

AUTOR: CHRISTIAN CUBAS JARAMILLO

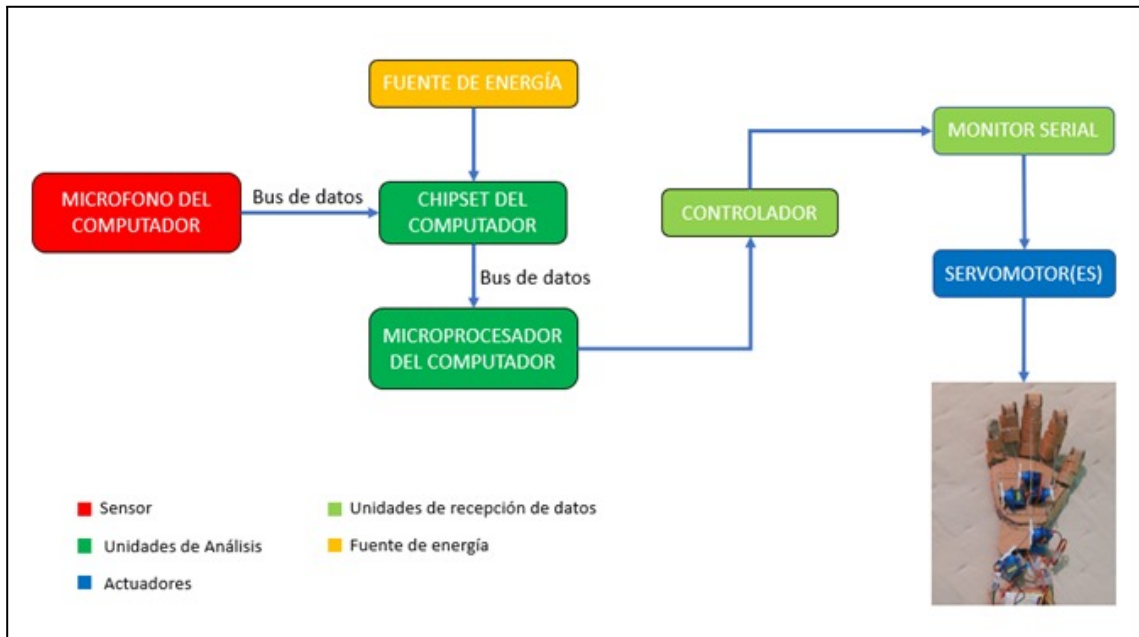
Brazo robotico controlado mediante comando de voz

La propuesta consta de un brazo robótico controlado por comandos de voz para esto se aplicó diferentes conocimientos para poder dar con su desarrollo y posterior implementación, pues durante el desarrollo se consideraron fases como modelado de piezas, planificación para la fabricación del brazo, conocimiento básico y claves de electrónica básica, entre otras más.

Cronograma de Trabajo

[illegible]

DESARROLLO DEL ALGORITMO



Breve explicación:

El brazo robótico desarrollado aplica los conceptos de sensores y actuadores algunos de forma indirecta; es decir, en este caso se hace de uso la librería de reconocimiento de voz quien hace uso de micrófono de la computadora para poder analizar e identificar el mensaje por voz realizado, este micrófono es nuestro sensor indirectamente este al igual que los sensores sirven como entrada de datos del exterior en este caso de nuestra voz este micrófono como el periférico que es sirve de entrada de datos quien a su vez usa el bus de datos para transmitir los datos recibidos del micrófono al programa quien a su vez es analizado por la librería reconocimiento de voz que sirve como una especie de filtro para interpretar el significado del comando en base a esto el programa tiene programada una serie de condicionales las cuales a su vez mediante pyserial emiten un mensaje al monitor serial de arduino quien finalmente al recibir el valor enviado por python mediante pyserial y lo que hace es que en base al valor recibido por el puerto serial realiza una determinada acción (mover mediante el servomotor(actuador) un determinado dedo o un conjunto de ellos).

Resumen de lo que ocurre:

micrófono del computador (recibe datos) --> reconocimiento de voz (analiza el mensaje emitido por voz)-->pyserial(envia mediante monitor serie la instruccion a realizar)--> arduino (recibe el valor del monitor serie, la evalúa y ejecuta la instrucción correspondiente) Fin

Cita:<https://github.com/ChristianCubas/Brazo-rob-tico-controlado-por-comandos-de-voz>

DESARROLLO Y MADURACIÓN DEL HARDWARE

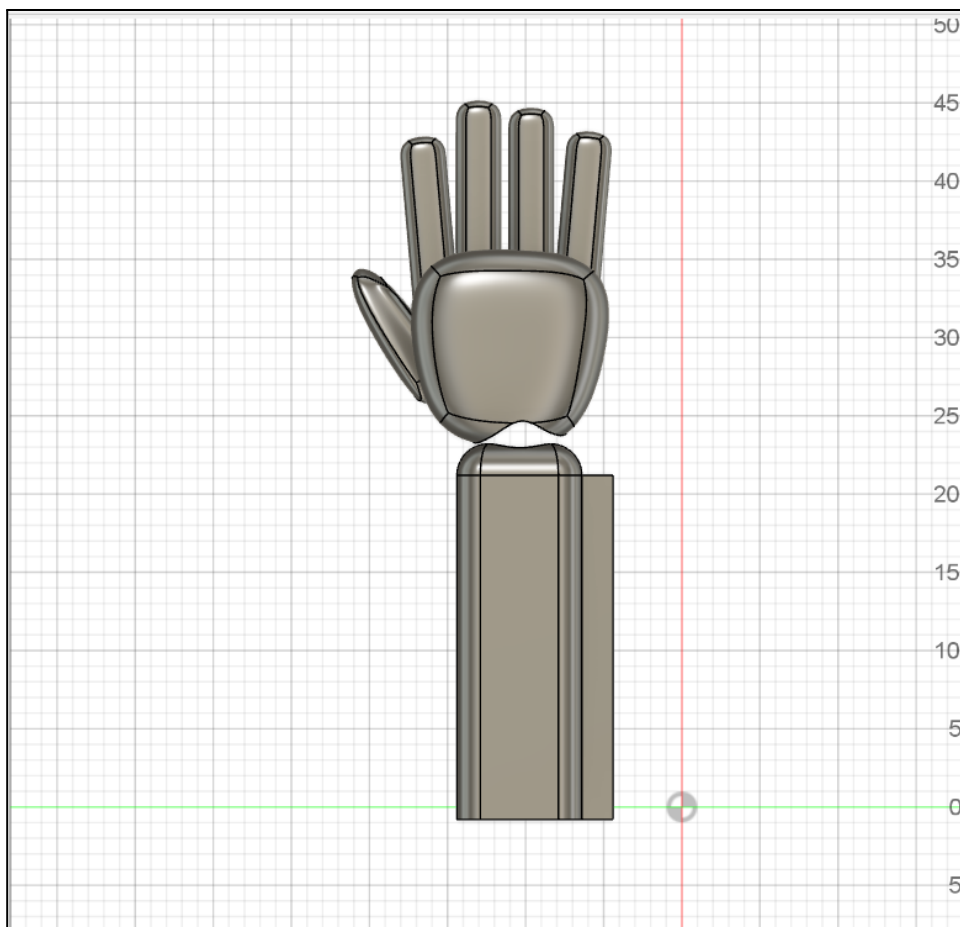
A) Desarrollo de partes:

Paso previo: Configurar la herramienta SolidWorks en cm como unidad de medida

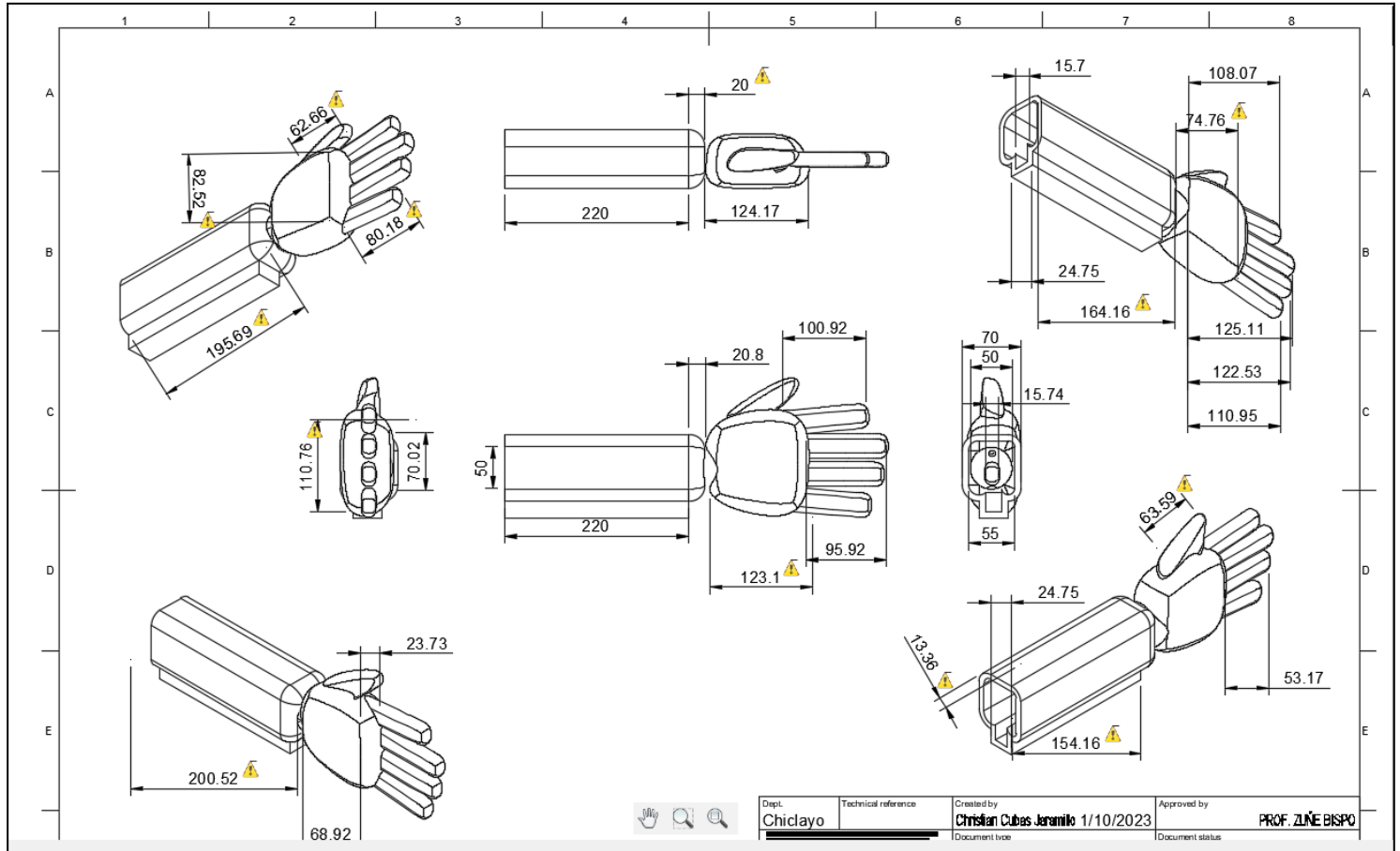
En esta etapa se sentaron las bases del hardware a utilizar y a su vez se desarrolló, cálculo y planificó cada una de las medidas a utilizar tomando en cuenta aspectos importantes como: las dimensiones a ocupar, los vacíos interiores destinados a ser ocupados por la circuitería propia de la prótesis robótica, el modelamiento de partes y la estructura general necesaria para poder maquetar y fabricar el brazo robótico.

Asimismo, cabe destacar que se siguió y cumplió con los estándares de calidad tomando como referencias la normativa ISO para el desarrollo de cada una de las partes.

En primer lugar se desarrolló el modelado del brazo robótico, siendo la herramienta usada SolidWorks. De igual manera, se establece las dimensiones en los tres ejes de cada una de las partes, el modelamiento de cada una de ellas, y las extrusiones internas destinadas para el cableado electrónico propia del proyecto



Posterior al modelaje y ensamblaje de cada una de las partes, consecuentemente se procedió a realizar el proceso de planificación para posteriormente poder implementarlo en la vida real mediante su desarrollo en físico.

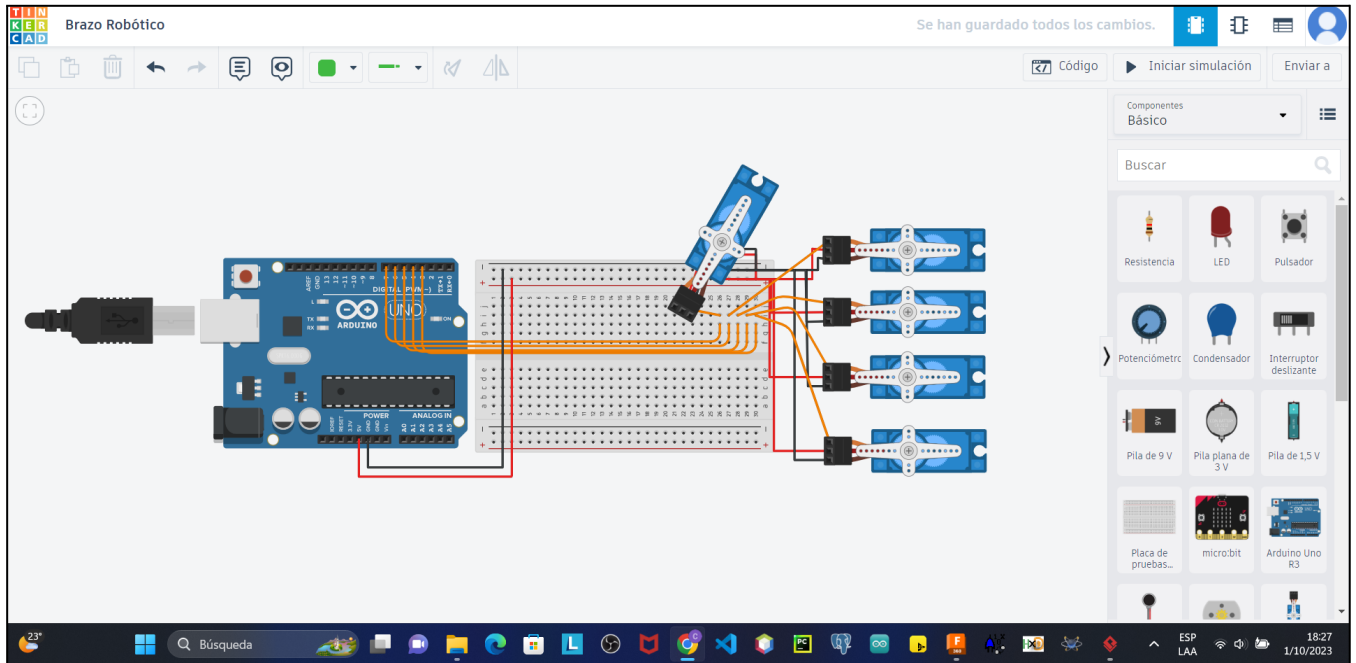


Tras el desarrollo de la planificación se aprecia cómo se activan los focos de advertencia esto debido a la reasociación; es decir, se pudo trabajar de forma más precisa usando un desarrollo desde un punto interior y partir de ahí generando asociaciones; sin embargo, debido a que no se trata de un proyecto de alta precisión se optó por considerar las medidas más relevantes para su desarrollo. Finalmente se procedió a imprimir el plano para basados en las medidas establecidas poder crearlo en físico.

B) Desarrollo electrónico:

En esta etapa se buscó sentar los componentes y cables a utilizar, siendo los materiales a usar: una protoboard, para la interconexión de cables; una placa arduino UNO Rev3, como placa de desarrollo y transmisor de instrucciones mediante su procesador Atmel; 5 servomotores, para mediante su torque generar el inclinar/levantar (180°) y regresar a la normalidad (0°) los dedos de la mano robótica.

A continuación, se visualiza la interconexión realizada entre los materiales anteriormente mencionados para el desarrollo electrónico del brazo robótico. Asimismo, en la parte inferior se detalla el paso a paso realizado.



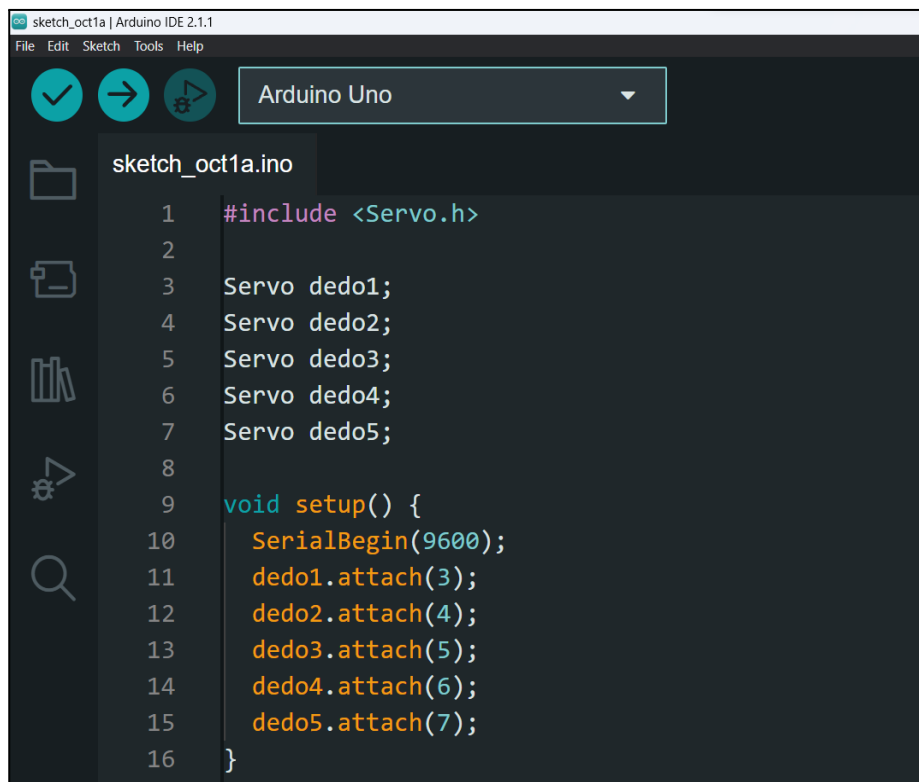
Pasos realizados:

- 1) Proveer de corriente y voltaje a la protoboard esto se realizó mediante un cable rojo(voltaje) conectando la entrada de 5V de Arduino con la región positiva de la protoboard, región señalizada bajo el color rojo y el símbolo (+).
- 2) Establecer conexión con tierra para evitar fugas de corriente; esto se realizó mediante la conexión entre la entrada GND de arduino y la región negativa de la protoboard señalizada bajo el color negro o azul con su característico símbolo (-).
- 3) Posterior a eso se realizó conexión entre los servomotores y la protoboard ya energizada, para esto se conectó la entrada de corriente del servomotor con la región (+) de la protoboard. Seguidamente, se conectó la entrada marrón del servomotor con la región negativa (-) de la protoboard. Posterior a esto se establece la conexión y transmisión de señales e instrucciones con Arduino mediante el cable anaranjado simbolizador de señal.
- 4) Continuando la secuencia se debe conectar la entrada de señal del servomotor con un orificio de la protoboard; posteriormente en otro orificio paralelo a donde se colocó el anterior se conectará al arduino (se realiza un puenteo servo-protoboard, protoboard-arduino)
- 5) Tras haber realizado el mismo proceso con cada uno de los servomotores finalmente se debe probar el circuito para esto lo puede hacer dando instrucciones desde el arduino IDE o haciendo uso de Tinkercad.

DESARROLLO E IMPLEMENTACIÓN DE SOFTWARE

A continuación se explicará a detalle, a nivel de programación los aspectos técnicos y configuraciones realizadas. Dando inicio al desarrollo de esta fase resulta de gran ayuda explicar que Arduino, como placa de desarrollo, cuenta con IDE de programación llamado Arduino IDE; este a su vez contiene diferentes librerías ya programadas las cuales permiten el desarrollo sencillo en componentes electrónicos. Dentro de estas librerías, una de las más importantes es el servomotor.

¿Qué es el servomotor? Bueno para entender esto se necesita remontar a los años del inventor Faraday quién con sus aportaciones al electromagnetismo permitió desarrollar en años posteriores el motor y con ello evolucionar a los servomotores que no son más que motores inteligentes los cuales se pueden controlar y programar. Para esto, retomando el desarrollo del brazo se hará uso de la librería <Servo.h> para poder declarar, inicializar e indicar el giro a realizar por cada uno de los servomotores.



```
sketch_oct1a | Arduino IDE 2.1.1
File Edit Sketch Tools Help
[Checkmark] [Next] [Upload] Arduino Uno
sketch_oct1a.ino
1  #include <Servo.h>
2
3  Servo dedo1;
4  Servo dedo2;
5  Servo dedo3;
6  Servo dedo4;
7  Servo dedo5;
8
9  void setup() {
10     SerialBegin(9600);
11     dedo1.attach(3);
12     dedo2.attach(4);
13     dedo3.attach(5);
14     dedo4.attach(6);
15     dedo5.attach(7);
16 }
```

En la siguiente captura se muestra como importar la librería <Servo.h>, cómo declarar variables de tipo servomotor (dedo1, dedo2, dedo3, dedo4 y dedo5) y como inicializarlas cada una con su respectivo puerto Arduino, puertos 3,4,5,6 y 7 respectivamente.

```

18 void loop() {
19     dedo1.write(180);
20     dedo2.write(180);
21     dedo3.write(180);
22     dedo4.write(0);
23     dedo5.write(180);
24     Serial.println("A continuacion ingrese la instruccion a realizar");
25     String instruccion = Serial.readStringUntil('\n');
26     if (instruccion == "dedo 1") {
27         dedo1.write(0);
28         delay(1000);
29         dedo1.write(180);
30         delay(1000);
31     } else if (instruccion == "dedo 2") {
32         dedo2.write(0);
33         delay(1000);
34         dedo2.write(180);
35         delay(1000);

```

```

31     } else if (instruccion == "dedo 2") {
32         dedo2.write(0);
33         delay(1000);
34         dedo2.write(180);
35         delay(1000);
36     } else if (instruccion == "dedo 3") {
37         dedo3.write(0);
38         delay(1000);
39         dedo3.write(180);
40         delay(1000);
41     } else if (instruccion == "dedo 4") {
42         dedo4.write(0);
43         delay(1000);
44         dedo4.write(180);
45         delay(1000);
46     } else if (instruccion == "dedo 5") {
47         dedo5.write(0);
48         delay(1000);

```

```
51 } else if (instruccion == "contraer-puño") {
52     delay(1000);
53     dedo1.write(0);
54     dedo2.write(0);
55     dedo3.write(0);
56     dedo4.write(0);
57     dedo5.write(0);
58     delay(1000);
59     dedo1.write(180);
60     dedo2.write(180);
61     dedo3.write(180);
62     dedo4.write(180);
63     dedo5.write(180);
64     delay(1000);
65 } else if (instruccion == "descontraer-palma") {
66     delay(1000);
67     dedo1.write(180);
68     dedo2.write(180);
69     dedo3.write(180);
70     dedo4.write(180);
71     dedo5.write(180);
72     delay(1000);
73     dedo1.write(0);
74     dedo2.write(0);
75     dedo3.write(0);
76     dedo4.write(0);
77     dedo5.write(0);
78     delay(1000);
79 } else {
```



```
79     } else {  
80         Serial.println("Aun no tenemos esa funcionalidad, estamos trabajando en ello :");  
81         delay(2000);  
82     }  
83 }
```

Lo realizado en las capturas posteriores fue:

- 1) Recibir un valor de tipo texto mediante el puerto serial para esto se hizo uso de `Serial.readStringUntil('\n')` y el valor recibido lo almacenamos en la variable instrucción.
- 2) posterior a eso se evalúa el contenido de la variable instrucción y en base a su contenido mediante el uso de condicionales se busco realizar ciertas acciones por parte de los actuadores (servomotores)

Ahora cabe destacar que hasta este punto el brazo solo respondería a lo que le escribamos por medio del puerto serial. Pero como se especificó en un inicio este proyecto trata sobre como generar un brazo robótico que sea controlado mediante comando de voz para esto se hará uso Python, lenguaje de programación, que por medio de dos librerías suyas `speech recognition` y `pyserial` se podrá controlar por medio de comandos de voz lo cuales serán recibido por la computadora

```
File Edit Selection View Go Run ... BLACKPINK - KISS AND MA...
main.py x docs.google.com/document/d/1vup0UPY1TUyC48JUTd4R1L_-Tck57Mq4np1VYWd5rfc/edit
C:\Users\CHRISTIAN CUBAS J\Downloads\Encontrar\Encontrar\main.py > ...

1 # Importando librerías a usar
2 import serial
3 import pytttsx3
4 import time
5 import speech_recognition as sr
6 import webbrowser as wb
7 import tkinter as tk
8 from tkinter import ttk,messagebox
9 from pdb import set_trace as breakpoint
10
11 #Nombre de la persona
12 dueño = "Christian"
13
14 # Solicitando hora mediante python
15 hora = time.strftime("%H")
16 hora = int(hora)
17
18 #Llamamos a Arduino
19 arduino = serial.Serial('COM3',9600)
20
21 # Condicional para determinar si es día,tarde o noche
22 if hora <= 12:
23     hour = "buenos días"
24 elif hora > 12 and hora <= 19:
25     hour = "buenas tardes"
26 elif hora > 19:
27     hour = "buenas noches"
28
```

```
29 #Saludo de bienvenida
30 engine = pytttsx3.init()
31 engine.say("Hola"+str(dueño)+" , muy "+ hour)
32 engine.say("Preparado y listo para recibir instrucciones señor")
33 x = engine.runAndWait()
34 print(x)
35 # Ejecucion y proceso de los comandos de voz realizados
36 r = sr.Recognizer()
37
38 i = 1
```

```
41 while i>=1:
42     with sr.Microphone() as source:
43         audio = r.listen(source)
44     try:
45         text = r.recognize_google(audio, language="es")
46         print(text)
47         #Ejecuta la instruccion dad mediante voz
48         if text=="dedo 1" or text=="D1" or text=="d1" or text=="uno" or text=="1":
49             pytttsx3.speak("Moviendo dedo uno")
50             arduino.write(b"dedo 1")
51         elif text=="dedo 2" or text=="D2" or text=="d2" or text=="dos" or text=="2" or text=="no" or text=="d23" or text=="22" or te
52             pytttsx3.speak("Moviendo dedo dos")
53             arduino.write(b"dedo 2")
54         elif text=="dedo 3" or text=="D3" or text=="d3" or text=="tres" or text=="3" or text=="malcriado" or text=="33":
55             pytttsx3.speak("Moviendo dedo tres")
56             arduino.write(b"dedo 3")
57         elif text=="dedo 4" or text=="D4" or text=="d4" or text=="cuatro" or text=="4":
58             pytttsx3.speak("Moviendo dedo cuatro")
59             arduino.write(b"dedo 4")
60         elif text=="dedo 5" or text=="D5" or text=="d5" or text=="cinco" or text=="5":
61             pytttsx3.speak("Moviendo dedo cinco")
62             arduino.write(b"dedo 5")
63         elif text=="puño" or text=="puño puño":
64             pytttsx3.speak("Generando puño")
65             arduino.write("contraer-puño".encode())
66         elif text=="detener programa":
67             pytttsx3.speak("Deteniendo el programa")
68             i-=1
69             break
70     except:
71         pytttsx3.speak("No te puede entender")
72         i+=1
73         continue
74     arduino.close()
75     DESARROLLO FÍSICO E IMPLEMENTACIÓN DEL PROYECTO
```

```
41 while i>=1:
42     with sr.Microphone() as source:
43         audio = r.listen(source)
44     try:
45         text = r.recognize_google(audio, language="es")
46         print(text)
47         #Ejecuta la instruccion dad mediante voz
48         if text=="dedo 1" or text=="D1" or text=="d1" or text=="uno" or text=="1":
49             pytttsx3.speak("Moviendo dedo uno")
50             arduino.write(b"dedo 1")
51         elif text=="dedo 2" or text=="D2" or text=="d2" or text=="dos" or text=="2" or text=="no" or text=="d23" or text=="22" or te
52             pytttsx3.speak("Moviendo dedo dos")
53             arduino.write(b"dedo 2")
54         elif text=="dedo 3" or text=="D3" or text=="d3" or text=="tres" or text=="3" or text=="malcriado" or text=="33":
55             pytttsx3.speak("Moviendo dedo tres")
56             arduino.write(b"dedo 3")
57         elif text=="dedo 4" or text=="D4" or text=="d4" or text=="cuatro" or text=="4":
58             pytttsx3.speak("Moviendo dedo cuatro")
59             arduino.write(b"dedo 4")
60         elif text=="dedo 5" or text=="D5" or text=="d5" or text=="cinco" or text=="5":
61             pytttsx3.speak("Moviendo dedo cinco")
62             arduino.write(b"dedo 5")
63         elif text=="puño" or text=="puño puño":
64             pytttsx3.speak("Generando puño")
65             arduino.write("contraer-puño".encode())
66         elif text=="detener programa":
67             pytttsx3.speak("Deteniendo el programa")
68             i-=1
69             break
70     except:
71         pytttsx3.speak("No te puede entender")
72         i+=1
73         continue
74     arduino.close()
75     DESARROLLO FÍSICO E IMPLEMENTACIÓN DEL PROYECTO
```

A continuación se presentó en capturas posteriores a la aclaración el código python utilizado para poder manipular el brazo robótico mediante comando de voz, para hacer el desarrollo de código más entendible se dejó escrito en comentario brevemente qué era lo que se iba a desarrollar líneas posteriores. Mayor detalle del código será brindado durante la exposición

DESARROLLO FÍSICO E IMPLEMENTACIÓN DEL PROYECTO

A continuación se detalla el proceso de construcción e implementación usando la documentación anteriormente generada para el desarrollo del brazo robótico. Para hacer de esta fase lo más detallado y preciso posible se optó por capturar en fotos el proceso paso a paso para una mayor comprensión del proceso de desarrollo del brazo robótico.

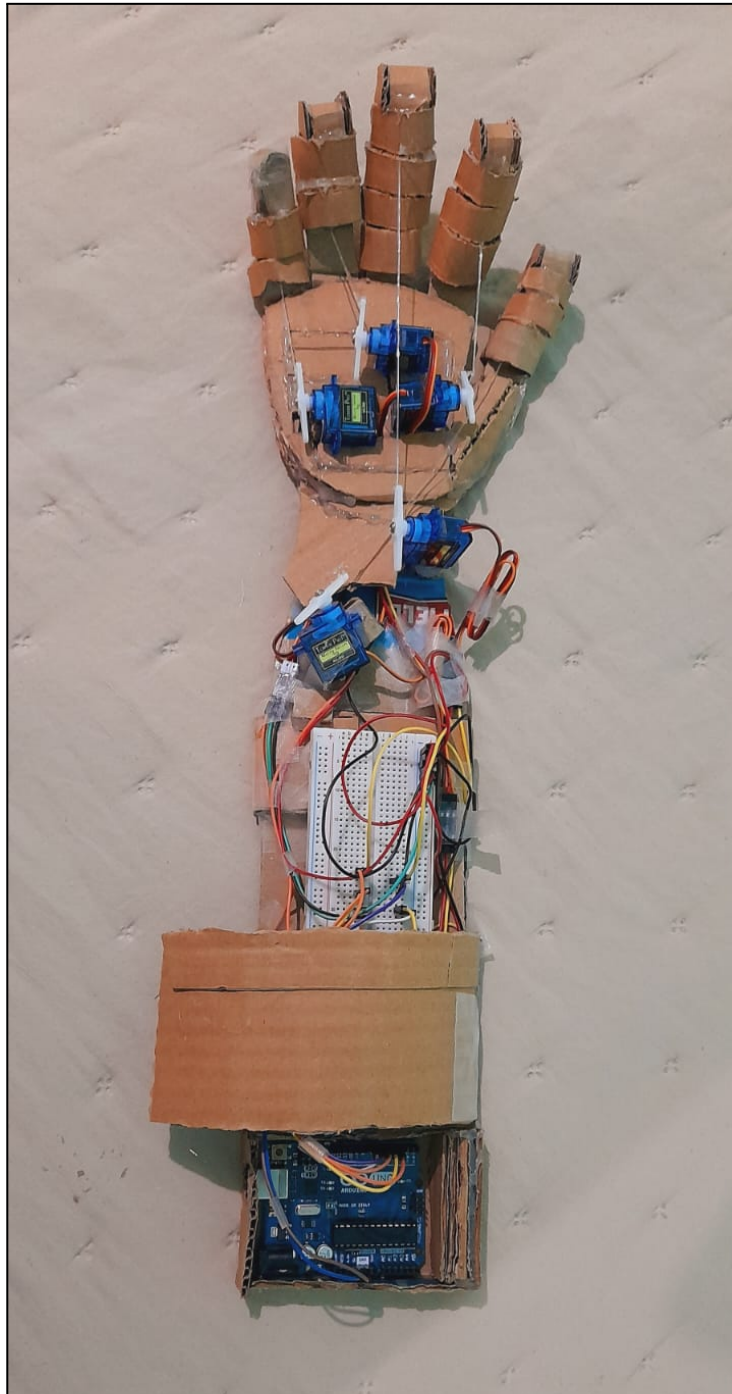


Imagen del brazo robótico próximo a realizar el cierre

CONCLUSIÓN

Tras la realización del proyecto se logró desarrollar las capacidades de: desarrollo y gestión de proyectos, modelamiento de partes, ensamblaje de partes, manipulación del entorno mediante actuadores, capacidad gestionar datos ingresantes por medio de sensores, desarrollo de circuitos electrónicos básicos, etc.

BIBLIOGRAFÍA Y REFERENCIA BIBLIOGRÁFICAS

<https://www.arduino.cc/>

<https://www.tinkercad.com/dashboard>

<https://pypi.org/project/pyserial/>

<https://cloud.google.com/speech-to-text>

<https://github.com/ChristianCubas/Brazo-rob-tico-controlado-por-comandos-de-voz>