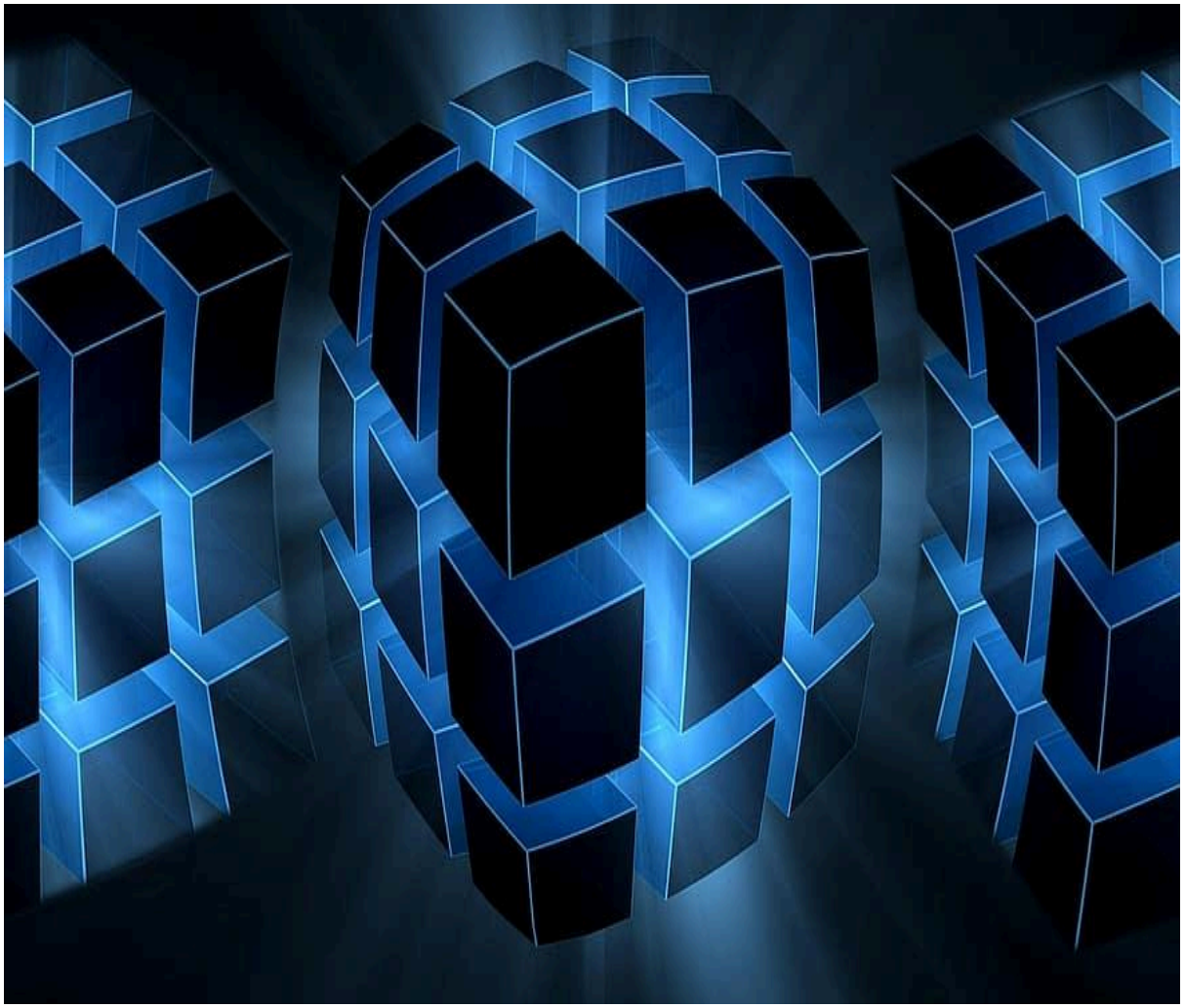
FER EL CUB DE RUBIK, AUTOMÀTIC?



Pseudònim: **GatFuster**

Grup: **101/201**

Any: **2023/2024**

Tutors: **Marc Gómez Llenas i Jaime Morcillo**

INSTITUT: **PUIG CASTELLAR**

INSTITUT
PUIG CASTELLAR

RESUMEN

Hoy en día estamos rodeados por máquinas, y cada día estas se hacen más inteligentes, esto significa que cada vez son más útiles y funcionales para los humanos. En este trabajo veremos una mezcla entre el rompecabezas más conocido del mundo y una máquina especializada para realizar la resolución de la misma.

Para poder realizar este proyecto hemos de entender las funciones de muchos aspectos de la programación, como por ejemplo, los lenguajes de programación, el entendimiento de las placas de Arduino o también las plataformas de edición de figuras en tres dimensiones. Por lo cual, este proyecto o reto personal, me hace real ilusión realizar para aprender e introducirme en el mundo de la programación y diseño en tres dimensiones.

De seguido con las hipótesis, podemos sintetizar en que la informática es una herramienta que revolucionó el mundo, y este, aún en sus inicios, tiene un potencial desmesurado, el cual cada vez se hace más evidente en nuestro día a día. La creación de máquinas inteligentes hace de cosas que imaginamos imposibles en hechos que garantizan multitud de posibilidades.

ABSTRACT

Nowadays we are surrounded by machines, and every day they are getting smarter, this means that they are becoming more useful and functional for humans. In this work we will see a mix between the most known puzzle in the world and a specialized machine to solve it.

To be able to do this project we have to understand the functions of many aspects of programming, such as programming languages, the understanding of Arduino boards or also the platforms for editing figures in three dimensions. Therefore, this project or personal challenge, makes me really excited to learn and introduce myself in the world of programming and in the three dimensions design .

Following the hypothesis, we can synthesize that the computing is a tool that revolutionized the world, and this, even in its the beginning, has an enormous potential, which is becoming increasing evidently in our daily lives. The creation of intelligent machines turns things we imagined impossible into facts that guarantee a multitude of possibilities.

INSTITUT
PUIG CASTELLAR

ÍNDEX

INSTITUT
PUIG CASTELLAR

1. INTRODUCCIÓ

L'automatització és el procés en la qual es executen activitats o processos sense intervenció humana. Consisteix en utilitzar tecnologia com un sistema, hardware o sensors entre altres, operar màquines o sistemes de manera autònoma.
L'automatització s'utilitza en amplis camps del món, en les indústries, en l'agricultura, en logística, en l'atenció mèdica i molts altres. Pot anar des de sistemes simples fins a sistemes sofisticats i complexos.

El meu tema consisteix a crear i automatitzar una màquina que fa el cub de Rubik, el meu objectiu és entendre i analitzar un programa en la qual es pugui resoldre automàticament el cub de Rubik.
Vull aprendre a crear un programa amb un llenguatge de programació, també fabricarè una màquina intel·ligent que pugui entendre el programa i resoldre el cub de Rubik, amb tot el material tecnològic necessari.

La meva motivació prové del meu futur treball en la indústria d'informàtica, em vull dedicar a dirigir programes/ webs/ jocs/ sistemes/ etc. Des de casa meva. Per obtenir un bon inici en aquest món. Sempre m'ha agradat molt la tecnologia i sobretot els mecanismes i sistemes informàtics.

L'objectiu del meu treball es pot considerar en aprendre i introduir-me en el món de la programació i dels circuits elèctrics, amb aquest fi podré tenir una major compressió de futures tasques i treballs.

En primer lloc, faré un sistema de cables, i ja acabada la primera part, adaptaré el sistema a la màquina que resol el cub de Rubik que serà realitzat al departament de tecnologia, i finalment connectar el programa al sistema de cables a la màquina i que aquesta realitzi el cub de Rubik.

1.1 PER QUÈ HE ESCOLLIT AQUEST TEMA?

He escollit aquest tema perquè ja fa temps que volia aprendre a fer un cub de Rubik, i vaig pensar, perquè no connectar-lo amb la tecnologia i aprendre també en el camp de la programació.

També com a excusa per dirigir-me a la comunitat dels trencaclosques, sempre vaig pensar que és una comunitat molt tranquila i que es pot aprendre matemàtiques i geometria de forma divertida i curiosa.

A més d'aprendre un món, també aprenc un nou sistema de construcció amb les màquines 3D i màquines en general de tall i creació d'objectes.

2. Què és la programació?

La programació s'ha incorporat a la nostra vida actual i anirà creixent en la seva utilització en un futur molt pròxim. Tots els aparells elèctrics que ens envolta té un programa, fent que la programació tingui un valor inigualable per la comoditat humana. Però què és exactament la programació?

La programació és el procés per crear i dissenyar instruccions específiques perquè una base d'informació pugui fer una tasca. És un complement essencial per la informàtica i s'utilitza com la base de la informàtica, en el software, aplicacions, pàgines web, sistemes d'automatitzacions, sistemes industrials, etc.

La informàtica implica escriure un conjunt de codis fent servir un llenguatge de programació en la qual l'ordinador ha d'entendre i executar les instruccions escrites. El codi pot ser molt complex o simple, segons el teu nivell de programació i també segons la tasca que vols realitzar.

Els llenguatges més comuns en la programació són Python, Java, C++, JavaScript, PHP, SQL. Aquests llenguatges permeten comunicar-se amb la màquina a partir de certes regles i sintaxi específica per cada llenguatge.

La programació és un punt essencial pel món digital, ja que tots els camps d'estudi depenen dels sistemes de software i aplicacions informàtiques per realitzar i facilitar processos; També el funcionament del món industrial depenen d'un complex sistema de la programació per automatitzar processos.
La programació és un camp d'estudi inacabat, té molt que millorar i innovar. Un camp en constant evolució. És el camp clau per seguir un progrés innovador, eficient i còmode pel futur de la humanitat.

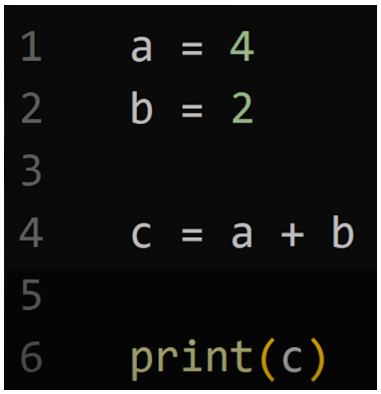
2.1 Tipus de llenguatge de programació

2.1.1 LLenguatge de propòsit general

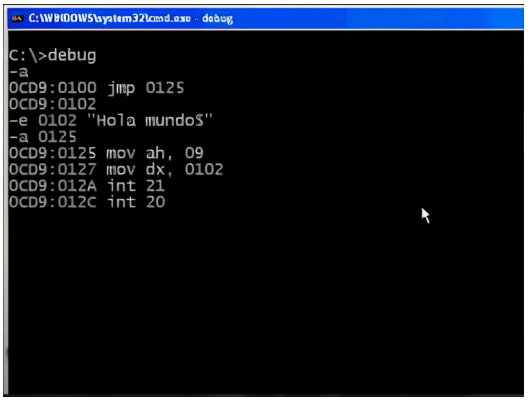
És un llenguatge que ho podem utilitzar per a qualsevol activitat, un llenguatge per crear jocs, pàgines web, aplicacions, intel·ligència artificial, automatització d'scrips i més, mentre que altres llenguatges estan especialitzats per x funcions.

2.1.2 Llenguatge d'alt nivell

Un llenguatge d'alt nivell és un llenguatge que és semblant al nostre llenguatge natural o forma d'escriure o comunicar, mentre més fàcil sigui el llenguatge més difícil serà per a nosaltres entendre el llenguatge.



Imatge 1 : Exemple de llenguatge d'alt nivell.
Extreta de: Font pròpia



Imatge 2 : Exemple de llenguatge de baix nivell.
Extreta de: Font pròpia

2.2 Llenguatges de programació

2.2.1 Python.

Com anteriorment havia comentat, existeixen desenes de llenguatges de programació com Java, JavaScript, C++, C#, Swift, PHP i Python són alguns dels exemples, cada una té les seves avantatges i desavantatges.

Com per exemple Java té un sistema multiplataforma i té inclòs un sistema de seguretat incorporat, però Java és massa complex per un principiant, la mitjana per aprendre adequadament Java són d'aproximat nou mesos.

Un altre exemple és C++, un llenguatge que amb només de 2 a 6 mesos es pot utilitzar amb força nivell, és un codi que es pot reutilitzar per fer nous programes, té una manera senzilla de programar qualsevol cosa nova, però és un llenguatge molt antic i no té gaires usuaris que arreglin els errors i tampoc té una llibreria senzilla.

De totes les opcions he escollit a Python, la meva raó principal és perquè és un llenguatge complex així que és més fàcil d'entendre i aprendre i també té un sistema especialment adaptat per la programació d'automatització, i és l'indicat pel que busco, però també té molts altres avantatges, aquestes 4 característiques són les més importants per a mi.

2.2.2 Avantatges de Python

1. Fàcil d'aprendre.
   A pesar de ser un llenguatge molt complex, és el llenguatge amb l'aprenentatge més ràpida, ja que com hem dit anteriorment és un llenguatge d'alt nivell, per tant té una gran similitud amb el nostre llenguatge natural.
   I també té una inmensa comunitat la qual quan tinguis qualsevol problema, algú ja li haurà passat, llavors qualsevol dubte que tinguis hi ha una probabilitat molt alta que ja estigui la solució al problema en línea.

2. Llenguatge de **TIPAT DINÀMIC**.
   Existeixen 2 tipus de **Tipat**, dinàmic i estàtic.
   Els de tipus dinàmic com Python, al moment d'escriure Python defineix directament la dada, per tant, diferencia les lletres i números entre altres variables. La variable s'adapta al tipus de dada.

```
1   nombre = "roberto"
2   edad = 21
```

Imatge 3 : Exemple de mètode de definició: Python defineix directament

Extreta de: Font pròpia

Els de tipus estàtic, com en un llenguatge com Java hi tenim que expressar primer quin tipus de variable és i, per tant, no ho defineix automàticament. La dada ha d'ajustar-se a la dada donada.

```
1   String nombre = "roberto";
2   int edad = 21;
```

Imatge 4 : Exemple de mètode de definició: Java expresant variables per definir

Extreta de: Font pròpia

3. Llenguatge orientat a objectes.

   És un estil de programació la qual escrivim el codi i desatrotllem pensant en les parts del programa com si fossin de l'objecte.

4. Llenguatge interpretat.

   Python té un sistema interpretat, significa que Python llegeix línia per línia les dades que anem escrivint i ho executa línia per línia. Els sistemes interpretats tenen una major facilitat per la correcció d'errors per aquesta característica.

   Per contrari està el sistema copulatiu, significa que recopila tot el coda i executa tot a la vegada. L'avantatge d'aquest sistema és que és més ràpid a l'executar.

3.Què és un cub de Rubik?

El cub de Rubik és el trencaclosques més popular del món, la seva expansió va ser instantània, la seva creació a Hongria el 1974 va popularitzar l'article amb només 1 any a tota Europa, més tard amb la globalització es va fer popular mundialment, atorgant-li el primer lloc en el rànquing mundial de trencaclosques.

Avui en dia hi ha tornejos, reptes i variants del Cub de Rubik, atorgant-li un lloc al World Guinness Records al més ràpid amb 3,134 segons que ho té Max Park. Aquest record hi varia moltes vegades per any, gent competeix per només mil·lèsimes de segons, però igualment volen aquest títol mundial.

Hi existeixen diversos tipos de cubs de Rubik, el mini cub, cub de gel o cub de butxaca (2x2x2), el cub de Rubik estàndard (3x3), més complexos com la venjança de Rubik (4x4x4), cub mestre (5x5)... fins a un cub de Rubik de 15x15x15. Per nomenar-los més informalment diem cub de Rubik de x per x.

Per fer el meu treball em centro en el bàsic, el cub de 3x3x3.

Cub de Rubik o també conegut com a Cub màgic, és un trencaclosques mecànic tridimensional creat el 1974 per Ernő Rubik.

3.1. Origen del cub de Rubik

El desenvolupament del cub de Rubik té origen en la base de la geometria, Ernő Rubik interessat en aquest àmbit i en la formes tridimensionals, aprofitant que en aquells anys era director d'una publicitat hongara, la qual patrocinava un programa de jocs d'enginy.

El mateix creador va tardar més d'un mes en resoldre la seva pròpia creació.

3.1.1 Ernő Rubik

Ernő Rubik és un arquitecte, escultor i dissenyador Hangar, a més del cub de Rubik amb el seu nom, té un altre trencaclosques amb el seu nom, el Rubik's Clock o Rellotge de Rubik.

Ernő Rubik va néixer el 13 de juliol de 1944 a la ciutat de Budapest en Hongria, fill d'un enginyer aeronàutic, qui Ernő Rubik admirava des de petit, del que va aprendre i va ovacionar: " Vaig aprendre molt sobre el treball en el sentit que és un procés de creació que té un objectiu i un resultat positiu, també tant en sentit figurat com a literal, mon pare va ser una persona capaç de moure una muntanya!"

Ernő Rubik va estudiar arquitectura a l'academia d'Arts Aplicades i Disseny de la capital Magiar i també va estudiar Escultura a la Universitat Tècnica de Budapest. Més tard, en un futur va ingressar com a professor d'arquitectura a la Universitat Tècnica de Budapest, en la qual utilitzava un cub format per 27 peces per explicar conceptes de l'espai i la geometria tridimensional, sí aquest és el cub màgic. Com a Cub màgic, és un trencaclosques mecànic tridimensional creat el 1974 per Ernő Rubik.

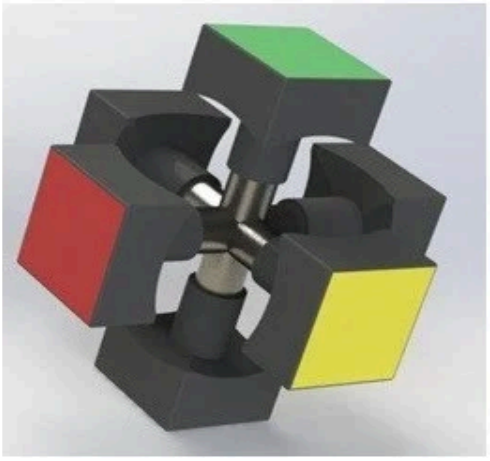
3.2. Funció d’un cub de Rubik

Un cub de Rubik 3x3x3 està format per 6 cares de diferents colors, els colors son blau, vermell, groc, blanc, verd i taronja, com norma general el blanc és la cara principal, però aquesta qüestió és indiferent.

El cub de Rubik està format 26 peces de cubs de colors petits, cada peça té un enllaç intern que permet que cada cub estigui entrellaçat amb els altres peces de cub i així és com aconsegueix girar i moure's a diferents posicions. Però les peces centrals de cada cara estan fixes al mecanisme intern del cub de Rubik, així podem

concloure que hi ha 6 peces fixes de diferents colors que pertanyen a cada cara i 20 peces mòbils qual són els externs.

Si fem la validació que les 6 cares i el mecanisme intern giratori són 1 peça, podríem dir que hi ha un total de 21 peces: 1 peça central que consisteix en 3 eixos giratoris enganxats a les 6 cares, i 20 peces de cubs que encaixen amb la peça central i formar el trencaclosques.



Imatge 5 : Mecanisme dins d'un cub de Rubik:

Imatge 6 :Mecanisme dins d'un cub de Rubik:

Aquest funciona per un mecanisme d'eixos enganxat als 6 centres de cada color, si ens fixem podem girar els centres, però podem observar que sempre estan al mateix lloc, ja que el groc sempre estarà al costat del verd i així entre tots, aquest mecanisme permet girar independentment cada eix podent així fer mescla de tots els colors.

Existeixen 43.252.003.274.489.856.000 possibles combinacions, però només una és la solució!

Imatge 7: Aresta i cantonada del cub de rubik

3.4. Objectiu del cub de Rubik.

L'objectiu del cub de Rubik és simple, tornar a la seva forma original, que cada cara tingui el seu color. Com he comentat anteriorment conté més de 43.000 bilions de combinacions de colors, però només té una solució correcta.

També existeixen objectius com el met, crear apps, programes, robots, cubs, cubs diferents, cubs enginyosos, manuals, vídeos, recerques... sobre el cub de Rubik, trobar una manera diferent i divertida de veure aquest món.

Un altre objectiu pels jugadors del trencaclosques és trencar un record, sigui personal, en tornejos o el record mundial. Gent de tot el món estan diàriament practicant amb el cub, pels centenars de campionats al voltant de tots llocs. A Espanya trobem **“L'Associació Española de speedcubing”**, entre altres més petits com a Catalunya, la companyia de **“Club de rubik de Catalunya”** o per exemple a les **Festes majors de l'Hospitalet 2023** van realitzar un campionat de **Rubik Martorell Open 2023**.

Milions de persones volen exhibir el seu talent amb el trencaclosques més famós mundialment conegut, volen recordar el seu nom en aquesta comunitat de cubs de Rubiks.

3.4.1 Campionat mundial del cub de Rubik.

Parlem de la **WORLD CUBE ASSOCIATION**, són el grup més conegut sobre competicions i reunions comunicatives de trencaclosques mecànics, principalment i sobresortint la secció dels cubs de Rubiks. Ells mateixos decideixen quines seccions entren a la World Cube Association oficialment.

Aquesta organització és totalment lliure d'impostos, significa que tot l'aconsegueixen per fer els seus esdeveniments en l'àmbit mundial és a partir de donacions.

L'objectiu d'aquesta organització és atraure joves per gaudir d'una experiència única en el món dels trencaclosques, guiar-los i ajudar-los en l'aprenentatge en activitats diàries de la World Cube Association. En aquest link trobareu més informació sobre els Objectius: Objectius WCA

Des del 2005 fins ara han tingut més de 100.000 participants de tot el món, han participat en més de 140 regions, i no pensen parar. La **WCA** organitzen els majors esdeveniments del món dels trencaclosques mecànics, milers de persones acudeixen els seus actes per gaudir d'una comunitat competitiva del Cub de Rubik.

3.4. Com es resol un cub de Rubik?

El mateix creador del Cub de Rubik va trigar més d'un mes en completar la seva pròpia creació, a primera vista sembla un trencaclosques simple, però al darrere porta un gran procés matemàtic i un procés complex.

Hi ha diverses formes de resoldre el cub de Rubik, clar, hi ha maneres més senzilles que altres, els principiants acostumen a utilitzar la tàctica de la creu o també coneguda com el mètode de Fridrich:

1. Creu primera capa.
   Aconseguir una creu en alguna de les cares, aquesta creu ha d'estar concorde els colors procedents. Per exemple l'aresta blanca amb blau ha de prosseguir amb el centre blau, i així amb tots els colors.

2. Cantonades primera capa.
   Hem d'obtenir col·locar cada cantonada de la primera capa amb el seu color indicat. Cada cantonada té 3 colors i cada color ha d'anar amb la seva cara. Per fer aquest pas hem d'utilitzar la tercera capa del cub de Rubik (la part inferior a la nostra cara principal), ja que en aquella capa no ens afectarà en canviar les peces.

3. Arestes segona capa.

Un cop tenim totes les cantonades. Hem d'assolir la segona capa. En aquest pas és important trobar en capa baixa arestes que tinguin dos colors i que en aquells colors no es trobi el color oposat a la teva cara principal (Per exemple si la teva cara principal és la blanca, les arestes de la tercera capa no ha de ser groga). Un cop tenim l'aresta, hem de col·locar-lo prosseguit el seu centre, un cop fet això, hem de fer el nostre primer algorisme.
Amb aquests algorismes col·locarem les arestes en el seu lloc, situada en la segona capa.

Aquests són els passos de l'algorisme.

- Si tenim la cara inferior al color de la dreta. D L D' L' D' F' D F
- Si tenim la cara inferior al color de l'esquerra. D' R' D R D F D' F'

4. Creu tercera capa.
   En aquest pas ja hem d'haver aconseguit tota la primera i segona capa amb els seus colors.
   En primer lloc, girarem el nostre cub de Rubik, col·locant el centre groc com cara principal, per fer la creu utilitzarem un nou algorisme i si tens sort només hauràs de fer-lo una vegada, però molts cops haurem de fer aquest logaritme més d'un cop.

   Hem d'utilitzar el següent algorisme:
   F R U R' U' F' (Repetir fins a aconseguir creu)

   Tenim 4 situacions diferents per trobar la creu, és una seqüència.
   - Només el centre groc. (fer l'algorisme 3 cops)
   - Centre groc i dues arestes en forma de L. (fer l'algorisme 2 cops)
   - Línia continua pel centre horitzontalment. (fer l'algorisme 1 cop)
   - Creu. (no has de fer cap vegada l'algorisme)

5. Coincidència en les arestes laterals amb el seu centre.
   Primer de tot hem d'observar si tenim alguna coincidència d'alguna aresta amb el seu centre lateral, i si aquesta coincidència és correcte hem de trobar

que a la cara dreta també estigui la coincidència. Si no tenim la sort d'aquesta casualitat haurem de fer servir un altre algorisme. Aquest algorisme també s'utilitza una vegada hàgim aconseguit les dues coincidències contínues, per aquest pas hem de centrar una cara amb coincidència i fer el següent algorisme:

- R' U' R U' R' 2U R U'

6. Col·locar les cantonades a la seva posició.

   Un cop tenim totes les coincidències, hem de veure que els colors de les cantonades estiguin al seu lloc (dona igual l'ordre dels colors, per exemple el blau pot estar a la cara del groc). Pot existir la casualitat de ja tenir-los al seu lloc, però si no hem de fixar-nos en la cantonada ben col·locada i fer el següent algorisme:

   - U R U' L' U R' U' L (En aquest pas s'aplicarà el mateix algorisme en el mateix lloc fins que totes les cantonades estiguin al seu lloc.)

7. Orientar les cantonades restants.

   Per últim, només queda orienta cada cantonada restant, ja que els tenim ben col·locades tan sols fa falta utilitzar un algorisme per canviar-les d'orientació.
   En aquest algorisme pot confondre una mica, però mai hem de canviar de cara principal en la qual farem l'algorisme.
   Utilitzarem el següent algorisme:
   R' D' R D (Repetir fins a orientar bé la cantonada)

4. PART PRÀCTICA

4.1 Definicions.

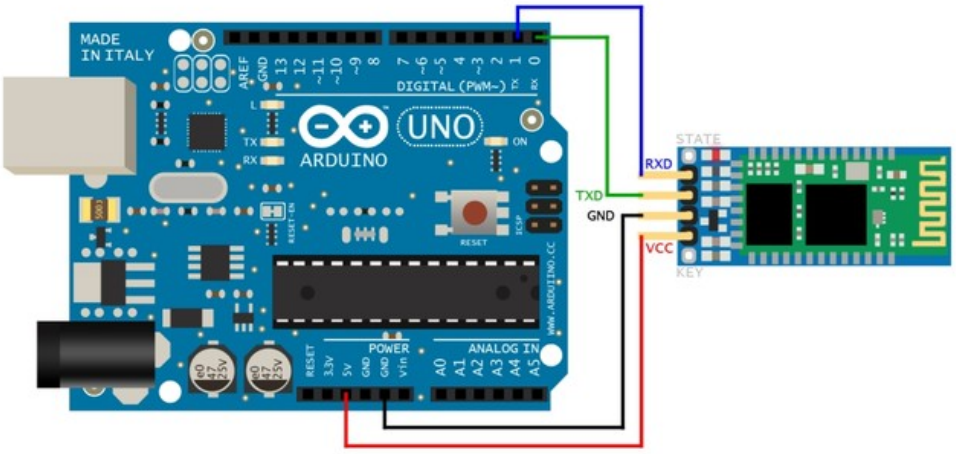
-Arduino:

Arduino és denominat com una de les marques més utilitzades per les seves plaques bases, i la més utilitzada per la robòtica. El que ho fa especial a diferència dels altres és que pots modificar la placa i crear una derivació per la teva concreta pràctica.

La popularitat d'Arduino es plasma en la seva senzillesa al utilitzar-ho, és molt potent i sobre tot molt econòmic.

Arduino és una plataforma de creació d'electrònica amb codi obert, per aquesta raó, és una placa fàcilment moldeable per qualsevol creador. El software d'Arduino és de tipus lliure i això deixa a qualsevol persona crear el seu propi programa i modificar-los.

El funcionament es basa en un microcontrolador ATMEL, que és un circuit integrat on pots gravar instruccions, els quals ho denominen amb un llenguatge en l'entorn d'Arduino IDE, que permet l'interacció amb els circuits de la placa.



Imatge 8: Placa Arduino, extreta de Pixbay

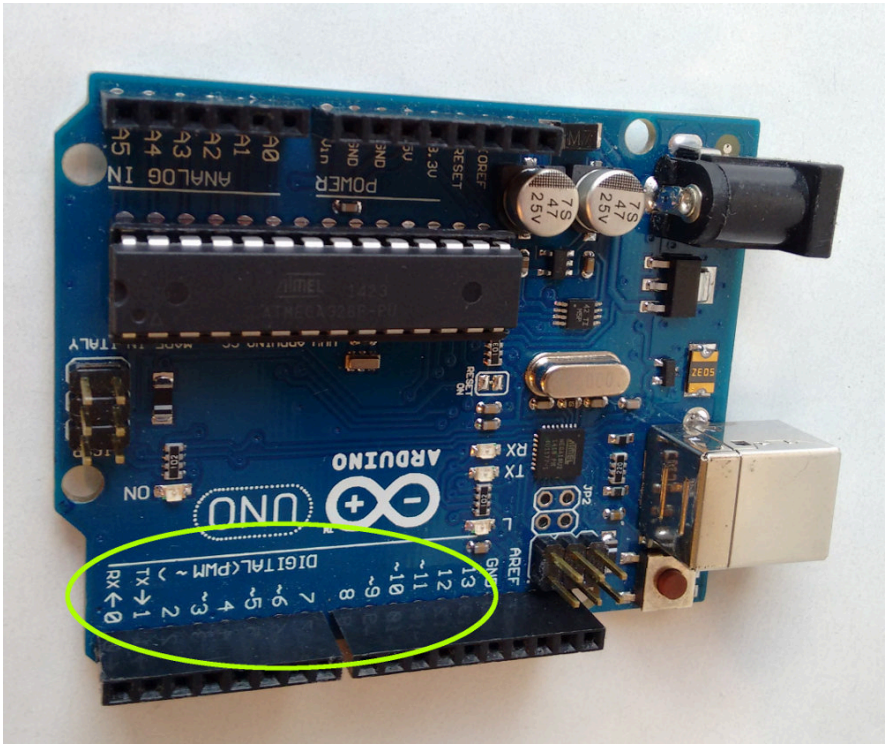
Les plaques Arduino contenen una sèrie de connexions d'entrada i sortida on connectem els nostres circuits, sensors o altres tipus de controladors.

- **Entrada:** Connexió de la placa on **recopila** les dades donades d'altres dispositius o component elèctric connectada a ella.
- **Sortida:** Connexió de la placa on **envia** les dades donades d'altres dispositius o component elèctric conectada a ella.

Les entrades i sortides poden ser de diversos tipus:

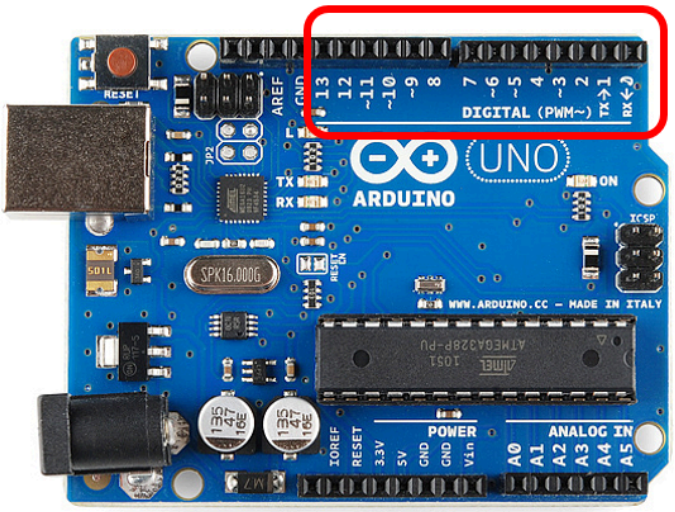
- **Entrades Analógiques:** Medeixen el voltatge de una señal d'entrada, aquesta entrades serveixen per concretar accions en concretes situacions, per exemple, qualsevol cotxe que encén les llums automàticament quan entren en un túnel.

  Amb aquestes entrades, per exemple, podem indicar el programa que faci alguna cosa només quan sigui de nit.



Imatge 9: Entradas analógiques de placa Arduino, extreta de dc722jrlp2zu8.cloudfront.net .

- **Entrades i sortides digitals:** Arduino pot saber en una entrada digital quan s'ha accionat un pulsador, mentre que la sortida digital pot accionar per ell mateix una alarma o encendre i apagar una LED.



Imatge 10: Entrades i sortides digitals de placa Arduino, extreta de Pines-Digitales-de-Arduino.png

-ARDUINO IDE:

L'entorn de desarrollatge integrat (IDE) d'Arduino és el llenguatge de programació de les plaques Arduino, és un llenguatge derivat de JAVA, aquest s'utilitza principalment en las placas de la mateixa marca, però també són compatibles amb altres plaques de tercers.

Aquestes plaques ofereixen l'Arduino IDE (Entorn de Desenvolupament Integrat), on és un entorn de programació per donar la creació d'aplicacions per les plaques Arduino, d'aquesta manera es poden fer tot tipus d'invents.

El IDE d'Arduino utilitza el programa AVRdude per convertir el codi executable en un fitxer de text en codificació hexadecimal que es carrega a la placa Arduino mitjançant un programa de càrrega en el firmware de la placa.

AVRdude: És un programa de línies de comandaments i es necessita arguments correctes per programar en Arduino

El llenguatge d'Arduino també conté la seva pròpia biblioteca de software o també coneguda com Llibreries de llenguatges, on se'n magatzemen codis escrits i verificats anteriorment pels futurs programadors agafin una referència per guiar i ajudar en els seus projectes.

-La impressora 3D

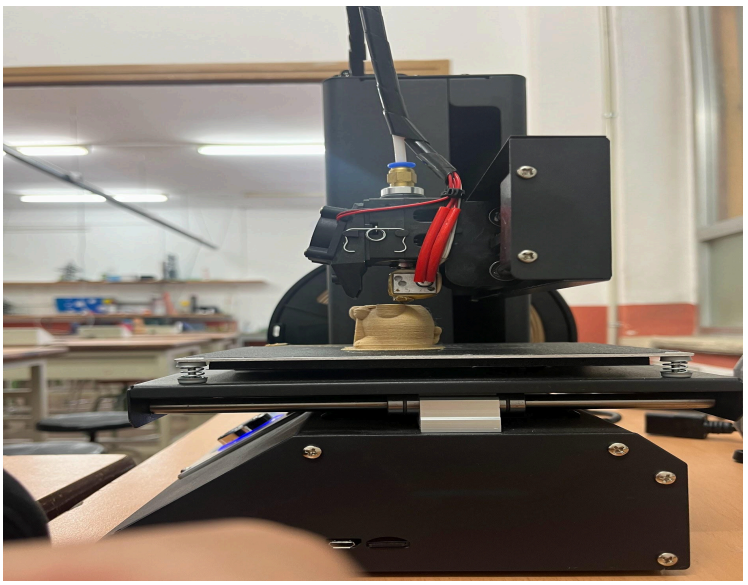
Una impressora 3D és una màquina capaç de crear figures de volum en 3 dimensions (Alçada, amplitud i llargada), partint prèviament d'un disseny fet pel programa CAD, que són els disenys assistits per computadora.

La finalitat d'aquesta màquina és convertir arxius digitals en projectes reals. Les impressores 3D han guanyat molta popularitat en tots els sectors de construcció, per la seva facilitat i és una manera molt econòmica en aquest àmbit. Moltes constructores ja estan utilitzen impressores 3D d'una mesura gegant per la creació d'edificis.

Imatge 11: Impressora 3D per construir estructures. Extreta de Impresora-3D-para-casas.jpg

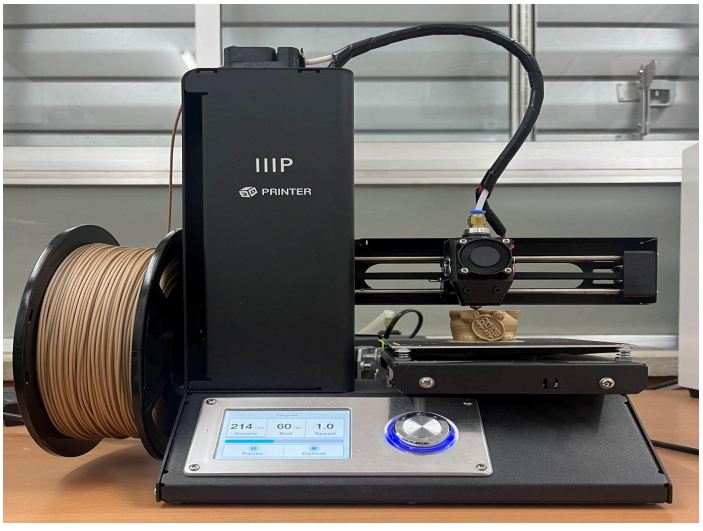
Per crear tot l'esquelet i el suport de la nostre màquina utilitzarem l'impressora 3D del departament de tecnología del meu centre IES Puig Castellar.

La Impressora és la Monoprice 121711 Select Mini 3D Printer V2 de la marca IIIP PRINTER i utilitzarem (x material) que és suficientment resistent.



Imatge 12: Impressora 3D del Puig Castellar. Font pròpia

Imatge 13: Impressora 3D del Puig Castellar. Font pròpia



Imatge 14: Impressora 3D del Puig Castellar. Font pròpia

Imatge 15: Impressora 3D del Puig Castellar. Font pròpia

-TinkerCad

TinkerCad és el programa web per excel·lència per la introducció al disseny 3D y programar circuits elèctrics. El seu ús gratuït i senzill permet l'ús d'aprenentatge pels alumnes.

TinkerCad ho vam utilitzar durant l'ESO al centre Puig Castellar a les hores d'informàtica i tecnologia, això em va permetre adquirir els suficients coneixements a l'hora d'analitzar i dissenyar o moldejar el que faci falta pel Treball de Recerca i altres activitats extraescolars.

El TinkerCad va sorgir el 2011, una fundació interna de Google ho va dissenyar. Gràcies el seu accés gratuït va ser molt popularitzat en la industria del disseny, TinkerCad es va centrar en la funció d'aprenentatge general d'aquest món, on té, per defecte, un sistema d'orientació i guionatge pels inexperts. Tant així que molts dissenyadors experts utilitzen TinkerCad per facilitar diverses accions, també s'utilitza per estalviar molts processos de renderització on tenen altres aplicacions més complexes.

Les formes es poden importar en tres formats: **STL i OBJ** per a 3D i formes **SVG** bidimensionals per extreure en formes 3D. Tinkercad exporta models en formats **STL o OBJ**, preparats per a la impressió 3D.

Tinkercad també inclou una funció per exportar models 3D a **Minecraft Java Edition**, i també ofereix la possibilitat de dissenyar estructures amb **maons Lego.**

També té la secció de circuits a TinkerCad, on és un simulador de circuits elèctrics molt pràctics, basat en navegador que admet microcontroladors Arduino Uno, plaques Micro:bit o xips ATtiny. On el codi es pot crear mitjançant el mètode de blocs **(CodeBlocks gràfics).**
El programa ofereix circuits preconstruïts anomenats **"Arrencadors"** o circuits que es poden construir utilitzant components separats.

-Parc tecnològic de Barcelona

El parc tecnològic de Barcelona situat a Nou Barris, exactament al Carrer Marie Curie 8-14, és una institució especialment indicada a ajudar a empreses tecnològiques. Ofereix serveis avançats de suport a la innovació, consolidació i creixement de les empreses, especialment les tecnològiques i del sector de les enginyeries.

L'ateneu de fabricació de Barcelona en el 9 Barris, a la mateixa instal·lació, situat a la Barcelona activa. Aquesta proporciona un espai d'aprenentatge en fabricació digital per a la innovació social i pedagògica, amb l'emprenedoria com a temàtica transversal.
Els objectius de l'Ateneu se centren a crear ponts entre la fabricació digital i la comunitat educativa, des dels infants i joves fins al teixit comunitari del barri; també es vol potenciar l'Ateneu com a punt d'innovació tecnològica i educativa. Cal que sigui un espai per avançar cap a la transformació social a través de les noves tecnologies com a eines per solucionar reptes i problemàtiques quotidianes o comunitàries.

Aquesta fundació ofereix ajuda i suport als alumnes i a qualsevol persona que vulgui aprendre de tecnologies, des d'entitats individuals (com el meu cas) fins a projectes grupals procedents de centres educatius.

Qualsevol estudiant o alumne pot intervenir i participar en les activitats que ofereixen, les activitats són molt variades, des de la reparació d'una bicicleta fins classes d'edició de vídeos o classes de Ròmbic, aquestes activitats es programen amb temps i pots apuntar-te quan vulguis, a partir de trucades o de la mateixa web.

L'Ateneu ens ofereix totes aquestes avantatges i coneixements gratuïtament, però, en canvi, ens demanen intentar fer una contraprestació, no econòmica, retornar alguna cosa, en canvi, com material o fer-nos voluntaris per preparar activitats socials, i si pot ser en tenir impacte comunitari, com tallers, projectes, ajuda als altres, qualsevol cosa que pugui repercutir en l'àmbit ciutadà
En els següents enllaços es troba una millor explicació de l'Ateneu tecnològic. També un vídeo explicat per la Clara Borras Coll, la Space Manager de l'Ateneu tecnològic.
WEB:
https://ajuntament.barcelona.cat/ateneusdefabricacio/es/ateneo-de-fabricacion-del-parque-tecnologico/
Vídeo: https://www.youtube.com/watch?v=pt7Z3HOvvek&t=133s

En el punt 4.3.2 trobem una explicació més detallada de la meva experiència personal a l'Ateneu.

4.2 Codificació

Utilitzem Arduino IDE per la programació del sistema. En aquest cas està escrita en **C++** per controlar el mecanisme, com he dit abans Arduino IDE es derivat de JAVA, però al mateix temps és compatible amb C i C++.

El programa conté un total de 731 línies, és escrita per Nikodem Bartnik, s'utilitza la pantalla LCD per mostrar-nos els processos i uns motors connectats a pins de la placa d'Arduino per realitzar moviments específics.

4.2.1 Punts claus del code

1. **Llibreria liquidcrystal1 i definicions de variables:**

```
#include <LiquidCrystal.h>

int step_number = 0;
int step_number_reverse = 7;

typedef struct moteur {
    int IN1;
    int IN2;
    int IN3;
    int IN4;
};

unsigned long myTime;
unsigned long Timeref;
const int rs = 2, en = 3, d4 = 4, d5 = 5, d6 = 6, d7 = 7;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

Imatge 16: Codi definicions i inclusió de llibreria. Font pròpia.

- S'inclouen la llibreria *LiquidCrystal 1* per treballar amb la pantalla LCD, on ens mostra els passos.
- Es definen les variables per els pins de la pantalla LCD i també l'estructura pels motors.

2. **Configuració inicial: (`setup()`):**

En aquesta part del codi, es fa la configuració inicial del programa. S'estableix el mode dels pins del 22 al 52 com a sortides, s'inicia la comunicació serial a una velocitat de 9600 bauds ***"(Serial.begin(9600))"***, i s'inicialitza l'objecte LCD per a l'ús

de la pantalla LCD amb una configuració de 16 columnes i 2 files ***"(lcd.begin(16, 2))"*.**

```
void setup() {
    for (int i = 22; i < 53; i++) {
        pinMode(i, OUTPUT);
    }
    Serial.begin(9600);
    lcd.begin(16, 2);
}
```

Imatge 17: Codi. Font pròpia.

- Es configura els pins desde la 22 fins la 52 com a sortides.
- S'inicia la comunicació serial a 9600 bauds. (El baud és una mesura per saber quantes vegades canvia d'estat un señal de comunicació)
- S'encén la pantalla LCD de 16x2.

3. **Bucle principal (`loop()`):**

Aquest bucle comença amb una condició que verifica si els pins 2 i 3 estan actius (en alt). Si és així, reinicieu la pantalla LCD a una configuració de 16 columnes i 2 files. Després, imprimiu el missatge *"Rubik's Cube"* a la primera fila de la pantalla LCD.

Després, llegiu dades del port serial. Si el que es llegeix és ***'1'***, ***'2'*** *o* ***'3'***, s'executa un bloc de codi que assigna certes seqüències de moviments a la variable ***"Myresult"*** i després truca a la funció ***"DoRotation()"*** amb aquesta seqüència. Finalment, mostra *"Done"* a la segona fila de la pantalla LCD.

```
void loop() {
    if (digitalRead(2) != 0 && digitalRead(3) != 0) {
        lcd.begin(16, 2);
    }
    lcd.setCursor(0, 0);
    lcd.print("  Rubik's Cube  ");

    if (Serial.read() == '1') {
        char* Myresult = "BFudlbrlFbludlfrlFudblRud";
        Serial.println("\n=========================================");
        DoRotation(Myresult);
        lcd.setCursor(7, 1);
        lcd.print("Done     ");
    }

    if (Serial.read() == '2') {
        char* Myresult = "LBBuBBlDLLFBBLLBRFFLLUDBBLLud";
        Serial.println("\n=========================================");
        DoRotation(Myresult);
        lcd.setCursor(7, 1);
        lcd.print("Done     ");
    }

    if (Serial.read() == '3') {
        char* Myresult = "LrDuLrDuLrDuLrDuLrDuLrDu";
        Serial.println("\n=========================================");
        DoRotation(Myresult);
        lcd.setCursor(7, 1);
        lcd.print("Done     ");
    }
```

Imatge 18: Codi. Font pròpia.

- Mostra "Rubik's Cube" a la LCD.
- Si rep certs caràcters (***'1', '2', '3'***) a través del port serial, fes seqüències de moviments definides i mostra *"Done"* a la LCD.

4. Funcions personalitzades:

Aquestes funcions realitzen diferents tasques dins del programa.

"DoRotation" maneja la seqüència de rotacions, ***"Rotation"*** executa els moviments en els motors, ***"OneRotation"*** i ***"TwoRotation"*** mouen els motors en una o dues direccions respectivament, ***"size"*** calcula la mida d'una cadena de caràcters i ***"disablemotor"*** atura el moviment d'un motor.

```
void DoRotation(char* str) {
    int len = size(str);
    Timeref = millis();
    for (int i = 0; i < len; i++) {
        myTime = millis();
        lcd.setCursor(0, 1);
        lcd.print((myTime - Timeref) / 1000);
        lcd.print("s    ");
        lcd.setCursor(7, 1);
        lcd.print(i + 1);
        lcd.print(" / ");
        lcd.print(len);
        if (i < len - 1) {
            if (Rotation(str[i], str[i + 1]) == 1) {
                i += 1;
            }
        } else {
            Rotation(str[i], 'p');
        }
        delay(2);
    }
}

int Rotation(char c, char r) {
    int res = 0;
    // ... (Codi del switch statement i la seva lògica)
    return res;
}
```

Imatge 19: Codi. Font pròpia.

```
void OneRotation(struct moteur Moteur, int rot) {
    // ... (Codi per a un sol moviment de motor)
}

void TwoRotation(struct moteur Moteur, struct moteur Reverse, int rot, int rotR) {
    // ... (Codi per a dos moviments de motor)
}

int size(char* arr) {
    int res = 0;
    while (arr[res] != '\0') {
        res++;
    }
    return res;
}

void disablemotor(struct moteur Moteur) {
    digitalWrite(Moteur.IN1, LOW);
    digitalWrite(Moteur.IN2, LOW);
    digitalWrite(Moteur.IN3, LOW);
    digitalWrite(Moteur.IN4, LOW);
}
```

Imatge 20: Codi. Font pròpia.

- ***DoRotation(char *str)***: Controla els moviments del cub basats en la seqüència de caràcters rebuts. Mostra el progrés a la pantalla LCD.
- ***Rotation(char c, char r)***: Realitza la rotació dels motors basant-se en els caràcters rebuts. Utilitza altres funcions per controlar els motors.
- Funcions com ***"OneRotation", "TwoRotation", "disablemotor"*** són utilitzades per controlar els motors pas a pas.

5. Funcions auxiliars:

La funció ***"size"*** calcula la longitud d'una cadena de caràcters i la funció ***"disablemotor"*** atura el moviment d'un motor, posant totes les seves entrades a un estat baix.

```
int size(char* arr) {
    int res = 0;
    while (arr[res] != '\0') {
        res++;
    }
    return res;
}

void disablemotor(struct moteur Moteur) {
    digitalWrite(Moteur.IN1, LOW);
    digitalWrite(Moteur.IN2, LOW);
    digitalWrite(Moteur.IN3, LOW);
    digitalWrite(Moteur.IN4, LOW);
}
```

Imatge 21: Codi. Font pròpia.

- ***"size(char *arr)"***: Calcula la mida d'una cadena de caràcter (es pot substituir amb strlen de la llibreria ***"(<string.h>)"***.
- ***"disablemotor(struct moteur Moteur)"***: Apaga el motor.

6. Control de motors:

Aquesta secció del codi conté la lògica per controlar els moviments dels motors, utilitzant funcions com ***"Rotation"***, ***"OneRotation"***, i ***"TwoRotation"***. La funció "***Rotation"*** realitza les diferents combinacions de moviments i controla els motors segons els caràcters rebuts.

Les funcions ***"OneRotation"*** i ***"TwoRotation"*** executen les seqüències de passos necessaris per fer les rotacions d'un sol motor i dos motors oposats.

```
void Rotation(char c, char r) {
    int res = 0;
    struct moteur RotD = {30, 32, 34, 36};
    struct moteur RotR = {22, 24, 26, 28};
    struct moteur RotL = {23, 25, 27, 29};
    struct moteur RotB = {39, 41, 43, 45};
    struct moteur RotF = {38, 40, 42, 44};
    struct moteur RotU = {31, 33, 35, 37};

    Serial.println(c);

    switch (c) {
        case 'F':
            if (r == 'B') {
                TwoRotation(RotF, RotB, 0, 1);
                res = 1;
            } else if (r == 'b') {
                TwoRotation(RotF, RotB, 0, 0);
                res = 1;
            } else {
                OneRotation(RotF, 0);
            }
            break;
        case 'D':
            // Lógica similar para los otros casos (D, U, R, L, B, f, d, u, r, l)
            break;
    }
```

Imatge 22: Codi. Font pròpia.

```
    disablemotor(RotF);
    disablemotor(RotD);
    disablemotor(RotU);
    disablemotor(RotB);
    disablemotor(RotL);
    disablemotor(RotR);
    return res;
}

void OneRotation(struct moteur Moteur, int rot) {
    int delayTime = 1;
    if (rot == 0) {
        // Código para realizar una rotación en un motor
    } else {
        // Código para realizar una rotación inversa en un motor
    }
}
```

Imatge 23: Codi. Font pròpia.

```
void TwoRotation(struct moteur Moteur, struct moteur Reverse, int rot, int rotR) {
    int delayTime = 1;
    if (rot == 0 && rotR == 0) {
        // Código para realizar dos rotaciones en motores opuestos
    } else if (rot == 1 && rotR == 1) {
        // Código para realizar dos rotaciones inversas en motores opuestos
    } else if (rot == 0 && rotR == 1) {
        // Código para realizar una rotación en un motor y una rotación inversa en ot
    } else if (rot == 1 && rotR == 0) {
        // Código para realizar una rotación inversa en un motor y una rotación en ot
    }
}
```

Imatge 24: Codi. Font pròpia.

- Es defineixen diferents moviments de rotació per als motors basats en les lletres ('F', 'B', 'U', 'D', 'R', 'L') i les seves combinacions entre ('f', 'b', 'u', 'd', 'r', 'l').
- Les funcions de rotació controlen els motors pas a pas per realitzar els moviments de la galleda de Rubik.

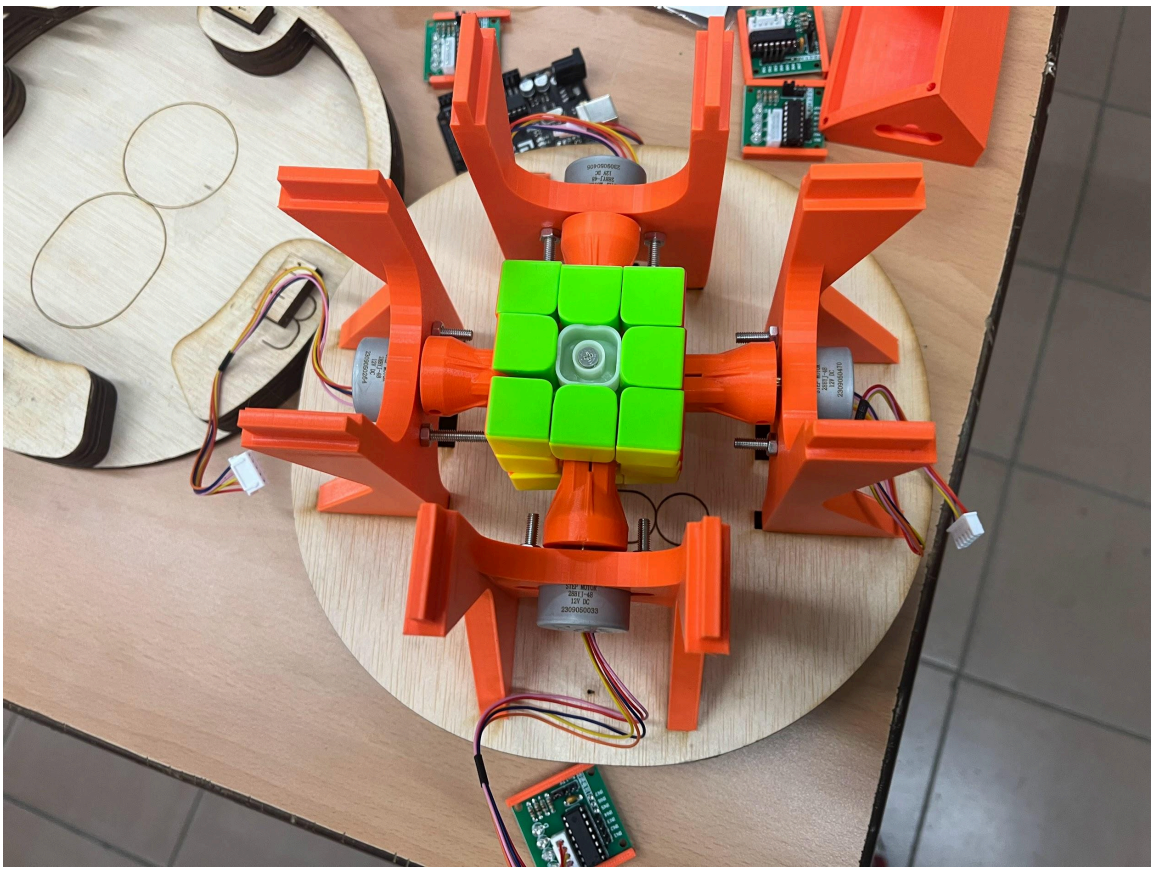
4.3 Creació de la maquinària.

La màquina ha sigut creada gràcies a diverses entitats, sense aquesta ajuda no hauria pogut fer el meu treball, la meva part pràctica.

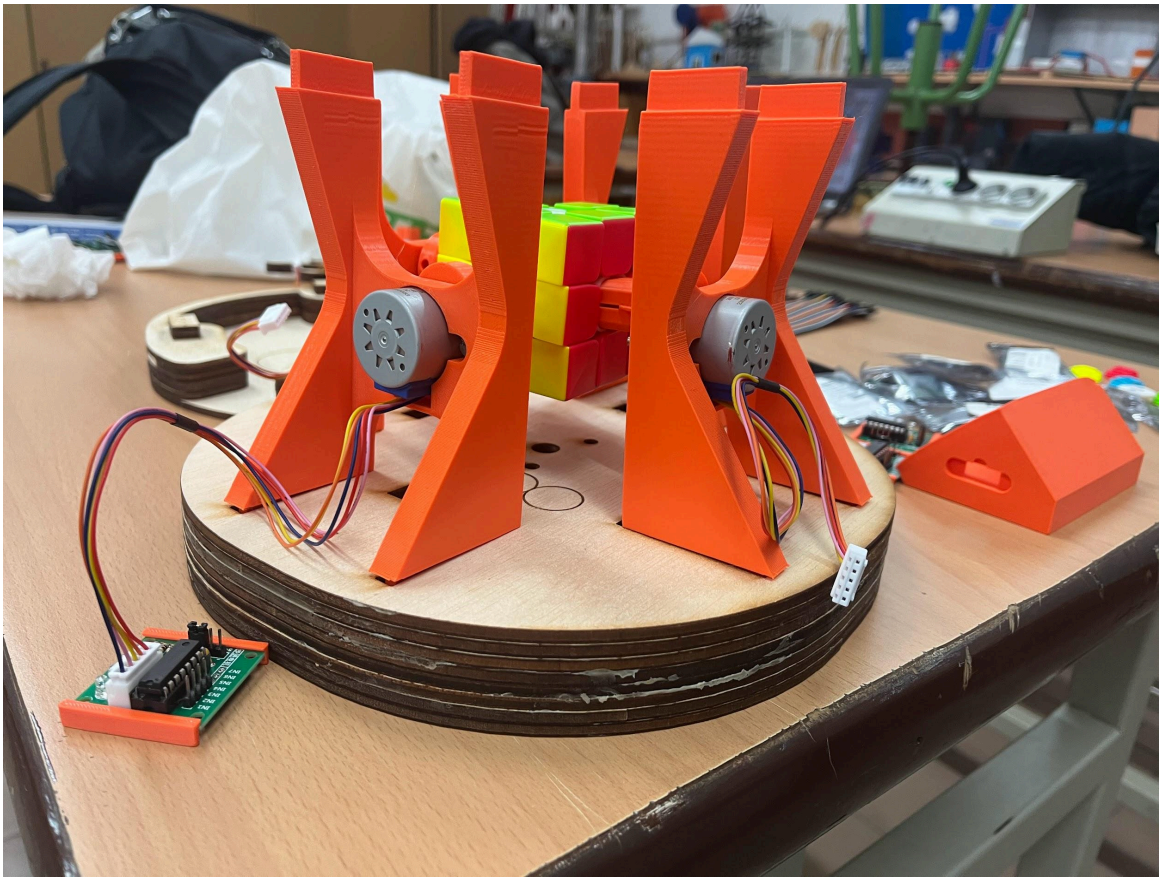
Per fer-lo he d'aconseguir totes les peces necessàries per muntar la màquina, és un procés llarg i pacient, però, amb temps ho aconseguim.

A continuació de les següents fotografies, explicaré atentament els processos per obtenir les peces.

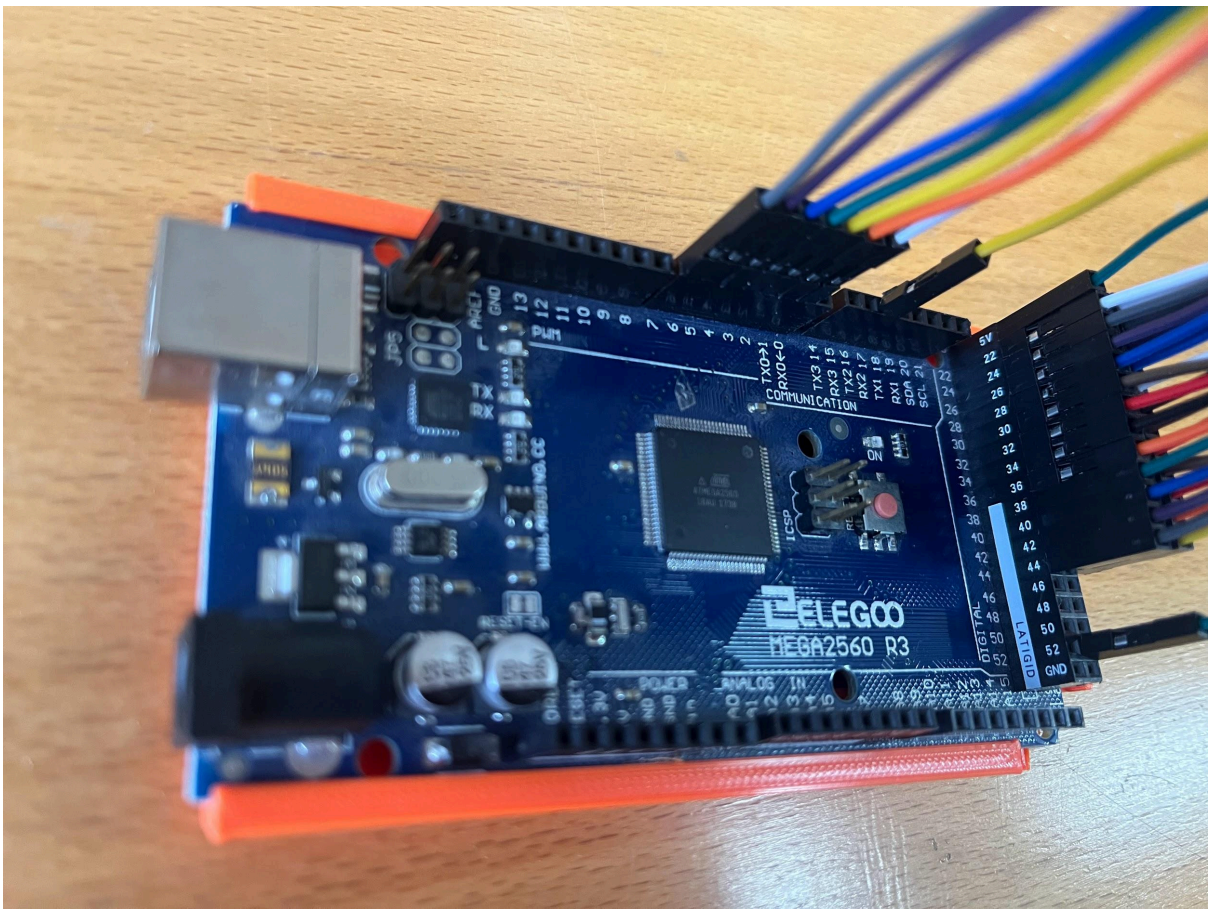
Imatge 25: Estructura final, maquinària. Font pròpia.



Imatge 26: Estructura final, maquinària. Font pròpia.



Imatge 27: Estructura final, maquinària. Font pròpia.



Imatge 28: Estructura final, placa Arduino. Font pròpia.

- 4.3.1 Peces al centre

Les peces de l'esquelet han sigut moldeades a partit del ***TinkerCad***, és una eina en línia i gratuïta, que ens permet crear models tridimensionals basats en la geometría sòlida constructiva, aquesta utensili és relativament molt més senzill que altres programes com ***Blender***.

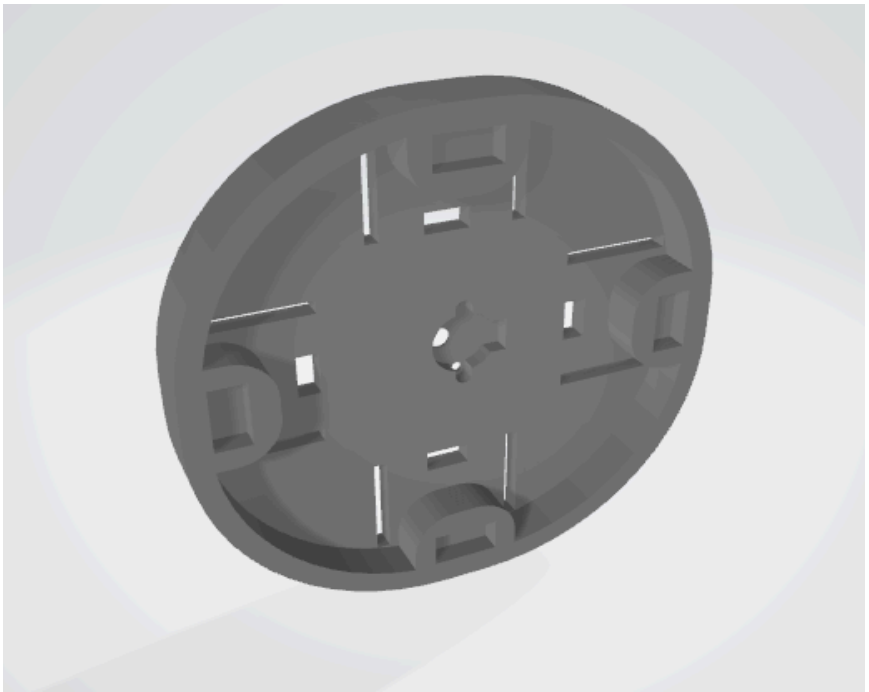
Les peces una vegada moldeades, han sigut instal·lades en format **STL**, un format universal per definir figures geométriques en 3 dimensions. Aquest format a diferència d'altres formats més pesats en emmagatzemament i complexos, no inclou informació dels colors o textures que dels altres si ofereixen.

Per crear la nostra màquina hem de fabricar-lo peça per peça.

La màquina constarà principalment d'una placa Arduino on està connectada tots els sistemes, 6 controladors de motors connectats contínuament amb 5 motors i també a una pantalla on ens pot mostrar els moviments en directe. Tot el mecanisme està subjecte a un esquelet com suport i protecció, on dona una seguretat i una estètica a la maquinària. Aquest esquelet està compost per:

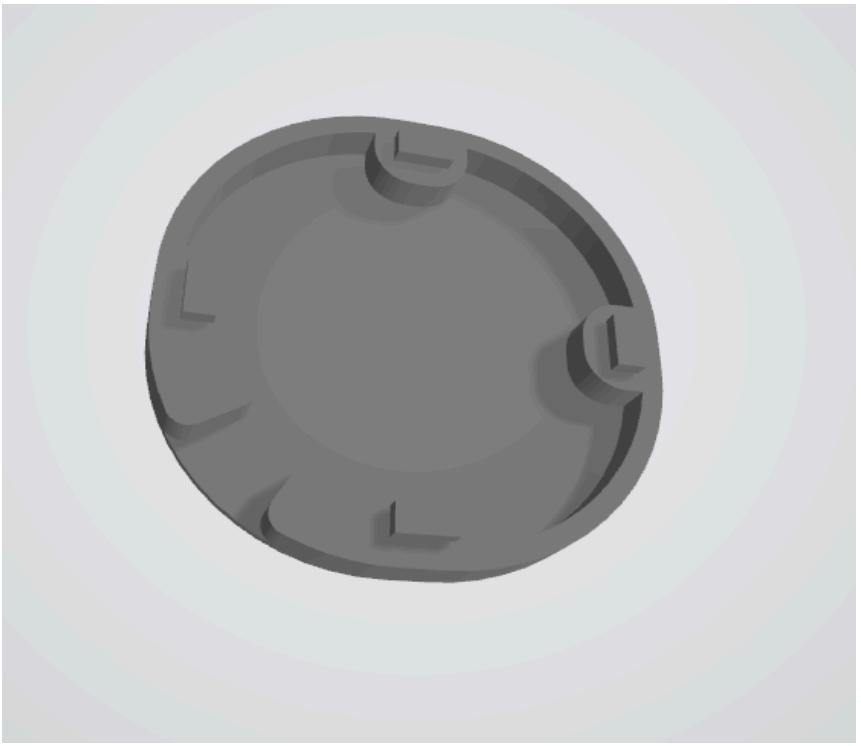
1. Una base

- Base inferiorA x1



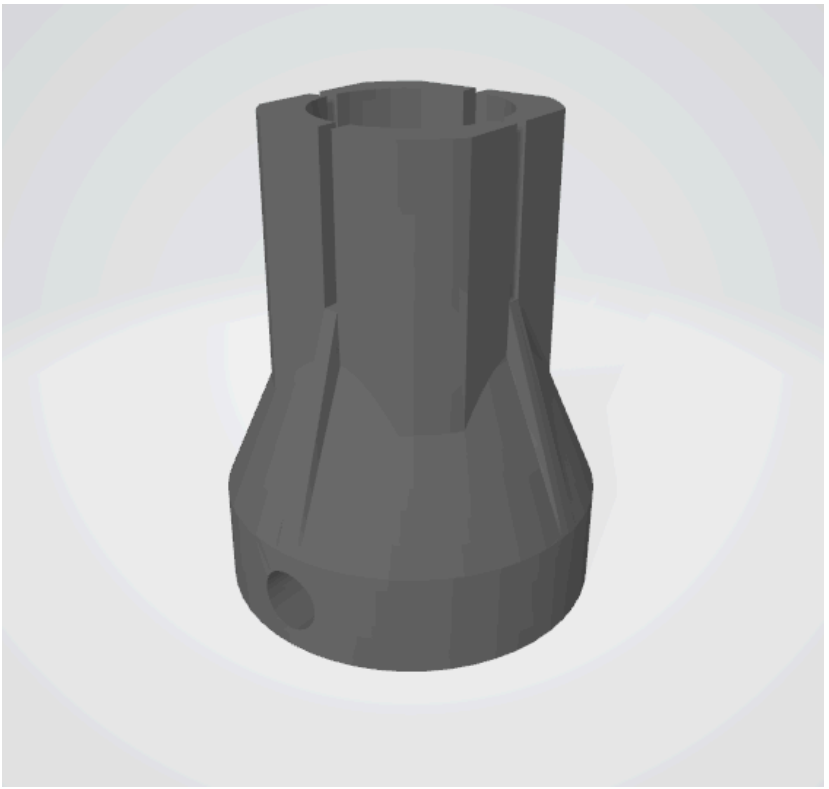
Imatge 29: Base inferiorA de l'esquelet. Font pròpia.

- Base inferiorB x1



Imatge 30: Base inferiorB de l'esquelet. Font pròpia.

2. Conector al cub de Rubik x4



Imatge 31: Connector de l'esquelet. Font pròpia.

3. Costat x4

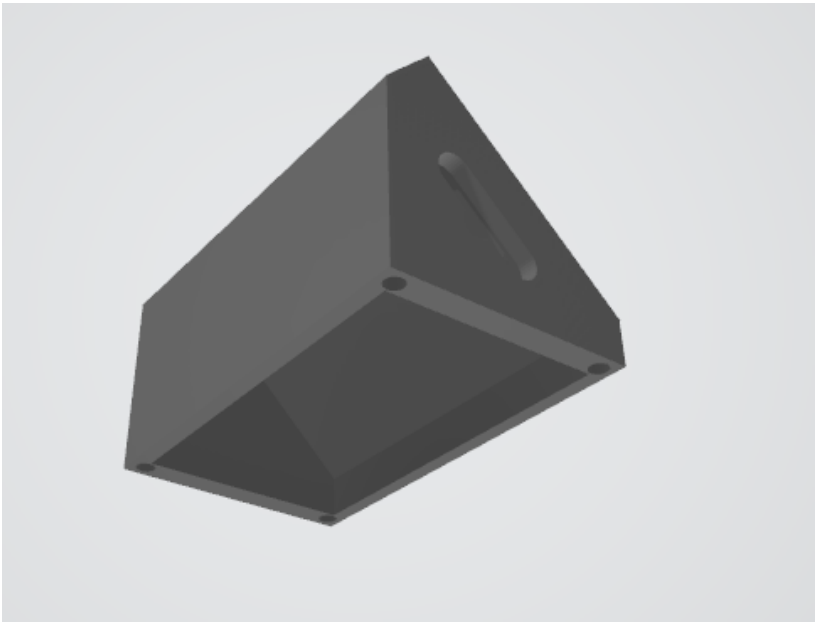
Imatge 32: costats de suport de l'esquelet. Font pròpia.

4. Suport per la placa Arduino x1

Imatge 33: Suport per la placa arduino de l'esquelet. Font pròpia.

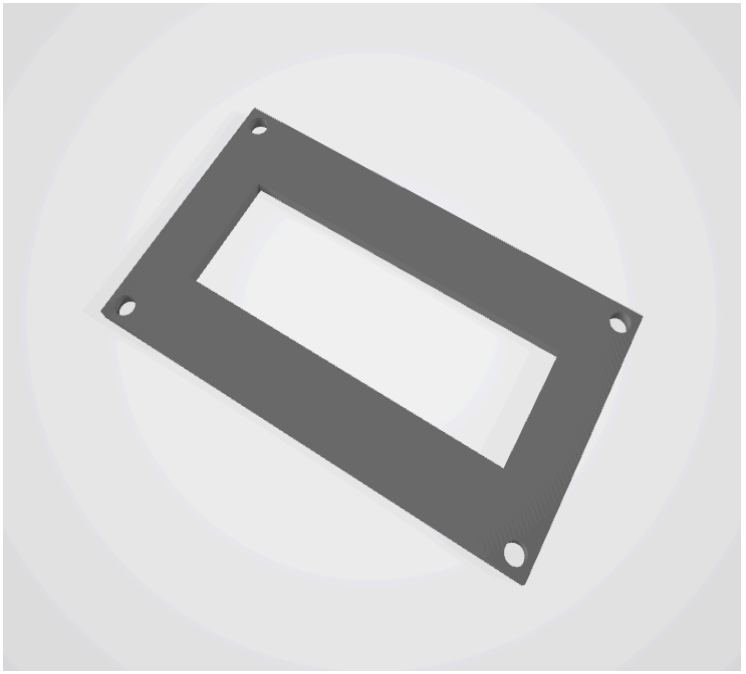
5. Suport per la pantalla

- Base x1



Imatge 34: Suport per la pantalla LED de l'esquelet. Font pròpia.

- Contorn per la pantalla x1



Imatge 35: Contorn per la pantalla LED de l'esquelet. Font pròpia.

6. Encaixament de la base per la pantalla x1

Imatge 36: Suport pels controladors dels motors de l'esquelet. Font pròpia.

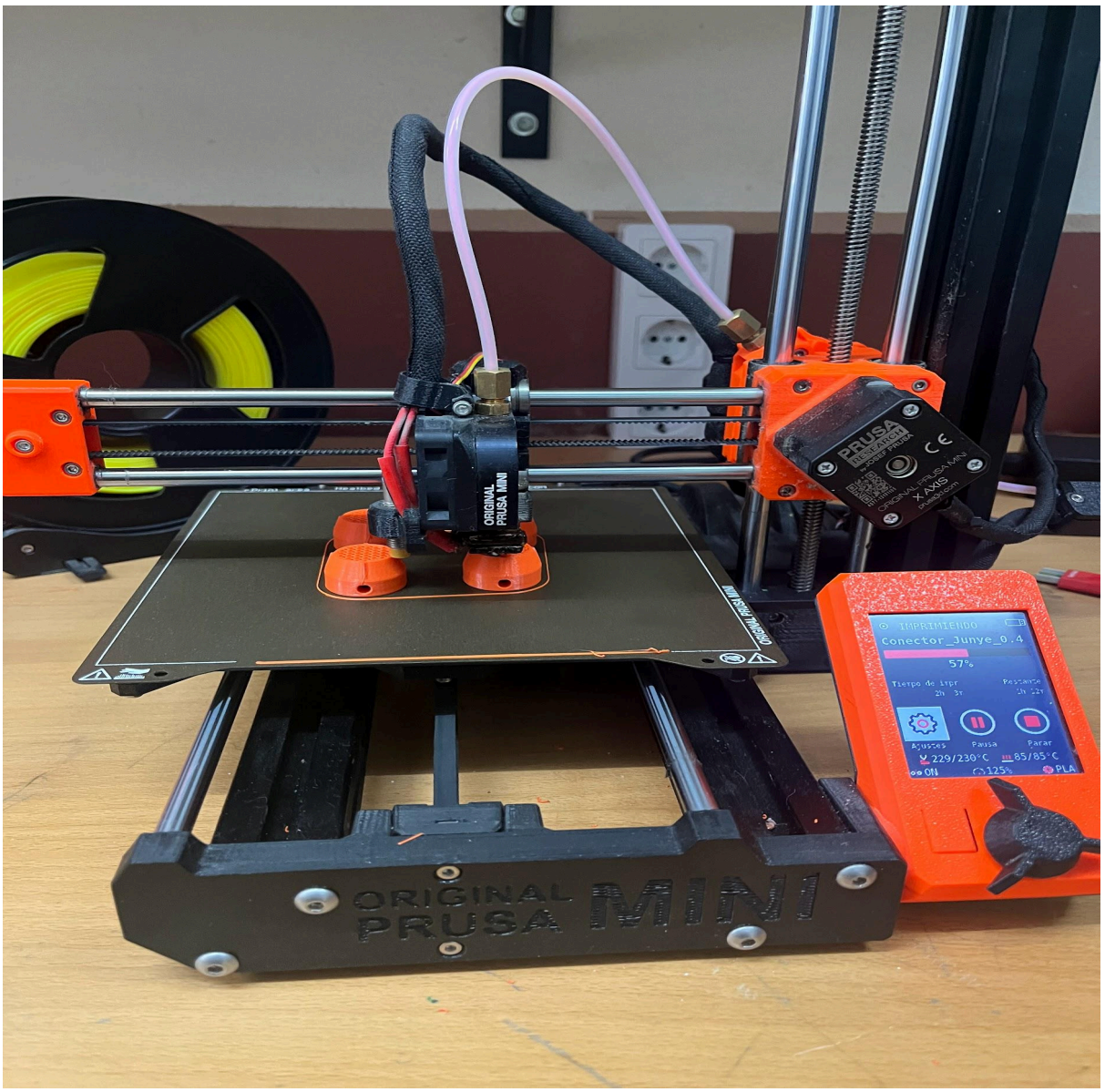
7. Una base o capa superior x1

Imatge 37: Base superior de l'esquelet. Font pròpia.

8. Barres per suportar els motors x6

Imatge 38: Barres internes per suportar el motor de l'esquelet. Font pròpia.

Les peces de la nostre màquina ho fabriquem a partir d'una impresora 3D, en el nostre cas utilitzem l'impressora *Monoprice 121711 Select Mini 3D Printer V2 de la marca IIIP PRINTER* de l'institut.



Imatge 39: Màquina 3D funcionant. Font pròpia.

Haurem de posar en marxa un total de 16 vegades la impressora 3D, consta de 1 base inferiorA, 1 base inferiorB (connectades totes dues bases per emmagatzemar els components), 4 connectors al cub de Rubik, 4 suport (costats) de l'esquelet,1 suport per la placa arduino, 1 suport per la pantalla que mostra, 1 contorn per la pantalla, 1 suport per el motor, 1 base superior i finalment 6 barres de suport dels motors.

La base superior, la base inferiorA i inferiorB, han de ser impreses en un altre lloc, ja que l'impressora del centre no és suficientment gran per la fabricacions d'aquestes.

-4.3.2 Experiència a l'ateneu tecnològic de Barcelona

Gràcies a l'Ateneu tecnològic de Barcelona, explicada anteriorment en l'últim punt dins el 4.1. vaig poder realitzar les bases inferiors i la base superior.

Per poder assistir a una classe o ajuda de l'Ateneu hem de demanar comanda, i si és possible amb antelació, la comanda pot ser per la pàgina web de l'ateneu o per trucada, i aquesta ha d'anar acompanyada de les dades de les entitats o entitat d'aquesta sol·licitud, com el nombre de participants, del centre procedent i més.

En el meu cas, l'assistència a la classe magistral de les màquines per fabricar objectes i sobretot l'ajuda que em van oferir a cooperar pel meu treball de recerca, va ser tractada el dia anterior mitjançant una trucada. A la trucada molt amablement em van fer una excepció d'un bell en sec i em van deixar assistir el dia posterior a les seves instal·lacions, amb monitors amb molt bona formació on ens garanteixen una bona estona i un gran aprenentatge.

L'endemà a les 11:00 a.m., vaig assignar-me en C/Marie Curie 8-14, al Parc Tecnològic de Barcelona, on vaig presenciar tota la innovadora instal·lació.

Imatge 40: Parc tecnològic. Extreta de Google: https:/e00-expa

Imatge 41 : Dins el Parc tecnològic.Extreta de https://www.barcelonactiva.cat

Em van portar cap a la zona de l'Ateneu, on em vaig trobar amb dues monitores/professores molt simpàtiques, també es troben altres persones on tenen dubtes pels seus casos personals, per exemple un d'ells en aprendre a utilitzar el TinkerCad i un altre sent assistit per una monitora en el disseny 3D.

A la meva presència, vaig ser assistit directament per una d'elles, em van mostrar tota la part de l'Ateneu, una zona molt còmoda i agradable, seguidament em va mostrar totes les màquines que podem fer-ne ús.
Començant parlant de la meva assistència per l'ajuda de la part pràctica, vaig explicar-li les meves necessitats d'una impressora 3D més gran per realitzar la meva construcció, llavors em va oferir diverses maneres de fer ús de les diverses màquines que tenen.

Imatge 42: Sala de l'Ateneu, màquines. Font pròpia.

Imatge 43: Sala de l'Ateneu, màquines. Font pròpia.

Va parlar d'un món de màquines especialitzades per la creació d'estructures i construcció, entre elles em van explicar la complexitat i totes:

- Les formes de fabricació per estalviar material.
- L'estalvi de costos innecessaris.
- Les millors textures.
- L'ús específic de cada material.
- El material més útil.
- L'estalvi de temps.

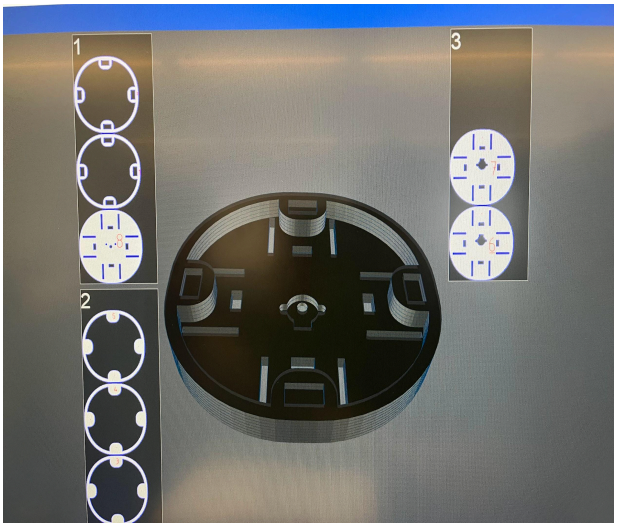
D'entre totes vaig escollir finalment per la talladora de fusta a làser.

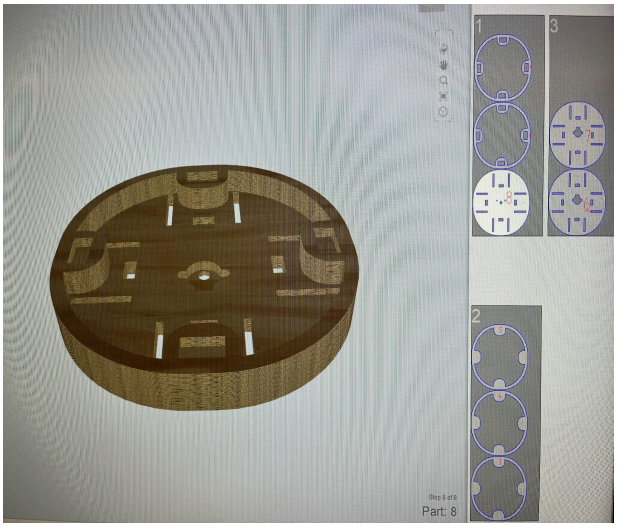
La talladora de fusta a làser, era la més útil i ràpida per les meves necessitats en aquell instant, ja que, per una banda, la impressora 3D triga massa temps a fabricar les 3 peces i per altre la impressora que ells tenen era d'una dimensió massa exacta, per mil·límetres, petita per les meves peces.

Van fer nombroses activitats abans de tallar la fusta.

1. Transportar les meves STL al programa.

   Les peces les portava en el PEN, i dins l'Ateneu vaig haver de passar-los al SLICER, on transforma els formats 3D en làmines de la mateixa dimensió, on podem observar totes les peces dividides.

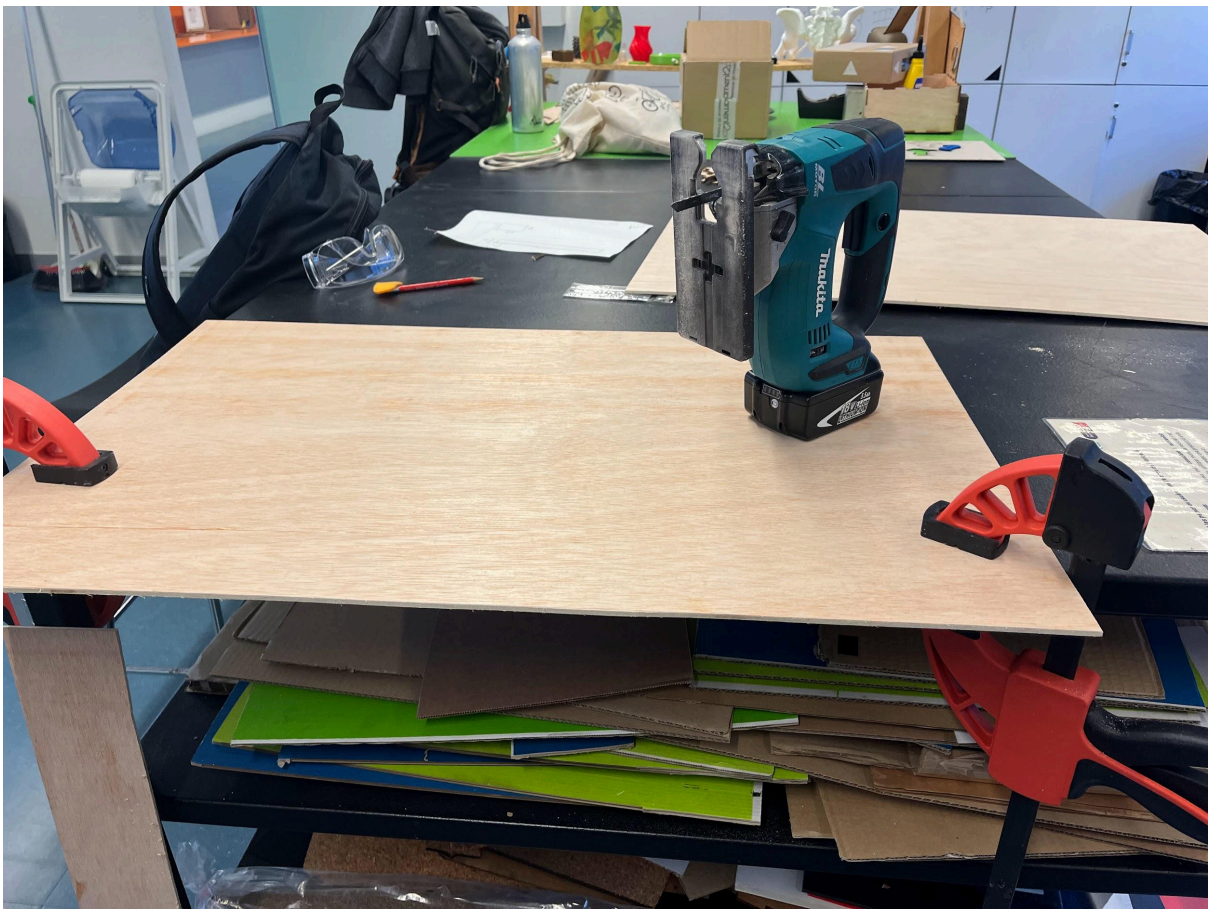




Imatges 44: SLICER, programa. Font pròpia.

2. Tallar les taules de fusta.

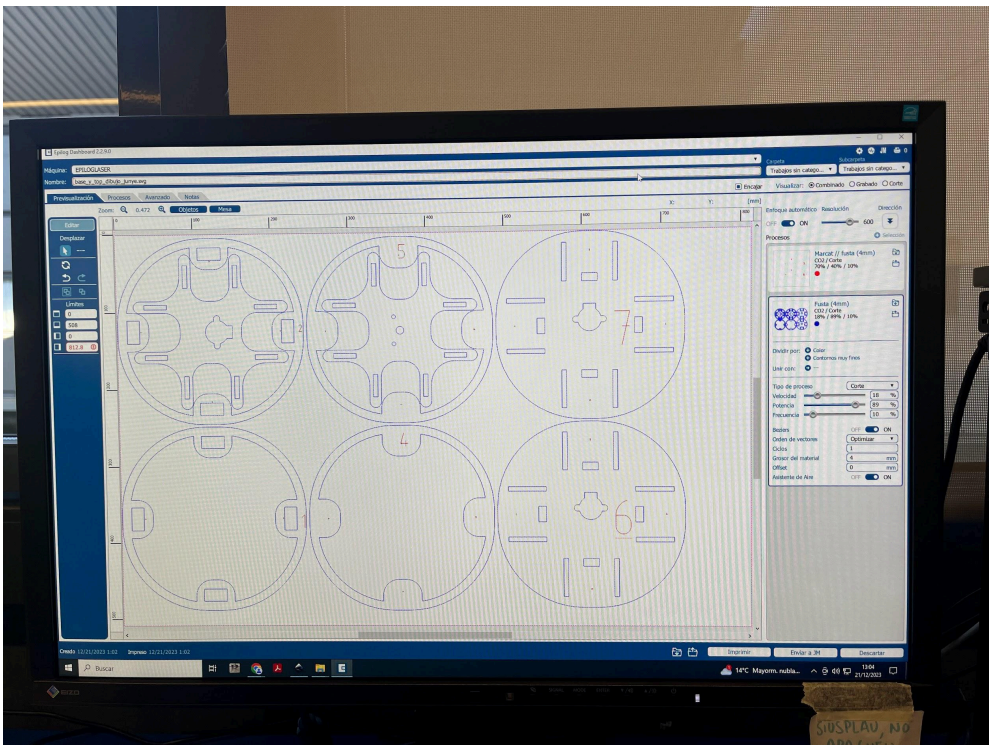
   Hem utilitzat taules de fustes de (500x800x4) mm per realitzar les peces, on em van ensenyar utilitzar la serra elèctrica i ho vaig tallar personalment, sempre amb protecció ulleres, guants i les instruccions necessàries per construir les dimensions indicades.



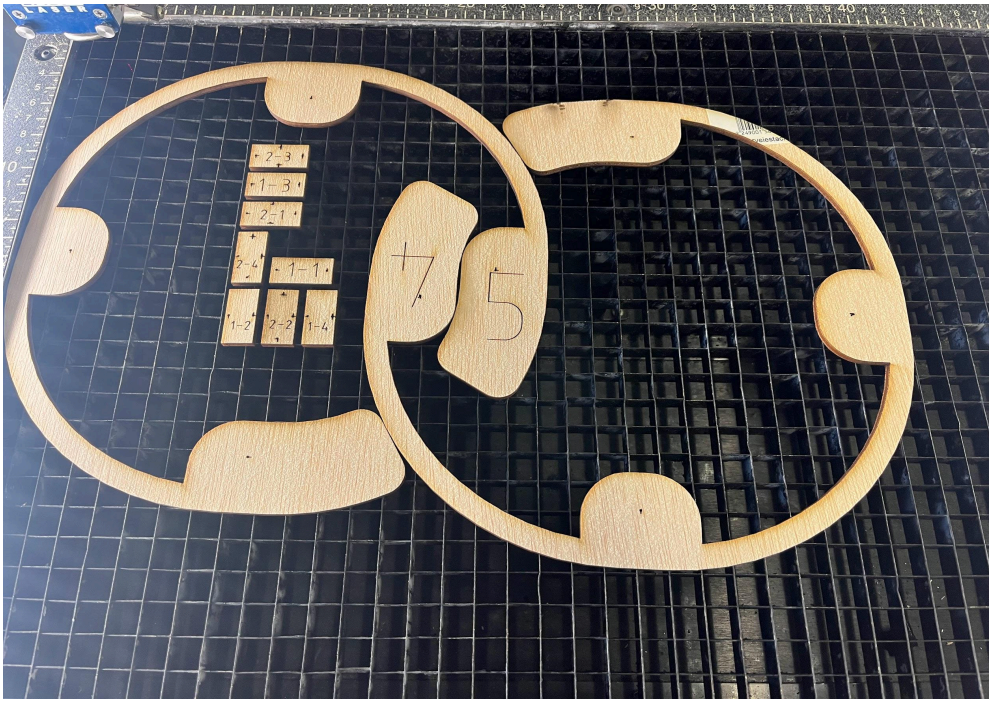
Imatge 45: Sala de l'Ateneu, màquines. Font pròpia.

3. Puzle amb SLICER.

   Un cop tenim les taules de les dimensions requerides, com ja he comentat abans, l'estalvi de material és imprescindible. Per aquesta raó, vam fer puzle per fer més petita possible els metres quadrats que necessitem per fer en cada taula a SLICER.



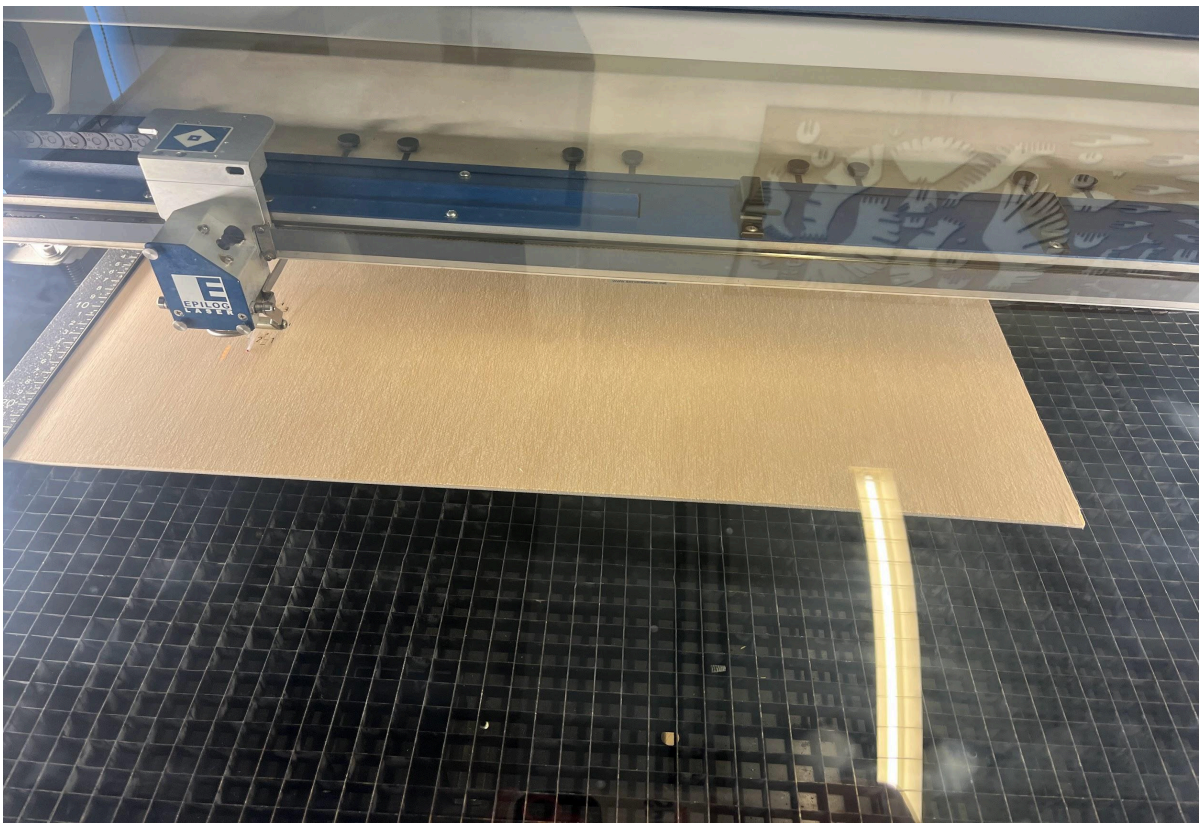
Imatge 46: Sala de l'Ateneu, SLICER, puzzle. Font pròpia.



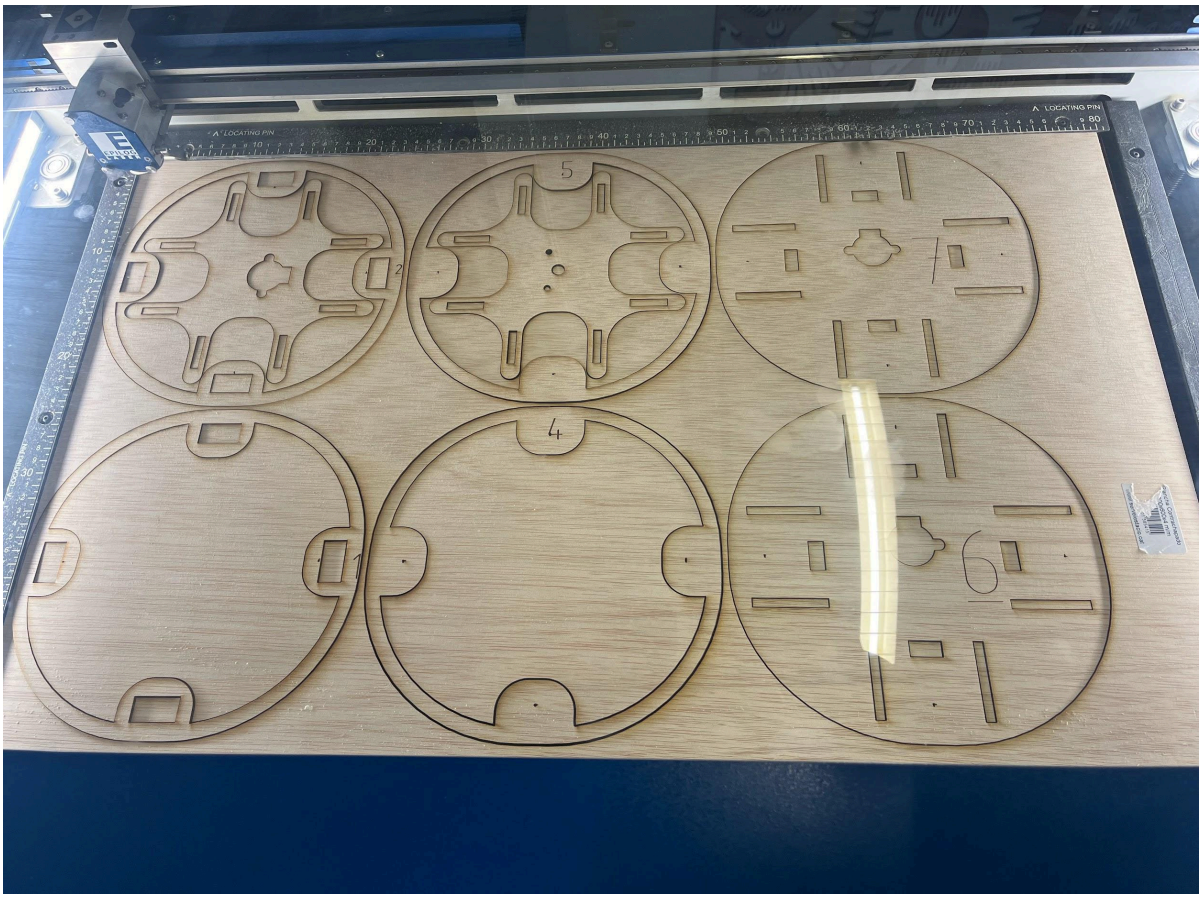
Imatge 47: Sala de l'Ateneu, màquines, talladora làser. Font pròpia.

4. Posar en marxa el tallador làser.

   Un cop fet el puzle de l'estalvi, comencem a tallar.



Imatge 48: Sala de l'Ateneu, màquines, talladora làser. Font pròpia.



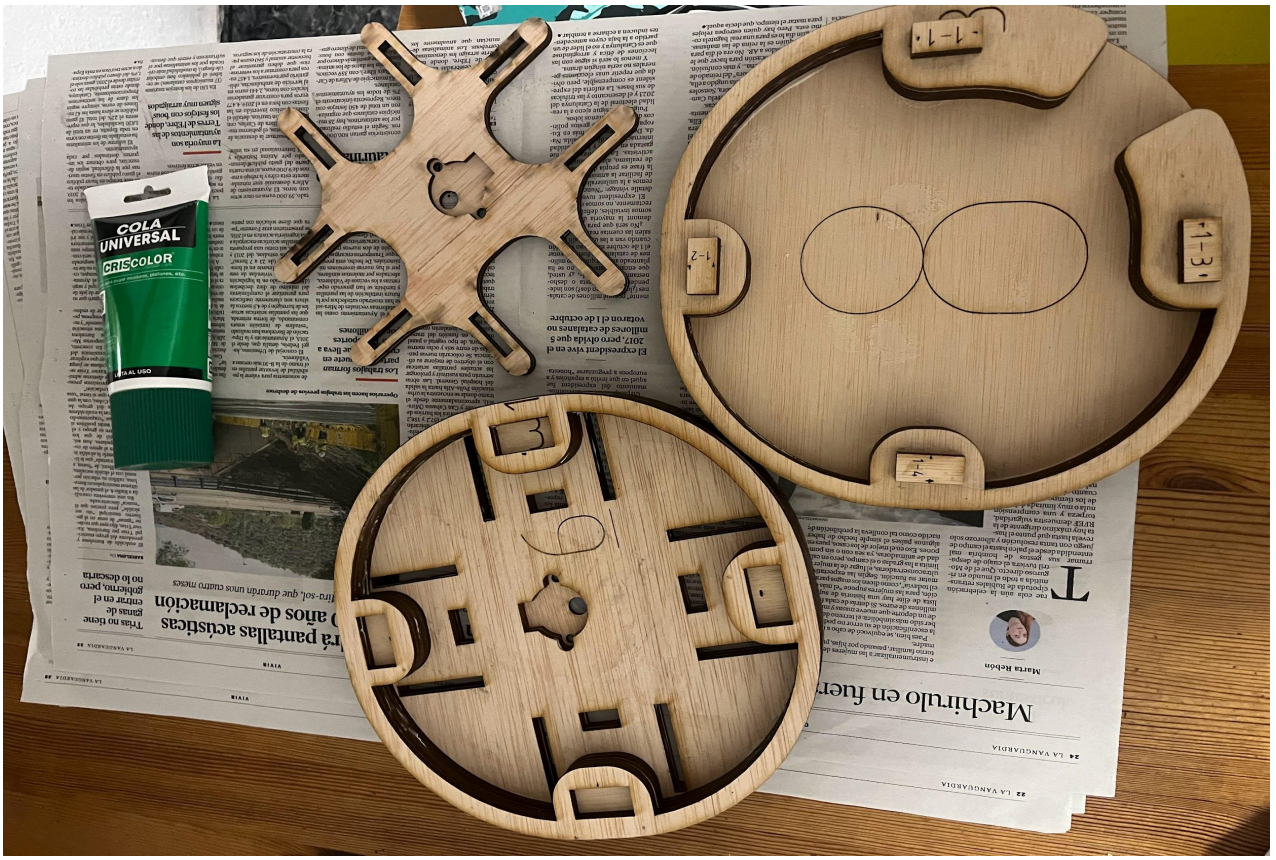
Imatge 49: Sala de l'Ateneu, màquines, talladora làser. Font pròpia.

Imatge 50: Sala de l'Ateneu, peces provinents de la talladora làser. Font pròpia.

5. Enganxar les peces.

   Finalment, ja obtinguts les peces tallades, hem d'unir totes les peces, per aquesta funció el SLICER, té una funció on nomena les capes per ordre i així podem enganxar-los amb seguretat.

Imatge 51: peces provinents de la talladora laser. Font pròpia.



Imatge 52: peces provinents de la talladora laser. Font pròpia.

En resum, aquesta va ser la meva experiència a l'Ateneu, volia agrair molt a l'Ateneu i sobretot a la monitora Clara, que em va donar una ajuda increïble i també per tota la masterclass que va fer-me del món del disseny i de la fabricació. L'Ateneu tecnològic de Barcelona és totalment aconsellable per experimentar i aprendre noves situacions.

5. Conclusions

Finalitzant aquest magnífic projecte, he après centenars de coses en les quals estic motivat de continuar adquirint coneixement, encara em situo a l'inici del primer pendent d'una serralada, perquè sé que soc un complet principiant en aquest món de la tecnologia.

En primer lloc, l'aprenentatge del funcionament dels llenguatges de programació i també les diferències entre aquestes m'han ajudat a decidir a aprendre un llenguatge, en el meu cas Python. Aquest curs de Python m'ha fet reflexionar sobre el futur com a programador que vull seguir, és una comunitat molt complexa i molt sincera, tothom s'ajuda i comparteix els coneixements per gaudir de millores per la informàtica a escala mundial.

En segon lloc, els cubs de Rubik són un trencaclosques molt atractiu i encara que no ho sembli, és una peça fonamental que ha ajudat a milions de persones crear una passió per les matemàtiques i la geometria tridimensional. També amb la informació de la WORLD CUBE ASSOCIATION podem dir que també és una comunitat meravellosa, on s'aprèn grans raons de la geometria i dels algoritmes matemàtics.

Per altra banda, ha sigut magnífic poder dissenyar figures en tres dimensions, ha sigut molt entretingut aconseguir formar i modelar cossos geomètrics, agafant coneixements generals dels programes per edició de formats tridimensionals. Seguidament, gràcies al centre Puig Castellar, amb les seves impressores 3D, fer realitat els dissenys obtinguts amb molt orgull.

A partir d'aquí aprendre sobre circuits elèctrics, on trobem el material que he necessitat pel funcionament de la màquina, la connexió dels pins sobre la placa Arduino, o el funcionament d'una pantalla LED, junt amb l'enteniment del programa o code escrita en C++ amb la relació directa amb el circuit.

Aquest treball m'ha ajudat a obrir els ulls i veure les possibilitats de tots els àmbits de les enginyeries en el món. L'enginyeria és una necessitat universal per millorar

les nostres ciutats, poblacions, carrers, funcionaments digitals, medi ambient i sobretot en el nostre dia a dia, facilitant milions de possibles problemes que hauríem d'afrontar si no tinguéssim totes les eines que utilitzem al llarg del nostre dia. Podem concloure amb l'afirmació de la importància de la completa relació entre totes les branques de les enginyeries, formant un sistema i comunitat meravellós on tots han aportat per realitzar un millor demà.

6. Agraïments

Ha sigut un honor realitzar i finalitzar aquest magnífic projecte de recerca. És un projecte el qual ja fa temps volia realitzar i aprendre sobre les principals tesis d'aquest document.

Vull comunicar i manifestar el meu agraïment al conjunt de persones que m'han ajudat en fer el meu treball, a totes les persones que han posat al seu gra de farina per contribuir la realització.

En primer lloc, els meus agraïments van directament als meus tutors del treball de recerca, Jaime Morcillo i Marc Gómez Llenas, pel seu enfocament i suport al meu treball, per tots els comentaris i correccions que hagi necessitat, sense vosaltres no hagués pogut dur a terme el projecte de la millor manera possible. També vull agrair al professorat del centre Puig Castellar per tot el suport que m'han oferit, especialment a Ione Rivas, qui m'ha animat i ajudat a recopilar idees pel meu enfocament al treball.

Per altra banda, m'agradaria donar les gràcies al centre IES Puig Castellar, per oferir-me el material essencial i maquinària invariable per la realització de la part pràctica. Sense la impressora 3D m'hauria sigut impossible portar a cap una estructura estèticament i funcionalment tan ergonòmica.

També fer referència, sent una peça central, a tots els amics i companys que m'han aportat la força i motivació necessària per mantenir-me firm davant d'aquesta formulació, i també per aquells que m'han suggerit i comentat matèria necessària per complementar el treball.

En últim lloc, expressar reconeixement a la meva família, quins han estat tot el temps darrere meu perquè la meva ànima no s'esgoti mai, donant-me ànims i força contínuament.

La col·laboració de tot el conjunt de persones ha sigut indispensable per l'acabament i per la realització del meu propòsit. L'ajuda que he rebut no podria ser de millor manera, ha sigut immancable a un altre nivell.

Els meus sincers agraïments, moltes gràcies a tothom!

7. Webgrafia i Bibliografia

- Llibre: **“APRENDE ELECTRÓNICA Y ROBÓTICA EDUCATIVA”** de *Roberto Montero Miguel,* Editorial: “ANAYA multimedia” <WebCompraLlibre "Aprende Electrónica y Robótica Educativa">

- Explicació biografía Erno Rubik [en línia]. Espanya, 21 de maig de 2023 a les 17:31 [Consultat: 2 de Septembre de 2023] Disponible a: <https://es.wikipedia.org/wiki/Ernő_Rubik>

- Explicació gràfica i complexa de com fer un cub de rubik amb la creu [En línea]. España, 10 de març de 2015 [Consultat: 28 d’Agost de 2023] Disponible a: <https://www.youtube.com/watch?v=GyY0OxDk5lI>

- Explicació biografía Erno Rubik [en línia]. Espanya, 2017 [Consultat: 6 de Setembre de 2023] Disponible a: <https://www.biografias.es/famosos/erno-rubik.html>

- Explicació Cub de Rubik [en línia]. Espanya, 2020 [Consultat: 7 de Setembre de 2023] Disponible a: <https://es.wikipedia.org/wiki/Cubo_de_Rubik>

- Explicació biografía Erno Rubik [en línia]. Espanya, 2015 [Consultat: 7 de Setembre de 2023] Disponible a: <https://es.wikipedia.org/wiki/Ernő_Rubik>

- Explicació Nombs dels diferents Cubs de Ruibik [en línia]. Espanya, 2019 [Consultat: 8 de Setembre de 2023] Disponible a: <https://kubekings.com/blog/post/tipos-de-cubos-rubik-y-sus-nombres>

- Explicació Història del Cub de Rubik [en línia]. Espanya, 2021 [Consultat: 8 de Setembre de 2023] Disponible a: <https://historia.nationalgeographic.com.es/a/historia-cubo-rubik-artefacto-prodigioso_17584>

- Explicació Campionat mundial, inscripcions Espanya [en línia]. EEUU, 2022 [Consultat: 8 de Setembre de 2023] Disponible a: <https://www.worldcubeassociation.org/competitions/FMCSpain2023>

- Explicació sobre els llengüatges de programació més utilitzats. [en línia]. Espanya, 2023 [Consultat: 14 de Setembre de 2023] Dsiponible a: <https://talently.tech/blog/los-10-lenguajes-de-programacion-mas-utilizados/>

- Material per la fabricació del mecanisme, Mòdul de Pantalla [en línia]. Xina, 2022 [Consultat: 12 d'octubre de 2023] Disponible a: <https://es.aliexpress.com/item/1967124495.html?spm=a2g0o.order_list.order_list_main.5.21ef194dnOFEYb&gatewayAdapt=glo2esp>

- Material per la fabricació del mecanisme, Motors 12W [en línia]. Xina, 2022 [Consultat: 12 d'octubre de 2023] Disponible a: <https://es.aliexpress.com/item/1005001494924034.html?spm=a2g0o.order_list.order_list_main.10.21ef194dnOFEYb&gatewayAdapt=glo2esp>

- Material per la fabricació del mecanisme, Controladors de motors [en línia]. Xina, 2022 [Consultat: 12 d'octubre de 2023] Disponible a: <https://es.aliexpress.com/item/1005001381657649.html?spm=a2g0o.order_list.order_list_main.16.21ef194dnOFEYb&gatewayAdapt=glo2esp>

- Informació d'Arduino, Definició [en línia]. Amèrica Latina, 2022 [Consultat: 13 d'octubre de 2023] Disponible a: <https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>

- Informació d'impressora, Monoprice 121711 Select Mini 3D Printer V2 [en línia]. Canadá, 2023 [Consultat: 15 de novembre de 2023] Disponible a: <https://www.amazon.ca/Monoprice-Select-Mini-Printer-Assembled/dp/B073ZLSMFT>

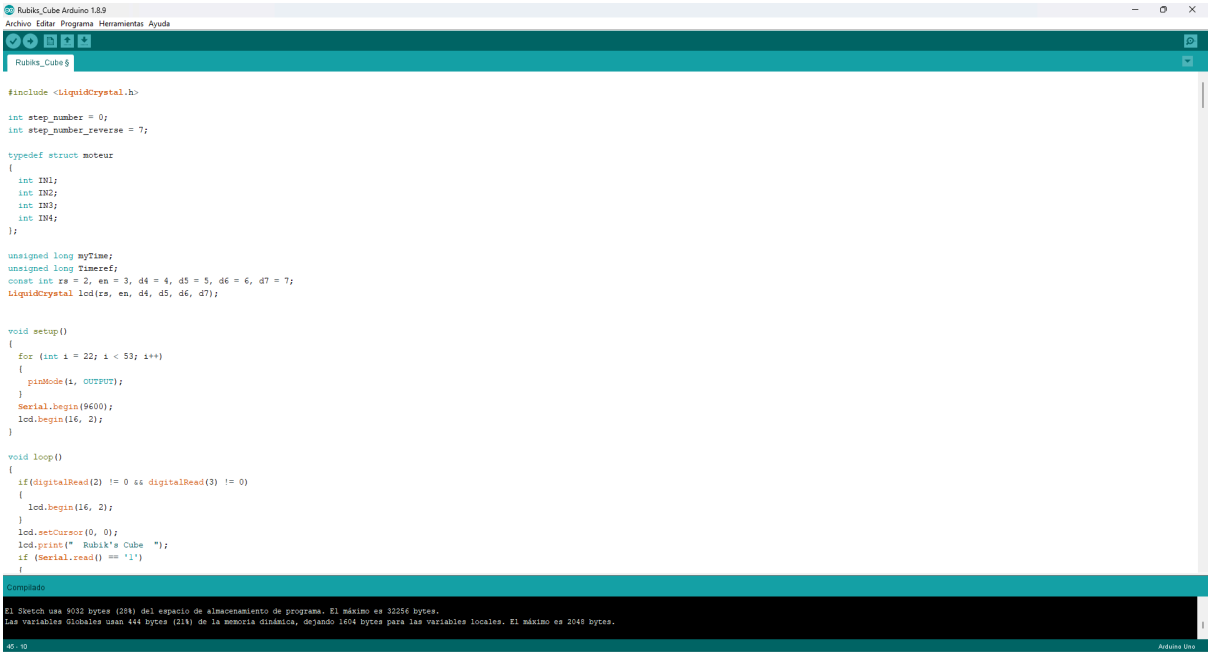
- Informació d'Arduino, Definició [en línia]. Europa, 2023 [Consultat: 15 de novembre de 2023] Disponible a: <https://es.wikipedia.org/wiki/Arduino>

- Informació d'Arduino, Definició [en línia]. Europa, 2023 [Consultat: 16 de novembre de 2023] Disponible a: <https://aprendiendoarduino.wordpress.com/2016/12/11/ide-arduino/>

- Informació d'Arduino, Definició [en línia]. Europa, 2023 [Consultat: 16 de novembre de 2023] Disponible a: <https://www.barcelonactiva.cat/es/-/parc-tecnologic>

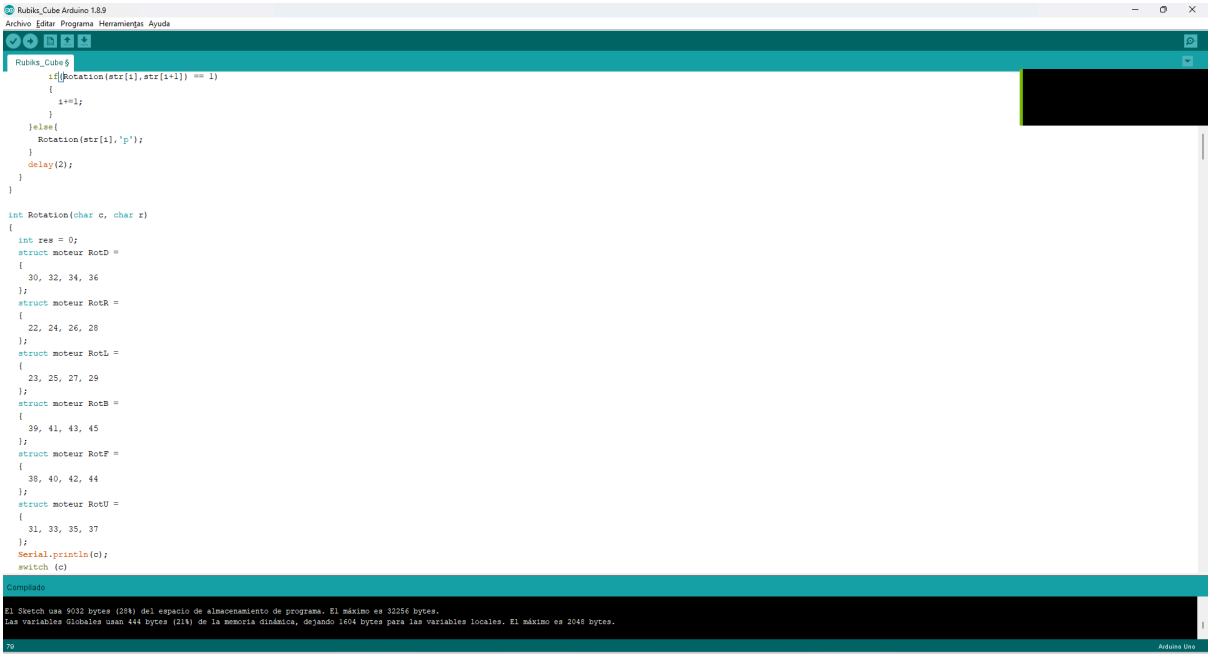
8. Annex

8.1 Programa sencer.

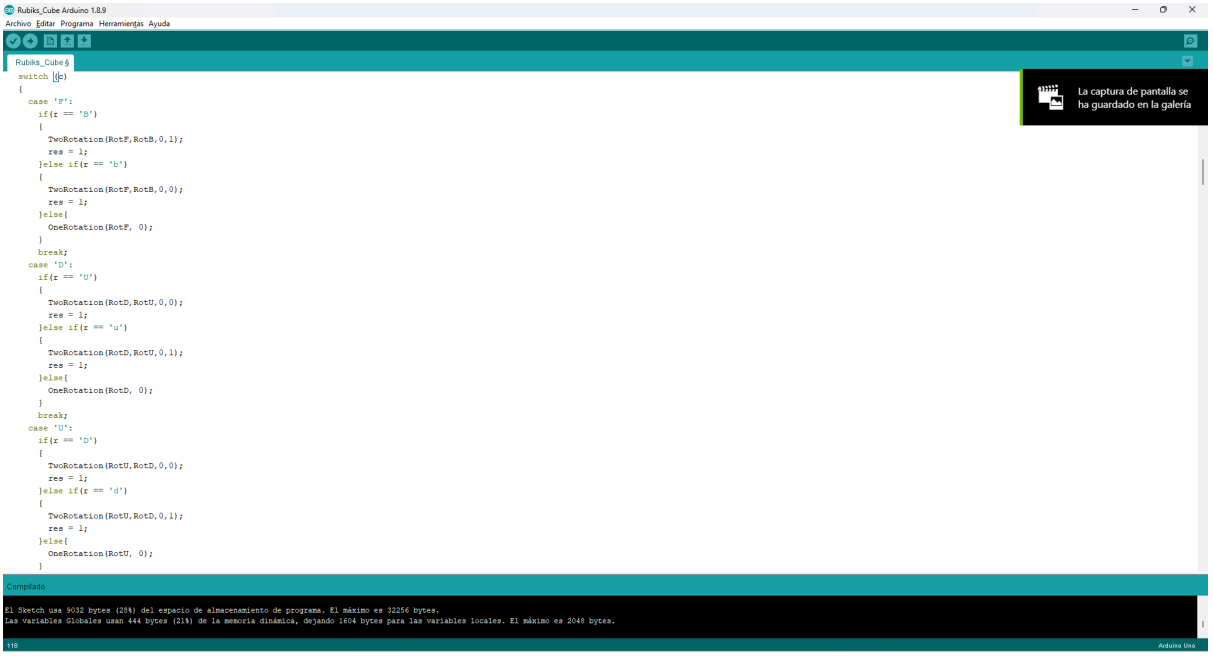


Imatge 53:: Programa sencer. Font pròpia

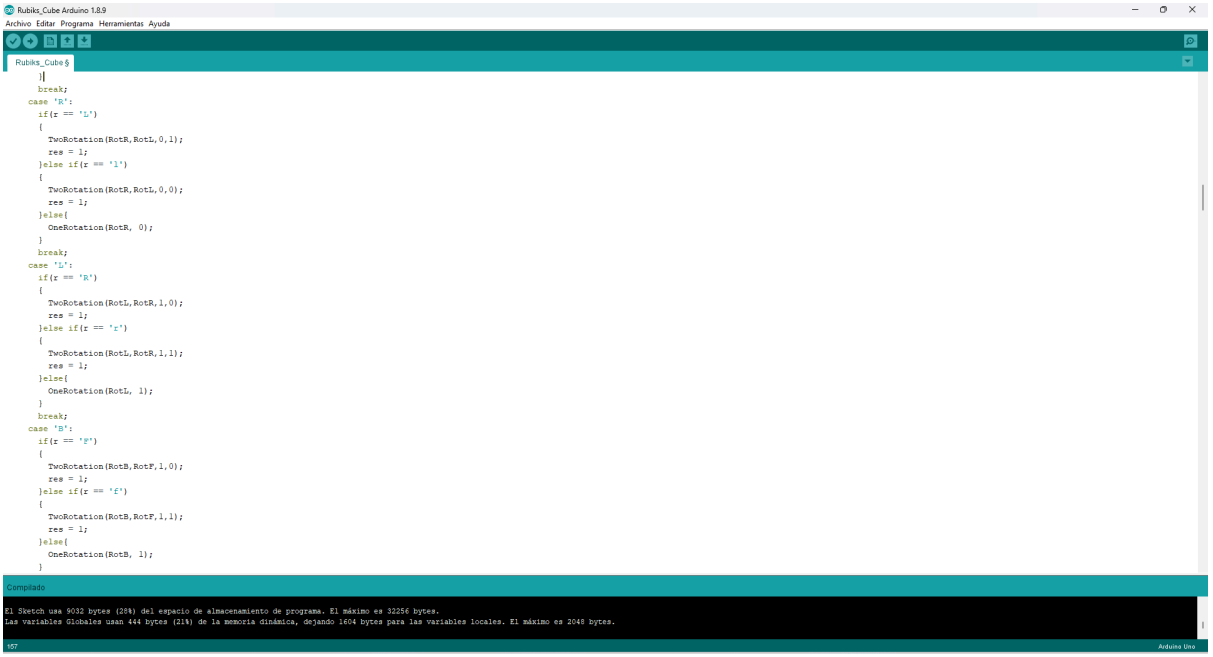
Imatge 54: Programa sencer. Font pròpia



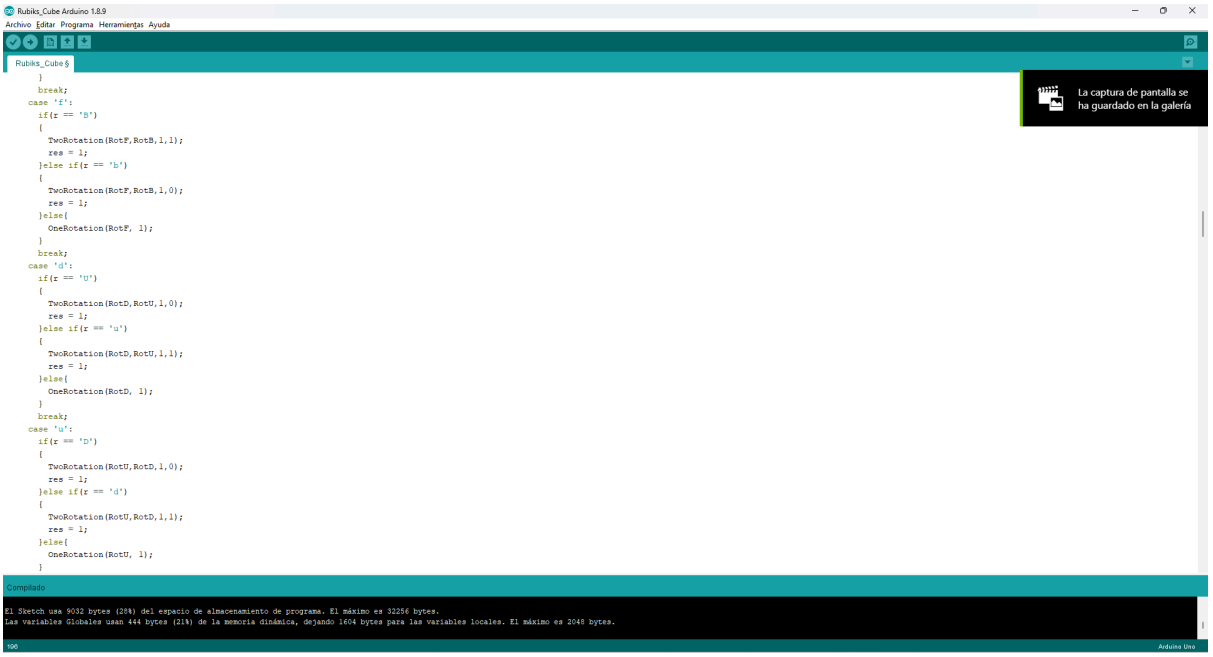
Imatge 55: Programa sencer. Font pròpia



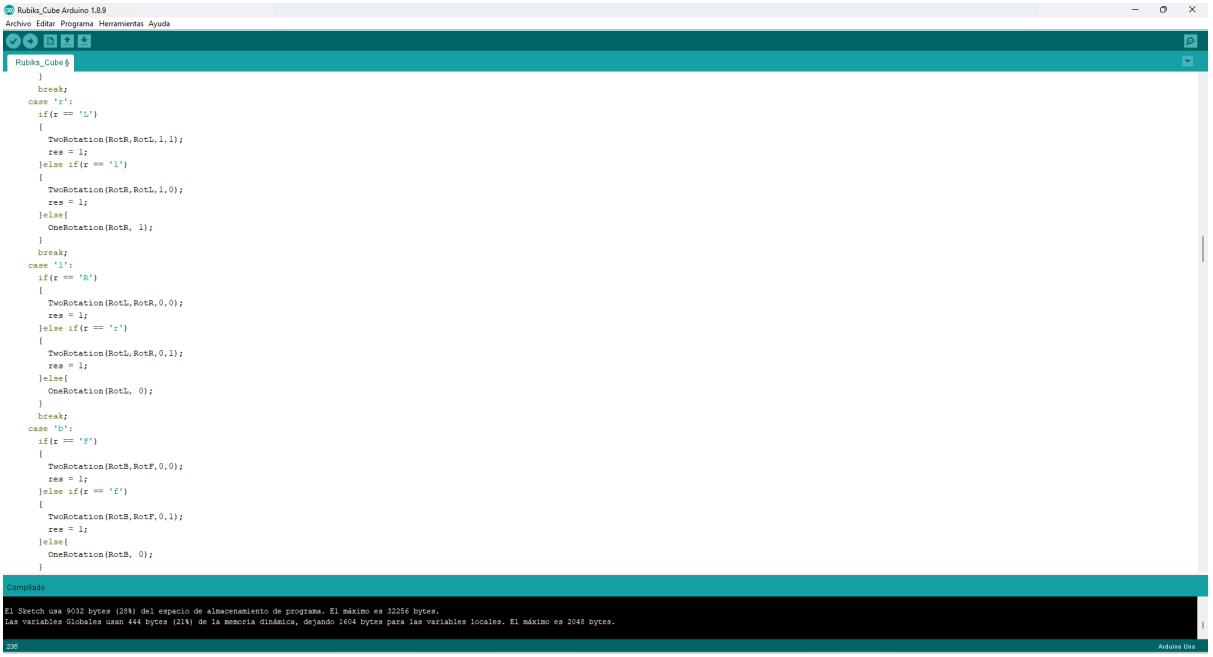
Imatge 56: Programa sencer. Font pròpia



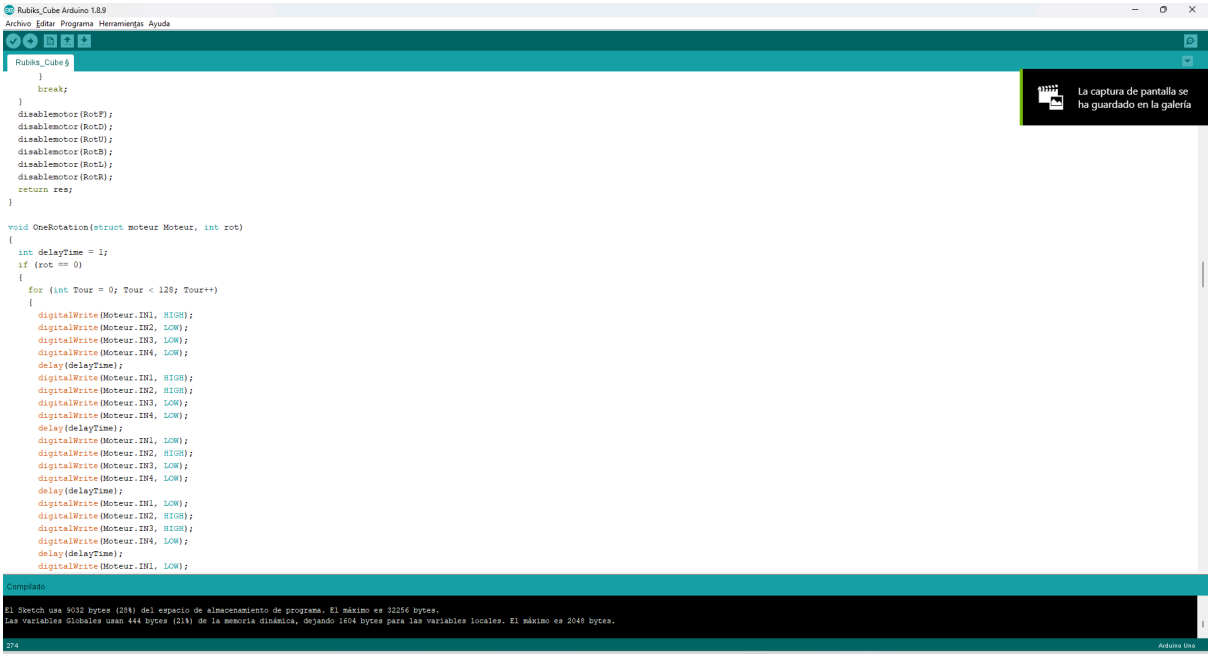
Imatge 57: Programa sencer. Font pròpia



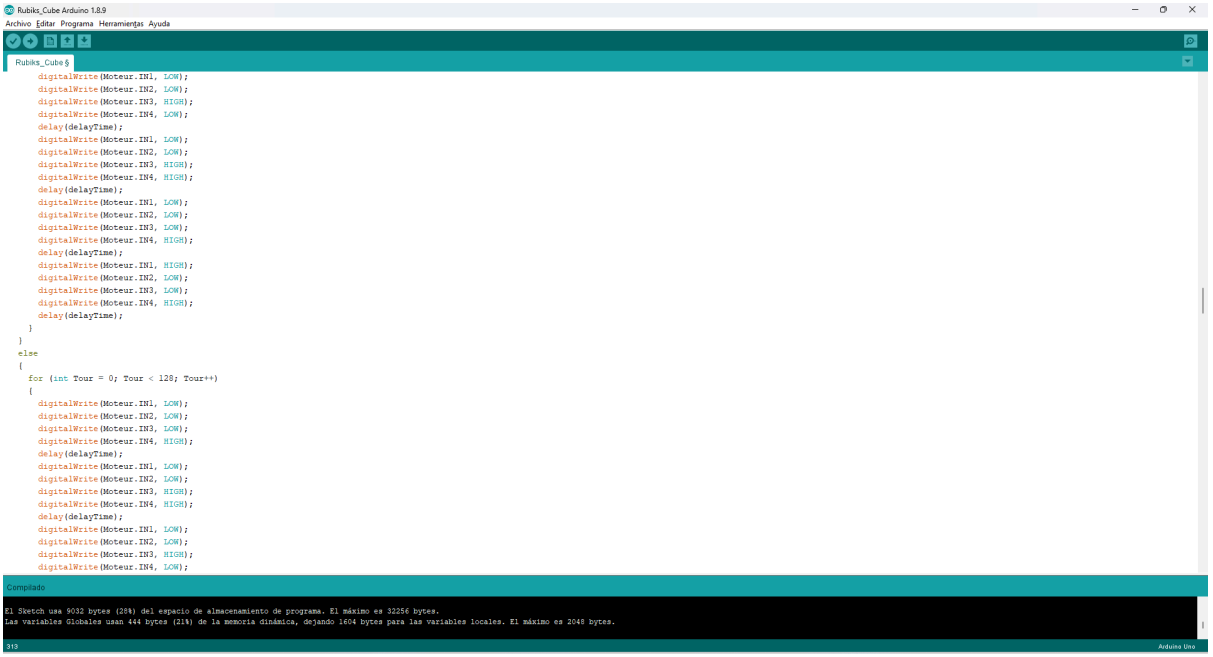
Imatge 58: Programa sencer. Font pròpia



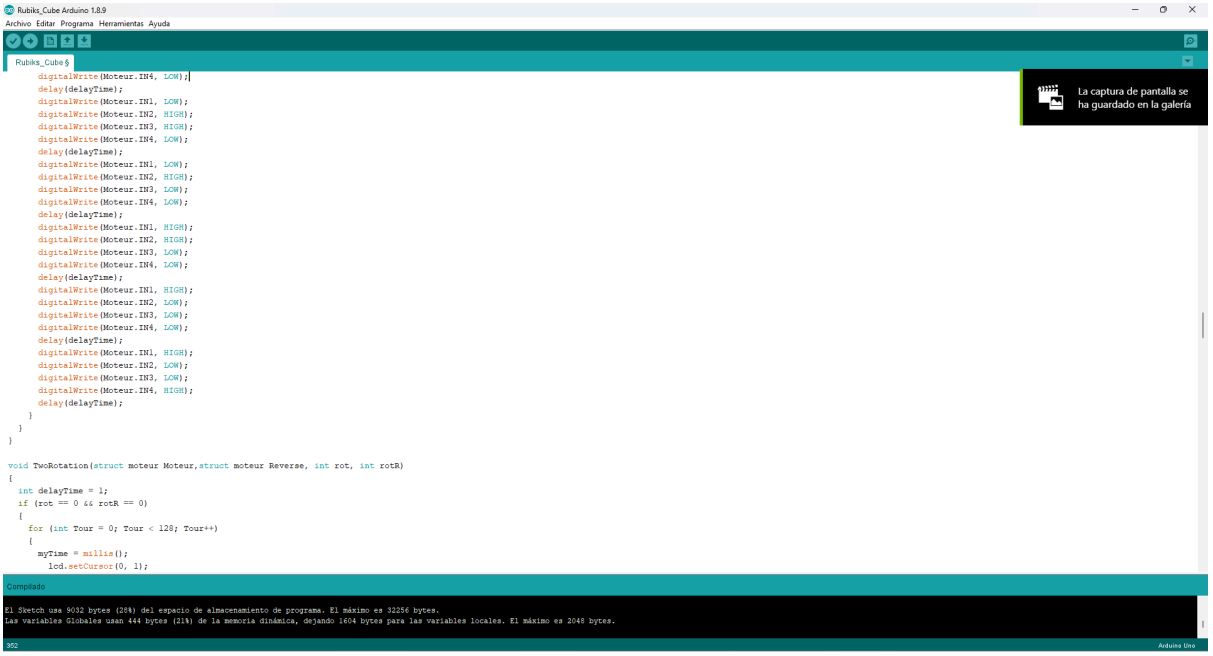
Imatge 59: Programa sencer. Font pròpia



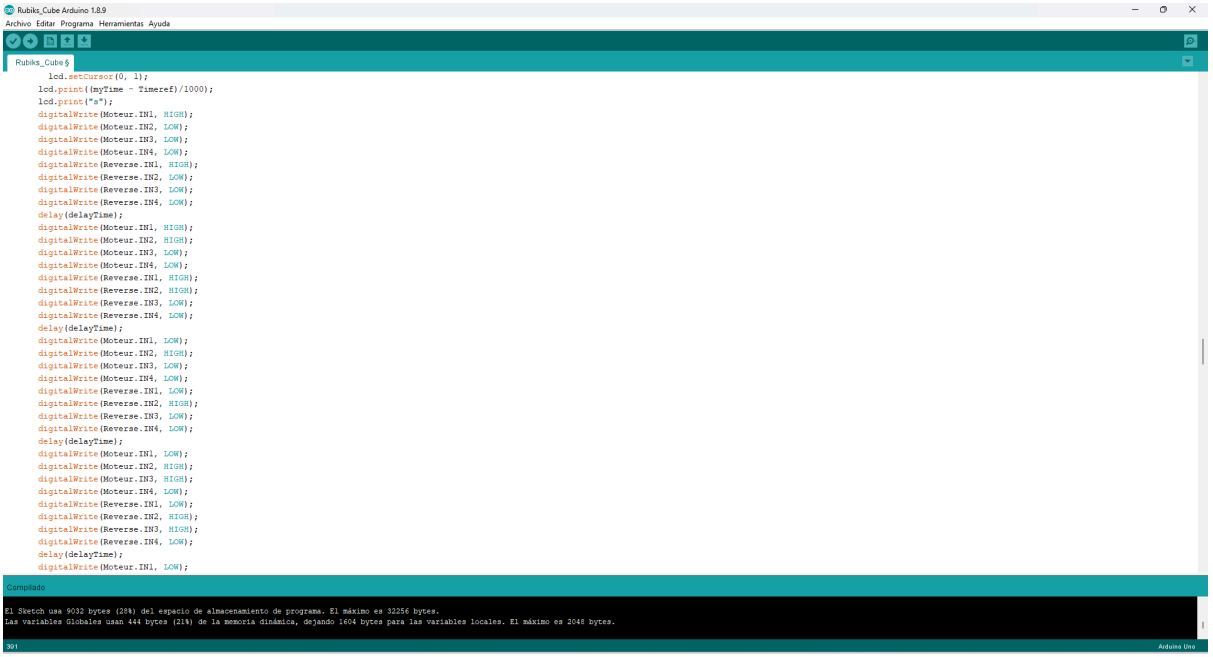
Imatge 60: Programa sencer. Font pròpia



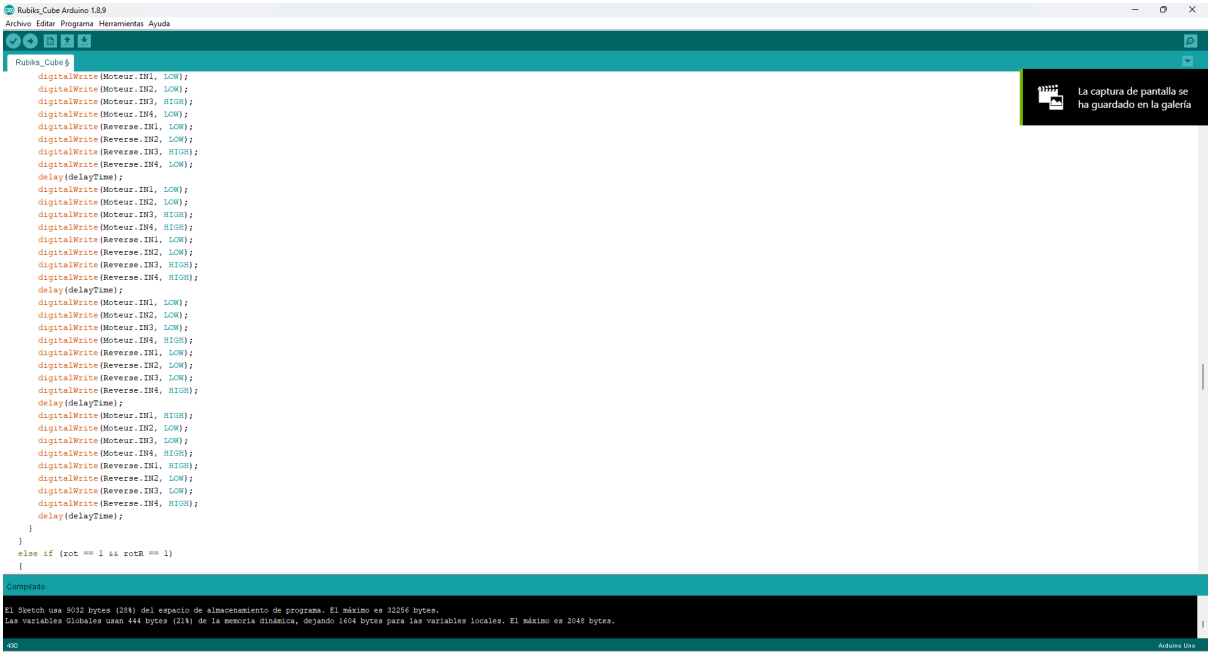
Imatge 61: Programa sencer. Font pròpia



Imatge 62: Programa sencer. Font pròpia

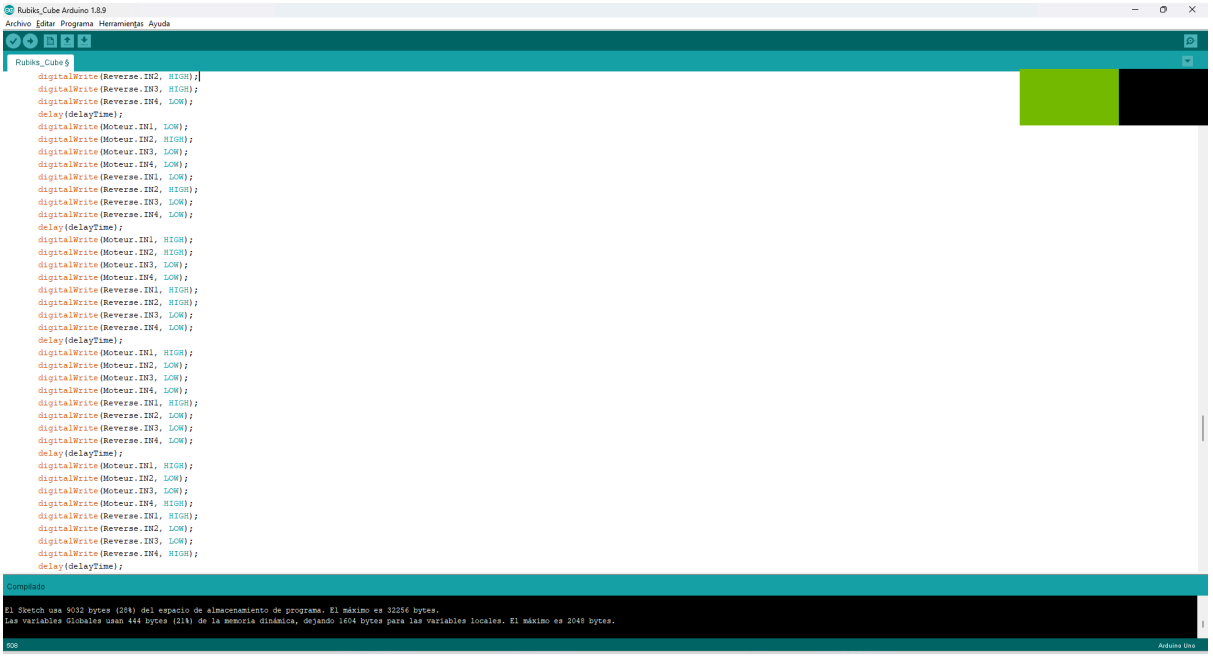


Imatge 63: Programa sencer. Font pròpia



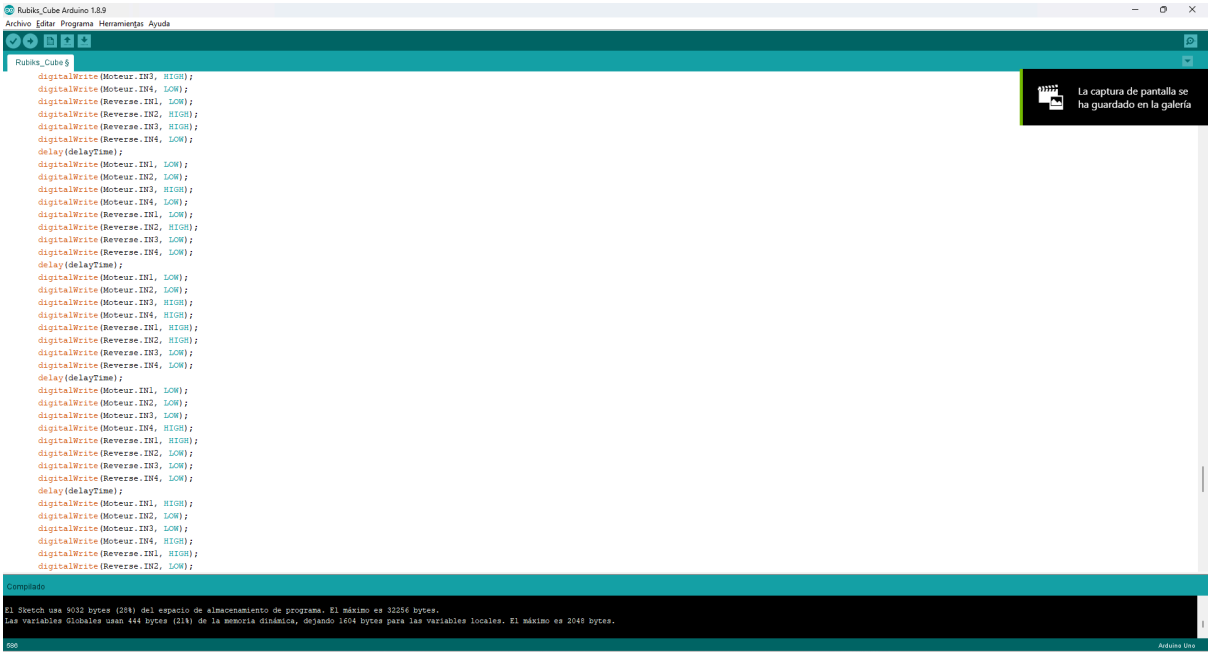
Imatge 64: Programa sencer. Font pròpia

Imatge 65: Programa sencer. Font pròpia

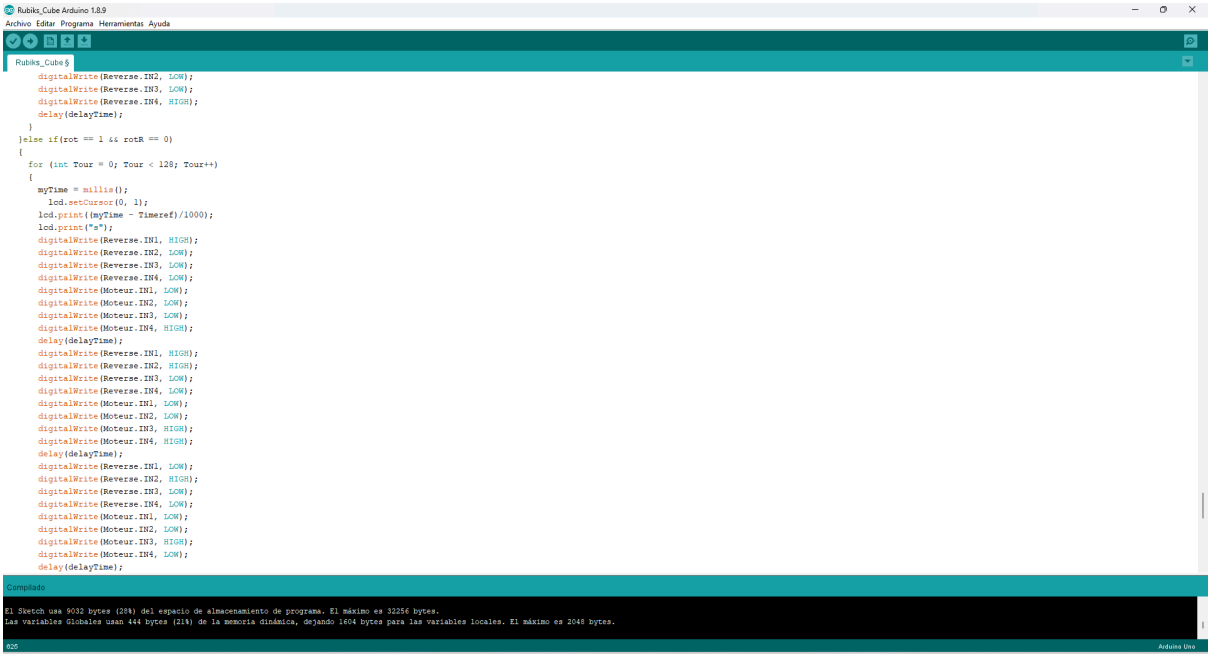


Imatge 66: Programa sencer. Font pròpia

Imatge 67: Programa sencer. Font pròpia



Imatge 68: Programa sencer. Font pròpia



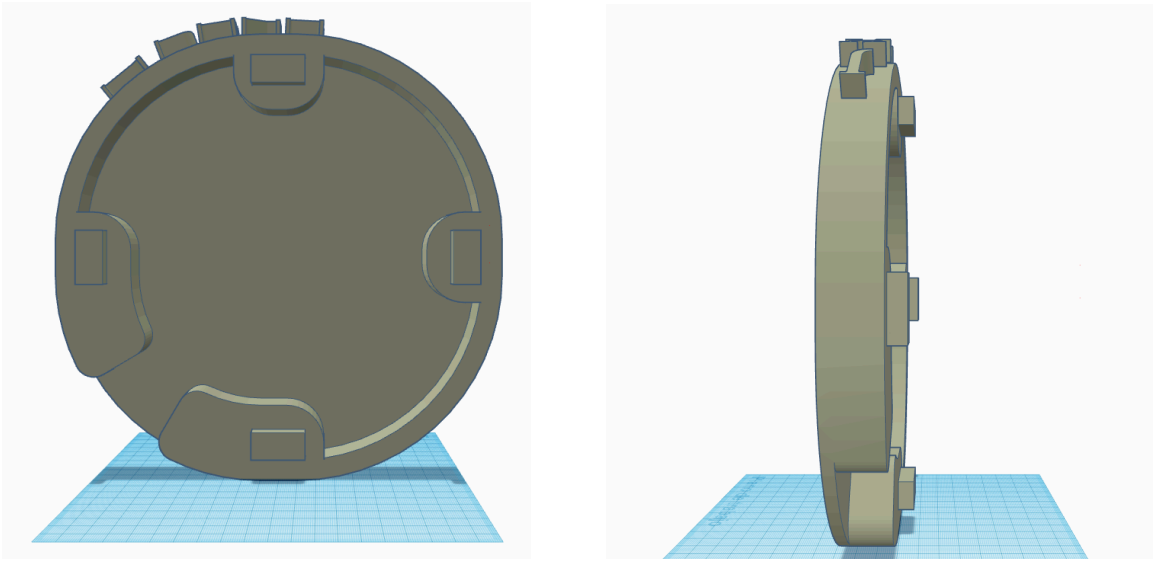
Imatge 69: Programa sencer. Font pròpia

Imatge 70: Programa sencer. Font pròpia

Imatge 71: Programa sencer. Font pròpia

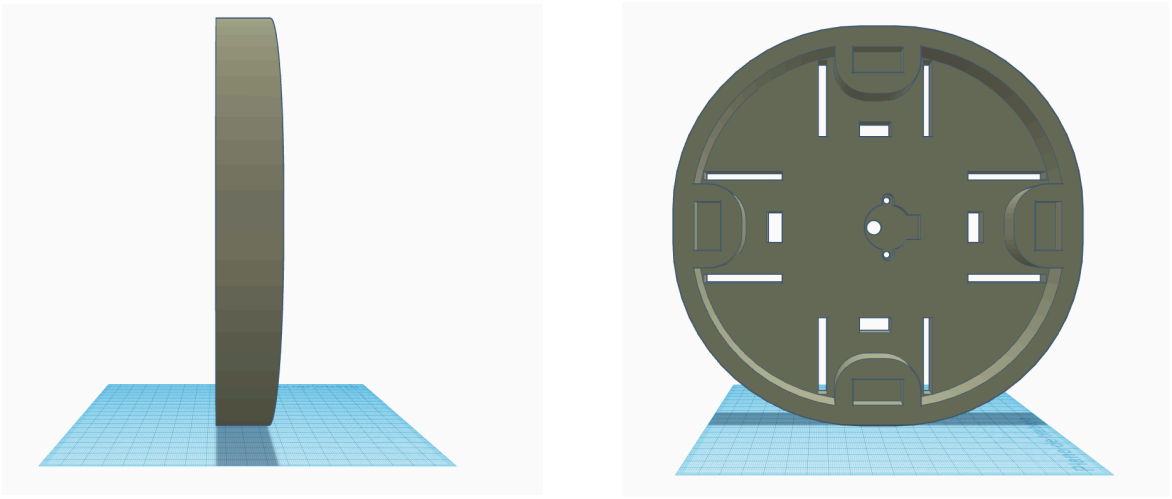
8.2 Edició de les peces.

Base inferior A:



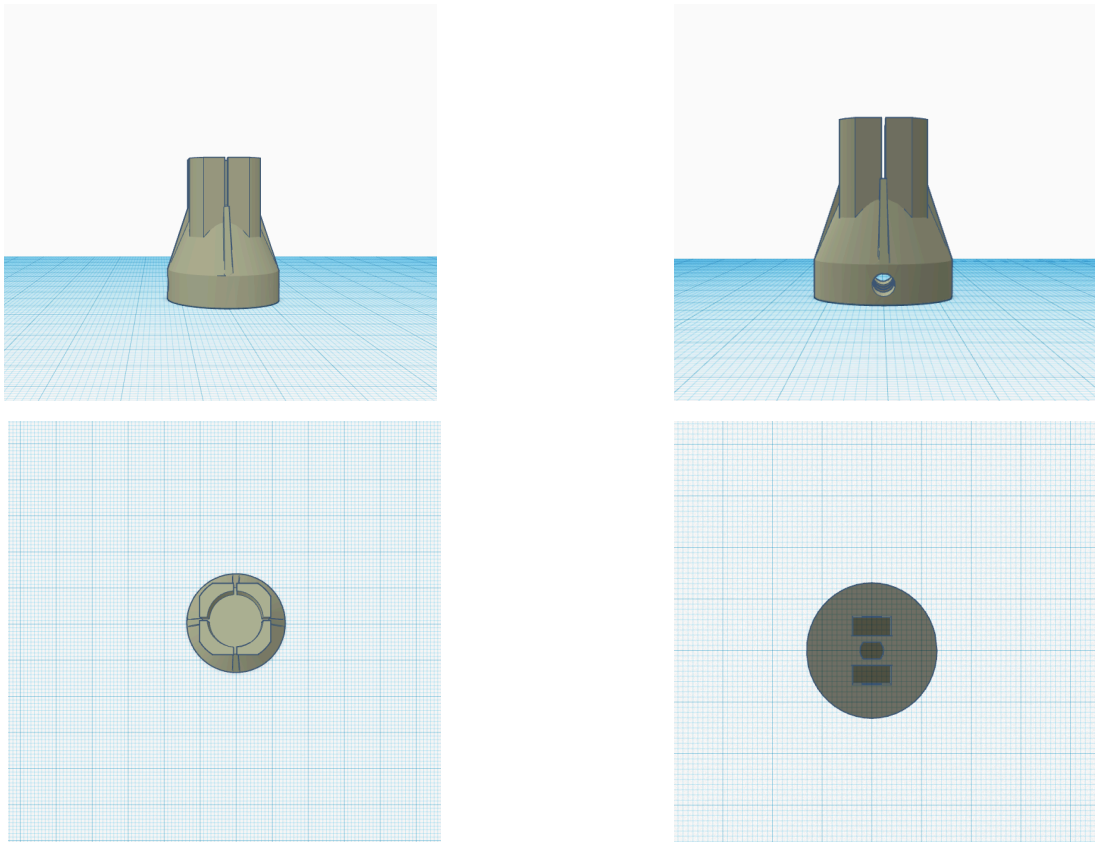
Imatges 72: Tinkercad. Font pròpia

Base inferior B:



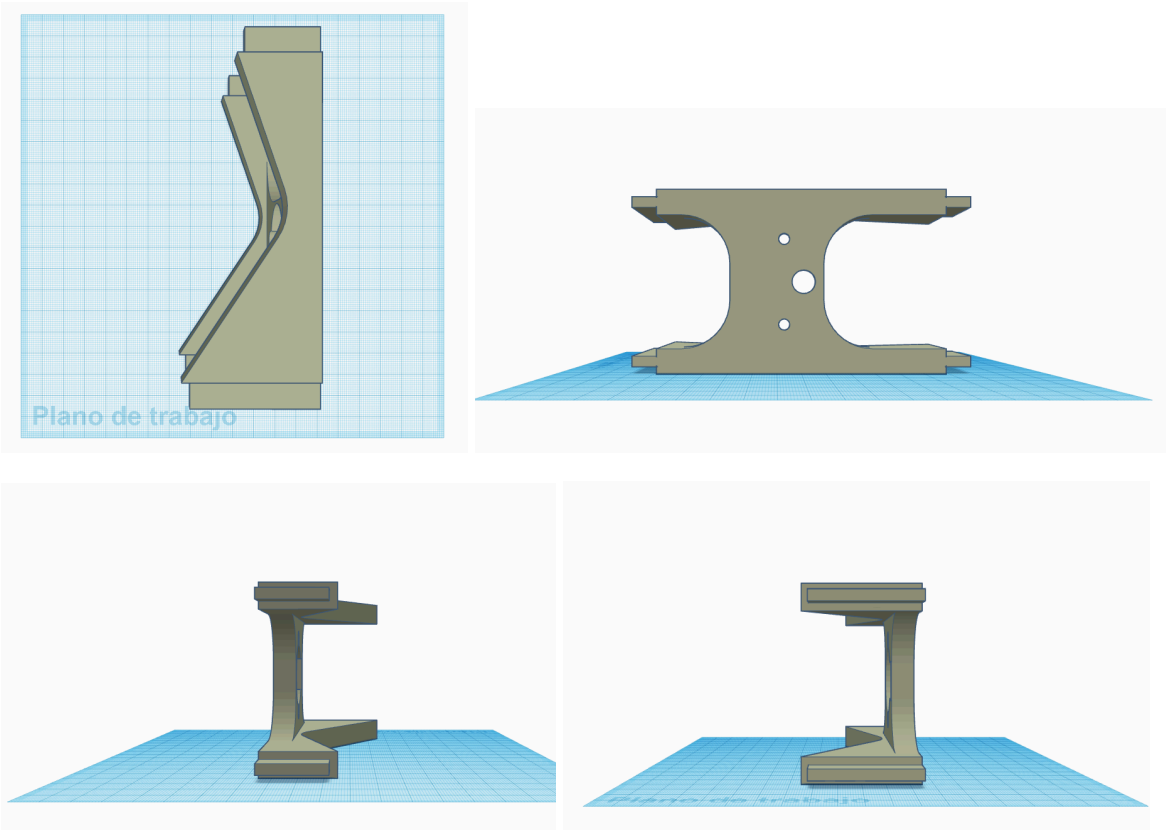
Imatges 73: Tinkercad. Font pròpia

Conector:



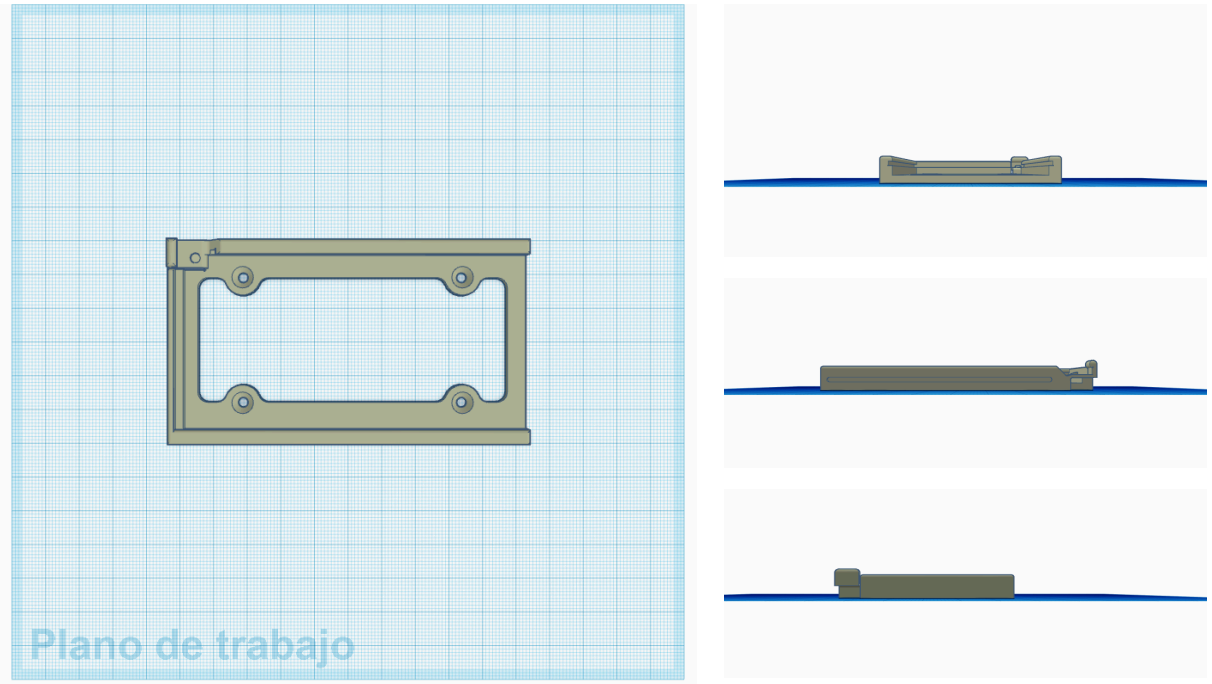
Imatges 74: Tinkercad. Font pròpia

Costat:



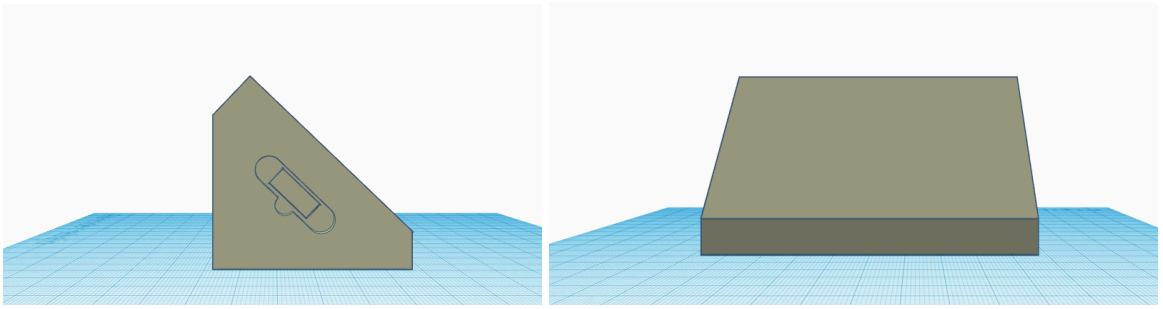
Imatges 75: Tinkercad. Font pròpia

Suport de placa Arduino:

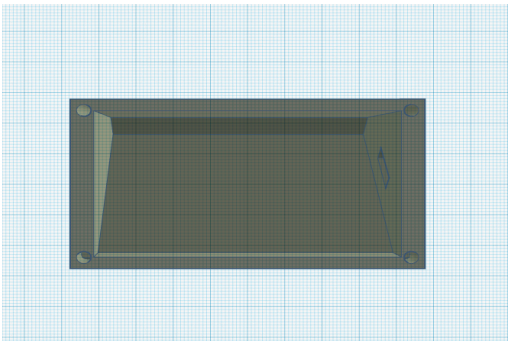


Imatges 76: Tinkercad. Font pròpia

Base de la pantalla:

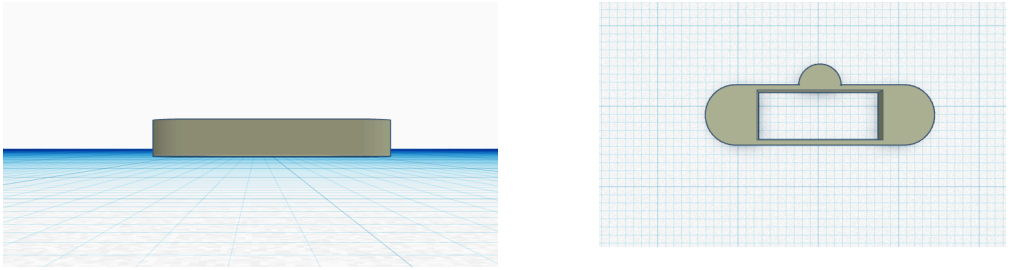


Imatges 77: Tinkercad. Font pròpia



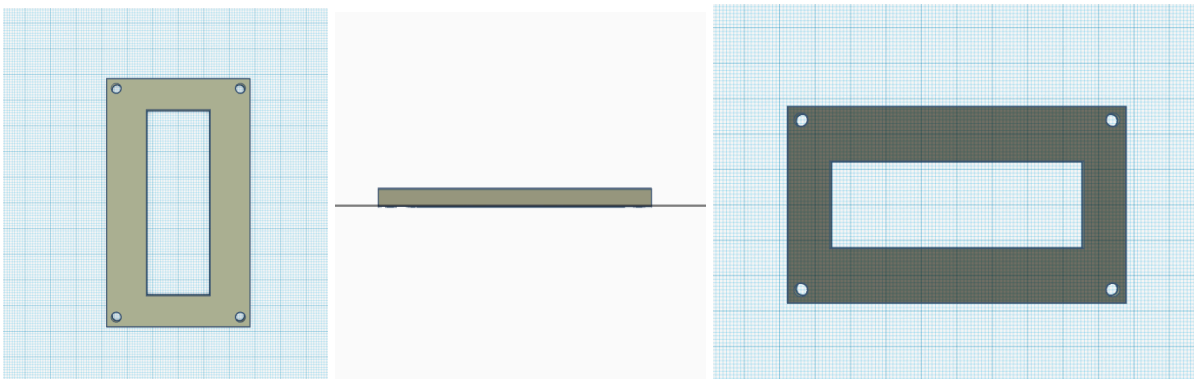
Imatge 78: Tinkercad. Font pròpia

Encaix de la base de la pantalla:



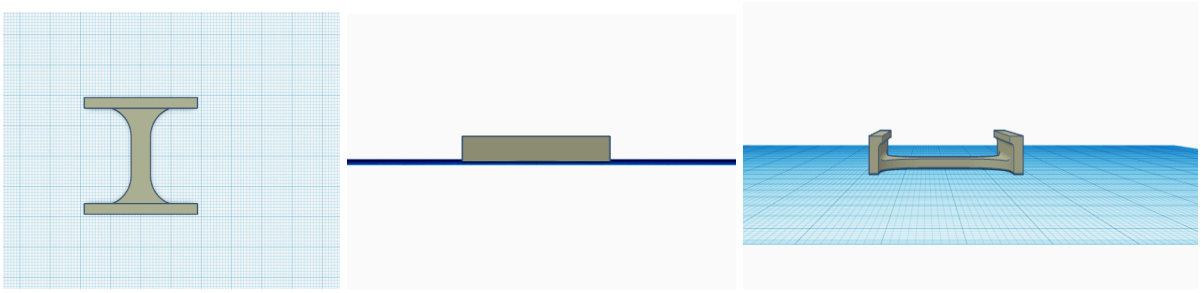
Imatges 79: Tinkercad. Font pròpia

Contorn de la pantalla:



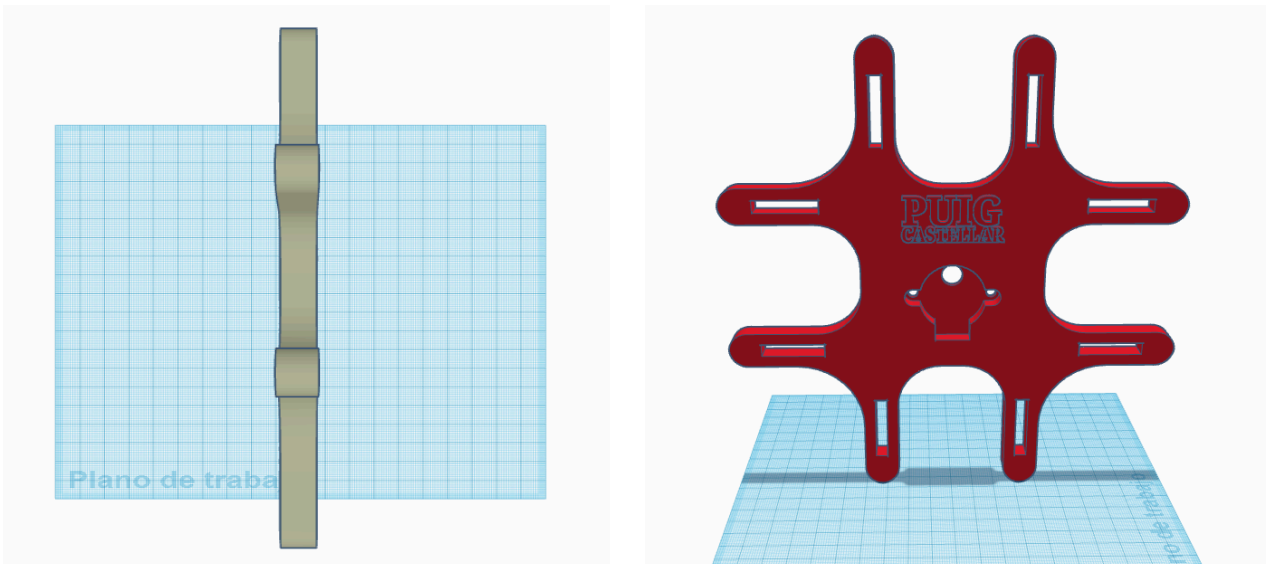
Imatges 80: Tinkercad. Font pròpia

Suport dels controladors dels motors:



Imatges 81: Tinkercad. Font pròpia

Base superior:



Imatges 82: Tinkercad. Font pròpia

INSTITUT
PUIG CASTELLAR