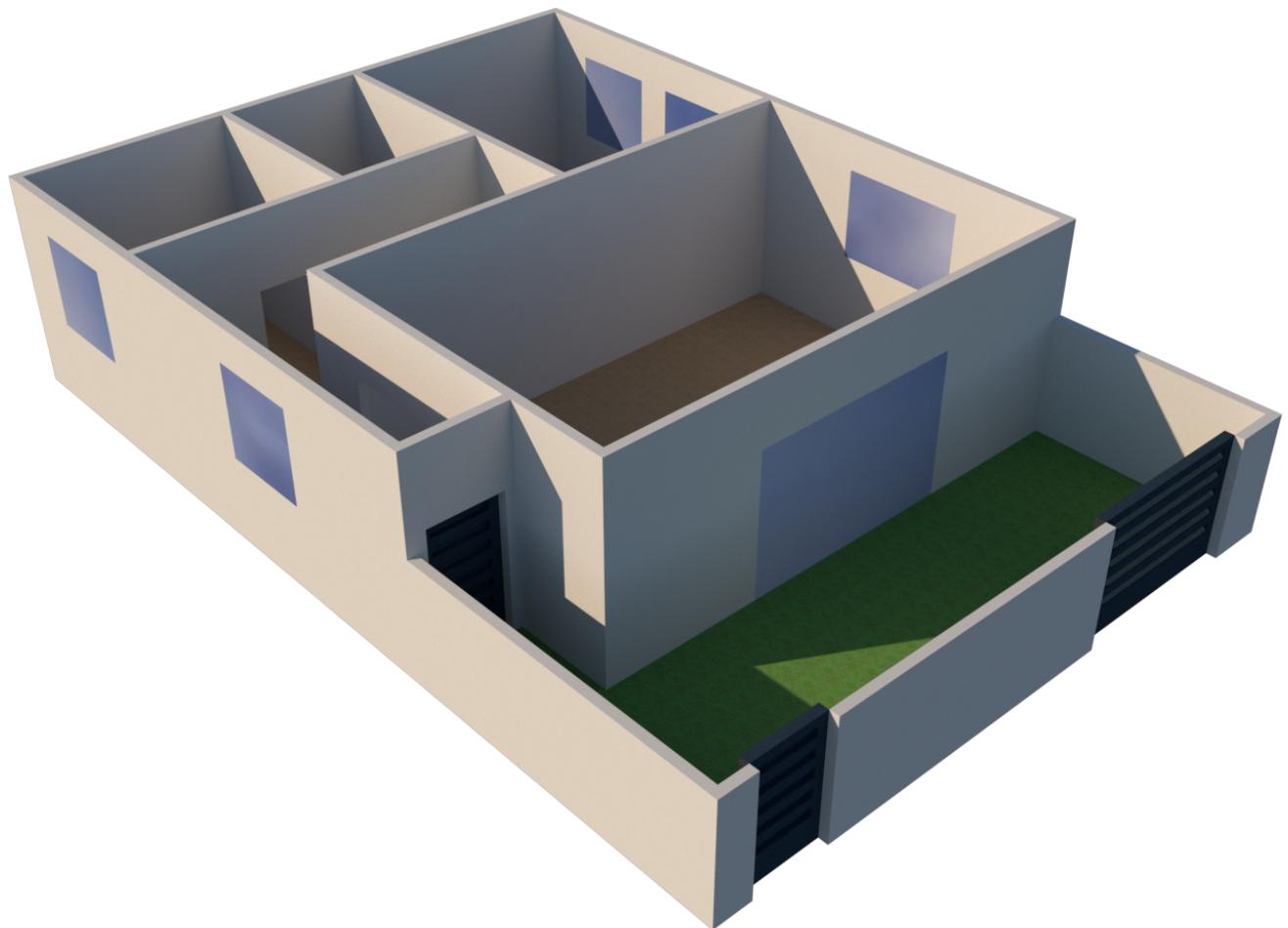


---

# CREACIÓ D'UNA CASA DOMÒTICA

---



Alumne: Anass Anhari Talib

Tutores: Engràcia Costa i Anna Ferrer

# ÍNDEX

<b>1.</b>	<b>INTRODUCCIÓ .....</b>	<b>1</b>
1.1.	Motivació.....	1
1.2.	Metodologia .....	2
<b>2.</b>	<b>OBJECTIUS.....</b>	<b>2</b>
<b>3.</b>	<b>QUÈ ÉS LA DOMÒTICA?.....</b>	<b>3</b>
3.1.	Tipus de sistemes domòtics.....	4
3.2.	Utilitats de la domòtica .....	6
<b>4.</b>	<b>ARDUINO .....</b>	<b>7</b>
4.1.	Microcontroladors Arduino .....	8
4.1.1.	Arduino Mega.....	9
4.2.	Arduino IDE .....	11
4.2.1.	Programació.....	11
<b>5.</b>	<b>TELEGRAM.....</b>	<b>17</b>
5.1.	Telegram Bot.....	17
5.1.1.	BotFather.....	18
5.2.	Python .....	19
5.2.1.	Comunicació amb el Bot i l'Arduino.....	21
<b>6.</b>	<b>iHomeBot.....</b>	<b>24</b>
6.1.	Control de casa.....	25
<b>7.</b>	<b>DISSENY I CONSTRUCCIÓ DE LA CASA.....</b>	<b>27</b>
7.1.	Model 3D.....	27
7.2.	Construcció .....	28
7.3.	Impressió 3D.....	32
7.3.1.	Ultimaker Cura.....	33
7.3.2.	Sistemes mecanitzats .....	35
7.4.	Components electrònics .....	37
7.4.1.	Esquemes.....	40
7.5.	Pressupost .....	41
<b>8.</b>	<b>CONCLUSIONS .....</b>	<b>42</b>
<b>9.</b>	<b>BIBLIOGRAFIA WEB .....</b>	<b>43</b>
	<b>ANNEX .....</b>	<b>45</b>

## 1. INTRODUCCIÓ

Avui en dia la tecnologia està integrada a la nostra cultura i societat, a més, ens és una eina molt útil que pràcticament l'apliquem cada dia en la nostra vida diària, ja que ens facilita i millora la qualitat de vida que portem.

En l'actualitat la podem veure aplicada en diversos àmbits com en la indústria, la medicina, l'educació, etc. Un exemple d'un gran avenç tecnològic és Internet, una xarxa exorbitant on a cada segon que passa s'hi puja una desmesurada quantitat de dades. Gràcies a Internet tots podem buscar informació d'una manera eficient en qualsevol moment i la podem comparar amb moltíssimes fonts que hi ha a la xarxa.

En aquest treball s'estudia l'aplicació de la tecnologia en el món de l'habitatge, és a dir, la implantació de la domòtica.

### 1.1. Motivació

Fa prop de dos anys el meu tiet Karim Talib, programador de robots industrials, em va regalar un Arduino, que és una placa programable.

Des del meu desconeixement d'Arduino, vaig començar a buscar informació sobre la placa i tutorials per aprendre a programar, i em va impactar el gran potencial que tenia la placa, ja que la placa permet crear infinitats de projectes, des de molts senzills com per exemple, un piano digital (QR 1), fins a més complexos com ara podria ser una impressora 3D o un dron.



QR 1: Piano digital

Al principi em semblava que la programació era costosa d'entendre, però el que feia era buscar codis per internet i entendre el que feien, i després modificar-los per realitzar accions desitjades. Finalment la programació és una sintaxi i requereix estudiar-la.

## 1.2. Metodologia

- En primer lloc, dissenyaré l'estructura de la casa utilitzant programes de disseny 3D per tenir una idea conceptual de com quedaria
- En segon lloc, hauré d'estructurar i crear el circuit elèctric de la casa, és a dir, pensar en quins components implementaré (sensors, motors, etc.). També crearé un Excel, ja que hi ha molts components i cadascun li toca un pin, per així saber a on va cada cable.
- En tercer lloc, procediré al muntatge de la casa començant per l'estructura i després per la part elèctrica.
- En tercer lloc, programaré tots els components de la casa des de l'Arduino IDE per finalment posar-la en funcionament.
- En darrer lloc, comprovaré si el codi realitza les instruccions programades i si hi ha algun error l'hauré de corregir.

## 2. OBJECTIUS

L'objectiu principal d'aquest treball serà construir una casa domòtica controlada per un Bot (robot) de Telegram (aplicació de missatgeria), i que pugui ser controlada remotament amb una comunicació molta fluïda. Això comporta que ha de estar tot ben programat, és a dir, la casa ha de ser programada i dissenyada amb molta cura des del principi i intentar que a mesura que incorporem més *hardware* (objectes controlables) a la casa, la comunicació sèrie amb el Bot sigui la més ràpida i fluïda possible.

Per fer això realitat, haurem de controlar els següents paràmetres:

- **Portes mecanitzades per servo motors:** Obrir o tancar portes remotament.
- **Portes de seguretat:** Obrir o tancar portes amb un sensor RFID (de radiofreqüència) i un teclat numèric.
- **Llums LED:** Encendre o apagar els llums de cada habitació.
- **Persianes motoritzades:** Pujar o baixar les persianes des de l'aplicació o automàticament depenent de la lluminositat detectada per un sensor de lluminositat.
- **Alarma d'intrusos:** Activació d'una alarma si es detecta moviment mitjançant un sensor de moviment.

### 3. QUÈ ÉS LA DOMÒTICA?

La paraula domòtica prové del llatí "domus" (que significa casa) i "tica" (d'automàtica).

És un conjunt de sistemes integrats que ens permeten automatitzar una casa, és a dir, que qualsevol mecanisme funcioni independentment, i que pugui ser prèviament programat per realitzar certes accions desitjades.

Ens podem fer una idea observant la imatge següent on la domòtica es pot aplicar en molts llindars:

- **Control independent de la casa:** Automatització de la casa en funció de les variables que es vulguin controlar (temperatura, lluminositat, ...).
- **Gestió de la casa:** Mitjançant una comunicació sèrie (casa - controlador).
- **Seguretat:** Dispositius de detecció de moviment.
- **Eficiència energètica:** Tenir una gestió controlada de la potència o energia consumida, per així, fer un estudi general (segons les variables que s'estudien, com per exemple, la temperatura, la lluminositat o l'hora) i veure si depenen d'unes certes condicions es pot obtenir un estalvi energètic.

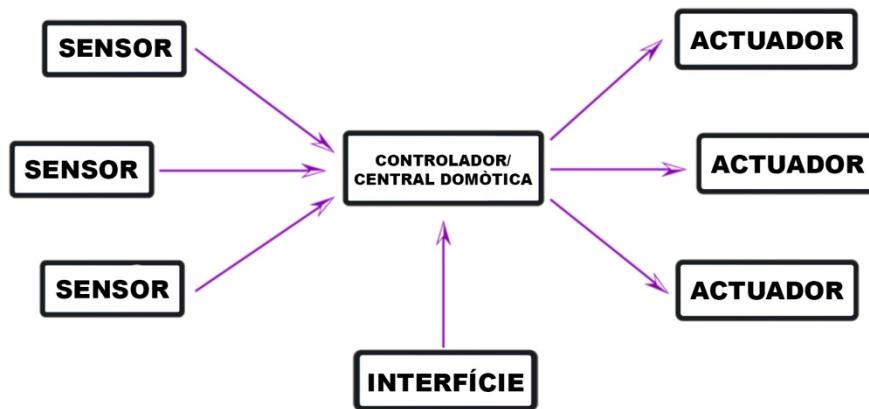


Fig. 1: Funcionalitat de la domòtica (Font: Escapada Rural)

### 3.1. Tipus de sistemes domòtics

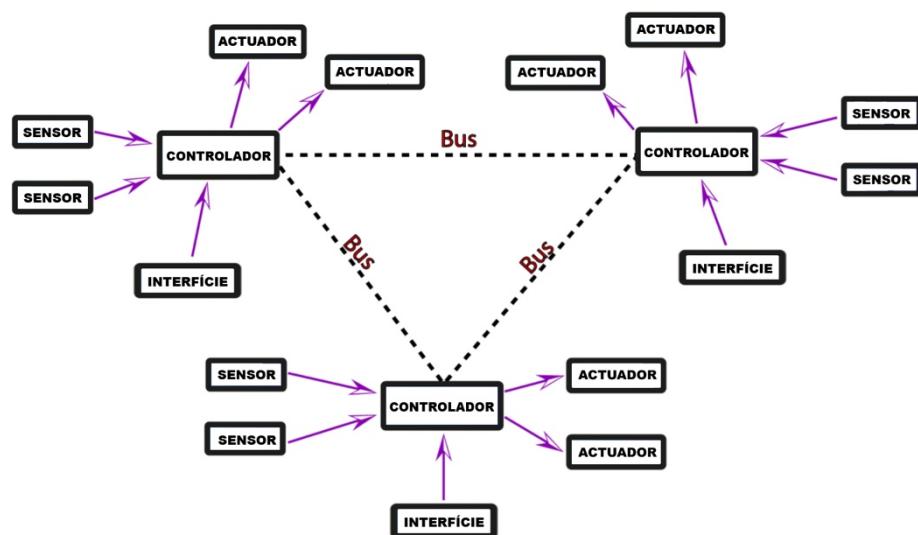
Principalment hi ha 3 tipus de sistemes:

- 1) **Sistema centralitzat:** Aquest és el sistema que s'ha seguit per construir la casa domòtica, podem veure que a l'*Esquema 1* tots els sensors envien informació a un controlador (o placa programable) que en el nostre cas és l'Arduino, i depenent de la informació que rep, dona unes instruccions als actuadors (motores, llums, etc.). I tota aquesta informació que s'envia i es rep viatja per una interfície que en aquest cas seria és un port de comunicació sèrie.



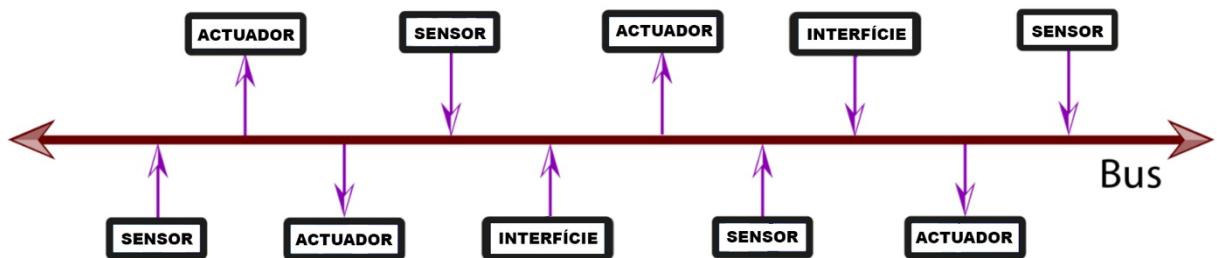
*Esquema. 1: Sistema centralitzat (Font: Pròpia)*

- 2) **Sistema descentralitzat:** Hi ha més d'un controlador, tots els controladors estan interconnectats per un bus, que s'encarrega d'enviar informació a tots els sensors i actuadors segons la configuració del programa.



*Esquema. 2: Sistema descentralitzat (Font: Pròpia)*

3) Sistema distribuït: Cada sensor i actuador funciona com un controlador, amb capacitat d'actuar i enviar informació al sistema segons el que rebi des d'un altre dispositiu. Això significa que incorpora una intel·ligència artificial i requereix més programació. Ens podem fer una idea general mirant el següent esquema.



Esquema. 3: Sistema distribuït (Font: Pròpia)

### 3.2. Utilitats de la domòtica

La domòtica és una eina de molta utilitat perquè ens optimitza qualsevol tipus de sistema integrat en una casa.

La domòtica té moltes funcions, les podem classificar de la següent manera:

- **Gestió energètica**

Ens ajuda a controlar tots els paràmetres que nosaltres vulguem reduir el consum energètic i així ajudant-nos a estalviar energia, com per exemple:

- Baixar les persianes d'una casa a l'hivern en funció de la temperatura obtinguda des d'un sensor exterior, així, mantenint l'escalfor interna de la casa.

- **Automatització**

Ens ajuda a portar un millor nivell de qualitat de vida facilitant-nos les tasques diàries, és a dir, manipular qualsevol mecanisme remotament, com per exemple:

- Controlar llums.
- Controlar portes.
- Controlar persianes.

- **Seguretat**

Podem tenir la casa absolutament controlada, i així oferint-nos una seguretat òptima, com per exemple:

- Tenir una alarma de seguretat amb sensor de moviment.
- Obrir portes amb RFID (identificació per radiofreqüència).
- Rebre notificacions quan la casa està desocupada.

- **Comunicació**

Ens permet establir una comunicació sèrie amb la casa des d'una aplicació mòbil per poder controlar qualsevol dels paràmetres anteriors.

## 4. ARDUINO

Arduino és una plataforma de codi obert que permet crear infinitat de projectes amb les eines que ens proporcionen.

Arduino ens dona l'opció de programar qualsevol cosa que se'ns plasmi al cap i portar-la a terme amb els microcontroladors que tenen i que poden ser programats amb els seu propi software (entorn de programació) anomenat Arduino IDE.



Logotip 1: Arduino (Font: Arduino.cc)

Arduino ha agafat molta importància arreu del món, i és el cervell de milers de projectes. Degut al gran triomf, avui en dia molts professionals utilitzen Arduino perquè és una eina de gran potencial, això permet a la persona que domina mitjanament la programació poder arribar a fer quasi qualsevol cosa.

Arduino també és l'opció més recomanable quan una persona s'introdueix en el món de la programació. Però per què Arduino es la millor opció?

- **És econòmic:** Els microcontroladors d'Arduino son més econòmics comparats amb altres microcontroladors que existeixen en el mercat, el més senzill pot arribar als 20€.
- **És multiplataforma:** El software d'Arduino (IDE) funciona en tots els sistemes operatius, Windows, Mac OS i Linux.
- **L'entorn de programació:** És un llenguatge senzill, clar, i a sobre intuïtiu, quan una persona s'aventura en el món de la programació.

## 4.1. Microcontroladors Arduino

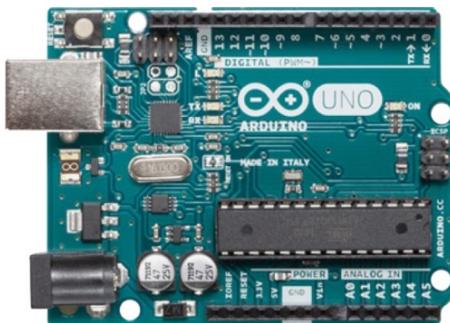


Fig. 2: Arduino UNO (Font: Arduino.cc)

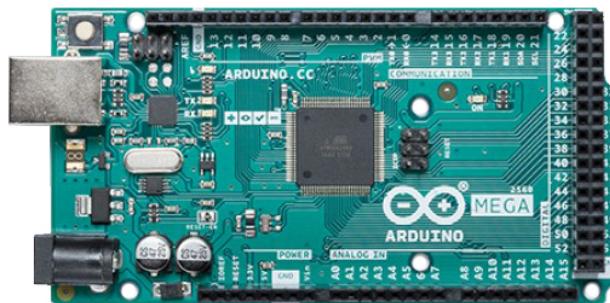


Fig. 3: Arduino MEGA (Font: Arduino.cc)

Arduino té molts tipus de microcontroladors al mercat, des de plaques amb capacitat de connectar-se a una xarxa Wi-Fi, fins a plaques molt petites per fer projectes senzills.

En aquest projecte domòtic no cal connectar el microcontrolador a una xarxa Wi-Fi ja que està controlada a través del terminal de l'ordinador, i aquest ja està connectat.

Com podem veure a les imatges de la part superior, les plaques més comunes i utilitzades són l'Arduino UNO (Fig. 2) i l'Arduino MEGA (Fig. 3).

Les dues plaques son similars però amb dues diferencies principals:

- L'Arduino Mega té més pins que l'Arduino UNO.
- Si el xip integrat de l'Arduino Mega es fon, no el podrem reemplaçar, en canvi, si es fon el xip integrat de l'Arduino UNO es podrà reemplaçar.

En aquest projecte he decidit treballar amb un Arduino Mega pel simple motiu que necessito molts més pins dels que té un Arduino UNO, així no farà falta tenir més d'un controlador.

#### 4.1.1. Arduino Mega

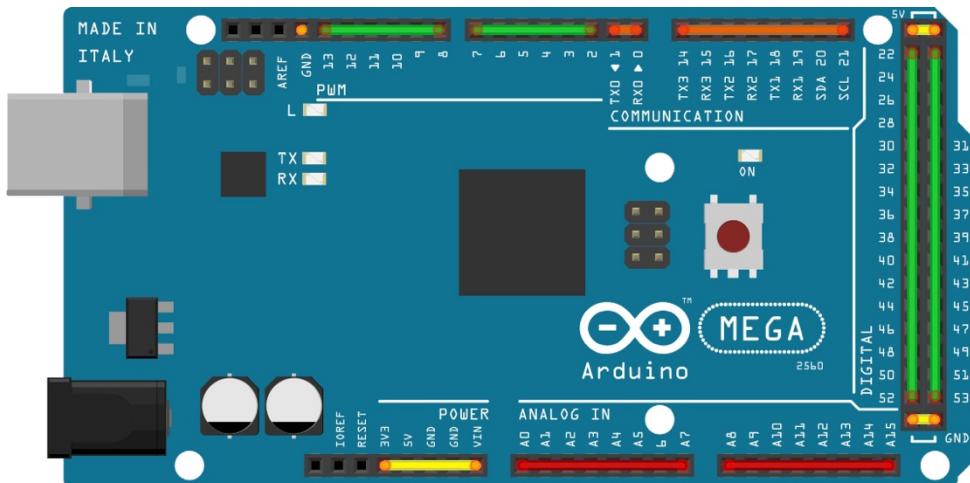


Fig. 4: Pins de l'Arduino MEGA (Font: Pròpia)

La placa té les següents característiques:

- Incorpora un xip integrat ATmega2560

Aquest xip (Fig. 5) és un circuit integrat, i s'encarrega d'executar les instruccions programades des de l'Arduino IDE.



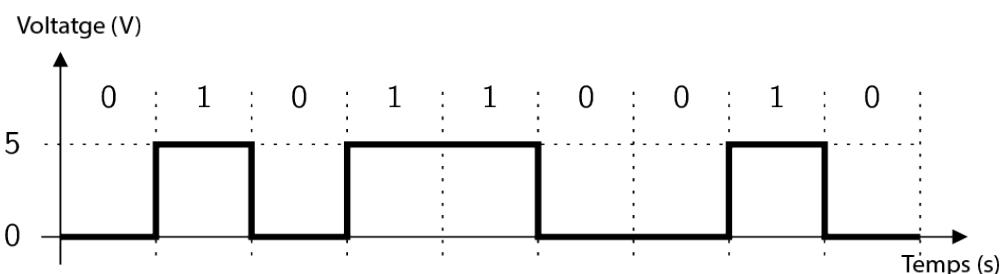
Fig. 5: Xip ATmega2560  
(Font: Rhydolabz)

- Té 52 pins digitals (color verd)

Els pins digitals ens permeten detectar si el valor que esta rebent l'entrada es "LOW" o "HIGH".

Les entrades digitals es poden interpretar com un valor binari 0 ("LOW") o 1 ("HIGH"), també es poden interpretar com un valor d'un voltatge, quan el valor és "HIGH" significa que la senyal és de 5V (volts) i quan el valor és "LOW" significa que la senyal és de 0V.

Cal remarcar que estem parlant de l'entrada específica de 5V d'Arduino per explicar els tipus d'entrada, ja que és la tensió de funcionament de l'Arduino.

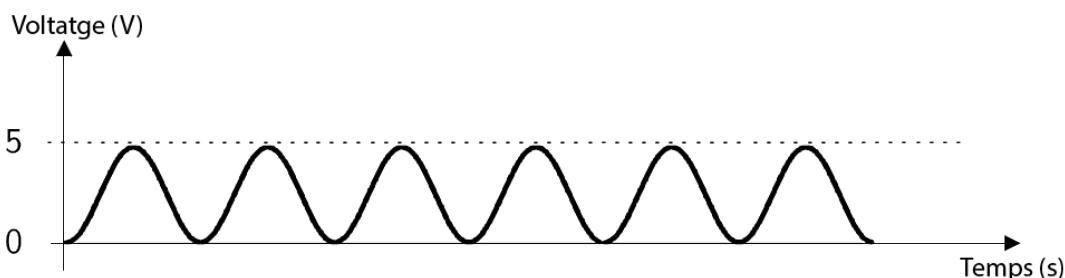


Gràfic 1: S'expressa el voltatge (0V o 5V) en funció del temps (Font: Pròpia)

- **Té 16 pins analògics (color vermell)**

Els pins analògics també ens permeten detectar el valor de l'entrada.

Com hem vist anteriorment, els pins digitals ens deixaven agafar només un valor de 1 o 0, però els pins analògics ens donen l'opció d'agafar qualsevol valor comprés entre 1 i 0, que seria el mateix que dir que proporcionen un voltatge entre 0V i 5V. És un bon exemple l'entrada analògica d'un led, que ens serviria per controlar la intensitat de la llum.



Gràfic 2: S'expressa el voltatge (de 0V fins a 5V) en funció del temps (Font: Pròpia)

- **Té 10 pins de comunicació (color taronja)**

Aquests pins ens permeten establir una comunicació sèrie en específic (consisteix en transmetre i rebre informació) entre l'Arduino i qualsevol altre objecte com ara podria ser un mòdul Bluetooth.

- **Té 10 pins de voltatge**

- **GND:** De l'anglès “Ground”, que significa el terra.
- **3,3V i 5V:** Ens proporciona el voltatge indicat.
- **Vin:** Ens proporciona el voltatge que li connectem per l'entrada que té la placa.

(**Important:** El màxim voltatge que se li pot subministrar a l'Arduino és de 20V, sinó podem cremar el microcontrolador.)

## 4.2. Arduino IDE

Arduino IDE (*Integrated Development Environment*), és el software amb el que nosaltres podem programar i donar instruccions a l'Arduino i que les pugui executar o posar en funcionament.

Des de la pàgina oficial d'Arduino ([www.arduino.cc/en/Main/Software](http://www.arduino.cc/en/Main/Software)) ens especificuen que l'entorn de programació està escrit en Java (llenguatge informàtic) i basat en Processing (llenguatge de programació).

En resum, perquè no hi hagin confusions, diem que el llenguatge d'Arduino és C++, ja que és una programació molt orientada a objectes controlables.

### 4.2.1. Programació

Ara procedirem a explicar breument com funciona l'entorn de programació:

En la imatge (Fig. 6) podem veure com és l'aspecte de l'entorn de desenvolupament.

Principalment veiem que el programa està estructurat en dues fases que procedirem a analitzar ara.

```

1
2 void setup() {
3 // put your setup code here, to run once:
4
5 }
6
7 void loop() {
8 // put your main code here, to run repeatedly:
9
10 }

```

Guardado.

Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) en /dev/cu.usbmodem1421

Fig. 6: Apecte de l'Arduino IDE (Font: Pròpia)

Per poder programar en qualsevol llenguatge hem de tenir clars 3 conceptes bàsics.

Primerament, el llenguatge de programació, s'estudia com qualsevol altre idioma, ja que tot es basa en la sintaxi i l'estructura del llenguatge.

Si ara generalitzem la sintaxi lingüística, sabem que s'estudia l'estructura del llenguatge, les categories gramaticals i les funcions sintàctiques.

Doncs, en la sintaxi informàtica s'estudia:

- L'estructura del "script" (anglicisme que significa codi del programa, és a dir, un conjunt d'instruccions informàtiques)
- Les funcions (depenent de l'objecte que s'estigui programant)
- Les variables, que s'assimilarien a les categories gramaticals

## 1) Estructura

Avançarem explicant l'estructura més habitual i utilitzada en C++, perquè és l'estructura que s'ha seguit en la programació de la casa domòtica.

Dins de l'estructura d'un script hi ha diferents tipus d'estructures:

### a) Estructura bàsica i essencial del script

Hi ha dues funcions, la `void setup()` i la `void loop()` com s'ha vist anteriorment en l'apartat de l'Arduino IDE.

En la funció `void setup()` es declaren tots els objectes que es volen programar, és a dir, estem indicant quins objectes (motors, sensors, etc.) volem programar i monitoritzar.

En la funció `void loop()` es troba la major part de les instruccions. L'Arduino ens les repetirà indefinidament, això significa que la funció `loop()` ens està fent un bucle, i la placa anirà llegint totes les instruccions des de dalt fins a baix i quan completi aquest cicle el tornarà a repetir.

La base del funcionament del meu programa segueix l'estructura definida anteriorment, i el codi del l'Arduino es troba als annexos.

### b) Estructura de control

Bàsicament l'objectiu és interrompre el `loop()` per fer alguna comprovació d'una variable.

Els controls que han estat utilitzats en aquesta casa han sigut els següents:

- `while()`

El que fa un `while()` és executar un altre `loop()`.

En el `while()` l'Arduino està comprovant les dades rebudes des de Python (que s'explicarà després), on s'executarà un bucle indefinidament per estar constantment rebent les dades per realitzar alguna acció programada.

- `if ... else`

Són els dos controls més utilitzats en l'Arduino i sobretot en el Python, on l'`if` (en català seria el condicional "si") ens comprova una condició específica per saber si ha d'executar alguna instrucció. En aquest cas si la condició que es vol executar és `true`, és a dir, que la condició es compleix, es realitzarà alguna acció.

L'`else` ens fa el contrari del que fa l'`if`, és a dir, si no es compleix la condició, es farà alguna altra cosa programada, dependent del que s'estigui programant.

- `switch ... case`

En aquest cas específic, el `switch` actua de manera similar que la condició `if`, però ara està segmentat en diferents casos (control `case`), és a dir, que hi ha una sèrie de condicions per cada cas en específic.

- `break`

El que fa és interrompre una condició loop, també ens podem fer una idea del que significa la paraula de l'anglès `break` que significa interrompre. I també ens serveix per interrompre un `switch`.

### c) Operadors de comparació

Són simples comparadors, que com bé ens indica, ens són molt útils per poder comparar valors de variables. En la programació de la casa hi ha molts comparadors (`!=`) per poder crear una memòria utilitzada en tots els motors, per guardar valors de `0` o `1`, per poder així, comprovar quin valor s'ha guardat; i aquest valor serà l'últim l'estat del motor, és dir, si el motor ha girat en sentit positiu (horari) o negatiu (antihorari).

Tots els tipus de comparadors estan explicats a la **Taula 1**.

SINTAXI INFORMÀTIC	SIGNIFICAT
<b>!=</b>	No és igual a ...
<b>&lt;</b>	Menor que ...
<b>&gt;</b>	Major que ...
<b>&lt;=</b>	Menor o igual a ...
<b>&gt;=</b>	Major o igual a ...
<b>==</b>	És igual a ...

Taula 1: Tipus de comparadors

## 2) Variables

Una variable (en programació), és un espai registrat en una memòria per emmagatzemar un valor que correspon a un tipus de dada. Els tipus de dades més típics són els de la taula següent:

TIPUS DE DADA	EMMAGATZEMATGE
<b>Int</b>	Emmagatzema un valor o número enter
<b>String</b>	Cadena de caràcters, pot emmagatzemar més d'una lletra
<b>Float</b>	Valor o número decimal
<b>Char</b>	Només emmagatzema un únic caràcter o número
<b>Boolean</b>	Pot emmagatzemar o un 0 o un 1, o també un valor "True" o "False"

Taula 2: Variables més comunes

També podem dir que hi ha variables una mica "més especials", que més que variables, serien un tipus d'entrada, però també poden ser classificats com una variable, que són les següents:

- **OUTPUT:** És qui rep l'acció
- **INPUT:** És l'encarregat d'accionar el programa i donar instruccions.

- Exemple demostratiu (Codi 1):

Un led que s'encén al pressionar un pulsador .

El led seria l'**OUTPUT** i el pulsador seria l'**INPUT**, així, quan el pulsador és premut li dona l'ordre al led, aquest, rebrà l'ordre i s'encendrà.

```
int led = 3; // Led connectat al pin 3
int pulsador = 4; // Pulsador connectat al pin 4

void setup() {
    pinMode (led, OUTPUT); // Es declara com OUTPUT
    pinMode (pulsador, INPUT); // Es declara com INPUT
}

void loop() {
    int estat = digitalRead (pulsador); // Ens guardarà el valor del pulsador

    if (estat == HIGH){ // Si el pulsador és premut
        digitalWrite (led, HIGH); // Encén el led
    }
    else{ // Si el pulsador es deixa de premer
        digitalWrite (led, LOW); // Apaga el led
    }
}
```

Codi 1: Encendre un led en funció d'un pulsador (Font: Pròpia)

### 3) Funcions

Una funció és un conjunt d'un programa que s'encarrega de dur una tasca específica, que prèviament hem programat a l'Arduino IDE, donant-li una sèrie d'ordres o instruccions.

Hi ha moltes funcions, ja que hi ha molts objectes programables que tenen una funció específica.

- Exemple demostratiu:

```

1 int led = 3; // Definim el led = 3
2
3 void setup() { // Declarem objectes
4
5     pinMode (led, OUTPUT); // Led al pin 3
6
7 }
8
9
10 void loop() { // Codi que es repetirà
11
12     digitalWrite (led, HIGH); // Encendre
13     delay (1000);
14
15     digitalWrite (led, LOW); // Apagar
16     delay (1000);
17
18 }
19

```

Codi 2: **Encendre un led** (Font: Pròpia)

- 1) Primer definim el led assignant-lo per un valor de 3, que significa que el connectem al pin 3 de l'Arduino.
- 2) En la funció **void setup()** declarem tots els objectes que vulguem utilitzar, és a dir, li estem indicant quin objectes (llums led, actuadors, ...) voldrem programar i monitoritzar.
- 3) El led es declara com a **OUTPUT** com s'ha explicat anteriorment.
- 4) La funció **digitalWrite ()** ens encén el led donant-li un valor **HIGH** com s'ha explicat anteriorment (apartat [4.1.1. Arduino Mega](#)).
- 5) Ara bé, cada **punt i coma** ; ens indica a on acaba la línia de la instrucció.
- 6) Els **claudàtors {}** ens indiquen a on comencen i acaben les línies d'instruccions .

## 5. TELEGRAM

Telegram és una aplicació de missatgeria que ens permet enviar missatges i continguts (fotos, vídeos, documents,...).



Logotip 2: **Telegram**  
(Font: Pròpia)

Com qualsevol altra aplicació de missatgeria, per poder utilitzar-la s'ha de registrar el número de telèfon per poder tenir un compte.

L'aplicació està disponible per mòbil i ordinador, això significa podrem controlar la casa des dels dos punts, sempre i quan tinguem connexió a dades mòbils, *Ethernet* o *Wi-Fi*.

### 5.1. Telegram Bot

Anteriorment hem dit que per poder utilitzar l'aplicació hem d'estar registrat als servidors de Telegram. Ara bé, l'aplicació ens proporciona una eina molt útil que ens permet crear Bots (robots), però que no estan registrats ni associats a cap numero de telèfon. Aquests, ens ajuden a automatitzar moltes coses, com ara un Bot que al rebre un correu el reenvii a una sèrie de contactes predefinits.

Els Bot ens faciliten o ens simplifiquen les tasques quotidianes en la nostra vida, cal remarcar que hi ha diferents tipus de Bot, per exemple, Bots per buscar informació a Google (imatges, vídeos a YouTube, ...) des del xat de Telegram fins a un Bot que t'envii les previsions meteorològiques.

Per poder crear un Bot ho fem mitjançant un altre Bot que s'anomena BotFather i que Telegram l'ha fet públic i es pot trobar a l'aplicació buscant-lo pel seu nom.

### 5.1.1. BotFather

Obrim conversa des de Telegram amb BotFather per poder crear el nostre propi Bot.

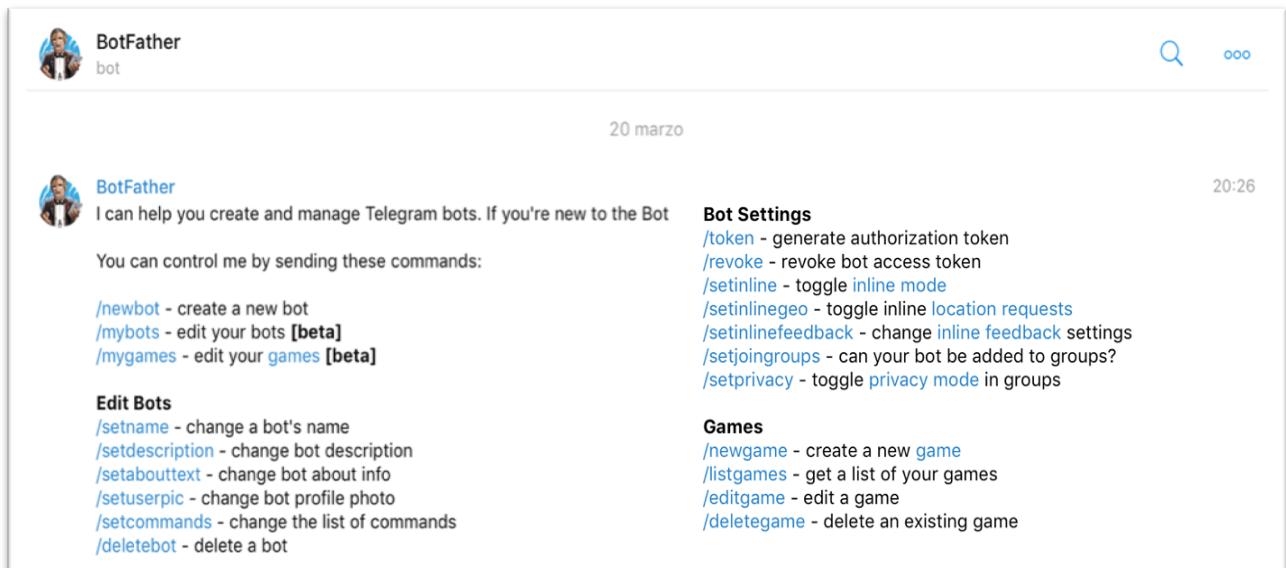


Fig. 7: Iniciem conversa (Font: Pròpia)

Amb l'ordre `/newbot` creem el nostre Bot.

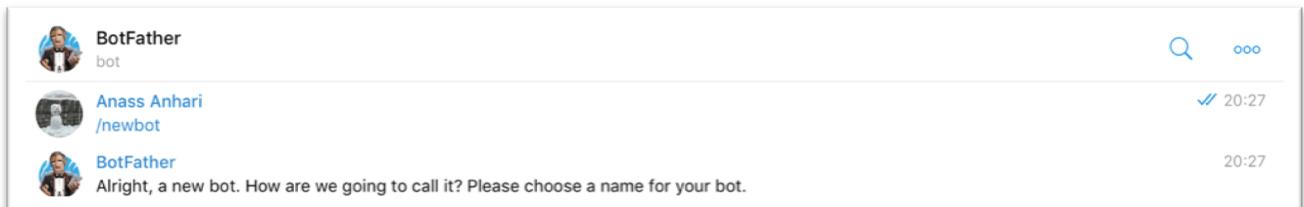


Fig. 8: Creació del Bot (Font: Pròpia)

Una vegada creat el Bot ens demanarà que li posem un nom al nostre Bot i ens donarà un Token de seguretat, que bàsicament és una cadena de caràcters que conté un identificador.

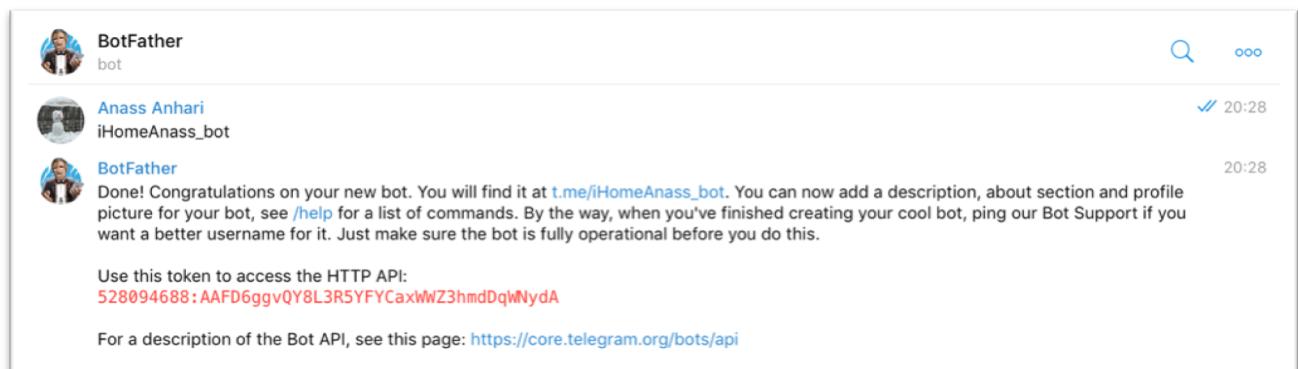


Fig. 9: Token o identificador de seguretat (Font: Pròpia)

## 5.2. Python

Python és un altre llenguatge informàtic, i convé explicar les principals diferències amb el C++.

Per començar, Python és un llenguatge que es programa en alt nivell, però, què significa això?

Existeixen dos nivells de programació:



Logotip 3: Python  
(Font: Python)

### 1) Nivell alt

Molta gent tendeix a pensar que programar en alt nivell és més difícil que programar en baix nivell però és totalment al contrari.

Quan programem en alt nivell generem un codi més senzill ja que estem treballant amb moltíssimes dades i funcions a la vegada i per poder crear un programa més senzill (però igual de funcional) necessitem l'ajuda de paquets, llibreries o biblioteques que incorporen funcions internes. També podem dir que es diu d'alt nivell perquè el nivell del llenguatge és més abstracte que el de baix nivell.

En aquest cas, per programar el Bot he hagut d'instal·lar les següents llibreries:

```
❶ iHomeBot.py ✘
1 import sys
2 import time
3 import telepot
4 import serial
5 from telepot.namedtuple import ReplyKeyboardMarkup, KeyboardButton
6 from telepot.namedtuple import ReplyKeyboardMarkup, KeyboardButton, ReplyKeyboardRemove
7 from telepot.namedtuple import InlineKeyboardMarkup, InlineKeyboardButton
```

Fig. 10: Llibreries Python (Font: Pròpia)

La més important és la **telepot**, una **framework** que ens facilita l'entorn de desenvolupament i que alhora també incorpora més biblioteques internes i ens permet interactuar amb l'API (Interfície de Programació d'Aplicació) que Telegram ens proporciona i que en aquest cas ens ajuda a fer funcionar el Bot amb Telegram.

### 2) Nivell baix

Aquest nivell està més orientat a la programació d'objectes, normalment és més difícil, però quan diem que és difícil no ens referim a la complexitat del codi sinó a la dificultat d'estructurar el codi, ja que incorpora moltes especificacions tècniques pròpies del hardware, és a dir, cada objecte té una funció en específic.

### Exemples demostratiu:

Compararem 2 scripts on un està programat en alt nivell i l'altre en baix nivell.

```

1 #include <Stepper.h> // Llibreria per programar el Motor
2
3 const int PASSOS = 1500; // Número de passos
4
5 Stepper Motor (PASSOS, 8, 10, 9, 11) // Pins de les entrades del Motor (IN1, IN3, IN2, IN4)
6
7 void setup(){
8     Motor.setSpeed (20); // Velocitat = 20 rpm (voltes/min)
9 }
10
11 void loop(){
12     Motor.step (PASSOS); // Gira PASSOS en Sentit Positiu (Dreta)
13     delay (1000); // Epera 1s
14     Motor.step (-PASSOS); // Gira PASSOS en Sentit Negatiu (Esquerra)
15     delay (500); // Epera 0,5s
16 }
```

Codi 3: Script programat en alt nivell (Font: Pròpia)

```

1 void setup() {
2     pinMode(11, OUTPUT); // Pin 11 = IN4
3     pinMode(10, OUTPUT); // Pin 10 = IN3
4     pinMode(9, OUTPUT); // Pin 9 = IN2
5     pinMode(8, OUTPUT); // Pin 8 = IN1
6 }
7
8 void loop() {
9     //Gir en Sentit Negatiu (Esquerra)
10    digitalWrite(11, HIGH); digitalWrite(10, HIGH); digitalWrite(9, LOW); digitalWrite(8, LOW); // Fase 1
11    delay(1000); // Espera 1s
12    digitalWrite(11, LOW); digitalWrite(10, HIGH); digitalWrite(9, HIGH); digitalWrite(8, LOW); // Fase 2
13    delay(1000); // Espera 1s
14    digitalWrite(11, LOW); digitalWrite(10, LOW); digitalWrite(9, HIGH); digitalWrite(8, HIGH); // Fase 3
15    delay(1000); // Espera 1s
16    digitalWrite(11, HIGH); digitalWrite(10, LOW); digitalWrite(9, LOW); digitalWrite(8, HIGH); // Fase 4
17    delay(1000); // Espera 1s
18 }
```

Codi 4: Script programat en baix nivell (Font: Pròpia)

Com podem veure el Codi 3 està programat en alt nivell ja que hem incorporat una llibreria que en ajuda a programar el motor. Per programar el gir del motor només s'ha d'escriure la funció `.setSpeed` i `.step()` per indicar la velocitat i el sentit del gir del motor.

En canvi, el Codi 4 està programat en baix nivell perquè s'ha de saber que un motor pas a pas consta de 4 fases i es pot apreciar veient que té 4 pins i cada pin influeix en el gir, i per programar el motor hauríem de saber què fa cada fase.

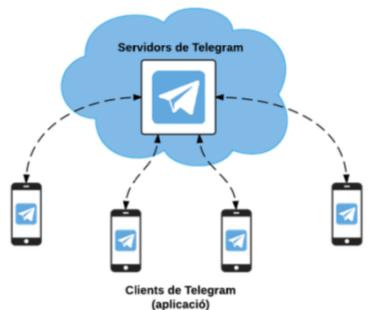
### 5.2.1. Comunicació amb el Bot i l'Arduino

Per començar, sabem que disposem de 3 punts de gestió:

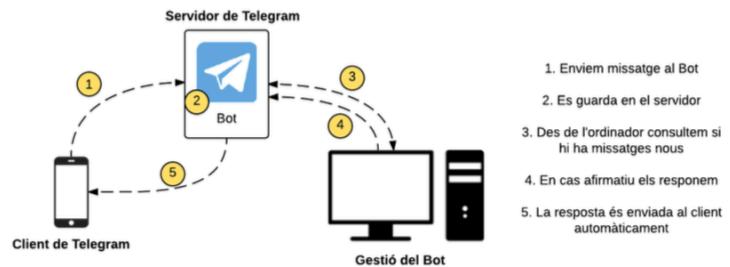
- 1) Telegram
- 2) Python
- 3) Arduino

Com podem veure a l'*Esquema 4* Telegram es basa en dos punts; tenim un client que és l'aplicació amb la que interactuem ja sigui des d'un ordinador o mòbil i per l'altra banda tenim un servidor que gestiona la comunicació entre els clients.

Tota la informació del Bot està emmagatzemada als servidors de Telegram però per poder fer funcionar el nostre Bot i que pugui interactuar amb nosaltres l'haurem de programar amb Python ajudant-nos d'una llibreria anomenada **Telepot**. Aquesta, està instal·lada en el terminal del nostre ordinador i s'encarregarà d'establir una connexió des del terminal amb el servidor on es troba el Bot.



*Esquema 4: Client – Servidor*  
(Font: Internet de les coses, Albert Babí)



*Esquema 5: Comunicació Terminal – Bot*  
(Font: Internet de les coses, Albert Babí)

A l'*Esquema 5* veiem que tenim un punt central de comunicació que és el servidor de Telegram, i funciona gràcies al nostre terminal que està connectat al servidor i hem programat en Python des d'un entorn integrat de desenvolupament (IDE) anomenat **Visual Studio Code** proporcionant-li a la llibreria (Telepot) el Token que ens ha establert BotFather anteriorment.

En resum, el que estem fent és programar en l'IDE en Python amb una sèrie de llibreries, on la més important és la Telepot, i amb el Token de seguretat podrem executar el codi des del terminal al servidor, així, el Bot podrà funcionar.

Per fer-nos una idea del que vol dir tot això cal veure com és l'IDE, el terminal i el Bot.

Visual Studio Code (Fig. 11), és un dels molts entorns de programació que existeixen i a continuació podem veure l'entorn on s'ha programat del codi del Bot.

```

import sys
import time
import telepot
import serial
ser = serial.Serial('/dev/cu.usbmodem1421, 9600') # Conexió amb el port d'arduino
bot = telepot.Bot("5280*****:AAFD6gg**Y8L3****axWZ3hmdD****") # TOKEN del bot
print('Esperant ordres ...')

def handle(msg):
    userName = msg['from']['first_name']
    content_type, chat_type, chat_id = telepot.glance(msg)
    print('Chat Message: ', content_type, chat_type, chat_id)

    if (content_type == 'text'):
        command = msg ['text']
        print ('Comando Obtenido: %s' % command)

    # Iniciació
    if '/start' in command:
        bot.sendMessage (chat_id, "Hola, " + userName + "\n" + "Soc el teu bot, per poder controlar la casa has d'entrar en el menú\n+ "a) Entrada" + "\n"
+ "b) Passadís" + "\n"
+ "c) Menjador" + "\n"
+ "d) Habitació 1" + "\n"
+ "e) Habitació 2" + "\n"
+ "f) Sensors")

```

Fig. 11: Visual Studio Code (Font: Pròpia)

En la Fig. 12 veiem com és el terminal del sistema operatiu MacOS i també una sèrie de comandaments per activar el Bot. En l'última instrucció (`python iHomeBot.py`) el que fem és executar el codi per connectar-nos al servidor on es troba el Bot, i el paper de la llibreria (Telepot) és buscar el Bot amb el Token.

```

MacBook-Pro-de-Anass:~ anassanhari$ cd desktop
MacBook-Pro-de-Anass:desktop anassanhari$ cd iHomeBot
MacBook-Pro-de-Anass:iHomeBot anassanhari$ ls
Icon?           iHomeBot.py      iHomeBot_env      mòduls.txt
MacBook-Pro-de-Anass:iHomeBot anassanhari$ source iHomeBot_env/bin/activate
(iHomeBot_env) MacBook-Pro-de-Anass:iHomeBot anassanhari$ python iHomeBot.py
Esperant ordres ...

```

Fig. 12: Terminal del sistema operatiu MacOS (Font: Pròpia)

Queda connectar l'Arduino amb el terminal per poder controlar remotament la casa domòtica des del Bot.

Hem de connectar l'Arduino a l'ordinador mitjançant un port USB, i agafarem la direcció sèrie del port per poder-lo comunicar amb el terminal. Per comunicar l'Arduino amb el terminal s'ha d'instal·lar una llibreria anomenada *PySerial* i per conèixer la direcció sèrie s'ha de fer a través de l'Arduino IDE (*Fig. 13*) on es veu a on es troba connectat l'Arduino.

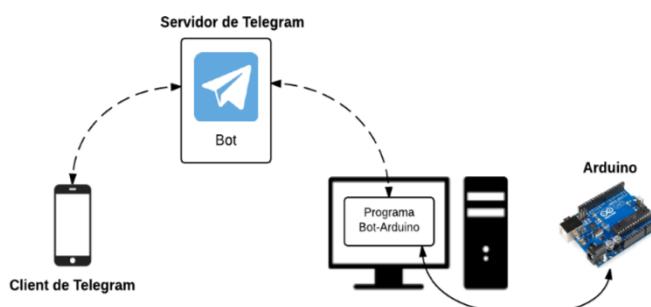


*Fig. 13: Direcció del port sèrie de l'Arduino*

### L'esquema següent resumeix totes les comunicacions:

Finalment, tenim que el client de Telegram està connectat al servidor on es troba el Bot que funciona gràcies al terminal del nostre ordinador proporcionant-li la intel·ligència programada per a què pugui interactuar.

Al mateix temps, el serial de l'Arduino està també connectat al terminal per a què pugui rebre informació enviada des del Bot al terminal i que pugui llegir les ordres per actuar.

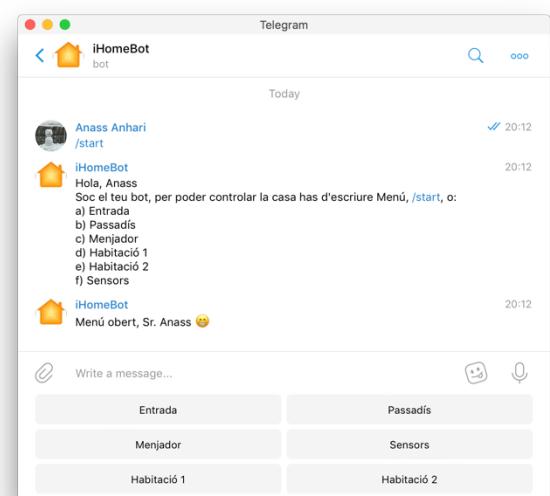


*Esquema 6: Comunicació Bot - Terminal – Arduino  
(Font: Internet de les coses, Albert Babí)*

## 6. iHomeBot

iHomeBot és el nom del Bot que s'ha creat per aquest treball. Es va decidir que tingués una estructura d'una aplicació mòbil; estructurat en pestanyes i botons. Es veu de la següent manera (*Fig. 14*), on cada botó controla una zona de la casa.

Per aconseguir crear una botonera com es mostra en la imatge cal importar una sèrie de llibreries que es troben internament en la llibreria Telepot que hem explicat anteriorment i la més important s'anomena **ReplyKeyboardMarkup**. El que fa aquesta llibreria és donar-nos l'opció de crear un teclat de manera matricial, és a dir, creem una matriu de manera personalitzada on podem definir el número de files i columnes on cada botó està inscrit en una determinada posició de la matriu.



*Fig. 14: iHomeBot (Font: Pròpia)*

En el *Codi 5* observem com s'ha programat el teclat matricial de la *Fig. 14*. La funció **KeyboardButton (text = "")** crea un botó i podem definir els botons que vulguem entre els claudàtors i separant-los per comes, així, creant una columna.

I per crear una fila s'ha de separar tot el contingut del claudàtor amb una coma a l'exterior.

```

33     # Buttons Menú
34         if 'Menú' or 'menú' or 'Esc' or 'esc' in command:
35             bot.sendMessage(chat_id, 'Menú obert, Sr. ' + userName + " 😊",
36                                     reply_markup=ReplyKeyboardMarkup(
37                                         keyboard=[
38                                             [KeyboardButton(text="Entrada"), KeyboardButton(text="Passadís")],
39                                             [KeyboardButton(text="Menjador"), KeyboardButton(text="Sensors")],
40                                             [KeyboardButton(text="Habitació 1"), KeyboardButton(text="Habitació 2")],
41                                         ]
42                                     ))

```

*Codi 5: Programació de la botonera (Font: Pròpia)*

## 6.1. Control de casa

Com que hi ha bastantes variables i cadascuna té una funció en concret, s'ha d'evitar cometre l'error de generar un programa desordenat perquè al final si es troba un error i es vol corregir s'han de revisar moltes línies de codi. En aquest cas, l'estructuració del codi ha estat ordenada i separada per parts. A més, a causa de treballar des de dos entorns de programació diferents on un programa envia dades a l'altre, hem de tenir clar qué és el que envia cada variable per no confondre'ns.

Principalment la casa es pot controlar automàticament o manualment, n'és un bon exemple el control de les llums exteriors (el pati), podem activar el Mode ECO per a què quan arribi la nit no s'encenguin, amb l'objectiu d'obtenir un estalvi energètic i que només s'encenguin manualment, però també podem desactivar-lo per a què s'encenguin de manera automàtica dependent de la intensitat de la llum.

En la imatge següent veiem dos codis; el de la dreta controla el bot, el de l'esquerra controla l'Arduino;

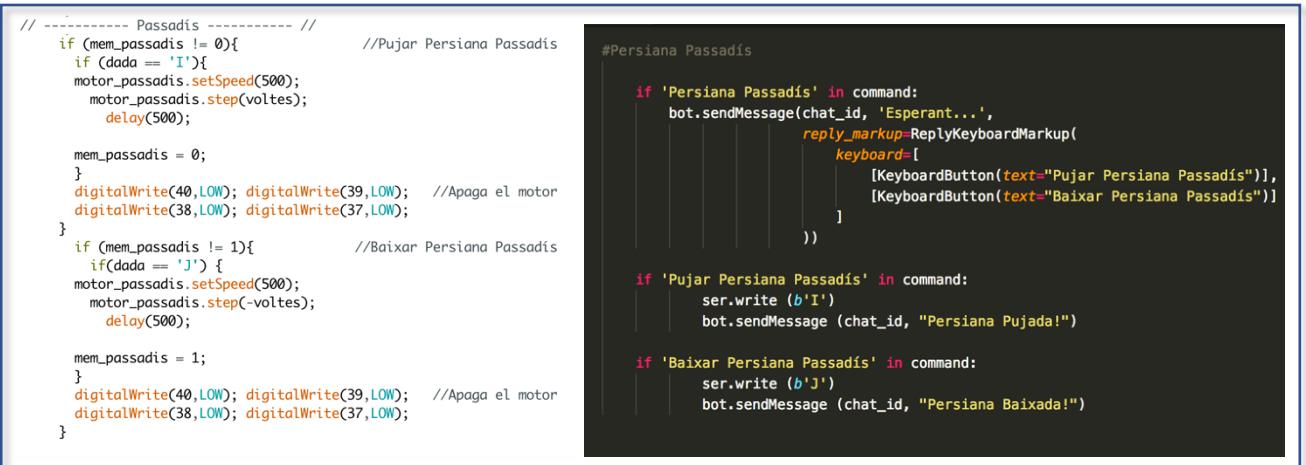
```
void loop() {
    // ----- AUTOMATITZACIÓ DES DE PYTHON -
    while (Serial.available()>0){ // Comprobem si hi ha dades
        dada = Serial.read();
    }
    // ----- Control Eco (Gestió LDR Automàtic
    if (dada == 'Y'){
        llums_auto = 0;
        llums_man = 1;
    }
    if (dada == 'Z'){
        llums_man = 0;
        llums_auto = 1;
    }
}

# ----- MODE ECO, func(LDR) -----
if 'Mode ECO' in command:
    bot.sendMessage(chat_id, 'Esperant...', reply_markup=ReplyKeyboardMarkup(
        keyboard=[[
            KeyboardButton(text="Activar Mode ECO"),
            KeyboardButton(text="Desactivar Mode ECO")
        ]])
)
if 'Activar Mode ECO' in command:
    ser.write (b'Y')
    bot.sendMessage (chat_id, "Mode ECO, Activat!")
if 'Desactivar Mode ECO' in command:
    ser.write (b'Z')
    bot.sendMessage (chat_id, "Mode ECO, Desactivat!")
```

Codi 6: Gestió del Mode ECO (Font: Pròpia)

Com podem veure en el Codi 6 hi ha dos scripts que junts controlen la gestió del Mode ECO, on el de la dreta (Bot) li envia dues dades (Y i Z) a través del port serial al de l'esquerra (Arduino). Quan l'Arduino rep les dades provinents del Bot, gestiona el control de cada una. En aquest cas, al rebre la dada Y l'Arduino segueix les instruccions programades i activa el Mode ECO, en canvi, si rep la dada Z el desactiva.

Per tenir una idea conceptual més àmplia de com funciona la casa i es controla explicarem la programació de les persianes que incorpora un sistema de memòries.



```

// ----- Passadis ----- //
if (mem_passadis != 0){           //Pujar Persiana Passadis
    if (dada == 'I'){
        motor_passadis.setSpeed(500);
        motor_passadis.step(voltes);
        delay(500);

        mem_passadis = 0;
    }
    digitalWrite(40,LOW); digitalWrite(39,LOW); //Apaga el motor
    digitalWrite(38,LOW); digitalWrite(37,LOW);
}
if (mem_passadis != 1){           //Baixar Persiana Passadis
    if(dada == 'J') {
        motor_passadis.setSpeed(500);
        motor_passadis.step(-voltes);
        delay(500);

        mem_passadis = 1;
    }
    digitalWrite(40,LOW); digitalWrite(39,LOW); //Apaga el motor
    digitalWrite(38,LOW); digitalWrite(37,LOW);
}

#Persiana Passadís

if 'Persiana Passadís' in command:
    bot.sendMessage(chat_id, 'Esperant...', reply_markup=ReplyKeyboardMarkup(
        keyboard=[

            [KeyboardButton(text="Pujar Persiana Passadís")],
            [KeyboardButton(text="Baixar Persiana Passadís")]
        ]
    ))

    if 'Pujar Persiana Passadís' in command:
        ser.write(b'I')
        bot.sendMessage (chat_id, "Persiana Pujada!")

    if 'Baixar Persiana Passadís' in command:
        ser.write(b'J')
        bot.sendMessage (chat_id, "Persiana Baixada!")

```

Codi 7: Sistema de memòries per als motors (Font: Pròpia)

Com ja és habitual i de sentit lògic totes les persianes elèctriques una vegada pugen o baixen no poden tornar a repetir el mateix moviment perquè s'espatlla el motor al girar en sentit contrari amb una forta resistència, per evitar aquest error es posa un final de carrera, un actuador que s'assimila a un interruptor que al ser premut no deixa que es torni a repetir el mateix moviment. Per fer-li això a la casa domòtica però sense instal·lar finals de carrera a la casa, s'ha ideat un sistema de memòries que guarda l'últim estat de la persiana.

### Explicació del Codi 7:

El programa del Bot sempre és el mateix, quan es clica sobre un botó per realitzar l'acció s'envia una dada pel serial a l'Arduino. Ara bé, el programa de l'Arduino no és igual, cada objecte es programa d'una manera diferent.

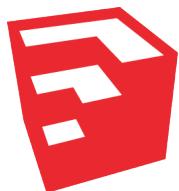
- 1) En tot el codi s'ha agafat un sistema de referència on el valor d'una persiana pujada és 0 i baixada és 1.
- 2) Quan l'Arduino rep la dada *I*, comprova si l'últim valor guardat és diferent a 0 per a què pugui pujar.
- 3) Si rep la dada *J*, comprova si l'últim valor guardat és diferent a 1 per a què pugui baixar.

## 7. DISSENY I CONSTRUCCIÓ DE LA CASA

Primerament, hem de tenir una idea conceptual de com ha de quedar la casa, quina distribució ha de tenir i plantejar-nos com s'han de distribuir tots els sistemes domòtics. S'ha creat i estructurat la distribució de la casa en SketchUp, un programa de disseny gràfic en 3D (tres dimensions). Encara que la importància del treball recau en la programació de la casa s'ha de tenir cura del disseny de la casa fent un bon plànol per acostar-nos el màxim possible a la realitat.

### 7.1. Model 3D

L'ús més quotidià que se li dona és en l'arquitectura i la creació d'objectes i peces simples. A més, com que disposem d'una impressora 3D que en parlarem més endavant, gràcies a al programa hem pogut crear peces i imprimir-les per mecanitzar com ara podria ser el sistema d'obertura de la porta del garatge, o el sistema d'enrotllament de les persianes.



Logotip 4: SketchUp  
(Font: Sketchup)

A base de molts dissenys finalment s'ha arribat a la conclusió que la distribució i la maqueta de la casa quedarà com la *Fig.15* amb una perspectiva superior com la *Fig.16*.



Fig. 15: Model 3D renderitzat de la casa (Font: Pròpia)

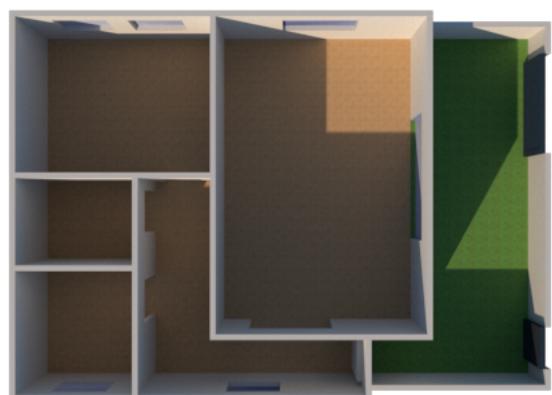


Fig. 16: Distribució amb perspectiva superior

- **Dimensions de la casa**

En la imatge següent hi ha representades totes les mesures de la casa.

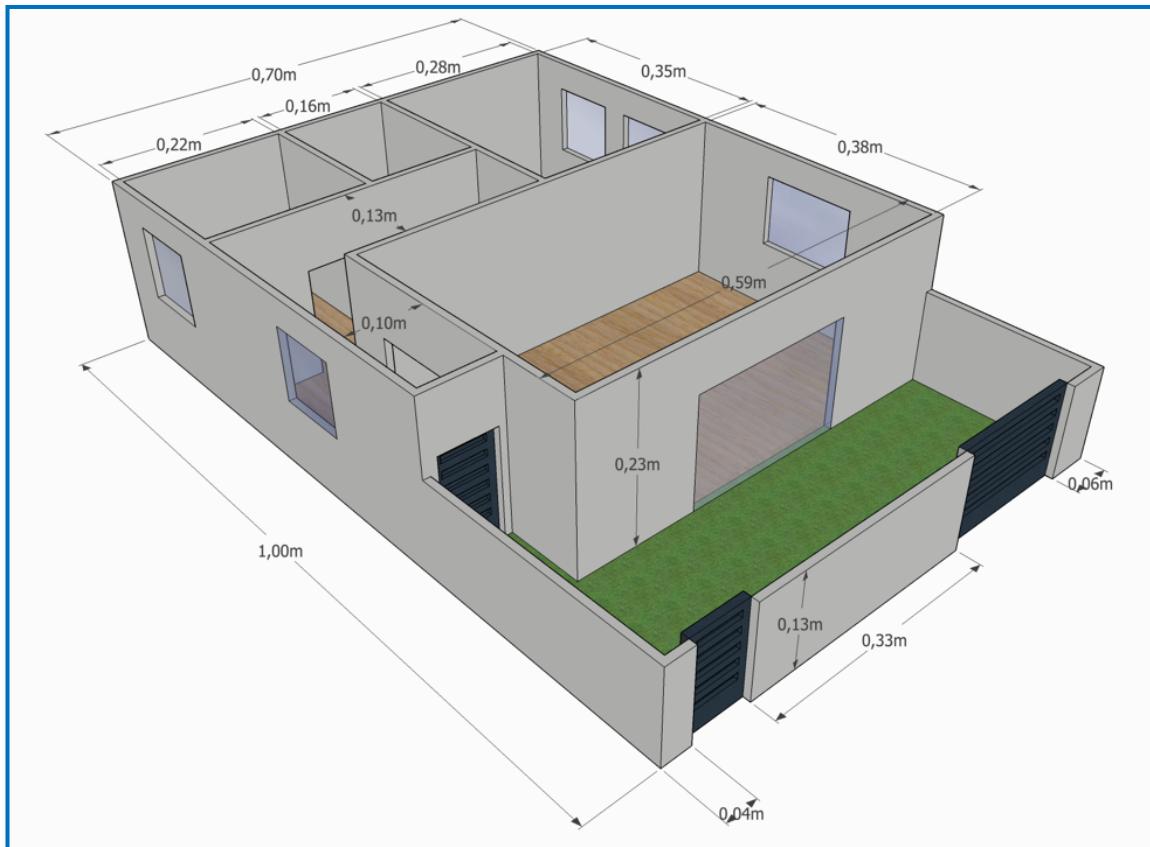


Fig. 17: Dimensions de la casa (Font: Pròpia)

## 7.2. Construcció

Una vegada tinguem el disseny i la distribució de la casa procedirem a fer-la a la realitat.

En la Fig. 18 es mostra el plànol de la casa:



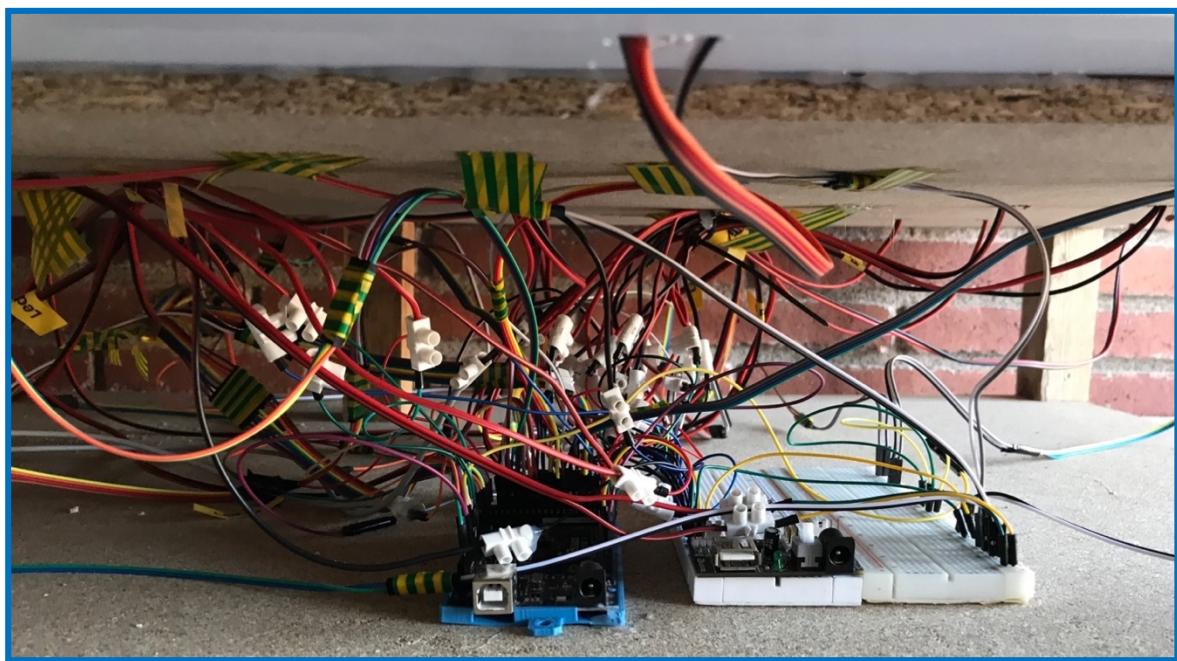
Fig. 18: Plànol de la casa (Font: Pròpia)

Per fer la casa més realista l'hem dissenyada amb paret doble com es veu en la *Fig. 19*, on els cables passen entre elles com és habitual en tots els circuits elèctrics de les cases. Cal saber que la casa es troba damunt d'una placa de fusta perquè la placa de cartró ploma és poc resistent i probablement sense la fusta inferior cauria. Les parets exteriors són més grans que les interiors com en la realitat, i les parts exteriors i interiors tenen una amplada de 2,5cm i 1,5cm respectivament.



*Fig. 19: Paret doble (Font: Pròpia)*

Tots els cables van a parar sota la casa on es troba la central domòtica (l'Arduino) per a què pugui ser controlada tal i com es mostra en la *Fig. 20*.



*Fig. 20: Central domòtica*

La principal dificultat del muntatge de la casa es troba en la connexió de cada component ja que hi ha molts llums led, sensors, motors i s'ha de tenir cura de connectar cada cosa a on toqui perquè tot està programat des de l'Arduino i ja s'ha declarat prèviament quin pin li toca a cada component electrònic. Per eliminar tot tipus de confusió s'ha creat anteriorment un Excel on s'ha fet una taula amb tots els pins de cada component.

<b>PATI</b>		<b>PASSADÍS</b>	
<u>Pins</u>	<u>Components</u>	<u>Pins</u>	<u>Components</u>
2 - 4	Llums led	A4 – A5	Llums led
7	Porta interior	37 - 40	Persiana passadís
8	Porta exterior	10	Sensor PIR (alarma)
12	Sensor IR	<b>HABITACIÓ 1</b>	
14 – 21	Teclat matricial 4x4	<u>Pins</u>	<u>Components</u>
25 - 28	Motor garatge	A6 – A7	Llums led
<b>HABITACIÓ 2</b>		33 - 36	Persiana habitació 1
<u>Pins</u>	<u>Components</u>	<b>MENJADOR</b>	
A10 – A12	Llums led	<u>Pins</u>	<u>Components</u>
41 – 44	Persiana habitació 2 (1)	A0 – A3	Llums led
45 – 48	Persiana habitació 2 (2)	11	Sensor DHT11
<b>BANY</b>		29 - 32	Persiana menjador
<u>Pins</u>	<u>Components</u>		
A8 – A9	Llums led		
9	Sensor PIR		

Taula 3: Pins dels components electrònics

- Pàgina web del treball

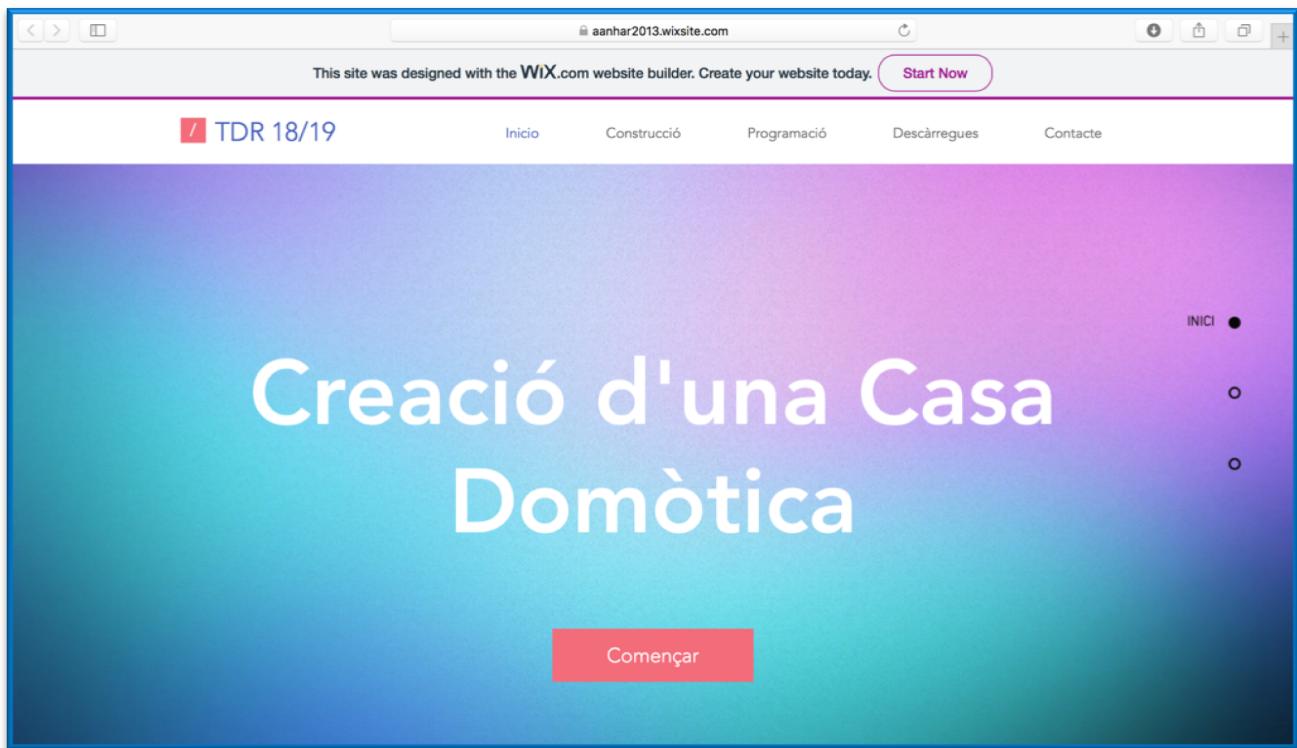


Fig. 21: **Pàgina web del treball (Font: Pròpia)**

Per obtenir informació més detallada sobre el procediment de construcció i programació s'ha creat una pàgina web (Fig. 21). S'ha penjat un vídeo sobre el procediment constructiu que també explica el funcionament de la casa.

Els codis del Bot i del Arduino es troben en la pàgina per a la gent més interessada en obtenir-los.



QR 2: **Pàgina web del TDR**

### 7.3. Impressió 3D

Anteriorment hem dit que disposem d'una impressora 3D per poder crear peces i mecanismes per a la casa, la impressora que hem utilitzat ha sigut la **Anycubic I3 Mega** que és la de la Fig. 22.

Abans, les impressores 3D quan havien sortit eren molt cares ja que era la primera vegada que s'introduïa al mercat el món del 3D. Molta gent es va sorprendre de la gran capacitat que tenien aquestes impressores d'imprimir objectes en 3D i de fer proves per veure com sortiria un objecte en la vida real imprès en un altre material que no fos plàstic.

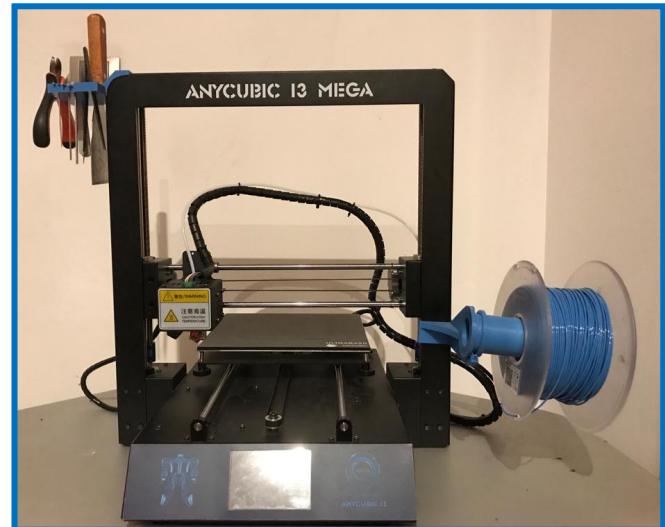


Fig. 22: Impressora 3D Anycubic I3 Mega (Font: Pròpia)

En el nostre cas la impressora és molt bona en quant a qualitat i preu, i hem de tenir uns coneixements previs abans d'imprimir. Una impressora consta de quatre motors i tres d'ells representen els eixos X, Y i Z per a què es pugui moure per la placa, el quart motor es dedica a extrudir el filament de plàstic. A la Fig. 23 veiem un petit resum de la impressió 3D.

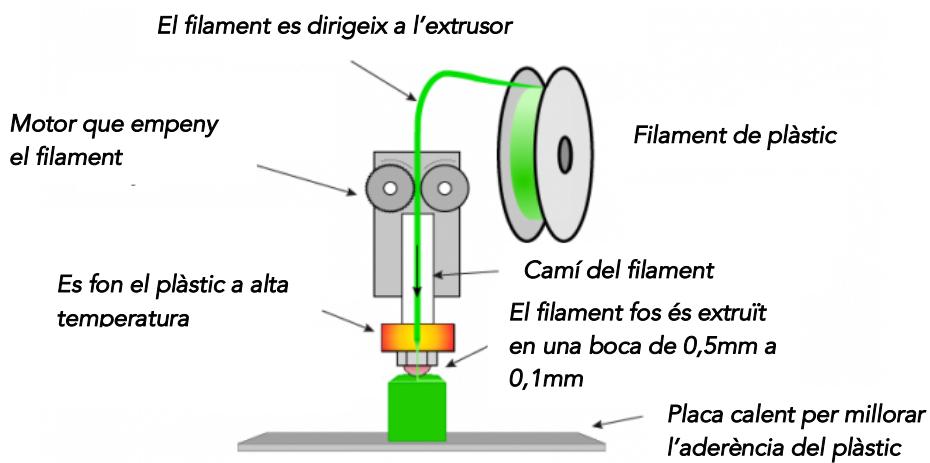


Fig. 23: Esquema d'una impressió en 3D (Scoop.it)

### 7.3.1. Ultimaker Cura

Ultimaker Cura és un programa indispensable per una impressió 3D. El que fa és convertir un objecte creat amb qualsevol programa de disseny 3D, que en aquest cas és SketchUp i transformar-lo en una sèrie d'instruccions per a què la impressora les pugui interpretar i pugui imprimir. El programa es veu de la següent manera:

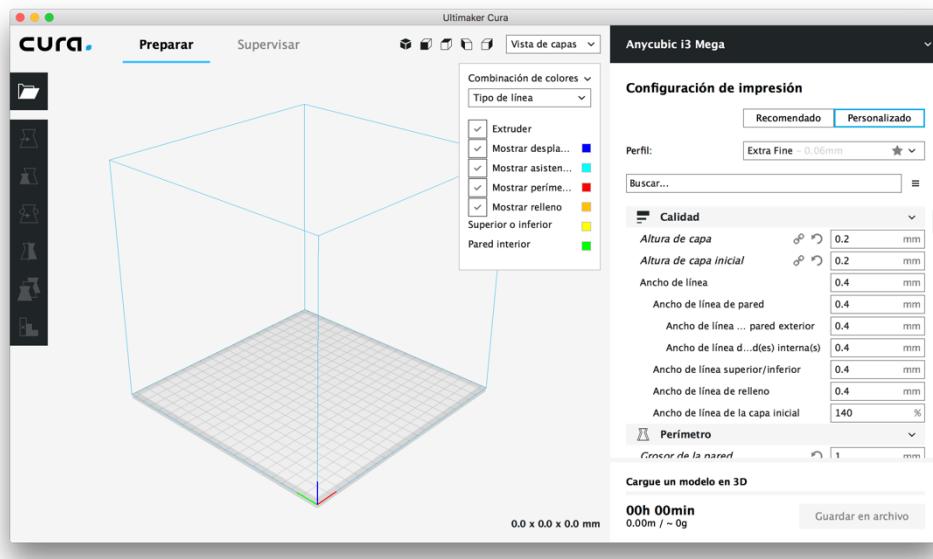


Fig. 24: **Ultimaker Cura** (Font: Pròpia)

Quan es té acabat un objecte s'exporta en un arxiu en format **.STL** (de l'anlgés, "Standart Triangle Language") que és el més senzill ja que com el nom indica ens guarda la superfície geomètrica de l'objecte en un conjunt de triangles on les unitats són arbitràries, és a dir, podem modificar en el programa les dimensions de l'objecte, escalant-lo uniformement. Després, l'arxiu s'obre en Ultimaker Cura i ens genera un **G-code**, que és un llenguatge molt habitual per a les màquines utilitzat en control numèric. Resumidament el que fa el programa és analitzar l'objecte, amb les configuracions que hem seleccionat i que després en parlarem i ens genera un arxiu amb moltíssimes línies d'instruccions indicant-li a tots els motors a quina coordenada han de anar, quan l'extrusor ha d'extrudir plàstic, quan s'ha d'escalfar l'extrusor, etc.

Per tenir una idea del món de la impressió 3D cal tenir tres coneixements bàsics i fonamentals:

- **Altura de capa**

L'altura de capa és el gruix que ha de tenir el filament de plàstic quan és extrudit, per definir la qualitat d'impressió cal donar un valor entre 0,05mm a 0,4mm depenent del diàmetre de la embocadura.

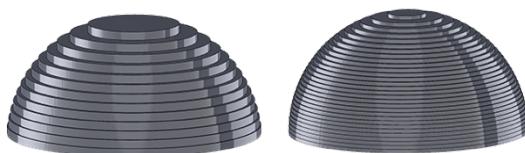


Fig. 25: Gruix de capa (Font: 3DVerkstan)

Com observem en la Fig. 25 veiem que com menys altura de capa es selecciona més alta és la qualitat de l'objecte però cal tenir en compte que el temps de la impressió arriba a ser quatre o cinc vegades més gran.

- **Estructura interna d'una peça**

Qualsevol peça ha de tenir una estructura interna ja que sinó la peça estaria buida i tindria una estructura molt feble.

En la Fig. 26 es veu la estructura interna d'un cilindre.

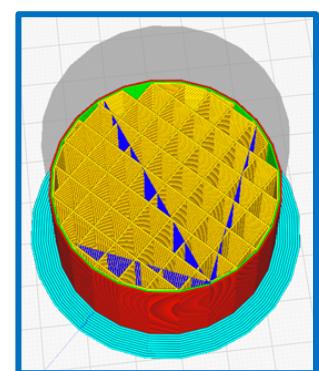


Fig. 26: Estructura interna d'un cilindre (Font: Pròpia)

- **Suports**

Com la paraula diu un suport subjecta la part d'una peça que es troba a amb un angle on la impressora li és impossible imprimir.

En la imatge següent es vol imprimir la mateixa peça dues vegades però a una se li col·loca el suport per a què no es caigui i l'altra es col·loca d'una manera eficient

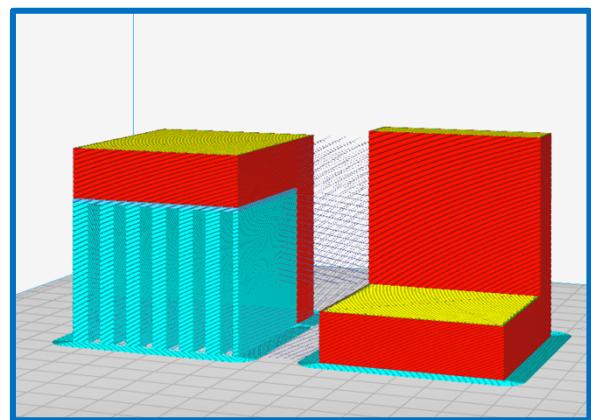


Fig. 27: Figura amb suport i sense (Font: Pròpia)

### 7.3.2. Sistemes mecanitzats

Tots els sistemes mecànics incorporats a la casa han sigut dissenyats en SketchUp i impresos en 3D.

- Persianes

Per moure la persiana, el motor es manté enganxat a la paret amb una peça impresa en 3D i el moviment de gir es transmet amb una tapa amb un forat de la mida de la entrada del motor enganxada a una barra de fusta i sostinguda per una altra peça amb un diàmetre més gran perquè la barra pugui moure's.

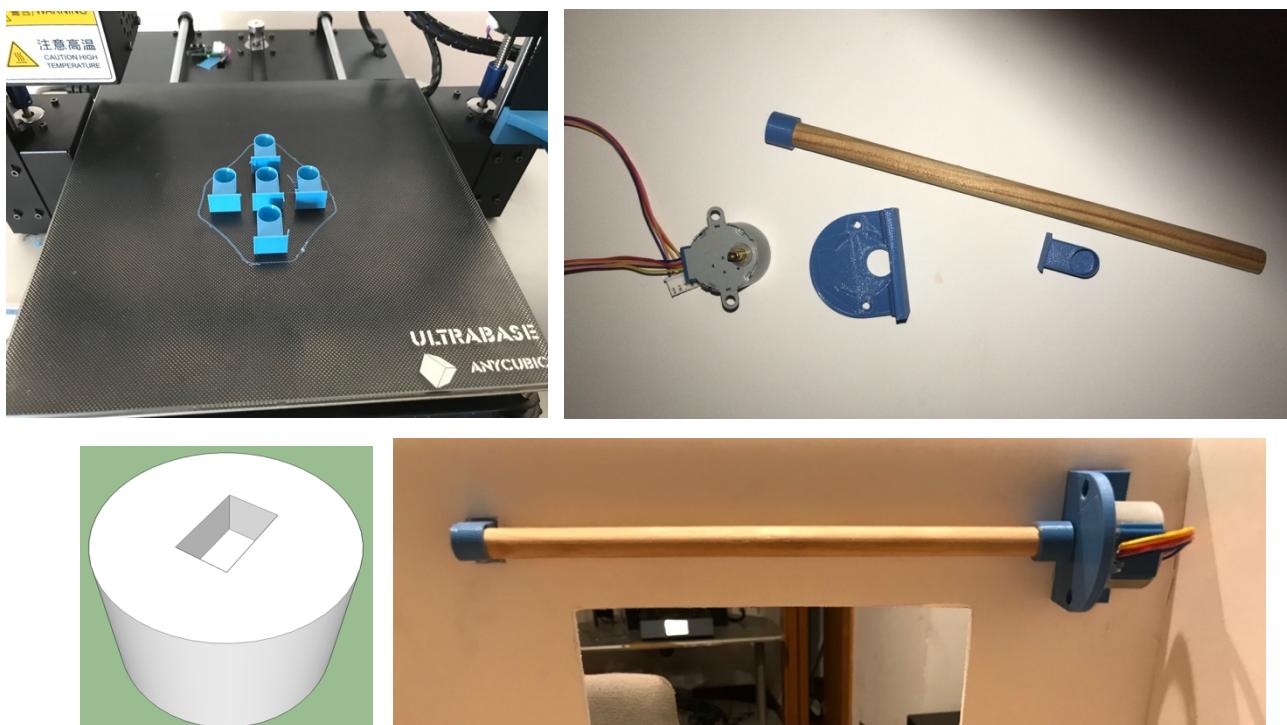


Fig. 28: Mecanisme de la persiana (Font: Pròpia)

- **Portes amb servos**

Com es mostra a continuació totes les portes de la casa estan enganxades a un servo motor que permet obrir-les a l'instant. La porta exterior té una peça impresa en 3D que la subjecta.

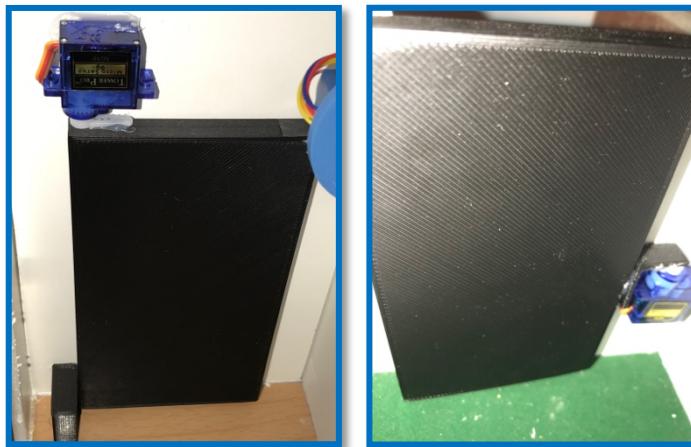


Fig. 29: Portes de la casa

- **Porta amb sistema d'engranatge**

Aquest és el mecanisme que s'ha tardat en dissenyar i imprimir. La porta del garatge es mou gràcies a un motor pas a pas, i se li ha dissenyat un engranatge que encaixi dins de la porta. Principalment l'engranatge s'ha dissenyat amb un programa que s'anomena FreeCAD, i l'única cosa que s'ha de fer és escriure el valor del diàmetre i el número de dents. Després per acabar s'edita amb el Sketchup.

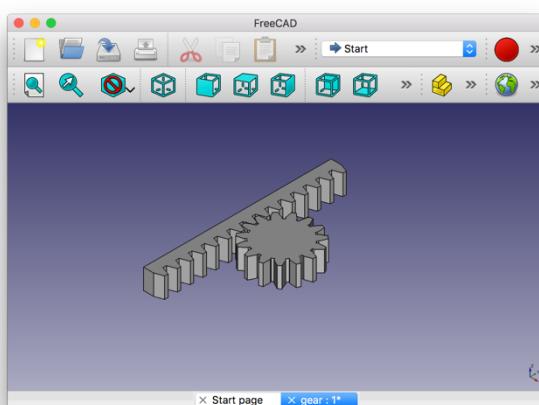


Fig. 31: FreeCAD (Font: Pròpia)



Fig. 30: Impressió de la porta (Font: Pròpia)

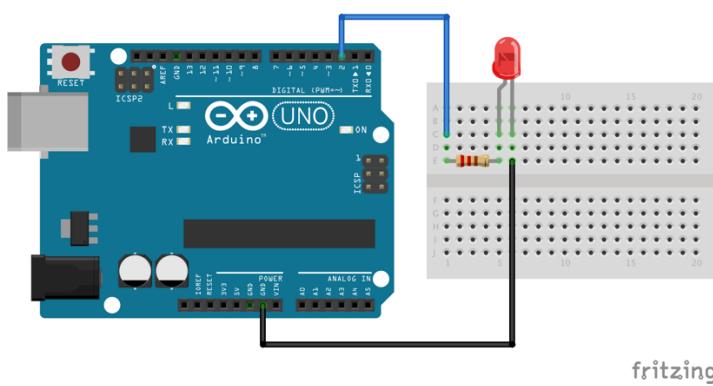
## 7.4. Components electrònics

En aquest apartat s'ha de tenir molta cura a l'hora de connectar cada component electrònic. S'han de tractar amb delicadesa ja que en l'electrònica es treballa amb un voltatge i corrent de funcionament en concret. N'és un bon exemple un led, que cal col·locar-li una resistència que normalment és de  $220\Omega$  perquè no es cremi ja que treballa a una intensitat petita. Subministrar un gran voltatge a un component no és el causant de que es cremi sinó que és el corrent elèctric que és la quantitat d'electrons que es troben en un punt del circuit i el voltatge és la força que fa moure els electrons del corrent. De fet el corrent que passa per un led podria provocar-nos una parada cardíaca però per això quan toquem el led o la seva pota no ens passa res, perquè el voltatge no té la suficient força per circular pel nostre cos perquè li oposem una major resistència.

$$V = R \cdot I \rightarrow I = \frac{V}{R}$$

En la llei d'Ohm com podem veure el voltatge és el producte entre la resistència i la intensitat, on el voltatge és directament proporcional, com més alt sigui el voltatge més gran serà el corrent. Però és equivalent a dir que la intensitat és el quotient entre el voltatge i la resistència, on la intensitat és indirectament proporcional a la resistència, com més elevat sigui el valor de la resistència menys intensitat hi haurà.

Per això sempre posem una resistència a un led ja que treballa a intensitats petites i la resistència redueix el valor del corrent perquè no es cremi. Convé saber que un led té dues potes i una és més llarga que l'altra. La pota més llarga amb una resistència incorporada es connecta als pins d'entrada de l'Arduino i la curta es connecta al GND (el terra).



Circuit 1: Connexió d'un led a l'Arduino (Font: Pròpia)

- Sensor DHT11

El sensor DHT11 mesura la humitat i la temperatura de l'ambient i podem veure com funciona en la Fig. 32 es rep al Bot la humitat mesurada pel sensor.



Fig. 32: Funcionament del DHT11 (Font: Pròpia)

- Mòdul d'infrarojos i sensor LDR

El sensor d'infrarojos és dedicat a rebre informació d'un aparell de control com es mostra en la imatge per encendre o apagar els llums, obrir o tancar la porta del garatge i activar o desactivar la alarma.

El sensor LDR s'encarrega de mesurar la intensitat lumínica que en aquest cas ens serveix per gestionar el Mode ECO explcitat anteriorment (apartat 6.1. Control de la casa).



Fig. 33: Mòdul d'infrarojos i aparell de control (Font: CDMX)



Fig. 34: Fotoresistència o LDR (Font: PotentialLabs)

- Sensor PIR

En tota la casa hi ha 2 sensors de moviment, el de la dreta detecta moviment per l'alarma i pot ser activat o desactivat des del Bot o des de l'aparell de control manualment. El de l'esquerra encén els llums del bany per amb la detecció de moviment automàticament.



Fig. 35: Sensors PIR (Font: Pròpia)

- Mòdul RFID

Aquest mòdul s'usa per la seguretat de la cassa, per accedir per la porta exterior del pati. L'RFID envia senyals de radiofreqüència i quan entra en contacte amb la etiqueta gravada interna d'una targeta, la llegeix i avalua si és la correcta.



Fig. 36: Sensor RFID  
(Font: Robomart)

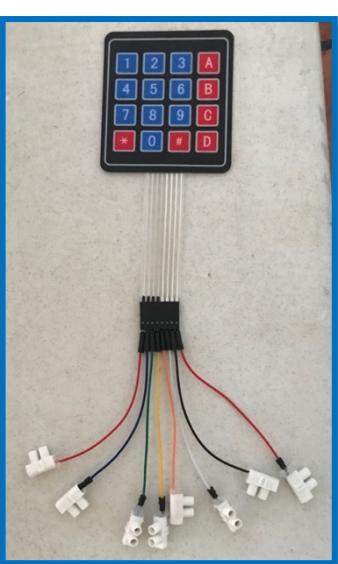


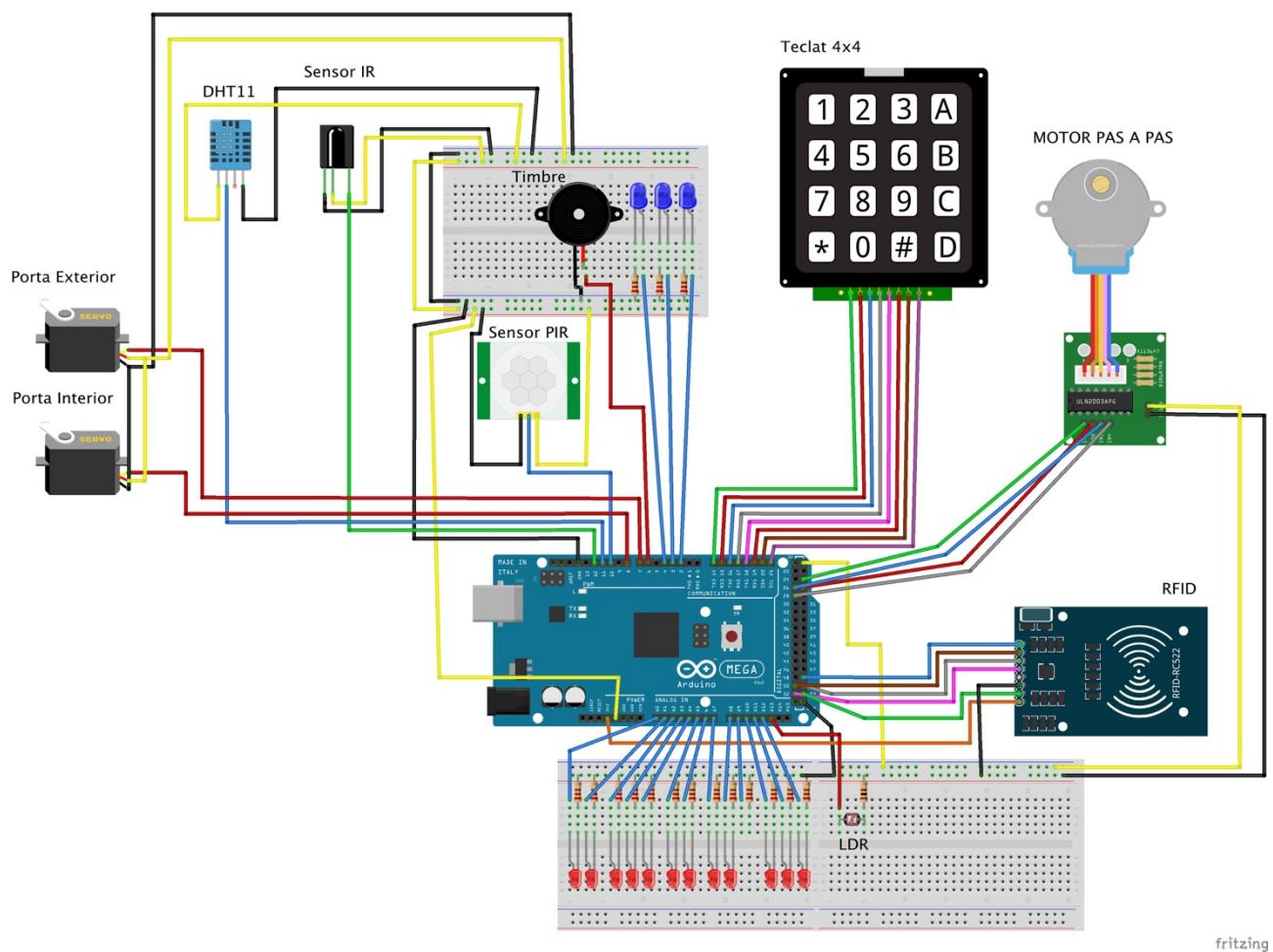
Fig. 37: Sensor RFID (Font: Pròpia)

Amb aquest component aconseguim tenir un altre sistema de seguretat per accedir a la porta principal de la casa.

S'ha de programar una contrasenya i a l'hora d'escriure-la correctament l'Arduino ens obrirà la porta. En el cas d'introduir una contrasenya errònia l'alarma s'activarà.

### 7.4.1. Esquemes

A la imatge següent veiem un esquema amb les connexions dels pins de cada component dissenyat en Fritzing, un programa que permet crear qualsevol tipus d'esquema. Per aquest en concret, s'ha hagut d'instal·lar alguns components des d'una font externa ja que són elements específics utilitzats en Arduino.



Circuit 2: Instal·lació elèctrica de la casa (Font: Pròpia)

## 7.5. Pressupost

PRESSUPOST			
Material	Unitats	Preu per unitat	Preu total
Kit Arduino Mega	1	59,99 €	59,99 €
Placa cartró ploma (1m x 0,7m x 5mm)	9	4,18 €	37,62 €
Placa de fusta (1m x 0,6m)	2	3,75 €	7,50 €
Cantonera 1,5m	4	2,35 €	9,40 €
Barra de fusta 50cm ( $\varnothing = 2$ cm)	3	1,50 €	4,50 €
Recobriment "gespa" (1m x 1m)	1	4,00 €	4,00 €
Recobriment plàstic (1m x 0,5m)	2	2,00 €	4,00 €
Làmina de plàstic (1m x 1m)	1	3,50 €	3,50 €
Cinta adhesiva	2	1,00 €	2,00 €
Brides (120/paquet)	1	0,75 €	0,75 €
Barres de silicona (50 unitats/paquet)	1	5,00 €	5,00 €
Cable 10m ( $\varnothing = 5$ mm)	2	4,50 €	9,00 €
Sensor PIR	1	1,95 €	1,95 €
Llum LED	16	0,19 €	3,04 €
Jumpers (cables d'Arduino)	5	3,99 €	19,95 €
Motors pas a pas	6	3,17 €	19,02 €
Servo motor	1	2,64 €	2,64 €
Regleta connexió	10	0,99 €	9,90 €
Total (IVA inclòs)			203,76 €

Com es pot apreciar, la construcció de la casa no és econòmica ja que hi ha materials que són molt cars com per exemple, el cartró ploma, però és veritat que es podria haver reduït el cost eliminant la doble paret però no quedaría igual d'original com ha resultat finalment. En el pressupost només surten els materials que han estat comprats la majoria de sensors ja venen en un Kit d'Arduino Mega que també està inclòs en el pressupost.

## 8. CONCLUSIONS

Al principi es van marcar uns objectius els quals s'han complert però no del tot. Hi ha dos apartats que es podrien haver millorat; trobar l'estabilitat domòtica de la casa i comunicar l'Arduino al Bot.

Podem corroborar que la domòtica ens facilita el control de la casa sempre i quan estiguï ben integrada, però aquí és on es troba el problema, s'ha de crear un sistema domòtic estable. Per entendre-ho, n'és un bon exemple el control de les persianes, que pot ser automàtic dependent de la lluminositat que es detecti a l'ambient o manualment baixant-la des del Bot, però no queda clar com funciona la persiana ja que té dos punts de control, per això, el que es podria haver fet, seria fer més proves a la casa per poder gestionar les persianes amb un control estable des dels dos punts.

En quant a la comunicació, com hem vist al llarg del treball el Bot exerceix el control sobre l'Arduino de manera directa però el que no s'ha pogut fer és que l'Arduino es comunique amb el Bot per a què pugui enviar-li notificacions en qualsevol circumstància ja que és la primera vegada que programem en Python i per poder programar el Bot requereix un nivell de programació més alt ja que els dos llenguatges són molt diferents.

Tot s'ha planejat des d'un principi però degut a la dificultat del muntatge i les hores que se li han dedicat a la casa, el treball s'ha vist afectat en quant a temps. La cara positiva és que tot ha sortit com s'esperava deixant de banda les petites imperfeccions.

En resum, des del meu punt de vista he après moltes coses, sobretot dels errors comesos, i m'he adonat que si jo hagués de treballar en grup, mai em posaria a la banda de construcció perquè requereix molt paciència i perfeccionisme, en canvi, a l'hora de programar m'ho passo molt millor ja que quan programo el que estic fent és jugar amb l'ordinador.

## 9. BIBLIOGRAFIA WEB

### ■ Pàgines web

- Arduino (s/d). *Language Reference* [en línia]. [Consultat el Setembre 2018]. Disponible a Internet: <https://www.arduino.cc/reference>
- Geom (2017 - 2018). Domótica [en línia]. [Consultat l'Abril 2018] Disponible a Internet: <https://es.wikipedia.org/wiki/Domótica>
- Anònim (s/d). Sistemas domóticos, centralizados, descentralizados y distribuidos. [en línia]. [Consultat l'Abril 2018]. Disponible a Internet: <http://hogartec.es/hogartec2/sistemas-domoticos-centralizados-descentralizados-y-distribuidos/>
- Anònim. [aloishemia]. (2016, Maig 9). How to use ReplyKeyboardMarkup. [Publicació en un fòrum en línia]. (Consultat el Juny 2018). Disponible a Internet: <https://github.com/nickoala/telepot/issues/74>

### ■ Vídeos

- Anònim. [Automating OSINT]. (2015, Juny 2). Installing Python Pip Mac OSX. Recuperat de <https://www.youtube.com/watch?v=yBdZZGPpYxg>
- Corey, S. [Corey Shafer]. (2017, Abril 13). Tutorial: virtualenv and why you should use virtual environments. Recuperat de <https://www.youtube.com/watch?v=N5vscPTWKOk>
- Keith, W. [Keith, the Coder]. (2017, Setembre 6). Installing Pip3 & Uninstalling Pip3 (Python 3 Package Manager) for Mac OSX in one line. Recuperat de [https://www.youtube.com/watch?v=W3k8\\_8ahsyg](https://www.youtube.com/watch?v=W3k8_8ahsyg)
- Miguel, R. [Mike Angeloo]. (2017, Novembre 9). Telegram BOT para control de domòtica a distància. Recuperat de [https://www.youtube.com/watch?v=U-A\\_Raj2zps](https://www.youtube.com/watch?v=U-A_Raj2zps)

### ■ Documents

- Simon Monk. (2017). El Kit más completo, tutorial MEGA2560. Versió V1.0.17.9 [Arxiu PDF] adquirit amb la compra del Kit però no es troba en línia
- Albert Babí. (2017). Internet de les coses. [Archiu PDF] de provinença desconeguda.

- **Imatges**

- QR 1: <https://goo.gl/CqGw4s>
- QR 2: <https://aanhar2013.wixsite.com/casadomotica>
- Fig. 1 [Editada]: <https://goo.gl/cSG1bc>
- Fig. 2: <https://goo.gl/ohdXCa>
- Fig. 3: <https://goo.gl/bC1k4P>
- Fig. 5: <https://goo.gl/4KbJ6P>
- Fig. 23: <https://goo.gl/AtfMN4>
- Fig. 25: <https://goo.gl/rmGh4q>
- Fig. 33: <https://goo.gl/AWG7GG> i <https://goo.gl/Y1tDrR>
- Fig. 34: <https://goo.gl/vPxX9z>
- Fig. 36: <https://goo.gl/LrKSW6>
- Esquema 4: Arxiu PDF de provinença desconeguda
- Esquema 5: Arxiu PDF de provinença desconeguda
- Esquema 6: Arxiu PDF de provinença desconeguda
- Logotip 1: <https://goo.gl/WmPuDc>
- Logotip 2: <https://goo.gl/8kTCwj>
- Logotip 3: <https://goo.gl/2qpdww>
- Logotip 4: <https://goo.gl/ozLtUw>

## ANNEX

### Codi de l'Arduino (C++)

```
// --- Python ---
unsigned int dada; // Variable "dada" és on es guardaran els valors rebuts des
de Python

// --- Llums ---

//Menjador - Cuina
int led2 = A0;
int led3 = A1;
int led4 = A2;
int led5 = A3;

//Passadís
int led1 = A4;
int led6 = A5;

//Habitació 1
int led7 = A6;
int led8 = A7;

//Bany
int led9 = A8;
int led10 = A9;

//Habitació 2
int led11 = A10;
int led12 = A11;
int led13 = A12;

//Exterior
int led14 = 2;
int led15 = 3;
int led16 = 4;

// ----- Timbre -----
const int boto_timbre = 5;
const int timbre = 6;

// ----- Sensor LDR (Lluminositat) -----
const int ldr = A13;
int llums_auto;
int llums_man;

// ----- Sensors PIR (Moviment) -----
const int PIR_Bany = 9;
const int PIR_Alarma = 10;

int PIR_Alarma_ON;
int PIR_Alarma_OFF;

// ----- Sensor DHT11 (Temperatura i Humitat) -----
#include <DHT.h>
const int sensorDHT = 11;
int temperatura, humitat;
DHT dht(sensorDHT, DHT11); //Objecte amb les variables definides
anteriorment.
```

```

// ----- Teclat 4x4 -----
#include <Keypad.h>
#include <Password.h>

>Password password1 = Password("123A"); //Definim la nostra contrasenya
>Password password2 = Password("123B");

const byte FILES = 4; // Quatre Files
const byte COLUMNES = 4; // Quatre Columnes
// Definim els Keys
char tecles[FILES][COLUMNES] = {
{'1','2','3','A'},
{'4','5','6','B'},
{'7','8','9','C'},
{'*','0','#','D'}
};

byte filesPins[FILES] = {21, 20, 19, 18}; // Conectar Files (4 pins)
byte columnesPins[COLUMNES] = {17, 16, 15, 14}; // Conectar Columnes (4 pins)

Keypad keypad = Keypad( makeKeymap(tecles), filesPins, columnesPins, FILES,
COLUMNES );

// --- Sensor IR ---
#include <IRremote.h>
const int receptorIR = 12;
IRrecv irrecv(receptorIR);
decode_results Codis;

// --- Servos ---
#include <Servo.h>
Servo Porta_Exterior;
Servo Porta_Interior;

// --- Motors pas a pas ---
#include "Stepper.h"

int voltes = 4100; //--> 1 Volta = 2048
const int PASSOS = 48;

Stepper motor_menj_1 (PASSOS, 32, 30, 31, 29); // In1, In2, In3, In4 en una
seqüència de 1-3-2-4
int mem_menjador = 0;

Stepper motor_passadis (PASSOS, 40, 38, 39, 37);
int mem_passadis;

Stepper motor_hab_1 (PASSOS, 36, 34, 35, 33);
int mem_hab_1 = 0;

Stepper motor_hab_2_1(PASSOS, 48, 46, 47, 45);

Stepper motor_hab_2_2(PASSOS, 44, 42, 43, 41 );
int mem_hab_2 = 0;

Stepper motor_garatge (PASSOS, 28, 26, 27, 25);
int mem_garatge = 0;

```

```

// --- Sensor RFID ( Ocupa els pins --> 53, 52, 51, 50, --, [GND], 49, [3.3V] )
--- //
#include <SPI.h>
#include <MFRC522.h>
const int SS_PIN = 53; // *#define*
const int RST_PIN = 49; // *#define*
MFRC522 RFID (SS_PIN, RST_PIN); //Objecte amb les variables definides
anteriorment.

//----- Accions Programades -----
-----
void setup() {
  Serial.begin (9600);
  SPI.begin();

  // -- Sensors -- //
  pinMode(ldr, INPUT); // LDR --> Llum --> A13
  pinMode(PIR_Bany, INPUT); // PIR --> Sensor de Moviment --> 9
  pinMode(PIR_Alarma, INPUT); // PIR --> Sensor de Moviment --> 10

  irrecv.enableIRIn(); // Control IR --> Infraroigos --> 12
  RFID.PCD_Init(); // RFID --> Magnètic -->
  dht.begin (); // DHT11 --> Humitat && Temperatura --> 11

  // -- Llums -- //
  pinMode(led2,OUTPUT); // --> A0
  pinMode(led3,OUTPUT); // --> A1
  pinMode(led4,OUTPUT); // --> A2
  pinMode(led5,OUTPUT); // --> A3

  pinMode(led1,OUTPUT); // --> A4
  pinMode(led6,OUTPUT); // --> A5

  pinMode(led7,OUTPUT); // --> A6
  pinMode(led8,OUTPUT); // --> A7

  pinMode(led9,OUTPUT); // --> A8
  pinMode(led10,OUTPUT); // --> A9

  pinMode(led11,OUTPUT); // --> A10
  pinMode(led12,OUTPUT); // --> A11
  pinMode(led13,OUTPUT); // --> A12

  pinMode(led14,OUTPUT); // --> 2
  pinMode(led15,OUTPUT); // --> 3
  pinMode(led16,OUTPUT); // --> 4

  // -- Timbre -- //
  pinMode(boto_timbre, INPUT); // Polsador --> 5
  pinMode(timbre, OUTPUT); // Buzzer --> 6

  // -- Teclat Matricial (4x4) -- //
  keypad.addEventListener(keypadEvent);
  // Servo en funció del teclat
  Porta_Exterior.attach(7); // Exterior --> 7
  Porta_Exterior.write(0);
  Porta_Interior.attach(8); // Interior --> 8
  Porta_Interior.write(0);
}

void loop() {

```

```

// ----- AUTOMATITZACIÓ DES DE PYTHON -----
while (Serial.available()>0){ // Comprovem que la comunicació serial
estigui disponible

    dada = Serial.read() ;

// ----- Control Eco (Gestió LDR Automàtic o Manual) -----
if (dada == 'Y'){ // Manual
    llums_auto = 0;
    llums_man = 1;
}

if (dada == 'Z'){ // Automàtic
    llums_man = 0;
    llums_auto = 1;
}

if (dada == '1'){ // Alarma Activada
    PIR_Alarma_ON = 1;
    PIR_Alarma_OFF = 0;
}

if (dada == '2'){ // Alarma Desactivada
    PIR_Alarma_ON = 0;
    PIR_Alarma_OFF = 1;
}

// ----- Exterior (Pati) -----
if(dada == 'A') { //Obrir Porta Exterior
    Porta_Exterior.write(90);
    break;
}

if(dada == 'B') { //Tancar Porta Exterior
    Porta_Exterior.write(0);
}

if (mem_garatge != 1){ //Obrir Porta Garatge
    if (dada == 'C'){

        motor_garatge.setSpeed(500);
        motor_garatge.step(4050);

        mem_garatge = 1;
    }
    digitalWrite (28, LOW);
    digitalWrite (27, LOW);
    digitalWrite (26, LOW);
    digitalWrite (25, LOW);
}

if (mem_garatge != 0){ //Tancar Porta Garatge
    if (dada == 'D'){

        motor_garatge.setSpeed(500);
        motor_garatge.step(-4050);
}

```

```

        mem_garage = 0;
    }
    digitalWrite (28, LOW);
    digitalWrite (27, LOW);
    digitalWrite (26, LOW);
    digitalWrite (25, LOW);
}

// ----- Passadís -----
if(dada == 'G') { //Encendre Llums Passadís
    digitalWrite (led1, HIGH);
    digitalWrite (led6, HIGH);
}
if(dada == 'H') { //Apagar Llums Passadís
    digitalWrite (led1, LOW);
    digitalWrite (led6, LOW);
}

// ----- Passadís -----
if (mem_passadis != 0){ //Pujar Persiana Passadís
    if (dada == 'I'){
        motor_passadis.setSpeed(500);
        motor_passadis.step(voltes);
        delay(500);

        mem_passadis = 0;
    }
    digitalWrite(40,LOW); digitalWrite(39,LOW); //Apaga el motor
    digitalWrite(38,LOW); digitalWrite(37,LOW);
}
if (mem_passadis != 1){ //Baixar Persiana Passadís
    if(dada == 'J') {
        motor_passadis.setSpeed(500);
        motor_passadis.step(-voltes);
        delay(500);

        mem_passadis = 1;
    }
    digitalWrite(40,LOW); digitalWrite(39,LOW); //Apaga el motor
    digitalWrite(38,LOW); digitalWrite(37,LOW);
}

// ----- Menjador -----
if(dada == 'K') { //Encendre Llums Menjador
    digitalWrite (led2, HIGH);
    digitalWrite (led3, HIGH);
    digitalWrite (led4, HIGH);
    digitalWrite (led5, HIGH);
}
if(dada == 'L') { //Apagar Llums Menjador
    digitalWrite (led2, LOW);
    digitalWrite (led3, LOW);
    digitalWrite (led4, LOW);
    digitalWrite (led5, LOW);
}

if (mem_menjador != 0){
if(dada == 'M') { //Pujar Persianes Menjador
    motor_menj_1.setSpeed (500);
    motor_menj_1.step(voltes);

    mem_menjador = 0;
}
}

```

```

        }
        digitalWrite (29, LOW);
        digitalWrite (30, LOW);
        digitalWrite (31, LOW);
        digitalWrite (32, LOW);
    }
    if (mem_menjador != 1){
        if(dada == 'N') { //Baixar Persianes Menjador
            motor_menj_1.setSpeed (500);
            motor_menj_1.step (-voltes);

            mem_menjador = 1;
        }
        digitalWrite (29, LOW);
        digitalWrite (30, LOW);
        digitalWrite (31, LOW);
        digitalWrite (32, LOW);
    }

// ----- Habitació 1 -----
    if(dada == 'O') { //Encendre Llums Habitació 1
        digitalWrite (led7, HIGH);
        digitalWrite (led8, HIGH);
    }
    if(dada == 'P') { //Apagar Llums Habitació 1
        digitalWrite (7, LOW);
        digitalWrite (8, LOW);
    }

    if (mem_hab_1 != 0){
        if(dada == 'Q') { //Pujar Persianes Habitació 1
            motor_hab_1.setSpeed(500);
            motor_hab_1.step(voltes);

            mem_hab_1 = 0;
        }
        digitalWrite (33, LOW);
        digitalWrite (34, LOW);
        digitalWrite (35, LOW);
        digitalWrite (36, LOW);
    }
    if (mem_hab_1 != 1){
        if(dada == 'R') { //Baixar Persianes Habitació 1
            motor_hab_1.setSpeed(500);
            motor_hab_1.step (-voltes);

            mem_hab_1 = 1;
        }
        digitalWrite (33, LOW);
        digitalWrite (34, LOW);
        digitalWrite (35, LOW);
        digitalWrite (36, LOW);
    }

// ----- Habitació 2 -----
    if(dada == 'S') { //Encendre Llums Habitació 2
        digitalWrite (led11, HIGH);
        digitalWrite (led12, HIGH);
        digitalWrite (led13, HIGH);
    }
    if(dada == 'T') { //Apagar Llums Habitació 2
        digitalWrite (led11, LOW);
        digitalWrite (led12, LOW);
    }
}

```

```

        digitalWrite (led13, LOW);
    }

    if (mem_hab_2 != 0) {
        if(dada == 'U') { //Pujar Persianes Habitació 2
            motor_hab_2_1.setSpeed(500);
            motor_hab_2_1.step(voltes);

            motor_hab_2_2.setSpeed(500);
            motor_hab_2_2.step(-voltes);

            mem_hab_2 = 0;
        }
        digitalWrite(48,LOW);
        digitalWrite(47,LOW);
        digitalWrite(46,LOW);
        digitalWrite(45,LOW);
        digitalWrite(41,LOW);
        digitalWrite(42,LOW);
        digitalWrite(43,LOW);
        digitalWrite(44,LOW);
    }
    if (mem_hab_2 != 1){ //Baixar Persianes Habitació 2
        if(dada == 'V') {
            motor_hab_2_1.setSpeed(500);
            motor_hab_2_1.step(-voltes);

            motor_hab_2_2.setSpeed(500);
            motor_hab_2_2.step(voltes);

            mem_hab_2 = 1;
        }
        digitalWrite(48,LOW);
        digitalWrite(47,LOW);
        digitalWrite(46,LOW);
        digitalWrite(45,LOW);
        digitalWrite(41,LOW);
        digitalWrite(42,LOW);
        digitalWrite(43,LOW);
        digitalWrite(44,LOW);
    }
}

//Sensors

if (dada =='W') { //Enviar Temperatura
    int temperatura = dht.readTemperature();
    Serial.println (temperatura + String("°C"));
}

//Enviar Humitat
if (dada =='X') { //Enviar Humitat
    int humitat = dht.readHumidity();
    Serial.println (humitat + String("%"));
}

} //S'acaba el While

```

```

// ----- AUTOMATITZACIÓ DES D'ARDUINO ----- //

char eKey = keypad.getKey(); //Variable en funcio de keypadEvent

// ----- Control Gestió ldr ----- //

int ldrEstat = analogRead(ldr);

// ----- Gestió llums exterior en automàtic ----- //

if ((ldrEstat > 600) and (llums_auto == 1)) {
    digitalWrite (led14, HIGH);
    digitalWrite (led15, HIGH);
    digitalWrite (led16, HIGH);
}

if ((ldrEstat < 600) and (llums_auto == 1)) {
    digitalWrite (led14, LOW);
    digitalWrite (led15, LOW);
    digitalWrite (led16, LOW);
}

// ----- Gestió llums exterior en automàtic ----- //

if((dada == 'E') and (llums_auto == 0) and (llums_man == 1)) { //Encendre
Llums Entrada. llums_man = 1
    digitalWrite (led14, HIGH);
    digitalWrite (led15, HIGH);
    digitalWrite (led16, HIGH);
}

if((dada == 'F') and (llums_auto == 0) and (llums_man == 1)) { //Apagar
Llums Entrada. llums_man = 1
    digitalWrite (led14, LOW);
    digitalWrite (led15, LOW);
    digitalWrite (led16, LOW);
}

// ----- Alarma ----- //
int Estat_Alarma = digitalRead(PIR_Alarma);

//--> ON
if ((PIR_Alarma_ON == 1) and (PIR_Alarma_OFF == 0)) {

    if (Estat_Alarma == HIGH) {
        digitalWrite(timbre, HIGH);
        delay(50);
        digitalWrite(timbre, LOW);
        delay(50);
    }

    if (Estat_Alarma == LOW) {
        digitalWrite(timbre, LOW);
    }
}

//--> OFF

if ((PIR_Alarma_ON == 0) and (PIR_Alarma_OFF == 1)) {

    if (Estat_Alarma == HIGH) {

```

```

digitalWrite(timbre, LOW);
}

if (Estat_Alarma == LOW) {
digitalWrite(timbre, LOW);
}
}

// ----- Bany -----
int Estat_Bany= digitalRead(PIR_Bany);

if (Estat_Bany == HIGH)
{
  digitalWrite(led9, HIGH);
  digitalWrite(led10, HIGH);

}
else
{
  digitalWrite(led9, LOW);
  digitalWrite(led10, LOW);
}

// ----- Timbre -----
int boto_timbre_status = digitalRead(5);

if (boto_timbre_status == HIGH) {
digitalWrite (timbre, LOW);
}

else if (boto_timbre_status == LOW) {
digitalWrite (timbre, LOW);
}

// ----- Receptor IR -----
if (irrecv.decode(&Codis))  {
switch (Codis.value)  {

  case 0xFF9867:
    PIR_Alarma_ON = 1;
    PIR_Alarma_OFF = 0;

    break;

  case 0xFFB04F:
    PIR_Alarma_ON = 0;
    PIR_Alarma_OFF = 1;

    break;

// ---- Llums ---- //
  case 0xFF30CF: // Passadís (Button 1)

    if (digitalRead(led1) and digitalRead(led6)) {
      digitalWrite(led1, LOW);
      digitalWrite(led6, LOW);
    }
    else{
      digitalWrite(led1, HIGH);
      digitalWrite(led6, HIGH);
    }

}
}

```

```

        break;

case 0xFF18E7:                                // Menjador (Button 2)

    if (digitalRead(led2) and digitalRead(led3) and digitalRead(led4)
and digitalRead(led5)) {
        digitalWrite(led2, LOW);
        digitalWrite(led3, LOW);
        digitalWrite(led4, LOW);
        digitalWrite(led5, LOW);
    }
    else{
        digitalWrite(led2, HIGH);
        digitalWrite(led3, HIGH);
        digitalWrite(led4, HIGH);
        digitalWrite(led5, HIGH);
    }

        break;

case 0xFF7A85:                                // Habitació 1 (Button 3)

    if (digitalRead(led7) and digitalRead(led8)) {
        digitalWrite(led7, LOW);
        digitalWrite(led8, LOW);
    }
    else{
        digitalWrite(led7, HIGH);
        digitalWrite(led8, HIGH);
    }

        break;

case 0xFF10EF:                                // Bany (Button 4)

    if (digitalRead(led11) and digitalRead(led12) and
digitalRead(led13)) {
        digitalWrite(led11, LOW);
        digitalWrite(led12, LOW);
        digitalWrite(led13, LOW);
    }
    else{
        digitalWrite(led11, HIGH);
        digitalWrite(led12, HIGH);
        digitalWrite(led13, HIGH);
    }

        break;

case 0xFF38C7:                                // Habitació 2 (Button 5)

if ((llums_auto == 0) and (llums_man == 1)) {

    if (digitalRead(led14) and digitalRead(led15) and
digitalRead(led16)) {
        digitalWrite(led14, LOW);
        digitalWrite(led15, LOW);
        digitalWrite(led16, LOW);
    }
    else{
        digitalWrite(led14, HIGH);
        digitalWrite(led15, HIGH);
        digitalWrite(led16, HIGH);
    }

}

```

```

        } } break;
// ----- Motor Garatge ----- //
case 0xFF22DD: // Botó Obrir

    if (mem_garatge != 0){
        motor_garatge.setSpeed(500);
        motor_garatge.step(-4050);
        mem_garatge = 0;
        break;
    }

case 0xFFC23D: // Botó Tancar

    if (mem_garatge != 1){
        motor_garatge.setSpeed(500);
        motor_garatge.step(4050);
        mem_garatge = 1;
        break;
    }
}//Acaba (Codis.value)
delay(150);
irrecv.resume();
}//Acaba el decode

// ----- RFID -----
if ( ! RFID.PICC_IsNewCardPresent()) { //Comprova si hi ha alguna targeta nova
    return;
}
// Select one of the cards
if ( ! RFID.PICC_ReadCardSerial()) { // Selecciona una targeta
    return;
}

String valor= "";
byte letter;
for (byte i = 0; i < RFID.uid.size; i++)

{
    valor.concat(String(RFID.uid.uidByte[i] < 0x10 ? " 0" : " "));
    valor.concat(String(RFID.uid.uidByte[i], HEX));
}

valor.toUpperCase();

if (valor.substring(1) == "0E 0D B3 73") // Targetes amb accès
{
    motor_garatge.setSpeed(500);
    motor_garatge.step(4050);
    delay (5000);
    motor_garatge.step(-4050);
    delay (500);
    digitalWrite (25, LOW);
    digitalWrite (26, LOW);
    digitalWrite (27, LOW);
    digitalWrite (28, LOW);
}

if (valor.substring(1) == "10 F0 8F A6") { // Accés porta entrada exterior
    Porta_Exterior.write(90);
}

```

```
delay (5000);
Porta_Exterior.write(0);
}

} // S'acaba el LOOP

void keypadEvent(KeypadEvent eKey) {
    switch (keypad.getState()) {
        case PRESSED:
            switch (eKey) {
                case '*': comprovar1(); break;
                case '0': comprovar2(); llums_man = 1; llums_auto = 0; break;
                case 'D':
                    digitalWrite (timbre, HIGH);
                    delay (50);
                    digitalWrite (timbre, LOW);
                    delay (50);
                    digitalWrite (timbre, HIGH);
                    delay (50);
                    digitalWrite (timbre, LOW);
                    delay (50);
                    digitalWrite (timbre, HIGH);
                    delay (50);
                    digitalWrite (timbre, LOW);
                    delay (50);
                    digitalWrite (timbre, HIGH);
                    delay (50);
                    digitalWrite (timbre, LOW);
                    delay (50);
                    digitalWrite (timbre, HIGH);
                    delay (50);
                    digitalWrite (timbre, LOW);
                    delay (50);
                    digitalWrite (timbre, HIGH);
                    delay (50);
                    digitalWrite (timbre, LOW);
                    delay (50);
                    digitalWrite (timbre, HIGH);
                    delay (50);
                    digitalWrite (timbre, LOW);
                    delay (50);
                    digitalWrite (timbre, HIGH);
                    delay (50);
                    digitalWrite (timbre, LOW);
                    delay (50);

                case '#':
                    delay(1000);
                    Porta_Interior.write (0);
                    digitalWrite (led1, LOW);
                    digitalWrite (led6, LOW);

                    break;
            }
        default:
            password1.append(eKey);
            password2.append(eKey);
    }
}
```

```
void comprovar1() {
    if (password1.evaluate()) {
        Porta_Interior.write (90);
        password1.reset();
    }
} else{
    delay (500);
    Porta_Interior.write (0);
    digitalWrite (timbre, HIGH);
    delay (100);
    digitalWrite (timbre, LOW);
    delay (150);
    digitalWrite (timbre, HIGH);
    delay (150);
    digitalWrite (timbre, LOW);
    delay (50);
    digitalWrite (timbre, HIGH);
    delay (150);
    digitalWrite (timbre, LOW);
    delay (150);
    digitalWrite (timbre, HIGH);
    delay (150);
    digitalWrite (timbre, LOW);
    delay (150);
    digitalWrite (timbre, HIGH);
    delay (150);
    digitalWrite (timbre, LOW);
    delay (150);
    digitalWrite (timbre, HIGH);
    delay (150);
    digitalWrite (timbre, LOW);
    delay (150);
    digitalWrite (timbre, HIGH);
    delay (150);
    digitalWrite (timbre, LOW);
    delay (150);

    password1.reset();
}
}

void comprovar2() {
    if (password2.evaluate()) {
        digitalWrite(led14, HIGH);
        digitalWrite(led15, HIGH);
        digitalWrite(led16, HIGH);
        password2.reset();
    }
} else{
    digitalWrite(led14, LOW);
    digitalWrite(led15, LOW);
    digitalWrite(led16, LOW);
    password2.reset();
}
}
```

## Codi del Bot (Python)

```

import sys
import time
import telepot
import serial
from telepot.namedtuple import ReplyKeyboardMarkup, KeyboardButton
from telepot.namedtuple import ReplyKeyboardMarkup, KeyboardButton, ReplyKeyboardRemove
from telepot.namedtuple import InlineKeyboardMarkup, InlineKeyboardButton

ser = serial.Serial('/dev/cu.usbmodem1421', 9600) # Conexió amb el port d'arduino
bot = telepot.Bot("528094688:AAFD6ggvQY8L3R5YFYCaxWZ3hmdDqWNydA") # TOKEN del bot
print('Esperant ordres ...')

def handle(msg):
    userName = msg['from']['first_name']
    content_type, chat_type, chat_id = telepot.glance(msg)
    print('Chat Message:', content_type, chat_type, chat_id)

    if (content_type == 'text'):
        command = msg ['text']
        print ('Comando Obtenido: %s' % command)

# Iniciació
    if '/start' in command:
        bot.sendMessage (chat_id, "Hola, " + userName + "\n" + "Soc el teu bot, per
poder controlar la casa has d'escriure Menú, /start, o:" + "\n"
+ "a) Entrada" + "\n"
+ "b) Passadís" + "\n"
+ "c) Menjador" + "\n"
+ "d) Habitació 1" + "\n"
+ "e) Habitació 2" + "\n"
+ "f) Sensors")

# Buttons Menú
    if 'Menú' or 'menú' or 'Esc' or 'esc' in command:
        bot.sendMessage(chat_id, 'Menú obert, Sr. ' + userName + " 😊",
                    reply_markup=ReplyKeyboardMarkup(
                        keyboard=[

                            [KeyboardButton(text="Entrada"),
                            KeyboardButton(text="Passadís")],
                            [KeyboardButton(text="Menjador"),
                            KeyboardButton(text="Sensors")],
                            [KeyboardButton(text="Habitació 1"),
                            KeyboardButton(text="Habitació 2")],
                            []
                        ]
                    ))
# Buttons Entrada
    if 'Entrada' in command:
        bot.sendMessage(chat_id, 'Esperant...', 
                    reply_markup=ReplyKeyboardMarkup(

```

```

        keyboard=[  

            [KeyboardButton(text="Portes Entrada")],  

        KeyboardButton(text="Llums Entrada")]  

            ]  

        ))  
  

    if 'Portes Entrada' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

            reply_markup=ReplyKeyboardMarkup(  

                keyboard=[  

                    [KeyboardButton(text="Porta Entrada")],  

        KeyboardButton(text="Porta Garatge")]  

                ]  

            ))  
  

    if 'Porta Entrada' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

            reply_markup=ReplyKeyboardMarkup(  

                keyboard=[  

                    [KeyboardButton(text="Obrir Porta")],  

        KeyboardButton(text="Tancar Porta")]  

                ]  

            ))  
  

    if 'Obrir Porta' in command:  

        ser.write (b'A')  

        bot.sendMessage (chat_id, "Porta Oberta!")  
  

    if 'Tancar Porta' in command:  

        ser.write (b'B')  

        bot.sendMessage (chat_id, "Porta Tancada!")  
  

    if 'Porta Garatge' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

            reply_markup=ReplyKeyboardMarkup(  

                keyboard=[  

                    [KeyboardButton(text="Obrir Porta Garatge")],  

                    [KeyboardButton(text="Tancar Porta Garatge")]  

                ]  

            ))  
  

    if 'Obrir Porta Garatge' in command:  

        ser.write (b'C')  

        bot.sendMessage (chat_id, "Porta Oberta!")  
  

    if 'Tancar Porta Garatge' in command:  

        ser.write (b'D')  

        bot.sendMessage (chat_id, "Porta Tancada!")  
  

    if 'Llums Entrada' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  


```

```

        reply_markup=ReplyKeyboardMarkup(
            keyboard=[
                [KeyboardButton(text="Encendre Llums Entrada")],
                [KeyboardButton(text="Apagar Llums Entrada")]
            ]
        ))

if 'Encendre Llums Entrada' in command:
    ser.write (b'E')
    bot.sendMessage (chat_id, "Llums Encesos!")

if 'Apagar Llums Entrada' in command:
    ser.write (b'F')
    bot.sendMessage (chat_id, "Llums Apagats!")

# Buttons Passadís

if 'Passadís' in command:
    bot.sendMessage(chat_id, 'Esperant...', 
        reply_markup=ReplyKeyboardMarkup(
            keyboard=[
                [KeyboardButton(text="Llums Passadís")],
                [KeyboardButton(text="Persiana Passadís")]
            ]
        ))

#Llums Passadís

if 'Llums Passadís' in command:
    bot.sendMessage(chat_id, 'Esperant...', 
        reply_markup=ReplyKeyboardMarkup(
            keyboard=[
                [KeyboardButton(text="Encendre Llums Passadís")],
                [KeyboardButton(text="Apagar Llums Passadís")]
            ]
        ))

if 'Encendre Llums Passadís' in command:
    ser.write (b'G')
    bot.sendMessage (chat_id, "Llums Encesos!")

if 'Apagar Llums Passadís' in command:
    ser.write (b'H')
    bot.sendMessage (chat_id, "Llums Apagats!")

#Persiana Passadís

if 'Persiana Passadís' in command:
    bot.sendMessage(chat_id, 'Esperant...', 
        reply_markup=ReplyKeyboardMarkup(

```

```

        keyboard=[  

            [KeyboardButton(text="Pujar Persiana Passadís")],  

            [KeyboardButton(text="Baixar Persiana Passadís")]  

        ]  

    ))  
  

    if 'Pujar Persiana Passadís' in command:  

        ser.write (b'I')  

        bot.sendMessage (chat_id, "Persiana Pujada!")  
  

    if 'Baixar Persiana Passadís' in command:  

        ser.write (b'J')  

        bot.sendMessage (chat_id, "Persiana Baixada!")  
  

# Buttons Menjador  
  

    if 'Menjador' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

            reply_markup=ReplyKeyboardMarkup(  

                keyboard=[  

                    [KeyboardButton(text="Llums Menjador")],  

                    [KeyboardButton(text="Persianes Menjador")]  

                ]  

            ))  
  

#Llums Menjador  
  

    if 'Llums Menjador' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

            reply_markup=ReplyKeyboardMarkup(  

                keyboard=[  

                    [KeyboardButton(text="Encendre Llums Menjador")],  

                    [KeyboardButton(text="Apagar Llums Menjador")]  

                ]  

            ))  
  

    if 'Encendre Llums Menjador' in command:  

        ser.write (b'K')  

        bot.sendMessage (chat_id, "Llums Encesos!")  
  

    if 'Apagar Llums Menjador' in command:  

        ser.write (b'L')  

        bot.sendMessage (chat_id, "Llums Apagats!")  
  

#Persianes Menjador  
  

    if 'Persianes Menjador' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

            reply_markup=ReplyKeyboardMarkup(

```

```

        keyboard=[  

            [KeyboardButton(text="Pujar Persianes Menjador")],  

            [KeyboardButton(text="Baixar Persianes Menjador")]  

        ]  

    ))  
  

    if 'Pujar Persianes Menjador' in command:  

        ser.write (b'M')  

        bot.sendMessage (chat_id, "Persianes Pujades!")  
  

    if 'Baixar Persianes Menjador' in command:  

        ser.write (b'N')  

        bot.sendMessage (chat_id, "Persianes Baixades!")  
  

#Buttons Habitació 1  
  

    if 'Habitació 1' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

                        reply_markup=ReplyKeyboardMarkup(  

                            keyboard=[  

                                [KeyboardButton(text="Llums Habitació 1"),  

                                 KeyboardButton(text="Persianes Habitació 1")]  

                            ]  

                    ))  
  

#Llums Habitació 1  
  

    if 'Llums Habitació 1' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

                        reply_markup=ReplyKeyboardMarkup(  

                            keyboard=[  

                                [KeyboardButton(text="Encendre Llums Habitació 1")],  

                                [KeyboardButton(text="Apagar Llums Habitació 1")]  

                            ]  

                    ))  
  

    if 'Encendre Llums Habitació 1' in command:  

        ser.write (b'0')  

        bot.sendMessage (chat_id, "Llums Encesos!")  
  

    if 'Apagar Llums Habitació 1' in command:  

        ser.write (b'P')  

        bot.sendMessage (chat_id, "Llums Apagats!")  
  

#Persianes Habitació 1  
  

    if 'Persianes Habitació 1' in command:  

        bot.sendMessage(chat_id, 'Esperant...',  

                        reply_markup=ReplyKeyboardMarkup(  

                            keyboard=[  


```

```

        [KeyboardButton(text="Pujar Persianes Habitació
1")],
        [KeyboardButton(text="Baixar Persianes Habitació
1")]
    ]
))

if 'Pujar Persianes Habitació 1' in command:
    ser.write (b'Q')
    bot.sendMessage (chat_id, "Persianes Pujades!")

if 'Baixar Persianes Habitació 1' in command:
    ser.write (b'R')
    bot.sendMessage (chat_id, "Persianes Baixades!")

#Buttons Habitació 2

if 'Habitació 2' in command:
    bot.sendMessage(chat_id, 'Esperant...', 
                    reply_markup=ReplyKeyboardMarkup(
                        keyboard=[

                            [KeyboardButton(text="Llums Habitació 2"),
                             KeyboardButton(text="Persianes Habitació 2")]
                        ]
                    ))
    )

#Llums Habitació 2

if 'Llums Habitació 2' in command:
    bot.sendMessage(chat_id, 'Esperant...', 
                    reply_markup=ReplyKeyboardMarkup(
                        keyboard=[

                            [KeyboardButton(text="Encendre Llums Habitació 2")],
                            [KeyboardButton(text="Apagar Llums Habitació 2")]
                        ]
                    ))
    )

if 'Encendre Llums Habitació 2' in command:
    ser.write (b'S')
    bot.sendMessage (chat_id, "Llums Encesos!")

if 'Apagar Llums Habitació 2' in command:
    ser.write (b'T')
    bot.sendMessage (chat_id, "Llums Apagats!")

#Persianes Menjador

if 'Persianes Habitació 2' in command:
    bot.sendMessage(chat_id, 'Esperant...', 
                    reply_markup=ReplyKeyboardMarkup(

```

```

        keyboard=[  

            [KeyboardButton(text="Pujar Persianes Habitació  

2")],  

            [KeyboardButton(text="Baixar Persianes Habitació  

2")]  

        ]  

    ))  
  

if 'Pujar Persianes Habitació 2' in command:  

    ser.write (b'U')  

    bot.sendMessage (chat_id, "Persianes Pujades!")  
  

if 'Baixar Persianes Habitació 2' in command:  

    ser.write (b'V')  

    bot.sendMessage (chat_id, "Persianes Baixades!")  
  

# Buttons Sensors  

if 'Sensors' in command:  

    bot.sendMessage(chat_id, 'Esperant...',  

        reply_markup=ReplyKeyboardMarkup(  

            keyboard=[  

                [KeyboardButton(text="Temperatura")],  

                [KeyboardButton(text="Humitat")],  

                [KeyboardButton(text="Alarma")],  

                [KeyboardButton(text="Mode ECO")]  

            ]  

        ))  
  

# ----- DHT11 ----- #  
  

if 'Temperatura' in command:  

    ser.write (b'W')  

    temperatura = ser.readline()  

    bot.sendMessage (chat_id, temperatura)  
  

if 'Humitat' in command:  

    ser.write (b'X')  

    humitat = ser.readline()  

    bot.sendMessage (chat_id, humitat)  
  

# ----- PIR_Alarma ----- #  
  

if 'Alarma' in command:  

    bot.sendMessage(chat_id, 'Esperant...',  

        reply_markup=ReplyKeyboardMarkup(  

            keyboard=[  

                [KeyboardButton(text="Activar Alarma")],  

                [KeyboardButton(text="Desactivar Alarma")]  

            ]  

        ))
```

```
if 'Activar Alarma' in command:
    ser.write (b'1')
    bot.sendMessage (chat_id, "Alarma Activada!")

if 'Desactivar Alarma' in command:
    ser.write (b'2')
    bot.sendMessage (chat_id, "Alarma Desactivada")

# ----- MODE ECO, func(LDR) -----
if 'Mode ECO' in command:
    bot.sendMessage(chat_id, 'Esperant...', 
                    reply_markup=ReplyKeyboardMarkup(
                        keyboard=[ 
                            [KeyboardButton(text="Activar Mode ECO")], 
                            [KeyboardButton(text="Desactivar Mode ECO")]
                        ]
                    ))
    
if 'Activar Mode ECO' in command:
    ser.write (b'Y')
    bot.sendMessage (chat_id, "Mode ECO, Activat!")

if 'Desactivar Mode ECO' in command:
    ser.write (b'Z')
    bot.sendMessage (chat_id, "Mode ECO, Desactivat!")

bot.message_loop (handle)

while 1:
    time.sleep (20)
```