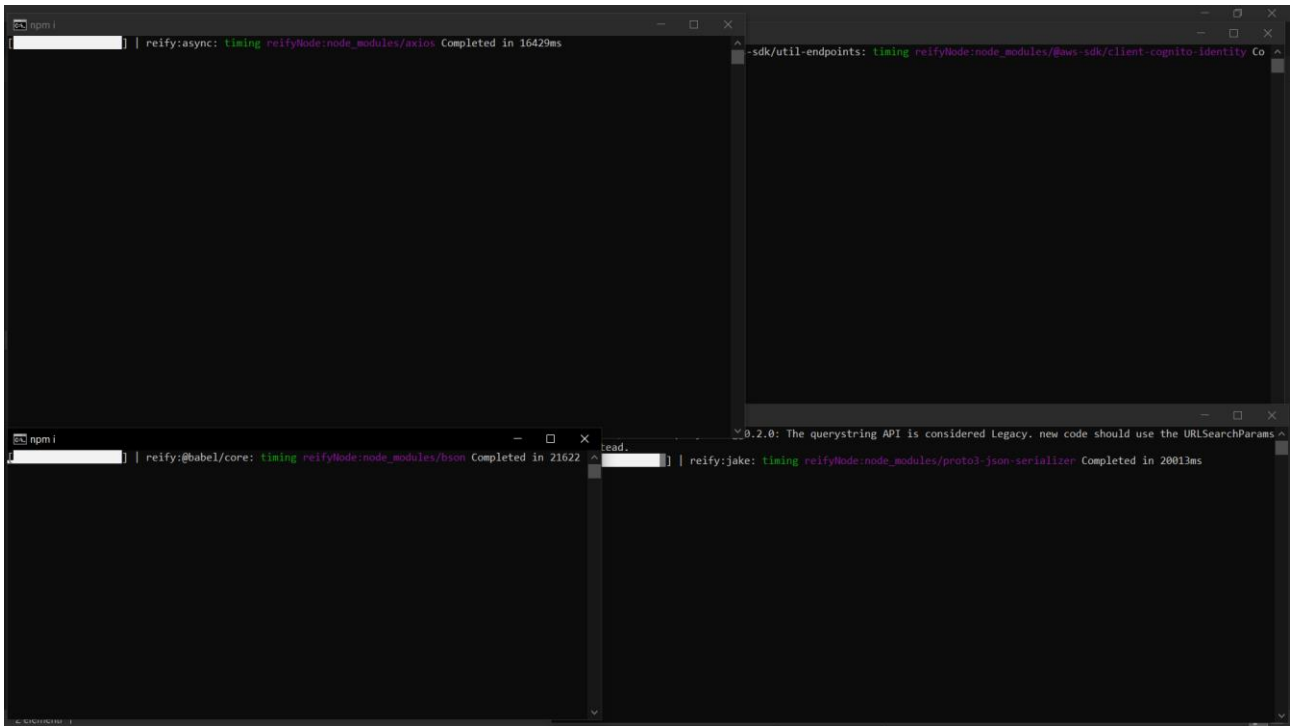
Aprire quindi il file “**install\_Dependencies\_Windows.py**” e cliccare in alto a destra il tasto per eseguire il file



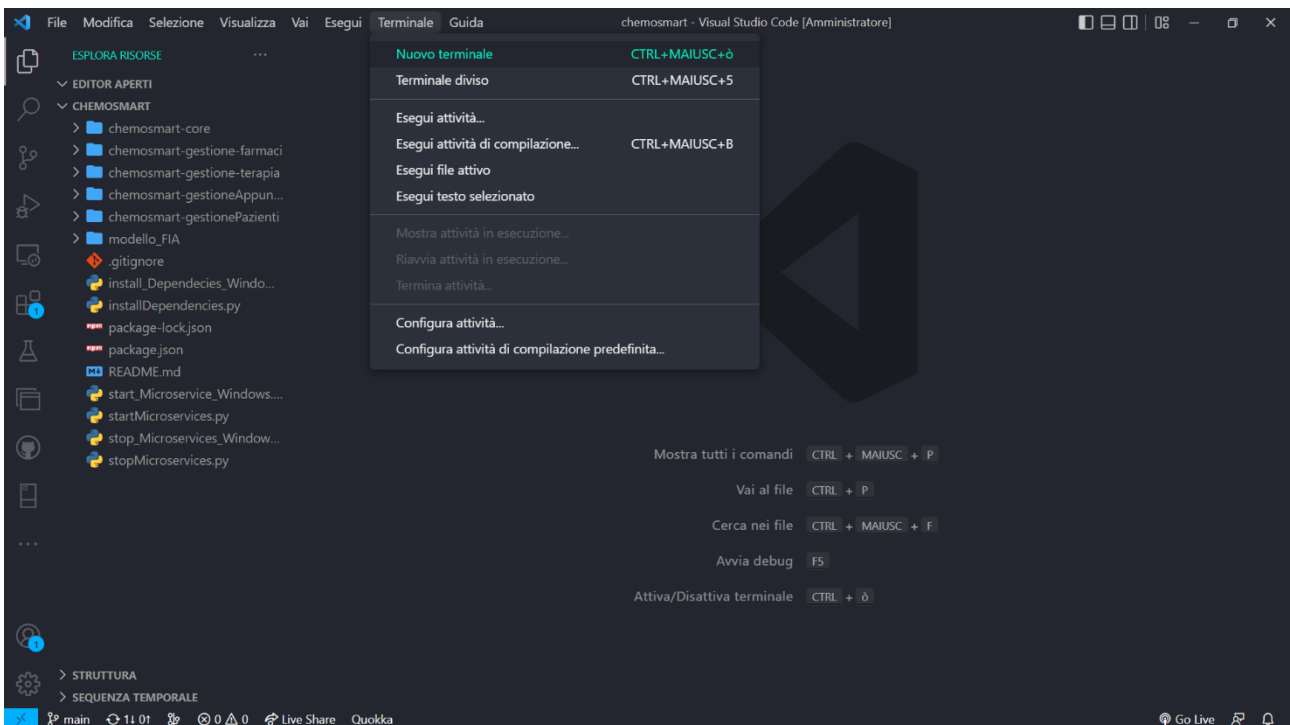
```
1 import os
2 import subprocess
3
4 # definisci la directory base in cui cercare i microservizi
5 base_dir = os.path.dirname(os.path.abspath(__file__))
6
7 # scandisci le cartelle all'interno della directory base
8 for folder in os.listdir(base_dir):
9     # costruisci il percorso completo della cartella corrente
10    current_dir = os.path.join(base_dir, folder)
11    # controlla se la cartella contiene un file package.json
12    if os.path.exists(os.path.join(current_dir, "package.json")):
13        # esegui il comando npm start nella cartella corrente
14        subprocess.Popen(f"start cmd /k npm i", shell=True, cwd=current_dir)
```



Si apriranno 5 terminali, uno per ogni microservizio; attendere il completamento dell'installazione delle librerie e successivamente si potranno chiudere i terminali.



Successivamente aprire un terminale su VSCode cliccando su “Nuovo terminale”





Eseguire i seguenti comandi:

**pip install pandas**

```
PS C:\Users\Luigi\Desktop\progettoIS\chemosmart> pip install pandas
```

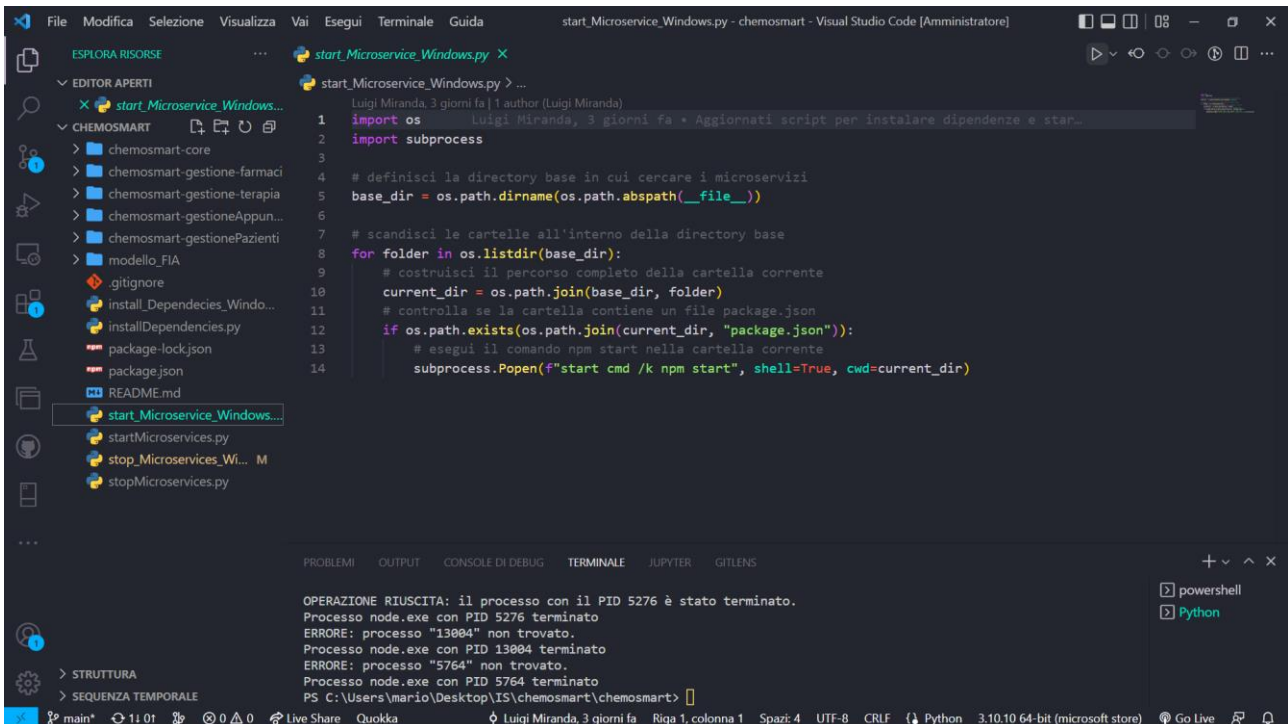
**pip install scikit-learn**

```
PS C:\Users\Luigi\Desktop\progettoIS\chemosmart> pip install scikit-learn
```

Per qualsiasi problema con l'installazione di queste librerie Python rifarsi alla documentazione ufficiale.

## 5.3 Eseguire i microservizi

Aprire il file “**start\_Microservices\_Windows.py**” e cliccare in alto a destra il tasto per eseguire il file

```

1 import os
2 import subprocess
3
4 # definisci la directory base in cui cercare i microservizi
5 base_dir = os.path.dirname(os.path.abspath(__file__))
6
7 # scandisci le cartelle all'interno della directory base
8 for folder in os.listdir(base_dir):
9     # costruisci il percorso completo della cartella corrente
10    current_dir = os.path.join(base_dir, folder)
11    # controlla se la cartella contiene un file package.json
12    if os.path.exists(os.path.join(current_dir, "package.json")):
13        # esegui il comando npm start nella cartella corrente
14        subprocess.Popen(f"start cmd /k npm start", shell=True, cwd=current_dir)

```

OPERAZIONE RIUSCITA: il processo con il PID 5276 è stato terminato.  
Processo node.exe con PID 5276 terminato  
ERRORE: processo "13804" non trovato.  
Processo node.exe con PID 13804 terminato  
ERRORE: processo "5764" non trovato.  
Processo node.exe con PID 5764 terminato  
PS C:\Users\mario\Desktop\IS\chemosmart\chemosmart>



Si apriranno 5 terminali, uno per ogni microservizio.

```
Amministratore: C:\WINDOWS\system32\cmd.exe
> chemosmart-core@1.0.0 start
> nodemon server.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting 'node server.js'
Server listening on http://localhost:3003

Amministratore: C:\WINDOWS\system32\cmd.exe
> chemosmart-gestioneappuntamenti@1.0.0 start
> nodemon index.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting 'node index.js'
(node:1744) [MONGOOSE] DeprecationWarning: Mongoose: the 'strictQuery' option will be switched back to 'false' by default in Mongoose 7. Use 'mongoose.set('strictQuery', false);' if you want to prepare for this change. Or use 'mongoose.set('strictQuery', true);' to suppress this warning.
(Use 'node --trace-deprecation ...' to show where the warning was created)
connessione al db...
Server running on port: http://localhost:3006

Amministratore: C:\WINDOWS\system32\cmd.exe
> chemosmart-gestionefarmaci@1.0.0 start
> nodemon index.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting 'node index.js'
(node:11972) [MONGOOSE] DeprecationWarning: Mongoose: the 'strictQuery' option will be switched back to 'false' by default in Mongoose 7. Use 'mongoose.set('strictQuery', false);' if you want to prepare for this change. Or use 'mongoose.set('strictQuery', true);' to suppress this warning.
(Use 'node --trace-deprecation ...' to show where the warning was created)
connessione al db...
Server running on port: http://localhost:3007

Amministratore: C:\WINDOWS\system32\cmd.exe
> chemosmart-gestionevisite@1.0.0 start
> nodemon index.js


[nodemon] 2.0.20
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting 'node index.js'
(node:17464) [MONGOOSE] DeprecationWarning: Mongoose: the 'strictQuery' option will be switched back to 'false' by default in Mongoose 7. Use 'mongoose.set('strictQuery', false);' if you want to prepare for this change. Or use 'mongoose.set('strictQuery', true);' to suppress this warning.
(Use 'node --trace-deprecation ...' to show where the warning was created)
connessione al db...
Server running on port: http://localhost:3008

Amministratore: C:\WINDOWS\system32\cmd.exe
> chemosmart-gestionevisite@1.0.0 start
> nodemon index.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter 'rs'
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting 'node index.js'
(node:19104) [MONGOOSE] DeprecationWarning: Mongoose: the 'strictQuery' option will be switched back to 'false' by default in Mongoose 7. Use 'mongoose.set('strictQuery', false);' if you want to prepare for this change. Or use 'mongoose.set('strictQuery', true);' to suppress this warning.
(Use 'node --trace-deprecation ...' to show where the warning was created)
connessione al db...
Server running on port: http://localhost:3009
```

Andare ora su: <http://localhost:3000/>



 **Entra con SPID**

Medico

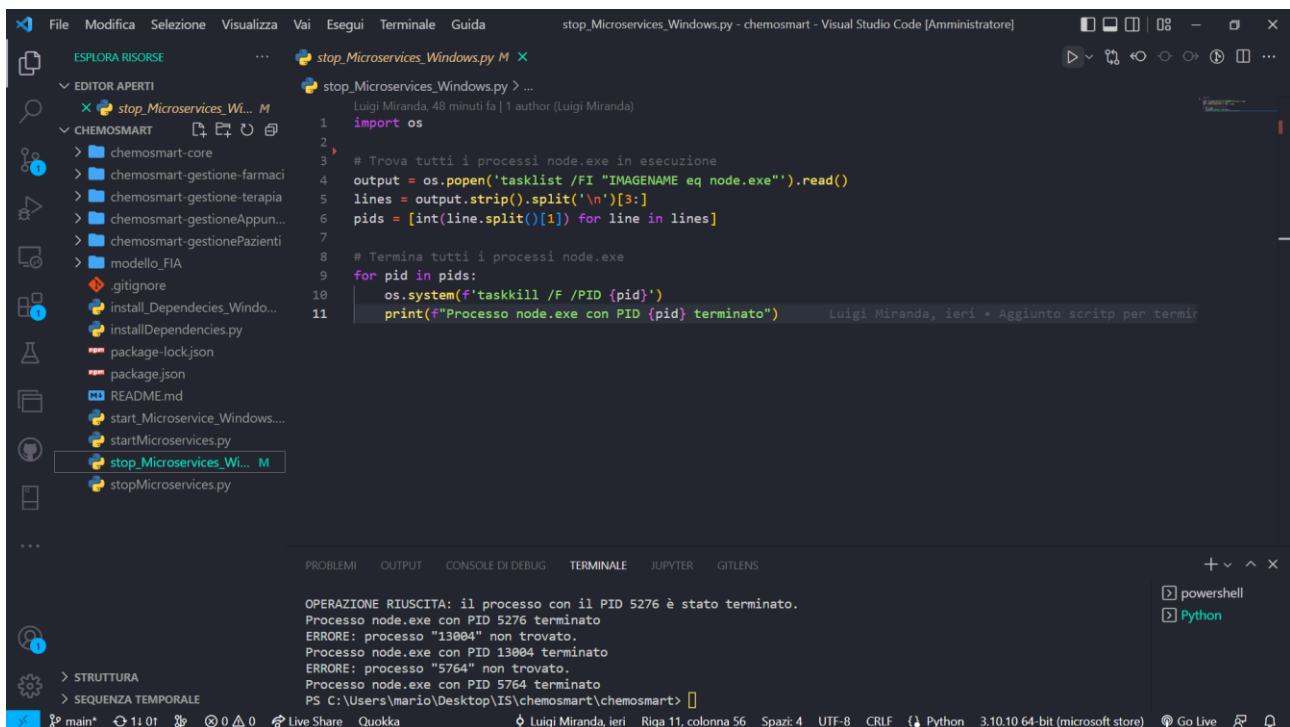
Medico

Segretario

Infermiere

## 5.4 Terminare i microservizi

Per terminare i microservizi aprire il file “stop\_Microservices\_Windows.py” e cliccare in alto a destra il tasto per eseguire il file 



```
1 import os
2
3 # Trova tutti i processi node.exe in esecuzione
4 output = os.popen('tasklist /FI "IMAGENAME eq node.exe").read()
5 lines = output.strip().split('\n')[3:]
6 pids = [int(line.split()[1]) for line in lines]
7
8 # Termina tutti i processi node.exe
9 for pid in pids:
10     os.system(f'taskkill /F /PID {pid}')
11     print(f"Processo node.exe con PID {pid} terminato")
```

OPERAZIONE RIUSCITA: il processo con il PID 5276 è stato terminato.  
Processo node.exe con PID 5276 terminato  
ERRORE: processo "13804" non trovato.  
Processo node.exe con PID 13804 terminato  
ERRORE: processo "5764" non trovato.  
Processo node.exe con PID 5764 terminato  
PS C:\Users\mario\Desktop\IS\chemosmart\chemosmart>

A questo punto tutti i microservizi saranno terminati correttamente.

## 6 Installazione manuale per tutti i sistemi

Rifarsi alle istruzioni del punto 5.1.

Seguire le seguenti istruzioni per ogni microservizio:

- chemosmart-core
- chemosmart-gestione-farmaci
- chemosmart-gestione-terapia
- chemosmart-gestioneAppuntamenti
- chemosmart-gestionePazienti

1. Aprire 5 terminali su VSCode

Per ciascun terminale recarsi in uno dei microservizi eseguendo il comando:





```
cd .\nomeMicroservizio\
```

### Esempio

Ripetere il comando per ogni terminale accedendo alla directory dei microservizi restanti.

2. Installare le dipendenze necessarie per ciascun microservizio eseguendo in ogni directory di ogni terminale precedentemente aperto il comando:

**npm i**

### Esempio

3. Installare librerie di python con i seguenti comandi in un in terminale:

**pip install pandas**

### Esempio

**pip install scikit-learn**

### Esempio

4. Infine, per poter eseguire il progetto, digitare in ogni terminale il comando:



## npm start

### Esempio

```
PS C:\Users\Luigi\Desktop\progettoIS\chemosmart> cd .\chemosmart-core\  
PS C:\Users\Luigi\Desktop\progettoIS\chemosmart\chemosmart-core> npm start
```

The screenshot shows a Windows PowerShell terminal window with the following content:

- Terminal title bar: + - ... ^ x
- Terminal tabs: PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL
- Terminal content:
  - PS C:\Users\Luigi\Desktop\progettoIS\chemosmart> cd .\chemosmart-core\  
PS C:\Users\Luigi\Desktop\progettoIS\chemosmart\chemosmart-core> npm start
- Terminal output (on the right side):
  - pwsh chemosmart-core
  - pwsh
  - pwsh
  - pwsh
  - pwsh