

## Relazione Tps

### Consegna:

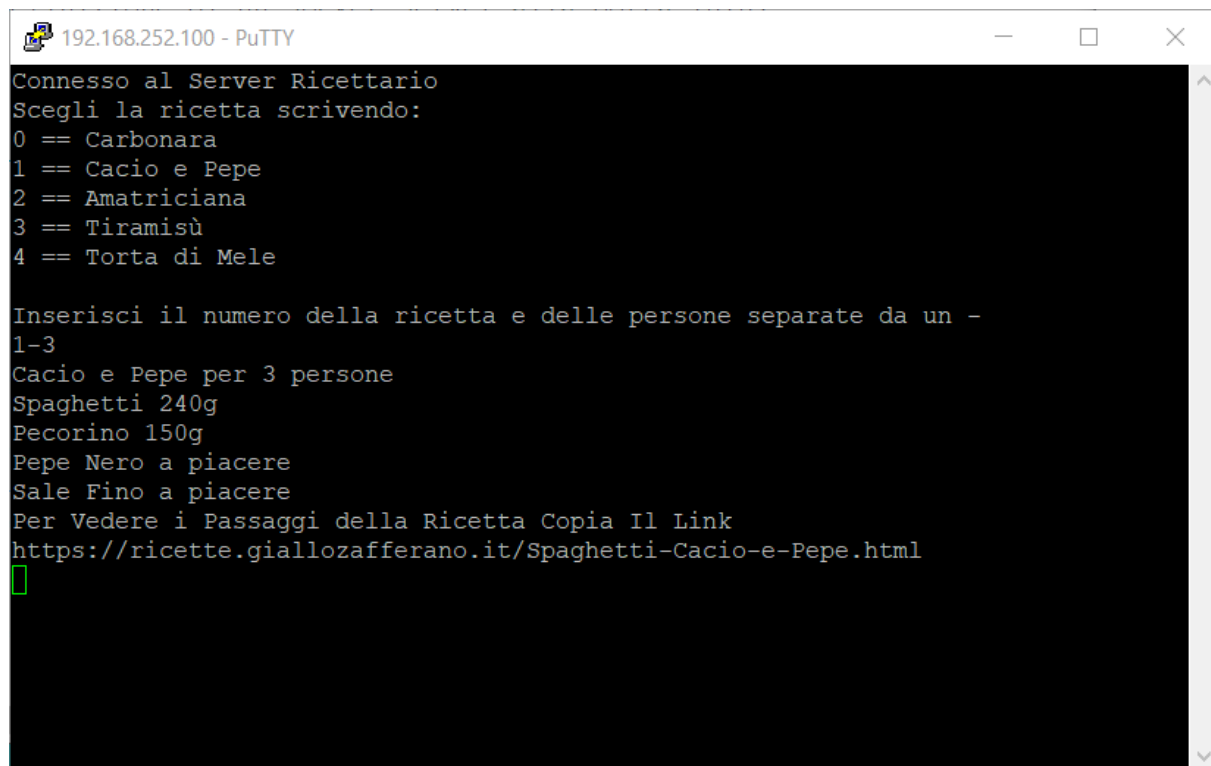
creazione di un server tramite l'utilizzo di un esp8266 programmabile attraverso l'IDE di Arduino

### Analisi del problema:

l'obiettivo è la creazione di un server utilizzando un microcontrollore esp8266 in ambiente Arduino ide. Il server dovrà elaborare le richieste del client secondo il protocollo ricettario.

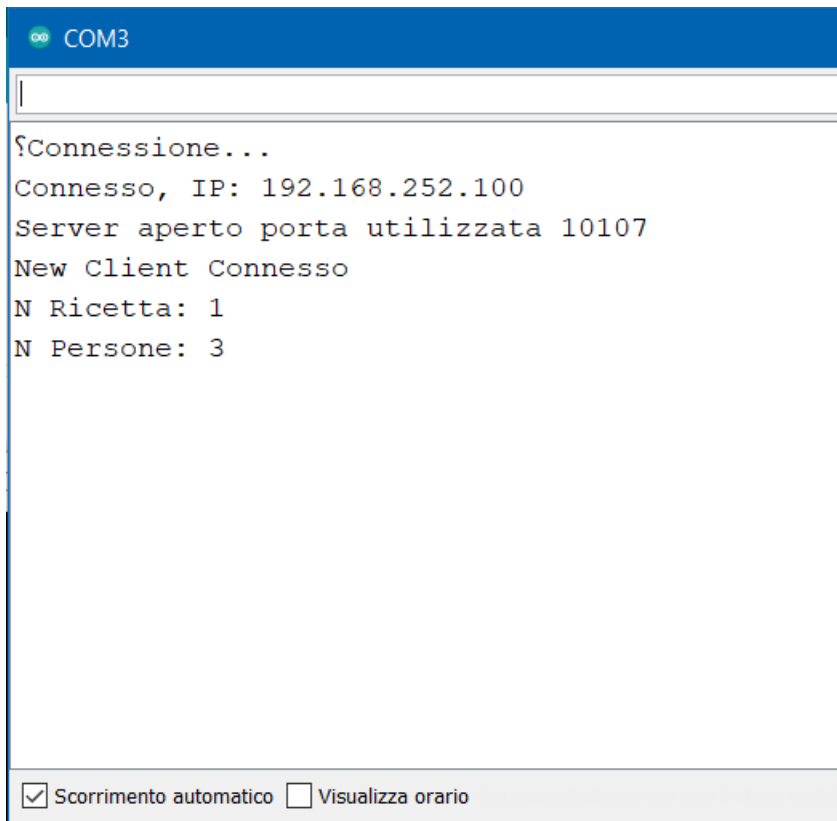
Il protocollo ricettario è un server che in base alla ricetta selezionata e in base al numero di persone elabora e restituisce gli ingredienti e le quantità necessarie alla realizzazione della ricetta.

### Funzionamento:



```
192.168.252.100 - PuTTY
Connesso al Server Ricettario
Scegli la ricetta scrivendo:
0 == Carbonara
1 == Cacio e Pepe
2 == Amatriciana
3 == Tiramisù
4 == Torta di Mele

Inserisci il numero della ricetta e delle persone separate da un -
1-3
Cacio e Pepe per 3 persone
Spaghetti 240g
Pecorino 150g
Pepe Nero a piacere
Sale Fino a piacere
Per Vedere i Passaggi della Ricetta Copia Il Link
https://ricette.giallozafferano.it/Spaghetti-Cacio-e-Pepe.html
█
```



## Svolgimento:

server →

- Importare la libreria:  
`#include <ESP8266WiFi.h>`
- Inserire le costanti per il nome della wifi e per la password:  
`const char* ssid = "nome password";  
const char* password = "password";`
- Dichiarare un array contenente il nome delle ricette disponibili:  
`const String Lista[] = { "Carbonara", "Cacio e  
Pepe", "Amatriciana", "Tiramisù", "Torta Di Mele"};`
- Creare un oggetto wi-fi server:  
`WiFiServer server(10107);`
- Impostare la velocità dati in bit al secondo:  
`Serial.begin(115200);`
- Passare alla classe wifi, con il metodo begin, i parametri ssid e la password:  
`WiFi.begin(ssid, password);`
- Creare un ciclo while che si ripeterà fin quando l'esp8266 non si conntterà alla rete wifi; Una volta connesso si avrà la conferma sul monitor seriele

assieme all'indirizzo ip associato al microcontrollore, mettendosi poi in ascolto:

```
while(WiFi.status() != WL_CONNECTED){ delay(500); }  
Serial.print("Connesso, IP: ");  
Serial.println( WiFi.localIP() );  
server.begin();  
Serial.println("Server aperto porta utilizzata 10107");
```

- Dichiarare l'oggetto wifi Client associato al client appena connesso:

```
WiFiClient cliente = server.available();
```

- Stampare l'elenco delle ricette disponibili:

```
cliente.println("Connesso al Server Ricettario");  
cliente.println("Scegli la ricetta scrivendo:");  
cliente.println("0 == Carbonara");  
cliente.println("1 == Cacio e Pepe");  
cliente.println("2 == Amatriciana");  
cliente.println("3 == Tiramisù");  
cliente.println("4 == Torta di Mele");
```

- Richiedere il numero della ricetta e delle persone:

```
cliente.println("Inserisci il numero della ricetta e delle persone  
separate da un -");
```

- Memorizzazione della ricetta scelta e delle persone su due variabili di tipo stringa:

```
String message = cliente.readStringUntil('\r');  
message.replace("\n", "");  
message.replace("\r", "");  
  
String ricetta="";  
String persone="";  
  
for(int i=0;i<message.length();i++)  
{  
    char num=message[i];  
    if(num != '-' && message.indexOf(num)<message.indexOf("-"))  
    {  
        ricetta+=num;  
    }else if(num != '-' && message.indexOf(num)>message.indexOf("-"))  
    {  
        persone+=num;  
    }  
}
```

- In base al numero della ricetta scelta e al numero di persone sono calcolate le quantità degli ingredienti per poi essere stampante. Per far ciò occorre verificare, attraverso un if, il valore della variabile ricetta:

```
if(ricetta=="0"){  
    stampa+=Lista[message.toInt()];  
    cliente.println(stampa + " per " + ricetta + " persone");  
    cliente.print("Spaghetti "); cliente.print (ottanta*ricetta.toInt());  
    cliente.println("g");  
    cliente.print("Guanciale "); cliente.print (trentasette*ricetta.toInt());  
    cliente.println("g");
```

```
cliente.print("Tuorli "); cliente.print (unocinque*ricetta.toInt());  
cliente.println("g");  
cliente.print("Pecorino "); cliente.print (dodici*ricetta.toInt());  
cliente.println("g");  
cliente.println("Pepe a piacere");  
cliente.println("Per Vedere i Passaggi della Ricetta Copia Il Link");  
cliente.println("https://ricette.giallozafferano.it/Spaghetti-alla-  
Carbonara.html");  
}
```

## Conclusioni:

Procedendo alla realizzazione del progetto ho riscontrato alcune difficoltà:

- Arduino compila il codice ma caricandolo nell'esp8266 da il seguente errore: *"espcomm\_upload\_mem"*  
Per risolvere il problema basta cambiare nella barra degli strumenti la scheda passando da Generic 8266 Module a Wemod D1 R2 & mini
- Quando si doveva prendere in input il numero della ricetta e delle persone non è stato possibile memorizzarli separatamente tramite due comandi distinti, perciò si è deciso di memorizzare entrambi attraverso un unico comando