

# CableShark

CableShark è un'applicazione che permette di effettuare il ping a uno o più indirizzi ip e visualizzarne lo stato.

## indice

- [Come avviare il progetto](#)
- [Guida all'uso](#)
- [Funzionamento dello script](#)
- [Funzionamento del ping](#)

## Come avviare il progetto

- clonare il progetto
- eseguire lo script 'Start.sh' se si è su linux/mac o 'Start.bat' se si è su windows, questo fa:
  - questo script controlla se è installato python 3.11 o superiore, se non lo è lo installa
  - installa le dipendenze necessarie per l'applicazione presenti nel file 'requirementsLinux.txt' o 'requirementsWindows.txt' a seconda del sistema operativo
  - avvia l'applicazione

```
git clone [url del progetto]
./Start.sh # se si è su linux/mac
.\Start.bat # se si è su windows
```

## Guida all'uso

All'avvio dell'applicazione compare una finestra che è divisa in due parti:

- a sinistra c'è la lista, inizialmente vuota, degli indirizzi ip aggiunti :
  - per aggiungere un indirizzo ip bisogna scrivere l'indirizzo ip nel textBox in alto a destra e premere il pulsante 'add to list'.
  - per ogni Ip address viene mostrato l'effettivo indirizzo ip, che può essere cliccato per rieffettuare il ping, e sotto di lui il suo stato:
    - 'online' se l'indirizzo ip risponde al ping
    - 'offline' se l'indirizzo ip non risponde al ping
    - 'unknown' se non è stato ancora effettuato il ping
    - 'error' se c'è stato un errore durante il ping
  - sotto al pulsante 'add to list' c'è il pulsante 'Clear List' che ripulisce la lista degli indirizzi ip aggiunti

- o sotto al pulsante 'Clear List' c'è il pulsante 'refresh list' che riefettua il ping a tutti gli indirizzi ip presenti nella lista
- a destra è presente:
  - o un textBox in cui è possibile scrivere l'indirizzo ip da pingare
  - o un pulsante 'ping' che effettua il ping all'indirizzo ip scritto nel textBox
  - o una label che mostra lo stato dell'indirizzo ip scritto nel textBox
  - o un textBox disabilitato in cui viene mostrato l'output del ping
- è possibile vedere anche tutti i risultati dei ping fatti in maniera automatica aprendo il file log.txt presente nella root del progetto

# Funzionamento dello script

Lo script 'Start.sh' o 'Start.bat' come ultimo passaggio eseguono 'python3 main.py' che avvia l'applicazione dentro al quale è presente:

- la funzione chiamata cliccando il pulsante 'ping' che effettua il ping all'indirizzo ip scritto nel textBox, per farlo però avvia un thread che effettua il ping e aggiorna la label con lo stato dell'indirizzo ip (thread necessario per non bloccare l'interfaccia grafica)
- la funzione refreshList che riefettua il ping a tutti gli indirizzi ip presenti nella lista
- inoltre viene creata una finestra creando un nuovo GraphicsElement che crea e gestisce la finestra grafica, compreso il resize della finestra.

Appena viene aggiunta un indirizzo ip alla lista viene fatto partire un thread che effettua il ping all'indirizzo ip e aggiorna la label con lo stato dell'indirizzo ip, questo metodo è presente nel file graphicAddAndUpdate.py.

## Funzionamento del ping

Il ping viene effettuato tramite l'apertura di una socket e la costruzione di un pacchetto icmp, il pacchetto viene inviato all'indirizzo ip e si aspetta una risposta. Il pacchetto viene costruito in questo modo:

- header icmp (8 byte) costruito con il modulo struct nella seguente maniera "headers = struct.pack('!BBHHH', 8, 0, 0, 1, 1)" che corrisponde a creare un pacchetto icmp di tipo 8 (echo request) e codice 0, con un identificativo 0 e un numero di sequenza 1, il secondo 0 è il checksum che viene calcolato dopo.
- data è un array di byte che contiene la stringa "Hello, World!" che viene aggiunta al pacchetto icmp
- il checksum viene calcolato con la funzione calculate\_checksum(header + data) presente nel file pinger.py

Il pacchetto viene inviato con la funzione sendto(header + data, (ip, 1)) presente nel file pinger.py, il secondo parametro della funzione è una tupla con l'indirizzo ip e la porta, la porta 1 è la porta standard per il protocollo icmp; in seguito si aspetta una risposta tramite la funzione select del modulo select, che aspetta per 5 secondi una risposta, se non arriva nessuna risposta si considera l'indirizzo ip offline, altrimenti si considera online.

## Dipendenze

- python 3.11 o superiore
- tkinter

Di seguito i comandi per installarle su linux e windows:

- linux:

```
- sudo apt-get install python3  
- sudo apt-get install python3-tk
```

- windows:

- installare python 3.11 #o superiore da <https://www.python.org/downloads/>  
(<https://www.python.org/downloads/>)
- installare tkinter con il comando

```
pip install tk
```

OPPURE eseguire lo script 'Start.sh' se si è su linux o 'Start.bat' se si è su windows che fa già tutto questo in maniera automatica

# Autore

- [Fabio Fattori \(https://github.com/FabioFattori\)](https://github.com/FabioFattori), numero di matricola: 0001071463