

Web Blinking Led

Un semplice circuito costruito e programmato con Arduino, NodeJs e Johnny-Five e controllato da una pagina HTML.

Setup iniziale

Questo tutorial presuppone che si sia prima realizzato il circuito e scritto il codice riportato nel tutorial [Blinking LED](#) presente anche lui nella pagina [pagina GitHub](#) del Coderdojo di Firenze.

Si rimanda al tutorial presente su quel repository per tutte le indicazioni. Qui si suppone che il circuito sia realizzato e funzionante.

Realizzazione di un semplice web server

NodeJS mette a disposizione moltissime funzionalità "prefabbricate" che possono essere utilizzate degli sviluppatori per realizzare funzionalità anche complesse senza bisogno di partire da zero.

Per questo, per esempio, realizzare un **web server** (un software che serve pagine web a client che ne facciano richiesta, tipicamente un browser come Firefox o Google Chrome) è veramente facile. Tutto il codice che dobbiamo scrivere è il seguente:

```
const http = require('http');

const server = http.createServer(myServer);

function myServer(req, res) {
  if (req.method == "GET") {
    res.writeHead(200, {'Content-type': 'text/html'});
    res.end(`
      <h1>La mia prima pagina web con NodeJS</h1>
    `);
  }
}

server.listen(1337, 'localhost', () => {
  console.log('Server avviato.');
```

Aggiungiamo questo codice al file `blink.js` che abbiamo scritto realizzando il tutorial [Blinking LED](#).

Una volta scritto il codice, salviamo ed eseguiamo il programma con il comando `node blink.js`. Il circuito controllato da Johnny-Five continuerà a funzionare come prima; adesso in più possiamo aprire un browser: collegandosi all'URL `http://localhost:1337` apparirà una pagina bianca con il titolo da noi inserito.

Abbiamo realizzato il nostro primo server web con meno di 15 linee di codice! Digitiamo Ctrl-C nella finestra dove si è lanciato il comando `node blink.js` per interrompere l'esecuzione.

Complichiamo la pagina

L'idea alla base di questo tutorial è quello di controllare tramite Johnny-Five il LED collegato alla scheda Arduino. Per fare questo dobbiamo fare in modo che tramite il browser sia possibile inviare dei "comandi" al server HTTP. Il modo più semplice per realizzare questo consiste nell'utilizzare un *form* e il metodo **POST**.

Completiamo quindi il codice del server che abbiamo realizzato in precedenza nel seguente modo:

```
const http = require('http');

const server = http.createServer(myServer);
var ledAttivo = true;

function myServer(req, res) {

  if (req.method == "GET") {
    res.writeHead(200, {'Content-type': 'text/html'});
    res.end(`
      <h1>La mia prima pagina web con NodeJS</h1>

      <form name="myForm" action="/" method="post">\
        <input type="checkbox" name="led1" value="attivo"
          ${ledAttivo ? 'checked' : ''}> LED attivo<br>
        <input type="submit" value="Imposta"> \
      </form> \
    `);
  }
}

server.listen(1337, 'localhost', () => {
  console.log('Server avviato. Collegarsi a http://localhost:1337 per
vedere la pagina');
});
```

Adesso abbiamo la possibilità di selezionare o delezionare il *checkbox* "LED Attivo". Come si può vedere però, cliccando sul tasto *Imposta* ancora non succede niente. Per completare le operazioni dobbiamo gestire lato server l'evento POST che il browser invia al server tutte le volte che si preme il pulsante.

Modifichiamo quindi il file `blink.js` aggiungendo il seguente codice al corpo della funzione `myServer`:

```
if (req.method == "GET") {
  //... codice già scritto in precedenza
}
else if (req.method == "POST") {

  var statoLed = '';
  req.on('data', function(rawData) {

    // converte i dati ricevuti dal messaggio POST in stringa
    var postData = rawData.toString();

    console.log(`POST received! formdata = ${postData}`);

    statoLed = postData.split("&")[0].split("=")[1];

  }).on('end', function() {

    if (statoLed === 'attivo') {
      console.log(`Stato LED = attivo`);
      modoLedCheckbox = 'checked';
      ledAttivo = true;
    }
    else {
      console.log(`Stato LED = inattivo`);
      modoLedCheckbox = '';
      ledAttivo = false;
    }

    res.writeHead(200, {'Content-type': 'text/html'});
    res.end(`
      <h1>La mia prima pagina web con NodeJS</h1>

      <form name="myForm" action="/" method="post">\
        <input type="checkbox" name="led1" value="attivo"
          ${ledAttivo ? 'checked' : ''}> LED attivo<br>
        <input type="submit" value="Imposta"> \
      </form> \
    `);

  });
};
```

Non succede molto, ma la pagina è adesso almeno di nuovo caricata quando premiamo il pulsante *Imposta*. Inoltre, cosa più importante, adesso il nostro server ha a disposizione la variabile `ledAttivo` allineata con lo stato del checkbox presente sulla pagina web.

Quello che ci rimane da fare è quindi utilizzare questa variabile per attivare o disattivare il lampeggiamento del LED sulla scheda Arduino.

Modifichiamo quindi il codice Johnny-Five presente nel file `blink.js` nel seguente modo:

```
board.on('ready', function() {  
  // All'avvio della scheda accendiamo o spengiamo il LED  
  // collegato al pin 13 a seconda dello stato  
  var led = new five.Led(13);  
  if (ledAttivo)  
    led.blink(500);  
  else  
    led.off();  
  
  // Successivamente controlliamo ogni 500 ms se il led  
  // è cambiato di stato  
  this.loop(500, () => {  
    // aggiorna lo stato del led  
    if (ledAttivo)  
      led.blink(500);  
    else  
      led.off();  
  });  
});
```