

“AdvancedMirror”

Specchio Multifunzionale

By Ferrario Christian

Wednesday, February 21, 2018
8:49 am

UPCOMING

Dr's appointment - Fast from mid...
Feb 2 at 4:00 AM
...tion at Nobu - London
...:00 AM
... Buenos Aires Arge-
...
... Chataway



5°

9 am	☁	2°	45%	Wind
12 pm	☁	2°	45%	Wind
3 pm	☁	2°	45%	Wind
Thu	☁	2°	45%	Wind
Fri	☁	2°	45%	Wind
Sat	☁	2°	45%	Wind

Speaker's Open Profile List
Simon Nelson, Doug, 10 Follows on LinkedIn
@The Journal, Daniel Hays (journalist)
@Doug Nelson - @dougnelson

2	Feed	1
0	Feed	1
2	Feed (10)	3
4	Feed	1
3	Feed	2
4	Feed	2
1	Feed	2
1	Feed	2
1	Feed	2

“AdvancedMirror”

Il mio elaborato tratta di un dispositivo domotico atto a migliorare la qualità della vita, puntando sulla gestione degli impegni quotidiani, e sull'integrazione del dispositivo nel contesto di una casa domotica, rendendolo in grado di interagire con gli altri dispositivi della casa.

Nella pratica il dispositivo è uno specchio intelligente, che deve il suo funzionamento ad uno specchio a due vie, ovvero una superficie specchiante da un lato, mentre trasparente dall'altro.

Grazie al comportamento di questo tipo di specchio è possibile mostrare delle informazioni all'utente attraverso di esso, mantenendo riflettente la superficie dietro la quale lo sfondo dell'applicativo è nero.

Il dispositivo permette di gestire gli impegni giornalieri, mostrando il calendario personale, le previsioni meteo, lo stato del traffico sul percorso stradale per raggiungere il luogo di lavoro, gli orari dei mezzi di trasporto...

Con il dispositivo si avrà accesso ad una applicazione smartphone per gestire i moduli visualizzati sullo specchio, in modo da dare la possibilità all'utente di scegliere i moduli da visualizzare in base a quali di essi ritiene utili, oltre che aggiungere altri moduli oltre a quelli di default, come ad esempio l'andamento di alcuni mercati finanziari, le notizie del giorno, i dati trasmessi da dispositivi di fitness.

Inoltre grazie ad un'integrazione con Amazon Alexa, l'assistente vocale di Amazon, sarà possibile gestire i dispositivi domotici comunicando con lo specchio, oppure attraverso script automatizzati, ad esempio accendendo luci, gestire le tapparelle, i climatizzatori, allarmi e telecamere.



Sommario

“AdvancedMirror”	2
Tecnologie Utilizzate	4
Sito Web	5
Aspetto Logico	5
Il sito web	5
Aspetto Tecnico	7
Sicurezza	9
Dispositivo.....	12
Struttura Logica	12
Struttura Tecnica	12
Architettura di rete.....	15
Applicazione.....	15

Tecnologie Utilizzate

- **Node.js:** runtime di JavaScript multiplatforma orientato agli eventi per l'esecuzione di codice JavaScript, costruita sul motore JavaScript V8 di Google Chrome.
- **HTML:** HyperText Markup Language (traduzione letterale: linguaggio a marcatori per ipertesti), comunemente noto con l'acronimo HTML, è un linguaggio di markup.
- **CSS:** linguaggio usato per definire la formattazione di documenti HTML, XHTML e XML, ad esempio i siti web e relative pagine web.
- **PHP:** un linguaggio di scripting interpretato, originariamente concepito per la programmazione di pagine web dinamiche.
- **MySQL:** un relational database management system (RDBMS).
- **Android Studio:** è un ambiente di sviluppo integrato (IDE) per lo sviluppo di applicazioni per la piattaforma Android, offre la possibilità di programmare applicazioni Android in linguaggio JAVA.
- **JAVA:** un linguaggio di programmazione ad alto livello, orientato agli oggetti, che si appoggia sull'omonima piattaforma software di esecuzione, specificamente progettato per essere il più possibile indipendente dalla piattaforma hardware di esecuzione.
- **Raspberry PI:** un single-board computer sviluppato nel Regno Unito dalla Raspberry Pi Foundation, progettato per ospitare sistemi operativi basati sul kernel Linux.
- **AWS Lambda:** una piattaforma di calcolo "serverless" guidata dagli eventi fornita da Amazon come parte di Amazon Web Services. È un servizio di calcolo che esegue codice in risposta ad eventi e automaticamente gestisce le risorse richieste dal codice di programmazione.
- **JSON:** JavaScript Object Notation è un formato file standard adatto all'interscambio di dati fra applicazioni client/server.
- **API di terzi:** application programming interface, un insieme di procedure atte all'espletamento di un dato compito; spesso tale termine designa le librerie software di un linguaggio di programmazione, in questo caso sono librerie offerte da un servizio esterno per poter comunicare con un servizio offerto da terzi.

Sito Web

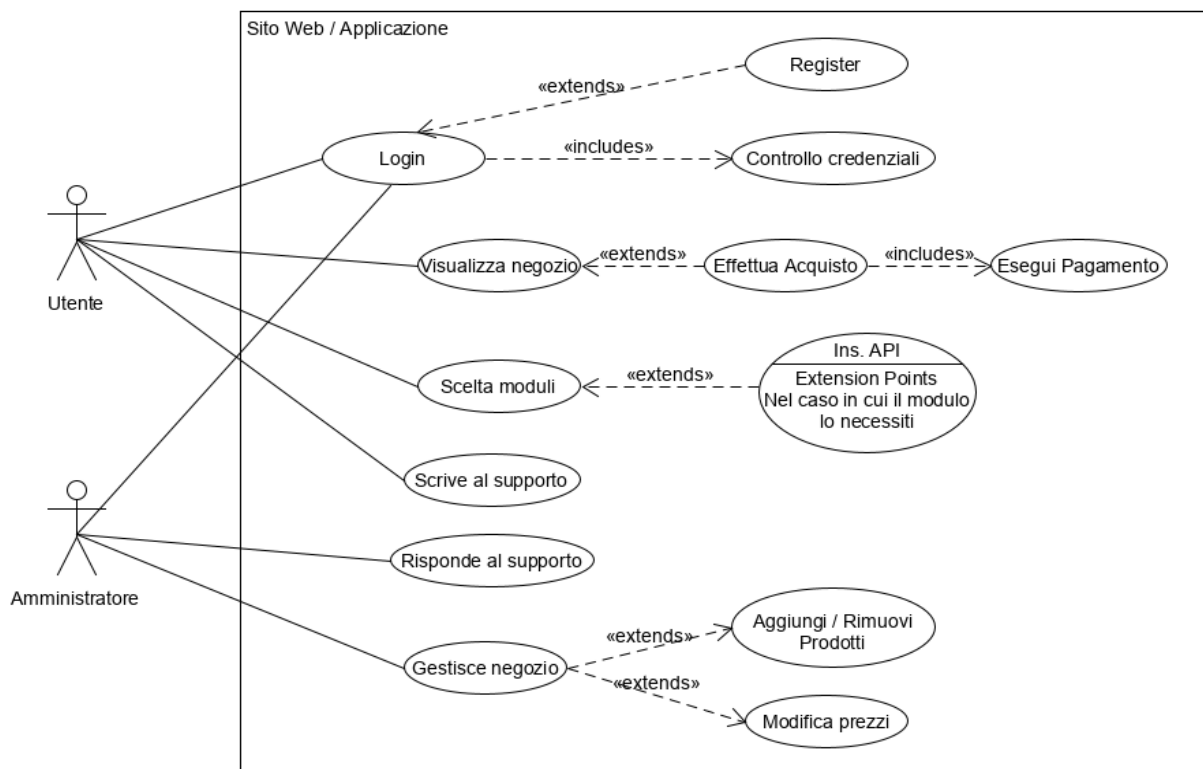
Aspetto Logico

Il sito web del prodotto è **composto da 7 pagine**:

- La pagina **home**, che viene mostrata all'apertura del sito, in cui è presentato il dispositivo attraverso alcune foto e le sue funzionalità, oltre che a delle semplici istruzioni per l'installazione.
- La pagina per effettuare il **login** o la **registrazione**, che permette di creare un account nel caso in cui l'utente ancora non lo abbia, o di effettuare l'accesso con il proprio profilo. Avere un account è necessario per comprare il prodotto e gestire le sue funzionalità.
- La pagina del **negozio**, in cui è possibile acquistare lo specchio, ed eventualmente altri prodotti domotici compatibili con esso, come luci controllabili da remoto per la casa o altri prodotti simili.
- Il **carrello**, dove verrà mostrato un riepilogo dei prodotti che stanno per essere acquistati, e dove si dovrà confermare l'ordine ed effettuare il pagamento, rigorosamente dopo aver effettuato l'accesso con il proprio account.
- La pagina per la **gestione** delle **API Key**, in cui è possibile inserire le chiavi API, per gestire i moduli che ricevono dati dall'esterno, come i calendari Google, le mappe, Spotify..., in modo da collegare i moduli dello specchio con i propri account personali ai vari moduli dedicati,
- La pagina per la gestione dei **moduli**, in cui è possibile scegliere i moduli da visualizzare sullo specchio, in base agli interessi dell'utente.
- La pagina del **supporto**, attraverso la quale l'utente potrà segnalare eventuali problemi tecnici o di installazione, al quale il supporto tecnico potrà rispondere per aiutare i clienti.

Per indicare quali funzioni ha a disposizione l'utente ho utilizzato un diagramma dei casi d'uso, il quale è composto da 3 parti:

- **L'attore**, che rappresenta un ruolo coperto da un'entità interagente col sistema (ovvero utenti umani, altri sistemi software, dispositivi hardware...)
- **Il caso d'uso**, che rappresenta una funzione o servizio offerto dal sistema a uno o più attori
- **La relazione**, che può essere di 3 tipi:
 - **Associazione**, Un attore può essere associato a un qualsiasi numero di casi d'uso, e viceversa, e implica uno scambio messaggi fra attori e use case associati
 - **Inclusione**, indica che la funzione rappresentata da uno dei due casi d'uso (quello alla base della freccia) include completamente la funzione rappresentata dall'altro
 - **Estensione**, indica che la funzione rappresentata dal caso d'uso "estendente" (alla base della freccia) può essere impiegata nel contesto della funzione "estesa"



Aspetto Tecnico

Il sito è **realizzato** utilizzando **HTML**, **CSS** e **PHP**. Tramite PHP inoltre le pagine sono collegate con un database relazionale **MySQL**, in cui i dati verranno salvati sotto forma di tabelle strutturate composte da righe e colonne.

Le tabelle del database sono:

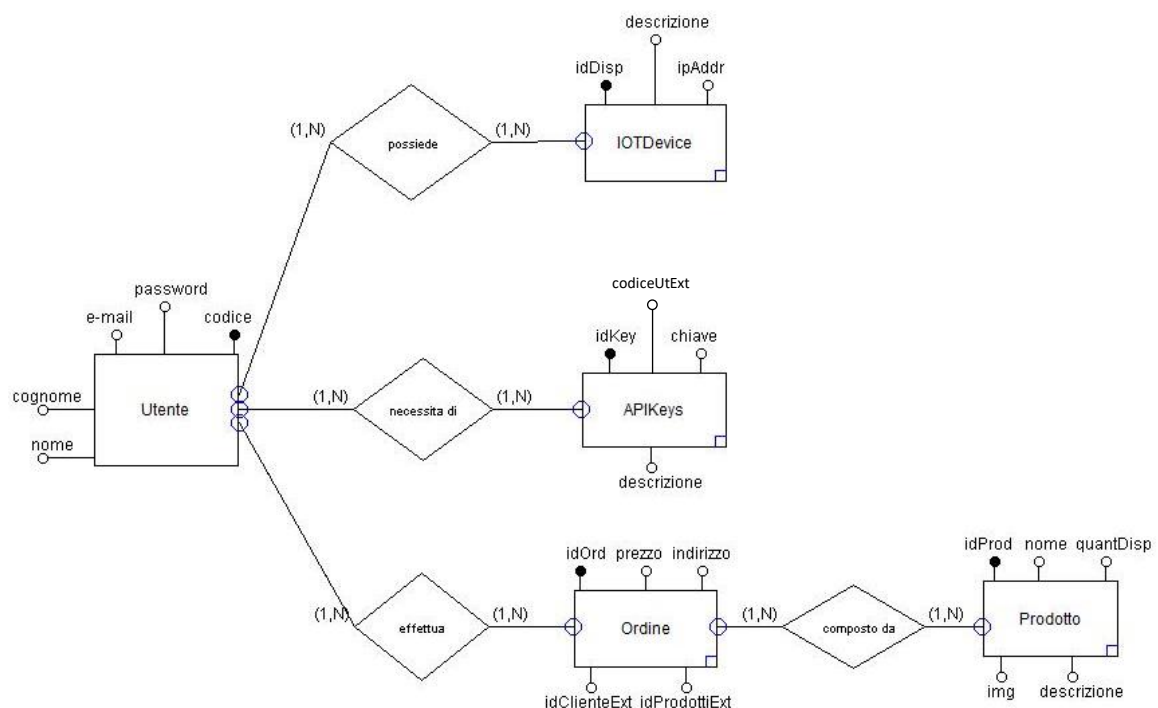
Utente (Codice (PK), e-mail, password, nome, cognome)

IOTDevice (IdDisp (PK), descrizione, ipAddr)

APIKeys (IdKey (PK), chiave, servizio)

Ordine (IdOrdine (PK), IdClienteExt (FK), idProdottoExt (FK), prezzo, indirizzo)

Prodotto (idProd (PK), nome, quantDisp, img, descrizione)



Esempio di creazione di una tabella:

```
CREATE TABLE UTENTE (  
  codice int(9) NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  email varchar(20) NOT NULL UNIQUE, password varchar(20) NOT NULL,  
  nome varchar(30) NOT NULL UNIQUE, cognome varchar(20) NOT NULL,  
);
```

Il sito web è **hostato** su una **VPS**, ovvero un'istanza di un sistema eseguito in ambiente virtuale, che permette l'esecuzione di più sistemi sullo stesso hardware. Ho scelto una VPS poiché il costo è comunque contenuto, ma non è necessario occuparsi della manutenzione, poiché è un servizio offerto dall'azienda venditrice del server, che in cambio di un pagamento mensile offre una gestione completa del server.

Sul server è installato un servizio che utilizza il protocollo **SMTP**, un protocollo standard per la trasmissione di email, che permette l'invio di una mail di conferma al cliente in seguito ad un acquisto o in risposta ad un ticket aperto per un problema tecnico.

Per quanto riguarda la modifica dei moduli visualizzati sullo specchio, i moduli sono sviluppati in javascript, e la schermata visualizzata sullo specchio è definita in un file chiamato javascript config.js, che contiene tutti i moduli disponibili, con un flag binario che ottiene il valore 1 nel caso in cui il modulo sia attivo, mentre ha il valore 0 nel caso in cui è disattivato. Quando l'utente selezionerà l'attivazione del determinato modulo tramite la pagina web, uno script PHP modifica il file di configurazione cambiando il flag da 0 a 1. Oltre al flag nel file di configurazione è anche possibile impostare la posizione sullo schermo dei vari moduli, tramite un comando javascript che modifica il file CSS di Node.js.

Infine nel caso in cui i moduli scelti dall'utente abbiano bisogno di una chiave API lo script PHP che modifica il flag del modulo aggiunge automaticamente la chiave al file config.js facendo una select dal database tramite il nome dell'utente.

```
SELECT chiave
FROM APIKeys as AK
INNER JOIN Utente as UT ON UT.codice=AK.codiceUtExt
AND AK.servizio=$valore;
```



```

<?php
if (isset($_POST['submit'])) {
    if (empty($_POST['username']) || empty($_POST['password'])) {
        $error = "Username or Password is invalid";
    }
    else{
        $username = $_POST['username'];
        $password = $_POST['password'];

        $conn = mysqli_connect("localhost", "root", "password", "database");

        $query = "SELECT username, password from Utente where username=\"".$username."\" AND password=\"".$password."\" ";
        $stmt = $conn->prepare($query);
        $stmt->execute();
        $stmt->bind_result($username, $password);
        $stmt->store_result();

        if($stmt->fetch()){
            $_SESSION['utente'] = $username;
        }else{
            $error = "Username or Password is invalid";
        }
    }

    mysqli_close($conn);
}
}
?>

```

Sicurezza

Il sito web utilizza il protocollo **HTTPS** per la protezione dei dati.

Il protocollo HTTPS è un protocollo per la comunicazione su Internet che protegge l'integrità e la riservatezza dei dati scambiati tra i computer e i siti, che viene usato per garantire agli utenti che l'utilizzo del sito web online avvenga in modo sicuro e privato.

I dati inviati tramite HTTPS vengono protetti tramite il protocollo Transport Layer Security (TLS), che fornisce tre livelli di protezione fondamentali:

- **Crittografia:** I dati scambiati vengono criptati per proteggerli dalle intercettazioni. Ciò significa che, mentre l'utente consulta un sito web, nessuno può "ascoltare" le sue conversazioni, tenere traccia delle attività svolte in più pagine o carpire le sue informazioni.
- **Integrità dei dati:** I dati non possono essere modificati o danneggiati durante il trasferimento, intenzionalmente o meno, senza essere rilevati.
- **Autenticazione:** Dimostra che gli utenti comunicano con il sito web previsto. Protegge da attacchi man-in-the-middle e infonde fiducia negli utenti, il che si traduce in altri vantaggi commerciali.

Il certificato viene emesso da un'autorità di certificazione (CA), che adotta misure per verificare che il tuo indirizzo web appartenga effettivamente alla tua organizzazione, proteggendo così i tuoi clienti da attacchi man-in-the-middle.

Per ottenere il certificato è necessario garantire un'elevata sicurezza utilizzando una chiave a 2048 bit.

Per quanto riguarda la **sicurezza** dei dati all'interno del **database** è necessario garantire 4 fattori:

- **Segretezza** dei dati, mantenendo la loro illeggibilità ai non autorizzati
- **Integrità** dei dati, protezione dei dati e delle informazioni nei confronti delle modifiche del contenuto, accidentali oppure effettuate volontariamente da una terza parte non autorizzata.
- **Confidenzialità**, protezione dei dati e delle informazioni scambiate tra un mittente e uno o più destinatari nei confronti di terze parti.
- **Disponibilità dei dati**, salvaguardia del patrimonio informativo nella garanzia di accesso, usabilità e confidenzialità dei dati.

Nonostante i DBMS moderni coprano già la sicurezza di questi fattori, proteggendo l'accesso al database con un'autenticazione tramite nome utente e password, ho deciso di incrementare la sicurezza criptando i file con un algoritmo **AES** e un hash **SH2** in modo da prevenire furti di dati tramite **SQL Injection**, e di diminuire il rischio di perdere dati tramite uno script **CronTab** per pianificare un backup automatico del database ogni giorno.

Crontab consente la pianificazione di comandi, ovvero consente la registrazione di questi presso il sistema per essere poi mandati in esecuzione periodicamente in maniera automatica dal sistema stesso.


Per quanto riguarda la **sicurezza dei pagamenti**, sono due le misure che un sito è obbligato ad adottare:

- Un gateway di pagamento
- Un certificato SSL

Un gateway di pagamento è un servizio di terze parti, ovvero del proprietario del metodo di pagamento, che processa il pagamento sul tuo sito web, è semplice da implementare e mantiene il pagamento sicuro. In questo modo il proprietario del sito non vede i dati del cliente, e viceversa, poiché il pagamento è gestito dall'esterno. Sul mio sito ho implementato il più famoso gateway di pagamento, ovvero quello di PayPal.

Il certificato SSL è essenziale per ogni sito web che riceve qualsiasi dato dall'utente, infatti è incluso nel momento dell'applicazione del protocollo HTTPS ad un sito web.

"AdvancedMirror"




[Home](#) [Negozio](#) [Moduli](#) [API Keys](#) [Supporto](#) [Carrello](#) [Login](#) [Registrati](#)

Il Dispositivo

Il dispositivo permette di gestire gli impegni giornalieri, mostrando il calendario personale, le previsioni meteo, lo stato del traffico sul percorso stradale per raggiungere il luogo di lavoro, gli orari dei mezzi di trasporto...

Con il dispositivo si avrà accesso ad una applicazione smartphone per gestire i moduli visualizzati sullo specchio, in modo da dare la possibilità all'utente di scegliere i moduli da visualizzare in base a quali ritiene utili, oltre che aggiungere altri moduli oltre a quelli di default, come ad esempio l'andamento di alcuni mercati finanziari, le notizie del giorno, i dati trasmessi da dispositivi di fitness.

Inoltre grazie ad un'integrazione con Amazon Alexa, l'assistente vocale di Amazon, sarà possibile gestire i dispositivi domotici comunicando con lo specchio, oppure attraverso script automatizzati, ad esempio accendendo luci, gestire le tapparelle, i climatizzatori, allarmi e telecamere.



Dispositivo

Struttura Logica

Il dispositivo è composto da un **Raspberry pi 3**, ovvero un single-board computer dal prezzo ridotto, su cui ho installato **Raspbian OS**, ovvero un sistema operativo basato su architettura Debian sviluppato appositamente per Raspberry, un sistema operativo molto leggero in grado di girare su un dispositivo con basse performance, dotato di una semplice interfaccia grafica.

Al Raspberry è collegato tramite un cavo HDMI uno schermo HD, al quale è applicato un sottile strato di vetro su cui è applicata una pellicola riflettente a due vie, in modo da poter avere la funzione di specchio e quella di schermo in simultanea.

I componenti sono contenuti in una struttura di legno, che ha anche funzione di cornice dello specchio.

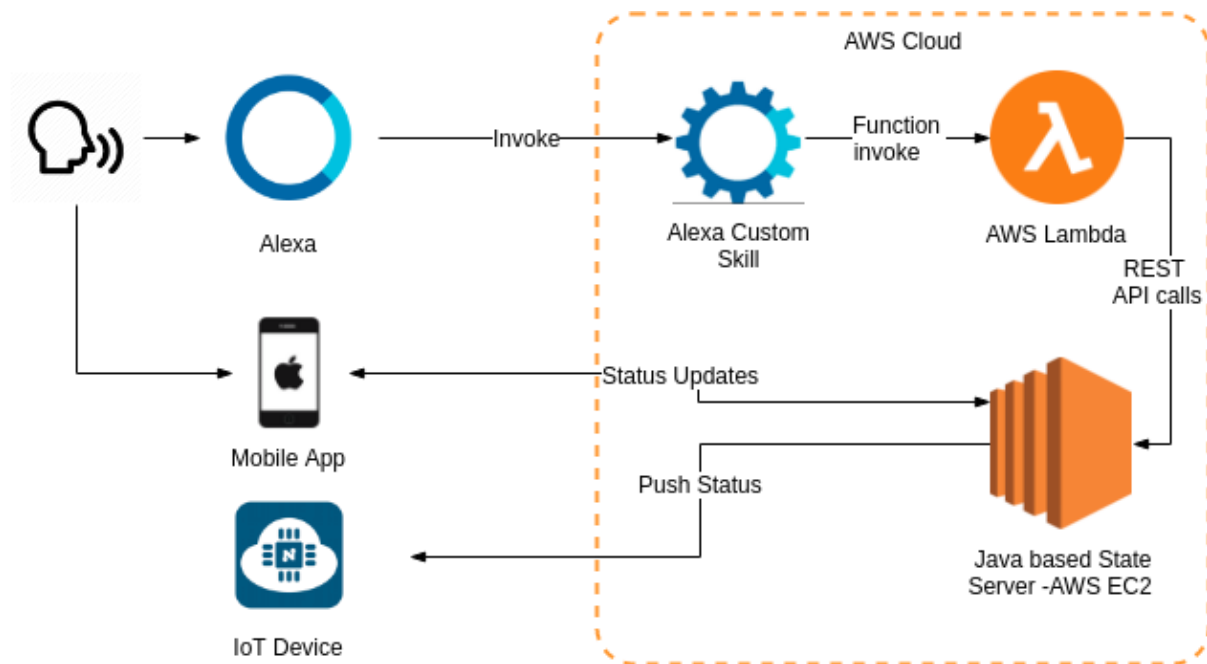
Inoltre al Raspberry è collegato tramite Jack audio un microfono e un amplificatore, necessari per la comunicazione con l'assistente vocale Amazon Alexa, e con i quali è anche possibile effettuare chiamate online.

Struttura Tecnica

Il software del dispositivo è basato su Node.js, un runtime di JavaScript, attraverso il quale verranno visualizzati i vari moduli, realizzati appunto in javascript.

I moduli riceveranno le informazioni da visualizzare a schermo tramite delle API di terzi, che permettono di accedere a informazioni di servizi esterni, come ad esempio i calendari, o le notizie di un giornale, o ancora l'andamento di un mercato in borsa.

Inoltre come già detto lo specchio è in grado di controllare dispositivi domotici compatibili con Amazon Alexa, tramite la piattaforma proprietaria AWS Lambda, che permette di sviluppare skill personalizzate.



Il meccanismo su cui si basa il funzionamento del dispositivo è quello dello specchio a due vie, è uno specchio che è riflettente su un lato e trasparente dall'altro. La percezione di trasmissione unidirezionale si ottiene quando un lato dello specchio è illuminato e l'altro lato è scuro. Questo permette di riflettere sulla zona con sfondo nero, mentre di vedere le immagini sulla zona illuminata.

Il dispositivo ha inoltre installato un server VPN, in modo tale da permettere di accedere tramite l'applicazione da remoto alla rete, cosicché sia possibile controllare i dispositivi domotici ovunque il cliente si trovi.

Un server VPN permette di creare una rete di telecomunicazioni privata, instaurata come connessione tra soggetti che utilizzano un protocollo di trasmissione pubblico e condiviso, come ad esempio la suite di protocolli Internet, in pratica permette di avere le stesse funzionalità delle linee private a noleggio, ma sfruttando reti condivise pubbliche: si può vedere dunque una VPN come l'estensione a livello geografico di una rete locale privata. Questo è permesso dal fatto che la connessione tra i device in una VPN è protetta, ovvero è cifrata dall'algoritmo RSA, che è basato su sull'esistenza di due chiavi distinte, che vengono usate per cifrare e decifrare, le quali nonostante siano fra loro dipendenti, non offrono modo di risalire dall'una all'altra, quindi nonostante si sia a conoscenza di una delle due chiavi, non è possibile risalire all'altra, garantendo così l'integrità della crittografia.

Infine ho fatto una stima dei costi del materiale fisico per la realizzazione di un singolo dispositivo, che non sono particolarmente elevati.

I componenti necessari per la realizzazione sono:

- Raspberry pi → 40€
- Microfono → 5€
- Altoparlante → 5€
- Schermo → 60€
- Specchio unidirezionale → 10€
- Legno per la struttura → 10€

Totale → 130€

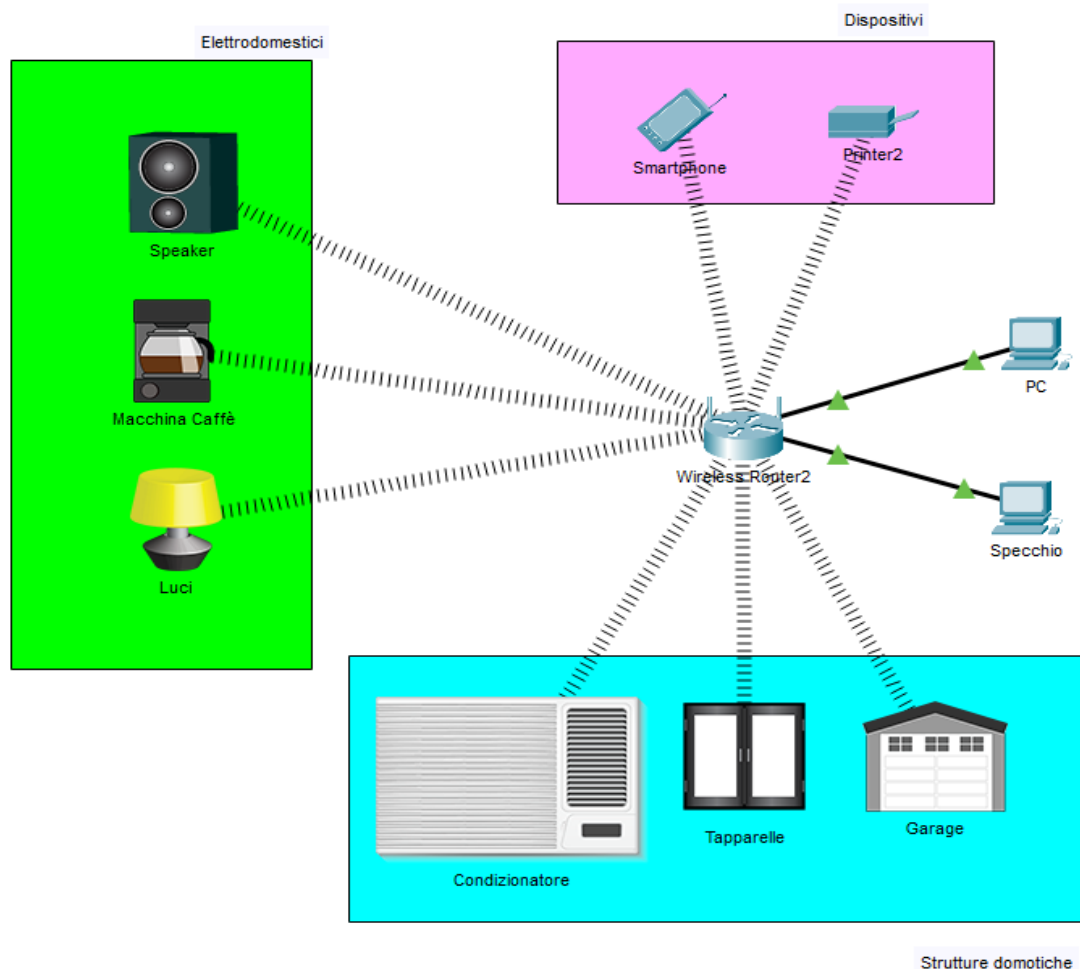
Il costo dei materiali necessari per un singolo dispositivo è di circa 100€, il quale è molto probabile che si abbassi grazie a una produzione in quantità più elevate.

The image shows a digital display interface with a black background. At the top left, it displays the date 'sabato, 13 giugno 2020' and the time '13:52' with a small '46' superscript. Below the time, there is a 'PROMEMORIA' section with two items: 'Consegna Materiale Orale' (today at 23:50) and 'Uscita Date Maturità' (Monday at 13:22). At the top right, it shows weather information: '3° 21:11', a sun icon, '25.0°', and 'Percepiti 25°'. Below this is a 'PREVISIONI METEO MILANO, IT' table. In the center, the text 'Buon Lavoro!' is displayed. At the bottom, there is a small line of text: 'Il Corriere della Sera, 26 minuti fa.' and a larger line: 'Fontana: «In Lombardia obbligatorio indossare le mascherine fino al 30 giugno»'.

PREVISIONI METEO MILANO, IT			
sab	☁	25.3°	17.9°
dom	☁	24.2°	15.1°
lun	☁	25.3°	16.8°
mar	☁	23.6°	16.4°
mer	☁	21.4°	16.5°
gio	☁	21.8°	15.5°

Architettura di rete

La struttura ideale della rete del cliente è questa:



Questa è appunto una rete teorica, poiché ogni cliente ha la propria rete casalinga formata dai dispositivi domotici che possiede.

Il pc fisso e lo specchio sono connessi tramite cavo ethernet, anche se lo specchio può essere connesso anche tramite Wi-Fi, per poter essere posizionato in ogni luogo della casa senza alcun problema.

Il resto dei dispositivi sono connessi tramite Wi-Fi. Poiché è una rete casalinga, gli indirizzi IP sono assegnati dal server DHCP, grazie al quale non è necessario gestire gli IP dei dispositivi.

Applicazione

L'applicazione smartphone è stata realizzata con AndroidStudio, un ambiente di sviluppo integrato (IDE) per la piattaforma Android mediante linguaggio JAVA e PHP.

Ho scelto di utilizzare questo IDE poiché è quello ufficiale di Android, ha molte funzionalità, tra cui un emulatore Android per provare l'applicazione in qualsiasi momento dello sviluppo. Inoltre nonostante il fatto che sia una piattaforma apposita per sviluppare applicazioni per Android è anche possibile convertirle in app iOS tramite un plugin.

Per quanto riguarda la struttura dell'applicazione è molto simile alla pagina web, è composta da 7 pagine:

- La pagina **home**
- La pagina per effettuare il **login** o la **registrazione**
- La pagina del **negozio**
- Il **carrello**
- La pagina per la **gestione** delle **API Key**
- La pagina per la gestione dei **moduli**
- La pagina del **supporto**

Inoltre anche la parte della comunicazione con il server attraverso PHP è molto simile, infatti le scelte dell'utente saranno salvate sul database tramite gli stessi script PHP del sito web, in modo da essere completamente uguali nel comportamento e non causare alcun tipo di problema.