

ITIS “G. e M. Montani” – Fermo

Indirizzo Informatica Abacus

anno scolastico 2012 / 2013

Progetto: SDS – Safe Domotic Shop

Sezione 5° A

Autore: Diana Bernabei

Ringraziamenti	4
Prefazione	5
Introduzione	5
Cenni teorici.....	6
Domotica	6
Arduino	6
Arduino Uno Rev3	6
Arduino Ethernet Shield	7
Raspberry PI.....	9
Progettazione software	11
Installazione sistema operativo nella scheda Raspberry PI.....	11
Collegamenti fondamentali	12
Primo avvio e configurazioni iniziali	12
Installazione LAMP (GNU/Linux – Apache – MySql – PHP).....	13
Configurazione della rete	14
Installazione client FTP	14
Installazione Motion (programma per la gestione delle webcam).....	15
Arduino e internet	16
Librerie Arduino.....	16
Procedure	16
Progettazione hardware.....	17
Installazione e funzionamento modellino.....	20
Spesa totale	21

La relazione e il progetto stesso è consultabile online all'indirizzo:

<http://safedomoticshop.altervista.org/>

Ringraziamenti

Un sentito ringraziamento va a tutti coloro che durante questi tre anni, mi hanno permesso di esprimere al meglio le mie passioni tecniche, di approfondirle senza limiti, attraverso dei concetti dell'informatica e dell'elettronica specifici a volte non insegnati in questo corso.

Si ringrazia quindi tutto l'apparato scolastico, dalla Preside Margherita Bonanni e il Vicepreside Luigi Sacchi, all'amministrazione, che mi hanno permesso di reperire tutte le componenti elettroniche.

Ma soprattutto ai professori per le tante consulenze date ma anche per le conoscenze che mi hanno tramandato, di seguito: Maura Rogante, Tiziana Magnante, Gino Tombolini, Giuseppe Capitani, Danila Angelici, Fabrizio Postacchini, Mauro Ruffini, Leonardo Paci, Meri Biancucci, Remo Mangiaterra.

Infine un grande ringraziamento va anche a tutti i miei compagni di classe, che mi hanno sopportato e supportato, in questi lunghi cinque anni.

Prefazione

"SDS – Safe Domotic Shop" è il titolo che si è scelto di assegnare a questo progetto d'esame di maturità per l'anno scolastico 2012/2013. Questa relazione inizierà analizzando appunto il titolo scelto, ovvero dando in primo luogo la definizione di "domotic" , in seguito i materiali che si sono utilizzati attraverso dei cenni teorici, per poi passare alla progettazione vera e propria dai componenti hardware a quelli software, fino ad arrivare alla costruzione del modellino che sarà esposto durante la fase orale dell'esame di maturità.

Introduzione

"SDS – Safe Domotic Shop" ha lo scopo di dimostrare come le conoscenze informatiche possano fondersi con quelle elettroniche, in aspetti ben importanti della vita quotidiana delle persone, sia da un lato della qualità della vita, che del risparmio, che della sicurezza.

Questo progetto consiste nella realizzazione di un apparecchiatura elettronica che permetta di gestire il proprio negozio ovunque ci si trovi ; per fruire di tutto ciò basta avvalersi di un qualsiasi dispositivo collegato alla rete Internet.

Ovviamente oltre all'apparecchiatura elettronica è stato anche sviluppato un sito Web che permette il controllo di tutti i sensori disponibili nel negozio e la gestione del video di sicurezza.

La possibilità di gestire un negozio o una casa a distanza non è fantascienza, viene già implementata, anche se a costi elevati e viene definita *domotica*.

La realizzazione della struttura "casa(o negozio)" domotica" non deve essere complicata ma aiutarci quindi nella semplificazione delle operazioni della vita quotidiana facilmente gestibili sia dal proprietario che da un utente qualsiasi.

Nel nostro caso per poterla implementare, si è scelto di utilizzare una scheda elettronica programmabile chiamata *Arduino* e il mini computer *Raspberry PI* facente parte degli Embedded Single Board Computer (SBC) cioè computer completi montati su una singola scheda, con microprocessore, memoria, input/output (I/O) e altre caratteristiche richieste a un computer funzionale.

Uno degli aspetti fondamentali di questo progetto non è stato solo quello di potere replicare la gestione domotica che già avviene ma di poter trovare un modo di progettazione a bassi costi e il più possibile flessibile dal punto di vista funzionale così da poter implementare tutti questi importanti concetti ad una vasta popolazione. Così la scelta è ricaduta sull'unione delle due schede:

- Arduino e l'SBC Raspberry PI; Arduino con poche decine di euro permette di gestire qualsiasi componente della casa e grazie alla sua natura open source, risulta facilmente programmabile consentendo una personalizzazione del software in base alle proprie esigenze.
- Raspberry PI con costi ridotti permette di creare un web server sufficientemente potente da permettere la gestione dell'intero sito web per il controllo dei sensori, attuatori e del video remoto.

Nel prossimo capitolo verranno illustrati i concetti teorici del progetto, quali: la domotica, Arduino, Raspberry PI e la tecnologia dei servizi utilizzati per far funzionare il webserver e la gestione dei video di sicurezza.

Cenni teorici

Domotica

La domotica è una scienza interdisciplinare che si occupa dello studio e delle tecnologie atte a migliorare la qualità della vita nella casa. Il termine domotica deriva dal latino "domus" che significa "casa".

Quest'area interdisciplinare richiede l'apporto di molte tecnologie e professionalità, tra le quali ingegneria edile, automazione, elettronica ed elettrotecnica, informatica e telecomunicazioni.

Più concretamente, la domotica è l'evoluzione tecnica dell'integrazione dei componenti dell'impianto elettrico "tradizionale" comunemente intesi come comandi luce, prese per prelievo energia, impianto antifurto, motorizzazioni tapparelle, tende, finestre, porte, portoni e cancelli, riscaldamento, climatizzazione, irrigazione, segnali audio e video, ecc... insomma, qualsiasi aspetto della vita quotidiana in cui è coinvolta l'elettricità e il suo controllo in locale o in remoto tramite l'informatica.

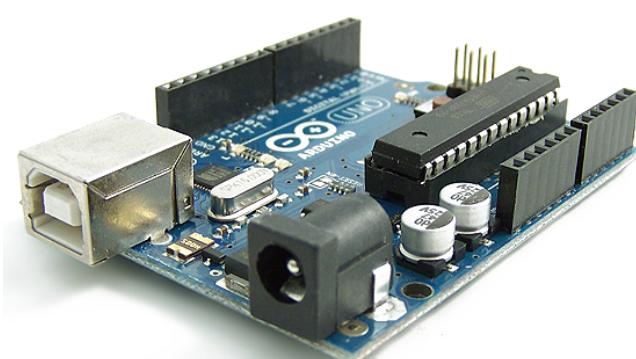
Spesso la domotica viene anche definita sotto l'espressione "casa intelligente", intendendo un ambiente opportunamente progettato tecnologicamente per rispondere in modo appropriato ai dovuti input senza l'intervento manuale dell'uomo. La gestione in locale o in remoto è quindi realizzata da opportuni software di interfaccia, acquisizione e di gestione delle risposte del sistema scritti dal programmatore.

Arduino

Arduino è stato creato da Olivetti nel 2005; esso è progettato e costruito interamente ad Ivrea (stabilimento di Scarmagno, quello dei famosi PC Olivetti) e ha conquistato un successo mondiale (100000 schede in tutti i Paesi, Corea compresa). La piattaforma è adottata, fra gli altri, dai laboratori di robotica del MIT e della Carnegie Mellon University, i due principali centri di ricerca mondiale in questo settore. E' adatta a molti impieghi, oltre che alla robotica: design, arte, programmazione evoluta. Arduino è presente all' Omar (?) dal 2010 (versioni Arduino Uno e Arduino Mega).

Aggiungi elenco

- **Arduino Uno Rev3**



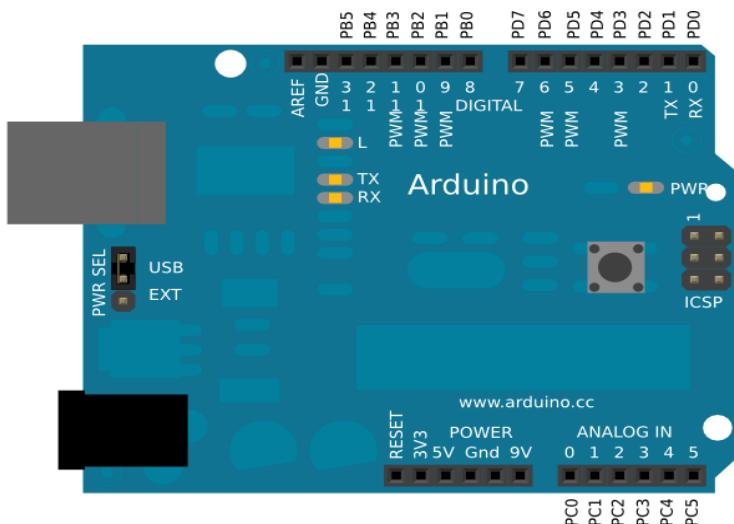
Arduino è una piattaforma hardware programmabile, con cui è possibile creare circuiti "quasi" di ogni tipo per molte applicazioni, soprattutto in ambito di robotica ed automazione. Si basa su un Microcontrollore della ATMega328. Un particolare molto importante è che ogni programma che si scrive su Arduino sarà naturalmente avviato a loop() finché non si toglie l'alimentazione dal dispositivo. Quando lo collegiamo ad una fonte di alimentazione (ad esempio la USB del

PC o anche una comunitissima Batteria da 9V) si accende e avvia il programma caricato. La scheda Arduino è in grado di interagire con l'ambiente in cui si trova ricevendo informazioni da una grande varietà di sensori. Anche se non si parla solo di sensori, Arduino può comandare luci, LED, motori e altri attuatori.

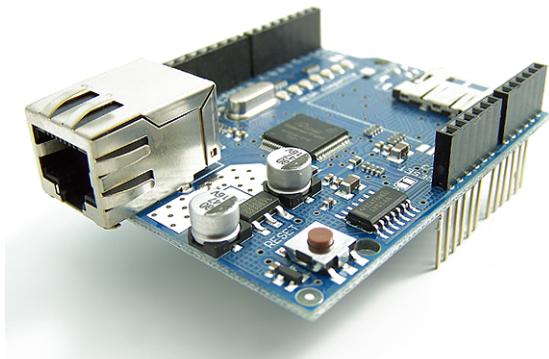
La scheda Arduino ha le seguenti caratteristiche:

Microcontrollore	ATmega328
Tensione di lavoro	5V
Tensione di ingresso raccomandata	7-12 V
Ingressi -Uscite	14
Ingressi analogici	6
Memoria Flash	32 KB
SRAM	2 KB
EEPROM	1 KB
Clock	16 MHz

Di seguito sono elencati i pin della scheda Arduino:



- **Arduino Ethernet Shield**



L'Arduino Ethernet Shield permette ad una scheda Arduino di collegarsi ad internet. Si basa sul chip ethernet Wiznet W5100. Il Wiznet W5100 implementa il protocollo ethernet con tutto lo stack TCP/IP e UDP; supporta fino a quattro connessioni simultanee. Lo shield è accompagnato dalla libreria Ethernet che consente di scrivere sketch che si collegano a Internet. Lo shield ethernet si collega ad una scheda Arduino tramite dei lunghi header che si estendono attraverso lo shield. Ciò mantiene intatto

il layout dei pin e permette di poter montare un altro shield sovrapposto. Grazie alla scheda Ethernet Shield possiamo controllare le cose da un PC, da un cellulare o qualsiasi altro dispositivo che può collegarsi a Internet. Ovviamente la scheda ha un **indirizzo MAC** e un **indirizzo IP**, parametri fondamentali per la comunicazione .

MAC – (MAC address)

L'indirizzo MAC detto anche *indirizzo fisico*, *indirizzo ethernet* o *indirizzo LAN*, è un codice di 48 bit (6 byte) assegnato in modo univoco dal produttore ad ogni scheda di rete ethernet o wireless prodotta al mondo.



A MAC address, or *Media Access Control* address, is a 48 or 64 bit address associated with a network adapter. While IP addresses are associated with software, MAC addresses are linked to the hardware of network adapters. For this reason, the MAC address is sometimes called the hardware address, the burned-in address (BIA), or the physical address. MAC addresses are expressed in hexadecimal notation in the following format: 01-23-45-67-89-AB, in the case of a 48-bit address, or 01-23-45-67-89-AB-CD-EF, in the case of a 64-bit address. Colons (:) are sometimes used instead of dashes (-).

MAC addresses are often considered permanent, but in some circumstances, they can be changed.

IP -(Internet Protocol)

Un **indirizzo IP** è un'etichetta numerica che identifica univocamente un dispositivo (host) collegato a una rete informatica che utilizza l'Internet Protocol come protocollo di comunicazione. Un indirizzo IP svolge principalmente due funzioni: identificare un dispositivo sulla rete e di conseguenza fornirne il percorso per la sua raggiungibilità da un altro terminale o dispositivo di rete in una comunicazione dati a pacchetto.

Più esattamente l'indirizzo IP viene assegnato a una *interfaccia* (ad esempio una scheda di rete) che identifica l'host di rete, che può essere un personal computer, un palmare, un router, elettrodomestici in generale, ecc. Va considerato, infatti, che un host può contenere più di una interfaccia: ad esempio, un router ha diverse interfacce (minimo due) per ognuna delle quali occorre un indirizzo IP.

L'indirizzo IP è un indirizzo di livello 3 del modello ISO-OSI che si rende necessario per l'*instradamento indiretto* tramite l'omonimo protocollo di rete IP ovvero per interconnettere (*internetworking*) più sottoreti diverse all'interno del paradigma TCP-IP della rete Internet. Si contrappone all'indirizzo fisico di livello 2 o indirizzo MAC utilizzato invece per l'*instradamento diretto* all'interno di una sottorete locale.



IP address is short form for **Internet Protocol (IP) address**.

An IP address is an identifier for a computer or device on a TCP/IP network. Networks using the TCP/IP protocol route messages based on the IP address of the destination.

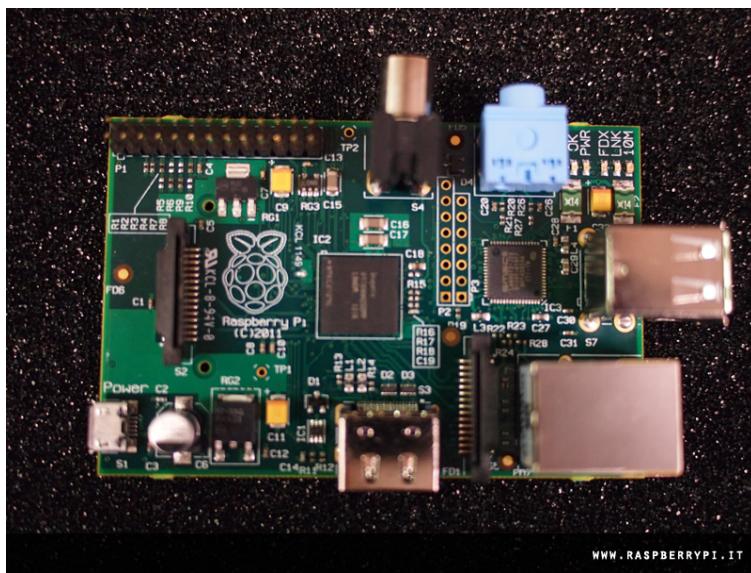
The format of an IP address is a 32-bit numeric address written as four numbers separated by periods. Each number can be zero to 255. For example, 1.160.10.240 could be an IP address.

An IP address can be static or dynamic. A static IP address will never change and it is a permanent Internet address. A dynamic IP address is a temporary address that is assigned each time a computer or device accesses the Internet.

La scheda Arduino ha le seguenti caratteristiche:

Microcontrollore	ATmega328
Tensione di lavoro	5V
Tensione di ingresso raccomandata	7-12 V
Ingressi -Uscite	14
Ingressi analogici	6
Memoria Flash	32 KB
SRAM	2 KB
EEPROM	1 KB
Clock	16 MHz
Controllo Ethernet integrato W5100 TCP/IP	
Power Over Ethernet ready Magnetic Jack	
Slot Micro SD card	

Raspberry PI



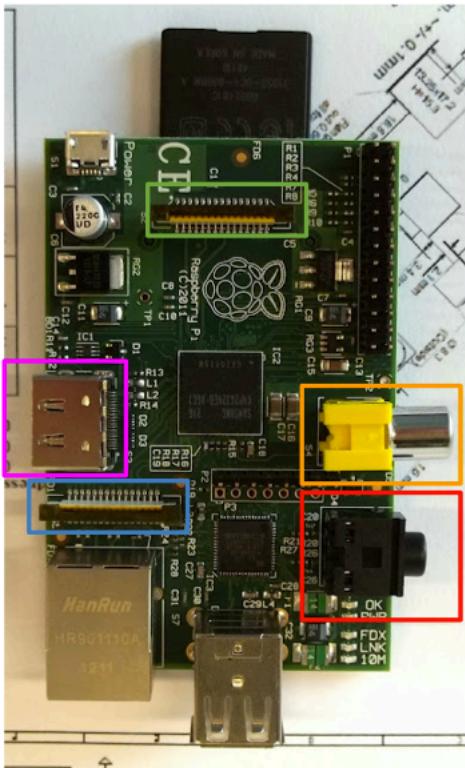
La Raspberry PI (raspi o RPi) è un **piccolo computer** in una scheda dalle dimensioni di una carta di credito (85,60mm x 53,98mm).

Dispone già del hardware necessario per funzionare compreso la parte ethernet e come in un PC basta aggiungere alcune periferiche come tastiera, mouse USB e un monitor con ingresso HDMI.

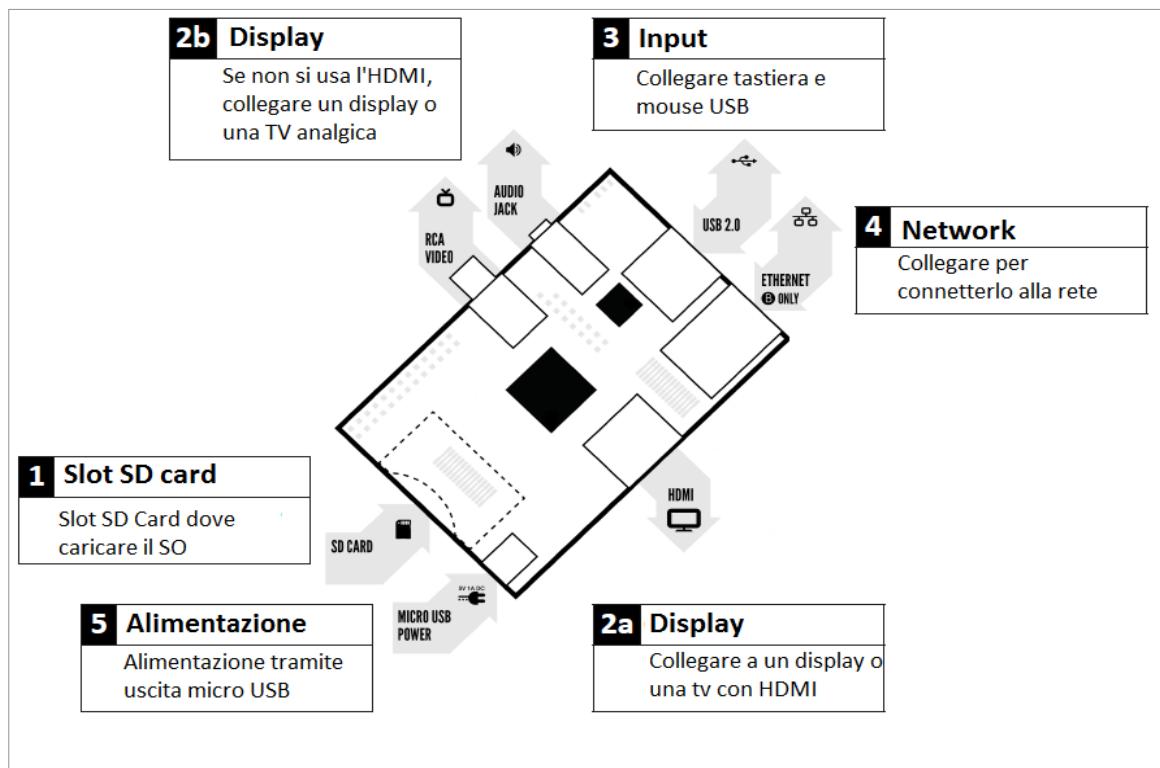
Per farlo funzionare serve un sistema operativo da installare in una SD. Essendo un computer servono anche dei programmi. Questi posso essere installati nella SD e/o in cloud.

Dispone di alcune General Purpose Input/Output ([GPIO](#)) per interfacciarsi a basso livello con altri dispositivi o componenti elettronici.

Questo rientra in quello che viene definito **embedded Single Board Computer (SBC)**



- interfaccia **DSI** per pannello LCD
- uscita **HDMI 1.3a** con connettore tipo A
- uscita Video Composite con connettore **RCA**
- connettore un modulo camera con specifiche **Camera Serial Interface MIPI CSI-2**.
(foto nella slide successiva)
- **USCITA audio su connettore da 3.5mm stereo jack**



Progettazione software

La parte fondamentale del progetto è il software, il quale permette di gestire gli upload continui dei sensori gestiti da Arduino, nel database gestito invece da Raspberry e il tutto raggiungibile da chiunque in qualsiasi parte del mondo.

Installazione sistema operativo nella scheda Raspberry PI

Prima di connettere all'alimentazione Raspberry Pi bisogna installare sulla scheda SD un sistema operativo valido tra quelli indicati sul sito RaspberryPi.org, in questo caso Raspbian.

Per l'installazione della distribuzione ufficiale **Raspbian** si può procedere scaricando l'ultima versione direttamente da <http://www.raspbian.org/RaspbianInstaller>.

Il file scaricato avrà un nome tipo: 2013-02-09-wheezy-raspbian.img e avrà una dimensione di circa 1,94 GB.

A questo punto è necessario formattare la SD e copiare l'immagine del sistema operativo nella SD. (le seguenti operazioni sono state fatte con un sistema operativo MACOSX)

- collegare la scheda sd;
- avviare Utility Disco che si trova nella cartella Applicazioni → Utility ;
- selezionare la scheda SD e cliccare su Inizializza
(nel tipo di formattazione selezionare: MS-DOS (FAT));
- aprire il terminale e digitare:

```
df -h
```

questa chiamata consente di visualizzare tutti i dispositivi collegati, dove si può notare la presenza dell'SD inserita con il nome “SENZA TITOLO”;

- sul terminale scrivere:

```
sudo diskutil unmount /dev/disk1s1
```

per eseguire l'unmount dell'SD (poichè si sta utilizzando il comando “sudo” verrà richiesto di inserire la password di amministratore per poter eseguire il comando);

- al termine comparirà l'avviso che il volume è stato “smontato” a questo punto bisogna procedere copiando l'immagine del sistema operativo scaricato, quindi digitare il comando:

```
sudo dd bs=1m if=~/Downloads/2013-02-09-wheezy-raspbian.img  
of=/dev/rdisk1
```

dove: 2013-02-09-wheezy-raspbian.img e l'indice **1** si riferisce al nome che il mio sistema operativo ha assegnato al volume “SENZA TITOLO”, l'SD (anche in questo caso verrà richiesto di inserire la password da amministratore per eseguire il comando);

- l'operazione durerà qualche minuto e al termine non arriverà un avviso ma un semplice ritorno alla riga di comando;
- ora espellere la scheda SD che ha il nome “SENZA TITOLO” da terminale digitando il comando:

```
sudo diskutil eject /dev/rdisk1
```

A questo punto la scheda SD è pronta.

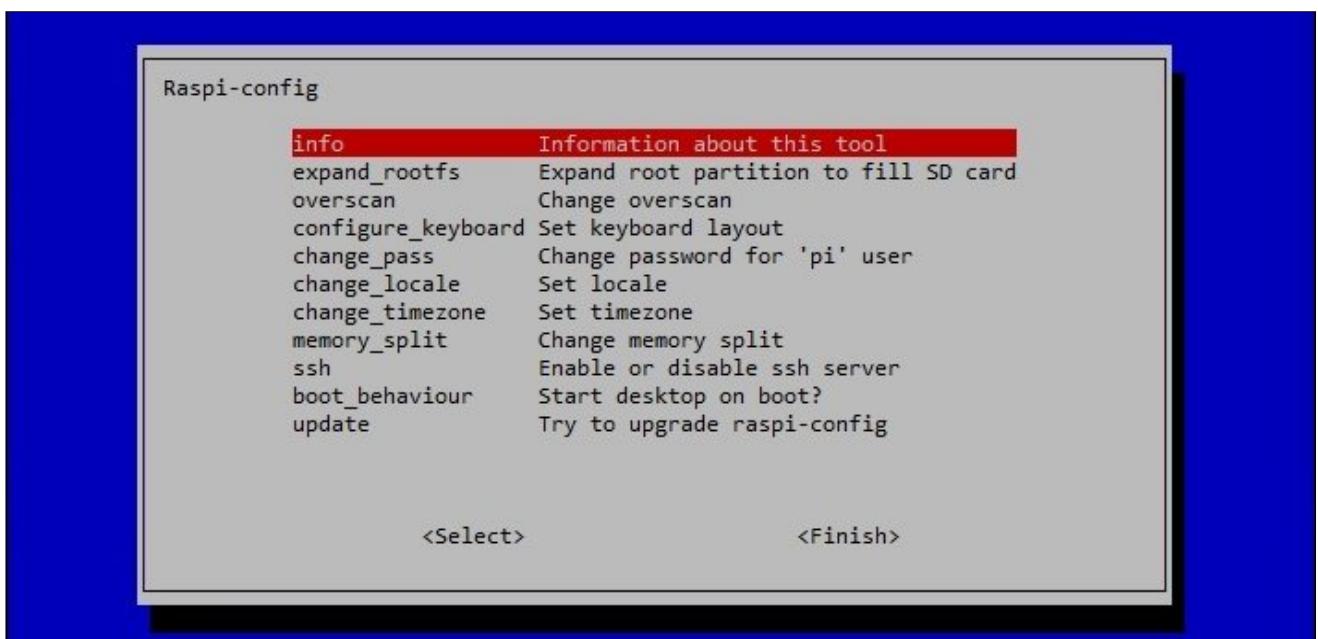
Collegamenti fondamentali

- Alimentazione mediante micro;
- Scheda SD con sistema operativo da almeno 2GB;
- Connettore HDMI (su HDMI viene convogliato anche l'audio);
- LAN: cavo Ethernet da connettere ad un hub o switch;
- USB: connessione mouse e tastiera (o un eventuale hub USB che non necessita dell'alimentazione).

Primo avvio e configurazioni iniziali

Una volta fatti tutti i collegamenti e installato il sistema operativo, basta collegare l'alimentazione e la scheda si accenderà automaticamente senza il bisogno di premere alcun pulsante.

Completata l'accensione comparirà la seguente finestra, dove si possono configurare le caratteristiche più basilari della scheda.



Posizionarsi con le frecce direzionali a `expand_rootfs` e premere invio, questa chiamata servirà per aumentare la partizione del sistema operativo a tutta la grandezza della SD e non solo quella strettamente necessaria a esso.

Finita quest'operazione, si ritornerà alla schermata Raspi-config.

Adesso scendere fino a `ssh` e premere invio per abilitare il protocollo di rete ssh, che permette di stabilire una sessione remota cifrata tramite un'interfaccia a riga di comando con un altro host.

(Quando si farà accesso da un host all'IP del server la prima volta è consigliabile disattivare la chiave RSA scrivendo: `ssh-keygen -R {IP SERVER}`)

Completata quest'operazione, posizionarsi su `Finish` e premere invio per riavviare il sistema.

Finito di riavviarsi, si avrà sulla riga di comando `raspberry login:_`. Per effettuare l'accesso basta scrivere: `pi`, e come password: `raspberry` (questi sono i dati per il login predefiniti, ovviamente si possono cambiare digitando `sudo raspi-config`).

Finito l'avvio, avremo: `pi@raspberrypi ~ $`

Installazione LAMP (GNU/Linux – Apache – MySql – PHP)

Con il termine LAMP si definisce una delle più comuni unioni di programmi per la creazione di webserver cioè:

Linux – sistema operativo

Apache – webserver http

MySql – server database

PHP – linguaggio di programmazione web

Ogni configurazione e installazione di questi software è fatta da riga di comando, questo modo non è uno dei più semplici ma ha sicuramente svariati vantaggi, come il completo controllo delle configurazioni e l'installazione personalizzata del server. Utilizzando Raspberry a riga di comando, il processore non sarà occupato a caricare la grafica e questo rende il server più veloce sia in fase di accensione sia di utilizzo.

Procedere ora all'installazione nella scheda SD precedentemente configurata del pacchetto di programmi LAMP.

- Per prima cosa fare un upgrade e update del sistema:

```
sudo apt-get upgrade  
sudo apt-get update
```

- Dopo l'aggiornamento procedere con il seguente ordine all'installazione dei programmi per far diventare Raspberry pi un webserver:

```
sudo apt-get install apache2  
sudo apt-get install mysql-server  
sudo apt-get install php5  
sudo apt-get install php5-mysql
```

Finito di installare tutti i programmi riavviare Raspberry scrivendo:

```
sudo reboot
```

Ora si può installare il client PhpMyAdmin scrivendo:

```
sudo apt-get install phpmyadmin
```

Completata l'installazione cambiare il file di configurazione di apache, e scrivere:

```
sudo nano /etc/apache2/apache2.conf
```

e scrivere:

```
Include /etc/phpmyadmin/apache.conf
```

Uscire dal file premendo ctrl + x e confermare la modifica scrivendo “yes”.

Riavviare il server apache per verificare le modifiche fatte:

```
sudo /etc/init.d/apache2 restart
```

Successivamente modificare i permessi di esecuzione della cartella www dove saranno caricate le pagine web:

```
cd ..  
cd ..  
cd var  
sudo chmod 777 www
```

Caricare il sito e in seguito aprire il pannello phpmyadmin scrivendo sul browser IP/phpmyadmin Entrando con le referenze impostate in precedenza e caricare i database.

Configurazione della rete

Per cambiare l'IP di Raspberry e farlo diventare statico in modo tale che si possa accedere in qualsiasi momento al server, è necessario cambiare il file di configurazione della rete:

```
sudo nano /etc/network/interfaces
```

(per uscire dell'editor di testo basta premere ctrl+x e scrivere "yes" per confermare il salvataggio delle modifiche effettuate)

Dove comparirà il seguente testo:

```
auto lo

iface lo inet loopback
iface eth0 inet dhcp

allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp
```

Ora basta cambiare la riga `iface eth0 inet dhcp` con :

```
iface eth0 inet static
    address     INDIRIZZO_IP
    netmask     NETMASK_NETWORK
    gateway    GATEWAY_NETWORK
```

A questo punto correggere il file che contiene il nameserver scrivendo:

```
sudo nano /etc/resolv.conf
```

E modificare il contenuto del file con:

```
nameserver 8.8.4.4
nameserver 8.8.8.8
```

Installazione client FTP

Ora procedere con l'installazione del client FTP per potersi connettere da remoto al server senza effettuare l'accesso via SSH.

Per installare il programma basta scrivere:

```
sudo apt-get install proftpd-basic
```

Alla richiesta della modalità di funzionamento, scegliere standalone.

Finito di installare il programma stoppare il "demone" scrivendo:

```
sudo /etc/init.d/proftpd stop
```

Per modificare alcune caratteristiche del client (ad esempio il ServerName) aprire il file:

```
sudo nano /etc/proftpd/proftpd.conf
```

salvare e riavviare il server:

```
sudo /etc/init.d/proftpd start
```

Installazione Motion (programma per la gestione delle webcam)

Per installare motion basta scrivere sulla riga di comando:

```
sudo apt-get install motion
```

Caratteristiche di Motion

- è un programma scritto in C
- è rilasciato sotto GPL (General Public License)
- si puo' liberamente scaricare dal sito <http://motion.sourceforge.org>
- è un programma a linea di comando
- la configurazione di motion avviene modificando il file motion.conf con un editor di testo es. vi, kate o nano

Il file di configurazione motion.conf

Finita l'installazione bisogna cambiare il file di configurazione, scrivendo:

```
sudo nano /etc/motion/motion.conf
```

Se si usano più telecamere contemporaneamente, è necessario mettere piu' thread nel file motion.conf (es. Se si usano due telecamere bisogna avere un file motion.conf e due file thread per un totale di tre file)

Le modifiche fatte al file di configurazione sono:

```
daemon on
framerate 100
output_normal off
ffmpeg_cap_new off
webcam_maxrate 100
webcam_localhost off
```

Adesso abilitare l'avvio automatico cambiando la parola 'no' con 'yes' nel file:

```
cd etc
cd default
sudo nano motion
```

Arduino e internet

La gestione del negozio fatta attraverso il sito Web, avviene sfruttando la classica comunicazione alla base di Internet, ovvero il modello Client-Server, dove da una parte si ha un client che effettua la richiesta di ricevere particolari informazioni, mentre dall'altra parte si ha un server che risponde a questa richiesta con altrettante informazioni.

In questo caso, il sito Web funge da connessione tra il circuito gestito da Arduino, e il database MySQL caricato nel webserver di Raspberry. Abbiamo infatti il client (Arduino) che aggiorna in modo ciclico tutti i sensori presenti nel codice di caricato nella scheda Arduino come funzioni, e il server (Raspberry PI) che riceve e gestisce i dati ricevuti, in particolare quest'azione viene eseguita nel file db_upload.php.

Si analizza ora il codice caricato su Arduino composta da librerie, procedure eseguite in un ciclo continuo di esecuzione chiamato loop().

Librerie Arduino

Le librerie utilizzate sono SPI.h e Ethernet.h, entrambe usate per la gestione della shield Ethernet (in particolare la definizione delle variabili byte mac[] e byte ip[]).

Procedure

Le procedure utilizzate sono le seguenti:

- void accedi(), gestisce il pin di accesso (0 – non eseguito, 1 – eseguito);
- void led(), gestisce il controllo delle luci;
- void prossimità(), esamina gli eventuali movimenti captati dal sensore Pir, nel suo raggio d'azione;
- void porta(), gestisce il sensore magnetico composto da una parte fissa sulla parete e una "mobile" sulla porta, quando queste due parti fanno contatto, la porta quindi è chiusa, la variabile gestita da questa funzione è zero, altrimenti è 1;
- void invia_risultati(), una volta rilevati i valori dalle precedenti procedure invia i valori alla pagina web "db_update.php", che a sua volta li memorizzerà nel database.

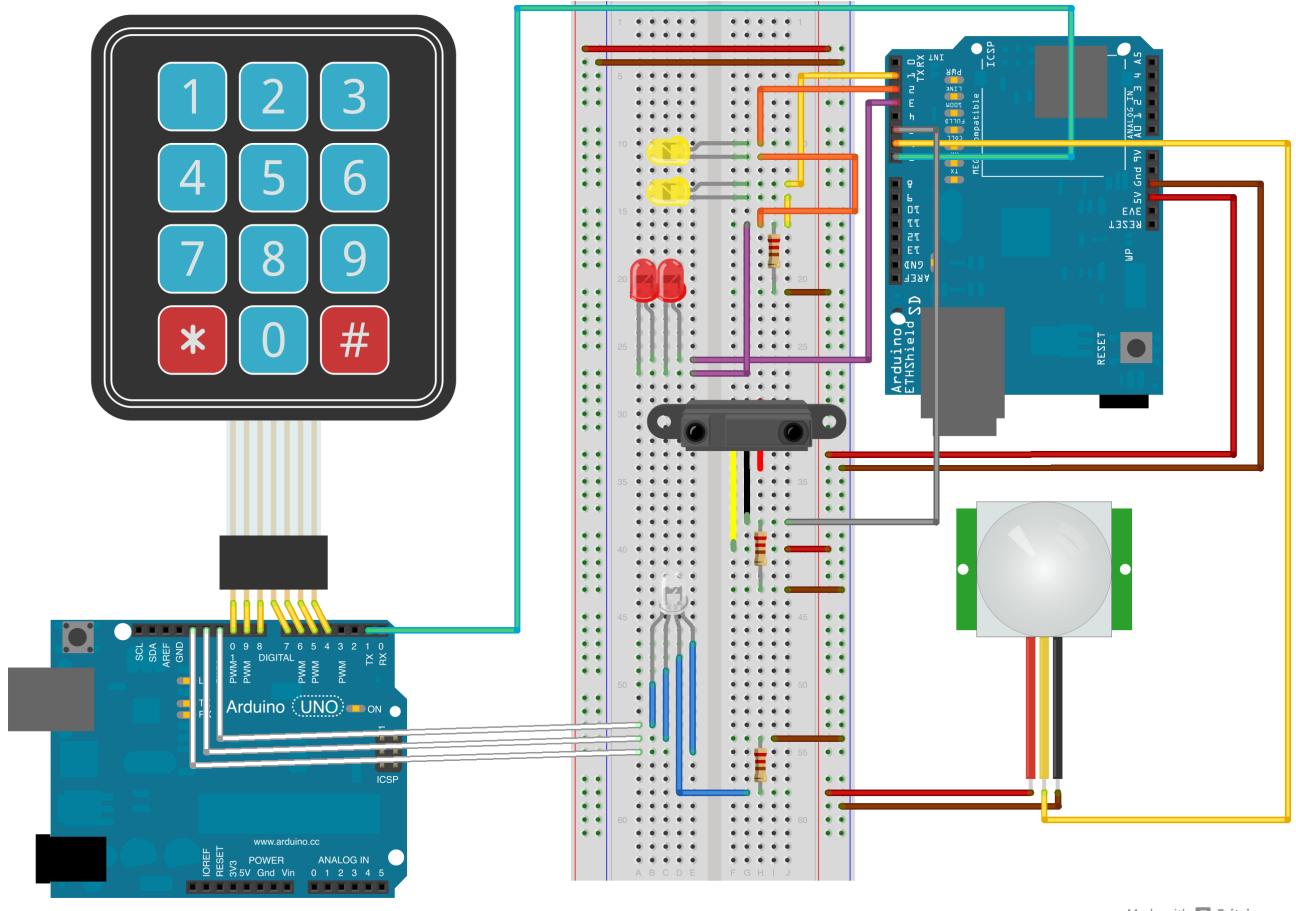
Tutti i programmi di Arduino, detti "sketch", sono caratterizzati dalla particolarità di avere una sezione apposita definita setup() in cui devono essere settate le porte utilizzate dalla board definendone il funzionamento in Input o in Output.

In questo caso sono stati impostati i led come output e come input il sensore magnetico, il sensore PIR e la variabile d'accesso gestita da un secondo Arduino perennemente in ascolto del codice inserito sul tastierino.

Lo scheletro del software di Arduino è dato da uno switch che attraverso il valore assunto da una precisa variabile definita scelta consente di svolgere determinate funzioni o meno.

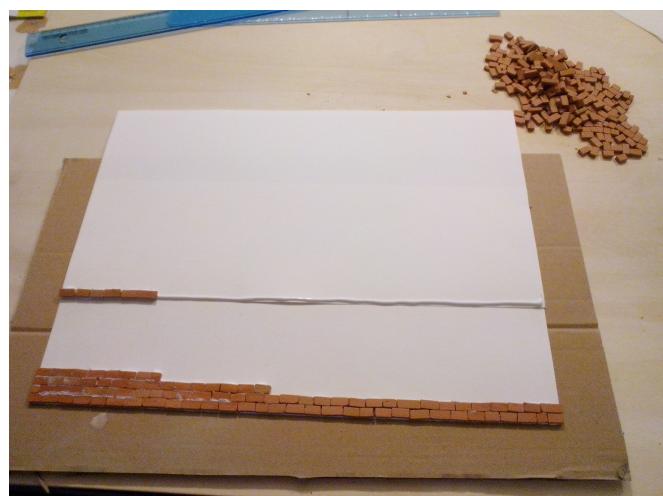
Progettazione hardware

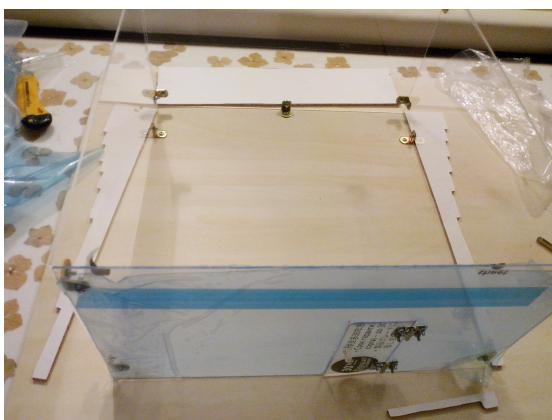
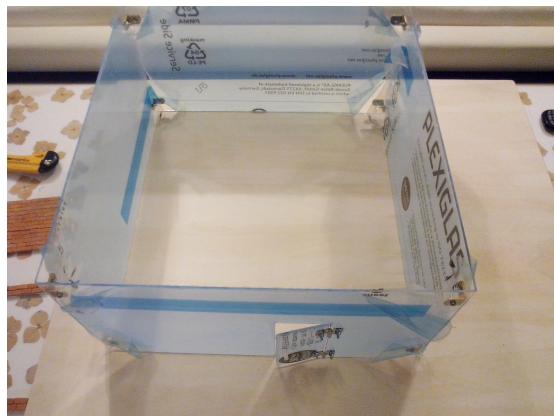
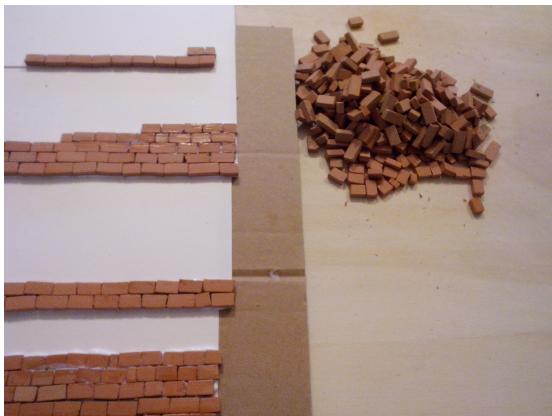
Circuito :

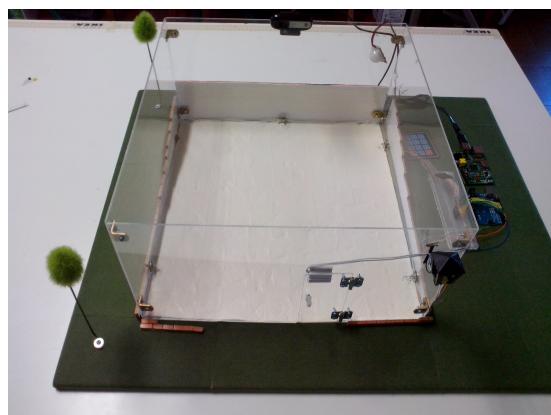
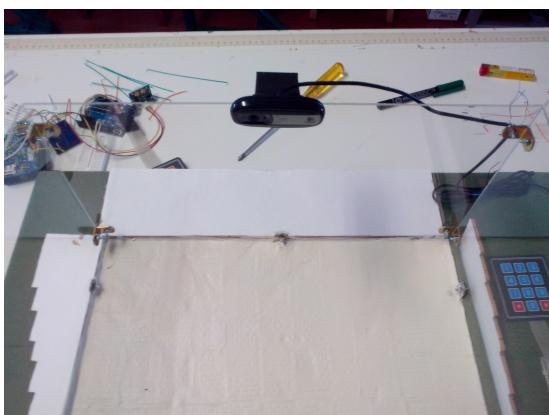
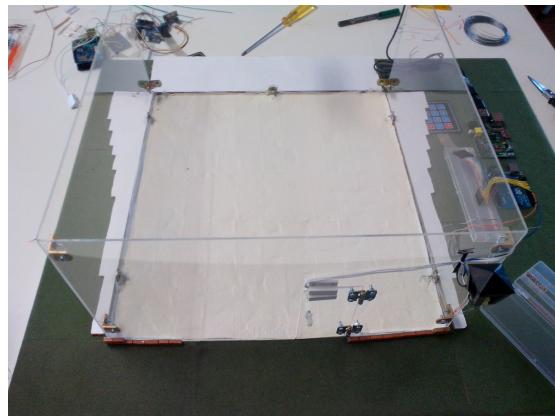
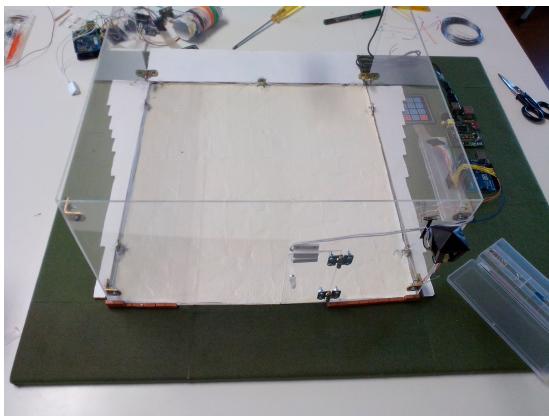


Made with Fritzing.org

Foto fasi costruttive del modellino:







Installazione e funzionamento modellino

L'installazione del modellino consiste nel cambiare l'IP di Arduino e l'IP di Raspberry con la configurazione della nuova rete.

I file da modificare sono:

1. ip di Raspberry e ip di Arduino nello sketch di Arduino (e subnet mask)

Installare l'IDE di Arduino dal sito <http://arduino.cc/en/Main/Software> e scaricare il codice di Arduino dal sito del progetto nella sezione "DOWNLOAD". Aprire il file arduino_principale_con_ETH_shield.ino e modificare gli ip in queste due parti: all'inizio del codice

```
// Dichiarazioni variabili  
  
// Arduino Ethernet Shield  
byte mac[] = { 0x90, 0xA2, 0xDA, 0x0d, 0x3E, 0xDA };  
byte ip[] = {192,168,1,3}; //IP Arduino  
byte subnet[] = { 255, 255, 255, 0 };  
char Data_RX; // Stringa di appoggio per la lettura  
String msg = ""; //Stringa di comunicazione
```

alla fine del codice nell'ultima e più importante funzione

```
void invia_risultati(EthernetClient pc_client){  
    // Attendo che tutti i byte siano letti quando Data_RX contiene il carattere di nuova line capisco tutti i byte sono stati letti  
  
    // Invio la risposta al client invio lo status code  
    pc_client.println("HTTP/1.1 200 OK");  
  
    // imposto il data type  
    pc_client.println("Content-Type: text/html");  
    pc_client.println();  
  
    // Vengono spediti i valori di stato al database attraverso una pagina Web update.php  
    pc_client.print("<html><head>");  
  
    // Cambiare ip di raspberry  
    pc_client.print("<meta http-equiv=\"refresh\" content=\"0; url=http://192.168.1.5/db_update.php?\"");  
  
    pc_client.print("scelta="+String(scelta)+"&app);  
    pc_client.print("</> ");  
    pc_client.print("</head></html>");  
  
    // Aspetto 1 ms affinche la risposta giunga al browser del clients  
    delay(1);  
}
```

2. ip statico di Raspberry (e subnet mask)

Per modificare l'ip statico di Raspberry se non è già stato fatto in precedenza (eventuale installazione del sistema operativo e configurazioni) bisogna collegare Raspberry ad uno schermo HDMI o tramite un convertitore HDMI to VGA ad un normale monitor di pc desktop con uscita VGA e procedere come detto nella sezione Progettazione software/Configurazione di rete.

3. file php per la configurazione degli ip di tutte le pagine (/var/www/pred.php), modificabile effettuando prima la registrazione, in seguito il login nel sito di presentazione del progetto <http://safedomoticshop.altervista.org/> .

Una volta configurati gli ip, il controllo del modellino è gestito tramite un sito web caricato nel webserver, quindi accessibile all'IP di Raspberry o semplicemente dal sito di presentazione del progetto nella sezione "PANNELLO DI CONTROLLO", dove dopo essersi autenticati in un pannello di login gestito con PHP, MySQL e database SQL, si entra nel vero e proprio pannello di controllo.



Ciao admin. Per effettuare il logout [clicca qui](#).

Pannello di controllo
Seleziona l'icona relativa alla sezione che vuoi gestire

	ACCESSO
	LUCI
	PORTE E FINESTRE
	VIDEOSORVEGLIANZA
	MOVIMENTO

ACCESSO

Accesso NON effettuato

LUCI

Principale
Tavolo
Davanti SX

PORTE

chiuso

VIDEO SORVEGLIANZA

PIR

Dove per ogni sensore c'è una sua sezione raggiungibile cliccando le icone relative alle sezioni desiderate nel pannello di sinistra che rinvierà al frame giusto del pannello di destra (rettangolo con il bordo tratteggiato).

Spesa totale

Componente	Prezzo
Arduino UNO x 2	23 € x 2
Arduino Ethernet Shield	42 €
Raspberry PI	30 €
Webcam Logitech C170	17 €
Breadboard	6 €
Sensore PIR	8 €
Sensore magnetico	2 €
Tastierino	2 €
LED RGB	2,50 €
Cavo USB A/MICRO B	10€
Cavo USB A/B x 2	2 € x 2
Cavo ethernet x 2	6 € x 2
Prolunga USB	8 €
Scheda SD 8GB	8 €
HUB USB 7 porte	15 €
Jumper per collegamento	0 € (recuperati da un vecchio cavo Ethernet)
TOTALE	212,5 €
Spese per la costruzione del modellino	30 €
TOTALE COMPLESSIVO	242,5 €