



UNIVERSITÀ DEGLI STUDI
DI GENOVA

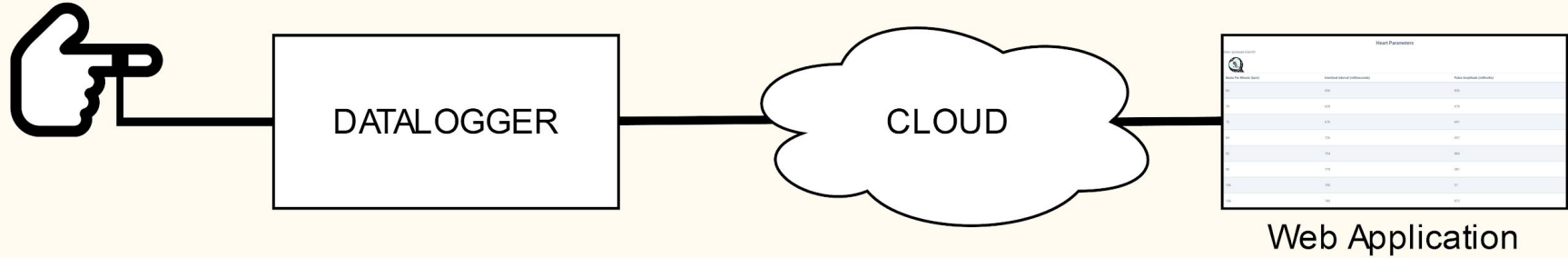
Progettazione e implementazione su modulo Arduino di un datalogger per il monitoraggio di parametri cardiaci

Marzo 2020

Candidato: Giorgio Demarzi

Relatore: Riccardo Berta

Introduzione

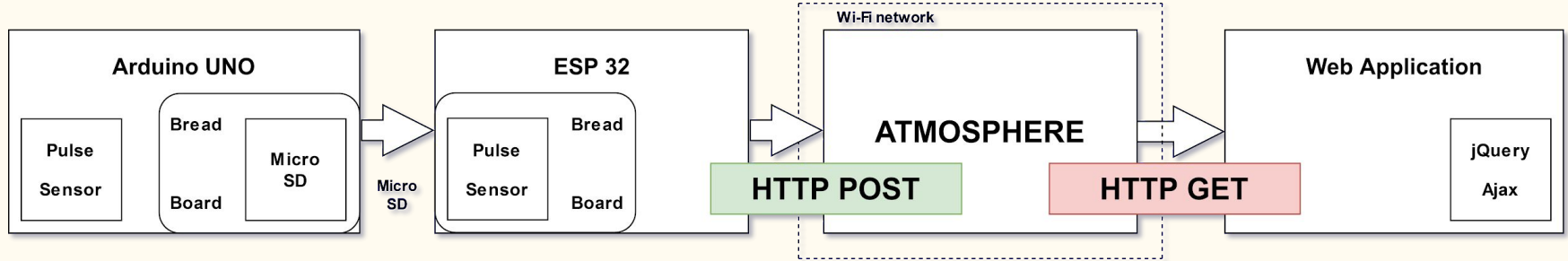


Obiettivo: Implementare un modulo capace di essere un data logger in grado di registrare parametri cardiaci attraverso un sensore esterno ed anche in grado di poter inviare questi valori ad un cloud per renderli disponibili per futuri utilizzi.

I parametri cardiaci considerati sono:

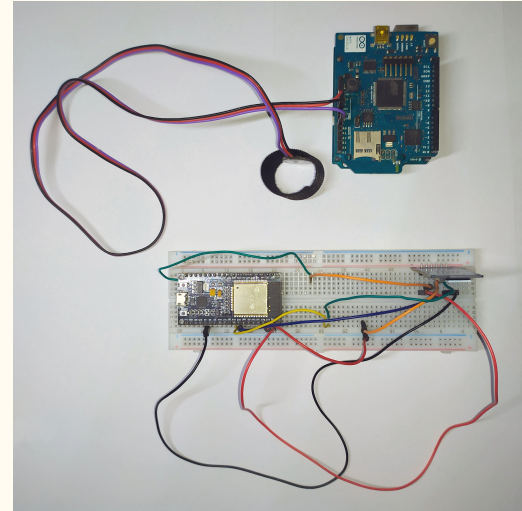
- **Numero di battiti del cuore al minuto** (Beats Per Minute - **BPM**)
- **L'intervallo di tempo fra due battiti** (Interbeat interval - **IBI**)
- **L'ampiezza massima della forma d'onda del battito** (Pulse Amplitude - **PA**).

Diagramma del sistema completo



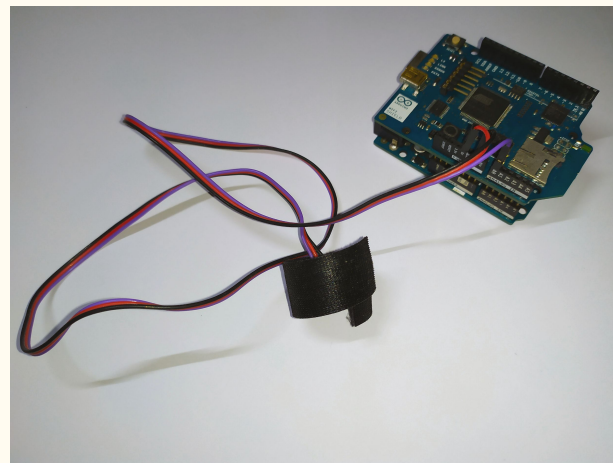
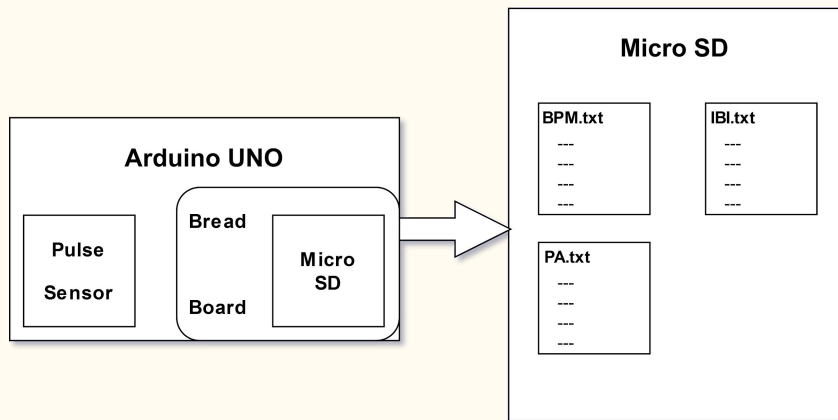
Il progetto è costituito da due moduli fisici e una interfaccia Web:

- **Modulo Arduino UNO con Pulse Sensor e MicroSD Card Adaptor**
- **Modulo ESP32 con MicroSD Card Adaptor**
- **Interfaccia Web basata su HTML, CSS e JavaScript**



Arduino UNO con il sensore e l'ESP32 con MicroSD Card Adaptor

Modulo Arduino UNO con Pulse Sensor e MicroSD Card Adaptor



Arduino UNO con il sensore

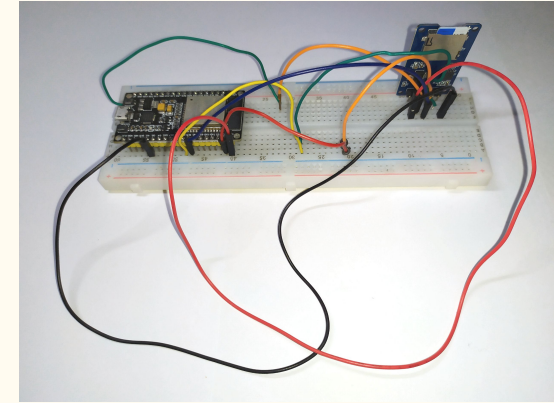
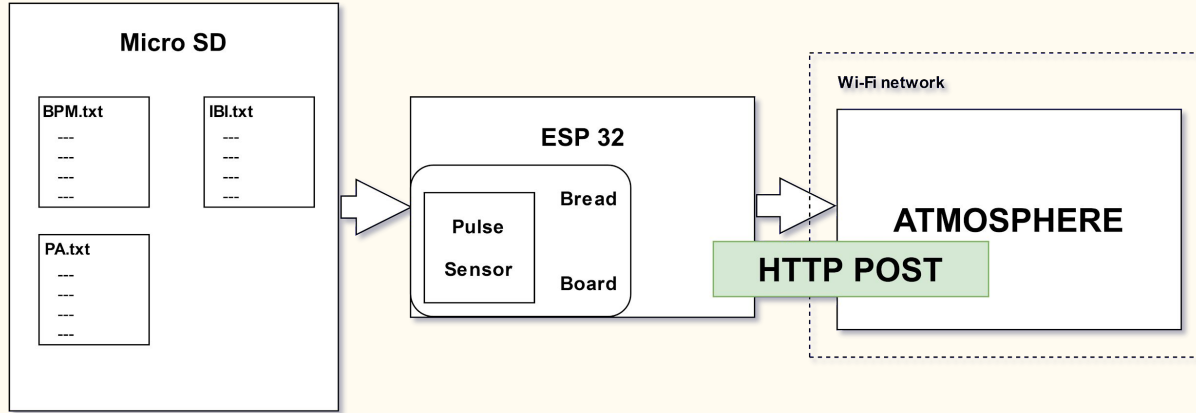
Il codice caricato sull'Arduino UNO si occupa di rilevare, ad ogni battito cardiaco percepito dal sensore, i valori di interesse e di memorizzarli su tre differenti file .txt sulla scheda SD.

Come ogni codice Wiring di Arduino è costituito da tre parti principali:

- **Introduzione:**
 - ➔ Dichiarazione delle variabili
 - ➔ Dichiarazione delle librerie (per il funzionamento del Pulse Sensor, per comunicare con le periferiche vicine e per il funzionamento della MicroSD)
- **void setup():** La seconda parte o parte di setup, viene eseguita solo all'avvio del programma.
 - ➔ Apertura della comunicazione ed il porto seriale,
 - ➔ Si creano i file BPM.txt, IBI.txt e PA.txt sulla MicroSD
 - ➔ Configurazione del sensore, inserendo i valori precedentemente definiti.
- **void loop():** La terza parte o parte di loop viene eseguita ripetutamente.
 - ➔ Lettura dei valori BPM, IBI e PA dal sensore, se è avvenuto un battito vengono formattati come String e sono memorizzati nei file nei rispettivi .txt.

Le misure vengono salvate ad ogni battito percepito dal sensore.

Modulo ESP32 con MicroSD Card Adaptor



MicroSD Card Adaptor

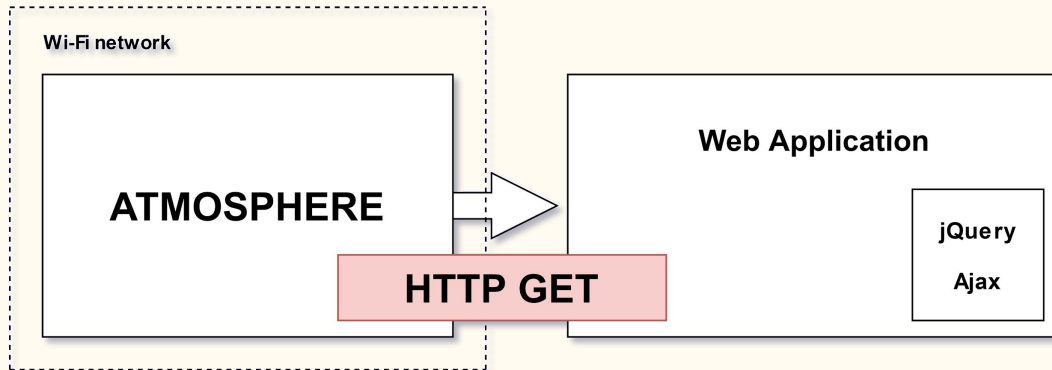
Il codice caricato sull'ESP32 si occupa di rilevare le misure presenti nei file .txt all'interno della Micro SD, di generare e inviare la richiesta HTTP Post alla Web API "Atmosphere".

L'ESP32 è compatibile con l' Arduino IDE e quindi il linguaggio di programmazione è lo stesso Wiring.

- **Introduzione:**
 - ➔ Dichiarazione di tutte le variabili necessarie per effettuare l'HTTP POST.
 - ➔ Viene definito il TOKEN di autorizzazione necessario per la comunicazione con "Atmosphere"
 - ➔ Dichiarazione delle librerie (per comunicare con le periferiche vicine, per il funzionamento della MicroSD, per la comunicazione Wifi e per eseguire richieste HTTP POST)
 - ➔ Si dichiarano tre array di lunghezza arbitraria che conterranno i valori memorizzati nella MicroSD.
- **void setup():**
 - ➔ Apertura della comunicazione ed il porto seriale
 - ➔ Inizializzazione della scheda MicroSD
 - ➔ Avvio connessione Wi-Fi
- **void loop():**
 - ➔ Si legge il file BPM.txt e si memorizzano le misure presenti nel rispettivo array. Se la connessione è presente, viene costruita la richiesta HTTP POST contenente i valori BPM e successivamente viene inviata al server.
 - ➔ Vengono ripetute esattamente le stesse istruzioni, per l'IBI e per il PA.

Il programma entra successivamente in un loop infinito e non vengono più inviate altre richieste HTTP, fino al prossimo RESET della scheda.

Interfaccia Web basata su HTML, CSS e JavaScript



La web application è basata su HTML, CSS e Javascript.

Il codice dell'interfaccia Web è distribuito su due files: **index.html** e **data.js**.

Il codice HTML si occupa di definire lo stile ed il layout della tabella.

Il codice JavaScript si occupa di effettuare la richiesta HTTP GET e di mettere a disposizione le misure all'HTML.

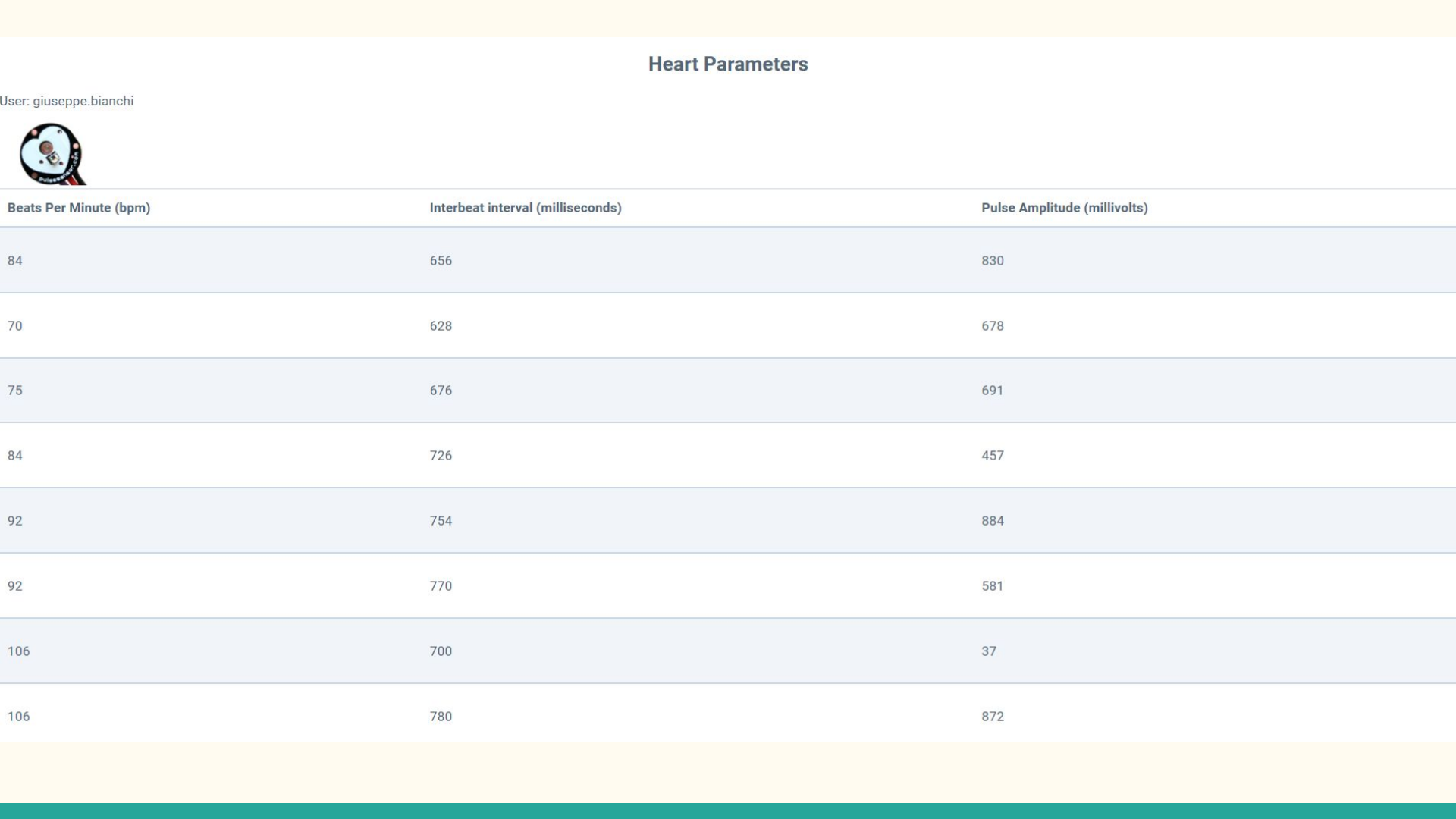
Codice index.html

Si costruisce la pagina html:

- Viene definito il titolo della pagina HTML
- Viene importato lo stile CSS per la creazione della tabella, messo a disposizione dal sito www.siimple.xyz
- Viene importata la risorsa JQuery necessaria per formulare la richiesta HTTP
- Viene importato lo script dove viene formulata la richiesta HTTP (data.js)
- Vengono definiti i parametri della tabella:
 - ➔ Nella prima riga viene definito il tipo di misurazione effettuata mentre le altre righe vengono riempire dalle misure recuperate dal server

Codice data.js

- Viene formulata la richiesta HTTP
- Vengono messe a disposizione all'HTML le misure recuperate dal server



Considerazioni conclusive

Le possibilità di ampliamento del progetto sono notevoli.

Il risultato ottenuto può essere considerato un buon punto di partenza per ulteriori sviluppi e miglioramenti:

- Riportare graficamente i valori ottenuti
- Aumentare il numero di parametri monitorati
- Effettuare la comparazione dei valori fra diversi utenti
- Permettere computazioni tra i dati rilevati
- ...

Grazie per l'attenzione