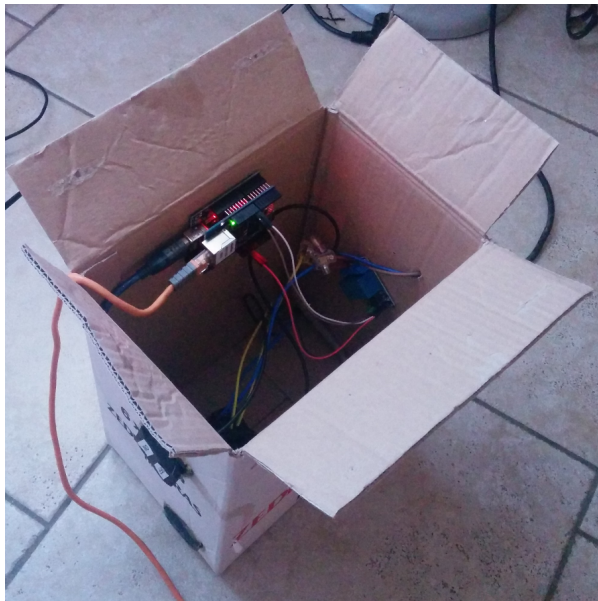


# Domotica con Arduino



Questo progetto, che chiamo sinteticamente come da titolo, si occupa di gestire l'accensione e lo spegnimento di elettrodomestici e illuminazione, attraverso una semplice interfaccia web. Per realizzare questo progetto mi sono servito di un Arduino, una scheda dotata di microcontrollore che permette l'interfacciamento con svariati moduli, che permettono a loro volta l'interfacciamento con una sterminata lista di oggetti elettronici.

Prima di proseguire con la descrizione del progetto, elenco, di seguito, il materiale utilizzato:

- Arduino Uno;
- Arduino Ethernet Shield;
- MicroSD da 512MB;
- Cavo Ethernet;
- 2 x Relay;
- Interruttore da muro;
- 2 x prese elettriche;
- 4 x Jumper;
- Cavo con spina elettrica;
- Portalampadina;
- Lampadina;
- Scatola di cartone.

Molto del materiale utilizzato è stato riciclato da vecchi lavori, o da oggetti che si trovavano già per casa, quindi non posso indicare una spesa complessiva dei componenti. In base alle componenti acquistate ad-hoc per il progetto, posso stimare però una spesa totale che si aggira intorno ai 30 euro.

Michael Planu 24/6/y 18:23

**Commenta [1]:** Inserire il dettaglio del clone

## Descrizione

Tutto il progetto si basa, come scritto in precedenza, su Arduino, ma a gestire le singole componenti sono due moduli, che estendono le funzionalità di quest'ultimo.

In particolare, i moduli utilizzati sono:

- **Arduino Ethernet Shield:** questo modulo permette di creare un server HTML e di gestire, tramite una pagina web, tutte le porte digitali di cui Arduino dispone. Per fare questo si interfaccia, come dice il nome stesso, via Ethernet ad un PC (che funge da client) o ad un router (permettendo così a più client di accedervi contemporaneamente).
- **Modulo Relay:** questo modulo contiene due relay (o relè che dir si voglia), che in parole povere sono degli interruttori comandabili (in questo caso tramite un segnale digitale, inviato da Arduino). Il funzionamento è semplice: colleghiamo la porta che funge da controllore ad una porta digitale di Arduino, il quale deciderà se tenere quella porta accesa (stato alto) o spenta (stato basso). In base a questo segnale, il relay aprirà o chiuderà il circuito, banalmente accenderà o spegnerà il dispositivo ad esso collegato.

In questo modo io ho potuto collegare l'Ethernet shield ad Arduino, creando così il webserver che andrà a rispondere alle richieste del PC, connesso tramite Ethernet.

Ad Arduino ho poi collegato il modulo relay che, tramite l'alimentazione ricevuta dalla presa elettrica, potrà fornire elettricità (in base al segnale datogli da Arduino), ai dispositivi connessi.

Ma distinguiamo il collegamento elettrico da quello digitale, utilizzato per comandare i relay.

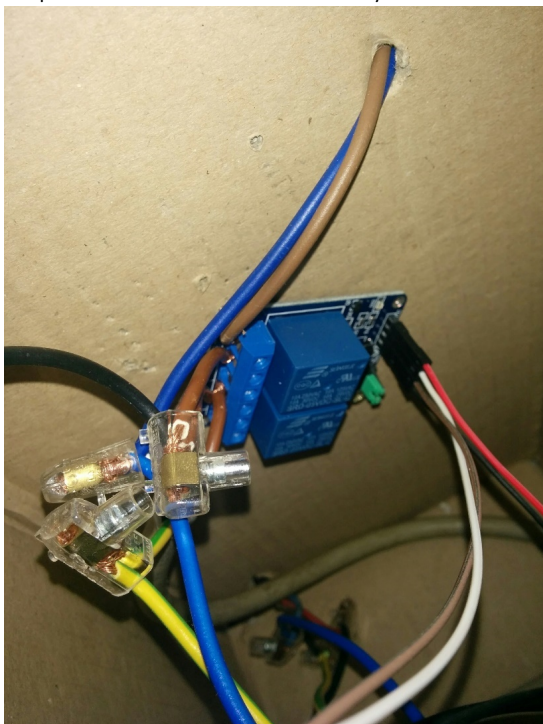
Il cavo elettrico è "spezzato" in vari punti, da varie componenti. Troviamo prima di tutto un interruttore generale, che serve a spegnere o accendere l'intero sistema:



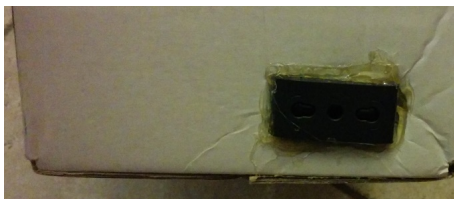
Troviamo poi una presa elettrica che, come si può vedere in figura, serve ad alimentare Arduino:



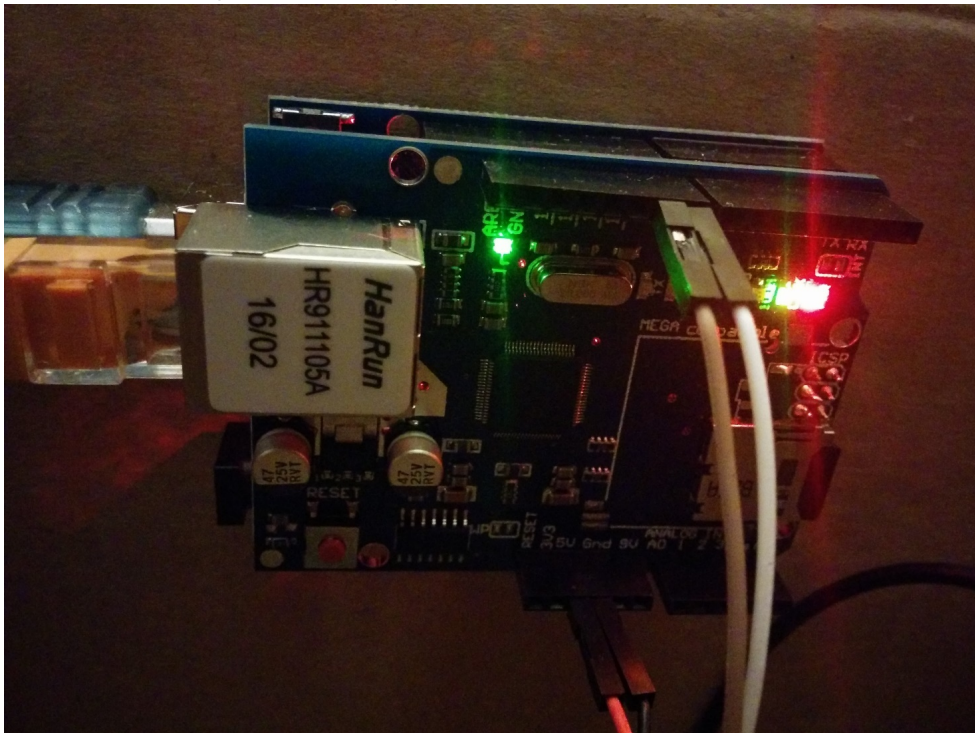
Per poi terminare alimentando i due relay:



Terminando con il lato hardware, troviamo il portalampadina e la presa, collegate ai due relay:



E il cavo ethernet, collegato allo shield e al pc:



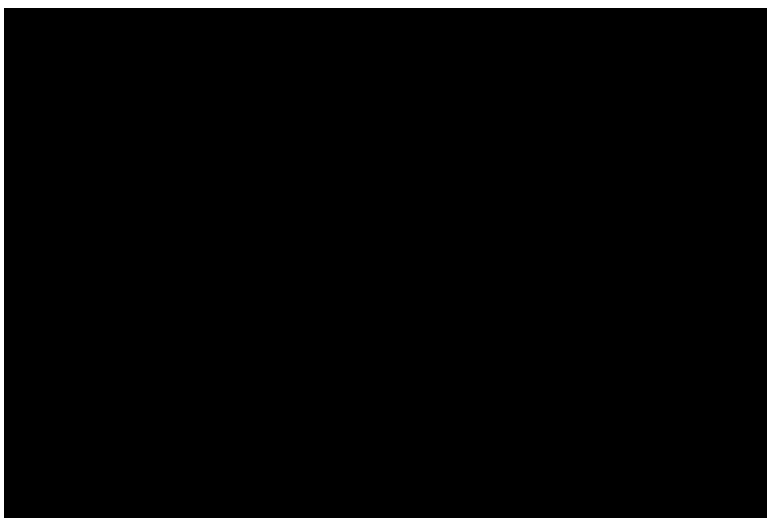
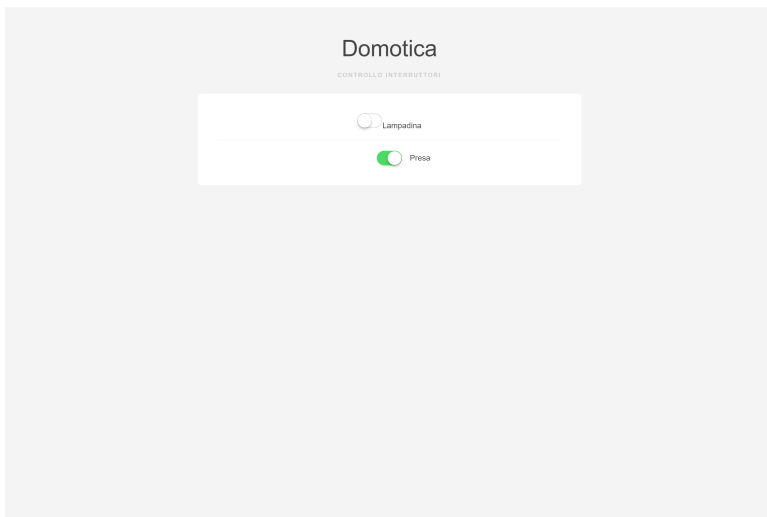
## Software

Lato software abbiamo una pagina HTML che risiede sulla microSD inserita sull'Ethernet Shield, ed un'applicativo Arduino che risiede, appunto, su Arduino.

L'applicativo si occupa di generare il server che attenderà delle richieste sull'indirizzo IP indicato in configurazione, mentre la pagina HTML contiene due "interruttori", che invieranno due richieste indipendenti l'una dall'altra che, una volta ricevute da Arduino, verranno tradotte in segnali digitali (alto o basso) sulle due porte di Arduino utilizzate per controllare i relay.

Tutto il codice, HTML e Arduino, è stato preso online da diversi progetti già esistenti, adattati da me per funzionare sul mio progetto e per funzionare tra loro (essendo i progetti originari incompatibili fra loro).

Di seguito, una schermata e un video che mostrano l'interfaccia utente finale:



## Video

Di seguito, un video che dimostra il funzionamento del progetto nel suo stadio finale:

