



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA



Departamento de Ciencias de la Energía y Mecánica
Carrera de Ingeniería en Mecatrónica

Trabajo de titulación, previo a la obtención del título de:
Ingeniero en Mecatrónica

Implementación de un panel automático para mejorar los reflejos cognitivos y físicos de los deportistas de artes marciales mixtas, mediante el uso de una cámara 3D y un sistema embebido de Inteligencia Artificial en la empresa MIDY Scientific Trainers de la ciudad de Quito

Autor: Noroña Echeverría, Ricardo Sebastian

Director: MSc. Pérez Gutiérrez, Milton Fabricio

Latacunga, 2023



CONTENIDO

1. Introducción
2. Requerimientos de la empresa
3. Propuesta de diseño
4. Selección de componentes
5. Diseño mecánico
6. Programación
7. Validación de datos
8. Pruebas y Resultados
9. Conclusiones y recomendaciones

Introducción

El mundo del deporte y las MMA



Desarrollo de la velocidad de reacción (reflejos)

En un combate rápido y dinámico como los de MMA, la velocidad de reacción es crucial. Los reflejos le permiten al deportista anticipar y responder de manera instantánea a los ataques, defensas y movimientos del oponente.



Una fracción de segundo puede marcar la diferencia entre bloquear un golpe o recibirlo.

Objetivo General

Implementar un panel automático para mejorar los reflejos cognitivos y físicos de los deportistas de artes marciales mixtas, mediante el uso de una cámara 3D y un sistema embebido de Inteligencia Artificial en la empresa MIDY Scientific Trainers de la ciudad de Quito.

Objetivo Específicos

- Recopilar información acerca de los sistemas de entrenamiento automáticos utilizados en la mejora de los reflejos cognitivos y físicos de los deportistas de artes marciales mixtas, para identificar el funcionamiento del sistema.
- Analizar el funcionamiento del sistema de entrenamiento automático para fundamentar teóricamente el sistema y realizar el diseño de la implementación del mismo.
- Diseñar el panel de entrenamiento automático enfocado en la mejora de los reflejos cognitivos y físicos de los deportistas de artes marciales mixtas, sustentado su desarrollo en planos mecánicos, programación y control mediante visión artificial.
- Construir e implementar el panel de entrenamiento automático basando el proceso de implementación en los diseños obtenidos previamente.
- Realizar pruebas de funcionamiento en el sistema de entrenamiento automático desarrollado y corregir los posibles errores existentes.

Objetivo Específicos

- Evaluar los parámetros de entrenamiento iniciales (antes de utilizar el panel de entrenamiento automático) de un grupo reducido de artistas marciales, con un enfoque en sus tiempos de respuesta y velocidad de golpeo.
- Evaluar los parámetros de entrenamiento finales (después de haber utilizado el panel de entrenamiento automático) de un grupo reducido de artistas marciales, con un enfoque en sus tiempos de respuesta y velocidad de golpeo
- Realizar un análisis comparativo de los parámetros de entrenamiento de un grupo reducido de artistas marciales, antes y después de haber utilizado el panel de entrenamiento automático.

Hipótesis

La implementación de un panel de entrenamiento automático contribuirá en el desarrollo de los reflejos cognitivos y físicos de los deportistas de artes marciales mixtas, mediante el uso de una cámara 3D y un ordenador de inteligencia artificial embebido, en la empresa MIDY Scientific Trainers en la ciudad de Quito

Sistemas de entrenamiento de reflejos en el mercado



RIVAL REFLEX BALL



LUMMIC



BLAZEPOD



J&A AEROBICS

Requerimientos de la empresa



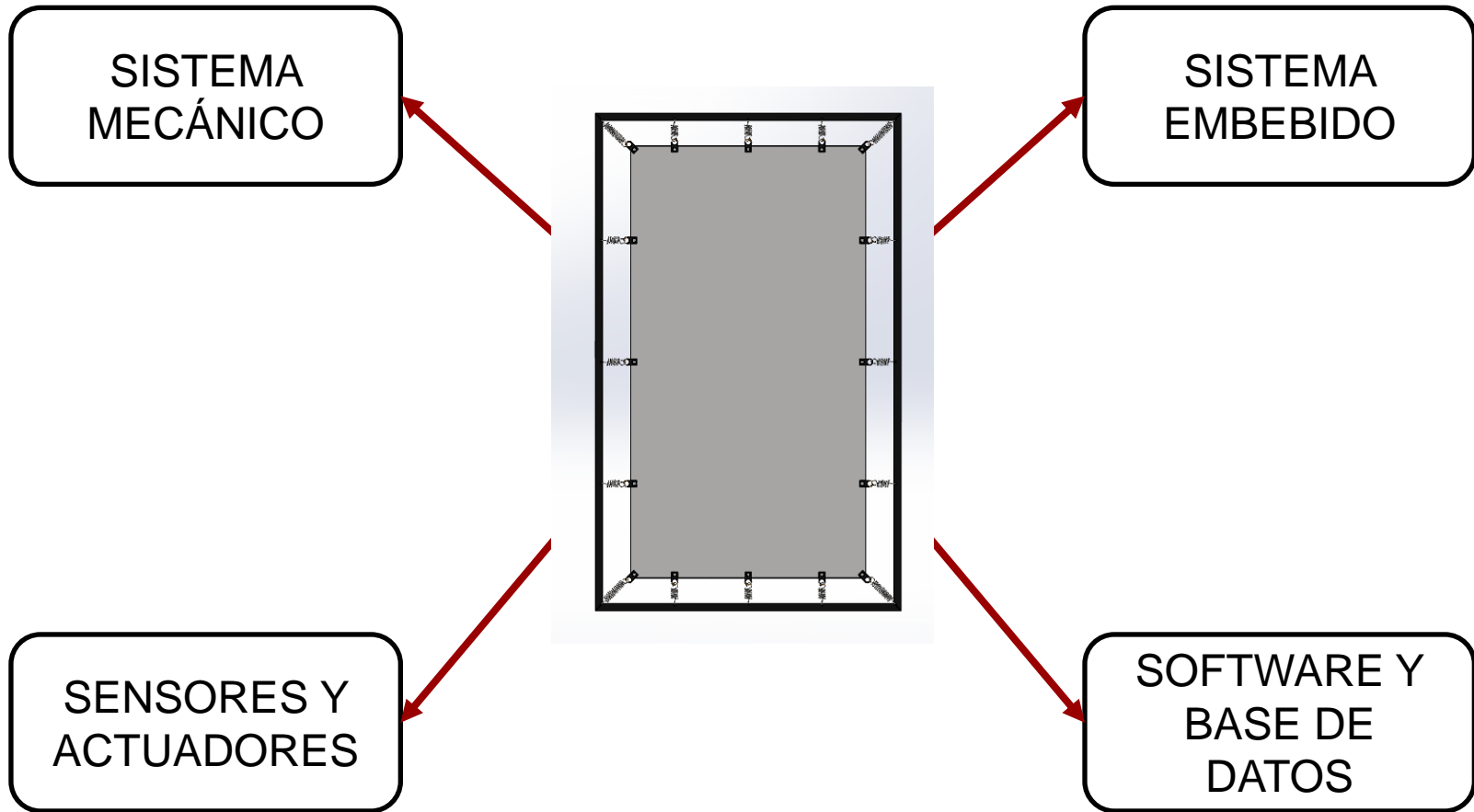
MIDY Scientific Trainers es una empresa encaminada al estudio y desarrollo del deporte.

Se requería de la implementación de un panel de entrenamiento de reflejos automatizado, capaz de adquirir datos de velocidad, aceleración y fuerza inferida en cada golpe realizado. Para posteriormente almacenarnos en una base de datos y generar un análisis estadístico del desarrollo del atleta.

Propuesta de diseño – Prototipo del panel deseado



Selección de Componentes



Selección de Componentes – Sistema Mecánico

Perfil angular de acero
ASTM A36



Lona con PVC adherido y
refuerzo de poliéster



Resortes Helicoidales de
acero ASTM A229



Selección de Componentes – Sensores y actuadores

Servomotores DS3218MG



Cámara estereoscópica Zed 2



Reflectores Sylvania 20W

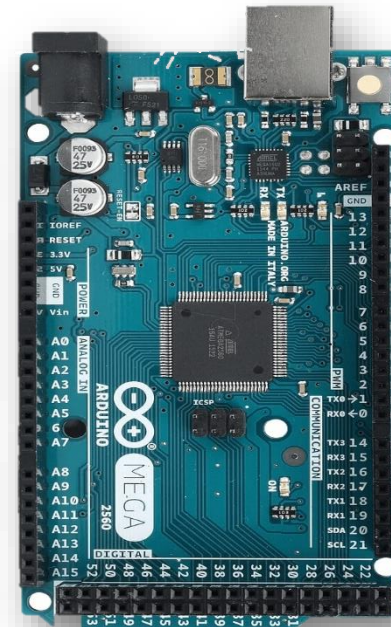


Selección de Componentes – Sistema Embebido

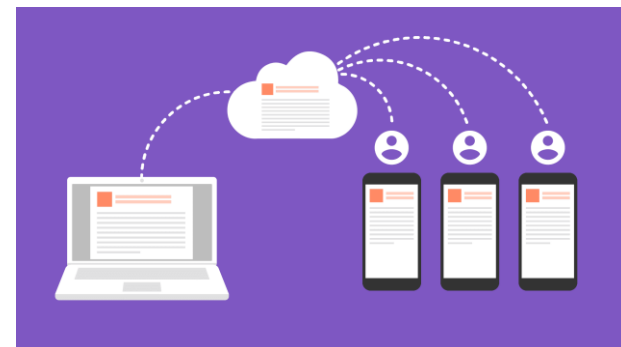
Jetson Nano



Arduino Mega 2560

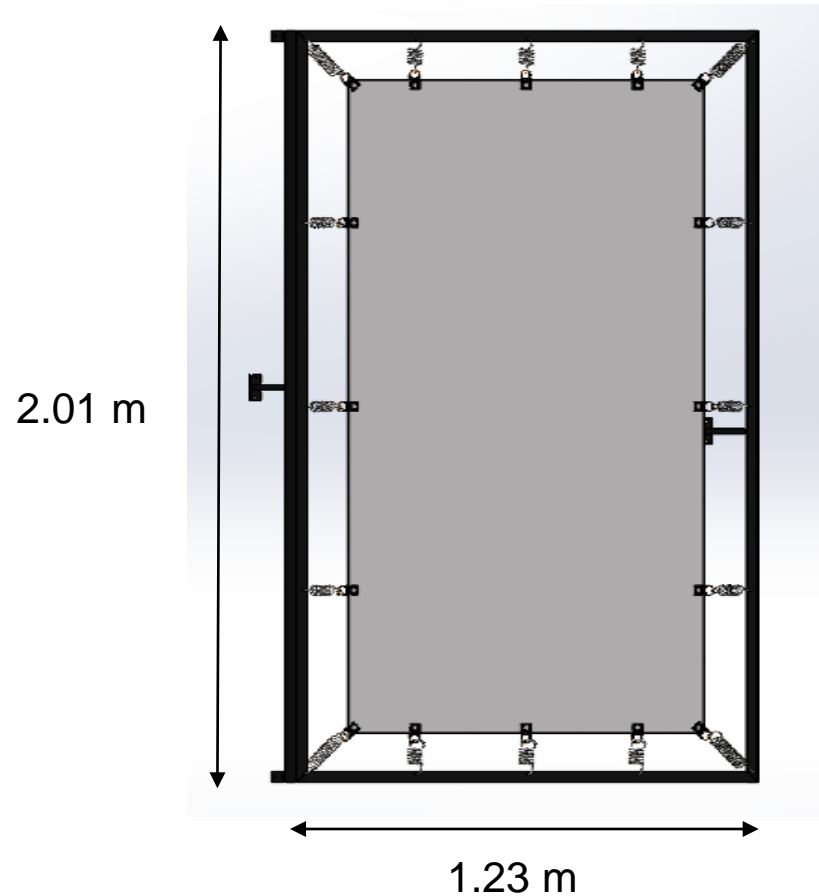


Selección de Componentes – Software y base de datos



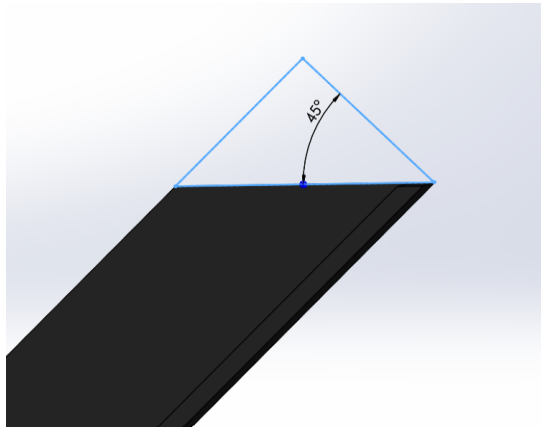
Diseño Mecánico

El diseño mecánico se basó en los requerimientos de la empresa.

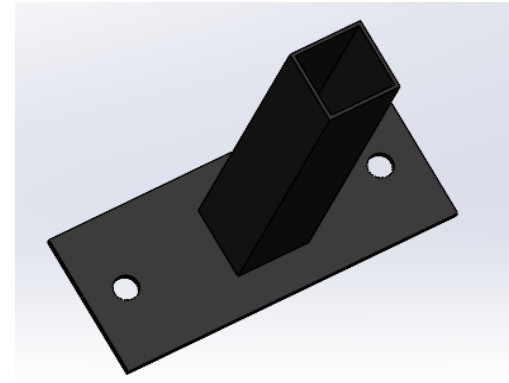


Estructura rectangular

Todos los componentes de la estructura rectangular son de acero
ASTM A36

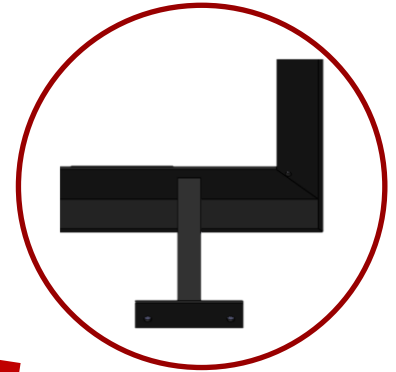
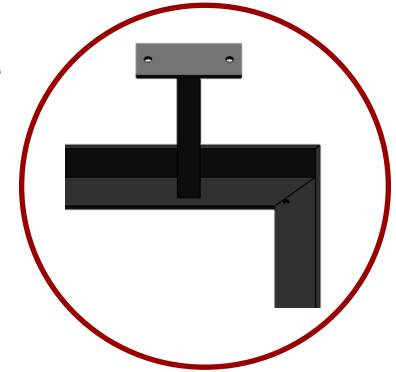
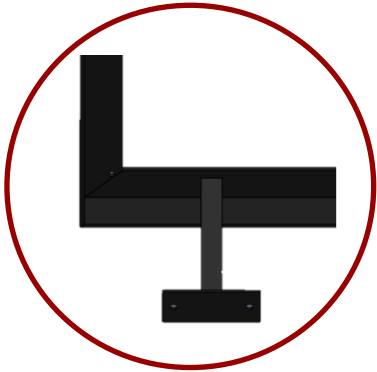


Corte a 45° en los
extremos de cada perfil



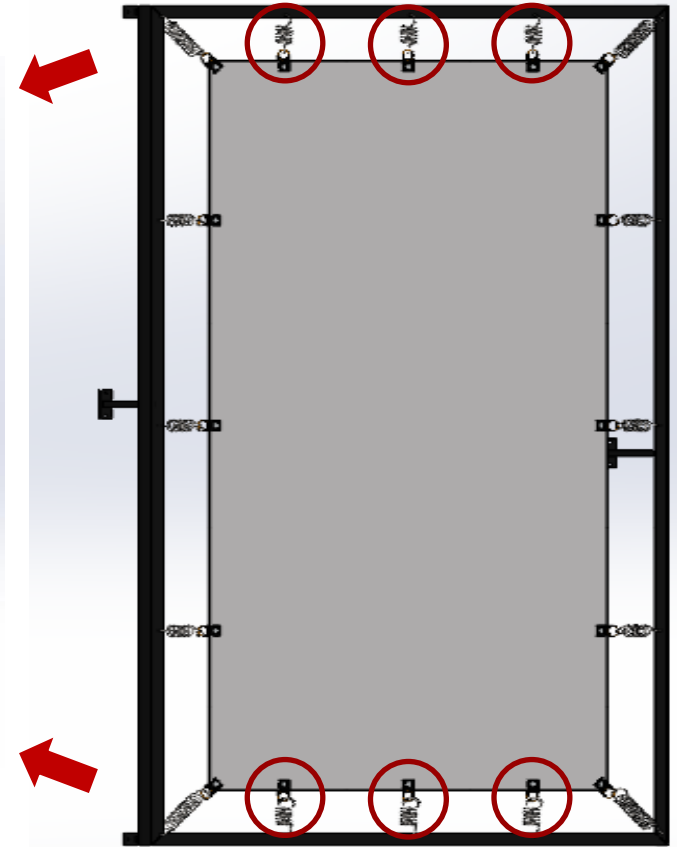
6 patas traseras
Tubo cuadrado 30x30x3mm
Platina de 1/8"

Estructura rectangular

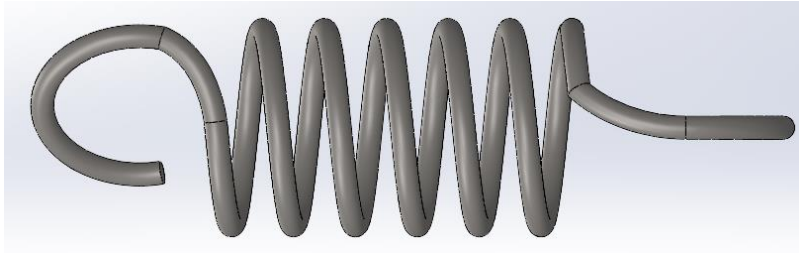


Resortes helicoidales de extensión - Verticales

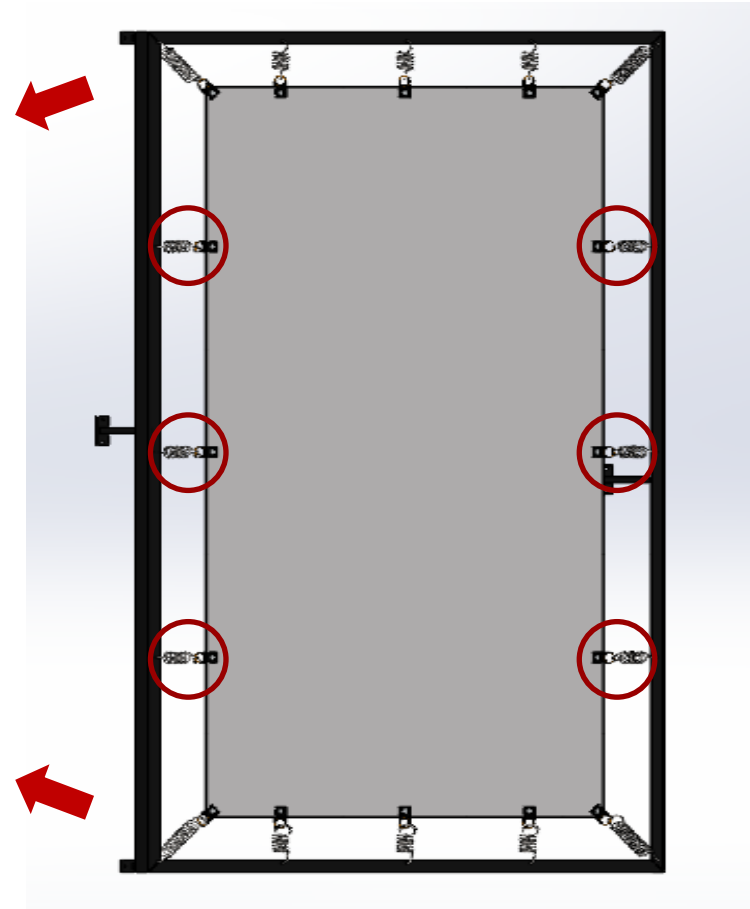
Parámetro	Representación	Valor	Unidades
Carga o fuerza aplicada	F	200	N
Desplazamiento o deformación	x	7.2	mm
Rigidez del resorte	k	27.77	mm
Diámetro del alambre	d	3.05	mm
Diámetro del resorte	D	25.7	mm
Material del resorte		ASTM A229	
Número de espiras	N	4.75	<i>vuelatas</i>
Paso del resorte	p	3.05	mm
Longitud del resorte	L	6.7	cm



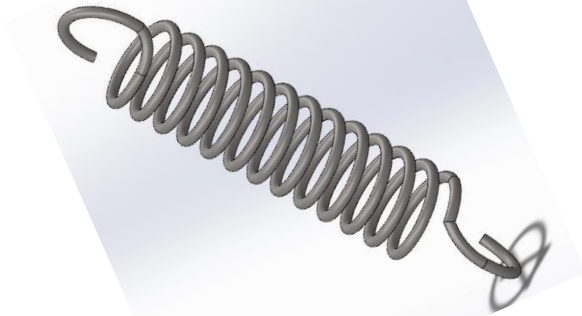
Resortes helicoidales de extensión - Horizontales



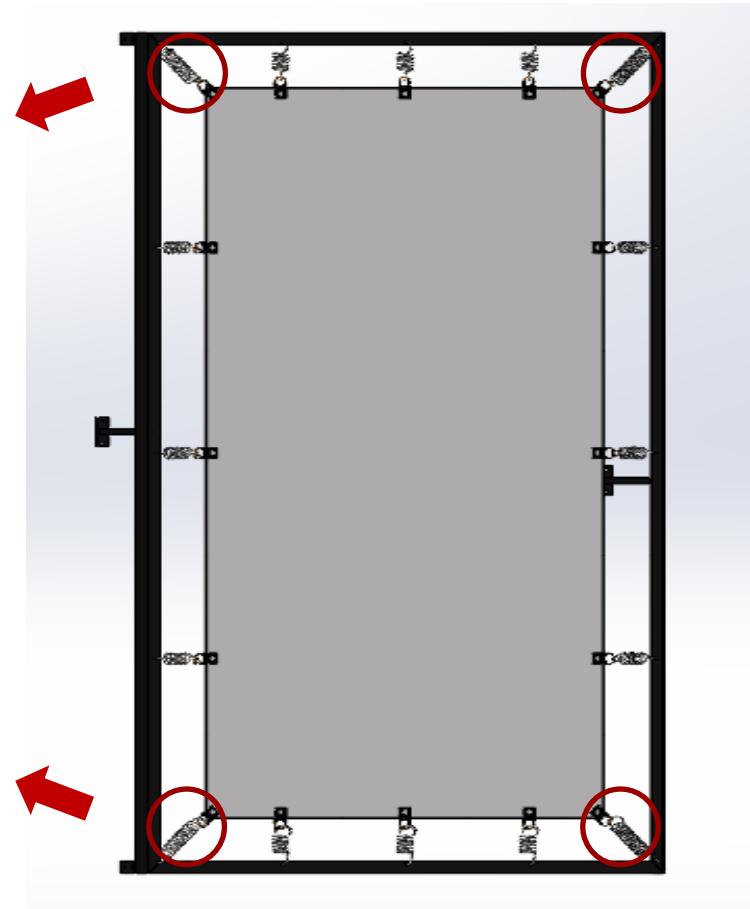
Parámetro	Representación	Valor	Unidades
Carga o fuerza aplicada	F	200	N
Desplazamiento o deformación	x	8.49	mm
Rigidez del resorte	k	23.55	mm
Diámetro del alambre	d	3.05	mm
Diámetro del resorte	D	25.7	mm
Material del resorte		ASTM A229	
Número de espiras	N	5.6	<i>vueltas</i>
Paso del resorte	p	3.05	mm
Longitud del resorte	L	7.3	cm



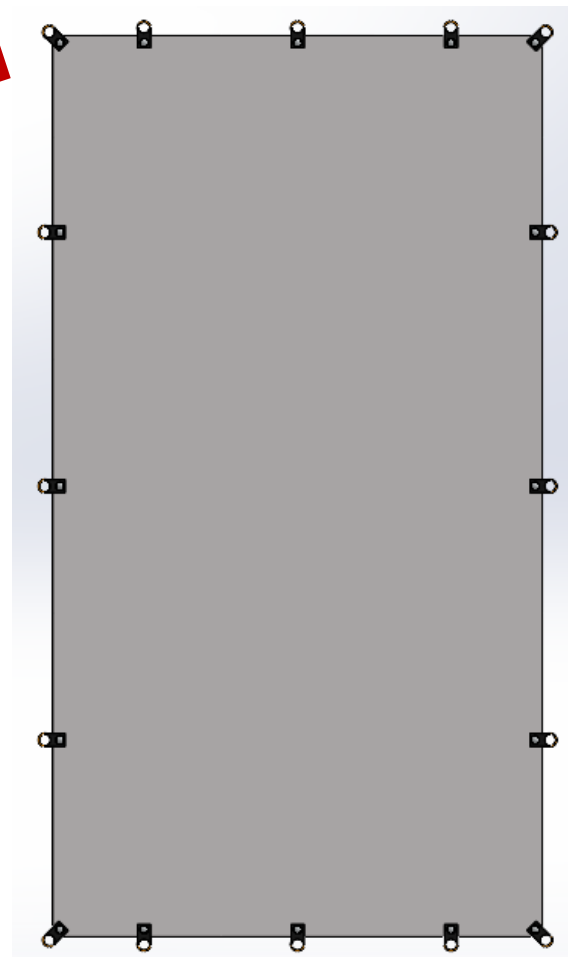
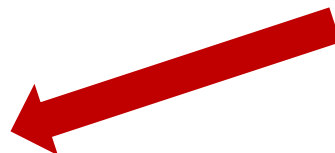
Resortes helicoidales de extensión - Diagonales



Parámetro	Representación	Valor	Unidades
Carga o fuerza aplicada	F	200	N
Desplazamiento o deformación	x	18.91	mm
Rigidez del resorte	k	10.63	mm
Diámetro del alambre	d	3.05	mm
Diámetro del resorte	D	25.7	mm
Material del resorte		ASTM A229	
Número de espiras	N	12.4	<i>vuel</i> tas
Paso del resorte	p	3.05	mm
Longitud del resorte	L	9.25	cm



Lona de soporte



Parámetro	Valor	Unidades
Alto	1.70	<i>m</i>
Ancho	0.90	<i>m</i>
Espesor	3	<i>mm</i>
Refuerzos	4	-
Remaches con argolla	16	-

Programación

Detección y seguimiento de puntos clave (Python / Zed SDK)

Configuración y desarrollo de la base de datos (Firebase)



Desarrollo de la interfaz gráfica (Tkinter)

Cálculo de la velocidad de reacción (Python)

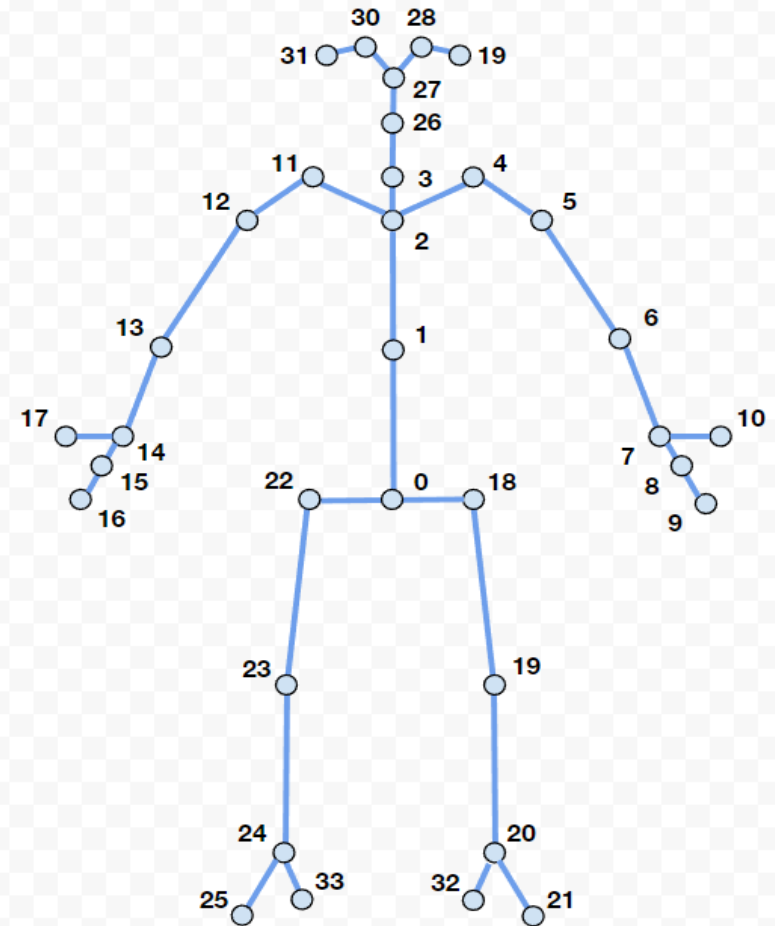
Algoritmo de detección y seguimiento de puntos clave

```
def cvt(pt, scale):
    """
    Function that scales point coordinates
    """
    out = [pt[0] * scale[0], pt[1] * scale[1]]
    return out
```

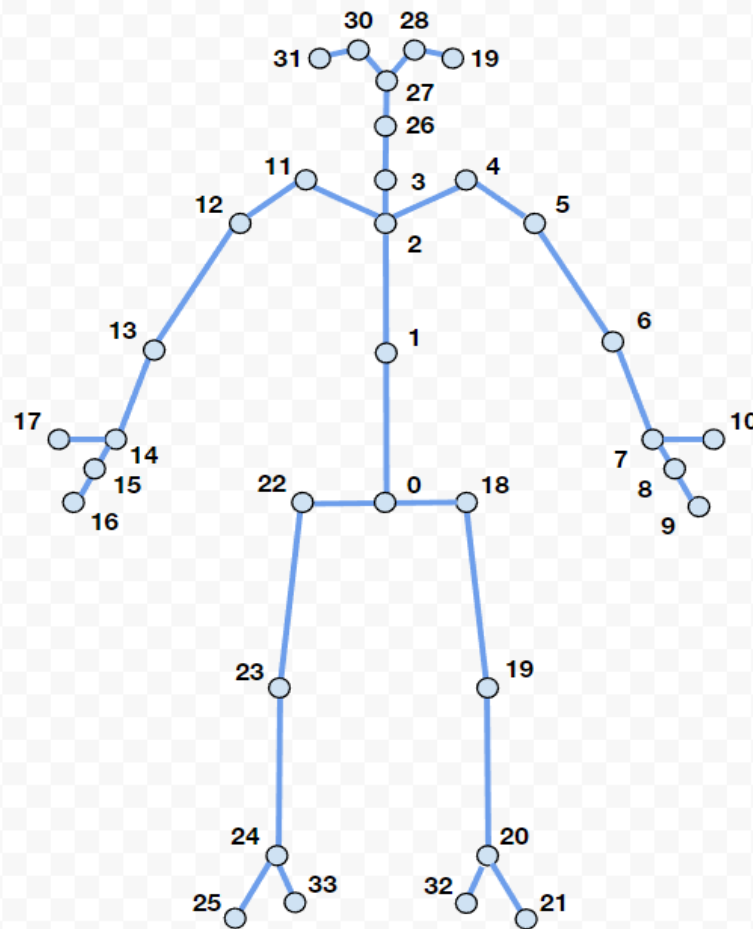
```
if zed.grab() == sl.ERROR_CODE.SUCCESS:
    # Obteniendo la imagen de la cámara izquierda
    zed.retrieve_image(image, sl.VIEW.LEFT, sl.MEM.CPU,
                      display_resolution)

    # Obteniendo objetos
    zed.retrieve_bodies(bodies, body_runtime_param)
    # Actualizando la ventana de vista
    image_left_ocv = image.get_data()
    overlay = image_left_ocv.copy()
    # Filtrado de puntos clave
    for obj in bodies.body_list:
        if render_object(obj, body_param.enable_tracking):
            if len(obj.keypoint_2d) > 0:
                color = generate_color_id_u(obj.id)
                IMP_KP = [obj.keypoint_2d[16], obj.keypoint_2d[9],
                        obj.keypoint_2d[25], obj.keypoint_2d[21]]
                NAMES = ["MANO DERECHA", "MANO IZQUIERDA",
                        "PIE DERECHO", "PIE IZQUIERDO"]

                k = 0
                for i in IMP_KP:
                    cv_kp = cvt(i, image_scale)
                    # Obteniendo las coordenadas "x" y "y" de los puntos clave
                    t2 = time.time()
                    x = int(cv_kp[0])
                    y = int(cv_kp[1])
```



Algoritmo de detección y seguimiento de puntos clave



Algoritmo de detección de golpes

```
cv2.imshow('img', image_left_ocv)
if (x < image_left_ocv.shape[1] and y <
    image_left_ocv.shape[0]):
    # Dibujando un círculo sobre cada punto clave
    cv2.circle(image_left_ocv, (x, y), 10, color,
               -1)
    # Colocando el nombre de cada punto clave
    cv2.putText(image_left_ocv, 'x: {},y: {}'.format(x, y),
                (x - 60, y - 15), cv2.FONT_HERSHEY_SIMPLEX, 0.7,
                color, 1)
    cv2.putText(image_left_ocv, '{}'.format(NAMES[k]),
                (x - 60, y - 55), cv2.FONT_HERSHEY_SIMPLEX, 0.7,
                color, 1)

xp = 550
# Dibujando la barrera de golpeo an x=550
cv2.line(image_left_ocv, (xp, 0), (xp, 720),
         color, 2)
```

```
# Calculando el error en x para detectar cada golpe
errorx = x - xp
v_error[k] = errorx

for i in v_error:
    if i > 0:
        c = c + 1
if c == 3:
    v_auxn.append(k)
elif c == 4:
    v_auxn.clear()
```

Algoritmo de detección de golpes

```
# Determinando el punto clave responsable del golpe
if len(v_auxn) == 1:
    if v_auxn[0] == 0 and auxmd == False:
        cg = cg + 1
        v_savenames.append("MANO DERECHA")
        v_savespeeds.append(vf)
        v_af.append(af)
        fc = round(
            v_m[k] * af * factor_conversion_kgf)
        v_save_fc.append(fc)
        auxmd = True
        ser.write(b"2\n")
        print("MANO DERECHA")
```

```
elif v_auxn[0] == 1 and auxmi == False:
    cg = cg + 1
    v_savenames.append("MANO IZQUIERDA")
    v_savespeeds.append(vf)
    v_af.append(af)
    fc = round(
        v_m[k] * af * factor_conversion_kgf)
    v_save_fc.append(fc)
    auxmi = True
    ser.write(b"1\n")
    print("MANO IZQUIERDA")
```

```
elif v_auxn[0] == 2 and auxpd == False:
    cg = cg + 1
    v_savenames.append("PIE DERECHO")
    v_savespeeds.append(vf)
    v_af.append(af)
    fc = round(
        v_m[k] * af * factor_conversion_kgf)
    v_save_fc.append(fc)
    auxpd = True
    ser.write(b"4\n")
    print("PIE DERECHO")
```

```
elif v_auxn[0] == 3 and auxpi == False:
    cg = cg + 1
    v_savenames.append("PIE IZQUIERDO")
    v_savespeeds.append(vf)
    v_af.append(af)
    fc = round(
        v_m[k] * af * factor_conversion_kgf)
    v_save_fc.append(fc)
    auxpi = True
    ser.write(b"3\n")
    print("PIE IZQUIERDO")
```

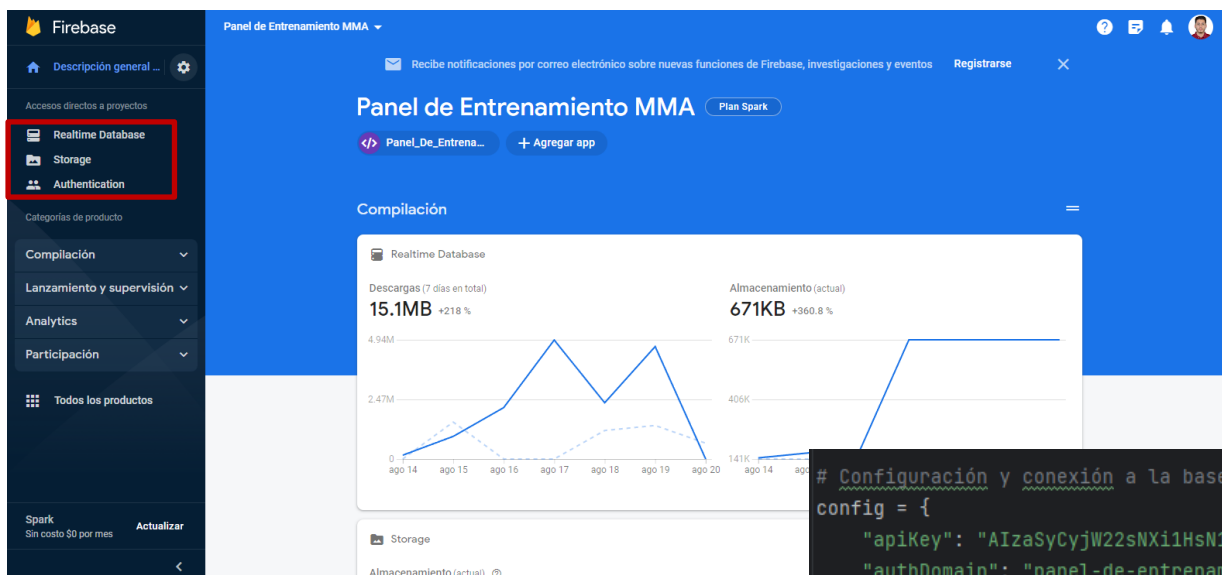
Cálculo de la velocidad en base a la variación de posición

```
        color, 2)
delta_t = v_times[0]
# Obteniendo la variación de posición en "x" y "y"
delta_x = abs(x - POSX[k])
delta_y = abs(y - POSY[k])
# Obteniendo las velocidades en "x" y "y"
vx = delta_x / delta_t
vy = delta_y / delta_t
# Obteniendo la velocidad final del golpe en m/s
vf = round((((vx ** 2) + (
        vy ** 2)) ** 0.5) * factor_escalonamiento,
        2)
af = abs(round((vf - VEL[k]) / delta_t, 2))
# Calculando el error en x para detectar cada golpe
errorx = x - xp
v_error[k] = errorx
```

```
t4 = time.time()
subt1 = t4 - t3
subt2 = t2 - t1
tiempo_final = subt2 + subt1
v_times[0] = tiempo_final
```

```
factor_escalonamiento = 0.00327435
```

Configuración y desarrollo de la base de datos



```
# Configuración y conexión a la base de datos en Firebase
config = {
    "apiKey": "AIzaSyCyjW22sNXi1HsN1qFdfygNoj2p0oZgi0c",
    "authDomain": "panel-de-entrenamiento-mma.firebaseio.com",
    "databaseURL": "https://panel-de-entrenamiento-mma-default-rtdb.firebaseio.com",
    "projectId": "panel-de-entrenamiento-mma",
    "storageBucket": "panel-de-entrenamiento-mma.appspot.com",
    "messagingSenderId": "561799190822",
    "appId": "1:561799190822:web:2613294506319312a989e7",
    "measurementId": "G-6JVHXB8PJ0"
}

# Variables de desarrollo e interacción con la base de datos

firebase = pyrebase.initialize_app(config)
auth = firebase.auth()
db = firebase.database()
storage = firebase.storage()
```

Firebase Authentication

```

if len(user_r_get) != 0:
    if len(password_r_get) >= 5:
        if password_r_get == confirm_password_r_get:
            try:
                user = auth.create_user_with_email_and_password(user_r_get, password_r_get)
                token = user['idToken']
                auth.send_email_verification(token)
                username_entry_r.delete(0, END)
                password_entry_r.delete(0, END)
                confirm_password_entry_r.delete(0, END)
                # MENSAJE DE EXITO
                ventana_registrarse.destroy()
                ventana_exito = tk.Toplevel()
                ventana_exito.geometry("400x400")
                ventana_exito.title("Registro exitoso")
                # ventana_exito.iconbitmap("Images/MIDY.ico")
                ventana_exito.resizable(False, False)
                Resize_Window(ventana_exito, 400, 400)

```

```

def Inicio():
    c = 0
    aux = 0
    palabra = ""
    ftli = True
    # OBTENER VALORES DEL REGISTRO
    usuario_get = username_ingreso.get()
    contraseña_get = password_ingreso.get()
    try:
        username_entry.delete(0, END)
        password_entry.delete(0, END)
        user_i = auth.sign_in_with_email_and_password(usuario_get, contraseña_get)
        token = user_i['idToken']
        email_verification = auth.get_account_info(token)['users'][0]['emailVerified']
        if email_verification == True:
            messagebox.showinfo(title="Éxito", message="Ha ingresado con éxito")
            # FORMING THE USER NAME FOR THE DB

```

Authentication

Users Sign-in method Templates Usage Settings Extensiones **NEVA**

Buscar por dirección de correo electrónico, número de teléfono o UID de usuario

Agregar usuario

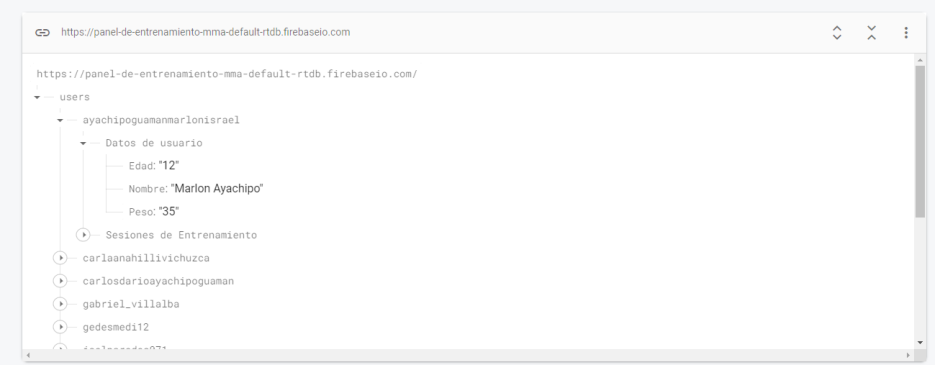
Identificador	Proveedores	Fecha de creación	Fecha de acceso	UID de usuario
gedesmedi1@gmail.com		12 jul 2023	12 ago 2023	qnuCDp7PA8gpFHHkYTTBprj3fKc3
joelparedes071@gmail.com		12 jul 2023	12 ago 2023	y6v8YtLlF0GMfmlL3JUngrzpr03
carlosdarioayachipoguama...		12 jul 2023	18 ago 2023	4V9xiDOkeRbCVkTCpXMQcDnj8P...
marypoveda1994@gmail.c...		12 jul 2023	12 ago 2023	VnjQas6Q5NCkTejHs9hq3G9RX2
ayachipoguanamarlonisr...		12 jul 2023	18 ago 2023	WgfnDBr76MVFVLgb32J850y0MT...
may884573@gmail.com		12 jul 2023	12 ago 2023	54Pd9UogQK0JUEC1fKrfzWJGX13
carlaanahillivichuzca@gm...		12 jul 2023	19 ago 2023	wjokFjXU2zZyA6i3RYD7144AN2m1
scarlettbrigittetechimbochis...		12 jul 2023	12 ago 2023	urpru4ZGokH0SHhDONGSWPD0...
luigipincaychoez@gmail...		12 jul 2023	12 ago 2023	Yl8NmMpmKQNZ2CunVnYblz6sZ7...

Firestore Realtime Database

```
datos = db.child("users").child(palabra).get()
aux3 = 0
aux4 = 0
for i in datos.each():
    aux3 = aux3 + 1
if aux3 == 2:
    datos_sesiones = db.child("users").child(palabra).child(
        "Sesiones de Entrenamiento").get()
    for i in datos_sesiones.each():
        aux4 = aux4 + 1
    nueva_s = aux4 + 1
    numero_sesion = str(nueva_s)
    db.child("users").child(palabra).child(
        "Sesiones de Entrenamiento").child(
        "Sesión {}".format(numero_sesion)).child("Golpes Realizados")
    db.child("users").child(palabra).child(
        "Sesiones de Entrenamiento").child(
        "Sesión {}".format(numero_sesion)).child(
        "Extremidades Utilizadas").set(v_savenames)
```

Realtime Database

Datos Reglas Copias de seguridad Uso Extensiones **NOVA**



The screenshot shows the Realtime Database console interface. The URL is <https://panel-de-entrenamiento-mma-default-rtdb.firebaseio.com/>. The data is organized into a tree structure under the 'users' node. The selected user is 'ayachipoguanmanlonsrael', which contains a 'Datos de usuario' node with fields: 'Edad: 12', 'Nombre: Marlon Ayachipo', and 'Peso: 35'. Below this is a 'Sesiones de Entrenamiento' node, which is expanded to show a list of session IDs: 'carlaanahillivichuzca', 'carlosdarioayachipoguanman', 'gabriel_villalba', 'gedesmedi12', and '...'. The interface includes navigation tabs for 'Datos', 'Reglas', 'Copias de seguridad', 'Uso', and 'Extensiones'.








Firestore Cloud Storage


Storage


Files Rules Usage Extensiones **NUEVA**

Protege tus recursos de Storage contra los abusos, como fraudes de facturación o suplantación de identidad. [Configurar la Verificación de aplicaciones](#) X

gs://panel-de-entrenamiento-mma.appspot.com [Subir archivo](#) + X

Nombre	Tamaño	Tipo	Modificación más reciente
 ayachipoguanmanmarlonisrael	94.33 KB	image/jpeg	26 jul 2023
 carlaanahilivichuzza	247.49 KB	image/jpeg	26 jul 2023
 carlosdarioayachipoguanman	97.42 KB	image/jpeg	26 jul 2023
 gabriel_villalba	232.13 KB	image/jpeg	17 ago 2023
 gedesmedi12	95.03 KB	image/jpeg	26 jul 2023
 joelparedes071	107.96 KB	image/jpeg	26 jul 2023
 luisgipincaychoez	99.54 KB	image/jpeg	26 jul 2023

 ayachipoguanmanmarlonisrael X



Nombre
ayachipoguanmanmarlonisrael [✕](#)

Tamaño
96,598 bytes

Tipo
image/jpeg

Creado
26 jul 2023, 19:23:04

Actualizado

```
# IMAGENES
# CARGA DE IMAGEN
try:
    url = storage.child(palabra).get_url(None)
    picture = urllib.request.urlopen(url)
    foto = Image.open(picture)
    foto = foto.resize((225, 225), Image.ANTIALIAS)
    foto = ImageTk.PhotoImage(foto)
    UserWindowImages.push(foto)

    foto = Label(canvas_usuario, image=foto)
    foto.place(x=62, y=32)
except:
    inicios_w_u = Label(canvas_usuario, image=inicios_u, bg=rojo)
    inicios_w_u.place(x=62, y=32)
```

```
def Actualizar_imagen():
    archivo = filedialog.askopenfilename(title="Seleccione una imagen",
                                        filetypes=[('Archivos png', '*.png'),
                                                  ('Archivos jpg', '*.jpg'),
                                                  ('Archivos jpeg', '*.jpeg')])

    if archivo is not None:
        imagen_usuario = Image.open(archivo)
        imagen_usuario = imagen_usuario.resize((225, 225), Image.ANTIALIAS)
        imagen_usuario = ImageTk.PhotoImage(imagen_usuario)
        UserWindowImages.push(imagen_usuario)

        imagen_usuario_w = Label(canvas_usuario, image=imagen_usuario)
        imagen_usuario_w.place(x=62, y=32)
        storage.child(palabra).put(archivo)
```

Desarrollo de la interfaz gráfica – Funciones básicas

```
def Window_Configuration(window_name,window_size,window_title):
    # CONFIGURACIÓN DE LA VENTANA
    window_name.geometry(window_size)
    window_name.title(window_title)
    window_name.resizable(False,False)
```

15 usages

```
def Resize_Window(window_name,window_width,window_height):
    wetotal = window_name.winfo_screenwidth()
    hetotal = window_name.winfo_screenheight()
    weventana = window_width
    heventana = window_height
    pewidth = round(wetotal / 2 - weventana / 2)
    peheight = round(hetotal / 2 - heventana / 2)
    peheight = peheight - 30
    window_name.geometry(str(weventana) + "x" + str(heventana) + "
```

INDICADORES



TIEMPO
(De cada golpe / Total)

Presione el icono para
EMPEZAR



FUERZA
(De golpe inferida basada
en el peso)

Presione el icono para ver las
INSTRUCCIONES



VELOCIDAD
(Instantanea al momento del
contacto)

Presione el icono para
SALIR



DATA BASE
(Registrado en la nube de
cada parámetro medido)

**MENU
PRINCIPAL**

PANEL DE ENTRENAMIENTO AUTOMÁTICO



Ventanas de la interfaz gráfica – Instrucciones

INSTRUCCIONES DE USUARIO

1

REGÍSTRATE / INICIA SESIÓN

Desde el Menu Principal, híz click en "Empezar" y completa el formulario de registro si es la primera vez que utilizas la aplicación, caso contrario, ingresa tu usuario y contraseña para iniciar sesión.

REALIZA TU ENTRENAMIENTO

Una vez que has ingresado tus datos, es momento de iniciar con el entrenamiento, golpea los lugares del panel en donde se pueda apreciar una luz encendida, para ello puedes utilizar puños o patadas.

2

3

VISUALIZA TUS RESULTADOS

Durante el entrenamiento se adquieren 4 tipos de datos: tiempo de duración de la sesión, velocidad, fuerza y aceleración inferida de cada golpe realizado. Todos los datos se ponen a tu disposición al finalizar el entrenamiento.

¿Estás listo?... Volvamos al Menu Principal

Ventanas de la interfaz gráfica – Menú de Ingreso / Registro



Ventanas de la interfaz gráfica – Formulario de Registro


FORMULARIO DE REGISTRO




CORREO ELECTRÓNICO



CONTRASEÑA



CONFIRMAR CONTRASEÑA



Registrarse

Ventanas de la interfaz gráfica – Ventana de Registro Exitoso

DATOS REGISTRADOS CON ÉXITO

Hemos enviado un mensaje de confirmación a su correo electrónico.

Por favor revise la bandeja de spam y haga click en el enlace generado, para terminar el proceso de registro



Menu de Ingreso

Ventanas de la interfaz gráfica – Formulario de Ingreso

INICIAR SESIÓN



CORREO ELECTRÓNICO



CONTRASEÑA



Olvide mi contraseña

Ingresar

Ventanas de la interfaz gráfica – Registro por primera vez

INGRESA TUS DATOS



NOMBRE Y APELLIDO

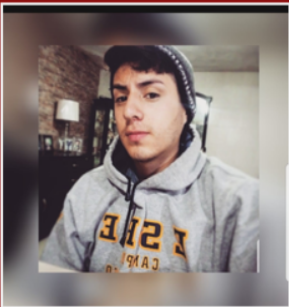


EDAD AÑOS

MASA CORPORAL KG.

Actualizar


Ventanas de la interfaz gráfica – Perfil de usuario





ACTUALIZAR

RICARDO NOROÑA

DATOS DE USUARIO

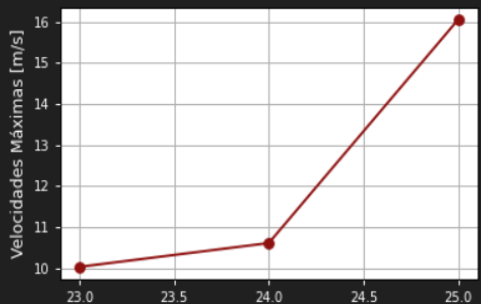
NOMBRE
 RICARDO NOROÑA

EDAD
 26 AÑOS

PESO
 82 KG

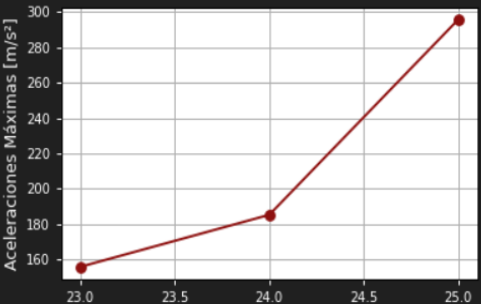
ACTUALIZAR

Velocidades Máximas vs Sesiones de entrenamiento




Sesión	Velocidad Máxima [m/s]
23.0	10.0
24.0	10.5
25.0	16.0

Aceleraciones Máximas vs Sesiones de entrenamiento



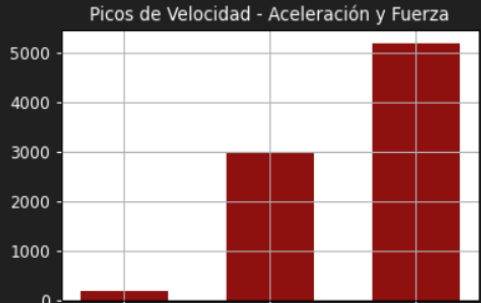
Sesión	Aceleración Máxima [m/s ²]
23.0	155
24.0	185
25.0	295

Fuerzas Máximas vs Sesiones de entrenamiento



Sesión	Fuerza Máxima [kgf]
23.0	230
24.0	320
25.0	520

Picos de Velocidad - Aceleración y Fuerza



Métrica	Valor Máximo
V _{máx} [m/s]	~200
A _{máx} [m/s ²]	~3000
F _{máx} [kgf]	~5200

Consultar Datos Previos

Iniciar Sesión de Entrenamiento

Reordenar Luces

Ventanas de la interfaz gráfica – Consultar datos previos



DATOS DE USUARIO

NOMBRE  RICARDO NOROÑA

EDAD  26 AÑOS

PESO  82 KG

DATOS OBTENIDOS PARA CADA SESIÓN

▾ Sesión 1

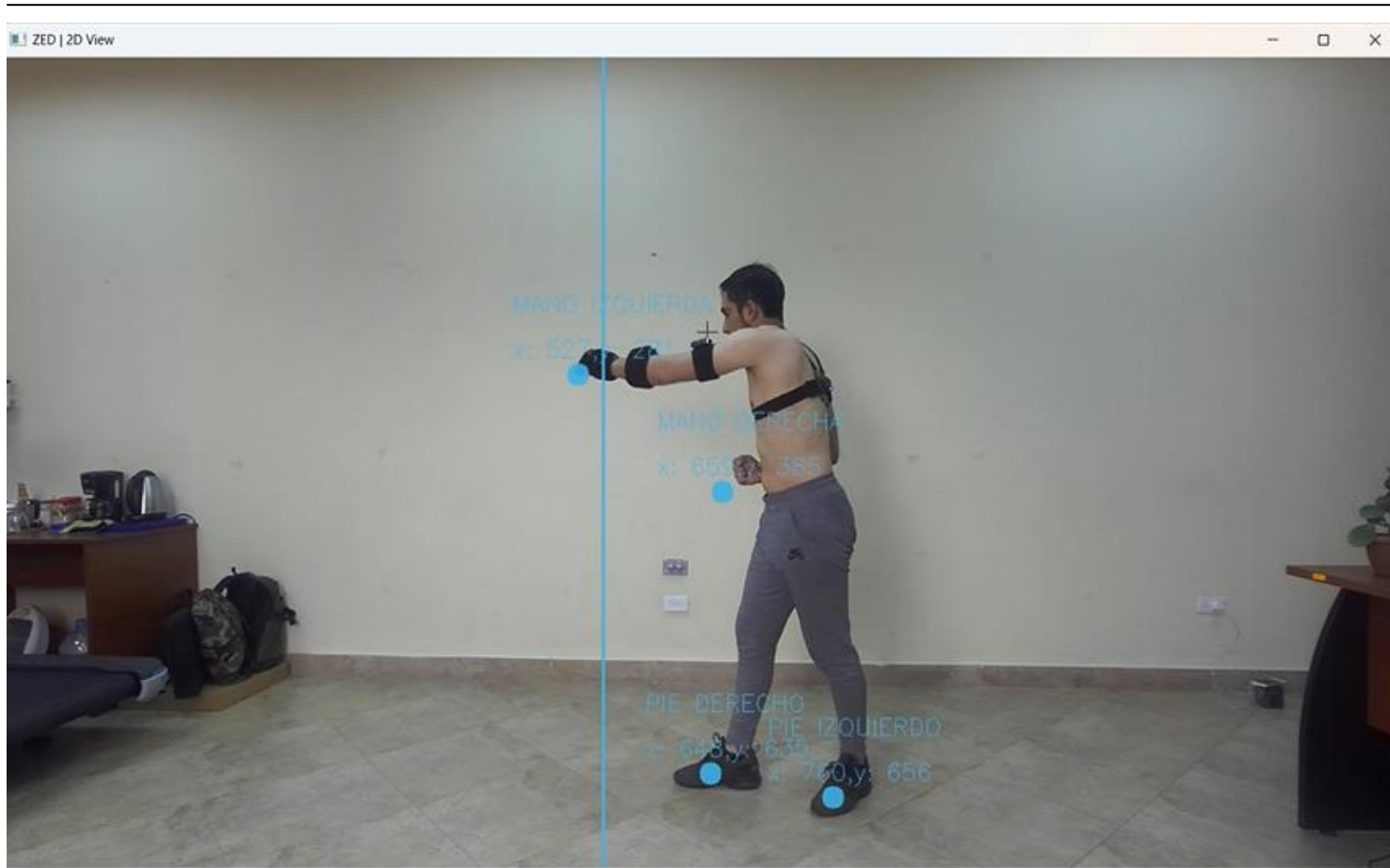
- Aceleración Máxima
- Aceleración de cada golpe
- Duración del entrenamiento
- Extremidades Utilizadas
- Fuerza Máxima
- Fuerza de cada golpe
- Golpes Realizados
- Velocidad Máxima
- Velocidad de cada golpe

▸ Sesión 2

▸ Sesión 3

[Volver al Menú de Usuario](#)

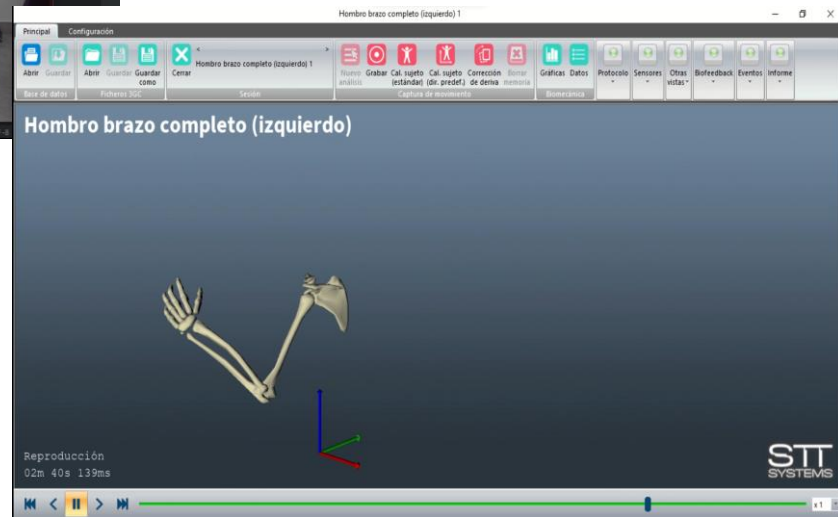
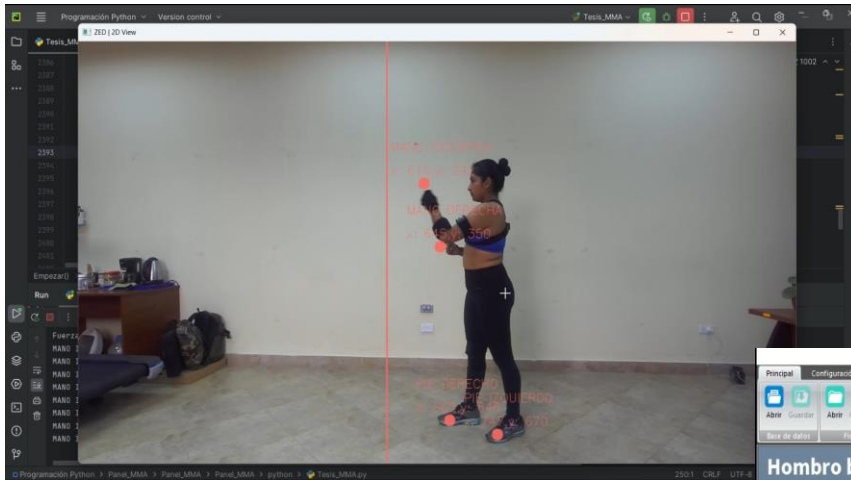
Ventanas de la interfaz gráfica – Iniciar Entrenamiento



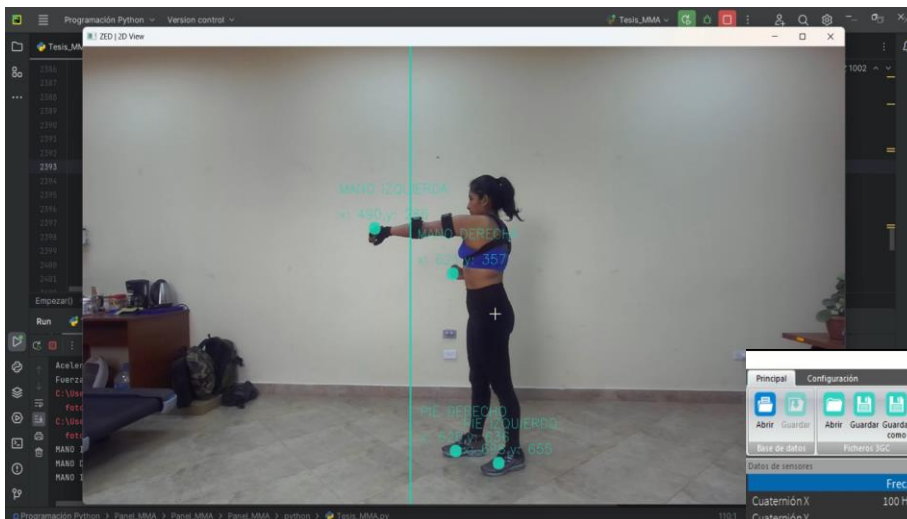
Validación de datos de velocidad obtenidos



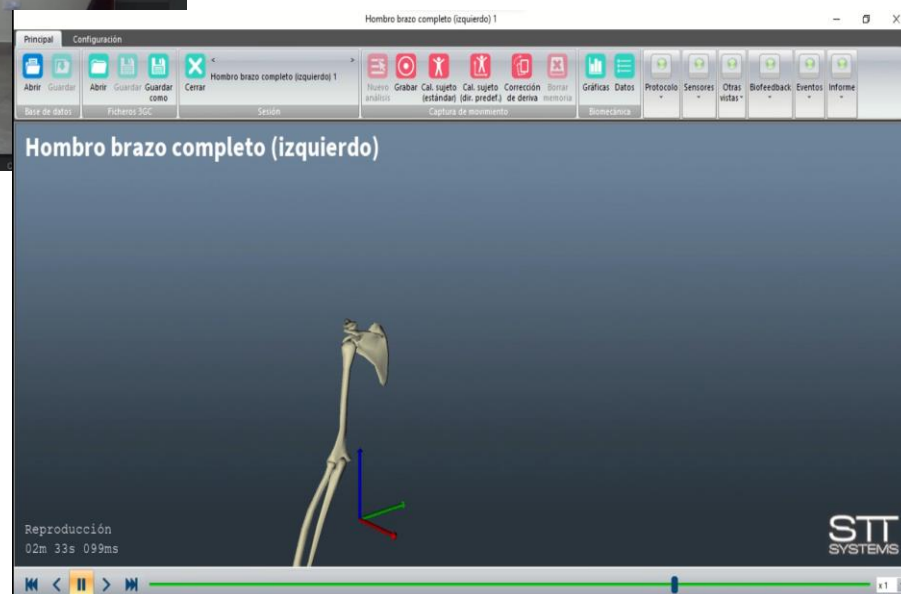
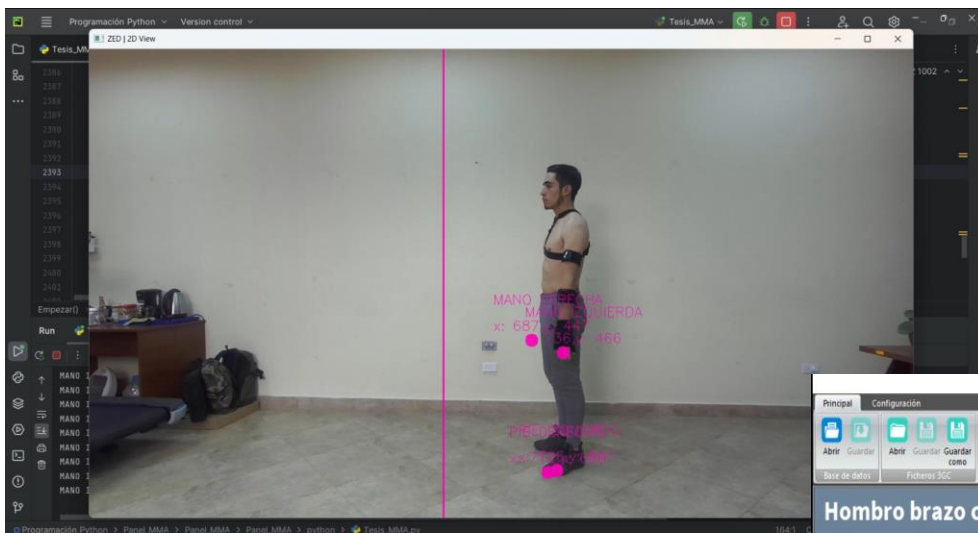
Validación de datos de velocidad obtenidos



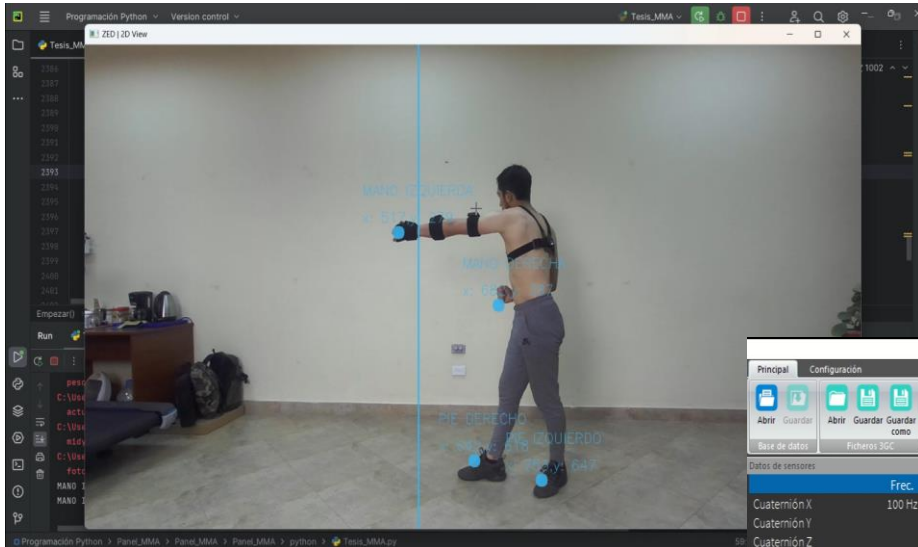
Validación de datos de velocidad obtenidos



Validación de datos de velocidad obtenidos



Validación de datos de velocidad obtenidos



Validación de datos de velocidad obtenidos

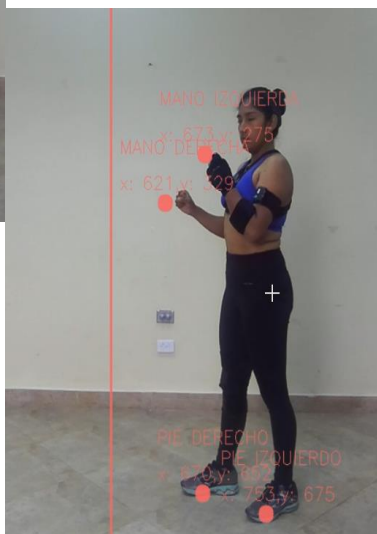
$$e_T = \frac{e_{s1} + e_{s2}}{2}$$

$$e_T = \frac{7.97 + 13.04}{2}$$

$$e_T = 10.050 \%$$

Porcentaje de confiabilidad del 89.95%

Pruebas y Resultados – Detección de puntos clave



Pruebas y Resultados – Detección de puntos clave

	Posiciones analizadas	Media de precisión	Media de exactitud
Sujeto 1	5	100%	80%
Sujeto 2	5	100%	65%
Totales	10	100%	72.5%

Pruebas y Resultados – Entrenamiento Semana 0



Pruebas y Resultados – Resultados Semana 0

Resultados Totales

Media de golpes realizados	Media de velocidad mínima (m/s)	Media de velocidad máxima (m/s)	Media de velocidad (m/s)
107.6	0.36	2.04	1.204

Pruebas y Resultados – Entrenamiento Semana 1

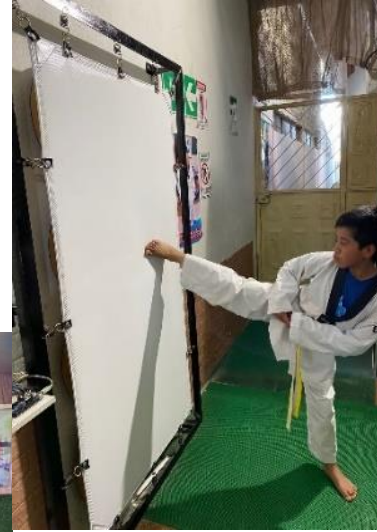


Pruebas y Resultados – Resultados Semana 1

Resultados Totales

Media de golpes realizados	Media de velocidad mínima (m/s)	Media de velocidad máxima (m/s)	Media de velocidad (m/s)
130.6	0.42	2.37	1.42

Pruebas y Resultados – Entrenamiento Semana 2



Pruebas y Resultados – Resultados Semana 2

Resultados Totales

Media de golpes realizados	Media de velocidad mínima (m/s)	Media de velocidad máxima (m/s)	Media de velocidad (m/s)
141	0.60	2.92	1.81

Validación de Hipótesis

Resultados Totales

Sujeto de prueba	Semana 0	Semana 1	Semana 2
1	1.54	1.58	1.97
2	0.98	1.15	1.47
3	1.11	1.33	1.73
4	1.00	1.26	1.82
5	1.39	1.78	2.10

Validación de Hipótesis – Prueba de Normalidad

Planteamiento de la hipótesis

(H_o) = Los datos tienen una distribución normal

(H_i) = Los datos no tienen una distribución normal

Nivel de confianza

Se trabajó con un nivel de confianza del 95% y un nivel de significancia del 5%.

Se describe el método estadístico empleado y se obtienen los resultados.

Validación de Hipótesis – Prueba de Normalidad

Pruebas de normalidad

	Kolmogorov-Smirnov ^a			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Semana_0	,247	5	,200 [*]	,876	5	,293
Semana_1	,238	5	,200 [*]	,940	5	,663
Semana_2	,157	5	,200 [*]	,981	5	,942

*. Esto es un límite inferior de la significación verdadera.

a. Corrección de significación de Lilliefors

Criterio de decisión

Si $P < 0.05$ se rechaza la hipótesis nula y se acepta la hipótesis alternativa.

Si $P \geq 0.05$ se acepta la hipótesis nula y se rechaza la hipótesis alternativa.

Validación de Hipótesis – Prueba de Normalidad

Decisión y conclusión

Dado que en todos los casos se obtuvo un valor de significancia $P \geq 0.05(*)$, se acepta la hipótesis nula y se rechaza la hipótesis alternativa, es decir los datos tienen una distribución normal, por lo tanto, se aplica el método paramétrico T student de muestras relacionadas.

Validación de Hipótesis – Prueba T Student

Planteamiento de la hipótesis

(H_o) = La implementación de un panel de entrenamiento automático no contribuirá en el desarrollo de los reflejos cognitivos y físicos de los deportistas de artes marciales mixtas en la empresa MIDY Scientific Trainers en la ciudad de Quito.

(H_i) = La implementación de un panel de entrenamiento automático contribuirá en el desarrollo de los reflejos cognitivos y físicos de los deportistas de artes marciales mixtas en la empresa MIDY Scientific Trainers en la ciudad de Quito.

Validación de Hipótesis – Prueba T Student

Prueba de muestras emparejadas

		Diferencias emparejadas				t	gl	Significación		
		Media	Desv. estándar	Media de error estándar	95% de intervalo de confianza de la diferencia			P de un factor	P de dos factores	
					Inferior					Superior
Par 1	Semana_0 - Semana_1	-,21600	,12779	,05715	-,37467	-,05733	-3,780	4	,010	,019
Par 2	Semana_1 - Semana_2	-,39800	,09808	,04386	-,51978	-,27622	-9,074	4	<,001	<,001
Par 3	Semana_0 - Semana_2	-,61400	,15884	,07104	-,81123	-,41677	-8,644	4	<,001	<,001

Criterio de decisión

Si $P < 0.05$ existen diferencias significativas en las muestras relacionadas

Si $P \geq 0.05$ no existen diferencias significativas en las muestras

relacionadas

Validación de Hipótesis – Prueba T Student

Decisión y conclusión

Dado que en todos los casos se obtuvo un valor de significancia $P < 0.05(*)$, se determina que existen diferencias significativas entre los valores de velocidad de reacción (reflejos) en todos los sujetos de prueba, entre los periodos de valoración iniciales y finales, favoreciendo a los valores obtenidos en la semana 2 (final) del entrenamiento, por lo tanto, se acepta la hipótesis alternativa, es decir, la implementación del panel de entrenamiento automático contribuirá en el desarrollo de los reflejos cognitivos y físicos de los artistas marciales, en la empresa MIDY Scientific Trainers de la ciudad de Quito.

Conclusiones

Se implementó un panel de entrenamiento automático que beneficia el desarrollo de la velocidad de reacción (reflejos) de los artistas marciales, basado en un sistema embebido de inteligencia artificial y la adquisición de datos mediante una cámara 3D, además, el sistema es capaz de obtener información del entorno y almacenarla en una base de datos.

Una vez realizada la investigación bibliográfica acerca de los sistemas de entrenamiento automáticos para la mejora de reflejos, se encontró que existe una variedad de dispositivos en el mercado los cuales permitieron fundamentar un nuevo diseño con innovaciones incorporadas.

Se diseñó el modelo CAD del panel de entrenamiento automático, teniendo en cuenta los requerimientos de cada componente y la aplicación para la que iban a ser utilizados.

Conclusiones

Se realizó un análisis del factor de seguridad de los resortes helicoidales para determinar si estos cumplen con las necesidades del sistema.

Se desarrolló un algoritmo de programación basado en visión artificial, capaz de detectar los puntos clave del cuerpo de una persona (manos y pies) y utilizar esta información para controlar el comportamiento del sistema.

Se desarrolló una interfaz gráfica en Python, esta permite la interacción del usuario con el panel de entrenamiento y sus sistemas, además se enlazó una base de datos al HMI, de modo que, los datos obtenidos por el sistema, se pueden visualizar directamente en la interfaz.

Conclusiones

Mediante pruebas de entrenamiento, se verificó el funcionamiento de todos los sistemas del panel automático y se validaron los datos de velocidad que éste proporciona a través de un experimento comparativo con el sistema profesional STT System.

Se realizó un análisis estadístico con los datos obtenidos por un conjunto de deportistas que entrenaron con el panel automático durante dos semanas, lo que permitió validar que el sistema contribuirá al desarrollo de la velocidad de reacción (reflejos) de los artistas marciales.

Se determinó que en todos los casos se obtuvo un valor de significancia $P < 0.05$, el cual determina que existen diferencias significativas entre los valores de velocidad de reacción (reflejos) en todos los sujetos de prueba, entre los periodos de valoración iniciales (semana 0) y finales (semana 2), en donde al término del experimento se evidenció un aumento de aproximadamente el 50% en la velocidad de reacción.

Recomendaciones

Se debe mantener la iluminación adecuada en el entorno controlado y limitar a una la cantidad de personas que aparecen en escena, para evitar lecturas erróneas o generar conflictos en la detección de puntos clave.

En el proceso de registro de un nuevo usuario, se debe utilizar únicamente correos electrónicos del dominio de Google, puesto que la plataforma Firebase Authentication solo admite este tipo de cuentas.

Se necesita mantener la cámara 3D a la distancia pre establecida, es decir a dos metros del extremo izquierdo de la estructura rectangular, para que los datos calculados por el sistema no tengan variaciones.

Recomendaciones

Se recomienda que la conexión de internet en la computadora Jetson Nano permanezca activa durante el entrenamiento, para que el sistema sea capaz de interactuar con la base de datos en la nube.

Una vez inicializado el sistema embebido, es indispensable abrir la terminal del sistema UBUNTU y ejecutar el código correspondiente, para la activación del ventilador de la computadora Jetson Nano, con el objetivo de evitar fallas y/o daños provocados por altas temperaturas generadas debido al uso del hardware.

Se requiere dar mantenimiento a los resortes helicoidales y la lona de soporte cada vez que se visualice un desgaste provocado por la utilización constante del panel de entrenamiento automático.

Recomendaciones

Se recomienda ejecutar al menos 3 entrenamientos a la semana para observar cambios significativos en la velocidad de reacción (reflejos).

GRACIAS!



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA