



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**Desarrollo de un sistema de
entrenamiento para la colocación de
dispositivos trocares en la cirugía
laparoscópica**

TESIS

Que para obtener el título de

Ingeniero en Mecatrónica

P R E S E N T A

Emmanuel Sánchez Ortiz

DIRECTOR DE TESIS

Dr. Fernando Pérez Escamirosa



Ciudad Universitaria, Cd. Mx., 2023

**Desarrollo de un sistema
de entrenamiento para la
colocación de dispositivos
trocares en la cirugía
laparoscópica**

Agradecimientos

Agradezco al Instituto de ciencias aplicadas y tecnología por brindarme la oportunidad de realizar mi trabajo de tesis, además extendiendo mi agradecimiento al programa UNAM-DGAPA-PAPIME, con número de proyecto PE217020, por brindarme el apoyo para la generación de la presente tesis de Licenciatura.

Mi más extensa admiración, reconocimiento y agradecimiento al Dr. Fernando Pérez Escamiroso por su iniciativa, consejos, motivación pero ante todo paciencia brindada en mi desarrollo profesional así como en la asesoría al realizar el presente trabajo, sin su valioso apoyo este trabajo no se habría logrado realizar, gracias por acompañarme en cada fase de la creación del presente trabajo y brindarme asesoría cada vez que fue necesario, sus observaciones me han brindado un abanico de posibilidades y han sido un fuerte referente para futuros trabajos.

Así mismo brindaré el siguiente espacio para agradecerle a cada uno de los miembros de mi familia, a mi papá, Francisco por brindarme consejos y palabras de aliento en cada fase de mi vida, especialmente en los momentos más complicados estar ahí para mí así como ser un gran referente del mundo académico, mi mamá Magdalena por ser ese ejemplo de carácter, fuerza, entrega y dedicación, pero sobre todo por enseñarme el valor de la resiliencia, a mis padres ambos les agradezco infinitamente su apoyo en todo momento desde mis decisiones más difíciles hasta mis éxitos, sin ustedes y su apoyo mis proyectos no habrían sido lo mismo.

A mis hermanos, Yoloxochitl, por brindarme momentos de compañía, escucharme en mis etapas más difíciles y siempre velar por mi salud tanto física como mental, gracias por tu compañía, tus recordatorios y por cuidar de mi en todo momento te agradezco tu sinceridad, cariño, apoyo y hermandad, gracias por tu admiración y entusiasmo.

Tonatiuh, muchas gracias por tu compañía y apoyo, gracias por cuidar de mi y aconsejarme a tu manera, entenderme y motivarnos mutuamente desde lo deportivo hacia otras áreas del conocimiento, gracias por tu apertura y confianza, gracias por brindarme cariño en todo momento, así como fortaleza en mis momentos más complicados.

A mis hermanos mi eterno agradecimiento por ser mis principales referentes de confianza y amor, gracias por enseñarme sus diferentes formas de pensar y ayudarme en mi crecimiento personal, gracias a ustedes he aprendido mucho de la vida, son un eje fundamental en cada uno de mis pasos.

De manera general quiero agradecer a todas las personas que me han acompañado a lo largo de mi vida, toda la admiración y respeto que se tiene hacia mi persona es totalmente reciproca, gracias a mis profesores y Sensei por compartir conmigo mi gran pasión por el arte marcial, así como enseñarme, orientarme, corregirme y motivarme. Sin lugar a dudas, los valores marciales me han forjado un carácter y liderazgo que puedo exportar en todas las actividades de mi vida personal y profesional, gracias por todos los años de entrenamiento y velar por mi formación tanto académica como deportiva.

Gracias a todos aquellos que han aportado un pequeño espacio para convivir, charlar, jugar, y cuidar de mi en mis diferentes etapas de vida, especialmente mi transición desde la Preparatoria – Universidad al mudarme de estado y vivir solo, persiguiendo mis sueños, gracias por su amabilidad, confianza y apoyo, siempre tendrán un lugar especial en mi vida.

Contenido

Agradecimientos	iii
Contenido	v
Índice de figuras	viii
Índice de tablas	x
Resumen	xi
Abstract	xii
Capítulo 1 Introducción	1
1. Cirugía mínimamente invasiva: Laparoscopia	1
1.1. Desarrollo de la cirugía mínimamente invasiva, énfasis en laparoscopia	2
1.2. Cirugía Laparoscópica	3
1.3. La cirugía laparoscópica en México	5
1.4. Ventajas y desventajas de CMI abdominal	6
1.5. Planteamiento del problema	7
1.1. Estructura de la tesis	8
1.2. Justificación	11
1.3. Objetivos	12
1.3.1. Objetivo general	12
1.3.2. Objetivos específicos	12
Capítulo 2 Estado del arte	13
2. Simuladores en cirugía laparoscópica	13
2.1. Introducción de la aguja de Veress	13
2.2. Trocar como instrumental de acceso	15
2.3. Simuladores Físicos	17
2.3.1. Entrenador Laparoscópico con Neumoperitoneo	17
2.3.2. Características de fuerza en la inserción abdominal del trocar en cirugía laparoscópica	19
2.3.3. El trocar inteligente: Trocar de detección de fuerza, desviación e impedancia para mejorar la cirugía Laparoscópica	21
2.3.4. Adquisición interactiva 3D del parámetro de Fuerza/Torque e identificación de la correlación durante la inserción del trocar primario en cirugía laparoscópica abdominal: 5 Casos	24

2.3.5.	Un nuevo dispositivo para la inserción segura del trocar en cirugía laparoscópica basado en las características de fuerza	25
2.4.	Simuladores Virtuales	27
2.4.1.	Simulación del neumoperitoneo en un paciente específico para la planeación de la cirugía laparoscópica	27
Capítulo 3 Metodología		32
3.1	Criterios generales de diseño	32
3.2.	Sensores	32
3.2.1.	Sensor Fuerza	33
3.2.1.1.	Colocación del sensor en el trocar	34
3.2.2.	Sensor de presión	36
3.2.3.	Acondicionamiento de señal	38
3.2.3.1.	Sensor de fuerza	38
3.2.3.2.	Sensor de presión	43
3.3.	Especificaciones generales	45
3.3.1.	El convertidor analógico – digital (ADC)	45
3.3.2.	Convertidor ADS1115	47
3.3.3.	Comparativa convertidor Arduino y comparador ADS 1115	49
3.4.	Calibración de sensores	49
3.4.1.	Calibración sensor de presión	49
3.4.1.1.	Fases de obtención de datos	50
3.4.2.	Calibración del sensor de fuerza	53
3.5.	Placa de desarrollo	56
3.5.1.	Sistema de adquisición de datos	56
3.5.2.	Diseño de diagrama eléctrico y simulación	58
3.5.3.	Diseño electrónico en placa PCB	60
3.6.	Sistema de insuflación	62
3.6.1.	Sistema de dosificación del gas	63
3.7.	Programación	65
3.7.1.	Desarrollo de código aplicado al microcontrolador	65
3.7.2.	Desarrollo de la interfaz gráfica (python)	67
3.7.3.	Base de datos	75
3.8.	Desarrollo de entorno gráfico	76

3.8.1. Diseño de interfaz gráfica	76
3.8.2. Estructura de vistas	77
3.8.2.1. Menú	77
3.8.2.2. Acceso de usuario (Log In)	78
3.8.2.3. Registro de usuario	79
3.8.2.4. Plataforma	79
3.9. Diseño de la interfaz de entrenamiento	83
3.9.1. Diseño Mecánico de interfaz externa	83
3.9.2. Adaptación pad de entrenamiento	86
Capítulo 4 Resultados	87
4.1. Dispositivo final	87
4.1.1. Dispositivo de entrenamiento	87
4.1.2. Plataforma de entrenamiento.	90
4.2. Instrucciones de uso y configuración de la plataforma de entrenamiento	92
4.3. Generación de pruebas	94
4.4. Demostración de funcionamiento	101
Capítulo 5 Conclusiones	111
5.1 Trabajo a futuro	112
Índice de figuras de apéndices y anexos	114
Índice de tablas de apéndices y anexos	115
Apéndice	116
Apéndice I Vista Isométrica diseño del entrenador	116
Apéndice II Vista Isométrica diseño del cierre semihermético	117
Apéndice III Vista Isométrico caja para soporte electrónico y caja de soporte al insuflador	118
Apéndice IV Tablas informativas	120
Apéndice V Código de lectura de valores analógicos	122
Apéndice VI Código controlador del insuflador, monitoreo de sensores y envío de datos	123
Apéndice VII Código programa principal (Interfaz Gráfica)	124
Apéndice VIII Programa calibración sensor de presión	135
Anexos	136

Anexo 1 Glosario	136
Referencias	137

Índice de figuras

Figura 1 A Equipo de cirujanos, B Laparoscopio e instrumental quirúrgico, C Punto de acceso y ubicación del trocar, D Torre laparoscópica E Insuflador F Monitor G Fuente de luz	5
Figura 2 Aguja de Veress. Imagen extraída de [43] (Saxena & Höllwarth, 2008)	14
Figura 3 Técnica correcta de inserción del trocar. Imagen extraída de [49] (Sociedad Argentina de Cirugía Digestiva, 2022)	16
Figura 4 Entrenador laparoscópico con neumoperitoneo de [23] (L. Ortiz Simón et al., 2012)	18
Figura 5 Diseño mecánico. Imagen extraída de [52] (Sun & Tadano, 2017)	19
Figura 6 Diseño de plataforma. Imagen extraída de [52] (Sun & Tadano, 2017)	19
Figura 7 Resultados del experimento. Imagen extraída de [52] (Sun & Tadano, 2017)	21
Figura 8 Trocar. Imagen extraída de [44] (Schrope et al., 2019)	21
Figura 9 Diseño del trocar inteligente. Imagen extraída de [44] (Schrope et al., 2019)	22
Figura 10 Desviación del trocar. Imagen extraída de [44] (Schrope et al., 2019)	23
Figura 11 Adaptación del trocar primario. Imagen extraída de [32] (Nillahoot et al., 2022)	24
Figura 12 Comparativa entre 2 casos. Imagen extraída de [32] (Nillahoot et al., 2022)	25
Figura 13 Configuración del dispositivo. Imagen extraída de [53] (Sun & Tadano, 2019)	26
Figura 14 Evaluación de la inserción del trocar de forma manual y automática. Imagen extraída de [53] (Sun & Tadano, 2019)	27
Figura 15 Comparación de fuerza en cada grupo. Imagen extraída de [53] (Sun & Tadano, 2019)	27
Figura 16 Imagen preoperativa extraída de [5] (Bano et al., 2012)	28
Figura 17 Diagrama modelo maestro – esclavo	29
Figura 18 Evaluación visual del error promedio, Imagen extraída de [5] (Bano et al., 2012)	30
Figura 19 Sensor Flexiforce Imagen Extraída de [54] (Tekscan, s. f.)	33
Figura 20 Vista lateral manejo de trocar, imagen extraída de [37] (Punto Quirurgico, 2018).	35
Figura 21 Visa superior del manejo de trocar, imagen extraída de [37] (Punto Quirurgico, 2018).	35
Figura 22 Colocación del sensor de fuerza en el trocar	36
Figura 23 Sensor de presión. Imagen extraída de [35] (NXP Semiconductors, 2018)	37
Figura 24 Sensor de presión acoplado 1 Acoplamiento neumático para adquirir la presión al interior del dispositivo de entrenamiento, 2 Sensor MP3V5050, 3 PCB para el montaje del sensor, 4 (Vcc) voltaje de alimentación, 5 GND, 6 (Vout) Señal de salida del sensor.	38
Figura 25 Circuito acondicionador de señal para flexiforce. Imagen extraída de [54] (Tekscan, s. f.)	39
Figura 26 Simulación en proteus del circuito acondicionador de señal ante un estímulo	41
Figura 27 Señal del Sensor de Fuerza sin carga.	42
Figura 28 Señal del sensor de Fuerza con carga Máxima.	42
Figura 29 Diagrama de bloques del sensor MP3V5050GP. Imagen extraída de [35] (NXP Semiconductors, 2018)	43
Figura 30 Gráfica de la lectura de presión sin activar la bomba	44
Figura 31 Gráfica de la lectura de presión 14.9 mmHg	44
Figura 32 Diagrama de bloques general de un ADC [6] (Bolton et al., 2013)	45
Figura 33 4 Fases en la conversión [6] (Bolton et al., 2013)	46
Figura 34 Cuantización de un ADC [6] (Bolton et al., 2013)	47

Figura 35 Estructura general del convertidor ADS1115. Imagen extraída de [56] (Texas Instruments, 2018)	47
Figura 36 Flujo de adquisición en fase de prueba.	50
Figura 37 Calibración y obtención de los datos.	51
Figura 38 Arduino nano	58
Figura 39 Diagrama esquemático.	59
Figura 40 Diagrama pcb	61
Figura 41 Fuente de alimentación.	62
Figura 42 Variación de corriente en función de la entrada de voltaje aplicado, utilizando el IRL 520. Imagen extraída de la hoja de datos del fabricante. Imagen extraída de [20] (International Rectifier, s. f.)	63
Figura 43 Diagrama de flujo correspondiente al código aplicado al microcontrolador.	65
Figura 44 Diagrama de clases del archivo "main.py" primera sección	68
Figura 45 Diagrama de clases del archivo "main.py" segunda sección	69
Figura 46 Diagrama del archivo user.py	74
Figura 47 Diagrama del archivo FindPorts.py	74
Figura 48 Diagrama del archivo ArduinoCommunication.py	75
Figura 49 Diagrama entidad relación	75
Figura 50 Entorno gráfico de Qt Designer.	76
Figura 51 Vista Menú.	77
Figura 52 Segunda vista: Log In	78
Figura 53 Tercera vista: Registro	79
Figura 54 Vista Plataforma	81
Figura 55 Pestaña Proceso	82
Figura 56 Pestaña fuerza y presión	82
Figura 57 Pestaña Resultados	83
Figura 58 Diseño de interfaz de entrenamiento.	84
Figura 59 Abrazadera cierre semi-hermético.	84
Figura 60 orificios de entrada/salida del flujo de aire, 1) Muestra la entrada de aire proveniente de la bomba y 2) orificio de salida hacia el sensor de presión.	85
Figura 61 Pad de entrenamiento	86
Figura 62 Dispositivo de entrenamiento	88
Figura 63 Insuflador	88
Figura 64 A Tarjeta de adquisición de datos, B Fuente de alimentación y C Sensor de presión, D Switch de control.	89
Figura 65 Trocar y sensor de fuerza	90
Figura 66 Gráfica final de presión	91
Figura 67 Gráfica final de fuerza	92
Figura 68 Sección de calibración	94
Figura 69 Búsqueda de puertos	100
Figura 70 Conexión exitosa del puerto serie	100
Figura 71 Separación del entrenador físico	101
Figura 72 Adaptación del pad de entrenamiento	102
Figura 73 Inserción de abrazaderas	102
Figura 74 Cierre semiermético, adición de tuercas de seguridad.	103
Figura 75 Inserción de tuercas mariposa	104
Figura 76 Preparación del A Entrenador, B insuflador y C Trocar	104
Figura 77 Pantalla inicio de sesión	105

Figura 78 Pestaña Inicio de la plataforma.	106
Figura 79 Configuración presión a 12 mmHG	107
Figura 80 A Preparación del trocar, B Técnica correcta de sujeción, C Punción, D Trocar colocado.	108
Figura 81 Pestaña Fuerza posterior al entrenamiento	109
Figura 82 Pestaña Presión posterior al entrenamiento	110
Figura 83 Pestaña Resultados posterior al entrenamiento	110

Índice de tablas

Tabla 1 Parámetros para el establecimiento del neumoperitoneo. Extraído de [22] (L. Ortiz Simón et al., 2012)	17
Tabla 2 Errores promedio, Tabla extraída de [5] (Bano et al., 2012)	30
Tabla 3 Características físicas sensor Flexiforce.	34
Tabla 4 Características eléctricas sensor Flexiforce.	34
Tabla 5 Características eléctricas del sensor de presión	37
Tabla 6 Ganancia programable, extraída de [10] (Del Valle, 2022)	48
Tabla 7 Comparativa entre ADC Arduino y ADS 1115	49
Tabla 8 Relación entre la presión registrada y la lectura obtenida en el conversor analógico digital.	52
Tabla 9 Datos fuerza en Newtons.	54
Tabla 10 Relación entre la lectura del sensor contra la fuerza registrada.	55
Tabla 11 Componentes del diagrama esquemático.	60
Tabla 12 variación de PWM con presión generada.	64
Tabla 13 Tercera sección código hacia Arduino.	66
Tabla 14 Tabla de contenidos de la clase userWindow.	73
Tabla 15 Descripción de pestañas de la vista plataforma	80
Tabla 16 Descripción de las pruebas de validación	99

Resumen

Introducción: La cirugía mínimamente invasiva se vuelve significativa en 1901, sin embargo, el desarrollo de la cirugía laparoscópica presenta importantes avances hasta 1986. Con lo anterior las nuevas técnicas de intervención quirúrgica tuvieron la necesidad de ser practicadas a través de simuladores, cajas de entrenamiento y demás dispositivos de apoyo.

El objetivo del presente proyecto de tesis es realizar un entrenador funcional que permita combinar el proceso de insuflación con la práctica de la inserción del trocar en un ambiente físico que simula el acceso a una cavidad.

Métodos: El presente documento describe el desarrollo de un sistema integrado de entrenamiento dedicado a una de las etapas cruciales en cirugía laparoscopia, la inserción del trocar, este objetivo se logra con el desarrollo de un sistema físico de entrenamiento. Por lo tanto, utilizando un sistema electrónico, las señales involucradas en este proceso (fuerza y presión), son adquiridas para ser procesadas, interpretadas y finalmente mostradas como resultado del proceso de entrenamiento, además la información es almacenada en una base de datos para futuras consultas.

De lo anterior expuesto, el presente desarrollo incluye una plataforma multiusuario la cual crea la interacción entre los sensores, computadora y usuario final, para finalmente generar una gráfica de ambas señales (fuerza y presión) y ser observadas en una tabla.

Resultados: El sistema de entrenamiento contiene una plataforma multiusuario, la cual realiza el control del insuflador, la adquisición y almacenamiento de datos, el registro de usuarios, el acceso y validación de cada usuario, la generación de gráficas de resultados y el cálculo de la fuerza máxima registrada.

Dicho sistema cuenta con una placa electrónica para la alimentación de los sensores e insuflador, control del motor correspondiente al insuflador, el cual puede manipular presiones desde 6 hasta 16 mmHg.

En la placa electrónica se tienen los 2 sensores conectados al microcontrolador, el primero es un sensor de fuerza el cual tiene un rango de operación desde 0 hasta 23.4 N, mientras que el segundo es un sensor de presión capaz de registrar lecturas entre 0 y 50 Kpa (375.031 mmHg).

El entrenador físico es un dispositivo cilíndrico semihermético capaz de albergar diferentes pads de entrenamiento de entre 9.5 y 10 cm de diámetro, contando con 2 abrazaderas externas, tuercas de seguridad y tornillos mariposa que brindan un cierre adecuado para cada sesión de entrenamiento.

Conclusión: El presente sistema de entrenamiento es un medio seguro para mejorar las habilidades psicomotrices de los médicos aprendices puesto que comprende la medición de la calidad de su punción a través de la obtención de la fuerza empleada, así como el cambio de presión durante la sesión de entrenamiento dichos datos son almacenados y graficados. Además, se cuenta con un medio físico para poder utilizar un trocar real acompañado de su sistema de insuflación el cual permite manipular diferentes presiones, dichos elementos crean una simulación realista de un entorno de operación.

Abstract

Introduction: Minimally invasive surgery becomes meaningful in 1901, but laparoscopic development had significant advances until 1986. With the foregoing, new surgery intervention techniques have the necessity to practice this new knowledge through simulators, training boxes, or other support devices.

The aim of this thesis project is to create a functional trainer which allows combining the insufflation process with the trocar insertion practices, through a real trocar inside one environment that simulates the cavity access.

Methods: This document describes the development of an integrated training system for one of the critical stages of laparoscopic surgery, trocar insertion, this target was reached by the development of a physical training system.

Therefore, using an electronic system the signals involved in this process (force and pressure) are acquired to be processed, interpreted, and finally show the results of a training process, furthermore, the information is stored in a database for further consults.

From the above, this development includes a multiuser platform that makes the interaction between sensors, computer, and final user, to finally generate a graph of both signals (force and pressure) and watch it on a grid table.

Results: System training contains a multiuser platform that makes the insufflator control, data storage and acquisition, user register, access, and validation for each user, results graphics generation, and force calculus of the maximum registered value.

This system was made on an electronic board to insufflator and sensors supply as well as the motor's control, this motor can manipulate pressure range from 6 to 16 mmHg. On this electronic device we have a microcontroller with 2 sensors connected, first is the force sensor which has an operation rate from 0 to 23.4 N, on the other side we have a pressure sensor able to register data from 0 to 50 Kpa(375.031 mmHg).

A physical trainer is a semi-hermetic cylindric device capable of harnessing different training pads made with a diameter between 9.5 and 10 cm. This device has 2 external brackets, security nuts, and butterfly screws that create the proper close for each training session.

Conclusions: The current training system is a secure environment to improve the trainee doctors' psychometrics abilities since it allows the quality puncture measure through used force data and the pressure fluctuation inside the training session where data is stored and plotted. Moreover, this system has a physical environment with the aim of using a real trocar with an insufflation system to manipulate different pressure values. Under the previous conditions, the system emulates a realistic environment of surgery.

Capítulo 1 Introducción

1. Cirugía mínimamente invasiva: Laparoscopia

Si bien es cierto que los procedimientos médicos han evolucionado con el paso del tiempo, siendo la tecnología un fuerte aliado que permite el avance de técnicas quirúrgicas novedosas para explorar el cuerpo humano y con ello obtener procedimientos mejor estructurados, así como mejores resultados estéticos en el paciente.

Al hablar de la cirugía entendemos que dicho término ha sido tradicionalmente utilizado para describir los procedimientos quirúrgicos que implican realizar incisiones o suturar tejidos en el cuerpo humano para tratar enfermedades, lesiones o deformidades. En algunas ocasiones, se utiliza el láser, la radiación u otras técnicas (en lugar del bisturí) para cortar tejidos. Sin embargo, estas otras técnicas las abordaremos más adelante para explorar un poco de la gama de opciones que tenemos dada la creciente ola de innovación en el desarrollo del equipo médico para las cirugías.

Actualmente, existen diferentes tipos de procedimientos quirúrgicos debido a que la cirugía es un área sumamente extensa, la cual incluye diferentes técnicas como lo son:

1. Extirpar tejido como un absceso o un tumor.
2. Abrir o desbloquear obstrucciones.
3. Conectar arterias y venas a nuevas posiciones.
4. Trasplantar, se pueden retirar del cuerpo órganos tales como la piel, los riñones o el hígado que luego se transfieren a otra parte del mismo cuerpo o a otro organismo diferente.
5. Injertos los cuales ocupamos para reemplazar vasos sanguíneos o tejido conjuntivo.
6. Biopsias que se utilizan para ayudar a establecer un diagnóstico, extrayendo una porción de tejido.

Hoy en día, los avances científicos y tecnológicos han permitido realizar intervenciones con incisiones más pequeñas y con menor rotura de tejido que en la cirugía tradicional. Lo anterior es gracias a nuevas técnicas quirúrgicas que engloban lo que se denomina: cirugía mínimamente invasiva.

La cirugía mínimamente invasiva recibe varios nombres en función de la zona dónde se realice: laparoscopia en el abdomen, artroscopia en las articulaciones y toracoscopia en el tórax citando algunos ejemplos. En el presente trabajo, abordaremos únicamente la cirugía laparoscópica que se realiza en la cavidad abdominal de un paciente.

1.1. Desarrollo de la cirugía mínimamente invasiva, énfasis en laparoscopia

No podemos adentrarnos al entorno de la laparoscopia sin antes abordar brevemente un poco de la historia que comprende en desarrollo de la cirugía mínimamente invasiva.

Fue George Kelling quien utilizó el cistoscopio urológico, con la finalidad de inspeccionar el contenido intestinal, introduciendo dicho instrumento a través de un orificio abierto en la pared abdominal de un perro. Dicha técnica fue denominada como “celioscopia”, y cuyos resultados fueron presentados en el Congreso de la Sociedad Médica y de Biología germana, en Hamburgo en septiembre de 1901, siendo el presente hecho una de las primeras aproximaciones al desarrollo de la técnica laparoscópica.

En lo sucesivo, los continuos adelantos y mejoras en los medios ópticos fue lo que permitió incrementar de un modo notable la calidad de la visión; la introducción de las fuentes de luz fría y su transmisión por medio de fibra óptica en 1966, el uso de lámparas de alta intensidad lumínica, de halógenas de xenón, el perfeccionamiento en el diseño de nuevos aparatos junto a su miniaturización, así como la incorporación de aparatos compuestos de microcámaras de video (que en un futuro conoceríamos como endoscopios) permitieron su incorporación al desarrollo de técnicas quirúrgicas más precisas y con mejores resultados estéticos.

En el año de 1986 Wickham, director del instituto de Urología de Londres, sugiere el término de “Cirugía Mínimamente Invasiva”, en adelante (CMI). Dicho término fue aceptado internacionalmente y ampliado en su concepto al “conjunto de técnicas que, sirviéndose de un instrumental muy específico evitan casi total o totalmente las lesiones inherentes a la vía de acceso quirúrgico, reducen la

morbilidad operatoria y el periodo de convalecencia, al tiempo que consiguen unos resultados superponibles o incluso mejores a los que se obtienen con la cirugía abierta “convencional”. (Albacete, 2005).

De lo anterior podemos rescatar la aplicación directa del desarrollo tecnológico en la cirugía laparoscópica, puesto que como podemos apreciar, a medida que existen avances científicos, encontramos por consiguiente un campo de aplicación, dónde paso a paso se van solucionando diferentes problemas a través del refinamiento de aparatos existentes y la creación de nuevos métodos y aparatos auxiliares. Esto deriva en la adopción de nuevas técnicas aplicables a distintos tipos de intervenciones quirúrgicas, mejorando así el desarrollo propio de la actividad tecnológica en este tipo de cirugía.

1.2. Cirugía Laparoscópica

Debido a los presentes antecedentes que abarcan tanto el desarrollo de los distintos procedimientos quirúrgicos que se utilizan en la CMI, como el desarrollo de la cirugía laparoscópica, podemos tener las herramientas necesarias para detallar más acerca del presente procedimiento, así como los elementos que componen el mismo. (Figura 1)

Inicialmente, la raíz griega láparos significaba “suave o flojo” y cambió a lápara para referirse a los flancos del abdomen posiblemente por su blandura, pero su uso evolucionó finalmente para significar abdomen. Por su parte, el verbo griego skopó significa “mirar u observar”. De la combinación de ambos vocablos surge la palabra laparoscopia que significa “ver dentro del abdomen” (Ruiz et al., 2016).

La cirugía laparoscópica consiste en poder realizar las operaciones de los diferentes órganos abdominales sin abrir el abdomen o realizar grandes incisiones como lo sugiere la cirugía clásica convencional. Para lograr lo anterior, se utiliza un instrumento que permite la visión de tejidos, así como órganos internos: laparoscópio o endoscopio.

Dicho instrumento alberga en su interior tanto fibras ópticas para la transmisión de luz como la óptica de visualización. Además, puede albergar un campo de visión desde 60° hasta 82°, mientras que su ángulo de visión puede estar disponible en 5°, 25°, 30°, 45° e incluso 70° lo que permite una visualización de tejidos que se encuentran en posiciones difíciles de acceder (Saxena & Höllwarth, 2008). En cuanto al diámetro exterior, tenemos diferentes presentaciones que van desde 1.9 mm hasta 12 mm, dependiendo del procedimiento quirúrgico al que serán expuestos.

Con el uso de dicho dispositivo se puede aumentar la imagen hasta 20 veces el tamaño normal, siendo el resultado final apreciado en un monitor. Como el interior del abdomen es oscuro, se tiene que agregar luz a través de mismo laparoscópio utilizando una fuente de luz fría, transmitida a través de fibra óptica. Asimismo, debido a la proximidad de los órganos se tiene que agregar un insuflador que es un aparato que ingresa gas CO₂ por una aguja previamente colocada en la pared abdominal; dicho aparato constituye una parte central del presente trabajo el cual estaremos abordando en el capítulo posterior.

Con la cavidad abdominal visualizada perfectamente, el siguiente paso es la introducción de las herramientas y material quirúrgico dentro de la cavidad, siendo los dispositivos trócares uno de ellos, cuya longitud es de entre 80 a 110 mm (generalmente) y su diámetro dependiendo de la intervención quirúrgica va entre 2 [mm] y 15 [mm]. Estos dispositivos que van a permitir la entrada y salida de los diversos instrumentos quirúrgicos laparoscópicos.

Estos instrumentos trócares constan de dos partes: 1) el punzón que atraviesa la pared abdominal, y 2) la camiseta o funda que queda para la parte operativa; dicha funda permite la introducción de los instrumentos largos y delgados sin perder la presión del gas CO₂ durante el neumoperitoneo (procedimiento con el cual se regula la insuflación).

La presión con la cual se regulará el neumoperitoneo debe permanecer entre los 15 mmHG (al momento de realizar la instalación de todos los instrumentos) y 12 ó 10 mmHG al estar realizando la operación. De esta forma, se evitan complicaciones en la capnografía (CO₂ exhalado) y capnemia (CO₂ en sangre), así como evitar desgarre de tejidos, úlceras, entre otras lesiones y riesgos que se abordarán con mayor detenimiento en el siguiente capítulo.

Una vez terminado el procedimiento, se aspira el neumoperitoneo y se suturan los puertos de entrada de los trócares que se hubieran realizado en el abdomen. El proceso de inserción de los trócares representará, en nuestro trabajo, un objetivo de estudio con lo cual se desarrollará un sistema de entrenamiento para estos dispositivos.

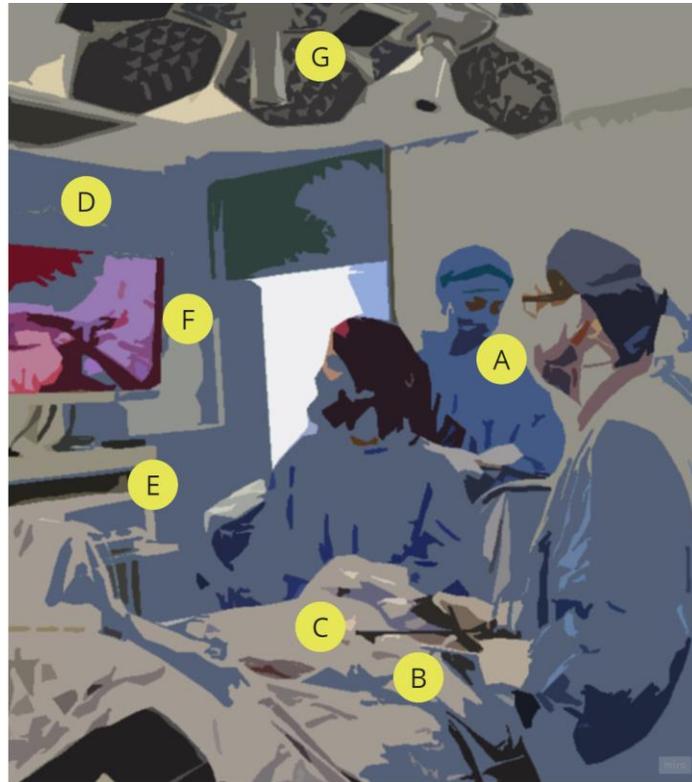


Figura 1 A Equipo de cirujanos, B Laparoscopio e instrumental quirúrgico, C Punto de acceso y ubicación del trocar, D Torre laparoscópica E Insuflador F Monitor G Fuente de luz

1.3. La cirugía laparoscópica en México

En México, las técnicas laparoscópicas aparecen de mano del cirujano general y endoscopista Leopoldo Gutiérrez Rodríguez, quien realizó la primera colecistectomía laparoscópica en territorio mexicano, el 29 de junio de 1990. Durante los primeros meses de 1990, se le presentaron dos oportunidades para realizar el primer caso; sin embargo, en ambos se tuvo que resolver de forma convencional debido a dificultades técnicas, siendo hasta su tercer intento, al interior del Hospital General de Zona No. 8 del Instituto Mexicano del Seguro Social (IMSS), que logró operar con éxito.

Fue así como la colecistectomía laparoscópica se inició en México, generando un interés entre los cirujanos generales de la última década del siglo XX, que aún con muy poca experiencia y casi sin guía decidieron incursionar en este nuevo abordaje quirúrgico de mínima invasión, lo cual trajo consigo la aparición de una serie de complicaciones quirúrgicas relacionadas con la falta de habilidades, destrezas y poca experiencia de esos cirujanos.

Sin embargo, manifestando un interés auténtico en la educación médica, y para poder expandir los beneficios de la laparoscopia a más pacientes, 18 cirujanos mexicanos, pioneros en la cirugía laparoscópica se reunieron el 26 de junio de 1991, con el ánimo de compartir experiencias y así fundar la Asociación Mexicana de Cirugía Laparoscópica.

Posteriormente, se llevaron a cabo los primeros cursos de cirugía laparoscópica en diferentes hospitales de la Ciudad de México a los que vinieron profesionales de calidad internacional. Tiempo después dichos cursos se extendieron al resto del país.

Con el objetivo de poder albergar las diferentes subespecialidades que utilizaron el abordaje laparoendoscópico para realizar sus respectivas cirugías, el nombre de la asociación se cambió en el año 2000 por el de Asociación Mexicana de Cirugía Endoscópica (AMCE), así de esta forma se dio cauce oficial a la capacitación de los cirujanos mexicanos en las técnicas endoscópicas, misión que hasta la fecha siguen cumpliendo (Ruiz et al., 2016).

1.4. Ventajas y desventajas de CMI abdominal

La cirugía de mínima invasión se ha convertido en un gran adelanto terapéutico, que reduce los costos derivados de las estancias hospitalarias prolongadas innecesarias, con el cual se consigue aumentar la satisfacción y comodidad del paciente, llevando a cabo procedimientos a través de incisiones cada vez más pequeñas. Estas ventajas que ofrece se enuncian a continuación:

- 1) Menor dolor posterior a la cirugía:
Permite a los pacientes levantarse y caminar a las pocas horas de la cirugía.
- 2) Estancia hospitalaria corta:
Si no hay complicaciones, el paciente puede estar de uno a tres días en su casa y continuar allí la convalecencia.
- 3) Menor tasa de infección:
Los tejidos internos del cuerpo no son expuestos al aire ambiental, a diferencia de la cirugía abierta. Además, la video magnificación permite manejarlos en forma más precisa y delicada, protegiendo órganos vitales.
- 4) Reincorporación a la actividad normal:
Al ser todo mucho más benigno, se vuelve a la vida diaria más rápidamente.

5) Mejor resultado estético:

Las cicatrices son más pequeñas y menos visibles. (Rovelo, s.f.)

A pesar de las múltiples ventajas que presenta la cirugía de mínima invasión, existen ciertos inconvenientes que se presentan los cuales se enuncian de la forma siguiente:

1. Solo una visión bidimensional del campo operatorio.
2. Mangos de mayor longitud y no ergonómicos, los cuales se controlan desde el exterior del cuerpo del paciente, por lo que el cirujano puede usarlos de forma menos natural que los instrumentos quirúrgicos tradicionales.
3. Procedimiento nuevo, lo que conlleva a que los errores sean más probables que con las técnicas tradicionales, por causa de la propia complejidad de este tipo de cirugía.
4. A menudo requiere más tiempo que la intervención quirúrgica tradicional.
5. Falta de experiencia en lo correspondiente a la inserción de los trócares y elementos punzo cortantes.

1.5. Planteamiento del problema

Como se mencionó anteriormente, durante el procedimiento quirúrgico de mínima invasión justo después de realizar nuestro neumoperitoneo constante del gas CO₂, se deben ingresar nuestro instrumental quirúrgico laparoscópico con el cual se va a realizar nuestra cirugía. Para lo anterior, se requiere de elementos punzocortantes denominados trocares.

Con los trócares, el cirujano realiza una primera pequeña incisión dependiendo del área que se va a trabajar. Para poder realizar la inserción de los trócares, actualmente durante el desarrollo de la cirugía se apremia a la experiencia de cada uno de los cirujanos justo al momento de realizar la inserción puesto que cada paciente es totalmente diferente. Dada esta diferencia, y la falta de cirujanos expertos en la inserción de los dispositivos trócares, es que se ven envueltos en una serie de riesgos para la operación que van desde una mala punción hasta desgarrar tejido u órganos y provocar lesiones innecesarias en el paciente,

además se corre el riesgo de provocar úlceras o hernias, lo cual implica mayor riesgo para el paciente.

Siendo este uno de los mayores problemas dentro de la cirugía laparoscópica, es decir la falta de medidas e instrumentos de entrenamiento que logren capacitar a los cirujanos novatos de acuerdo con los estándares puestos por parte de cirujanos expertos al momento de realizar la función de punción, ya que no se tienen registros puntuales de la cantidad de fuerza necesaria, así como la calidad de la punción generada al momento de la inserción del instrumento.

Por lo anterior, el objetivo principal del presente trabajo será realizar un simulador que permita a los futuros cirujanos obtener herramientas dónde puedan desarrollar sus habilidades psicomotrices, teniendo una referencia en la calidad de su punción con el instrumento trocar, que facilite su entrenamiento y al mismo tiempo logre ser un medio de evaluación de los aprendices por parte de cirujanos expertos.

1.1. Estructura de la tesis

En el presente trabajo, se dará la siguiente estructura con el objetivo de tener una guía de apoyo para la obtención de datos significativos correspondientes a cada capítulo.

Capítulo 1. En el este primer capítulo, se realiza una exploración histórica concerniente al desarrollo de la cirugía de mínima invasión, partiendo desde su antecedente tradicional de la cirugía convencional hasta llegar a la cirugía laparoscópica con algunos antecedentes, así como la realización de la primera colecistectomía laparoscópica en el país. Además, se abordan las ventajas y desventajas del procedimiento y se analiza el problema a resolver mediante el desarrollo de la presente tesis. Por otra parte, se establece la justificación de dicha problemática y los objetivos a realizar tanto de forma general como específica.

Capítulo 2. En el segundo capítulo, se aborda la necesidad de la creación de entrenadores y dispositivos para cirugía laparoscópica explorando las causas que originaron la creación de estos dispositivos, así como la técnica correcta de inserción del trocar, los elementos necesarios al momento de realizar este procedimiento, sus riesgos, vías de acceso y consideraciones generales.

Dichos dispositivos se enfocan en la correcta inserción del trocar, así como adquirir diferentes valores físicos como lo es la fuerza, presión, ángulos de entrada, impedancia, aceleración entre otras medidas, a partir de las cuales

podemos interpretar analíticamente los resultados de sus experimentos, y cómo estos influyen en mejores prácticas médicas.

Por otra parte, se muestran algunos dispositivos virtuales que como su nombre lo indica, la principal característica es la carencia de elementos físicos y de interacción; sin embargo, estos últimos realizan simulaciones y generan modelos virtuales que nos permiten explorar con precisión lo que podría suceder en un entorno real.

En cada uno de los dispositivos se presenta la metodología, el desarrollo, así como pruebas de validación y los resultados obtenidos, mediante gráficas y tablas que resumen los resultados de cada trabajo.

Capítulo 3. El tercer capítulo representa la metodología utilizada para la creación de este entrenador. Partiendo de los sensores, sus alcances y limitantes, así como su acondicionamiento de señal y calibración para poder obtener una medida de fuerza y presión adecuadas, a través de sus respectivos modelos matemáticos. En esta sección, se observan las gráficas obtenidas al realizar el proceso de calibración con su respectiva ecuación final que nos servirá para ser utilizada posteriormente en el código de la plataforma una vez hechas las lecturas por parte de la tarjeta de adquisición de datos.

Además, se aborda la necesidad de un convertidor analógico – digital externo y las causas que animaron al uso de este, así como la comparativa entre el convertidor propio del microcontrolador y el externo.

Una vez terminada la calibración, se explora el modelo de insuflación, su calibración y respectiva adaptación, así como el código utilizado para realizar el control de la bomba, modelo matemático y gráfica obtenida.

Por otra parte, se puede observar la electrónica empleada, su simulación y la creación de la placa PCB, así como cada uno de los elementos que la constituyen, la forma en que fueron creados y su respectivo plano electrónico.

En cuanto al modelo físico, se establecen los objetivos de diseño a considerar como lo fueron los tipos de inserción del trocar, así como la necesidad de una cavidad semi-hermética que permita el ingreso del trocar, el ingreso del aire y también se permita la obtención de la presión al interior de este, con lo cual se presenta el modelo final desarrollado que cumple con las características mencionadas.

Por último, encontramos la programación de la plataforma de entrenamiento, así como las tecnologías utilizadas y la estructura de esta con diferentes vistas. En primera instancia, se explica brevemente el contenido gráfico de la aplicación, una interfaz gráfica con diferentes módulos. Posteriormente, cada uno de esos módulos se adaptan como “clases” del programa principal el cual utiliza archivos auxiliares que representan la comunicación exterior con el microcontrolador, así como los modelos matemáticos adquiridos previamente de la sección de calibración.

En esta sección, se explica mediante diagramas y tablas la interacción de cada elemento, sus principales métodos y atributos. Por otra parte, encontramos la descripción de la base de datos utilizada para guardar los datos de cada sesión de entrenamiento. Dicha plataforma contiene diferentes secciones que muestran tanto las gráficas finales como los resultados finales, además de la inclusión de la búsqueda de sesiones pasadas y su correspondiente histórico de todas las sesiones de entrenamiento.

Capítulo 4. En este capítulo, se presenta el diseño final de los diferentes dispositivos empleados para el desarrollo del entrenador para trocares, los cuales son elementos de independiente configuración. Es en este capítulo que se presenta el dispositivo final de entrenamiento adaptado con sus propios “pads” de entrenamiento, así como sus características fundamentales.

Además, se muestra la caja electrónica que logra contener en su interior el microcontrolador, la fuente de poder, el sensor de presión, así como la tarjeta PCB para controlar cada uno de los dispositivos periféricos, su alimentación, etapa de potencia y adquisición de datos.

Posteriormente, se presentan ejemplos de las gráficas finales al operar el dispositivo tanto la gráfica de fuerza como la de presión, donde vemos los resultados que obtenemos al configurar una determinada presión o ejercer determinada fuerza durante el entrenamiento.

Se explica brevemente la configuración general desde la plataforma de entrenamiento, así como las instrucciones de uso, para poder operar correctamente la interfaz gráfica con los dispositivos periféricos, además de mostrar el resultado final de dicha interfaz.

Para finalizar el capítulo 4, enlistan todas las pruebas de validación de cada uno de los elementos que componen el sistema de entrenamiento, desde los

dispositivos eléctricos y electrónicos, hasta las pruebas de escritorio de cada una de las vistas de navegación para el usuario.

1.2. Justificación

Actualmente, el sector salud dentro de la cirugía mínimamente invasiva presenta problemas en la formación de futuros cirujanos, debido al constante aumento y la necesidad de cirugías por esta técnica quirúrgica. También, se hace presente la necesidad de profesionales expertos que logren dar una solución a la creciente demanda. Para lograr atender lo anterior, resulta fundamental hacer uso del desarrollo científico y tecnológico que, mediante la investigación se propongan soluciones profesionales que logren ser aplicables a los problemas actuales, de ahí emana el sentido y esencia de la actividad ingenieril.

En este proyecto, dicha actividad tiene la propuesta de crear una solución, mediante el uso y creación de un simulador, que logre formar recursos humanos cada vez más capacitados y mejor entrenados.

¿Cómo lograremos este objetivo?, utilizando diferentes técnicas para la parametrización de la presión y fuerza que emplea un cirujano experto en la cirugía laparoscópica, y que esta medida sea un punto de referencia para el entrenamiento y capacitación de nuevos cirujanos que logren aplicar sus conocimientos de una forma más rápida para su pronta inserción dentro de su vida profesional quirúrgica, y su contribución en la constante lucha por atender una población que necesita del apoyo del capital humano.

1.3. Objetivos

1.3.1. Objetivo general

El objetivo de esta tesis de licenciatura es el desarrollo de un sistema que permita el entrenamiento de los cirujanos aprendices en la colocación de elementos punzocortantes como los dispositivos trocares para mejorar la seguridad y las habilidades psicomotrices en la cirugía laparoscópica.

1.3.2. Objetivos específicos

1. Desarrollar un sistema de bombeo que simule la insuflación de gas CO₂, en un ambiente de entrenamiento funcional controlado.
2. Creación de un entorno de cavidad abdominal simulada.
3. Diseño mecánico de la interfaz de entrenamiento.
4. Diseño electrónico de los elementos, diagrama y simulación de nuestro sistema de recepción de señal.
5. Acondicionamiento de la señal, modelado matemático y calibración de los sensores.
6. Programación de microcontrolador, así como un entorno gráfico que pueda registrar los datos proporcionados por nuestros sensores.
7. Obtención de modelo matemático que discrimine la calidad de la punción.
8. Desarrollo de base de datos para una posterior consulta de los cirujanos.

Capítulo 2 Estado del arte

2. Simuladores en cirugía laparoscópica

El mundo de la cirugía mínimamente invasiva (MIS, por sus siglas en inglés), tiene entre sus grandes beneficios la reducción del tiempo de recuperación del paciente, procedimientos con una menor cantidad de incisiones debidos al avance científico, así como la evolución en el instrumental quirúrgico, pero también gracias a la disponibilidad y talento que los cirujanos quienes han mostrado dedicar tiempo y esfuerzo hacia la adopción de nuevas técnicas y por supuesto nuevas herramientas (Harmon, 1959).

La adopción de nuevo equipo, herramientas y técnicas, también trae consigo un proceso de aprendizaje de estos; apoyados por el interés de la comunidad médica por incluir dentro de sus programas de entrenamiento el uso de simuladores con el objetivo de evitar el entrenamiento clásico (Handelman et al., 2018), fundamentado en lo descrito coloquialmente como “Lo veo, lo hago y lo enseño”, que implica un modelo basado en la demostración, lectura, observación, pero con la carencia de práctica y riesgos al exportar lo aprendido en un entorno real (Javaux et al., 2018).

La importancia de la práctica segura cobra sentido cuando en casos anteriores han demostrado lesiones de tejidos, órganos y vasos sanguíneos durante una cirugía mínimamente invasiva (Handelman et al., 2018). De acuerdo con Vilos (Vilos et al., 2008), cerca del 50% de las lesiones del tracto gastro intestinal y vasos mayores toman presencia durante la entrada primaria y más del 80% de las lesiones ocurren durante la colocación del trocar primario. Asimismo, se señala en su estudio que el 83% de las lesiones vasculares, 75% en intestinos y 50% de las hemorragias locales han sido causadas por la inserción del primer trocar a la cavidad abdominal.

2.1. Introducción de la aguja de Veress

Como primer paso, se tiene que verificar el correcto funcionamiento de la aguja de Veress (Figura 2), es decir, su mecanismo de seguridad, la luz de la aguja y que el orificio de salida de gas no se encuentre parcial o totalmente oculto. Además, se sugiere que la aguja tenga un buen afilado para lograr una mayor facilidad en la introducción de esta y disminuir así la fuerza necesaria para su inserción.

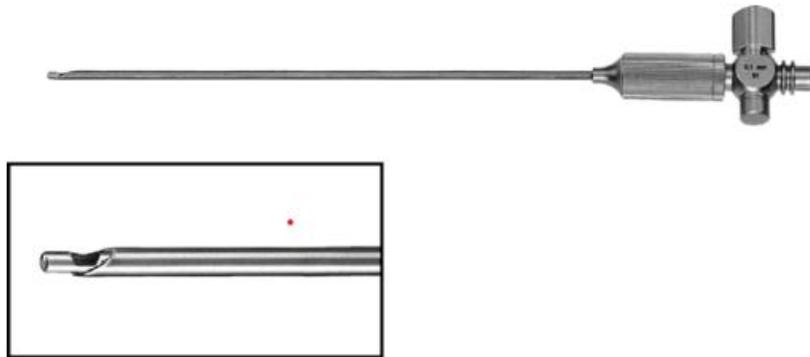


Figura 2 Aguja de Veress. Imagen extraída de [43] (Saxena & Höllwarth, 2008)

Se recomienda utilizar la vía umbilical, ya que tiene la ventaja de tener una distancia menor de 5 mm entre la piel y el peritoneo, colocando la aguja a 45°. También, existe la entrada localizada en el hipocondrio izquierdo como alternativa cuando no se ha logrado conseguir una entrada umbilical o en caso de adherencias. Utilizando el hipocondrio izquierdo se realiza la punción a mitad de camino entre una línea teórica trazada del ombligo a la parrilla costal izquierda situada a 3 traveses de dedo por encima y 3 traveses de dedo a la izquierda del ombligo; sin embargo, dicha vía de acceso debe evitarse en pacientes con escoliosis dorsales.

La vía subcostal es otra alternativa ya que teóricamente es la región abdominal con menos posibilidades de adherencias, pero (así como la vía anterior) existe la posibilidad de lesión intestinal (preferentemente del estómago) si éste se encuentra muy distendido. En esta vía se tiene que introducir la aguja en el hipocondrio a 1-2 cm del reborde costal en una línea teórica que va 1 cm medial a la cresta ilíaca anterosuperior en dirección ascendente hacia la mama.

Una vez seleccionada la vía de acceso y hecha la introducción de la aguja, se deben evitar los movimientos de la aguja libre dentro de la cavidad peritoneal. Para verificar la correcta posición de la aguja, se realiza una aspiración con una jeringa de 20 cc, en caso de tener una localización correcta la aspiración demostrará el vacío intraperitoneal. Después de realizar esta verificación, se procede a inyectar el suero fisiológico cuya recuperación será imposible si la aguja se encuentra correctamente situada. La función de este suero es permitir separar

el epiplón próximo a la punta de la aguja disminuyendo así la posibilidad de un enfisema a ese nivel.

Una vez hecho lo anterior, continuamos con la insuflación dónde la presión intraperitoneal al inicio debe ser próxima a 0 y en todo caso inferior a 7 – 8 mmHG, en caso de presiones superiores se debe revisar la posición de la aguja. Si la presión sube rápidamente a 15 mmHG se interrumpe el flujo ya que existe la posibilidad de que la aguja se encuentre en el espacio preperitoneal. Típicamente, cuando la localización de la aguja se mantiene de forma correcta se producen descensos en la presión intraperitoneal.

2.2. Trocar como instrumental de acceso

El dispositivo trocar es un instrumento cuyo diseño permite proporcionar acceso a la cavidad abdominal siendo el canal de trabajo a través del cual, se introduce la óptica y los distintos instrumentos. El trocar está constituido por un sistema de válvulas y un canal de ingreso de CO₂, el primer sistema tiene como finalidad evitar la pérdida del neumoperitoneo, mientras que el canal de ingreso procura mantener el flujo de CO₂.

Podemos encontrar trócares de 2,3,5,10,12,15,18 y 20 mm de diámetro interno, mientras que su longitud varía de acuerdo con el paciente a intervenir, requiriendo mayor longitud aquellos que son utilizados en cirugía bariátrica que los utilizados en cirugía pediátrica. Dichos instrumentos están compuestos de una cánula externa o camisa y un punzón de punta cónica, roma o piramidal.

Podemos encontrar trócares metálicos reutilizables y desechables, pero también existen trocares descartables que están provistos de un protector de plástico de seguridad que se activa al momento de atravesar el peritoneo.

Una vez seleccionado el trocar a utilizar, la técnica empleada para su inserción consiste en colocar el extremo superior (del trocar) en la palma de la mano, y apoyar los dedos índice y medio a una distancia variable tal como podemos apreciar en la Figura 3.

Dicha posición de dedos servirá como tope para evitar que el trocar entre de manera excesiva. Para poder ingresarlo, se aplica una fuerza controlada y se gira la mano que sostiene al trocar. Se debe tomar en consideración que la incisión para el acceso del trocar debe tener un diámetro exacto para poder evitar el desplazamiento del trocar y la respectiva salida del gas.



Figura 3 Técnica correcta de inserción del trocar. Imagen extraída de [49] (Sociedad Argentina de Cirugía Digestiva, 2022)

Dado lo anterior, el aprendizaje, el desarrollo de habilidades y la práctica fuera de la sala de operaciones es de vital importancia para los aprendices. Los principales retos con los cuales se han enfrentado los cirujanos al realizar una cirugía laparoscópica han sido identificados de la siguiente manera: Coordinación ojo-mano, ausencia de la percepción de profundidad, pobre realimentación háptica, dificultad en el manejo y maniobrabilidad de los instrumentos, así como el efecto de fulcro. Dichas carencias requieren preparación y práctica constante (Horeman et al., 2015).

Hoy en día, tenemos múltiples opciones de entrenamiento para las habilidades básicas descritas por las FLS (*Fundamentals of Laparoscopic Surgery*, por sus siglas en inglés) [15] (Handelman et al., 2018), las cuales sugieren enfocar los entrenamientos en 5 módulos: 1) transferencia de objetos, 2) corte de patrones, 3) ligamiento de estructuras, 4) sutura con procedimientos de anudado extracorpóreo y 5) sutura con anudado intracorpóreo.

Las opciones para el entrenamiento más populares son, por un lado, los modelos físicos que representan las cajas de entrenamiento o *training box*, las cuales pueden ser constituidas desde formas y modelos similares al sujeto anatómico hasta cajas de plástico perforadas, utilizando desde poco a mucho presupuesto focalizadas en diferentes habilidades y recursos. En contraste, encontramos aquellos simuladores hechos mediante software con realidad virtual con los cuales el entorno y entrenamiento visual adquiere mayor optimización, además de contar con tareas que pueden ser reusables sin riesgo de desgaste en contraparte con los simuladores físicos.

2.3. Simuladores Físicos

2.3.1. Entrenador Laparoscópico con Neumoperitoneo

El neumoperitoneo constituye una etapa sumamente importante durante la cirugía laparoscópica puesto que implica la separación de órganos a partir de la introducción de CO₂, o algún gas inerte manteniendo una cierta presión y flujo constante de tal manera que se evite el daño de órganos internos. Dicha presión, así como el flujo requerido dependen del tipo de operación, edad y peso de nuestro paciente. Sin embargo, no se puede tener una medida exacta de una presión base estándar para todos los casos.

Para tener un acercamiento, se han creado parámetros para establecer el correcto neumoperitoneo, como lo podremos observar en la siguiente tabla:

Edad	Presión	Flujo
Recién nacido	5 mmHg	0.5 lt
Infante <	5 – 7 mmHg	1.0 lt
Infante >	7 – 19 mmHg	2.0 lt
Niños en edad preescolar	10 – 12 mmHg	3.0 lt
Niños en edad escolar	10 – 12 mmHg	4.0 lt
Adolescentes	10 – 13 mmHg	5.0 lt
Adultos	<15 mmHg	

Tabla 1 Parámetros para el establecimiento del neumoperitoneo. Extraído de [23] (L. Ortiz Simón et al., 2012)

Esencialmente, el presente entrenador fue construido con un microcontrolador cuyo objetivo fue recibir la información proveniente del sensor de presión para controlar el flujo de aire presente dentro de la cavidad a insuflar. Además, se añade la inclusión de válvulas calibradas en mmHg para el control fino del aire. El entrenador consiste en una cámara de un balón creado de neopreno de baja densidad insertado dentro de un maniquí, con trocares laparoscópicos de 5 mm, una aguja de veress, un compresor de aire, el cual es utilizado como fuente de gas, un laparoscopio en 0 grados para la visualización interna y una televisión comercial como monitor externo.

Una vez hecho esto, se analizaron diferentes casos clínicos que derivaron en el uso de diferentes presiones; sin embargo, dados los parámetros sugeridos en este caso la presión seleccionada fue de 11 mmHg a 4L/min.

Para realizar el neumoperitoneo, se utilizó una aguja de veress, con la cual se logró acceder a la cavidad peritoneal. En seguida la presión fue elevada a 7 mmHg, como presión inicial con el objetivo de verificar la inexistencia de fugas de aire. Una vez corroborado lo anterior, se procedió a elevar la presión hasta los 11 mmHg recomendados y fueron colocados 2 trócares de 5mm para tener una configuración final de 3 entradas.

La tarea por ejecutar fue el corte en la cavidad abdominal simulada, posteriormente se procedió con la ejecución de tareas de atado, transferencia y sutura realizadas bajo la presión constante de 11mmHg.

El desarrollo de un entrenador físico con neumoperitoneo busca la integración de los cuidados necesarios que el cirujano debe tener al momento de la preparación de la operación, debido a que, en muchos entrenadores el neumoperitoneo se deja en segundo término, enfocándose principalmente en el entrenamiento sobre las tareas de aprendizaje para desarrollar habilidades.

Con este entrenador (Figura 4) reducimos el riesgo de presentar lesiones en la sala de operaciones debidas a un mal manejo del nuestro equipo quirúrgico punzocortante. Sin embargo, el presente entrenador no es capaz de adquirir la fuerza necesaria durante la inserción del trocar.



Figura 4 Entrenador laparoscópico con neumoperitoneo de [23] (L. Ortiz Simón et al., 2012)

2.3.2. Características de fuerza en la inserción abdominal del trocar en cirugía laparoscópica

En el presente trabajo, se realiza la construcción de una plataforma cuya finalidad es lograr soportar la fuerza vertical ejercida al momento de la inserción del trocar, utilizando 2 cojinetes angulares, Figura 5. Cuenta además con 2 motores que controlan el movimiento vertical (Motor 1) y el movimiento rotatorio (Motor 2), ofreciendo así una plataforma experimental de 3 grados de libertad tal como podemos apreciar en la Figura 6.

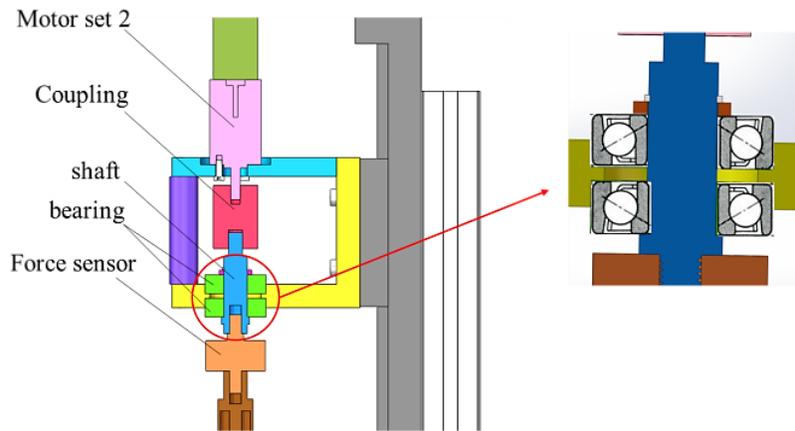


Figura 5 Diseño mecánico. Imagen extraída de [52] (Sun & Tadano, 2017)

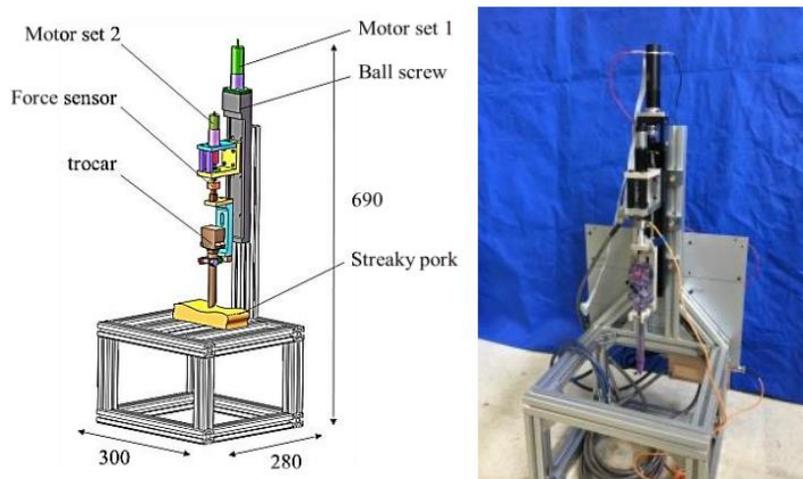


Figura 6 Diseño de plataforma. Imagen extraída de [52] (Sun & Tadano, 2017)

El movimiento vertical es manejado por el motor principal (Motor 1, Figura 6) el cual tiene una salida de 217 rpm, con un momento de 0.53 Nm junto a un husillo de bola con el objetivo de emplear ambos movimientos y conducir el mecanismo en descenso y rotación. Por su parte el motor 2 (Figura 6) es conducido a una velocidad de 39 rpm y un momento de 0.22 Nm. La adquisición de la fuerza se realiza a través de un sensor uniaxial colocado en la parte superior del trocar cuya capacidad máxima es de ± 10 Kg.

Utilizando un controlador PID de posición, se estableció una señal de orden lineal, para obtener una velocidad constante de 4mm/s, y una cámara de alta definición para poder grabar el proceso completo. De esta manera se logra separar el video en diferentes fotografías por cada segundo.

Una vez hecho el diseño mecánico de la plataforma (Figura 5) así como el ensamble de la plataforma, se realizaron experimentos de inserción del trocar utilizando carne de cerdo de 45 mm de espesor dado que la dureza y espesor que presenta es similar al de los seres humanos. El trocar utilizado para realizar los experimentos es de 12 mm de diámetro, es decir el más común utilizado en cirugía laparoscópica.

Terminado el experimento, se obtuvieron diferentes conclusiones, como lo es, entender los diferentes picos de fuerza dependiendo de la capa abdominal penetrada, siendo el peritoneo el segundo pico mostrado en la gráfica (después del pico máximo de fuerza), dado que es la última capa abdominal.

Además, se registró que la fuerza máxima durante todo el proceso de inserción es de alrededor de 42 N tal como podemos apreciar en la Figura 7 Resultados del experimento. Imagen extraída de [52] (Sun & Tadano, 2017), siendo este el registro máximo en el segundo 18 del proceso, cuando el trocar aún no ha penetrado totalmente el abdomen.

Es gracias a la combinación de ambos movimientos, con lo que podemos disminuir las heridas causadas al insertar un trocar, además, se puede observar que, con una velocidad de inserción pequeña (2 mm/s) y una rotación en la plataforma se pueden disminuir los riesgos de daño al tejido, contrariamente de lo que sucede al utilizar altas velocidades de inserción (8 mm/s) o al utilizar velocidades de inserción pequeñas y no utilizar el movimiento rotatorio.

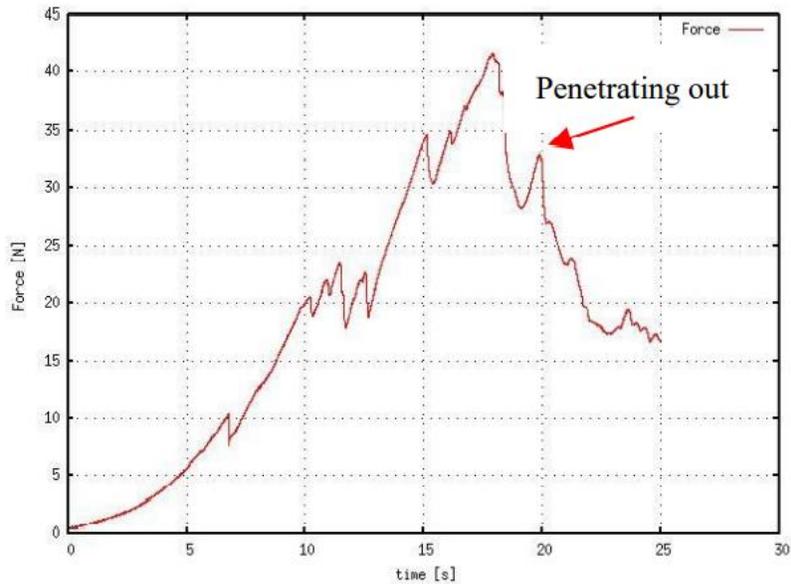


Figura 7 Resultados del experimento. Imagen extraída de [52] (Sun & Tadano, 2017)

2.3.3. El trocar inteligente: Trocar de detección de fuerza, desviación e impedancia para mejorar la cirugía Laparoscópica

El trocar (Figura 8) está compuesto de 2 partes esenciales: la primera un afilado obturador que se encuentra albergado en la segunda parte, es decir, la envoltura de la cánula. Dicho obturador cumple con la labor de permitir la inserción del laparoscopio, así como otros instrumentos quirúrgicos.

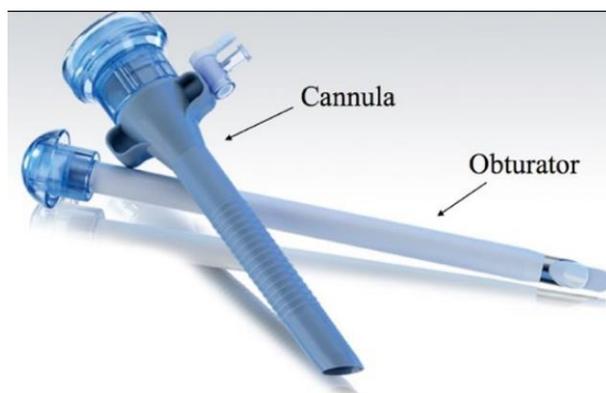


Figura 8 Trocar. Imagen extraída de [44] (Schrope et al., 2019)

El presente dispositivo fue diseñado con el propósito de proporcionar realimentación en tiempo real de la fuerza axial aplicada, la desviación angular y la posición dentro de la pared abdominal a través de la medida de la impedancia eléctrica.

Para lograr la obtención de las variables mencionadas previamente se diseñó un mango ergonómico utilizando 1.75 mm de filamento PLA con un obturador hecho de policarbonato. La punta del instrumento consta de 2 electrodos concéntricos hechos de acero inoxidable y separados por una capa de polioximetileno, colocado en la parte inferior del eje. El mecanismo para manejar el obturador fue conectado a la cánula a través del mecanismo de ajuste por fricción mostrado en la Figura 9.

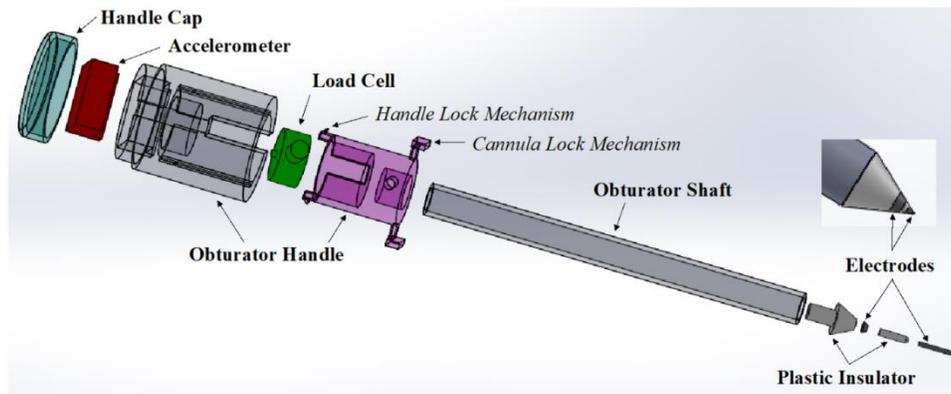


Figura 9 Diseño del trocar inteligente. Imagen extraída de [44] (Schrope et al., 2019)

La adquisición de la fuerza se realiza a través de un amplificador de la celda de carga (HX711), quien recibe la señal proveniente de la celda de carga TAS606, para posteriormente ser interpretada por un Arduino Zero a través de su comunicación SPI. Por otra parte, la desviación angular fue leída a través de un acelerómetro (ADXL345) colocado en el mango del obturador, quien envía al Arduino Zero datos en crudo correspondientes a la posición tridimensional utilizando el protocolo de comunicación I2C. Los valores de impedancia fueron calibrados utilizando un rango de resistencias entre 100 y 10 k, necesarios para arrojar valores fisiológicamente relevantes.

Después de adquirir los datos provenientes de los sensores, se utilizó una Raspberry pi 3 para recibir la información enviada por el Arduino Zero a través de su puerto de comunicación serial el cual logra un registro de 9.86 muestras por segundo. Posteriormente la información se muestra por medio de una GUI diseñada para visualizar la realimentación continua en una pantalla de 7 pulgadas (LANDZO), una vez hecho el post procesamiento del desempeño.

Para poder evaluar el dispositivo, se utilizó una pared abdominal de un porcino, dónde 19 médicos estudiantes probaron la efectividad del trocar para posteriormente evaluarlo y comparar los resultados con un trocar estándar. Dichas evaluaciones mostraron que 95% de los estudiantes reportaron al dispositivo como una herramienta efectiva de entrenamiento, la cual puede ser implementada posteriormente en entrenamientos médicos, gracias a que este dispositivo ofrece la ventaja única de informar a los cirujanos sobre la localización de la punta del trocar dentro del tejido y así asegurar la correcta colocación del trocar.

Como podemos observar, en el presente trabajo se obtuvo una fuerza máxima de más de 20 Lbs (Aproximadamente 88.9644 N, Figura 10). Cabe resaltar que la fuerza empleada se incrementó debido al desvío del trocar al momento de su colocación, lo cual es un indicativo de una colocación incorrecta que podría derivar en el riesgo de complicaciones para el paciente.

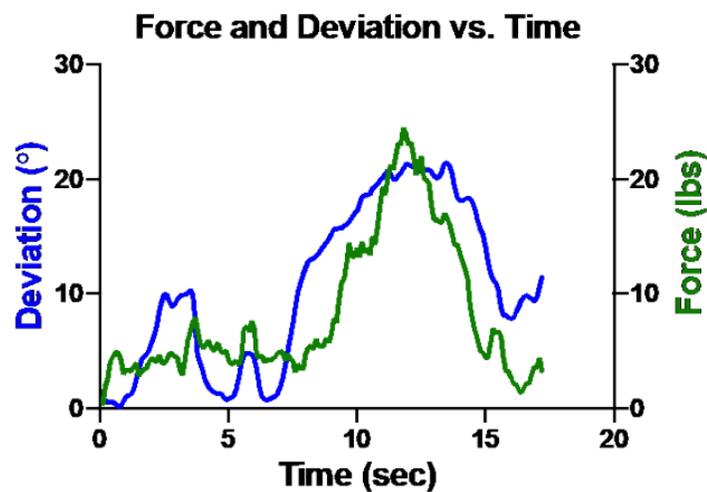


Figura 10 Desviación del trocar. Imagen extraída de [44] (Schrope et al., 2019)

2.3.4. Adquisición interactiva 3D del parámetro de Fuerza/Torque e identificación de la correlación durante la inserción del trocar primario en cirugía laparoscópica abdominal: 5 Casos

El propósito del presente estudio es recolectar y analizar las características de la fuerza y el torque, así como la conducción del trocar primario por parte del cirujano en el proceso de inserción. Además, el análisis de las líneas de respuesta de fuerza y torque que reflejan el carácter de cada capa abdominal dónde están incluidos: epidermis, dermis, tejido subcutáneo, línea alba fascia, grasa preperitoneal, y peritoneo parietal.

Para poder adquirir las características de fuerza y momento, se necesitó de un sensor ATi Nano25 el cual registró los 3 ejes de fuerza, así como los mismos correspondientes a su momento. El sensor puede medir un máximo de 500 N con una resolución de 1/16 N y un momento 3 Nm con una resolución de 1/2640 Nm. Dicho sensor fue ensamblado dentro de un soporte personalizado impreso en 3D, hecho de ácido polilactico (PLA), con lo cual se demuestra una alta biocompatibilidad (Figura 11).

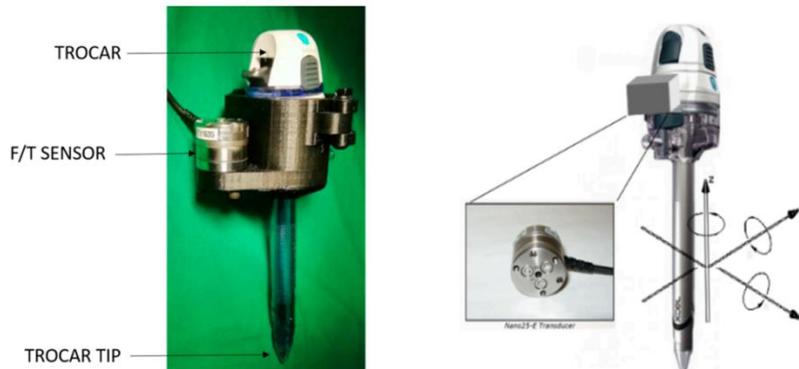


Figura 11 Adaptación del trocar primario. Imagen extraída de [32] (Nillahoot et al., 2022)

Una vez creado el dispositivo (Figura 11), el cirujano principal comenzó la sesión de pruebas insertando el trocar en el abdomen de un cadáver, siendo la línea alba el punto de inserción justo debajo del ombligo. Durante la inserción todos los registros de datos fueron importados a MATLAB 2019b para su análisis y visualización.

Al analizar los registros, la máxima cantidad de fuerza utilizada fue de 61.86 N, 16.83 N, 29.22 N, 19.16 N y 28.60 N, provenientes de 5 hernias abdominales de tejidos humanos blandos, por su parte las cantidades máximas de momento

medidas por cada caso fueron las siguientes: 1.76 Nm, 0.552 Nm, 0.721 Nm, 0.629 Nm, 1.11 Nm.

La gran diferencia entre la variación de fuerza registrada ocurre debido a la técnica del cirujano, así como la variación de la estructura anatómica (Figura 12). Por otra parte, al analizar los datos obtenidos de cada uno de los casos podemos observar que la cantidad de fuerza requerida para penetrar tejido elástico debe ser mayor en la capa de la dermis que en la capa subcutánea y de grasa peritoneal, además el cirujano disminuye la cantidad de fuerza cuando se acerca al peritoneo como una forma de precaución, lo que dibuja un patrón común de análisis el cual es la rotación (alrededor del eje z) y la fuerza de empuje alrededor del mismo. Dicha técnica resulta ser la más efectiva ya que reduce la cantidad de fuerza vertical.

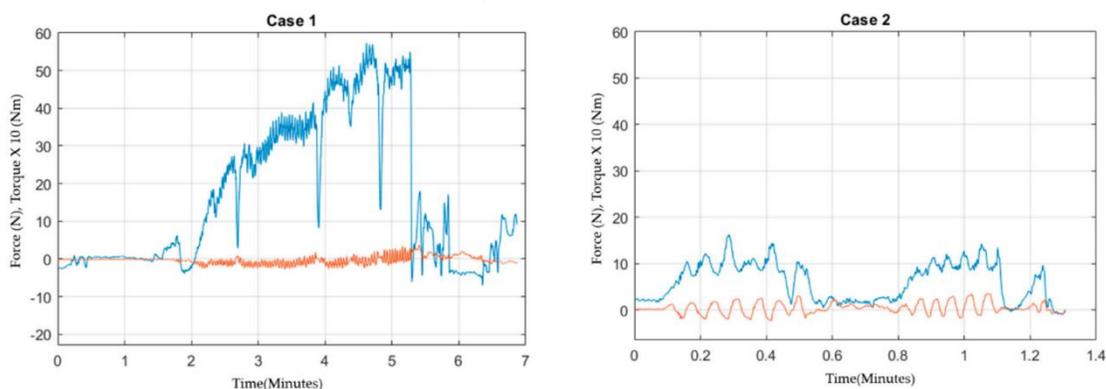


Figura 12 Comparativa entre 2 casos. Imagen extraída de [32] (Nillahoot et al., 2022)

2.3.5. Un nuevo dispositivo para la inserción segura del trocar en cirugía laparoscópica basado en las características de fuerza

El presente dispositivo tiene la forma de una pistola de (282 mm * 122 mm * 237 mm), el cual permite a los operadores el fácil manejo de este al colocar el trocar a lo largo del eje central. Contiene 2 combinaciones de motor: el primer motor se encuentra ensamblado al interior del mango para poder manejar la inserción del trocar al avance frontal a través del husillo de bola, una banda y una banda dentada. En cuanto al segundo motor, este se encuentra ensamblado en la parte trasera del trocar, el cuál manejará la rotación a través del eje central.

Con la finalidad de levantar y sostener la pared abdominal al momento de la inserción, se añadieron 6 cilindros neumáticos alrededor del trocar (Figura 13) los cuales están divididos 4 grupos, y cada cilindro contiene una ventosa conectada a un generador de vacío, además cuentan con 4 codificadores (Encoder), para poder medir la deformación causada por inserción en el momento del

desplazamiento de estos. Debido a la irregularidad de la pared abdominal, se añadieron juntas de bola entre las ventosas y los cilindros para asegurar su fijación.

El generador de vacío produce -80 kPa de manera constante, lo que resulta en una fuerza adhesiva de 62.17 N. El motor 1 es controlado a través de un controlador PID de tal forma que el trocar logre ser insertado a una velocidad constante entre 0 a 12 mm/s. Por otra parte, el motor 2, es el encargado de la rotación, dicho motor cuenta con un controlador PID posicional, el cual produce una rotación de amplitud $\pi/4$ y un periodo T de 1 Hz.

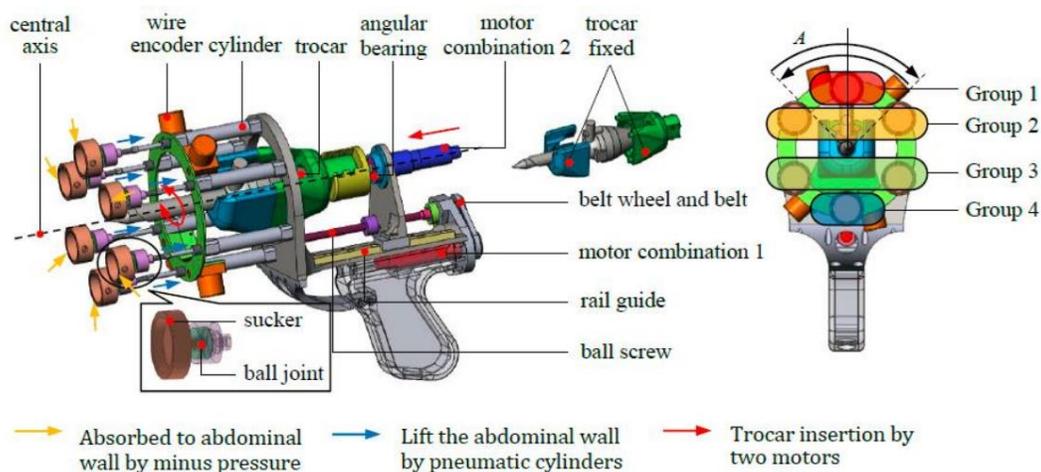


Figura 13 Configuración del dispositivo. Imagen extraída de [53] (Sun & Tadano, 2019)

Con la finalidad de comprobar el correcto funcionamiento del dispositivo, se utilizó la pared abdominal de un porcino debido a la estructura multicapas. Es necesario entender que el tamaño de la ruptura puede cambiar dependiendo de diferentes factores como el tipo de trocar, la velocidad de inserción, las diferencias individuales de cada paciente, etc.

Durante los experimentos, el trocar penetró hasta fuera del abdomen, siendo los cilindros neumáticos los encargados de restringir la deformación exitosamente, además se calculó el desplazamiento de cada grupo de ventosas, lo que permitió también generar el algoritmo de paro en automático una vez hecha la penetración.

Finalmente, lo anterior derivó en determinar la efectividad del dispositivo al lograr insertar de forma rápida y segura el trocar, tal como podemos apreciar en la Figura 14 que demuestra claramente la comparación en tiempo y exposición de la punta

del trocar, tanto de forma manual como por el dispositivo mostrando mejoras significativas.

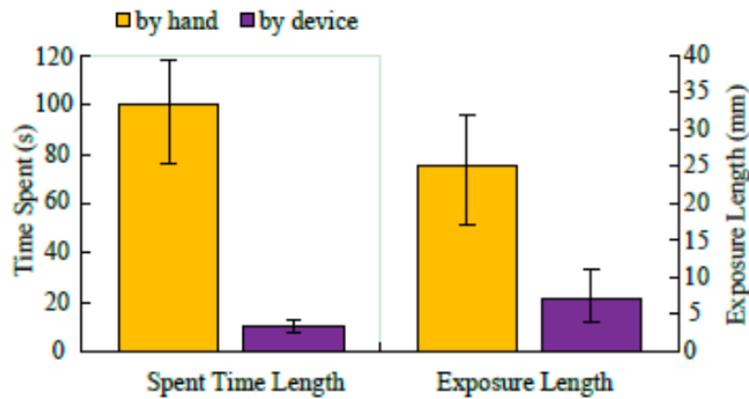


Figura 14 Evaluación de la inserción del trocar de forma manual y automática. Imagen extraída de [53] (Sun & Tadano, 2019)

Por último, logramos apreciar que la fuerza utilizada al momento de la inserción del trocar no va más allá de los 15 N, además logramos visualizar un cierto patrón al realizar la inserción en la última capa ya que la fuerza necesaria disminuye (Figura 15).

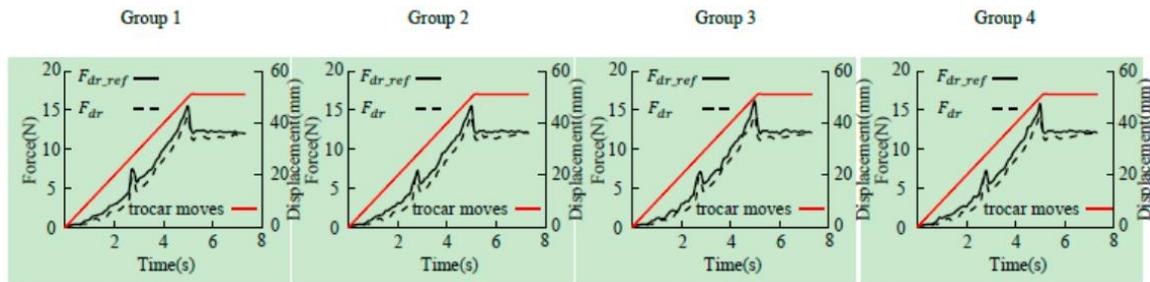


Figura 15 Comparación de fuerza en cada grupo. Imagen extraída de [53] (Sun & Tadano, 2019)

2.4. Simuladores Virtuales

2.4.1. Simulación del neumoperitoneo en un paciente específico para la planeación de la cirugía laparoscópica

El simulador virtual es una metodología que permite simular el neumoperitoneo a partir de parámetros biomecánicos reales. Para poder generar un modelo, se partió de 3 pasos principales: la segmentación, generación de mallas y por último la parametrización mecánica.

Durante la segmentación, se hace uso de una imagen preoperativa 3D adquirida a través de una tomografía computarizada de 2 cerdos. Con dichas imágenes, se identifican 3 zonas principales: 1) la pared abdo-torácica (azul), 2) las vísceras abdominales (rojo) así como 3) las vísceras torácicas (cian), las cuales podemos apreciar en la Figura 16.

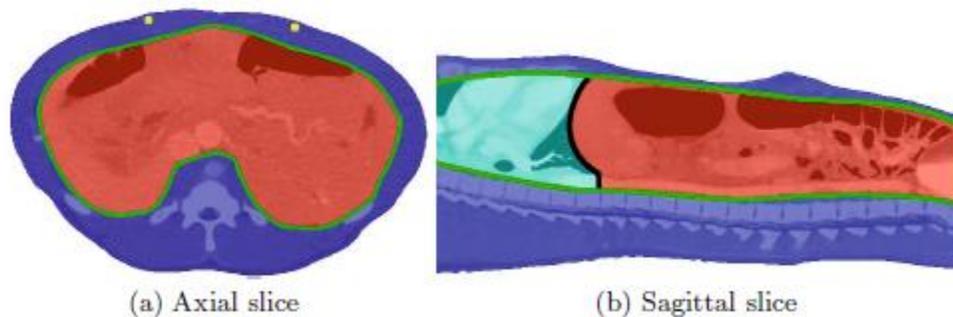


Figura 16 Imagen preoperativa extraída de [5] (Bano et al., 2012)

En la generación de las mallas de volumen y superficie, se utilizó una librería especial, llamada Librería computacional de Geometría y Algoritmos (*Computational Geometry Algorithms Library*), para el análisis posterior de elemento finito que tendrá uso durante la deformación de tejido, así como para la detección de colisiones.

Una vez hecha la aproximación con elemento finito, los parámetros mecánicos asociados a la pared abdominal son el Módulo de Young equivalente a 24kPa, así como el coeficiente de Poisson de 0.49, los cuales por motivos de simplicidad se consideran las vísceras abdominales de manera homogénea y se les asocia un módulo de Young de 15 Kpa. En el caso de las vísceras torácicas (conteniendo al corazón y los pulmones), se establece un módulo de Young de 7kPa.

La simulación es realizada mediante el software SOFA, un programa computacional especializado que provee de un conjunto de métodos especiales para el modelado de tejidos virtuales, así como sus interacciones en un mundo de simulación virtual.

Posteriormente, en la etapa de mapeo y colisión del simulador virtual, las mallas de superficie son usadas para la visualización, detección de colisiones y fuerzas de presión. Dentro de cada malla, se cuenta con un vértice asociado con una malla de superficie, la cual a su vez cuenta con un tetraedro en la malla de

volumen. De esta manera, se puede aplicar el principio de propagación maestro-esclavo, un concepto utilizado para enfatizar en la herencia, es decir, lo que se puede enviar desde una base hacia los elementos asociados. De esta manera, si existe un desplazamiento aplicado sobre un nodo del elemento “maestro” dicho desplazamiento se verá reflejado sobre nuestra malla “esclava”, lo mismo ocurrirá si una fuerza es aplicada sobre un nodo “maestro” (Figura 17).



Figura 17 Diagrama modelo maestro – esclavo

La parte posterior de la piel en el ambiente virtual es situada hacia la dirección correspondiente con la mesa de operaciones. Una vez hecho esto, se aplica fuerza sobre la superficie interna de la pared abdominal utilizando la inyección de gas con una presión de 12 mmHg, (aproximadamente 1.6 KPa). Después del mapeo anterior y bajo las condiciones anteriores, el volumen ocupado por la malla de la pared abdominal se extiende sufriendo también una deformación. Esta simulación toma alrededor de 30 segundos.

Para poder realizar la evaluación de la simulación en esta aplicación, se compararon las paredes abdominales, vísceras, así como posiciones de las arterias con las imágenes adquiridas por las tomografías computarizadas, es decir, para cada estructura simulada, se encuentra una gran nube de puntos extraídos desde una máscara de segmentación binaria de la tomografía computarizada.

De esta forma, se registra la distancia entre cada vértice de cada estructura simulada al punto más cercano de la nube de puntos. Con ello, se obtienen errores promedios, los cuales se pueden observar dentro de la siguiente tabla y figura.

En la Tabla 2, podemos observar los errores adquiridos a partir de los datos del cerdo No.1, mientras que dentro de las presentes imágenes podemos observar la

evaluación del error de manera visual, Figura 18. De acuerdo con el código de color, tenemos el error existente en piel (superficie externa), en superficies internas y las vísceras abdominales. En la Figura 18 (d) podemos observar las arterias de la pared abdominal de color rojo y la predicción de la posición de las arterias, mientras que de color verde tenemos la segmentación de las arterias después del neumoperitoneo.

Piel(mm)	Superficie interna (mm)	Vísceras abdominales (mm)	Rango de colores
0 – 1.3	0 -1.0	0 – 0.6	Azul a turquesa
1.3 – 2.7	1.0 – 2.4	0.6 – 1.5	Turquesa a verde
2.7 – 5.1	2.4 – 4.9	1.5 – 3.8	Verde a amarillo
5.1 – 10.6	4.9 – 16.5	3.8 – 20.1	Amarillo a rojo.
3.2 (\pm 2.2)	3.4 (\pm 3.0)	2.9 (\pm 3.5)	Media (\pm desviación estándar)

Tabla 2 Errores promedio, Tabla extraída de [5] (Bano et al., 2012)

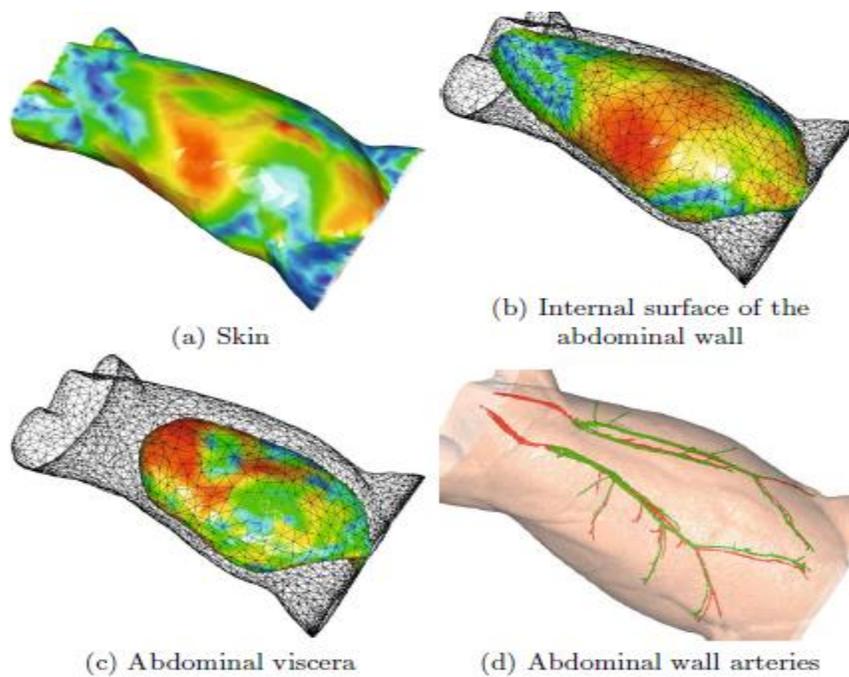


Figura 18 Evaluación visual del error promedio, Imagen extraída de [5] (Bano et al., 2012)

Los resultados adquiridos con este simulador virtual de neumoperitoneo muestran que tienen una aproximación realista de la piel simulada, pared abdominal, así como arterias, reportando una precisión por debajo de 0.5 cm, lo cual puede ser de ayuda al momento de realizar una planeación quirúrgica en procedimientos laparoscópicos. Cabe resaltar que el presente trabajo nos brinda ayuda al momento de realizar un modelo virtual del comportamiento del neumoperitoneo; sin embargo, no nos ofrece información sobre la fuerza necesaria al momento de realizar la inserción del trocar proceso posterior al neumoperitoneo.

Capítulo 3 Metodología

3.1 Criterios generales de diseño

Nuestro sistema de forma general se regirá bajo los siguiente criterios de diseño acorde a los objetivos generales y específicos establecidos en el primer capítulo del presente trabajo de tesis de licenciatura.

1. Dispositivo de fácil transporte además de interacción sencilla y diseño intuitivo.
2. Una cavidad en el entrenador que permita el ingreso de aire para simular la insuflación.
3. Un dispositivo entrenador que permita simular una cavidad del cuerpo humano.
4. Cavidad en el entrenador para la lectura de la presión al interior de este.
5. Mantener el dispositivo trocar con la menor cantidad de aditamentos externos.
6. Generar una placa electrónica que logre integrar todos los dispositivos eléctricos.
7. Dispositivo semi-hermético que permita el ingreso de un flujo constante de aire.
8. Superficie de entrenamiento tipo *pad*, flexible y desechable.
9. Dispositivo auxiliar que permita enviar aire al interior del entrenador.
10. Entorno gráfico que permita visualizar los resultados de la adquisición de señales.
11. Plataforma virtual multiusuario.
12. Sistema gestión de usuarios.

3.2. Sensores

Los sensores, así como los respectivos actuadores, corresponden una etapa fundamental dentro del desarrollo del presente trabajo, debido a que en ellos radica la interpretación del sistema o entorno de desarrollo, y las múltiples variables físicas que son la fuente de información para la debida manipulación del respectivo trocar y así como su insuflador.

3.2.1. Sensor Fuerza

En el presente trabajo, se colocó un sensor de fuerza adaptado en la parte superior del trocar, con la finalidad de poder adquirir la fuerza con la que se inserta un trocar dentro de nuestro simulador.

Las condiciones para la selección de este sensor fueron:

- 1.- Ergonomía y libertad de movimiento.
- 2.- Dispositivo flexible.
- 3.- Posición correcta para poder tomar el dispositivo trocar.
- 4.- Uso en aplicaciones médicas.
- 5.- Criterios de diseño generales.

Bajo los puntos anteriores, se seleccionó el sensor Flexiforce Standar Model A201 (Figura 19), el cual presenta las características físicas y eléctricas descritas en la tabla 3 y 4, respectivamente.



Figura 19 Sensor Flexiforce Imagen Extraída de [54] (Tekscan, s. f.)

Características Físicas	
Tipo	Resistivo
Largo	191 mm
Ancho	14 mm
Espesor	0.203 mm
Área sensible	9.53 mm
Conectores	3 conectores macho (1 No se conecta)
Rango	0 – 4448 N

Tabla 3 Características físicas sensor Flexiforce.

Características Eléctricas	
Tiempo de respuesta	< 5 μ Sec
Temperatura de operatividad	-40 [°C] – 60 [°C]
Durabilidad	> Más de 3 millones de repeticiones.

Tabla 4 Características eléctricas sensor Flexiforce.

3.2.1.1. Colocación del sensor en el trocar

Antes de poder determinar la Zona en la cual se tiene que colocar el sensor, es necesario retomar la técnica correcta para poder manipular el trocar, la cual podemos apreciar en las imágenes siguientes (Figura 20 y Figura 21):



Figura 20 Vista lateral manejo de trocar, imagen extraída de [37] (Punto Quirurgico, 2018).



Figura 21 Visa superior del manejo de trocar, imagen extraída de [37] (Punto Quirurgico, 2018).

Como podemos observar, en la Figura 20 el trocar se coloca sobre la palma de la mano, dejando la válvula de insuflación en la parte anterior del trocar, los dedos anular y medio se colocan a un extremo del vástago del trocar para poder tener una sujeción adecuada y por ende una técnica correcta.

En la Figura 21, observamos una vista de la mano completa al momento de realizar la inserción sobre una superficie de prueba. La zona de contacto principal

con el trocar radica en la palma de la mano, por lo que será esta zona la ideal para poder colocar el sensor trocar y con ello tener la mejor medición posible.

Para este proyecto, podemos apreciar el lugar exacto donde fue colocado el sensor de fuerza Flexiforce dadas las condiciones anteriores (Figura 22).



Figura 22 Colocación del sensor de fuerza en el trocar

3.2.2. Sensor de presión

Como se mencionó en el capítulo 1, la insuflación corresponde a una etapa fundamental para poder realizar el neumoperitoneo al momento de llevar a cabo una cirugía laparoscópica, debido a que la presión ejercida por el gas de operación genera el espacio adecuado para poder insertar nuestro trocar de forma eficiente.

Para realizar la medición de la presión de forma adecuada, se analizó el uso del sensor MP3V5050GP (Figura 23) de NXP Semiconductors, el cual es un sensor piezo-resistivo ampliamente usado en diferentes aplicaciones como lo son: control

de bombeo, robótica, diagnósticos médicos, medición de la presión en la sangre y detectores de nivel. Este sensor fue creado con la finalidad de ser utilizado especialmente por microprocesadores y microcontroladores con entradas ADC, por lo que presenta las siguientes características (Tabla 5).



Figura 23 Sensor de presión. Imagen extraída de [35] (NXP Semiconductors, 2018)

Características Eléctricas	
Rango de presión	0 – 50 KPa
Tipo	Piezo-Resistivo
Voltaje de alimentación	3.3 V como valor máximo recomendado.
Corriente de alimentación	10 mA DC
Tiempo de respuesta	1 mSec
Sensibilidad	54 mV/Opa

Tabla 5 Características eléctricas del sensor de presión

Para poder ocupar de forma efectiva el sensor de presión se realizaron ciertos ajustes comenzado con la placa PCB para poder montar dicho sensor y obtener sus respectivas salidas como se puede apreciar en la Tabla 5 correspondiente a las adecuaciones hechas al sensor (Figura 24).

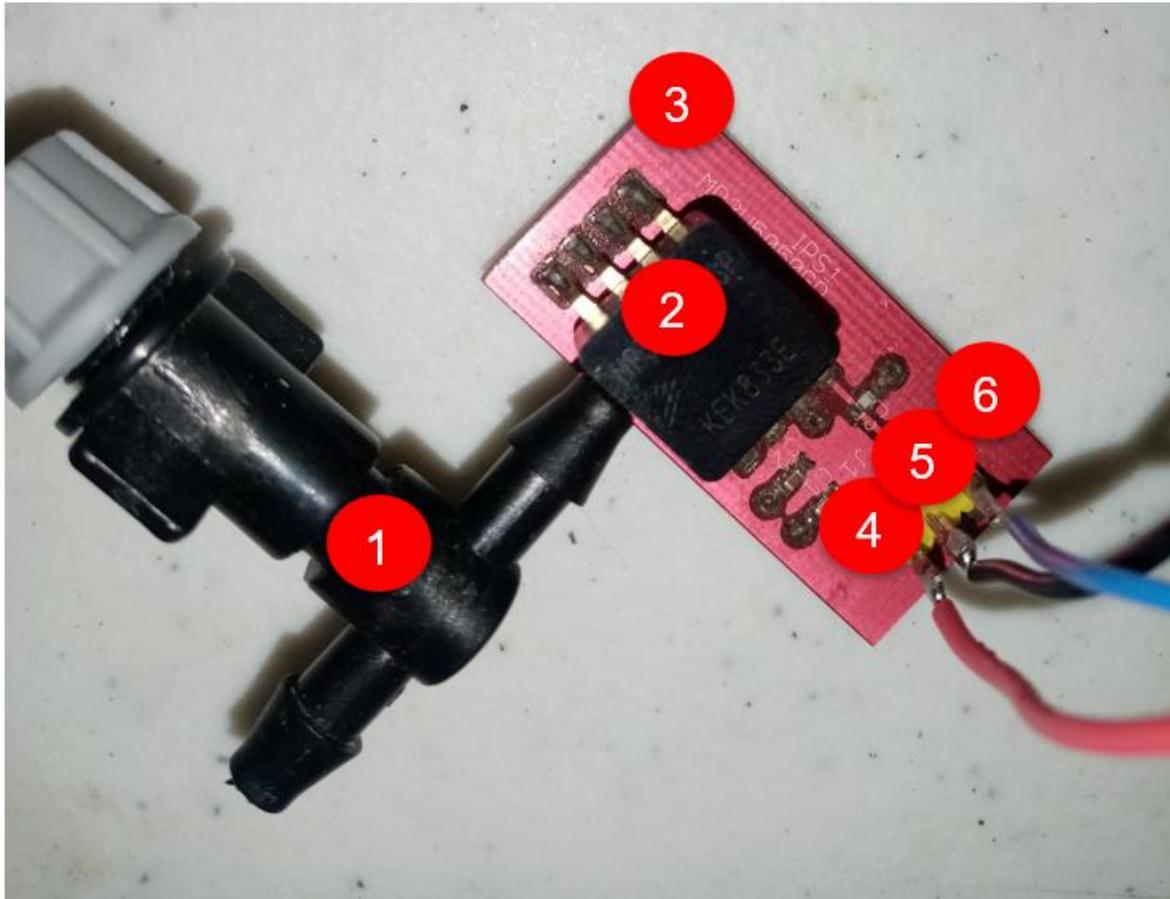


Figura 24 Sensor de presión acoplado 1 Acoplamiento neumático para adquirir la presión al interior del dispositivo de entrenamiento, 2 Sensor MP3V5050, 3 PCB para el montaje del sensor, 4 (Vcc) voltaje de alimentación, 5 GND, 6 (Vout) Señal de salida del sensor.

3.2.3. Acondicionamiento de señal

Previo al uso de cualquier sensor, debemos tomar en cuenta una etapa de vital importancia durante la adquisición de señales. El sensor cumple la función de reaccionar ante un fenómeno físico ya sea fuerza, presión, temperatura, flujo de líquido, entre algunos otros, es el transductor el encargado de interpretar dichas señales y darles un valor de acuerdo con la calibración, así como las instrucciones y recomendaciones del fabricante.

3.2.3.1. Sensor de fuerza

Para el caso del sensado de la fuerza, al ser un sensor resistivo y sumamente sensible, se deben adicionar elementos externos que ayuden a darle una interpretación correcta a la señal emitida por el sensor, de lo contrario si utilizamos

el sensor dentro de su configuración básica y lo tratamos como una resistencia variable mostrará lecturas que no tienen alguna interpretación y quedan lejos del valor realmente registrado.

Ante esta irregularidad el fabricante (TekScan), proporciona un circuito externo (Figura 25) para poder brindarle un mejor tratamiento a la señal de nuestro sensor, el cual consiste en utilizar un amplificador operacional (MCP6004) en una configuración inversora para posteriormente calibrar la sensibilidad del sensor a través de una resistencia de realimentación (R_f), la cual se sugiere sea una resistencia con valores muy altos para fuerzas pequeñas, mientras que para fuerzas muy grandes una resistencia pequeña.

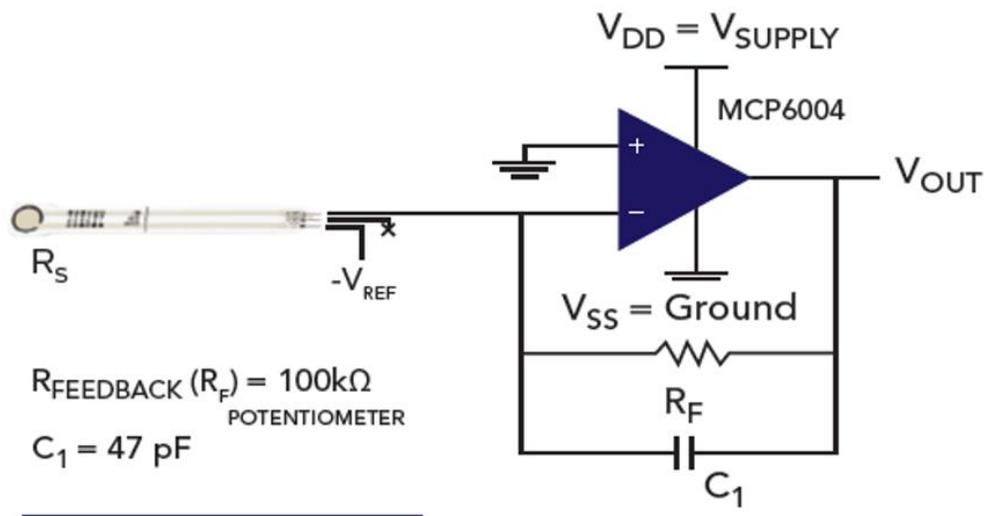


Figura 25 Circuito acondicionador de señal para flexiforce. Imagen extraída de [54] (Tekscan, s. f.)

En el diagrama se sugiere el uso de un potenciómetro de 100 Kohm; Sin embargo, para nuestro sistema dada la forma correcta de sujeción del trocar, esperamos fuerzas del orden de 10 a 30 N (Considerando una correcta inserción), siendo la fuerza de 15 N la más estable en experimentos con tejido humano como se vio en el segundo capítulo, es decir, una fuerza intermedia.

Dado lo anterior, se determinó experimentalmente utilizar una resistencia de 2 MOhm para poder adquirir los datos de la variación de fuerza al tomar el trocar con la técnica correcta.

Una vez determinada la resistencia, se simuló el circuito electrónico añadiendo 2 reguladores externos, para poder alimentar el circuito y generar la salida los cuales se describen a continuación:

1) Regulador -5V LM7905. Es un regulador lineal de voltaje fijo que presenta las características descritas en el apéndice No.2. Este regulador establece la entrada negativa al amplificador operacional, y dada la recomendación del fabricante ante un estímulo de fuerza pequeño se recomienda que $-V_{ref}$ (Figura 25) sea un valor elevado.

2) Regulador 3.3V LF33CV. Es un regulador de voltaje lineal fijo, capaz de suministrar 3.3V a su salida (apéndice No. 3). Con este regulador, vamos a suministrar la alimentación necesaria para nuestro circuito y al mismo tiempo generar el rango de valores posibles (0 a 3.3v) ante el estímulo externo.

Una vez definidos los elementos a utilizar, se procedió a simular el circuito previo al montaje en una placa electrónica. Como podemos observar, se simuló el sensor de fuerza con una resistencia variable RV4, a la cual se ingresa el voltaje de -5V y de acuerdo con la variación de la resistencia es la salida observada, tal como podemos apreciar en la Figura 26.

Cabe resaltar que, para propósitos de simulación, se utilizó un regulador de 5V (U9); sin embargo, en el montaje real ya se utilizó el regulador correcto manteniendo el mismo comportamiento, pero con un máximo de 3.3 V a la salida del amplificador operacional.

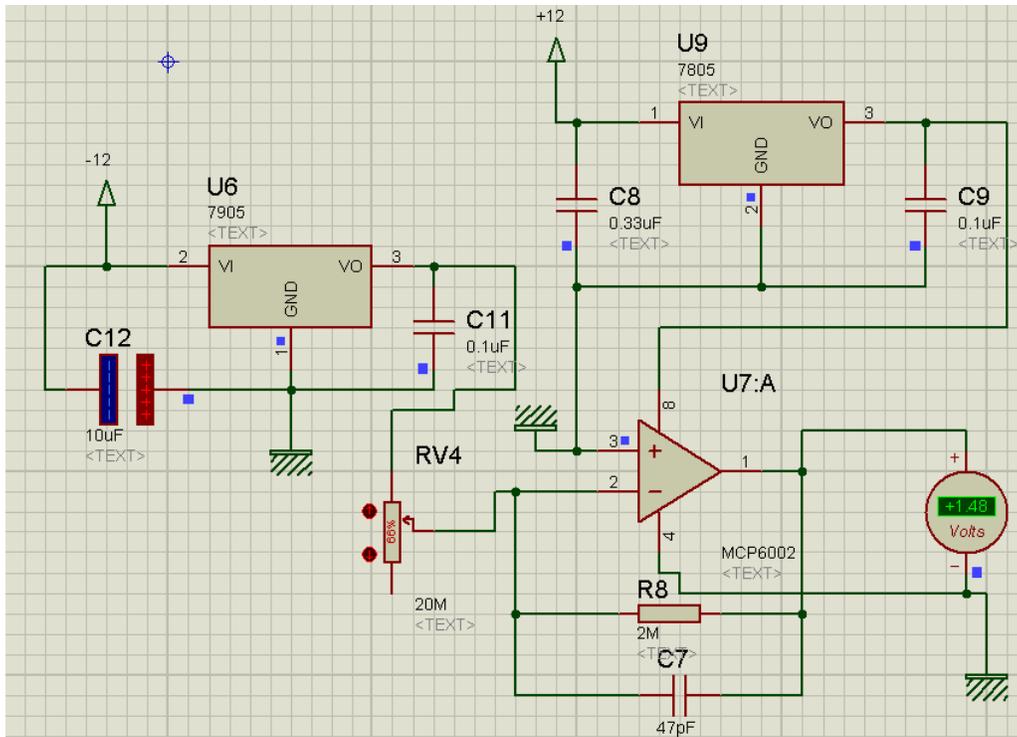


Figura 26 Simulación en proteus del circuito acondicionador de señal ante un estímulo

Una vez hechas las simulaciones, se procedió a realizar las primeras pruebas a través del circuito de la Figura 25, enviando la señal a la placa Arduino Nano, y utilizando un programa para adquirir la señal a través del convertidor analógico-digital del Arduino Nano.

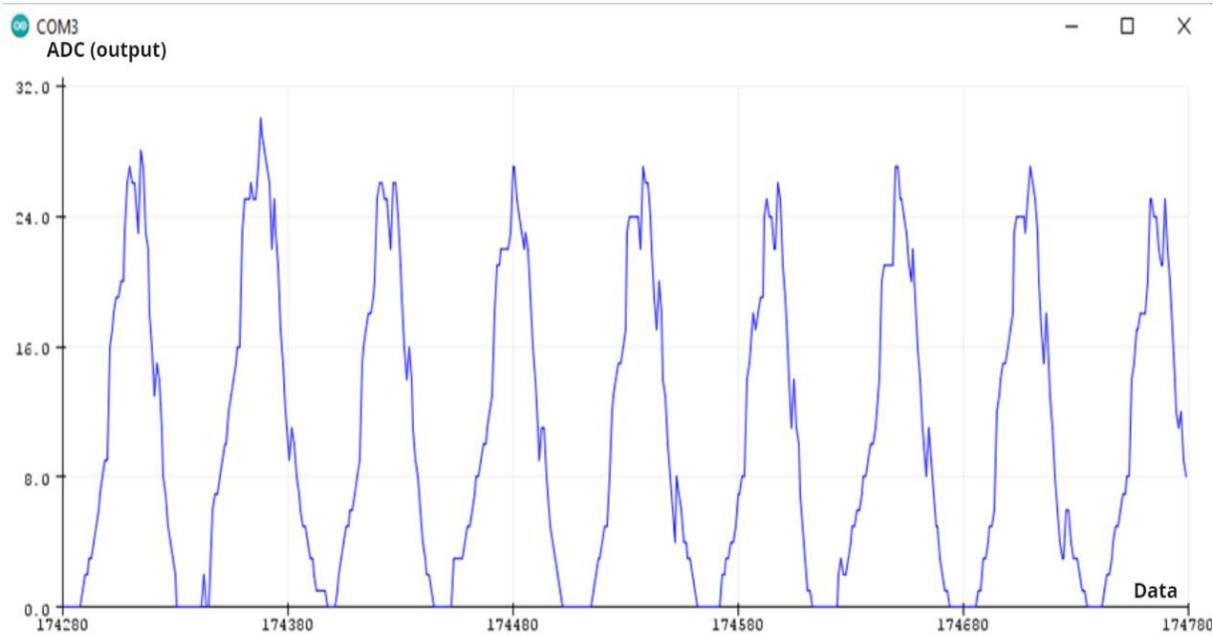


Figura 27 Señal del Sensor de Fuerza sin carga.

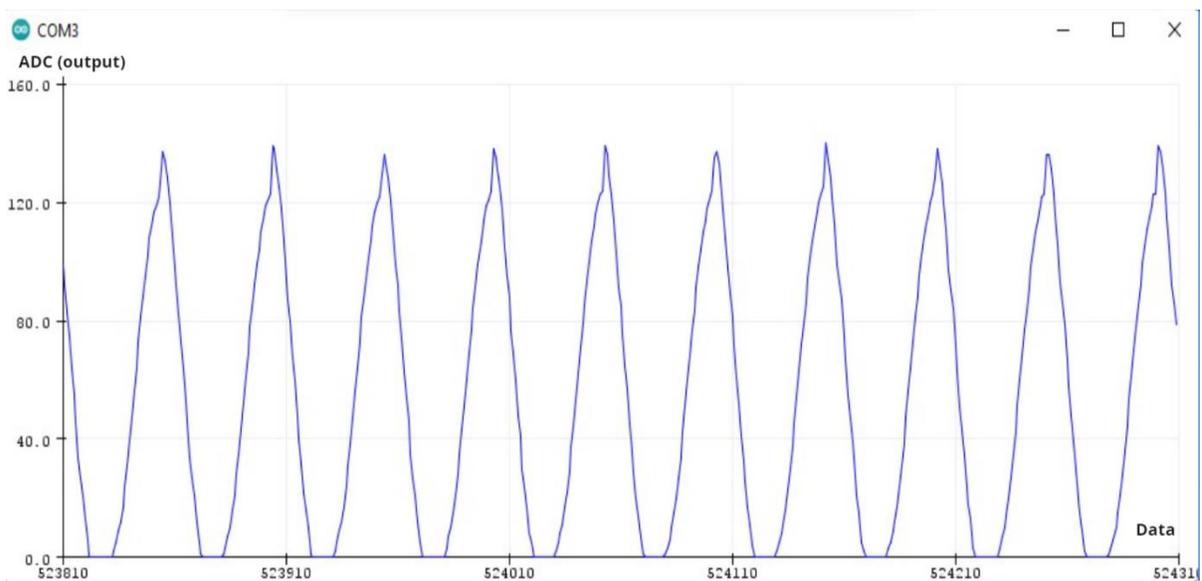


Figura 28 Señal del sensor de Fuerza con carga Máxima.

Lo anterior es una representación del comportamiento con la implementación del circuito de la Figura 25, dónde podemos ver claramente una mejora considerable en el rango de lecturas que va desde un valor en crudo (sin realizar su interpretación física) entre 0 y 32 sin carga (Figura 27), y un valor entre 0 y 120 con la aplicación de la carga máxima (Figura 28).

Ahora bien, lo anterior no significa la fuerza medida ni es aún una interpretación correcta del fenómeno físico, es simplemente la lectura que ofrece el convertidor analógico digital, el cual presenta un rango de valores entre 0 y 1023, correspondientes a 8 bits; donde 0 equivale a 0v y 1023 a 5v.

3.2.3.2. Sensor de presión

El sensor de presión cuenta con esta etapa de pre-procesamiento de manera interna, lo cual nos ejemplifica el fabricante a través del siguiente diagrama de bloques (Figura 29)

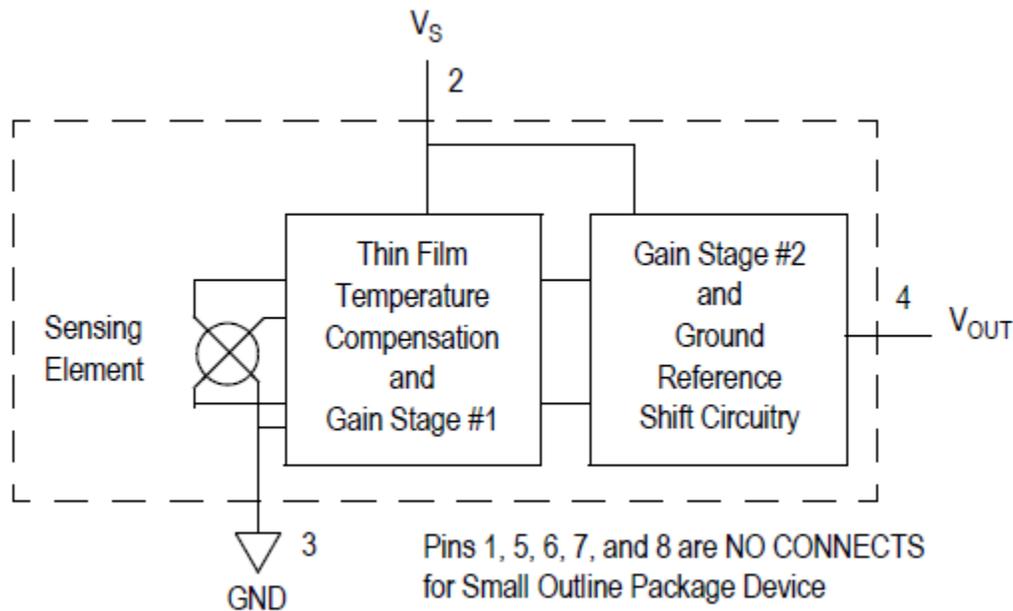


Figura 29 Diagrama de bloques del sensor MP3V5050GP. Imagen extraída de [35] (NXP Semiconductors, 2018)

La Figura 29 presenta de forma general la estructura interna del sensor, la cual presenta 2 etapas de amplificación con una determinada ganancia no proporcionada por el fabricante, así como una compensación por temperatura aplicación típicamente usada al utilizar sensores resistivos, además de la referencia a tierra para poder tener una salida de voltaje legible para el convertidor analógico-digital.

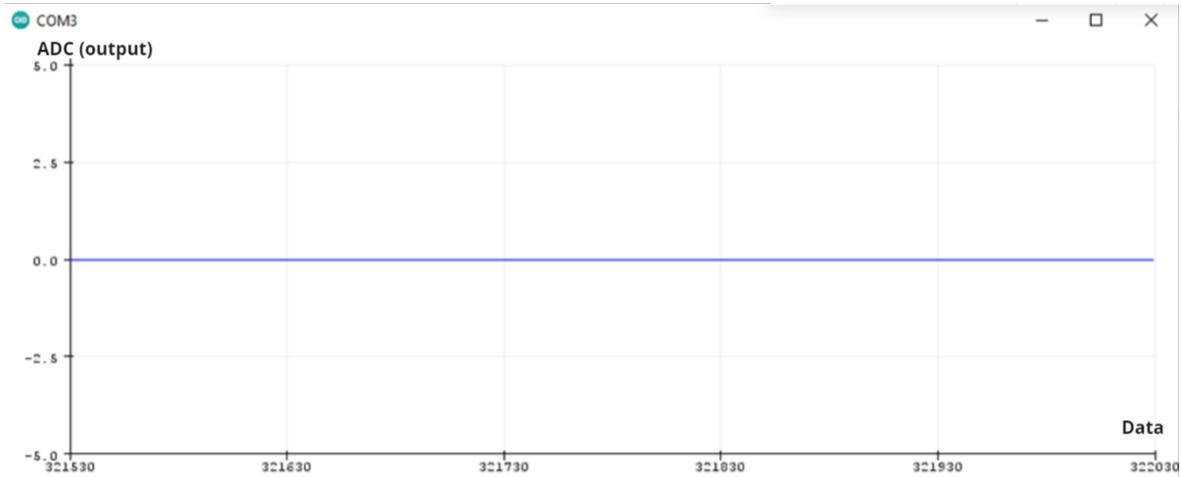


Figura 30 Gráfica de la lectura de presión sin activar la bomba

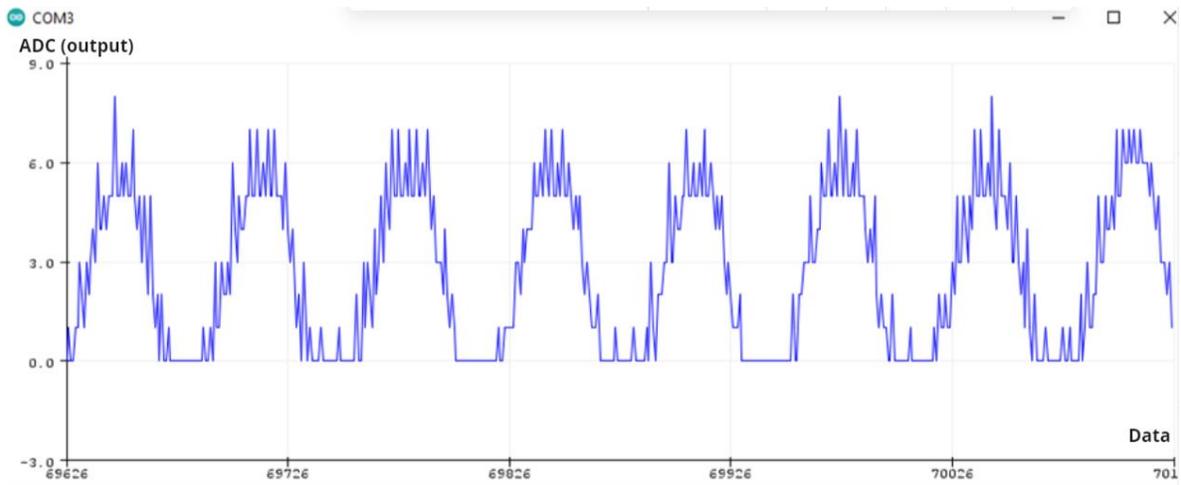


Figura 31 Gráfica de la lectura de presión 14.9 mmHg

Como podemos observar, el rango de valores posibles tiene una oscilación entre 0 y 6 para una lectura correspondiente a 14.9 mmHg (Figura 31), mientras que la presión de aproximadamente 0.1 mmHg, nos otorga un valor de 0 a la salida del conversor analógico digital (Figura 30).

Ambos valores presentan un rango sumamente pequeño para poder establecer el futuro modelo matemático necesario para poder entender el valor de la presión ejercida al interior de nuestro entrenador, por lo cual se exploró otra opción para poder obtener un rango de valores que presenten una mejor exactitud en la medición de los datos.

3.3. Especificaciones generales

En el presente apartado, se abordará la solución previa a la implementación de la calibración de los sensores debido a los problemas mencionados en el apartado anterior. Si bien el acondicionamiento de la señal proveniente del sensor nos brindó una señal entendible y medible a través de nuestro multímetro, al intentar interpretar dicha señal por medio de nuestra tarjeta de adquisición de datos Arduino Nano, se presentó uno de los principales inconvenientes al utilizar el mismo, y es que nuestro microcontrolador ATmega328p presenta un convertidor analógico digital de 8 bits.

3.3.1. El convertidor analógico – digital (ADC)

El convertidor analógico digital es un proceso para poder “convertir señales analógicas en palabras binarias” (Bolton et al., 2013). Mediante el proceso descrito por la Figura 32, de manera general transformar una señal analógica en una digital se compone de 2 fases fundamentales, la primera el muestreo y retención (Figura 33) la cual mapea la señal analógica y la segunda interpreta la señal mapeada.

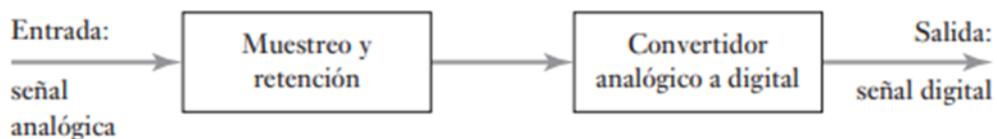


Figura 32 Diagrama de bloques general de un ADC [6] (Bolton et al., 2013)

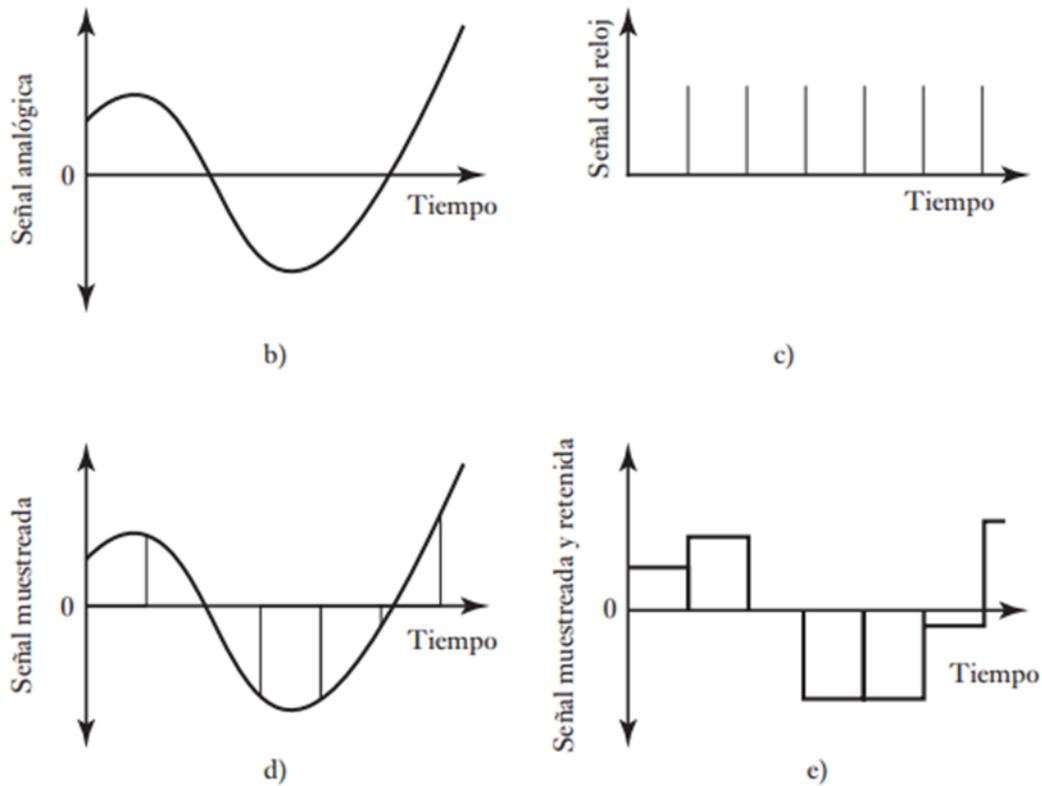


Figura 33 4 Fases en la conversión [6] (Bolton et al., 2013)

La segunda parte de la conversión consiste en realizar una interpretación de la señal teniendo una escala de referencia, la cual depende de la resolución de nuestro ADC, a este proceso se le conoce como cuantización de la señal (Figura 34), sin embargo, existe un error directamente relacionado dicha resolución (error de cuantización), por lo que, para una lectura con mayor precisión se requiere una mayor resolución.

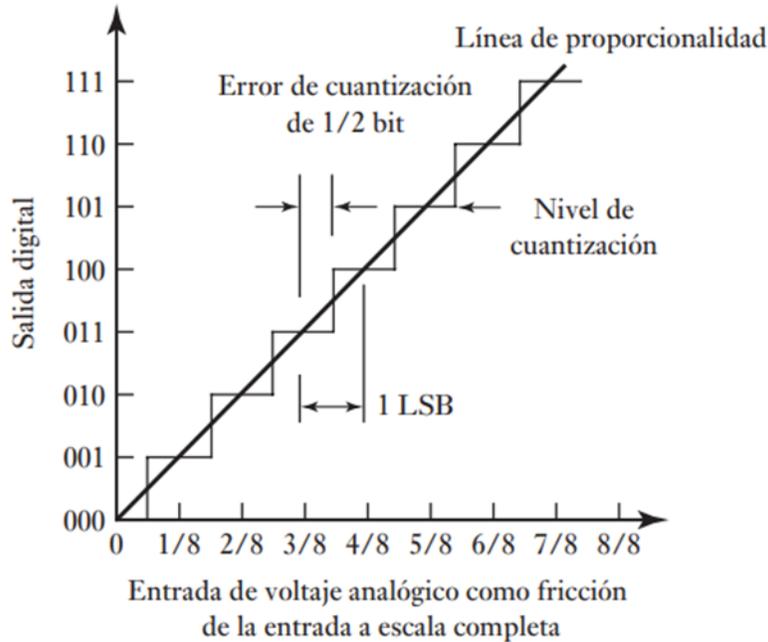


Figura 34 Cuantización de un ADC [6] (Bolton et al., 2013)

3.3.2. Convertidor ADS1115

El convertidor ADS 1115 es un convertidor analógico-digital de 16 bits perteneciente al modo sigma delta, el cual incorpora un amplificador de ganancia programable (que va desde los 256mv hasta los 6.14 V a la entrada), así como un comparador digital, un oscilador de reloj y un protocolo de comunicación I2C el cual permite la conexión de hasta 4 dispositivos conectados al mismo bus de datos, para ello se dispone de 4 direcciones (GND, VCC, SCL y SDA). (Figura 35).

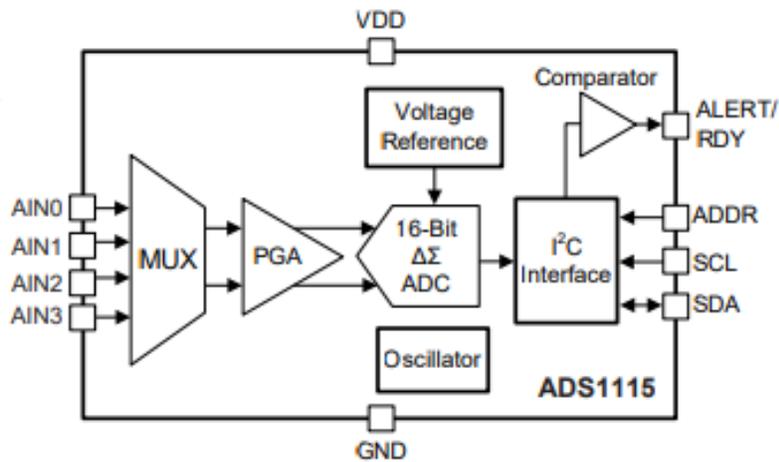


Figura 35 Estructura general del convertidor ADS1115. Imagen extraída de [56] (Texas Instruments, 2018)

Cabe resaltar que, a la salida del multiplexor (Figura 35), encontramos un bloque PGA programable, lo que significa que mediante una codificación externa podemos seleccionar el nivel de ganancia de acuerdo con la Tabla 6, es decir, dichas líneas permiten tener una diferente resolución al operar el convertidor.

PGA (ADS1115)		
FSR (V)	LSB size (μV)	Code
6.144	187.5	GAIN_TWOTHIRDS
4.096	125	GAIN_ONE
2.048	62.5	GAIN_TWO
1.024	31.25	GAIN_FOUR
0.512	15.625	GAIN_EIGHT
0.256	7.8125	GAIN_SIXTEEN

Tabla 6 Ganancia programable, extraída de [10] (Del Valle, 2022)

El voltaje FSR, indica el máximo valor de voltaje admitido en cada entrada (AIN), mientras que LSB, el bit menos significativo nos da la resolución de mínima lectura que el convertidor puede ser capaz de registrar, estos valores obedecen a la siguiente fórmula:

$$LSB = \frac{FSR}{2^n}$$

Ecuación 1 Resolución mínima.

3.3.3. Comparativa convertidor Arduino y comparador ADS 1115

	ADC Arduino	ADS 1115
Resolución	10 [Bit]	16 [Bit]
Precisión (FS)	4.88 [mV]	(PGA = 6.144) 93.75 [uV]
Voltaje de referencia	1.8V - 5V	6.144, 4.096, 2.048, 1.024, 0.512, 0.256 [V]
Offset error	2 LSB (9.76 [mV])	+ - 3 LSB (281.25 [uV])
LSB (FS)	4.88 [mV]	(PGA = 6.144) 93.75 [uV]
Frecuencia de muestreo	15 [kSPS] escala absoluta.	8, 16, 32, 64, 128, 250, 475, 860 [SPS]
Tipo de comunicación		I2C
Voltaje de alimentación	1.8 - 5.5 [V]	2 - 5.5 [V]
Temperatura de operación		-60 [°C] - 150 [°C]
Impedancia de entrada		(FS) 10 [MOhm]
Consumo de corriente		150 [uA]

Tabla 7 Comparativa entre ADC Arduino y ADS 1115

3.4. Calibración de sensores

3.4.1. Calibración sensor de presión

Una vez hecha la selección del convertidor analógico digital, se procede a la integración de cada uno de los elementos necesarios para poder realizar la adquisición de datos. En este caso, el ensamble de la tarjeta “Arduino Nano” así como la respectiva programación enfocada en la adquisición de las diferentes señales provenientes de los sensores.

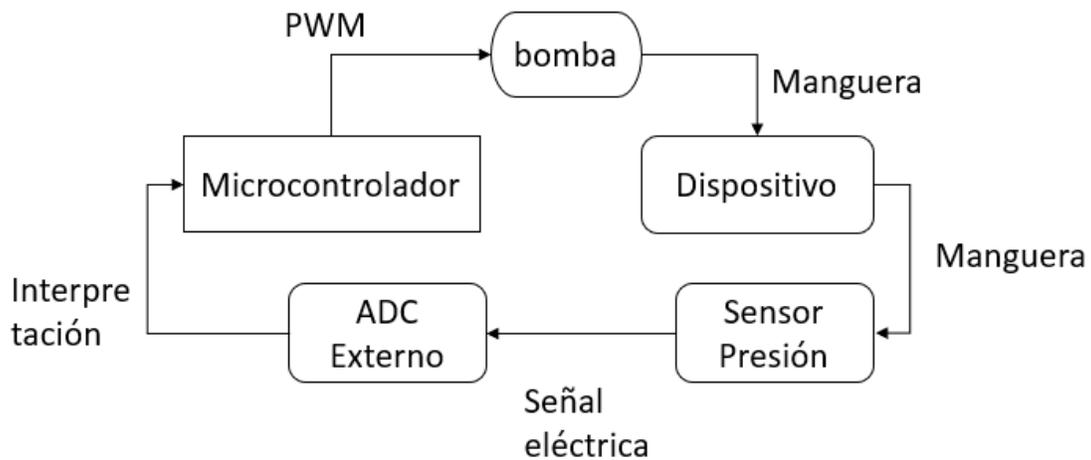


Figura 36 Flujo de adquisición en fase de prueba.

Como podemos observar en el anterior diagrama (Figura 36), la señal proveniente de los sensores previamente ensamblados a su lugar final será interpretadas por el convertidor analógico-digital externo, esto debido a las ventajas que tiene sobre el convertidor natural del microprocesador Atmega.

Para poder calibrar el sensor de presión, se generaron diferentes presiones a través de nuestra bomba de insuflación. Dicha bomba es controlada por el propio Arduino Nano al ofrecer señales PWM de diferente magnitud, que provocan en consecuencia presiones de diferente magnitud a su vez.

3.4.1.1. Fases de obtención de datos

Como primer paso, se obtuvieron distintas lecturas de una presión constante generadas a partir de la instrucción al Arduino Nano, el cual generó diferentes señales PWM para poder regular la potencia de la bomba. Una vez hecha la lectura, se procedió a mostrar en tiempo real las lecturas del microcontrolador (Figura 37).

```
[ [124.60,1748.70  
[ [123.90,1748.30  
[ [124.20,1748.50  
[ [124.60,1748.70  
[ [123.90,1748.50  
[ [123.00,1748.40  
[ [125.10,1748.40  
[ [123.60,1748.90  
[ [124.50,1748.10  
[ [124.60,1748.40  
[ [124.60,1748.80  
[ [124.60,1748.70  
[ [123.80,1748.30  
[ [124.40,1748.40  
[ [124.20,1748.80
```

Figura 37 Calibración y obtención de los datos.

Con los datos obtenidos en el IDE de Arduino, se procedió a crear una plantilla en el programa Excel, la cual contuviera en la columna "A", la información de 1000 muestras, para posteriormente procesarlas a través de Python.

En la presente etapa, se creó un programa capaz de analizar el archivo en Excel con las 1000 muestras y generar un reporte de salida con la información de las muestras agrupadas por valor y número de repeticiones, ordenándose de mayor número de repeticiones hasta el menor número de repeticiones, consiguiendo así el valor que es más susceptible de adquirir al realizar un estudio con una determinada presión constante.

Una vez hecho el análisis de los datos, se lograron generar 11 archivos Excel, con la información correspondiente al equivalente digital de los distintos valores de presión (desde 6 hasta 16 mmHg) (ver Tabla 8).

X(ADC)	Y(Presión[mmHg])
109.3	6.2
113.7	7.3
117.3	8.2
121	9.1
124.2	9.9
128.3	10.9
131.1	11.7
135.2	12.5
139.8	13.9
143.9	14.9
147.8	15.9

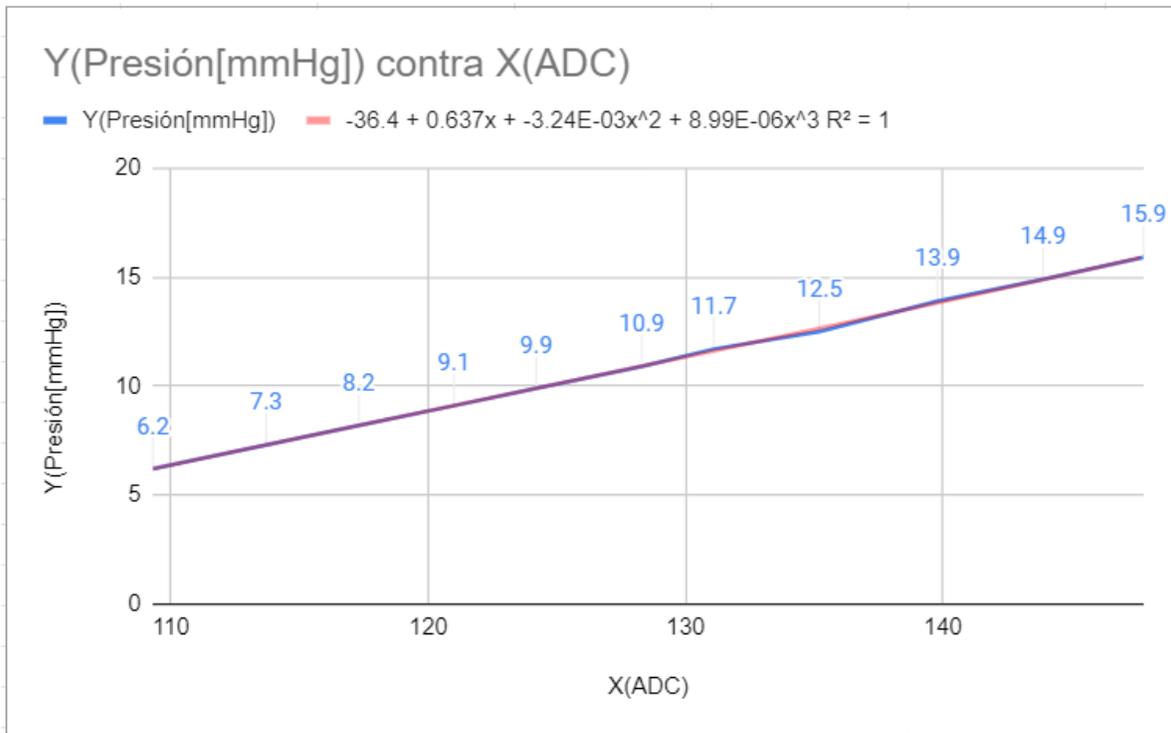
Tabla 8 Relación entre la presión registrada y la lectura obtenida en el conversor analógico digital.

Con los valores clave obtenidos en el punto anterior, se generó una gráfica de los valores de presión y su correspondiente valor digital, y con ello logramos obtener la gráfica 3, la cual nos muestra el comportamiento de la presión durante la calibración del sensor. A su vez, logramos obtener el modelo matemático (ecuación 2) de la presión en función de la lectura digital.

$$Presión(mmHg) = F(analogInput)$$

$$Presión(x) = 0.00000899x^3 - 0.00324x^2 + 0.637x - 36.4$$

Ecuación 2 Modelo matemático de presión vs lectura adc



Gráfica 1 Gráfica de presión contra lectura de nuestro ADC.

3.4.2. Calibración del sensor de fuerza

Para poder realizar la calibración del sensor de fuerza, una vez hecho el acondicionamiento de la señal emitida por dicho sensor, se procedió a utilizar una báscula electrónica para poder adquirir diferentes medidas del peso generado al aplicar una determinada fuerza sobre el punto de contacto.

Se realizaron 15 muestras en la aplicación de una fuerza directa sobre el sensor, donde cada una de ellas fue generada con diferentes niveles de fuerza. Debido al fenómeno físico que implica nuestro estudio, la punción que se da, sucede en orden de milisegundos, por ello, fue necesario medir la cantidad de voltaje emitida por el sensor al momento de aplicar una fuerza constante sobre el punto de aplicación.

Como auxiliar, se utilizó una base de soporte para poder evitar el contacto sobre una superficie fuera de la mayor área de sensibilidad, para nuestro sensor Flexiforce, la cual se encuentra rodeando el punto central del mismo. Una vez hecho esto, se tomó con una mano el trocar en la posición correcta de inserción y

se procedió a tomar diferentes muestras las cuales arrojaron los valores descritos en la tabla siguiente:

Fuerza(g)	Fuerza(N)
2.2	21.582
1.557	15.27417
1.311	12.86091
1.028	10.08468
0.818	8.02458
0.654	6.41574
0.59	5.7879
0.402	3.94362
0.345	3.38445
0.3	2.943
0.18	1.7658
0.165	1.61865
0.104	1.02024
0.083	0.81423
0	0

Tabla 9 Datos fuerza en Newtons.

Cabe resaltar que los valores de fuerza en gramos fueron transformados a Newton's (N) utilizando como valor de gravedad $g = 9.81 \left[\frac{m}{s^2} \right]$.

Una vez adquiridos los valores de voltaje (V) y fuerza (N), procedimos a utilizar como base la referencia máxima de voltaje, la cual es de 3.3V idealmente. Aunque la fase de acondicionamiento de señal regularmente arroja valores de hasta 0.03 superior e inferior, la referencia inicial en un estado sin carga es 3.3V, y a medida que se incrementa la fuerza el valor de voltaje decrece, es decir, tenemos un comportamiento inversamente proporcional a la salida del módulo de acondicionamiento de señal.

Una vez adquiridos los datos se procedió a adaptarlos para su mejor entendimiento manteniendo en todo momento un orden de proporcionalidad. Se

transformaron los valores de voltaje de voltios a milivoltios con el objetivo de poder transformar correctamente el valor de lectura del convertidor analógico digital.

Hecho lo anterior, se procedió a dividir el valor en milivoltios con el factor de escala para el convertidor al tener como referencia un voltaje de 6,144 V, procediendo a obtener el valor binario después de realizar la conversión ADC.

Posteriormente, se obtiene una nueva relación entre la fuerza en N y el valor en binario a la salida del ADC. Ahora bien, para poder facilitar el análisis de la curva generada, se procedió a invertir el valor binario, es decir, se tomó como base el valor binario que representa un 0 como 17600, y de esta forma iniciar la curva con un origen en 0, en lugar de comenzar la misma con su origen en 2187.

Voltaje	Fuerza(g)	X(ADC)	Fuerza(N)
0.41	2.2	2187	21.582
1.54	1.557	8213	15.27417
1.76	1.311	9387	12.86091
2.05	1.028	10933	10.08468
2.35	0.818	12533	8.02458
2.72	0.654	14507	6.41574
2.83	0.59	15093	5.7879
2.9	0.402	15467	3.94362
2.93	0.345	15627	3.38445
3	0.3	16000	2.943
3.15	0.18	16800	1.7658
3.2	0.165	17067	1.61865
3.24	0.104	17280	1.02024
3.25	0.083	17333	0.81423
3.3	0	17600	0

Tabla 10 Relación entre la lectura del sensor contra la fuerza registrada.

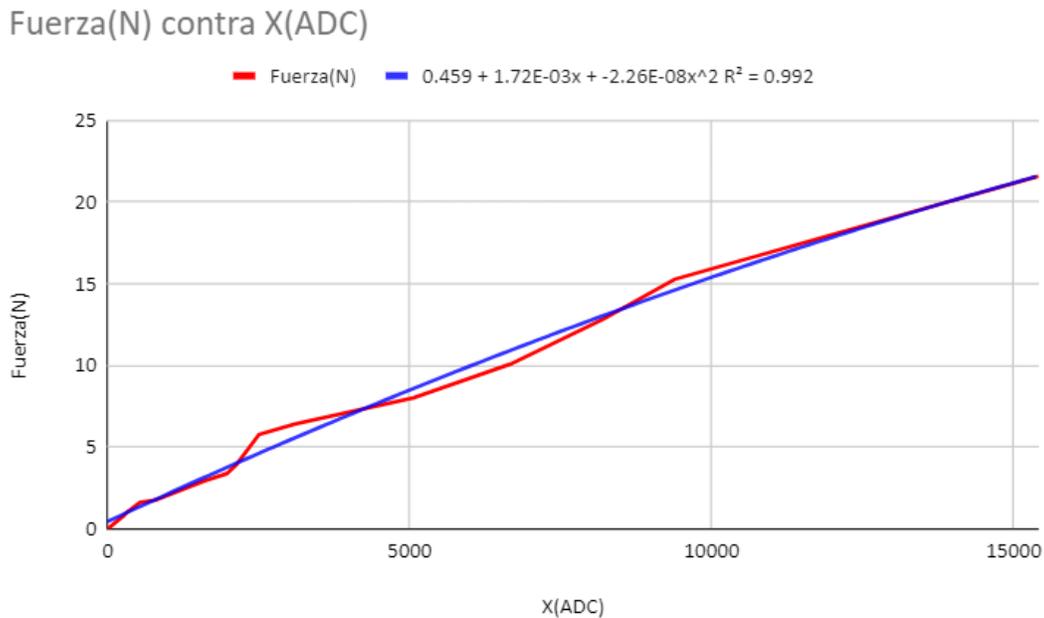
Con este ajuste matemático (Tabla 10), se logró tener una ecuación que mantiene la misma proporción y evidentemente tendrá el mismo resultado esperado para cada valor del sensor de tal forma que se generó una curva que satisface la condición:

$$Fuerza(N) = F(analogInput)$$

Dada la anterior relación, se obtuvo el siguiente gráfico (gráfica 4), donde podemos apreciar la curva de muestreo, así como la línea de tendencia y su relación matemática, en donde de esta gráfica podemos obtener el siguiente modelo matemático (Ecuación 3):

$$Fuerza(N) = -0.0000000226x^2 + 0.00172x + 0.459$$

Ecuación 3 Modelo matemático de fuerza



Gráfica 2 Relación entre la fuerza contra la lectura de nuestro convertidor ADC.

3.5. Placa de desarrollo

3.5.1. Sistema de adquisición de datos

Un sistema de adquisición de datos es el conjunto de elementos electrónicos que, en su conjunto, realizan la lectura de variables periféricas a través de sensores, les dan una interpretación y arrojan un resultado. En el presente proyecto, ocupamos una tarjeta Arduino Nano para poder realizar la lectura de los 2 sensores, así como el control de la potencia de la bomba.

La placa Arduino Nano (Figura 38) fue apropiada para el presente proyecto puesto que las entradas necesarias para la lectura de nuestras variables físicas presión y

fuerza fueron 2. Además, en conjunto con el ADC externo obtenemos una mejor precisión en la lectura de los valores emitidos por ambos sensores.

Por otra parte, para el control de la bomba se necesitó una salida analógica con un control PWM para poder variar la potencia de esta. Asimismo, los datos adquiridos y procesados se envían a través del puerto serie, mismos que utilizamos para el control del flujo de aire, por lo que no existe la necesidad de añadir otro puerto serie o algún otro medio de comunicación con la interfaz de usuario.

La placa Arduino nano cuenta con las siguientes características principales como lo es tener un microcontrolador ATmega328p, 14 pines de entrada/salida, 6 pines PWM, 6 entradas analógicas, 1 cristal de 16Mhz, conexión mini usb, botón de reset y 5 v como voltaje de operación

Protocolos de comunicación:

1. UART: Recepción transmisión de datos asíncrona. Dicha comunicación utiliza una línea de transmisión para poder enviar datos y otra para recibir donde 8 bits son utilizados para el envío de datos, un bit en estado bajo para el inicio de la transmisión y un bit en estado alto para el término de esta.
2. SPI: Sistema de comunicación simple a través del cual un maestro envía a un esclavo una señal de reloj y a través de este cada pulso se envía un bit al esclavo y se recibe otro bit.
3. I2C: Protocolo ampliamente usado el cual consta de 2 cables para poder realizar la comunicación la primera es utilizada para los datos (SDA) y la segunda para la señal de reloj (SLK). Cabe resaltar que no se utiliza un esclavo sino un direccionamiento lo que permite conectar múltiples dispositivos a una misma línea de comunicación lo que permite el ahorro de cables y circuitos periféricos, tal como se usa en el ADC externo.

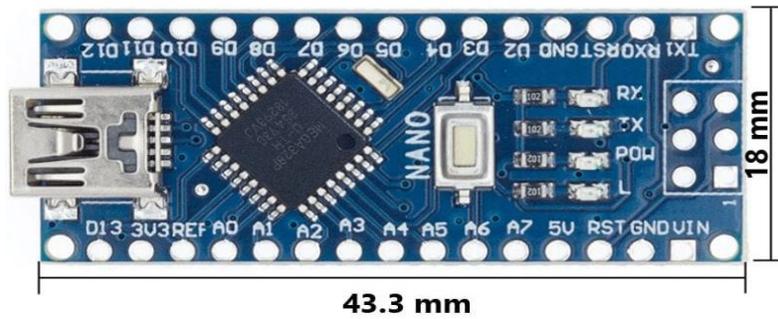


Figura 38 *Arduino nano*

3.5.2. Diseño de diagrama eléctrico y simulación

Para realizar la adquisición de las señales provenientes de los sensores, el procesamiento de los datos, así como el control del insuflador, se diseñó una placa PCB que compacta toda la electrónica necesaria para el funcionamiento de los sensores, la tarjeta de adquisición de datos y la operación gradual del insuflador.

Para su diseño, se utilizó el programa Eagle de Autodesk, con el cual se planteó el diseño del diagrama esquemático dónde se realizan las conexiones de cada uno de los elementos necesarios para el funcionamiento del sistema electrónico.

Dentro de las librerías de Eagle se tiene un gran número de componentes disponibles; sin embargo, en el proyecto se necesitó crear el módulo LM2596, el cuál es un convertidor step-down (Buck) DC-DC. Además, se adquirió de Snapeda los diagramas correspondientes al Arduino nano, amplificador operacional MCP6002 y el mosfet IRL520N.

En la Figura 39 podemos observar las diferentes secciones y elementos contenidos en el diagrama esquemático, así como las conexiones con cada uno de los componentes.

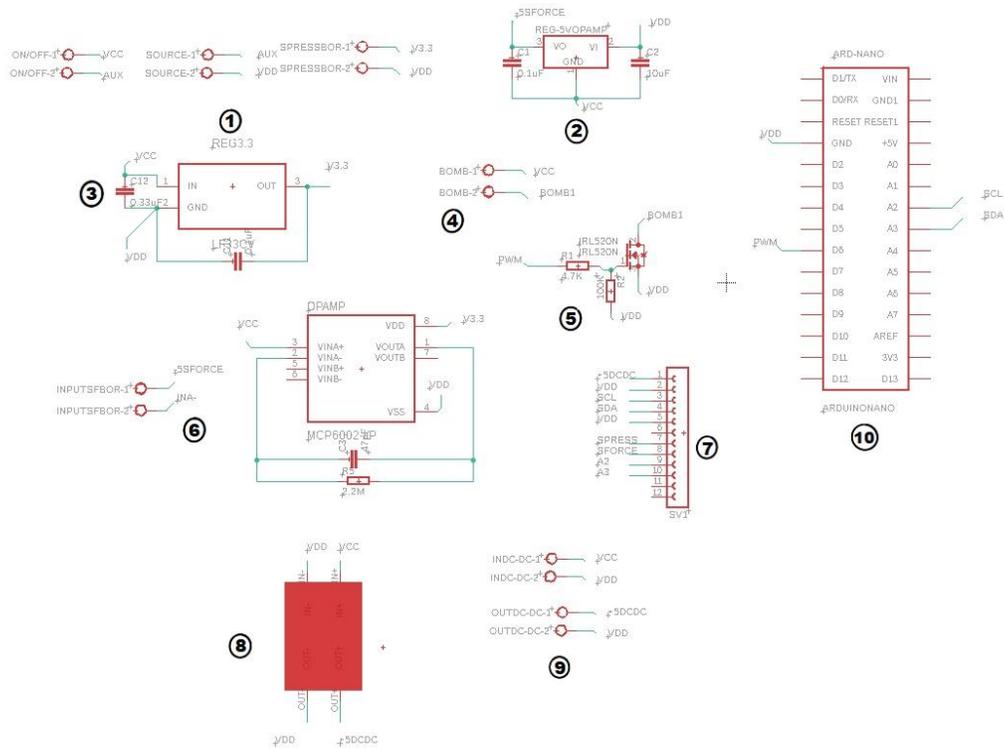


Figura 39 Diagrama esquemático.

En la Tabla 11 podemos ver la descripción de cada uno de los componentes:

Sección	Nombre	Descripción
1	Borneras de alimentación.	En dichas borneras colocamos la salida +3.3 [V], como alimentación del sensor de presión. En las borneras aledañas colocamos el sistema de encendido.
2	Regulador -5 V	Regulador hacia sensor de fuerza.
3	Regulador 3.3 V	Regulador fuente de alimentación del sensor de presión, así como voltaje de referencia en el amplificador operacional.
4	Borneras del insuflador	Alimentación y regulación de la potencia del insuflador, salida.
5	Mosfet control del insuflador	Elemento de control la variación de PWM provocará el aumento o disminución de la potencia en la bomba, la conexión es hacia la salida PWM del Arduino nano, y una de las borneras hacia el insuflador.

6	Amplificador operacional	Es el elemento encargado del acondicionamiento de la señal proveniente del sensor de fuerza se alimenta con 3.3[V],
7	Headers módulo convertidor analógico digital.	Se utilizaron headers tipo hembra para la colocación del módulo ADS1115.
8	Convertidor step down.	Elemento de regulación de voltaje DC-DC, utilizado como fuente de alimentación del módulo ADS1115 y la bornera de salida de voltaje regulado a +5[V].
9	Borneras auxiliares con salida a +5V	Salidas auxiliares con un voltaje regulado a +5[V]
10	Arduino Nano	Tarjeta de adquisición de datos.

Tabla 11 Componentes del diagrama esquemático.

3.5.3. Diseño electrónico en placa PCB

Una vez realizado el diagrama esquemático de nuestra placa de desarrollo, se adaptó en un espacio de 122.3 mm x 96.5 mm, dónde se colocó la etapa de potencia a la derecha de la placa electrónica (Figura 40), mientras que la entrada de las señales de los sensores se adaptó del lado superior e izquierdo de la misma.

Cabe mencionar que la entrada de la fuente de alimentación se colocó en la parte inferior de esta placa para tener una disposición adecuada a la salida de la fuente.

Se utilizaron 2 capas en la placa, la primera (color rojo) es la capa superior correspondiente principalmente a conexiones entre componentes, así como la parte positiva de la fuente de alimentación, mientras que la capa inferior corresponde al GND (tierra) de nuestra placa.

En cuanto al rectángulo colocado al lado de la tarjeta de adquisición de datos, es el regulador de voltaje DC-DC, el cuál fue diseñado previamente como un componente especial y añadido como un componente de una librería personalizada de Eagle software.

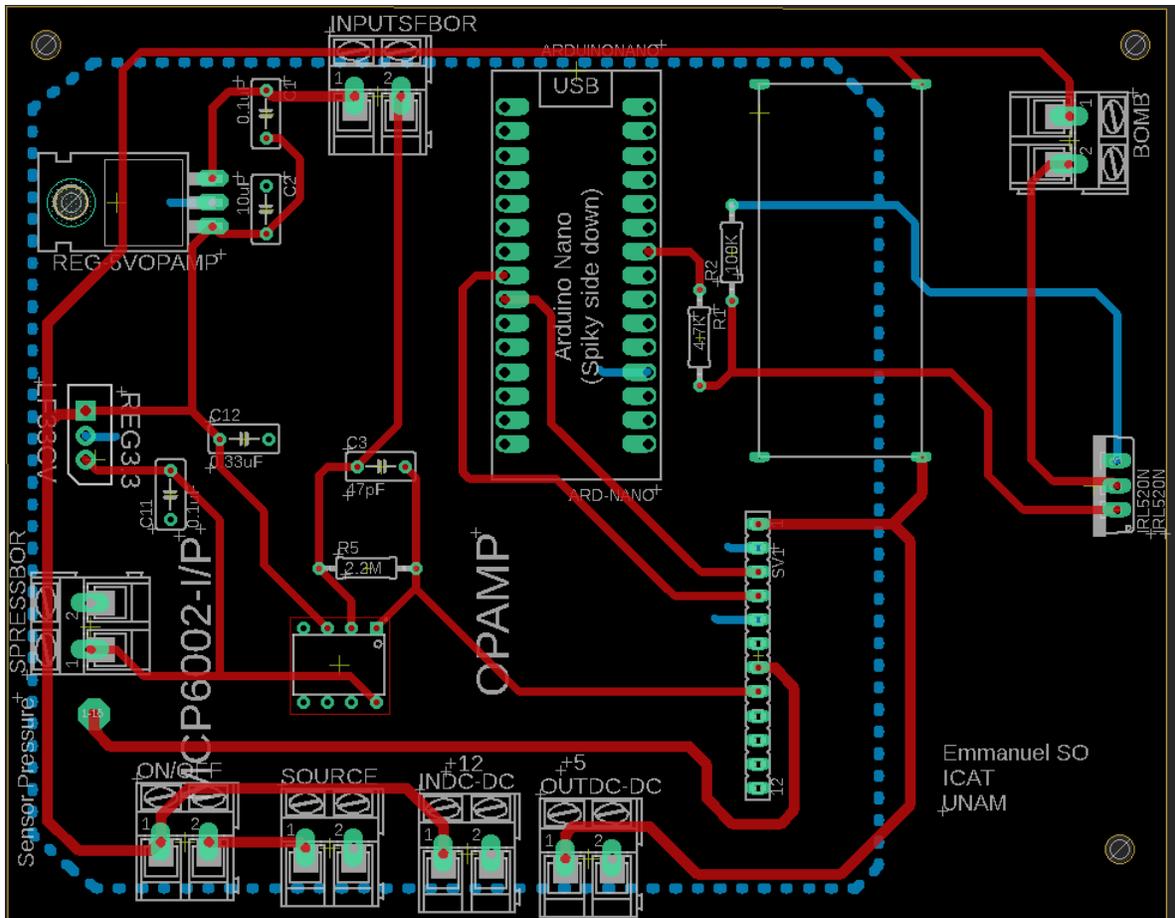


Figura 40 Diagrama pcb

Cabe mencionar que la fuente de alimentación (Figura 41) para todo el proyecto fue una fuente conmutada que ofrece una salida de 5V a 10A. Esta fuente posee las especificaciones eléctricas citadas en la tabla informativa No. 2, donde encontramos características suficientes para la correcta operatividad de todo nuestro proyecto desde la alimentación a la electrónica digital como la alimentación hacia la propia bomba que corresponde a la electrónica de potencia.



Figura 41 Fuente de alimentación.

3.6. Sistema de insuflación

El sistema de insuflación o neumoperitoneo implica añadir un gas inerte al interior del cuerpo humano con el fin de separar órganos internos y así generar la cavidad necesaria para poder tener una adecuada manipulación de los instrumentos médicos al interior del paciente.

Por lo tanto, el generar un ambiente de entrenamiento dónde existe la dosificación de un gas es indispensable para poder establecer un entorno de entrenamiento similar al que se obtendría con un sistema laparoscópico real.

Para lograr lo anterior, se realizó la adaptación y regulación de la potencia de una bomba de aire, la cual genera una potencia de hasta 50 w. Lo anterior se traduce en el aumento o disminución de la presión al interior del dispositivo de entrenamiento.

Para poder lograr estos incrementos y decrementos de potencia se adaptó un transistor tipo mosfet para utilizar con microcontroladores (Serie "IRL"). Al variar la entrada de voltaje proveniente del pin gate, se puede controlar la cantidad de corriente que podemos drenar hacia nuestra fuente, como podemos observar en el siguiente diagrama (Figura 42).

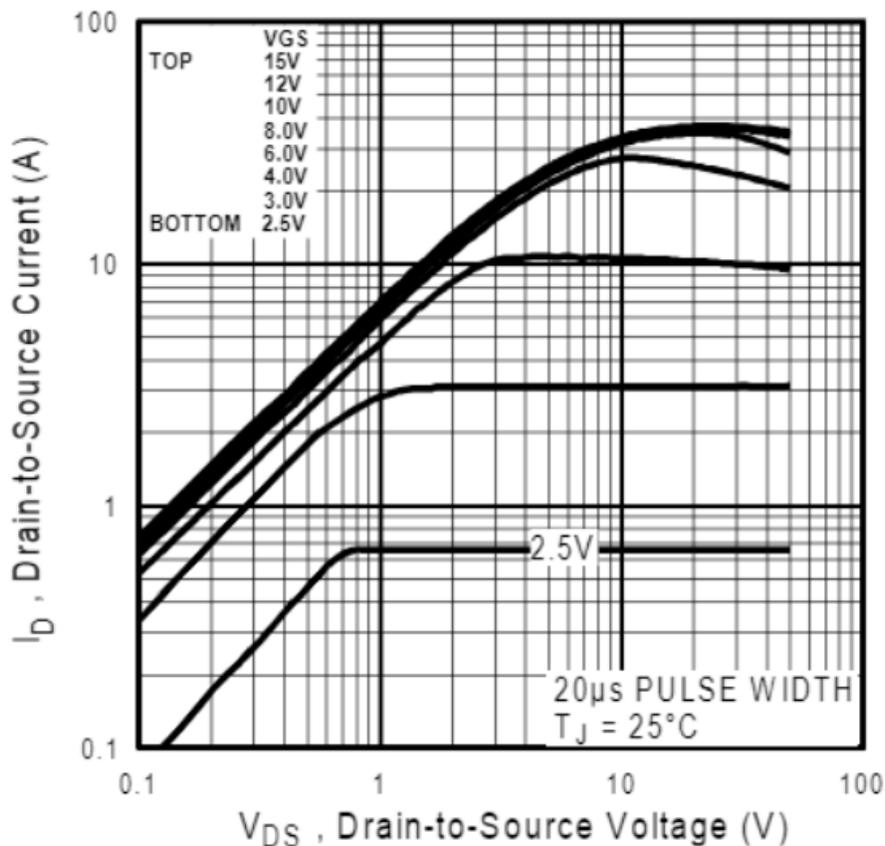


Figura 42 Variación de corriente en función de la entrada de voltaje aplicado, utilizando el IRL 520. Imagen extraída de la hoja de datos del fabricante. Imagen extraída de [20] (International Rectifier, s. f.)

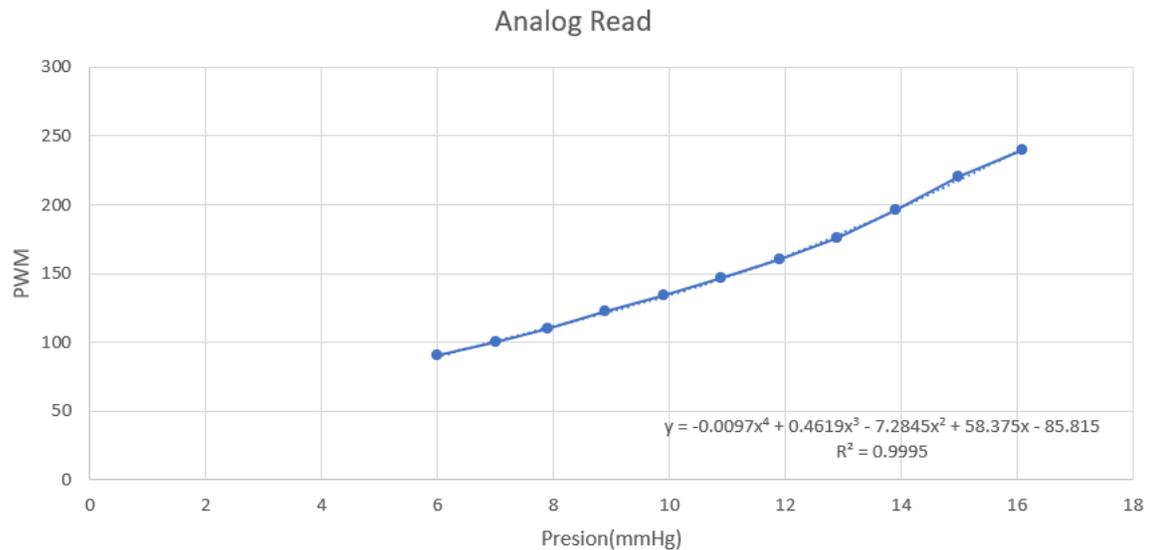
Con ello, logramos realizar una variación en la potencia generada por la bomba y así lograr manipular la presión al interior del dispositivo de entrenamiento. Además, para lograr el control de las variaciones de voltaje, se procedió a utilizar la modulación PWM, proveniente de nuestro Arduino Nano, la cual utiliza hasta 8 bits, 255 divisiones para un voltaje de referencia de 5 V.

3.6.1. Sistema de dosificación del gas

La generación de las señales PWM se determinaron experimentalmente al variar el valor PWM y registrar su equivalente en presión calculada en mmHg (Tabla 12), con ello procedimos a realizar el análisis de la Gráfica 5.

Analog Read	Voltaje	Presión (mmHg)
90	1.5	6
100	1.64	7
110	1.78	7.9
122	1.95	8.9
134	2.12	9.9
147	2.31	10.9
160	2.49	11.9
176	2.72	12.9
196	3	13.9
220	3.35	15
240	3.63	16.1

Tabla 12 variación de PWM con presión generada.



Gráfica 3 Variación de salida PWM con respecto a la entrada de presión deseada.

La gráfica anterior nos muestra la relación que existe entre la variación PWM y la presión generada por la bomba. A partir del análisis de los datos y la misma gráfica generamos el siguiente modelo matemático (Ecuación 4):

$$PWM = f(Presión)$$

$$PWM(x) = -0.0097x^4 + 0.4619x^3 - 7.2845x^2 + 58.375x - 85.815$$

Ecuación 4 Modelo matemático insuflador

Este modelo obedece el ingreso idealmente de números enteros ya que en un entorno real se ingresan presiones de alrededor de 10 y 12 mmHg, lo cual, se logra satisfacer exitosamente puesto que nuestro modelo matemático logra cubrir desde los 6 mmHg hasta los 16 mmHg, con lo anterior, logramos un ambiente de entrenamiento ideal para diferentes pacientes, es decir desde paciente pediátricos hasta pacientes adultos.

3.7. Programación

3.7.1. Desarrollo de código aplicado al microcontrolador

El código utilizado para el control del insuflador (Figura 43), así como la interacción entre el microcontrolador y la computadora fue desarrollado en el IDE de Arduino. La estructura del código comienza con la importación de la librería "Adafruit_ADS1X15.h", con la que lograremos adquirir la información proveniente del convertidor analógico digital mediante la creación de un objeto de la clase Adafruit_ADS1115.

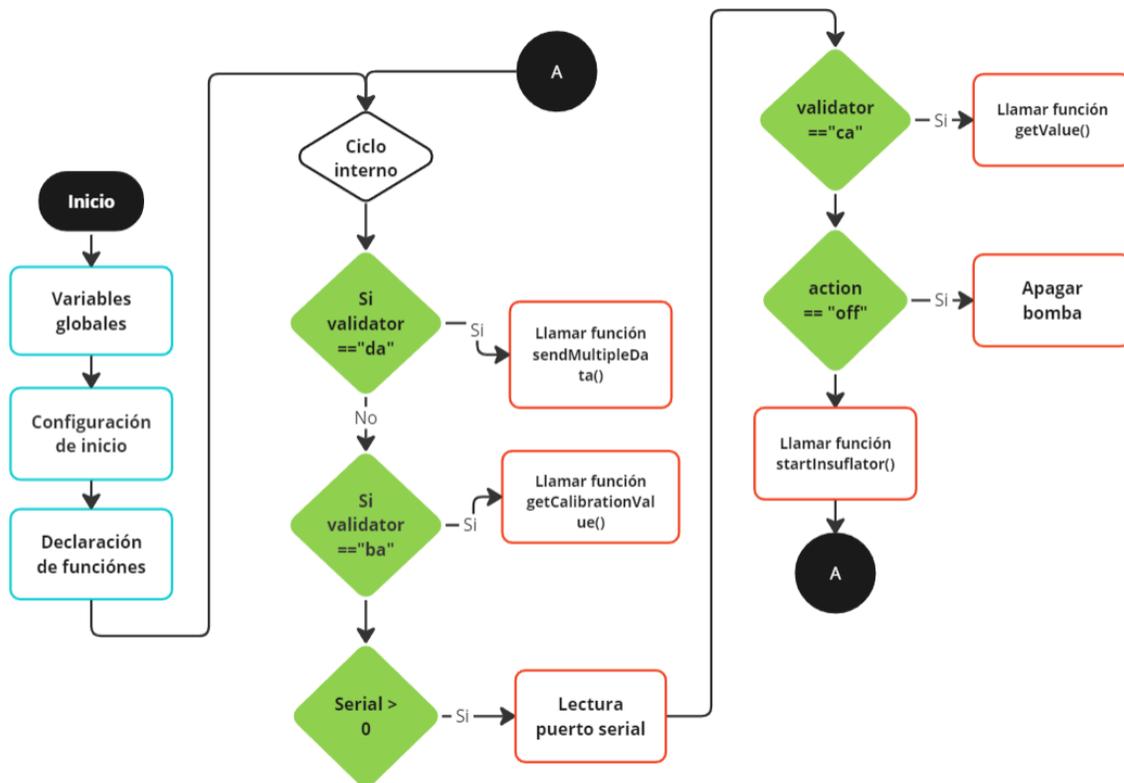


Figura 43 Diagrama de flujo correspondiente al código aplicado al microcontrolador.

Asimismo, en la primera sección encontraremos la declaración de las variables globales utilizadas en el control y almacenamiento temporal de información. La sección 2 inicializa el puerto serial y el convertidor analógico digital, además establece el pin correspondiente al mosfet como una salida digital (es por este pin que utilizaremos el PWM). En la siguiente sección, encontramos los métodos de control tanto para la adquisición de datos, control de la bomba y envío de datos. La siguiente tabla (Tabla 13) muestra la descripción de cada uno de los métodos.

Método	Descripción
getValue	El presente método adquiere el valor de la presión enviada a través del puerto serie en el código "payload", así como la acción a ejecutar, por un lado, se almacena el valor en tipo entero en la variable "sPressure", mientras que la acción a ejecutar se almacena en la variable "action".
startInsuflator	Es el método que realiza la transformación matemática de la presión seleccionada hacia un pwm determinado, una vez realizada la conversión se escribe en el pin designado para el mosfet como un analogWrite.
sendMultipleData	<p>El presente método realiza el envío de los datos tanto de fuerza como presión leídos por el convertidor analógico digital, a través de la siguiente estructura:</p> <p>“[[“+fuerza+”,” +presión.</p> <p>Es decir, la fuerza y la presión son separados por una coma, siendo también los caracteres “[[”, el inicio de cada línea de datos.</p> <p>Cabe mencionar que, los datos adquiridos se mandan en crudo para posteriormente dejar la interpretación de los mismos a la computadora.</p>

Tabla 13 Tercera sección código hacia Arduino.

La última sección es la correspondiente al “loop” o ciclo, dónde se busca en cada iteración la instrucción a ejecutar. En caso de encontrar las iniciales “ca” en el puerto serie, se procede a ejecutar la lectura del valor de presión con el cual activar la bomba. Una vez realizada la lectura se busca la segunda acción a ejecutar, dependiendo si es encender o apagar la bomba leyendo el valor de la variable “action”.

Si la acción es “off” entonces se procede a colocar en “LOW”, el valor del pin del mosfet, en caso contrario, se enciende el insuflador mediante el método “startInsuflador” y así permanecerá hasta que el valor de la variable “action” cambie.

Si al realizar la lectura del puerto serie encontramos las iniciales “da”, se procede a realizar el envío de datos a través del puerto serie.

3.7.2. Desarrollo de la interfaz gráfica (python)

El diseño del control proporcionado a la interfaz, así como el respectivo procesamiento de datos, gráficos y conexión serial fueron generados a través del lenguaje Python, gracias a su accesibilidad para el procesamiento de datos mediante los “dataframes” y su estructura orientada a objetos, la cual se ajusta a la interacción con el código generado por la interfaz de diseño de la aplicación.

El código se estructuró en diferentes sub-módulos, encargados de tareas específicas para poder evitar la redundancia de código.

El primer módulo es el código principal de la aplicación por ende lo llamamos “main.py”, y es dónde se unen las diferentes clases las cuáles al mismo tiempo representan las distintas vistas de la aplicación.

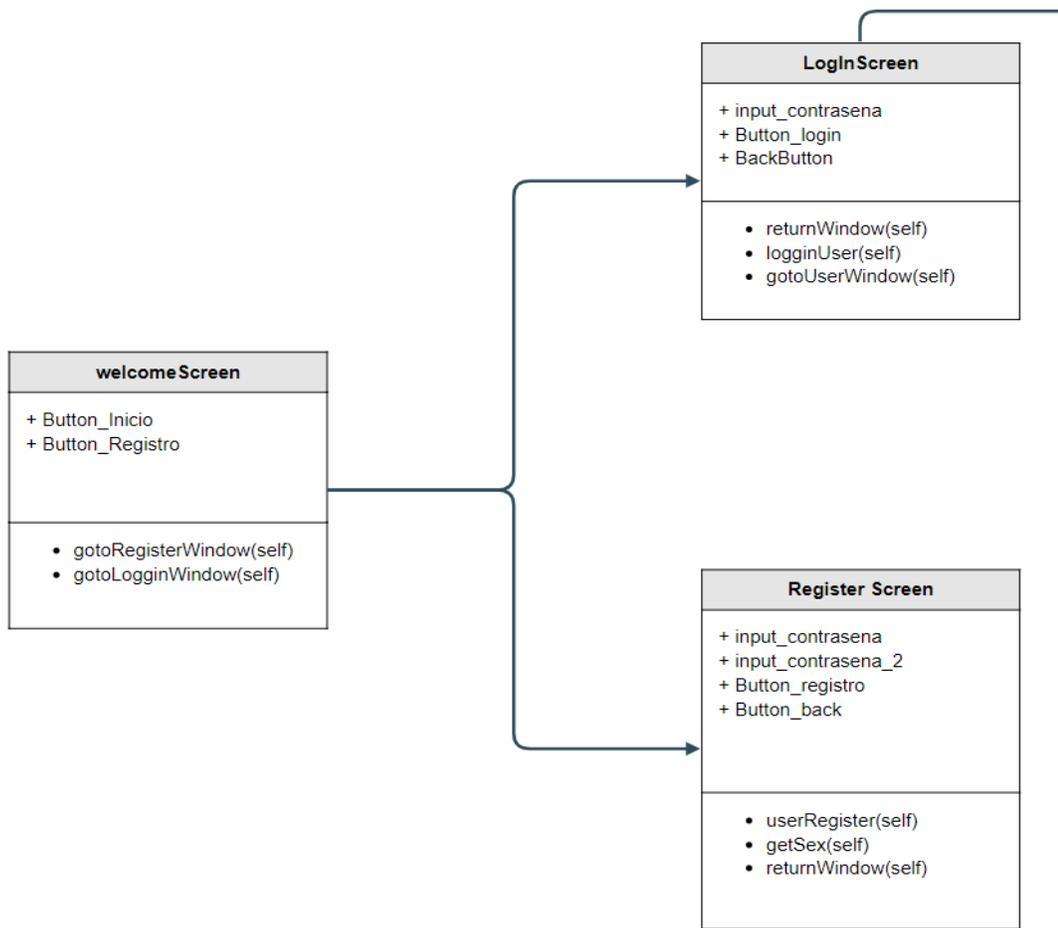


Figura 44 Diagrama de clases del archivo "main.py" primera sección

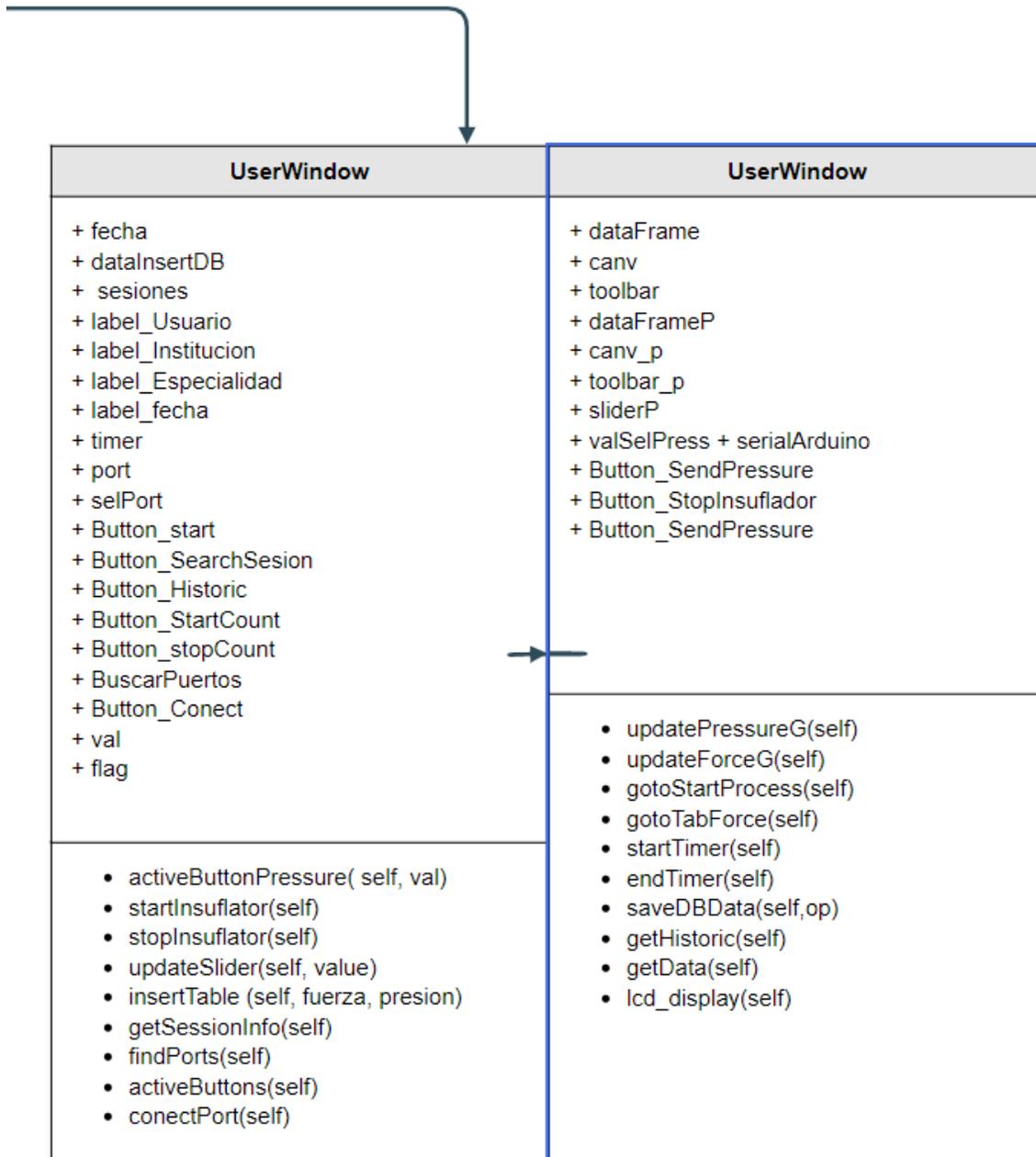


Figura 45 Diagrama de clases del archivo "main.py" segunda sección

Como podemos observar en el anterior diagrama (Figura 43), la primera clase ("WelcomeScreen") contiene las acciones necesarias para comenzar el flujo ya sea que el usuario esté registrado mandándonos hacia la vista de inicio de sesión "LogInScreen, o para registrar un nuevo usuario al "RegisterScreen". Dichas acciones, son efectuadas agregando y quitando "widgets", ya que, dentro del

código para la creación de la interfaz gráfica, cada elemento es conocido como widget, los cuales, a través de sus propiedades particulares se pueden acceder, mostrar, destruir y ocultar, lo que genera el efecto de cambio de vista, es decir los widgets son los objetos generados por cada una de las clases padre de la interfaz de diseño.

LoginScreen: Esta clase (Figura 44) contiene un pequeño formulario dónde el usuario ingresa sus credenciales, es decir usuario y contraseña y, una vez ingresadas puede acceder a la vista principal o de entrenamiento.

Esta vista cuenta con botones para regresar a la vista de inicio, y validación de la contraseña. Cabe mencionar que dicha información fue previamente almacenada en una pequeña base de datos generada mediante sqlite.

RegisterScreen: Como su nombre lo indica la presente clase/interfaz (Figura 44), es un formulario el cual almacena toda la información personal del usuario, nombre, apellido, edad, sexo, dirección, institución, especialidad, e-mail y contraseña. Una vez hecho el registro, la clase cuenta con atributos de validación de la contraseña para evitar registrar la misma de forma errónea. Asimismo, encontramos el botón de regresar a la pestaña principal, y el botón correspondiente a realizar el registro de usuario, quien realiza la ejecución del método “userRegister”.

Este método es el encargado de realizar la inserción de datos hacia la entidad de “usuario” de la base de datos, que será la encargada de realizar el registro de todos los usuarios que utilicen la plataforma.

UserWindow: Por último, tendremos la clase/interfaz UserWindow (Figura 45), dónde encontraremos todo el panel de control para cada una de las operaciones necesarias en el proceso de entrenamiento.

Para tener una mejor descripción de la clase en la siguiente tabla de contenidos (Tabla 14) encontraremos una mejor organización de cada una de las pestañas contenidas en la presente clase.

Pestaña	Descripción
Inicio	<p>Se encuentra una breve descripción del usuario loggeado, información que se adquiere a través de una consulta SQL. Una vez hecha dicha consulta, se concatenan los datos hacia los diferentes elementos tipo "label", además se guarda la información en la clase usuario del archivo "user.py" (Figura 46). Así mismo, el botón nuevo proceso realiza un cambio de pestaña hacia la pestaña proceso.</p>
Proceso	<p>En la presente pestaña se encuentran todas las operaciones de control y adquisición de datos, es aquí donde encontraremos la búsqueda del puerto serial para poder realizar la conexión entre el computador y el microcontrolador mediante el botón "Buscar puertos" que usa el archivo auxiliar "FindPorts.py" (Figura 47), un elemento que contiene llamadas a la librería serial para poder leer los puertos serial disponibles y realizar la apertura de la comunicación serial.</p> <p>Por su parte el elemento "slider", y los botones "Send", "Stop", realizan el control propio de nuestro insuflador (Bomba). A través de la apertura del puerto serial, se realiza el envío de datos hacia el microcontrolador, utilizando propiamente el archivo auxiliar "ArduinoComunication.py" (Figura 48). Donde encontraremos el método "sendData", el cual utiliza la propiedad "write" para poder escribir la información hacia el monitor serial, y así comenzar la operación de la bomba (utilizando además la estructura <i>payload</i> con el número seleccionado del "slider").</p> <p>En cuanto al "slider" o barra de control deslizante, el número mostrado es actualizado conforme cambia de valor con el método "updateSlider", dicho método actualiza el valor visual del slider, así como su valor guardado en la variable "valSelPress".</p> <p>Por su parte los botones de "StartProcess" y "StopCount", son los encargados de realizar la adquisición de datos instruyendo al microcontrolador que envíe las lecturas de los sensores (por medio de nuestro <i>payload</i>). Por su parte, el método principal para la adquisición de las lecturas, así</p>

	<p>como su interpretación matemática es “receiveData”, el cual será un ciclo controlado cuya condición principal para estar recibiendo datos es que el atributo “flag” se mantenga en “False”.</p> <p>Además, es en este método que se realiza la conversión y adquisición de tiempo mediante la llamada al método “calcTime”, dicho método transforma los microsegundos en segundos y su respectiva fracción de tiempo.</p> <p>Por su parte una vez que la adquisición de datos ha concluido se realiza la generación de las gráficas, y mediante un cuadro de diálogo se pregunta al usuario si éste desea guardar los datos, para la generación de los cuadros de diálogo utilizamos la tecnología auxiliar de Tkinter.</p> <p>En caso de tener un valor afirmativo se guarda en la entidad “Datos” los datos correspondientes a la fuerza, y presión además de la fecha y la sesión en dónde fueron adquiridos dichos datos. Por su parte en la entidad “Usuario” se actualiza el número de sesión, así mismo en la entidad “Duración” se insertan los datos correspondientes a la duración de la presente sesión, la sesión misma y el id del usuario.</p> <p>Así mismo se adquieren los valores máximos de la sesión en curso, para luego ser mostrados en la pestaña de resultados.</p>
Fuerza	<p>La pestaña Fuerza es generada con la información almacenada en la propiedad force del objeto serialArduino. A través de numpy se genera un array que posteriormente será parte del dataframe de fuerza, el cuál une en su índice la información del tiempo transcurrido (almacenada en el atributo “dt_string”) con la información de fuerza. Ahora bien, para la generación de la figura utilizamos la clase</p>

	<p>Canvas_grafica2, y es en este objeto que añadimos la información del dataframe de fuerza.</p>
Presión	<p>La pestaña Presión es generada a través de la información almacenada en el atributo pressure del objeto serialArduino. Además, se une la información del tiempo transcurrido (almacenada en el atributo "dt_string") y se utiliza como índice del "dataFrameP". Dicho dataframe será la información de la figura generada por parte de la clase Canvas_grafica_p</p>
Resultados	<p>En la pestaña Resultados, observamos los datos de la sesión, la presión y la fuerza mostrados en una sola tabla. Por otro lado, tenemos un buscador que realiza una consulta sql para obtener la información de los datos de sesiones pasadas o mediante el botón histórico se adquiere la información de todas las sesiones del usuario.</p> <p>Además, es en esta pestaña dónde se muestra el resultado de la duración total del proceso que corresponde al dato de mayor valor almacenado en el atributo dt_string, del objeto serialArduino, lo propio se hace con la información de fuerza.</p>

Tabla 14 Tabla de contenidos de la clase userWindow.

User Diagram



Figura 46 Diagrama del archivo user.py

FindPorts Diagram



Figura 47 Diagrama del archivo FindPorts.py

Arduino Communication diagram

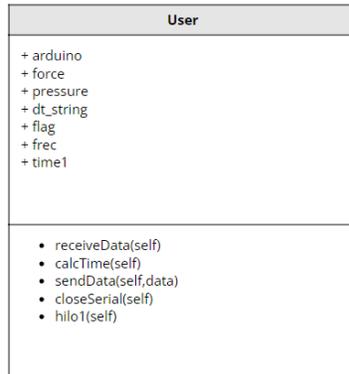


Figura 48 Diagrama del archivo ArduinoCommunication.py

3.7.3. Base de datos

Para este proyecto, se estructuró una pequeña base de datos conformada de 3 entidades principales: Datos, Usuario y Duración, la cual nos brinda toda la información necesaria de los usuarios registrados, así como sus respectivas sesiones de entrenamiento, con los datos adquiridos y la duración total de su sesión. De este modo, estructuramos la base de datos como podemos observar en la Figura 49.

Diagrama entidad-relación

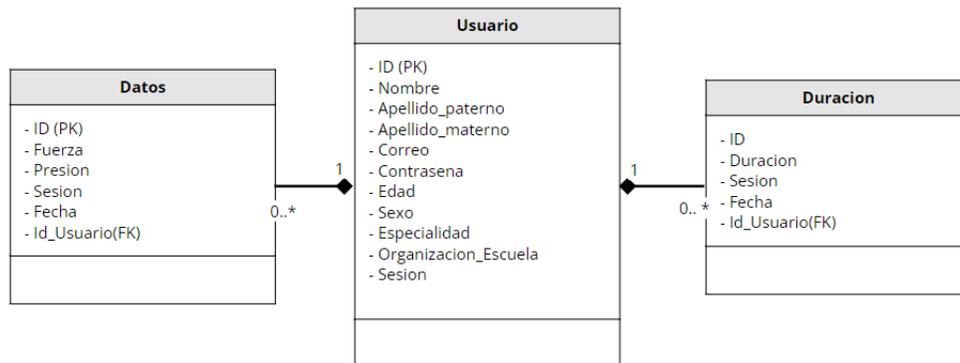


Figura 49 Diagrama entidad relación

3.8. Desarrollo de entorno gráfico

3.8.1. Diseño de interfaz gráfica

El desarrollo de la interfaz de entrenamiento fue realizado a través del programa Qt Designer, el cual, genera un archivo .xml con todas las especificaciones de diseño de la ventana gráfica, para posteriormente ser interpretado por Python como diferentes clases. Con lo anterior y utilizando la programación orientada a objetos, podemos acceder a cada una de las propiedades del elemento desde botones hasta gráficas.

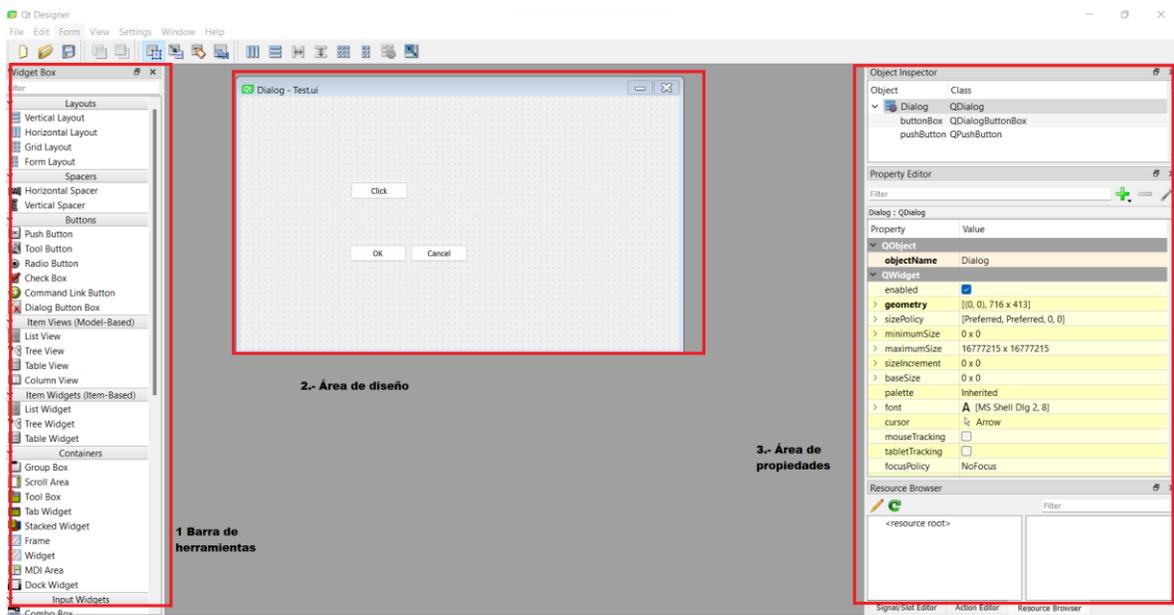


Figura 50 Entorno gráfico de Qt Designer.

Dentro del entorno (Figura 50) podemos encontrar 3 áreas principales, lo correspondiente a la barra de herramientas se cuenta con los widgets u objetos que podemos utilizar dentro de nuestro cuadro de diálogo, al centro encontramos el área de trabajo dónde podemos ir editando el diseño de cada una de nuestras vistas, mientras agregamos o eliminamos elementos sólo arrastrando el elemento.

Por último, encontramos el área de propiedades, dónde se edita el nombre de cada elemento, para tener un mejor control al movernos dentro del código a ejecutar.

3.8.2. Estructura de vistas

3.8.2.1. Menú

En la vista inicial (Figura 51) encontraremos un pequeño menú de selección para poder acceder a la plataforma o registrarnos a la misma en caso de no tener un usuario y contraseña previamente creados. Lo anterior fue creado de esta forma para poder tener la opción de dar soporte a diferentes usuarios.

La primera opción (Botón “Iniciar sesión”) nos ofrece el salto hacia la interfaz de acceso de usuario, dónde mediante la inserción de sus credenciales puede acceder a la plataforma de entrenamiento. En cuanto a la segunda opción, se despliega la opción de registro, es decir, una interfaz mediante la cual, el usuario tiene la opción de registrar sus futuras credenciales, así como datos institucionales.



Figura 51 Vista Menú.

3.8.2.2. Acceso de usuario (Log In)

Para la segunda vista, nos encontramos con el acceso al usuario, el cual permite el ingreso a la plataforma o niega el mismo mediante mensajes de validación (Figura 52). En dicha vista encontramos la interacción entre “labels” indicativos, así como cuadros de textos dónde se almacenarán los valores ingresados por el usuario.

Por otro lado, la flecha es una imagen que fue adaptada para su uso como botón de retorno mientras que el botón de “Iniciar sesión” se coloca al centro brindando una mejor experiencia de usuario.

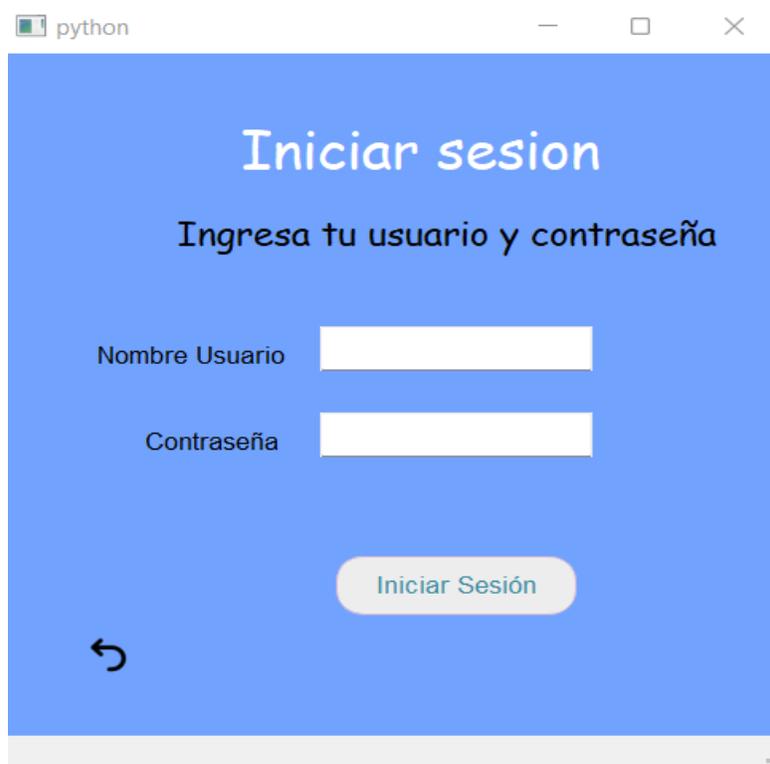


Figura 52 Segunda vista: Log In

3.8.2.3. Registro de usuario

En la opción de registro del usuario encontraremos un breve formulario con el cual se logrará obtener información particular del usuario y que, posteriormente, servirá para poder llevar un adecuado acceso, así como una plataforma multiusuario que brinde la opción de atender múltiples cuentas (Figura 53). En esta vista se utilizaron cuadros de texto, cuadros de selección tipo checkbox así como un spin box para poder seleccionar la edad adecuada.



The image shows a Qt dialog box titled "Dialog - ventanaRegistro.ui". The main heading is "Registro nuevo usuario" in blue. Below it, the instruction "Favor de llenar cada campo" is displayed. The form contains the following fields and controls:

- Nombre Usuario:
- Apellido Paterno:
- Apellido Materno:
- E-mail:
- Contraseña:
- Confirmar Contraseña:
- Especialidad:
- Organización / Escuela:
- Sexo: Masculino Femenino
- Edad:

At the bottom, there are two buttons: "Regresar" and "→ Registrar Usuario".

Figura 53 Tercera vista: Registro

3.8.2.4. Plataforma

El último elemento de la aplicación es la interfaz de entrenamiento, la cual consta de una pestaña de bienvenida que ofrece un pequeño resumen del usuario, así como la última sesión de entrenamiento registrada (Figura 54). En seguida a la pestaña de Inicio encontraremos las siguientes pestañas y su breve descripción de acuerdo con la siguiente tabla:

Pestaña	Descripción
Proceso	Dentro de la pestaña Proceso encontraremos los botones de control para comenzar con el proceso de entrenamiento. En primer lugar, encontramos la conexión entre la plataforma y la computadora a través del puerto serial (Figura 55). Para ello requerimos buscarlo y realizar la conexión, una vez hecha la conexión activamos nuestra bomba (Insuflador), y pasamos a la adquisición de datos. Una vez que se comenzó el registro de datos procedemos a realizar el entrenamiento manual, es decir, el ingreso del trocar en nuestra plataforma de entrenamiento dónde al finalizar, detendremos el proceso con el botón de control correspondiente.
Fuerza	En la pestaña de Fuerza y Presión encontraremos una gráfica con el registro de tiempo sobre el comportamiento del fenómeno, además encontraremos una barra de herramientas para realizar un mejor análisis, tomar capturas de pantalla, agrandar la gráfica o disminuirla entre otras características (Figura 56).
Presión	
Resultados	Por último, en la pestaña de Resultados encontraremos la información detallada de nuestro proceso, es decir la presión y fuerza registrados en determinados instantes de tiempo, así como un buscador de registros por sesión y de todas las sesiones pasadas, un resumen de la duración total del proceso y así como la fuerza máxima registrada en el análisis de sesión correspondiente (Figura 57).

Tabla 15 Descripción de pestañas de la vista plataforma



Figura 54 Vista Plataforma

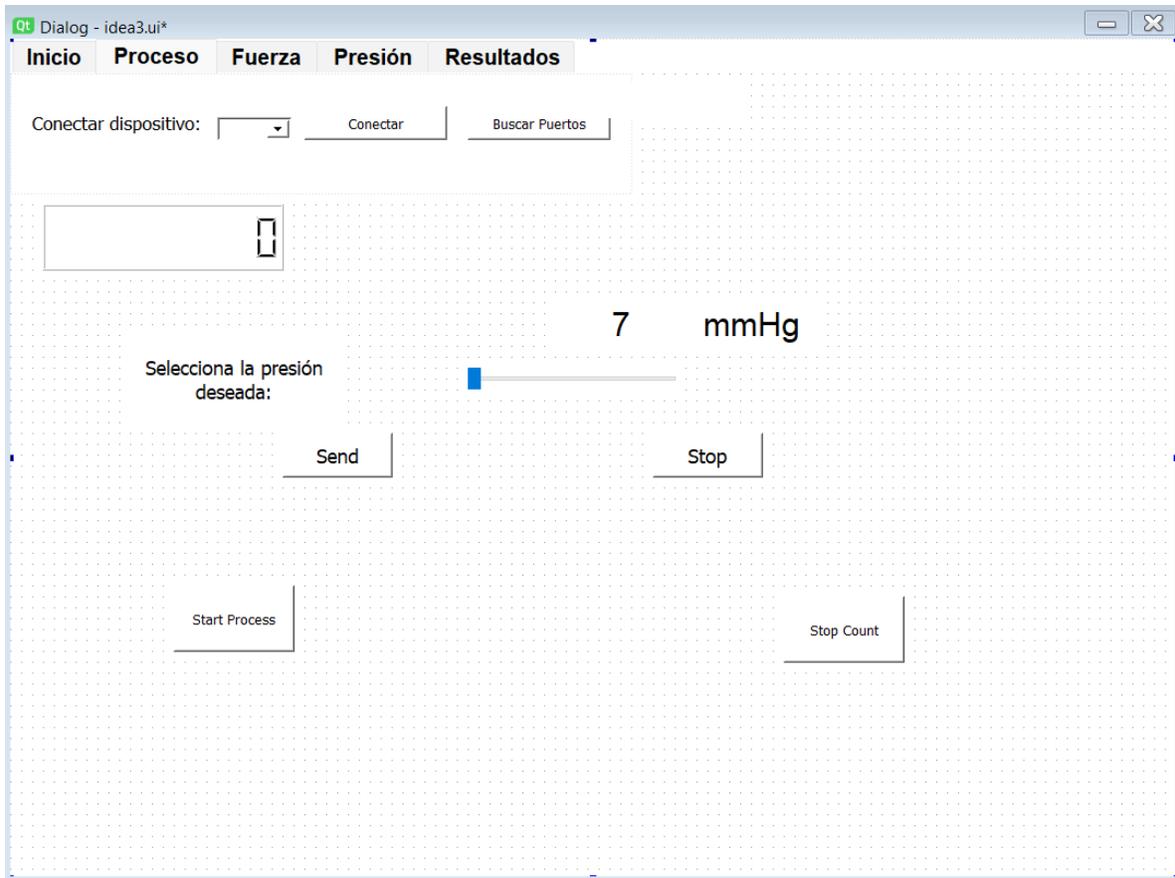


Figura 55 Pestaña Proceso

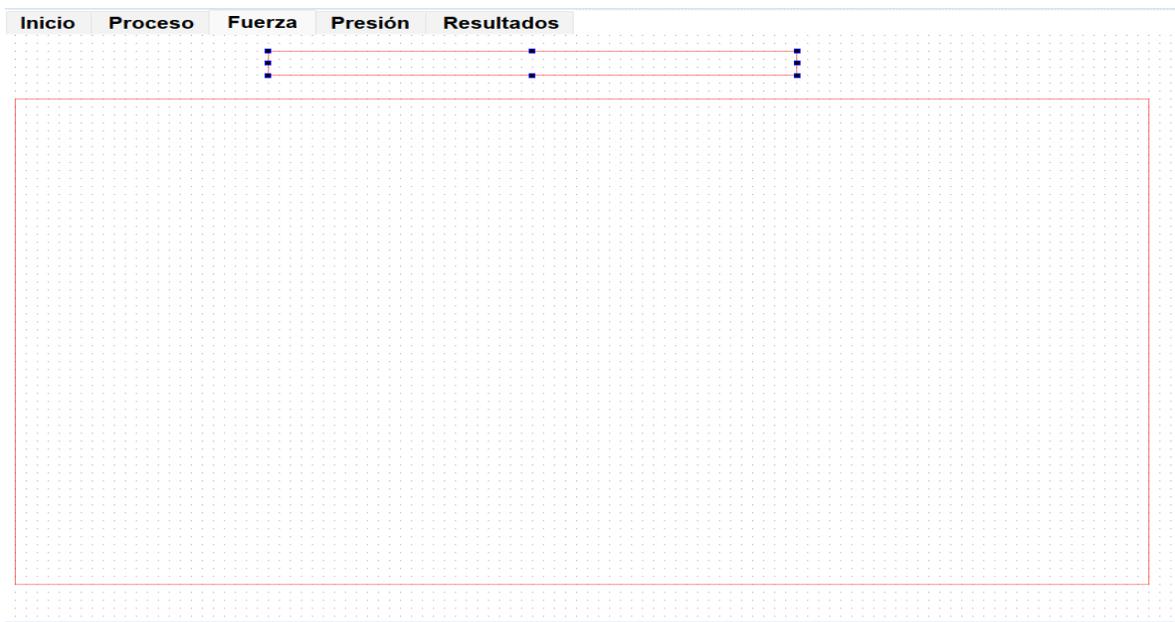


Figura 56 Pestaña fuerza y presión



Figura 57 Pestaña Resultados

3.9. Diseño de la interfaz de entrenamiento

3.9.1. Diseño Mecánico de interfaz externa

El dispositivo de entrenamiento externo con el cual se tiene una frecuente interacción realiza la simulación de una proporción de tejido por donde se realizará la inserción del trocar.

El presente diseño toma su forma a partir de 3 necesidades básicas a cumplir para crear la interfaz de entrenamiento, los cuales son enumerados de la siguiente manera:

1. Cavidad que permita la inserción del trocar, y proporcione la sensación de punción.
2. Permite el ingreso de aire por lo cual requiere un cierre semi hermético, además se requiere medir la presión interna.
3. Permite la operación e inserción del dispositivo a través de los diferentes ángulos de operación.

Para lograr cumplir con las especificaciones de diseño, se planteó una estructura cilíndrica con la finalidad de poder tener la libertad para manipular la inserción de trocar bajo diferentes ángulos (Figura 58), y a su vez se adicionaron 2 cierres laterales (abrazaderas), ver Figura 59, cuya función radica principalmente en ser el cierre semi-hermético que lograra mantener el gas o aire al interior del dispositivo.

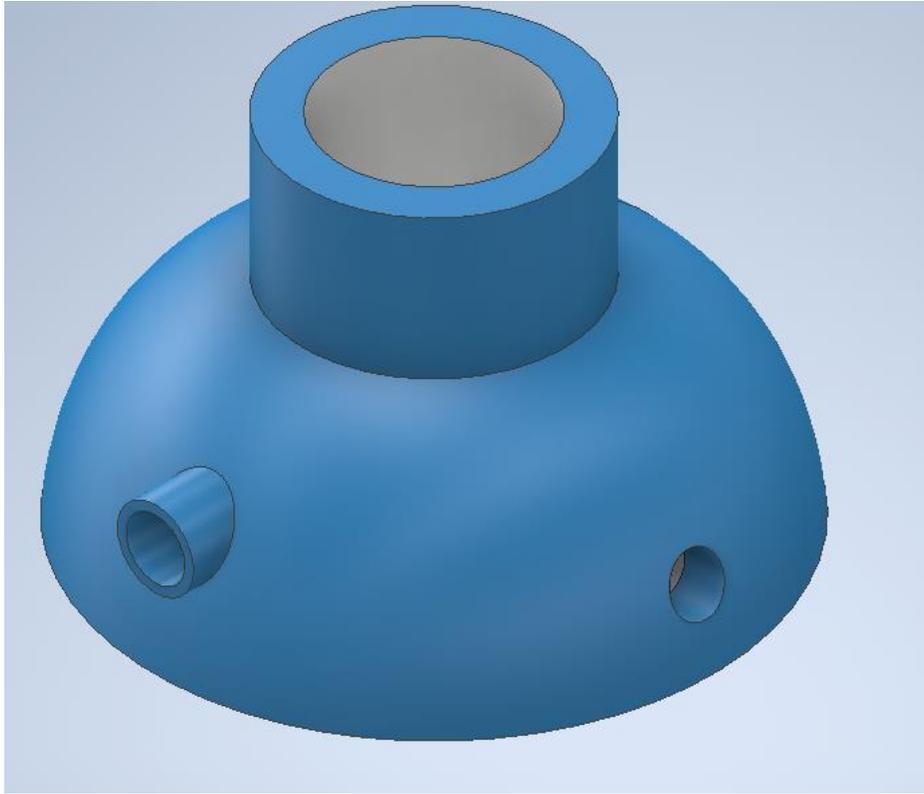


Figura 58 *Diseño de interfaz de entrenamiento.*

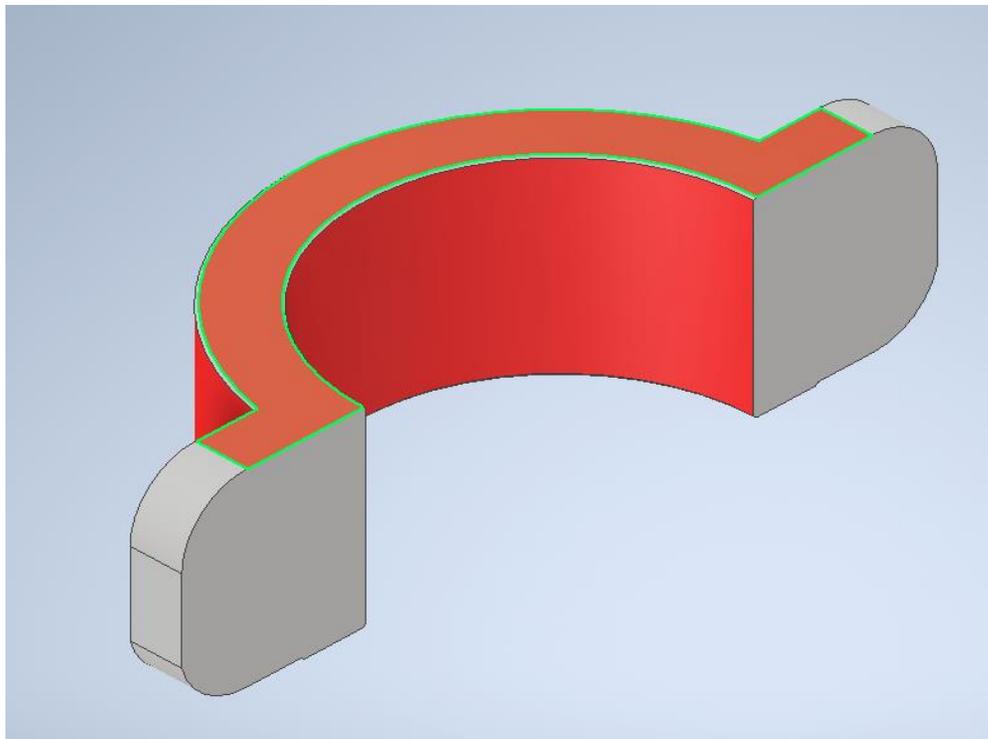


Figura 59 *Abrazadera cierre semi-hermético.*

En el diseño, se dejaron 2 orificios con diferente diámetro y diseño (Figura 60). El primero con un diámetro de 17 mm **(1)**, el cual se implementó para lograr inyectar el aire proveniente de la bomba y el segundo con un diámetro de 15 mm **(2)**, el cual se añadió para ser la salida hacia la entrada de nuestro sensor de presión.

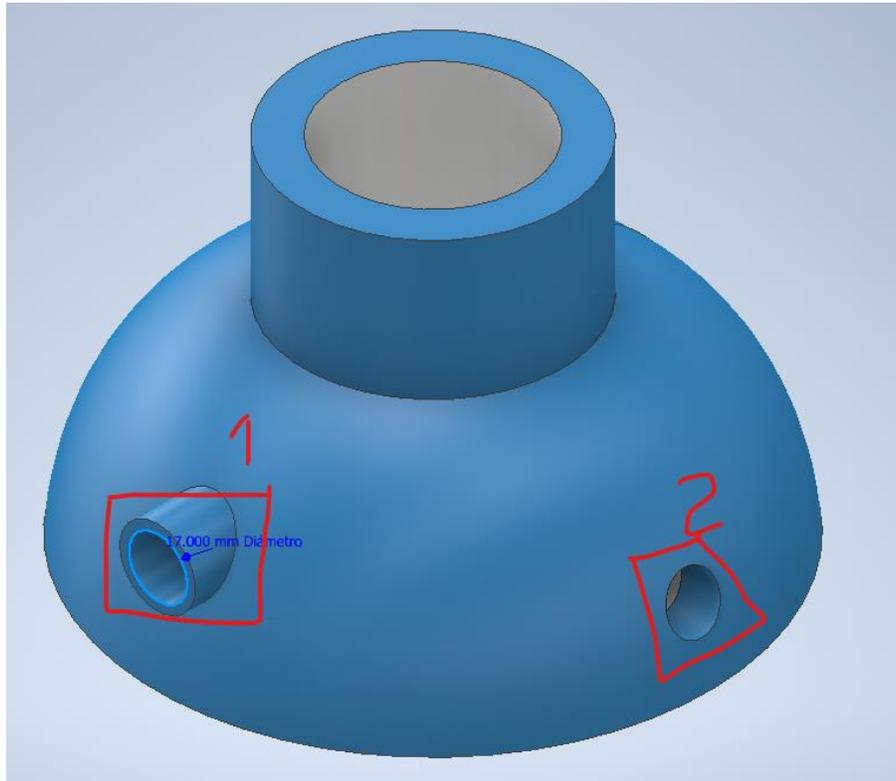


Figura 60 orificios de entrada/salida del flujo de aire, 1) Muestra la entrada de aire proveniente de la bomba y 2) orificio de salida hacia el sensor de presión.

Además, para poder realizar una adecuada sujeción, se añadió 1 par de tornillos con sus respectivas tuercas de seguridad y tuercas de tipo mariposa para lograr mantener fijo el material a utilizarse para la simulación de la piel.

Una vez realizados los diseños, se procedió a imprimirlos en tecnología de manufactura en 3D para posteriormente añadir las mangueras de flujo de aire y la salida hacia el respectivo sensor de presión.

3.9.2. Adaptación pad de entrenamiento

Por otro lado, se evaluaron diferentes materiales que pudieran ser candidatos a nuestra simulación de piel; sin embargo, la cuestión del grosor, espesor y flexibilidad fueron determinantes para poder decidir un tipo de pad de entrenamiento adecuado para nuestro modelo un tanto desechable dada la naturaleza del fenómeno en estudio que es una punción lo que implica romper.

Uno de los candidatos fue el látex, puesto que contiene una gran flexibilidad. Sin embargo, al intentar fabricar algunos para nuestro uso se evaluó que cada operación generaría un costo superior al estimado y un proceso tardado en la creación del propio pad.

Por lo tanto, se prefirió utilizar pads hechos de fomi (Figura 61), los cuales son flexibles y proporcionan una cierta oposición al trocar al momento de su ingreso lo que sucedería exactamente en un proceso real. Además, su fácil adquisición y creación de los pads generan, una opción viable para utilizar en el proceso de entrenamiento.

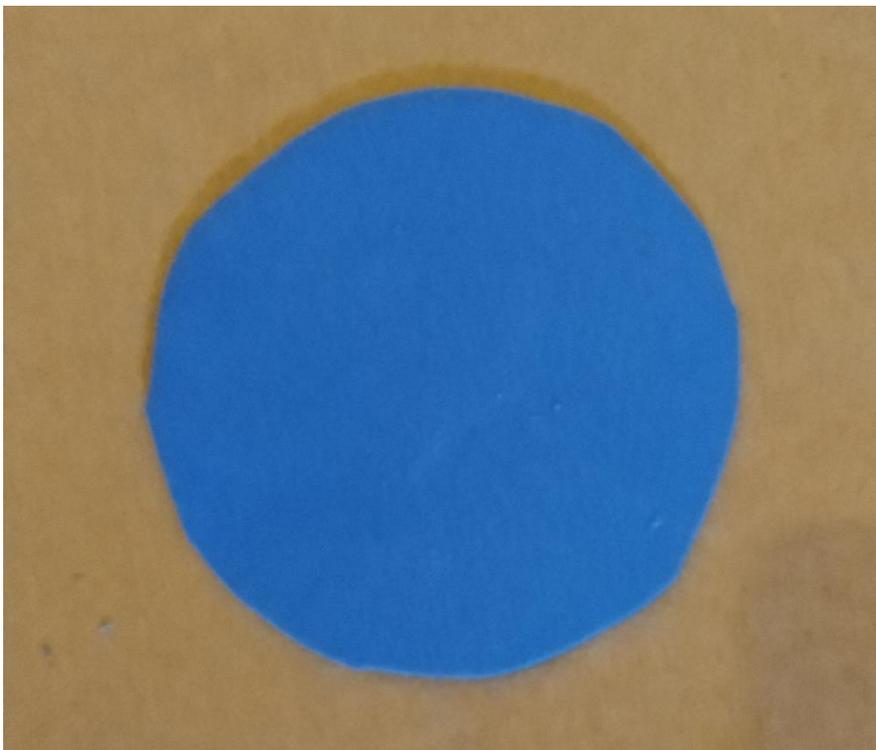


Figura 61 Pad de entrenamiento

Capítulo 4 Resultados

Posterior a la integración de cada uno de los módulos, se realizaron pruebas para poder validar el correcto funcionamiento de cada elemento del entrenador, esto con la finalidad de garantizar que todo el desarrollo realizado en el capítulo anterior tenga los resultados que se esperaría tener. Por lo cual, dividiremos en 2 partes lo concerniente al dispositivo físico y todos los elementos de carácter virtual.

El presente trabajo de tesis fue desarrollado durante la pandemia debida al SARS-CoV-2 virus, lo que limitó y retrasó el desarrollo de esta, ya que la comunicación con proveedores fue limitada. Dadas las recomendaciones sanitarias, la adquisición de materiales para pruebas, herramientas, circuitos integrados, además el material electrónico fue limitado y generó gastos mayores a los estimados al inicio del proyecto debido al constante uso de servicios de paquetería hasta fechas posteriores a la implementación de la vacuna. Sin lugar a duda, el desarrollo del presente trabajo bajo las condiciones anteriores provocó una dificultad superior al estimado en condiciones normales.

4.1. Dispositivo final

4.1.1. Dispositivo de entrenamiento

El dispositivo final de entrenamiento en su exterior consta del ensamble de ambos sujetadores unidos a través de 2 tornillos con tuercas de seguridad y mariposa para poder ajustar el 'pad' de entrenamiento cuyo diámetro final fue de 9.5 – 10 cm, la manguera negra que es la encargada de conducir el aire interior hacia el sensor de presión (Figura 62). Por otra parte, el orificio central que podemos apreciar en la Figura 62 es la entrada proveniente de la manguera conectada a la bomba (Insuflador) Figura 63.



Figura 62 Dispositivo de entrenamiento



Figura 63 Insuflador

Para la adquisición de datos, así como el control de la bomba, se adaptó una base en la cual se colocó la tarjeta de adquisición, fuente de poder y sensor de presión para encapsular los dispositivos lógicos y eléctricos como se puede apreciar en la Figura 64. Para poder accionar la bomba, se encuentra un switch que permite la alimentación de la bomba. La fuente se conecta directamente a la corriente eléctrica, y nos sirve como fuente de alimentación de los sensores y la bomba.

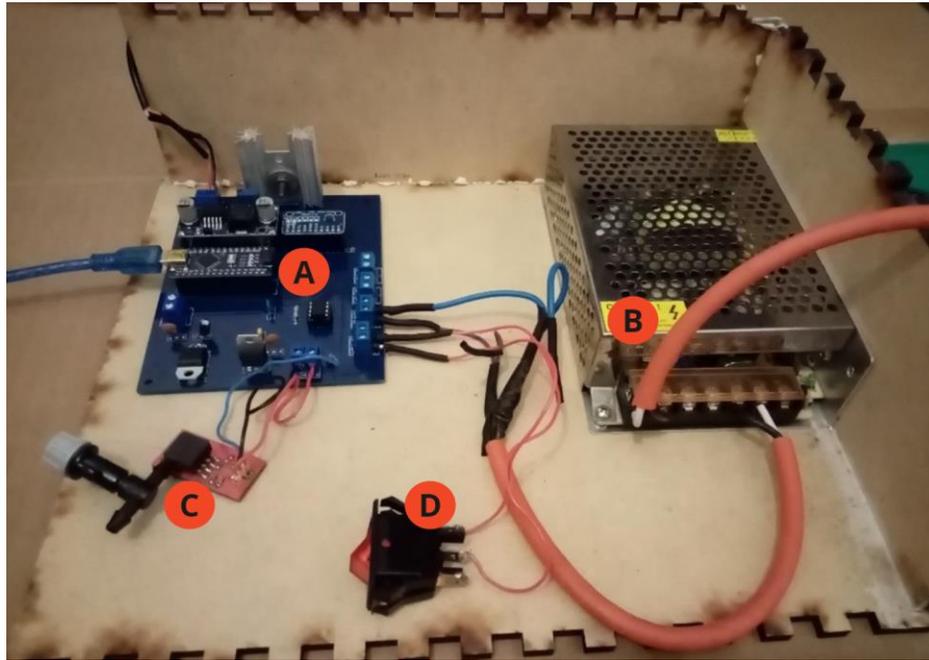


Figura 64 *A Tarjeta de adquisición de datos, B Fuente de alimentación y C Sensor de presión, D Switch de control.*

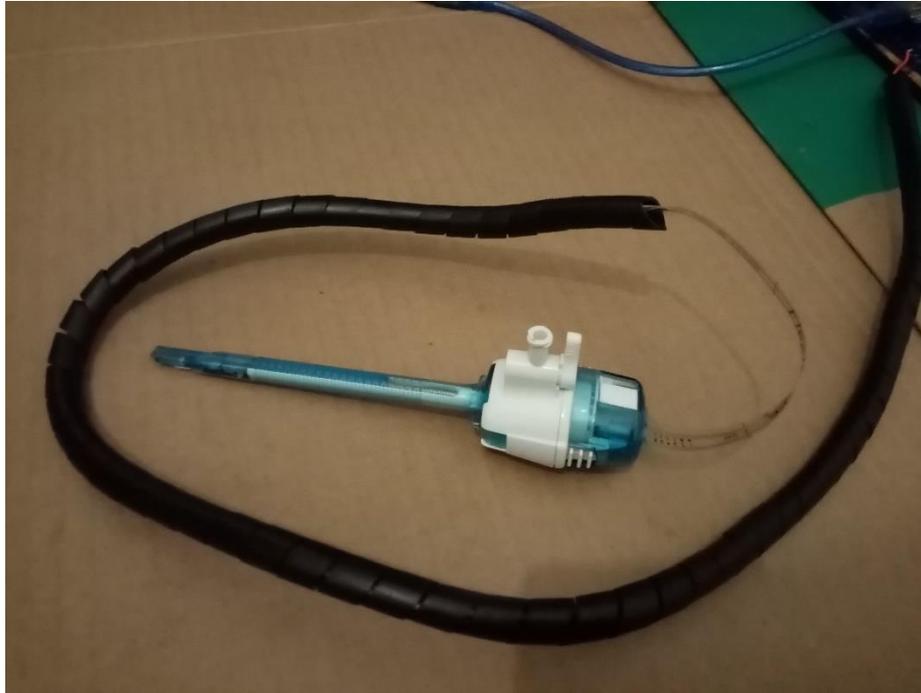


Figura 65 Trocar y sensor de fuerza

En cuanto al dispositivo principal de estudio, el trocar, se añadió un adhesivo en la parte superior del mismo para poder colocar el sensor de fuerza de manera adecuada y que este logre ser manipulado en la posición correcta. Por otra parte, se añadió un protector a los cables que enviarán la señal hacia la tarjeta de adquisición de datos, con la finalidad de permitir la maniobrabilidad del instrumento con la mayor ergonomía posible (Figura 65).

4.1.2. Plataforma de entrenamiento.

En la plataforma de entrenamiento las vistas descritas en el capítulo anterior constituyen la entrega y diseño final, siendo la Figura 66, el resultado final correspondiente a la lectura del sensor de presión al momento de seleccionar un valor específico.

En dicha gráfica logramos apreciar el cambio y aumento de presión, así como las diversas oscilaciones debido a la inestabilidad del aire al estar en un cuerpo semi hermético, pero que, en la gran mayoría del tiempo tiende a mantener el mismo comportamiento, lo que necesitamos en el momento de realizar una cirugía.

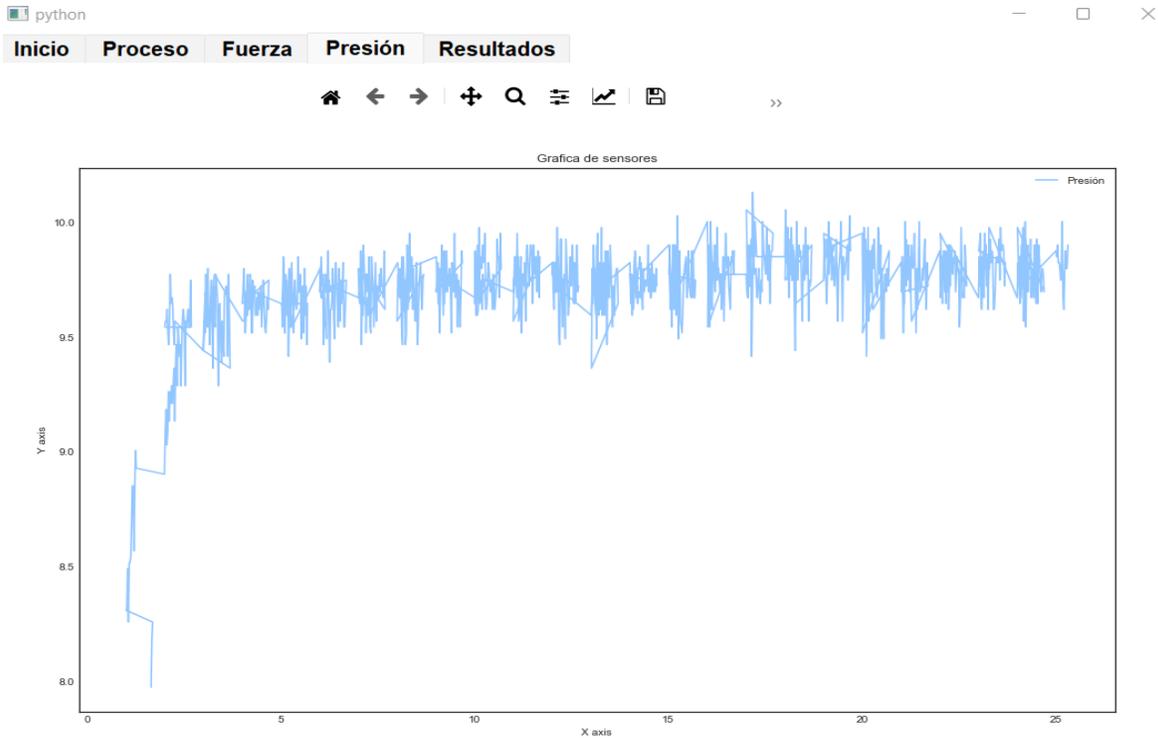


Figura 66 Gráfica final de presión

Por otro lado, la Figura 67 refleja el comportamiento exacto de la punción hecha por un trocar, y es aquí dónde podemos obtener la fuerza que emplea un usuario al momento de realizar la inserción del dispositivo, como podemos apreciar nuestro rango de lectura comprende desde los 0 N hasta los 23.4 N. Y una frecuencia de muestreo promedio de 33 datos/segundo.

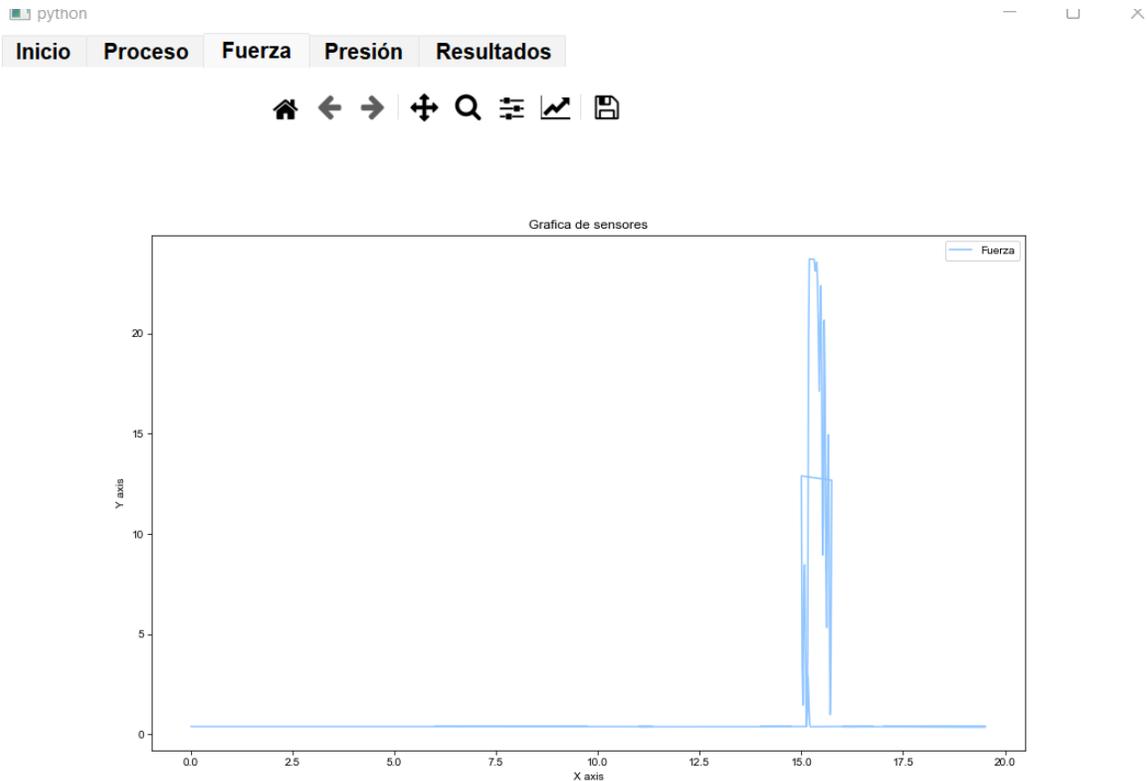


Figura 67 Gráfica final de fuerza

4.2. Instrucciones de uso y configuración de la plataforma de entrenamiento

Para poder realizar el uso de nuestra plataforma, se encuentran una serie de pasos que debemos verificar constantemente para poder tener un buen uso de esta y así evitar la pérdida de información sensible.

En primera instancia, debemos verificar que el botón de alimentación general se encuentre inactivo. Una vez hecho esto procedemos a conectar la fuente de alimentación a la conexión ordinaria, y verificamos que el botón de encendido del insuflador se encuentre activo esperando la señal proveniente del microcontrolador.

Posteriormente, realizamos la conexión del cable USB entre el Arduino y la computadora. Este paso es de suma importancia, consiste en verificar que la terminal USB se encuentre conectada adecuadamente, debido a que, en caso contrario se podrían producir fallas al momento de la adquisición de datos.

Una vez hecho lo anterior, procedemos con el inicio de la aplicación de la plataforma, se accede mediante nuestro usuario registrado. En este momento debemos tener cuidado de que ningún elemento se desconecte puesto que podríamos causar una falla y errores durante la adquisición de los datos.

Finalmente, realizamos la apertura del botón general de alimentación para poder realizar la sesión de entrenamiento, una vez verificadas las conexiones.

Adicional a los procesos anteriores, se debe tomar en cuenta el factor temperatura, ya que, al ser un gas, la lectura de datos puede verse afectada debido a cambios en la temperatura ambiental. Para prevenir errores al momento de realizar la adquisición se recomienda tener en cuenta un ajuste previo a la operación final de nuestra plataforma.

Dicho ajuste se realiza mediante el botón “add” en la sección calibración de la pestaña proceso (Figura 68). Durante este ajuste se debe realizar una conexión con el puerto serial exitosa y, una vez hecho esto, realizar una prueba del funcionamiento del insuflador enviando una presión entre 7 y 9 mmHg, para no sobrecalentar el dispositivo.

Al enviar los datos esperamos alrededor de 10 segundos, detemos el proceso y observamos en la gráfica de presión si la presión solicitada corresponde a lo representado en la gráfica en caso de que los valores arrojados por la gráfica sean inferiores al valor solicitado debemos calibrar manualmente el dispositivo añadiendo valores a la sección de calibración (Figura 68)

Después de añadir un valor a la sección de calibración, se debe probar nuevamente el proceso y continuar de esta forma hasta tener la presión deseada oscilando entre ± 0.4 . Cuando finalmente se llegue al ajuste final, podremos utilizar el dispositivo de forma normal.

Calibración



Figura 68 Sección de calibración

4.3. Generación de pruebas

Prueba	Descripción	Resultado
Encendido / apagado del insuflador (Plataforma).	En la prueba de encendido apagado del insuflador se procedió a realizar el inicio de sesión de usuario, establecer una conexión serial a través de nuestro buscador de puertos serial, y enviar la presión seleccionada a través del botón: 'Send' y posteriormente detener el proceso con el botón stop.	Una vez que se ha establecido una comunicación serial exitosa existe un breve retardo de aproximadamente 1 segundo hasta comenzar su operatividad, lo propio sucede al detener el proceso mediante el botón de "stop".
Reguladores de voltaje	Para la presente prueba se requiere que, previo al montaje de cada elemento al interior de la placa PCB, se pruebe en una tarjeta de pruebas (protoboard), con la misma configuración que se utilizará dentro de la PCB. Todos los elementos deben funcionar correctamente antes de su montaje final.	Para los reguladores se realizó el montaje de las conexiones necesarias en una tarjeta de pruebas, utilizando la fuente de poder como su fuente de alimentación, así que cada uno de los reguladores tuvo un espacio propio para poder verificar su correcto funcionamiento, midiendo a través de un multímetro la salida final de cada regulador.

Verificación de los sensores	<p>Para validar los sensores y su correcto funcionamiento se requiere observar la salida analógica de los mismos a través de un multímetro evitando en todo momento que alguno de los elementos involucrados en su alimentación presente algún daño o falla.</p>	<p>En lo concerniente al amplificador operacional se utilizaron las salidas de los reguladores para poder alimentar el amplificador operacional. A dicho elemento se le conectó el sensor de fuerza para poder observar la salida del amplificador una vez que se accionaba el sensor. Una vez hecho lo anterior se verificó en el multímetro un rango de valores al utilizar diferentes estímulos de fuerza.</p> <p>En cuanto al sensor de presión a través del multímetro y diferentes estímulos de presión se observó su correcto funcionamiento.</p>
Conexión general y cableado	<p>La prueba consiste en realizar una conexión general desde la fuente de alimentación y verificar el correcto funcionamiento de cada elemento una vez montados en la PCB. Se debe tomar en cuenta el correcto funcionamiento del botón de paro de emergencia además del Arduino. Para verificar lo anterior debemos utilizar la plataforma para poder realizar una conexión serial, enviar y recibir datos.</p>	<p>En esta prueba se conectaron todos los elementos, involucrados, desde la bomba, hasta nuestro microcontrolador. Y se observó un funcionamiento correcto al recibir información de los sensores a diferentes presiones, con lo cual también se logró verificar que la bomba es capaz de operar a diferentes presiones dependiendo de la opción elegida.</p>

Registro de usuario	Para realizar la prueba de registro de usuario se debe acceder a la segunda vista proporcionada por la plataforma en la cual debemos registrar un usuario utilizando validaciones sencillas como las correspondientes a los campos de entrada de datos para posteriormente realizar el inicio de sesión una vez hecho el registro.	Durante la presente prueba se intentó registrar un usuario con contraseñas diferentes por lo que la plataforma no permitió el registro, además de intentar registrar un usuario menor a 18 años, lo que la propia plataforma tampoco permite. Una vez llenados correctamente todos los campos, se realiza el registro y el acceso a la sesión de entrenamiento es correcto.
Inicio de sesión	La presente prueba consiste en realizar el acceso a la segunda pestaña que es la vista de entrenamiento, utilizando nuestro usuario y contraseña la plataforma debe permitirnos acceder a la vista de entrenamiento, o negarnos el acceso en caso de no estar registrado o ingresar las credenciales incorrectas.	Durante esta prueba se intentó acceder a la plataforma sin ningún éxito cuándo se ingresaba usuario y contraseña incorrecto, sin embargo, una vez ingresadas las contraseñas correctas la plataforma nos permite realizar el cambio de vista hacia la correspondiente al entrenamiento, otorgándonos la información de la última sesión, fecha actual, así como la información del usuario loggeado.
Apertura del puerto serial	En la presente prueba, a través de la vista de entrenamiento se debe establecer una conexión serial exitosa con nuestra tarjeta de adquisición de datos.	Una vez encontrado el puerto serial (Figura 69), donde se encuentra conectado nuestro dispositivo, se selecciona dicho puerto y al existir una conexión exitosa nuestro sistema lo notifica al usuario (Figura 70), en caso contrario se notifica que no

		se pudo realizar una conexión con el dispositivo.
Generación de gráficas.	Para la presente prueba se tiene que realizar una correcta conexión serial, y, una vez hecho esto podemos adquirir datos mediante el botón de "start process", una vez transcurrido un cierto tiempo finalizar el proceso y observar que las gráficas han sido generadas adecuadamente.	<p>En la presente prueba después de una conexión serial exitosa, se procedió a presionar el botón de adquisición de datos, colocando un error, por lo cual se recomienda dejar un breve lapso entre 1 y 3 segundos antes de utilizar el botón de adquisición, y así evitar el error de sincronización.</p> <p>Una vez hecho el entrenamiento y detener de forma general todo el sistema, aparece un mensaje si es que los datos adquiridos en la presente sesión desean guardarse en la base de datos.</p> <p>En caso afirmativo o negativo se produce un salto a la pestaña de fuerza mostrando adecuadamente el comportamiento de la fuerza en una gráfica, así como lo propio en la pestaña de presión.</p>
Adquisición de datos.	Prueba en el monitor serial, y posteriormente al recibir datos en el ordenador.	En la presente prueba se verificó que nuestro convertidor analógico digital funcionara correctamente al observar diferentes lecturas tanto en el monitor serial de Arduino como en la plataforma de

		entrenamiento. Dichas lecturas fueron posteriormente interpretadas por nuestros modelos matemáticos.
Lectura de valores del sensor de presión.	Lectura correcta del convertidor adc externo en su pin de entrada correspondiente a la presión.	<p>En esta prueba verificamos con el manómetro digital la correspondencia entre la lectura de nuestro sensor y la lectura del manómetro.</p> <p>Una vez hecho lo anterior, a través de la gráfica de presión logramos apreciar el comportamiento del aumento o disminución del gas al interior del dispositivo de entrenamiento.</p> <p>Para un mayor nivel de detalle se verificó en la tabla de resultados cada uno de los valores mostrados en la gráfica.</p>
Lectura de valores del sensor de fuerza.	Para la lectura de los valores de fuerzas tenemos 2 pruebas la primera de forma analógica antes de enviar la salida del amplificador operacional hacia el convertidor adc, y la segunda en la plataforma que los datos sean adquiridos correctamente y logren mostrarse.	<p>Con respecto a la prueba en su fase analógica, el multímetro registró las variaciones de voltaje al exponer el sensor a diferentes cargas.</p> <p>Por otra parte, en la fase digital se logra apreciar el pico correspondiente a un aumento casi instantáneo de fuerza al momento de ingresar el trocar en el dispositivo de entrenamiento, y su correspondiente verificación</p>

		a través de la tabla de resultados.
Inserción a la base de datos, consulta de usuario, consulta de sesión actual e histórico.	Verificar que se logra actualizar y registrar la información tanto del usuario como la correspondiente a la sesión de entrenamiento, y que dichos valores adquiridos sean guardados con el usuario correcto en la sesión correcta.	<p>La primera validación la obtenemos al poder registrar un usuario en la ventana de registro, posteriormente al iniciar sesión encontramos la segunda validación correspondiente al uso correcto del <i>query</i> para poder otorgar acceso a nuestro usuario final, mostrando la información personal.</p> <p>Por último, la inserción de los datos de fuerza y presión en la tabla correspondiente, así como su visualización una vez que son llamados por medio del botón "Histórico" o el botón Buscar correspondiente a la sección de "sesión". Se logró verificar con éxito, corroborando de esta forma el correcto funcionamiento de las diferentes consultas a la base de datos.</p>

Tabla 16 Descripción de las pruebas de validación



Figura 69 Búsqueda de puertos



Figura 70 Conexión exitosa del puerto serie

4.4. Demostración de funcionamiento

El proceso general para el uso de la plataforma comienza con la preparación del entrenador físico, el cual tiene 2 abrazaderas que deben ser separadas como podemos apreciar en la Figura 71.

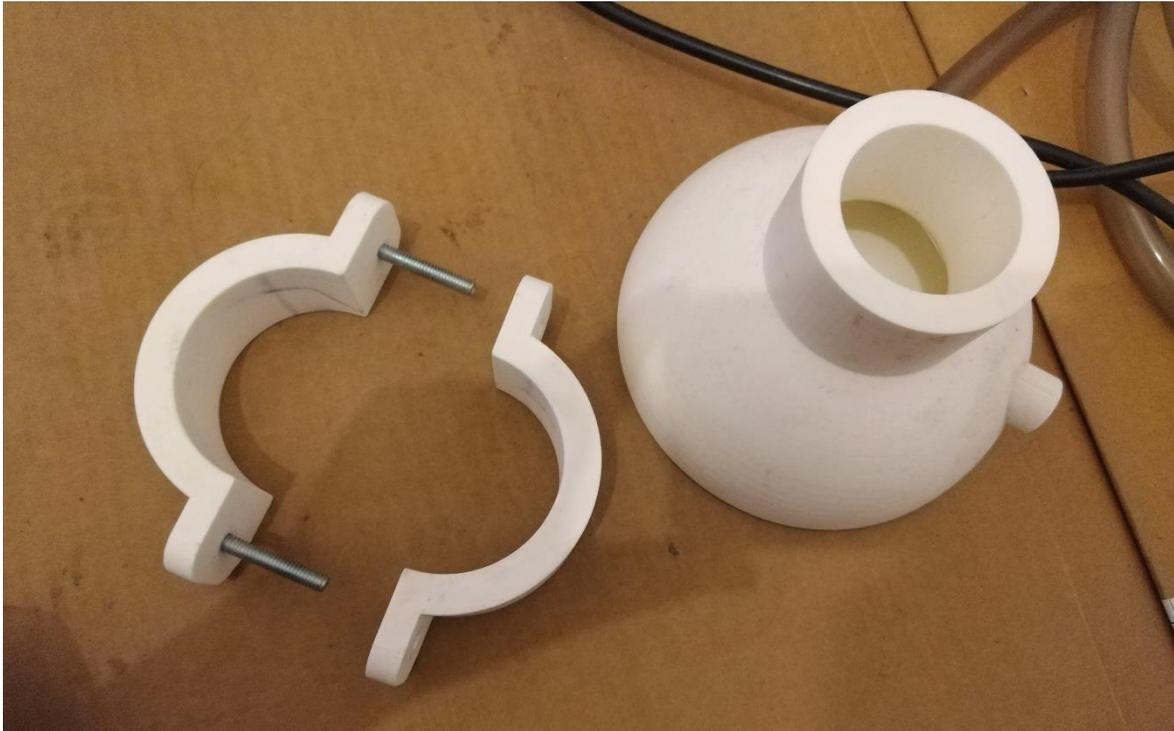


Figura 71 Separación del entrenador físico

Una vez hecho lo anterior procedemos a colocar en su respectivo lugar el pad de entrenamiento (Figura 72). Es en este momento cuando se colocan de forma paralela cada una de las abrazaderas (Figura 73), con la intención de sellar el pad de entrenamiento de la forma más hermética posible, para evitar así demasiadas fugas de gas. Posteriormente, bajamos las abrazaderas hasta llegar a su tope donde comenzamos a ajustar al ancho del borde superior del entrenador y de esta manera dejar el entrenador listo antes del ajuste final como podremos apreciar en la Figura 74.

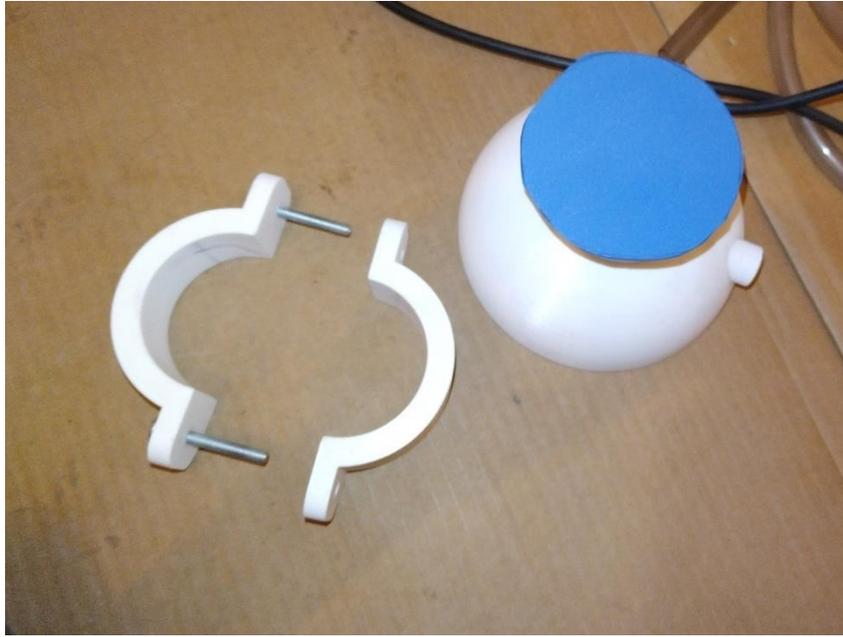


Figura 72 Adaptación del pad de entrenamiento

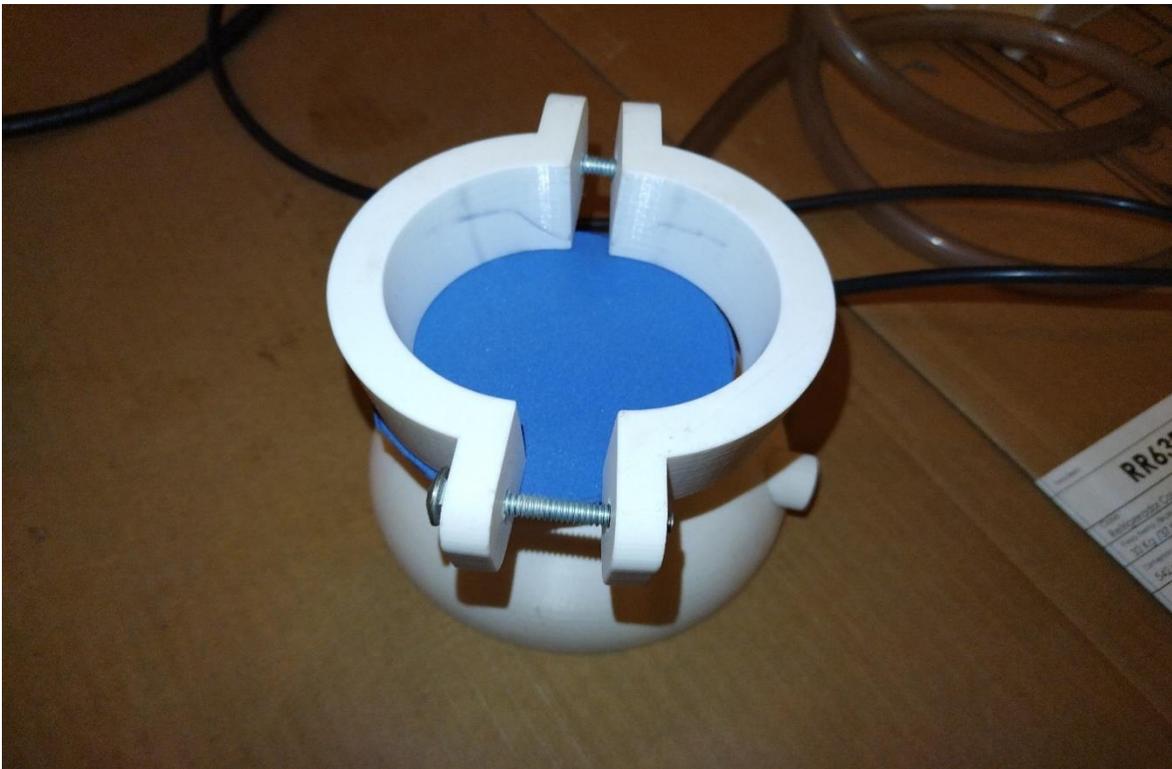


Figura 73 Inserción de abrazaderas



Figura 74 Cierre semiermético, adición de tuercas de seguridad.

Una vez colocadas las abrazaderas, se procede a insertar las tuercas de seguridad hasta la pared lateral de la abrazadera, como podremos apreciar en la Figura 74. Finalmente se realiza un ajuste final mediante las tuercas mariposa (Figura 75), quienes son las encargadas de realizar un cierre final y así garantizar que el pad de entrenamiento se quedará en su lugar durante toda la sesión. Además, garantizamos que el aire al interior del dispositivo de entrenamiento no presente grandes fugas.

Una vez terminados los ajustes de la sección anterior podemos ver en la Figura 76 la agrupación de los elementos externos a utilizar, en este caso la conexión de la manguera al dispositivo de entrenamiento, así como el correcto posicionamiento para lograr manipular el trocar de forma ergonómica.



Figura 75 Inserción de tuercas mariposa

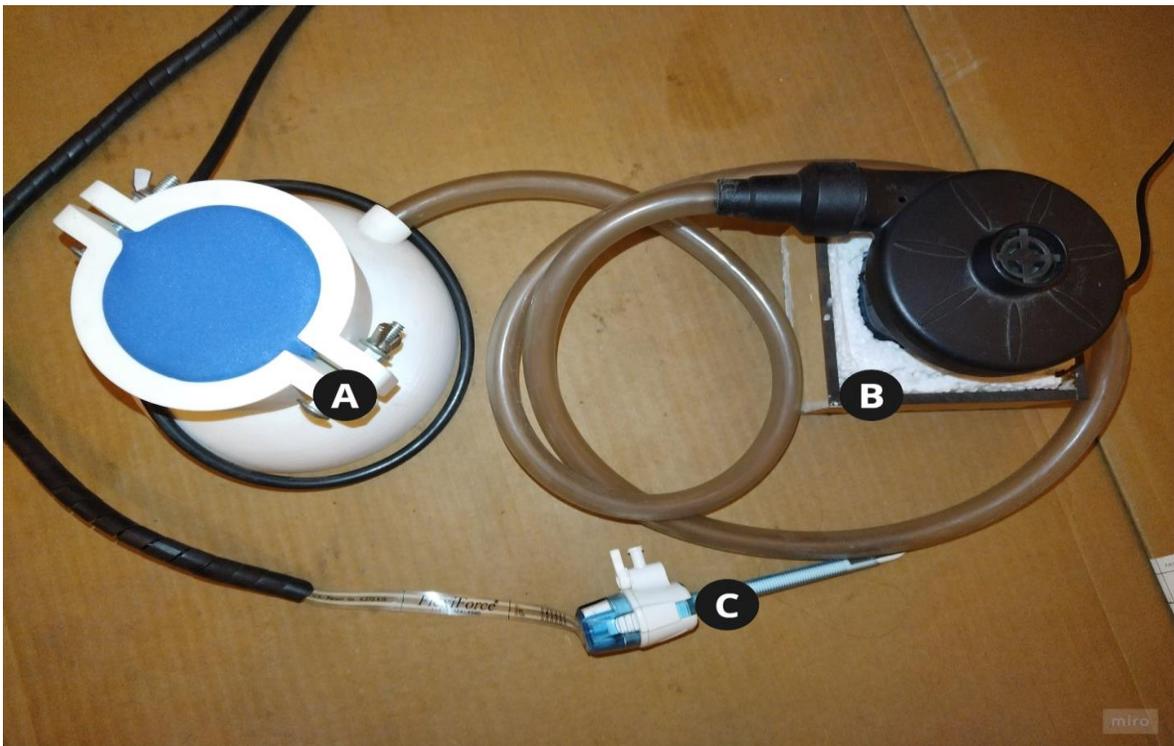


Figura 76 Preparación del A Entrenador, B insuflador y C Trocar

Posteriormente, se procede a realizar la conexión de la fuente de alimentación (teniendo en cuenta las consideraciones vistas en la sección de instrucciones de uso), e iniciamos sesión (Figura 77) en la plataforma ingresando el correo y contraseña que guardamos al realizar el registro de usuario.

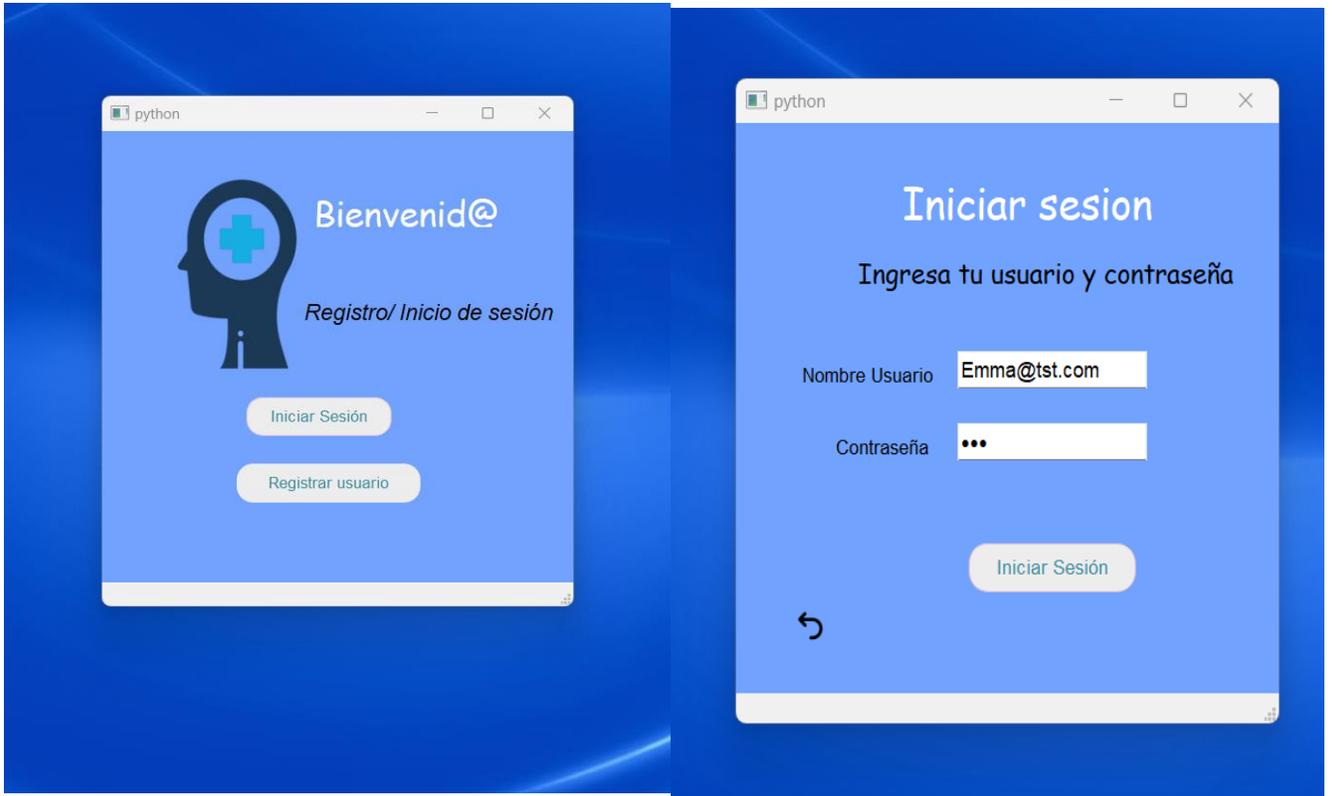


Figura 77 Pantalla inicio de sesión

Una vez dentro de la plataforma (Figura 78), podemos observar nuestra información básica, además de la fecha en la cual estamos accediendo y el número de la última sesión. Continuando con el proceso, iniciamos una nueva sesión de entrenamiento mediante el botón “Nuevo proceso” y procedemos a realizar el ajuste y calibración tal como se indicó en la sección de instrucciones de uso y configuración de la plataforma. Para este ejemplo se consideró utilizar una presión estándar de 12 mmHG, como podemos apreciar en la Figura 79.

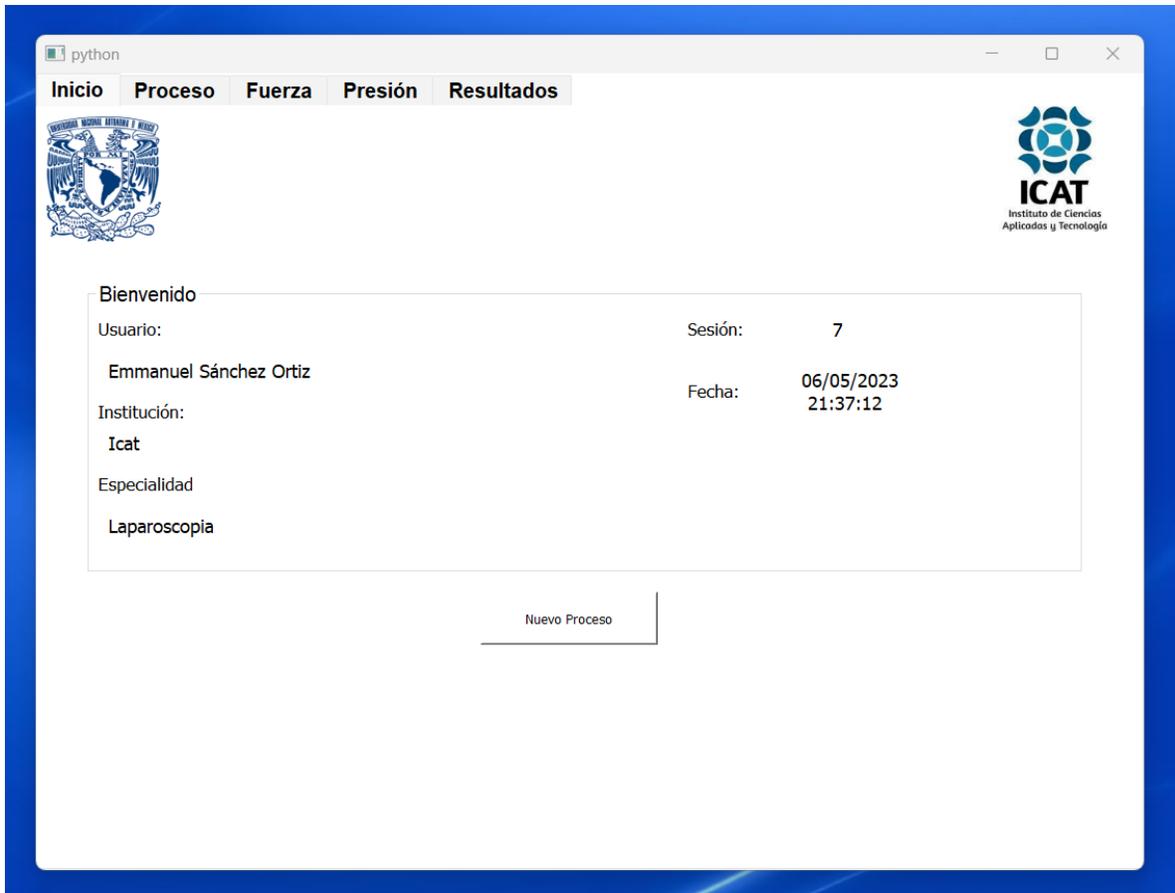


Figura 78 Pestaña Inicio de la plataforma.

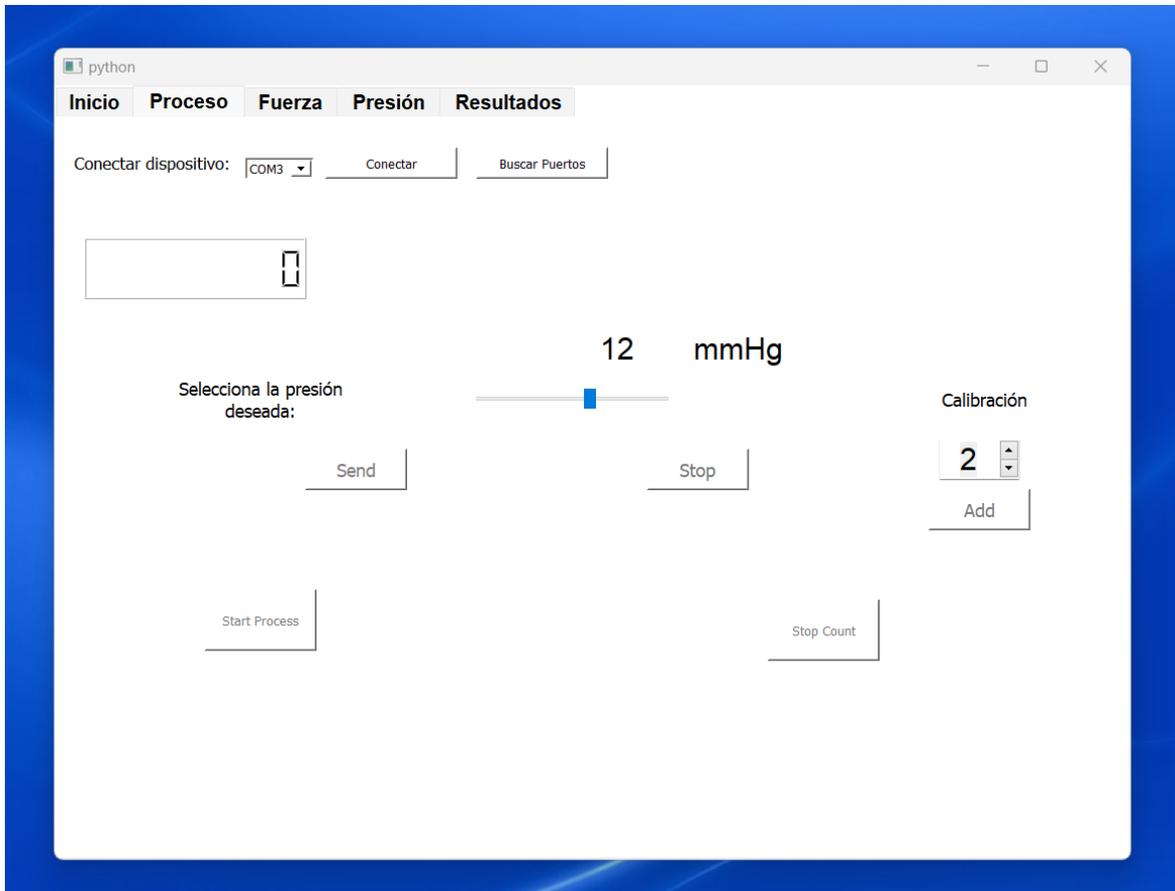


Figura 79 Configuración presión a 12 mmHG

Una vez activado el insuflador y con nuestro dispositivo recibiendo datos, preparamos nuestro trocar desactivando el dispositivo de seguridad como podemos apreciar en la Figura 80A, enseguida se acomoda la mano con la técnica correcta de sujeción de trocar (Figura 80B), y se procede a realizar la punción (Figura 80C) ingresando la muñeca y rotando al mismo tiempo para disminuir el riesgo de lesiones. Después de la punción, dejamos de presionar y podemos dejar nuestro trocar listo para los siguientes procesos de la operación. Al concluir la práctica, se detiene el proceso de adquisición de datos mediante el botón de *Stop Count*.

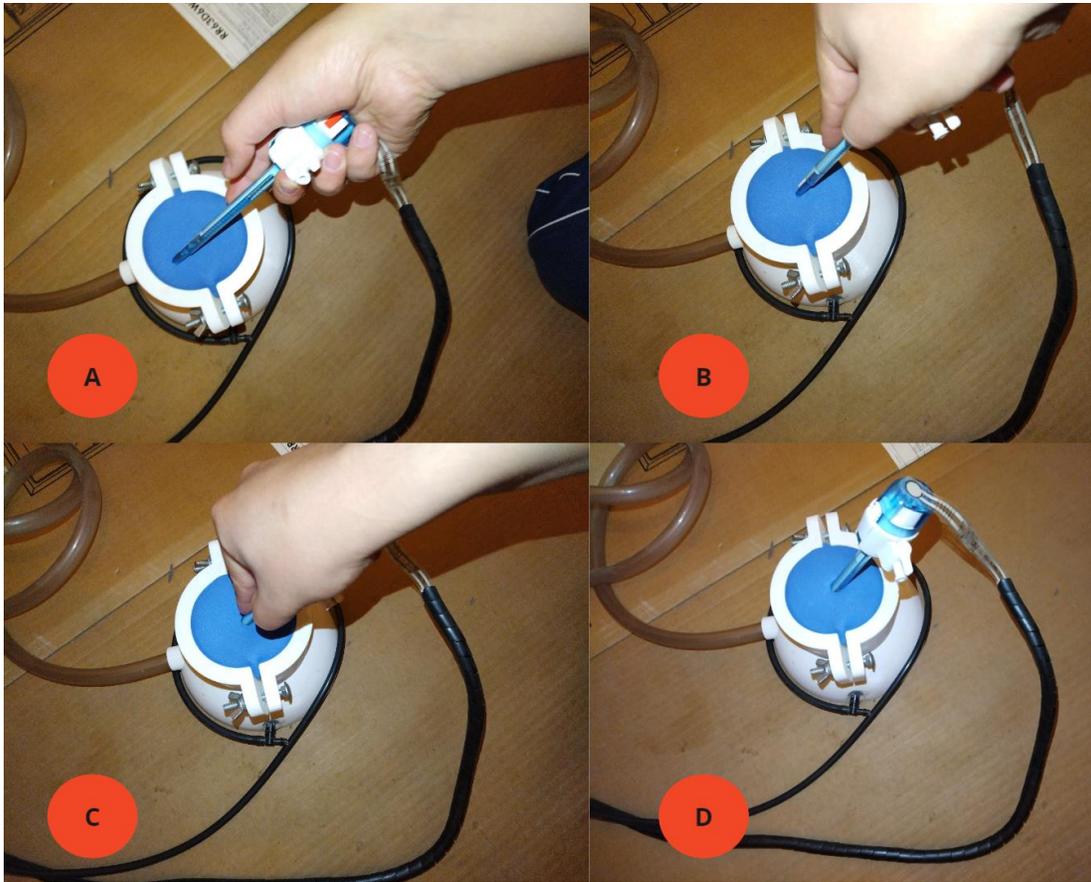


Figura 80 A Preparación del trocar, B Técnica correcta de sujeción, C Punción, D Trocar colocado.

Al finalizar el proceso guardamos el registro de la presente sesión en la base de datos y seremos redirigidos a la pestaña de fuerza (Figura 81), en este caso, podemos ver que la gráfica despliega diferentes proyecciones de fuerza, por un lado, fuerzas que no superan los 2 N, las cuales pueden ser interpretadas como aquellas que suceden al momento de realizar el ajuste del trocar a la posición correcta. Por otro lado, existe un pico de fuerza correspondiente al momento de realizar la punción del trocar, en el presente ejemplo podemos apreciar una fuerza superior a los 16 N.

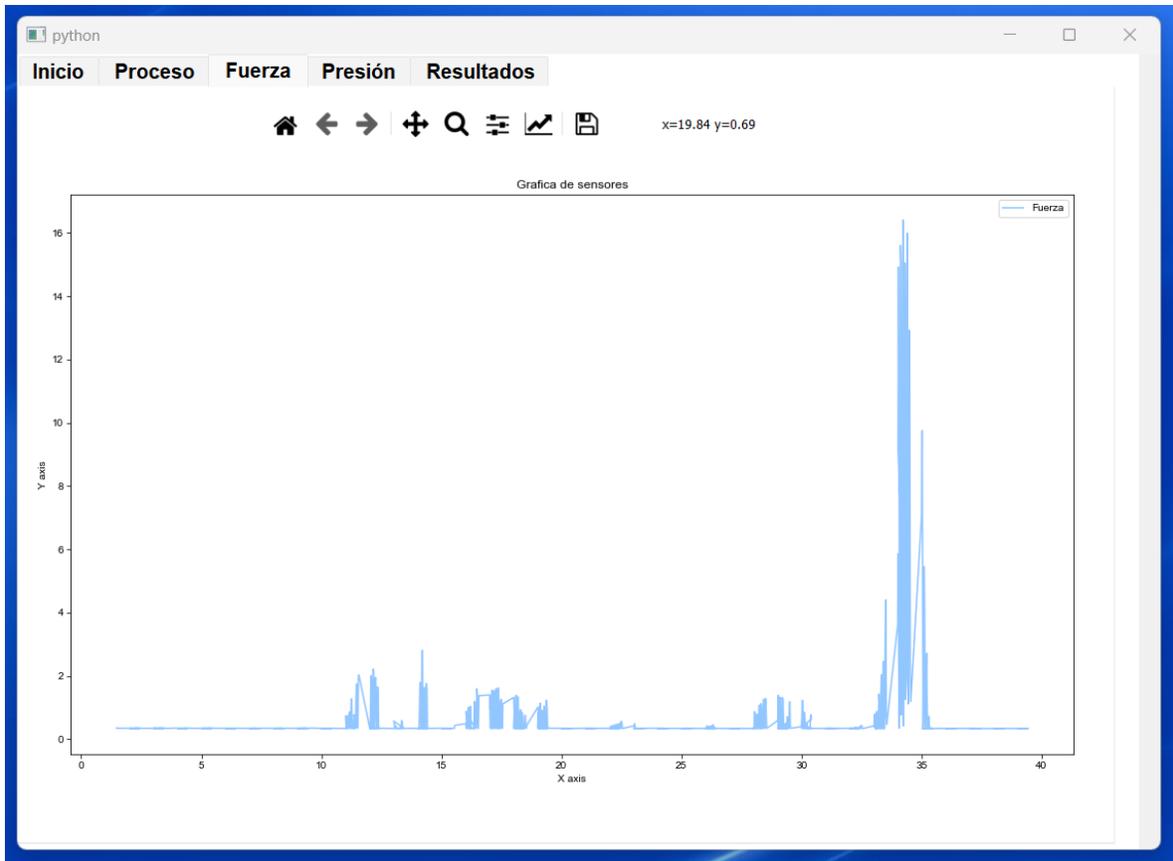


Figura 81 Pestaña Fuerza posterior al entrenamiento

En la información de la presión (Figura 82), se aprecia que entre el segundo 30 y 35 existe una variación de la presión correspondiente al mismo tiempo en el cual en la pestaña de Fuerza (Figura 81) registra el mayor pico de fuerza. Tenemos disponible la información detallada de la presión y la fuerza registradas durante el entrenamiento en la pestaña resultados dónde para este ejemplo vemos que la sesión No. 8 tuvo una duración total de 39.43 Segundos, y una fuerza máxima de 16.407 N (Figura 83).

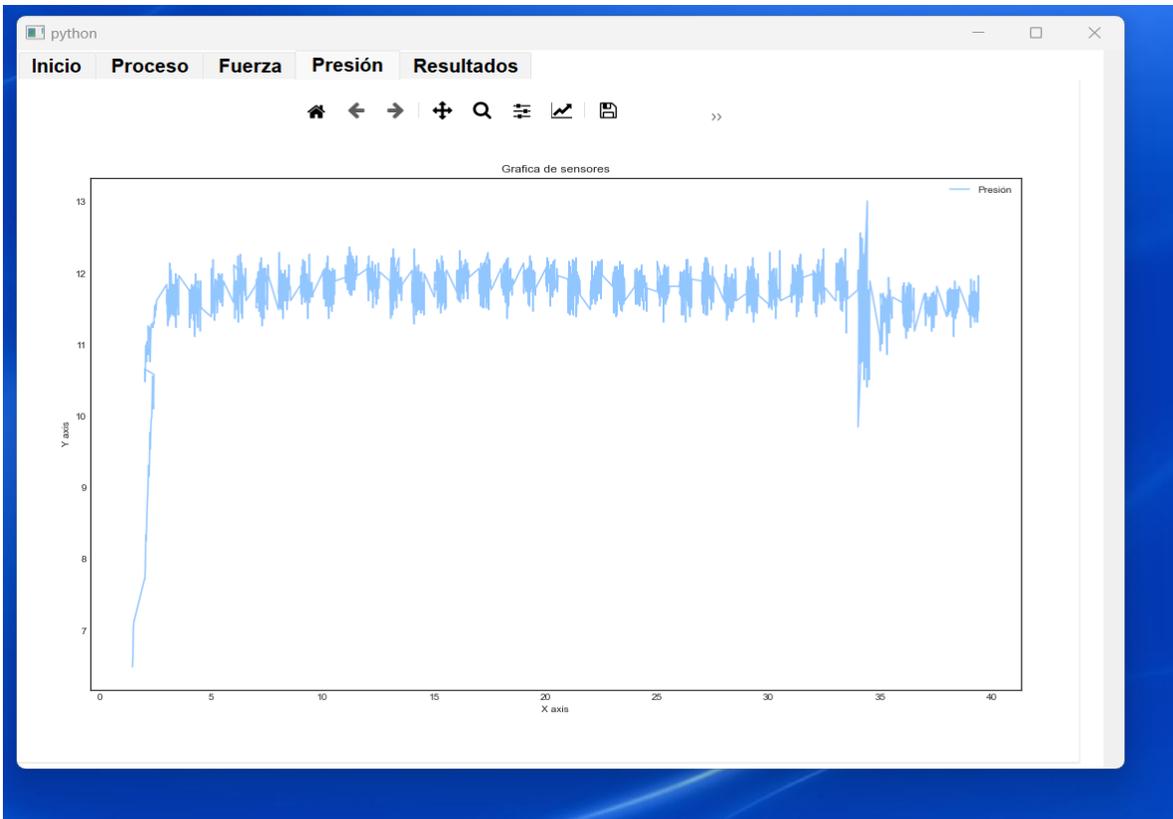


Figura 82 Pestaña Presión posterior al entrenamiento

	Sesion	Fuerza	Presión
1	8	0.35	6.49
2	8	0.34	6.62
3	8	0.35	7.04
4	8	0.35	7.12
5	8	0.34	7.15
6	8	0.35	7.72
7	8	0.35	7.72
8	8	0.35	7.87
9	8	0.35	8.34
10	8	0.35	8.26
11	8	0.35	8.52
12	8	0.34	8.72
13	8	0.34	8.85
14	8	0.34	9.03
15	8	0.35	9.24

Duración Total: 39.43010 Sec

Fuerza Máxima: 16.4075 Newtons

Figura 83 Pestaña Resultados posterior al entrenamiento

Capítulo 5 Conclusiones

En el presente trabajo de tesis de licenciatura, se desarrolló un sistema de entrenamiento funcional para el entrenamiento de la colocación de dispositivos trocares en la cirugía laparoscópica, la cual logró cumplir tanto con el objetivo general como con los objetivos específicos:

1. Desarrollar un sistema de bombeo que simule la insuflación de gas CO₂, en un ambiente de entrenamiento funcional, dicho sistema fue creado a través del control de la bomba por medio del Arduino Nano.
2. Creación de un entorno de simulación para una cavidad abdominal. Se diseñó un entorno físico capaz de adecuar pads de entrenamiento que simulan el tejido abdominal, además de albergar la entrada de aire y brindar un cierre semihermético para las punciones con el trocar.
3. Programación de microcontrolador y un entorno gráfico que pueda registrar los datos proporcionados por nuestros sensores para poder mandar la señal proveniente de los sensores y al mismo tiempo controlar el flujo de aire generado por la bomba, así como el desarrollo de la plataforma para el registro de las diferentes sesiones de entrenamiento.
4. Diseño mecánico de la interfaz de entrenamiento, diseñado con el software Inventor (Autodesk Student Edition) para la parte física de la plataforma de entrenamiento, así como el diseño en nanoCAD (Free edition) de ambas cajas para el soporte de los materiales electrónicos y el insuflador.
5. Diseño electrónico de los elementos, diagrama y simulación de nuestro sistema de recepción de señal, lo cual se generó mediante un programa especializado para el diseño y simulación de elementos eléctricos.
6. Acondicionamiento de la señal, modelado matemático y calibración de los sensores, realizados con éxito para poder obtener los diferentes modelos matemáticos para el control y graficación de las señales físicas provenientes de los sensores, así como el control de la bomba que representa la etapa de insuflación.
7. Obtención de modelo matemático que discrimine la calidad de la punción, el cual corresponde al registro obtenido a través del sensor de fuerza que nos

brinda el comportamiento de la fuerza empleada al momento de realizar una sesión de entrenamiento, que podemos guardar para futuros análisis.

8. Desarrollo de base de datos local para posterior consulta, generada a través de sqlite, dónde podemos guardar los datos exactos de fuerza y presión, además de los diferentes usuarios y la duración de cada sesión de entrenamiento.

5.1 Trabajo a futuro

El presente trabajo logró el desarrollo de un entorno funcional de entrenamiento; sin embargo, existen múltiples observaciones a tomar como punto de partida para futuros desarrollos, los cuales se enlistan en la siguiente sección:

1. Agregar elementos estadísticos en una pestaña auxiliar para poder visualizar, clasificar, ordenar y segmentar la información de acuerdo con los diferentes parámetros de usuario registrados, y así de esta forma generar reportes para brindar una realimentación crítica correspondiente al desempeño del aprendiz.
2. Incluir el rol de profesor, y generar el entorno de control de estudiantes mediante el cual se pueda observar y consultar el desempeño de los alumnos en la misma plataforma lo cual brindaría un apoyo en la docencia.
3. Añadir diferentes tipos de cierre y empaques para la adaptación de la manguera de flujo de aire y la manguera de salida hacia el sensor de presión, esto con la finalidad de asegurar un correcto acoplamiento entre ambos dispositivos mejorando en la facilidad y practicidad de uso.
4. Realizar un cambio en los diseños de la interfaz de entrenamiento de tal modo que brinden una mejor experiencia de usuario con elementos más amigables con el usuario final.
5. Encapsular y diseñar un modelo 3D para la caja electrónica final, así como su respectivo modelo para la caja del insuflador de tal forma que ambas logren permanecer en un solo sistema evitando así tener los sistemas por separado.
6. Añadir un sensor de posicionamiento para poder adquirir la información de la posición del instrumento durante la sesión de entrenamiento.
7. Adquisición y visualización en tiempo real del dispositivo, así como los múltiples registros de información. En esta aplicación se podría implementar

un modelo tridimensional virtual que brindara una simulación de lo ocurrido en el exterior.

8. Utilizar sensores que brinden una mayor resolución, así como un ADC externo con mayor capacidad. Aunado a lo anterior, se sugiere implementar un dispositivo de adquisición de datos diferente con la finalidad de mejorar la frecuencia de muestreo y con ello incrementar la sensibilidad y precisión de nuestro sistema de entrenamiento.

Índice de figuras de apéndices y anexos

Figura A I Plano de la vista Isométrica del entrenador	116
Figura A II Plano de la vista isométrica del cierre semihermético.....	117
Figura A III Plano de la vista Isométrica de la caja de soporte electrónica	118
Figura A IV Plano de la vista Isométrica de la caja de soporte al insuflador.....	119

Índice de tablas de apéndices y anexos

Tabla A I Regulador de voltaje lineal fijo (- 5 V)	120
Tabla A II Regulador de voltaje lineal fijo 3.3[V]	120
Tabla A III Amplificador operacional MCP6002	121
Tabla A IV Fuente de voltaje	121

Apéndice

Apéndice I Vista Isométrica diseño del entrenador

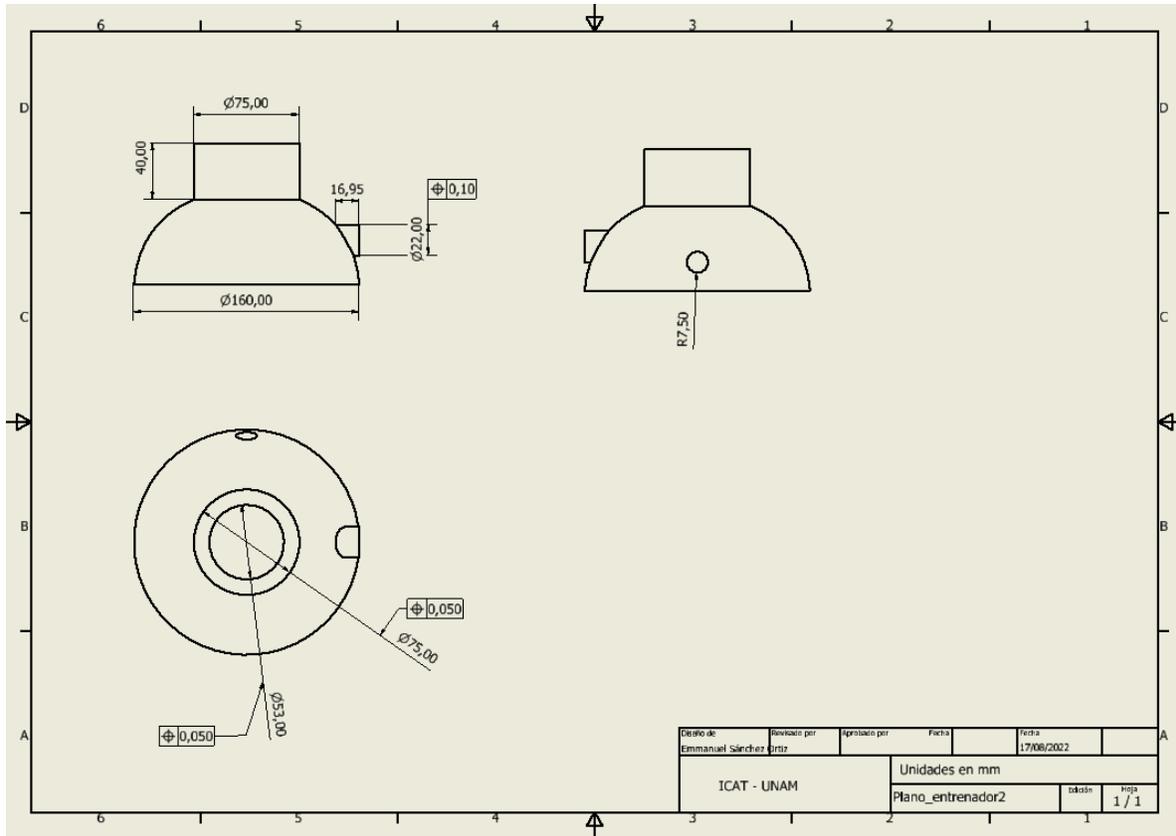


Figura A I Plano de la vista Isométrica del entrenador

Apéndice II Vista Isométrica diseño del cierre semihermético

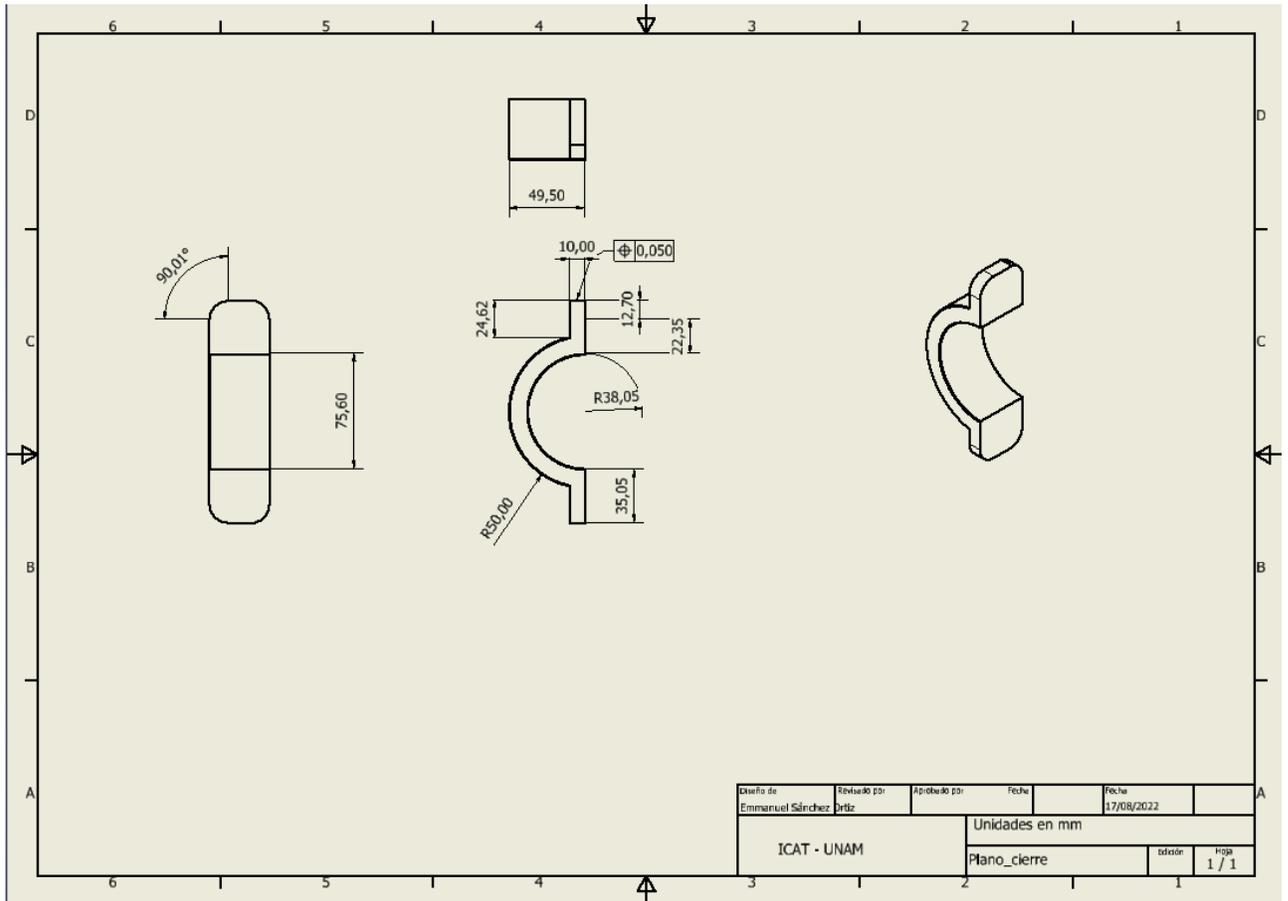


Figura A II Plano de la vista isométrica del cierre semihermético

Apéndice III Vista Isométrico caja para soporte electrónico y caja de soporte al insuflador

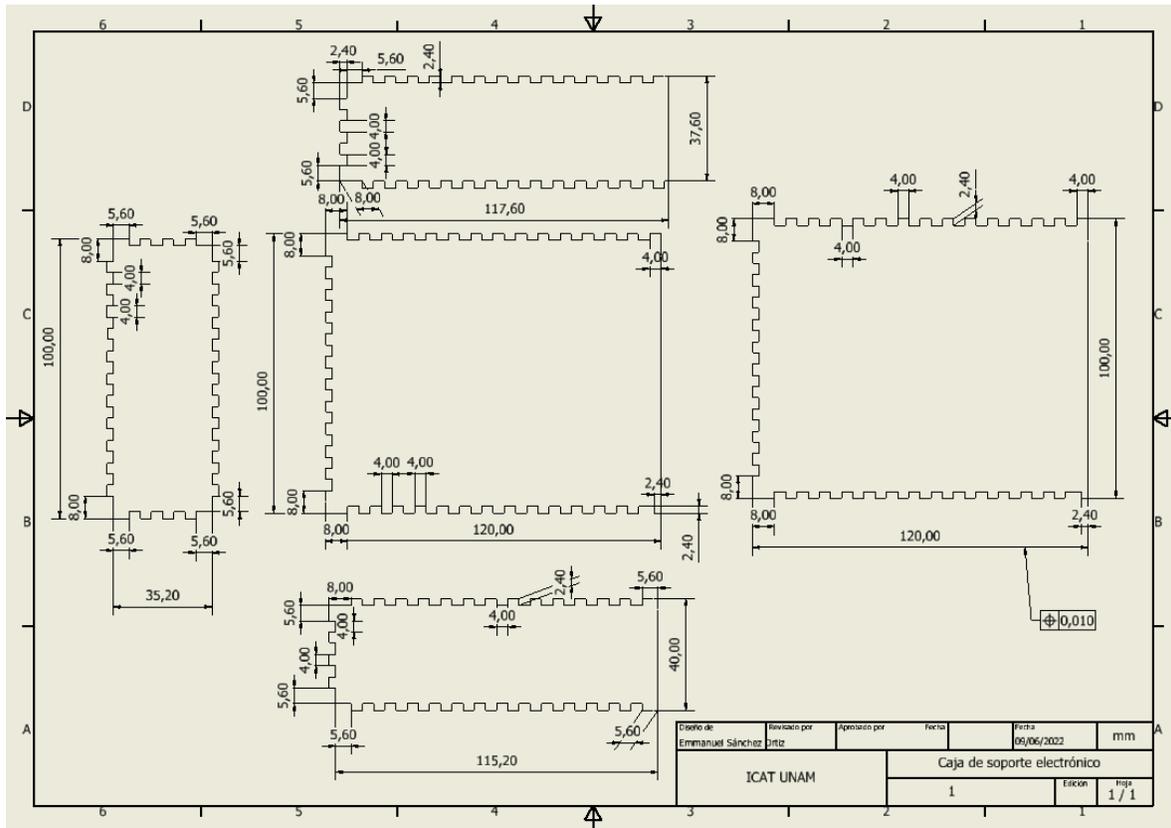


Figura A III Plano de la vista Isométrica de la caja de soporte electrónica

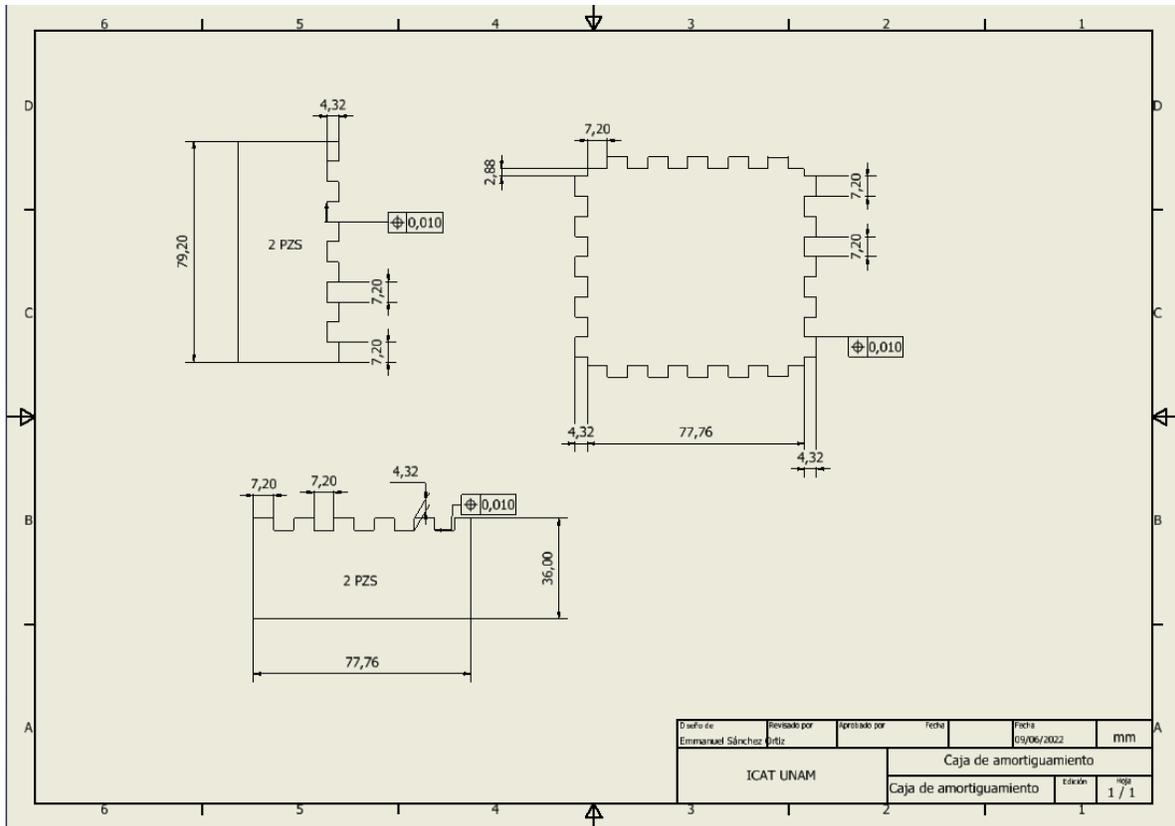


Figura A IV Plano de la vista Isométrica de la caja de soporte al insuflador

Apéndice IV Tablas informativas

Tabla A I Regulador de voltaje lineal fijo (- 5 V)

Características Eléctricas (LM7905C – Regulador de voltaje)	
Voltaje entrada máxima	-25 [V]
Temperatura máxima de operación	0 [°C] – 124[°C]
Corriente de salida	500 [mA]
Voltaje de salida	-5 [V]
Disipación de potencia	1.5 [W]
Coeficiente de temperatura media a la salida de voltaje	0.4 [mV/°C]

Tabla A II Regulador de voltaje lineal fijo 3.3[V]

Características Eléctricas (LF33CV – Regulador de voltaje)	
Voltaje entrada máxima	16 [V]
Temperatura máxima de operación	-40[°C] – 125[°C]
Corriente de salida Máxima	1 [A]
Voltaje de salida	3.3 [V]

Tabla A III Amplificador operacional MCP6002

Características Eléctricas (MCP6002 – Amplificador operacional)	
Voltaje de alimentación	1.8 [V] – 6 [V]
Temperatura de alimentación	-40 [°C] – 125[°C]
Corriente de alimentación	100 [uA]
Voltaje de entrada offset	4.5[mV]
Corriente en los pines analógicos	± 2[mA]
Entradas analógicas	$V_{ss} - 1[V]$ a $+ V_{dd} + 1[V]$

Tabla A IV Fuente de voltaje

Características Fuente voltaje	
Tipo	Fuente conmutada de 5 Vcd a 10 [A]
Voltaje de entrada	110 – 220 [v] 50/60 [Hz]
Potencia máxima de salida	50 [W]
Temperatura de operación	-10[°C] – 50 [°C]
Protección	Cortocircuito – Sobrecarga – Sobretensión
Peso	0.35 [Kg]
Dimensiones	16 [cm] * 9.9 [cm] * 4.2 [cm] (L * W *H)
Terminales (salida)	-V (terminal negativa), +V (terminal positiva), ADJ (ajuste del voltaje de salida al ±5 [V])
Terminales (Entrada)	T (Terminal a tierra física), N (Terminal neutra), L (Terminal a fase).

Apéndice V Código de lectura de valores analógicos

```
#include <Adafruit_ADS1X15.h>
//Creamos un objeto de la clase
Adafruit_ADS1115 adc;

String valor= "";
String validator,valueP = "";
String action = "off";

void setup() {
  // put your setup code here, to run once:
  adc.begin();
  Serial.begin(115200);
}

void loop() {
  // put your main code here, to run repeatedly:
  if(validator == "da"){
    sendData();
  }

  if(Serial.available() > 0){
    valor = Serial.readString();
    validator = valor.substring(0,2);
  }
}

void sendData(){

  short RealP = adc.readADC_SingleEnded(0);
  short RealF = adc.readADC_SingleEnded(1);
  Serial.println(RealP,2);

}
```

Apéndice VI Código controlador del insuflador, monitoreo de sensores y envío de datos

```
ProgramaFinal
/*
 * One arduino for dataloggin
 * @autor: Emmanuel Sánchez Ortiz
 * Fecha: 21/05/2022
 *
 */
#include <Adafruit_ADS1X15.h>
//Creamos un objeto de la clase
Adafruit_ADS1115 adc;

#define pinMosfet 6

String valor= ""; // Variable temporal para guardar la información proveniente de la configuración de la presión deseada.
String validator,valueP,cal = "";
//Validator: son los primeros 2 dígitos del payload mandado al arduino desde la computadora dicho validador comprende de 2 estados
//CA = Custom Acquisition, nos sirve para poder determinar si se desea interactuar con la bomba.
//DA = Data acquisition, mediante el presente código se le indica al microcontrolador que debe mandar las lecturas de presión y
//fuerza registradas mediante el
//ADC externo.

String action = "off";
//La variable action, me da el control del estado específico de la bomba si se desea encender o apagar.
int sPressure;
// sPressure = Selected pressure, transforma la variable a entero para poder ser ingresada dentro de la formula de control de pwm,
// y así regular la potencia de la bomba.
int calibration = 0;

void setup() {
  // put your setup code here, to run once:
  pinMode(pinMosfet, OUTPUT);
  Serial.begin(500000);
  //adc.setGain(GAIN_FOUR);
  //adc.setGain(GAIN_ONE);
  adc.begin();
}

void getCalibrationValue(){
  cal= valor.substring(2,4);
  calibration = cal.toInt();
}

void getValue(){
  valueP= valor.substring(2,4);
  sPressure = valueP.toInt();
  action= valor.substring(4,7);
}

void startInsuflator(){
  // -0.0097*POWER(C24,4)+0.4619*POWER(C24,3)-7.2845*POWER(C24,2)+58.375*C24-85.815
  float pwmval = -(0.0097*pow(sPressure,4)) + (0.4619*pow(sPressure,3)) - (7.2845*pow(sPressure,2)) + (58.375*sPressure) - 85.815;
  pwmval = round(pwmval) + calibration;
  if(sPressure >= 11 ){
    pwmval = pwmval +1;
  }
  if(sPressure >=14 ){
    pwmval = pwmval +1;
  }
  //Accionar la bomba
  analogWrite(pinMosfet,pwmval);
}

void sendMultipleData(){
  int RealP = adc.readADC_SingleEnded(0);
  int RealF = adc.readADC_SingleEnded(1);

  Serial.print("[");
  Serial.print(RealF);
  Serial.print(",");
  Serial.println(RealP);
  delay(1);
}
}
```

```

void loop() {
  // put your main code here, to run repeatedly:
  if (validator == "da") {
    sendMultipleData();
  } else if (validator == "ba") {
    getCalibrationValue();
  }
}

if (Serial.available() > 0) {
  valor = Serial.readString();
  validator = valor.substring(0, 2);
  //Serial.println(validator);
  if (validator == "ca") {
    getValue();
    digitalWrite(13, HIGH);
    delay(1000);
    digitalWrite(13, LOW);
    delay(1000);

    if (action == "off") {
      digitalWrite(pinMosfet, LOW);
    } else {
      //hola();
      startInflator();
    }
  }
}
}
}
}

```

Apéndice VII Código programa principal (Interfaz Gráfica)

```

1  import sys
2  import time
3  from PyQt5.uic import loadUi
4  from PyQt5 import QtWidgets
5  from PyQt5.QtWidgets import QDialog, QApplication, QStackedWidget, QMainWindow, QTableWidgetItem
6  from PyQt5.QtCore import QTimer, QDateTime
7
8  import numpy as np
9  import matplotlib.pyplot as plt
10 import pandas as pd
11 from tkinter import messagebox
12 import tkinter as tk
13
14 from matplotlib.backends.backend_qt5agg import FigureCanvasQTAgg as FigureCanvas, NavigationToolbar2QT as Navi
15 from matplotlib.figure import Figure
16
17 from FindPorts import *
18 from ArduinoCommunication import *
19 from user import *
20
21 import sqlite3
22 import os.path
23
24
25 root = tk.Tk()
26 root.withdraw()
27
28 # Obtener el path donde se encuentra ese archivo
29 BASE_DIR = os.path.dirname(os.path.abspath(__file__))
30 db_path = os.path.join(BASE_DIR, "usuarios.db")
31
32

```

```

33 class logInScreen(QMainWindow):
34     def __init__(self):
35         super(logInScreen, self).__init__()
36         db_pathLogIn = os.path.join(BASE_DIR, "ventanaLogIn.ui")
37         loadUi(db_pathLogIn, self)
38         self.input_contrasena.setEchoMode(QtWidgets.QLineEdit.Password)
39         self.Button_login.clicked.connect(self.logginUser)
40         self.BackButton.clicked.connect(self.returnWindow)
41
42     def returnWindow(self):
43         curr = widget.currentWidget()
44         widget.removeWidget(curr)
45
46     def logginUser(self):
47         # obtenemos los campos de usuario y contraseña
48         user = self.input_usuario.text()
49         password = self.input_contrasena.text()
50
51         if len(user) == 0 or len(password) == 0:
52             self.label_alerta.setText("* Llenar todos los campos")
53             self.alerta_u.setText("")
54             self.alerta_p.setText("")
55         else:
56             try:
57                 db = sqlite3.connect(db_path)
58                 cursor = db.cursor()
59                 # query = 'SELECT Contraseña FROM USUARIO where Correo = \''+user+ '\''
60                 # cursor.execute(query)
61                 cursor.execute('SELECT * FROM usuario where Correo = \''+user+ '\',()')
62
63                 resp_pwd = cursor.fetchone()
64                 global currentUser
65                 currentUser = usuario(resp_pwd[0], resp_pwd[1], resp_pwd[2],
66
67                 resp_pwd[3], resp_pwd[4], resp_pwd[5], resp_pwd[6], resp_pwd[7],
68                 resp_pwd[8], resp_pwd[9], resp_pwd[10])
69                 nombre = currentUser.Nombre
70                 pas = currentUser.contrasena
71
72                 if pas == password:
73                     self.label_alerta.setText(f"Bienvenido!!{nombre} {currentUser.aPaterno} {currentUser.aMaterno}")
74                     self.alerta_u.setText("")
75                     self.alerta_p.setText("")
76                     self.gotoUserWindow()
77                 else:
78                     self.label_alerta.setText("Usuario/contraseña incorrectos")
79
80             except:
81                 print("no user")
82
83             db.commit()
84             db.close()
85
86     def gotoUserWindow(self):
87         userWin = userWindow()
88         widget.addWidget(userWin)
89         widget.setCurrentIndex(widget.currentIndex() + 1)
90         widget.resize(1071, 771)
91
92

```

```

93 class welcomeScreen(QMainWindow):
94     def __init__(self):
95         super(welcomeScreen, self).__init__()
96         db_pathWE = os.path.join(BASE_DIR, "ventanaPrincipal.ui")
97
98         loadUi(db_pathWE, self)
99         self.Button_Inicio.clicked.connect(self.gotoLoginWindow)
100        self.Button_Registro.clicked.connect(self.gotoRegisterWindow)
101
102    def gotoRegisterWindow(self):
103        register = registerScreen()
104        widget.addWidget(register)
105        widget.resize(500,595)
106        widget.setCurrentIndex(widget.currentIndex() + 1)
107
108    def gotoLoginWindow(self):
109        login = logInScreen()
110        widget.addWidget(login)
111        widget.setCurrentIndex(widget.currentIndex() + 1)
112
113
114 class registerScreen(QDialog):
115     def __init__(self):
116         super(registerScreen, self).__init__()
117         db_pathRE = os.path.join(BASE_DIR, "ventanaRegistro.ui")
118
119         loadUi(db_pathRE, self)
120         self.Button_registro.clicked.connect(self.userRegister)
121         self.input_contrasena.setEchoMode(QtWidgets.QLineEdit.Password)
122         self.input_contrasena_2.setEchoMode(QtWidgets.QLineEdit.Password)
123         self.Button_back.clicked.connect(self.returnWindow)
124
125
126     def userRegister(self):
127         # obtener todos los campos
128         nombre_usuario = self.input_nombre.text()
129         apellido_p = self.input_aPaterno.text()
130         apellido_m = self.input_aMaterno.text()
131         correo = self.input_correo.text()
132         contrasena = self.input_contrasena.text()
133         contrasena2 = self.input_contrasena_2.text()
134         especialidad = self.input_Especialidad.text()
135         organizacion = self.input_organizacion.text()
136         sexo = self.getSex()
137         edad = self.edad.value()
138
139         if (len(nombre_usuario) == 0 or
140             len(apellido_p) == 0 or
141             len(apellido_m) == 0 or
142             len(correo) == 0 or
143             len(contrasena) == 0 or
144             len(contrasena2) == 0 or
145             len(especialidad) == 0 or
146             len(organizacion) == 0 ) or (sexo == '-') or (
147             edad < 12 ):
148             if (edad < 18 ):
149                 self.label_alerta.setText("* No puede ser una edad menor a 18 años")
150
151             else:
152                 self.label_alerta.setText("* Por favor registrar todos los campos")
153         elif (contrasena != contrasena2):
154             self.label_alerta.setText("")
155             self.label_alerta.setText("* Las contraseñas no pueden \n ser distintas")
156         else:
157             db = sqlite3.connect(db_path)
158             cursor = db.cursor()
159             user_info = [nombre_usuario, apellido_p, apellido_m,
160                 correo, contrasena, edad, sexo, especialidad, organizacion, 0]

```

```

161 query = ('INSERT INTO usuario (Nombre,Apellido_paterno,' +
162         'Apellido_materno,Correo,Contrasena,Edad,Sexo,Especialidad,' +
163         'Organizacion_Escuela,Sesion) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)')
164 cursor.execute(query,user_info)
165 db.commit()
166 db.close()
167 option = messagebox.askokcancel("Registro",
168 f"Registro de {nombre_usuario} {apellido_p} {apellido_m} exitoso")
169 if option == True:
170     messagebox.showinfo("Bienvenido","Ahora puedes iniciar sesión desde la " +
171     "venana principal")
172     self.returnWindow()
173 else:
174     messagebox.showinfo("Error","Usuario no registrado")
175
176
177 def getSex(self):
178     if self.sexo_masculino.isChecked():
179         return "M"
180     elif self.sexo_femenino.isChecked():
181         return "F"
182     else:
183         return "-"
184
185 def returnWindow(self):
186     regWin = widget.currentWidget()
187     widget.removeWidget(regWin)
188     widget.resize(490,500)
189
190
191 class Canvas_grafica2(FigureCanvas):
192     def __init__(self,parent =None):
193         figure = Figure(dpi = 73)
194         #figure
195         self.axes = figure.add_subplot(111)
196
197         # self.axes.set_ylim(0, 25)
198         # self.ax.set_xlim(0, 100)
199         super(Canvas_grafica2,self).__init__(figure)
200
201 class Canvas_grafica_p(FigureCanvas):
202     def __init__(self,parent =None):
203         figure_p = Figure(dpi = 73)
204         self.axes = figure_p.add_subplot(111)
205         # self.axes.set_ylim(0, 16)
206         # self.ax.set_xlim(0, 100)
207         super(Canvas_grafica_p,self).__init__(figure_p)
208         figure_p.tight_layout()
209
210
211 class userWindow(QDialog):
212     def __init__(self):
213         super(userWindow,self).__init__()
214         self.fecha = time.strftime("%d/%m/%Y")
215         self.dataInsertDB = []
216         self.sesiones = 0
217         db_pathRE = os.path.join(BASE_DIR, "idea3.ui")
218
219         loadUi(db_pathRE,self)
220         # Eneable the user logIn
221         self.label_Usuario.setText(f"{currentUser.Nombre} {currentUser.aPaterno} {currentUser.aMaterno}")
222         self.label_Institucion.setText(f"{currentUser.organizacion}")
223         self.label_Especialidad.setText(f"{currentUser.especialidad}")
224         self.label_Sesion.setText(f"{currentUser.sesion}")
225         # Disable buttons to start the process
226         self.activeInButtons(False)
227         fecha = time.strftime("%d/%m/%Y")
228         hora = time.strftime("%H:%M:%S")
229         self.label_fecha.setText(f"{fecha} \n {hora}")
230         self.timer=QTimer()

```

```

231     self.port = puertos()
232     self.selPort = ''
233     #Initialize the control buttons
234     self.Button_start.clicked.connect(self.gotoStartProcess)
235     self.Button_SearchSession.clicked.connect(self.getSessionInfo)
236     self.Button_Historic.clicked.connect(self.getHistoric)
237     self.Button_StartCount.clicked.connect(self.startTimer)
238     self.Button_StopCount.clicked.connect(self.endTimer)
239     self.buscarPuertos.clicked.connect(self.findPorts)
240     self.Button_Conect.clicked.connect(self.connectPort)
241     self.val = 0
242     self.flag = False
243
244     # Initialize the ForceGraph
245     self.dataFrame = []
246     self.canv = Canvas_grafica2(self)
247     self.toolbar = None
248     # Initialize the pressure Graph
249     self.dataFrameP = []
250     self.canv_p = Canvas_grafica_p(self)
251     self.toolbar_p = None
252     #Initialize the slider
253     self.sliderP.valueChanged.connect(self.updateSlider)
254     self.valSelPress = "7"
255
256     #initialize arduino conection
257     self.serialArduino = arduino()
258     self.Button_SendPressure.clicked.connect(self.startInsuflator)
259     self.Button_StopInsuflator.clicked.connect(self.stopInsuflator)
260     self.activeButtonPressure(False)
261
262     #initialize the pressure calibration:
263     self.Button_SendCalibration.clicked.connect(self.sendCalibration)
264
265 def activeButtonPressure(self,val):
266
267     self.Button_SendPressure.setEnabled(val)
268     self.Button_StopInsuflator.setEnabled(val)
269     self.Button_SendCalibration.setEnabled(val)
270
271 def dataPressure(self,pressure):
272     if(int(pressure)>9):
273         return str(pressure)
274     else:
275         return '0'+ str(pressure)
276
277 def sendCalibration(self):
278     auxCalibration = self.calibration.value()
279     calibrationValue = self.dataPressure(auxCalibration)
280     value = "ba"+ calibrationValue
281     self.serialArduino.sendData(value)
282
283 def startInsuflator(self):
284     data2ard = self.dataPressure(self.valSelPress)
285     value = "ca"+data2ard+ "onn"
286     print("Datos:\n")
287     print(value)
288     self.serialArduino.sendData(value)
289
290 def stopInsuflator(self):
291     #Habilitar los botones.
292     data2ard = self.dataPressure(self.valSelPress)
293     value = "ca"+data2ard+ "off"
294     self.serialArduino.sendData(value)
295
296 def updateSlider(self,value):
297     self.label_slider.setText(str(value))
298     self.valSelPress = value
299
300 def insertTable(self,fuerza,presion):
301     # Create the structure for the data

```

```

302 self.tableRes.clearContents()
303 row = 0
304 self.tableRes.setRowCount(len(fuerza))
305 for force in fuerza:
306
307     dInsert = (currentUser.sesion, str(force))
308     # sesion
309     self.tableRes.setItem(row, 0, QTableWidgetItem(str(dInsert[0])))
310     # Fuerza
311     auxForce = round(force, 2)
312     self.tableRes.setItem(row, 1, QTableWidgetItem(str(auxForce)))
313     row += 1
314 row = 0
315 for pressure in presion:
316     # Presión
317     auxPress = round(pressure, 2)
318     self.tableRes.setItem(row, 2, QTableWidgetItem(str(auxPress)))
319     row += 1
320
321 def getSessionInfo(self):
322     sesion = self.InputSesion.text()
323
324     if sesion != '':
325         db = sqlite3.connect(db_path)
326         cursor = db.cursor()
327         # Aquí prepararemos los queries para insertar en la tabla
328         cursor.execute(f'SELECT * FROM Datos where Sesion = {sesion} AND Id_Usuario = {currentUser.id}')
329         try:
330             data = cursor.fetchall()
331             db.commit()
332
333             self.tableRes.clearContents()
334             row = 0
335             self.tableRes.setRowCount(len(data))
336             for el in data:
337                 # sesion
338
339                 self.tableRes.setItem(row, 0, QTableWidgetItem(str(el[3])))
340                 # Fuerza
341                 self.tableRes.setItem(row, 1, QTableWidgetItem(str(el[1])))
342
343                 # Presion
344                 self.tableRes.setItem(row, 2, QTableWidgetItem(str(el[2])))
345                 row += 1
346
347         except:
348             print("Error!, No sesion info")
349
350     db.close()
351
352
353 def findPorts(self):
354     self.port.search()
355     if(self.port.disponibles == False):
356         # option = messagebox.askokcancel("Puertos disponibles",)
357         messagebox.showinfo("Puertos disponibles", "No hay puertos disponibles por favor Vuelva a intentarlo")
358     else:
359         self.CBpuertos.addItem(self.port.puertos)
360         messagebox.showinfo("Puertos disponibles", "Puertos Encontrados")
361
362 def activeInButtons(self, val):
363
364     self.Button_StartCount.setEnabled(val)
365     self.Button_StopCount.setEnabled(val)
366
367 def conectPort(self):
368     if self.port.disponibles == True:
369         port = self.CBpuertos.currentText()
370         self.selPort = port
371         try:
372             self.serialArduino.arduino.port = port
373             self.serialArduino.arduino.baudrate = 500000

```

```

374         self.serialArduino.arduino.open()
375         messagebox.showinfo("Conexion",f"Conexión exitosa con el puerto: {port}")
376         self.activeButtonPressure(True)
377         self.activeInButtons(True)
378     except:
379         messagebox.showinfo("Conexion",f"Error de conexión con el puerto: {port}")
380
381     else:
382         messagebox.showinfo("Puertos disponibles", "No hay puertos disponibles por favor Vuelva a intentarlo")
383
384     def updatePressureG(self):
385         plt.clf()
386         #estilos_ :
387         # 'classic',bmh, 'dark_background', 'fast',
388         # 'fivethirtyeight', 'ggplot', 'grayscale', 'seaborn-bright',
389         # 'seaborn-colorblind', 'seaborn-dark-palette', 'seaborn-dark',
390         # 'seaborn-darkgrid', 'seaborn-deep', 'seaborn-muted', 'seaborn-notebook',
391         # 'seaborn-paper', 'seaborn-pastel', 'seaborn-poster', 'seaborn-talk',
392         # 'seaborn-ticks', 'seaborn-white', 'seaborn-whitegrid', 'seaborn',
393         # 'Solarize_Light2', 'tableau-colorblind10']
394         plt.style.use("seaborn-white")
395         try:
396             self.layout_toolbar.removeWidget(self.toolbar_p)
397             self.layout_graphPress.removeWidget(self.canv_p)
398             self.toolbar_p = None
399             self.canv_p = None
400         except Exception as e:
401             print(e)
402             pass
403
404         self.canv_p = Canvas_grafica_p(self)
405         self.toolbar_p = Navi(self.canv_p,self.tab_5)
406         self.layout_toolbar.addWidget(self.toolbar_p)
407         self.layout_graphPress.addWidget(self.canv_p)
408         self.canv_p.axes.cla()
409         ax = self.canv_p.axes
410
411         self.dataFrameP.plot(ax = ax)
412         legend = ax.legend()
413         legend.set_draggable(True)
414         ax.set_xlabel('X axis')
415         ax.set_ylabel('Y axis')
416         ax.set_title('Grafica de sensores')
417         self.canv_p.draw()
418
419     def updateForceG(self):
420         #This is the main graphic
421         plt.clf()
422         #estilos_ :
423         # 'classic',bmh, 'dark_background', 'fast',
424         # 'fivethirtyeight', 'ggplot', 'grayscale', 'seaborn-bright',
425         # 'seaborn-colorblind', 'seaborn-dark-palette', 'seaborn-dark',
426         # 'seaborn-darkgrid', 'seaborn-deep', 'seaborn-muted', 'seaborn-notebook',
427         # 'seaborn-paper', 'seaborn-pastel', 'seaborn-poster', 'seaborn-talk',
428         # 'seaborn-ticks', 'seaborn-white', 'seaborn-whitegrid', 'seaborn',
429         # 'Solarize_Light2', 'tableau-colorblind10']
430         plt.style.use("seaborn-pastel")
431         try:
432             self.horizontalLayout_2.removeWidget(self.toolbar)
433             self.verticalLayout.removeWidget(self.canv)
434             self.toolbar = None
435             self.canv = None
436         except Exception as e:
437             print(e)
438             pass
439
440         self.canv = Canvas_grafica2(self)
441         self.toolbar = Navi(self.canv,self.tab_2)
442         self.horizontalLayout_2.addWidget(self.toolbar)
443         self.verticalLayout.addWidget(self.canv)
444         self.canv.axes.cla()
445         ax = self.canv.axes
446         self.dataFrame.plot(ax = ax)

```

```

446 legend = ax.legend()
447 legend.set_draggable(True)
448 ax.set_xlabel('X axis')
449 ax.set_ylabel('Y axis')
450 ax.set_title('Grafica de sensores')
451 self.canv.draw()
452
453 def resetProcess(self):
454     plt.clf()
455     try:
456         self.serialArduino.flag = False
457         self.horizontalLayout_2.removeWidget(self.toolbar)
458         self.verticalLayout.removeWidget(self.canv)
459         self.toolbar = None
460         self.canv = None
461         self.layout_toolbar.removeWidget(self.toolbar_p)
462         self.layout_graphPress.removeWidget(self.canv_p)
463         self.toolbar_p = None
464         self.canv_p = None
465         self.serialArduino.force = []
466         self.serialArduino.dt_string = []
467         self.serialArduino.pressure = []
468         if(self.selPort != ''):
469             self.serialArduino.arduino.open()
470     except Exception as e:
471         print(e)
472         pass
473
474     # Initialize the ForceGraph
475     self.dataFrame = []
476     self.toolbar = None
477     # Initialize the pressure Graph
478     self.dataFrameP = []
479     self.toolbar_p = None
480
481     #This is for reset the ports.
482     self.port.puertos = []
483     self.CBpuertos.clear()
484
485     def gotoStartProcess(self):
486         self.tabWidget.setCurrentIndex(1)
487         self.resetProcess()
488
489
490     def gotoTabForce(self):
491         self.tabWidget.setCurrentIndex(2)
492
493     def startTimer(self):
494         self.timer = QTimer()
495         self.timer.start(1)
496         self.Button_StartCount.setEnabled(False)
497         self.Button_StopCount.setEnabled(True)
498         self.timer.timeout.connect(self.lcd_display)
499         self.dataFrame = []
500         self.dataFrameP = []
501
502         self.serialArduino.sendData("da")
503         time.sleep(1)
504         # Inicializamos el envío de datos
505         self.serialArduino.hilo1()
506         #Removemos los datos anteriores gráfica Fuerza
507         self.horizontalLayout_2.removeWidget(self.toolbar)
508         self.verticalLayout.removeWidget(self.canv)
509         #Removemos los datos anteriores gráfica Presión
510         self.layout_toolbar.removeWidget(self.toolbar_p)
511         self.layout_graphPress.removeWidget(self.canv_p)
512
513     def updateResultValues(self,duration,force):
514         self.lb_totalDuration.setText(f"{duration}")
515         self.lb_FzaMax.setText(f"{force}")

```

```

516
517
518 def endTimer(self):
519     self.timer.stop()
520     self.Button_StartCount.setEnabled(True)
521     self.Button_StopCount.setEnabled(False)
522     self.timer.deleteLater()
523
524     self.serialArduino.flag = True
525     data2ard = self.dataPressure(self.valSelPress)
526     value = "ca"+data2ard+ "off"
527     self.serialArduino.sendData(value)
528     self.serialArduino.closeSerial()
529     print("Muestreo::~\n")
530     print(self.serialArduino.frec)
531     frec = 100 / self.serialArduino.frec[0]
532     print("Frecuencia muestreo:::",frec,"datos x segundo")
533
534     self.temp2 = np.array(self.serialArduino.force)
535     dt2 = np.array(self.serialArduino.dt_string)
536     pressure = np.array(self.serialArduino.pressure)
537     # plt.plot(dt2,temp2)
538     # data = pd.DataFrame(self.temp2,columns=['Fuerza'],
539     # index=dt2)
540     df = None
541     if len(self.temp2) >0:
542         df = pd.DataFrame({'Fuerza':self.temp2,
543             'Presión':pressure},index=dt2)
544         self.dataFrame = pd.DataFrame({'Fuerza':self.temp2},index=dt2)
545
546         self.dataFrameP = pd.DataFrame({'Presión': pressure},index=dt2)
547
548     maxForce = self.dataFrame['Fuerza'].max()
549     maxDuration = np.amax(dt2)
550
551     self.updateResultValues(maxDuration,maxForce)
552
553     currentUser.saveData(df)
554     # Insertamos en la tabla
555     self.insertTable(self.serialArduino.force,self.serialArduino.pressure)
556     option = messagebox.askokcancel("Información","Desea guardar la información de la presente sesión?")
557     self.saveDBData(option)
558     self.val = 0
559
560     #This is the fuction relative to the main graphic
561     self.gotoTabForce()
562     self.updateForceG()
563
564     #This is the function relative to the 2nd main graph
565     self.updatePressureG()
566
567     #This is for reset the ports.
568     self.port.puertos = []
569     self.CBpuertos.clear()
570
571
572 def saveDBData(self,op):
573
574     if op == True:
575         self.sesiones +=1
576         currentUser.sesion = currentUser.sesion + 1
577         # for fuerza in self.serialArduino.force:
578         #     data = (fuerza,currentUser.sesion,self.fecha,currentUser.id)
579         #     self.dataInsertDB.append(data)
580
581         for i in range(len(self.serialArduino.force)):
582             auxFza = round(self.serialArduino.force[i],2)
583             auxP = round(self.serialArduino.pressure[i],2)
584             data = (auxFza,auxP,currentUser.sesion,self.fecha,currentUser.id)
585             self.dataInsertDB.append(data)
586
587         db = sqlite3.connect(db_path)

```

```

588     try:
589         cursor = db.cursor()
590         # para listas de listas necesitamos realizar un executemany:
591         # user_info = [nombre_usuario,apellido_p,apellido_m,
592         # correo,contrasena,edad,sexo,especialidad,organizacion,0]
593         query = ('INSERT INTO Datos (Fuerza,Presion,Sesion,' +
594                 'Fecha,Id_Usuario) VALUES (?,?,?,?)')
595         cursor.executemany(query,self.dataInsertDB)
596         cursor = db.cursor()
597         seconds = self.serialArduino.dt_string[-1]
598
599         params = [seconds,currentUser.sesion,self.fecha,currentUser.id]
600         query2 = ('INSERT INTO Duracion (Duracion,Sesion,' +
601                 'Fecha,Id_Usuario) VALUES (?,?,?,?)')
602         cursor.execute(query2,params)
603
604         query3 = (f'UPDATE usuario SET Sesion = {currentUser.sesion} where ID = {currentUser.id}')
605         cursor.execute(query3)
606         db.commit()
607         db.close()
608
609     except:
610         print("No se actualizo db")
611     else:
612         messagebox.showinfo("Datos",f" \nSesión día: {self.sesiones} \n"+
613                                f"Sesión General: {currentUser.sesion}")
614
615     def getHistoric(self):
616         db = sqlite3.connect(db_path)
617         cursor = db.cursor()
618         # query = 'SELECT Contraseña FROM USUARIO where Correo = \''+user+ '\''
619         # cursor.execute(query)
620         # Aquí prepararemos los queries para insertar en la tabla
621
622     try:
623         print(currentUser.id)
624
625         cursor.execute(f'SELECT * FROM datos where Id_Usuario = {currentUser.id}',())
626         data = cursor.fetchall()
627         db.commit()
628         self.tableRes.clearContents()
629         row = 0
630         self.tableRes.setRowCount(len(data))
631         for el in data:
632             # sesion
633             self.tableRes.setItem(row,0,QTableWidgetItem(str(el[3])))
634             # Fuerza
635             self.tableRes.setItem(row,1,QTableWidgetItem(str(el[1])))
636             # Presion
637             self.tableRes.setItem(row,2,QTableWidgetItem(str(el[2])))
638             row += 1
639
640         except:
641             print("No se pudo ejecutar query historico!")
642
643         db.close()
644
645     def ldc_display(self):
646         # self.timer.timeout.connect(self.ldc_display)
647
648         time = QDateTime.currentDateTime()
649         text = time.toString('ddd dd-MM-yyyy hh:mm:ss:zz')
650         # self.label_date.setText(text)
651         self.val = self.val + 1
652         # print(self.val)
653         self.lcdScreen.display(self.val)
654
655
656     # main
657     app = QApplication(sys.argv)
658     welcomeWindow= welcomeScreen()
659     widget = QStackedWidget()

```

```
660 widget.addWidget(welcomewindow)
661 # widget.setFixedHeight(550)
662 widget.resize(490,500)
663 # widget.setFixedWidth(490)
664 widget.show()
665 try:
666     | sys.exit(app.exec_())
667 except:
668     | print("Exit")
```

Apéndice VIII Programa calibración sensor de presión

```
1  # @Autor: Emmanuel Sánchez Ortiz
2  # Fecha: 22/11/22
3  # Nombre: Programa de calibración para sensor de presión.
4
5  import pandas as pd
6  import openpyxl as op
7
8  data = pd.read_excel("./Valores16mmadcx.xlsx")
9  arr = data['Values'].values.tolist()
10
11 orderedInfo = []
12 a=0
13
14 for data1 in arr:
15     for data2 in arr:
16         if data2 == data1:
17             a+=1
18             val = {
19                 "No": data1,
20                 "Times":a
21             }
22             orderedInfo.append(val)
23             a=0
24
25 ordenated = sorted(orderedInfo, key=lambda item : item['Times'],reverse=True)
26 valUnique = []
27
28 for value in ordenated:
29     if not value in valUnique:
30         valUnique.append(value)
31
32
33 book = op.Workbook()
34 sheet = book.active
35 b=1
36
37 sheet["A1"] = "Valor"
38
39 sheet["B1"] = "No."
40 for row in valUnique:
41     b+=1
42     sheet["A"+str(b)] = row["No"]
43     sheet["B"+str(b)] = row["Times"]
44
45 book.save("./datosOrd16mm.xlsx")
46
47
```

Anexos

Anexo 1 Glosario

LSB: (Less significant Bit) Bit menos significativo.

SPS: (Samples Per Second) Muestras por segundo.

FSR: (Full Scale Rate) Rango de escala completa

Payload: El código payload es un código de control generado a través de una línea tipo string, la cual se conforma de la siguiente estructura:

1. "CA"_Presión_Acción.
2. "BA"_Calibración
3. "DA"

De la anterior estructura el valor de la presión se da en 2 dígitos de tal forma que presiones inferiores al valor de 10 se les añade un 0.

En la acción encontramos el estado encendido (onn) y apagado (off) del insuflador.

Por otro lado, también a través del mismo código enviamos la instrucción "da", la cual indica la adquisición de datos.

ADC: (Analog-Digital Converter) Conversor Analógico digital.

Referencias

- [1] 3DFactoryMx. (s. f.). *PLLA - 3D Factory MX*. <https://impresion3d.3dfactory.mx/materials/753/>
- [2] Adafruit. (s. f.). *Adafruit_ADS1X15*. GitHub. Recuperado 7 de febrero de 2022, de https://github.com/adafruit/Adafruit_ADS1X15
- [3] Albacete, M. (2005). Historia de la cirugía laparoscópica y de la terapia mínimamente invasiva. En *Clínicas Urológicas de La Complutense* (pp. 15-44). https://historia.aeu.es/clasica/Docs/HISTORIA_DE_LA_CIRUGIA_LAPAROSCOPICA.pdf
- [4] Arduino. (s. f.). *Language Reference*. Arduino. <https://www.arduino.cc/reference/en/>
- [5] Bano, J., Hostettler, A., Nicolau, S. A., Cotin, S., Doignon, C., Wu, H. S., Huang, M. H., Soler, L., & Marescaux, J. (2012). *Simulation of Pneumoperitoneum for Laparoscopic Surgery Planning*. SpringerLink. https://link.springer.com/chapter/10.1007/978-3-642-33415-3_12
- [6] Bolton, W., Díaz, L. Á. L., & Giannetto, M. G. (2013). *Mecatrónica: sistemas de control electrónico en la ingeniería mecánica y eléctrica*. Alfaomega.
- [7] Circuit, E. (2020, 25 diciembre). *LM2596 3-A Step-Down Voltage Regulator*. Electronic Circuit. <https://www.elcircuit.com/p/lm2596-3-step-down-voltage-regulator.html>
- [8] Code First with Hala. (2020a, agosto 24). *PyQt5 QTableWidget tutorial: Load data, fill tables, format tables [Table Widget]* [Video]. YouTube. <https://www.youtube.com/watch?v=xL2NdSubiNY>
- [9] Code First with Hala. (2020b, octubre 18). *PyQt5 QTreeWidget tutorial: Load data, organize tree, access elements and parents [Tree Widget]* [Video]. YouTube. <https://www.youtube.com/watch?v=dgg0L7Qw3ko>
- [10] Del Valle, L. (2022, 13 enero). *ADS1115 convertidor analógico digital ADC para Arduino y ESP8266. Programar fácil con Arduino*. <https://programarfácil.com/blog/arduino-blog/ads1115-convertidor-analogico-digital-adc-arduino-esp8266/>
- [11] Electronica Avanzada. (2019a, octubre 21). *ADC Conversor analogico a digital* [Video]. YouTube. <https://www.youtube.com/watch?v=qDRgoVo6kkE>
- [12] Electronica Avanzada. (2019b, octubre 21). *ADC Conversor analogico a digital*. YouTube. <https://www.youtube.com/watch?v=qDRgoVo6kkE>
- [13] Electrónica FP. (2020, 18 noviembre). *Conversión analógico - digital. PARTE 1*. YouTube. <https://www.youtube.com/watch?v=5uz7VZ-LvcQ>
- [14] Electronilab. (s. f.). *Módulo LM2596 Convertidor de Voltaje DC-DC Reductor Buck 1.25V-35V*. <https://electronilab.co/tienda/modulo-lm2596-convertidor-de-voltaje-dc-dc-buck-1-25v-35v/>

- [15] Handelman, A., Schnaider, S., Schwartz-Ossad, A., Barkan, R., & Tepper, R. (2018). Computerized model for objectively evaluating cutting performance using a laparoscopic box trainer simulator. *Surgical Endoscopy*, 33(9), 2941-2950. <https://doi.org/10.1007/s00464-018-6598-x>
- [16] Harmon, C. (1959). EQUIPMENT AND INSTRUMENTS. *Analytical Chemistry*, 31(1), 93A-102A. <https://doi.org/10.1021/ac60145a799>
- [17] Horeman, T., Sun, S., Tuijthof, G. J., Jansen, F. W., Meijerink, J. W., & Dankelman, J. (2015). Design of a Box Trainer for Objective Assessment of Technical Skills in Single-port Surgery. *Journal of Surgical Education*, 72(4), 606-617. <https://doi.org/10.1016/j.jsurg.2015.02.002>
- [18] Hunter, J. (s. f.). *Matplotlib 3.6.3 documentation*. <https://matplotlib.org/stable/index.html>
- [19] iDEVELOPING. (2020, 2 marzo). *How to visualize (Plot 2D & 3D) in GUI PyQt5 & Matplotlib Txt file #tutorialpython #matplotlib #Pyqt* [Video]. YouTube. https://www.youtube.com/watch?v=XyCUuikw_3s
- [20] International Rectifier. (s. f.). IRL520NLPBF Datasheet (3/10 pages) IRF | Logic-Level Gate Drive. <https://html.alldatasheet.com/html-pdf/149084/IRF/IRL520NLPBF/186/3/IRL520NLPBF.html>
- [21] Javaux, A., Bouget, D., Gruijthuisen, C., Stoyanov, D., Vercauteren, T., Ourselin, S., Deprest, J., Denis, K., & Vander Poorten, E. (2018). A mixed-reality surgical trainer with comprehensive sensing for fetal laser minimally invasive surgery. *International Journal of Computer Assisted Radiology and Surgery*, 13(12), 1949-1957. <https://doi.org/10.1007/s11548-018-1822-7>
- [22] John Ortiz Ordoñez. (2019, 14 julio). *PyQt - Receta 27: Uso del Componente QTableWidgetItem* [Video]. YouTube. <https://www.youtube.com/watch?v=eq0Zl9DVjdc>
- [23] L. Ortiz Simón, J., Minor Martínez, A., Prado Coronado, N., & Ordorica Flores, R. (2012). Laparoscopic Trainer with Pneumoperitoneum. *Surgical Science*, 03(10), 499-502. <https://doi.org/10.4236/ss.2012.310099>
- [24] Laparoscopia: técnicas y vías de abordaje. (2006). *Progresos de Obstetricia y Ginecología*, 49(3), 159-166. [https://doi.org/10.1016/s0304-5013\(06\)72587-0](https://doi.org/10.1016/s0304-5013(06)72587-0)
- [25] Llamas, L. (2014, 16 abril). *Comunicación de Arduino con puerto serie*. Luis Llamas. <https://www.luisllamas.es/arduino-puerto-serie/>
- [26] Magno Efren. (2021a, enero 27). *Cómo graficar datos de arduino en Python*. YouTube. <https://www.youtube.com/watch?v=hHz4XGIUAI4>
- [27] Magno Efren. (2021b, mayo 8). *Control de LEDs con PySide2 / PyQt5 y Arduino* [Video]. YouTube. <https://www.youtube.com/watch?v=GctwWhW2RPo>
- [28] Magno Efren. (2021c, julio 14). *Graficas Matplotlib con PySide2/ PyQt5 y Qt Designer* [Video]. YouTube. <https://www.youtube.com/watch?v=AfEoGSbOSoU>

- [29] MEDUTCTeditorial. (2011, 10 junio). *Homemade laparoscopic simulators for surgical trainees*. YouTube. <https://www.youtube.com/watch?v=GUheFddh86A>
- [30] Microchip Technology. (s. f.). ATMEGA328P PDF, ATMEGA328P Description, ATMEGA328P Datasheet, ATMEGA328P view :: ALLDATASHEET :: <https://pdf1.alldatasheet.com/datasheet-pdf/view/1425005/MICROCHIP/ATMEGA328P.html>
- [31] Mohabir, P. K., & Coombs, A. V. (2023, 1 febrero). *Cirugía*. Manual MSD versión para público general. <https://www.msmanuals.com/es-mx/hogar/temas-especiales/cirug%C3%ADa/cirug%C3%ADa>
- [32] Nillahoot, N., Pillai, B. M., Sharma, B., Wilasrusmee, C., & Suthakorn, J. (2022). Interactive 3D Force/Torque Parameter Acquisition and Correlation Identification during Primary Trocar Insertion in Laparoscopic Abdominal Surgery: 5 Cases. *Sensors*, 22(22), 8970. <https://doi.org/10.3390/s22228970>
- [33] NumFOCUS, Inc. (s. f.). *User Guide — pandas 1.5.3 documentation*. pandas. https://pandas.pydata.org/docs/user_guide/index.html
- [34] NumPy Developers. (s. f.). *NumPy documentation — NumPy v1.24 Manual*. NumPy. <https://numpy.org/doc/stable/>
- [35] NXP Semiconductors. (2018, diciembre). *MP3V5050GP*. nxp. <https://www.nxp.com/docs/en/data-sheet/MP3V5050.pdf>
- [36] Parkhomenko, E., Yoon, R., Okhunov, Z., Patel, R. M., Dolan, B., Kaler, K., Schwartz, M. J., Shah, P. H., Bierwiler, H., Gamboa, A. J., Miano, R., Germani, S., Fabbro, D. D., Zordani, A., Micali, S., Kavoussi, L. R., Clayman, R. V., & Landman, J. (2019). Multi-institutional Evaluation of Producing and Testing a Novel 3D-Printed Laparoscopic Trainer. *Urology*, 124, 297-301. <https://doi.org/10.1016/j.urology.2018.06.034>
- [37] Punto Quirúrgico. (2018, 29 enero). El VERSAPORT TM Plus V2 Trocar. YouTube. <https://www.youtube.com/watch?v=JAY8X1c7tYA>
- [38] Python Software Foundation. (s. f.). *Documentation*. Python.org. <https://www.python.org/doc/>
- [39] Python Software Foundation. (2023, 9 febrero). *tkinter — Interface de Python para Tcl/Tk*. Python documentation. <https://docs.python.org/es/3/library/tkinter.html>
- [40] Roboticoss. (2020, 19 marzo). *Python y Arduino Interfaz Gráfica de Usuario (GUI) Trazado de datos de sensores en tiempo real [Vídeo]*. YouTube. <https://www.youtube.com/watch?v=ymDcflVSdxo>
- [41] Rovelo, P. (s. f.). *Cirugía de Mínima Invasión*. Médica Sur. https://www.medicasur.com.mx/es/ms/Cirugia_de_Minima_Invasion
- [42] Ruiz, A. C., Rodríguez, L. J. C., & Garcia, J. M. (2016). Evolución histórica de la cirugía laparoscópica. *Revista mexicana de cirugía endoscópica*, 17(2), 93-106. <https://www.medigraphic.com/cgi-bin/new/resumen.cgi?IDARTICULO=67138>

- [43] Saxena, A. K., & Höllwarth, M. E. (2008). *Essentials of Pediatric Endoscopic Surgery*. Springer Publishing.
- [44] Schrope, J., Olmanson, B., Fick, C., Motameni, C., Viratyosin, T., Miller, Z. D., Harmon, J., & Emerson, P. (2019). The SMART Trocar: Force, Deviation, and Impedance Sensing Trocar for Enhanced Laparoscopic Surgery. *2019 Design of Medical Devices Conference*. <https://doi.org/10.1115/dmd2019-3244>
- [45] SnapEDA. (2015, 15 julio). *ARDUINONANO*. <https://www.snapeda.com/parts/ARDUINONANO/Arduino/view-part/?ref=search>
- [46] SnapEDA. (2018, 5 marzo). *IRL520NPBF*. <https://www.snapeda.com/parts/IRL520NPBF/International+Rectifier/view-part/?ref=search>
- [47] SnapEDA. (2019, 19 diciembre). *MCP6002-I/P*. <https://www.snapeda.com/parts/MCP6002-I/P/Microchip/view-part/?ref=search>
- [48] SnapEDA & michaelw. (2015, 15 enero). *LF33CV*. SnapEDA. <https://www.snapeda.com/parts/LF33CV/STMicroelectronics/view-part/?ref=search>
- [49] Sociedad Argentina de Cirugia Digestiva. (2022, 4 enero). *Tomo I*. SACD. <https://sacd.org.ar/tomo-i/>
- [50] Ssj6. (2017, 5 julio). *PyQt: Show Data Into QTableWidgetItem From Database*. [Vídeo]. YouTube. <https://www.youtube.com/watch?v=l2OoXj1Z2hM>
- [51] STMicroelectronics. (2017, 26 junio). *LFXX: Very low drop voltage regulator with inhibit*. STlife.augmented. <https://www.st.com/en/power-management/lfxx.html>
- [52] Sun, J., & Tadano, K. (2017). Force Characteristics of Trocar Insertion Abdomen in Laparoscopic Surgery. *World Congress on Electrical Engineering and Computer Systems and Science*. <https://doi.org/10.11159/icbes17.119>
- [53] Sun, J., & Tadano, K. (2019). A Novel Device for Safe Trocar Insertion in Laparoscopic Surgery Based on the Insertion Force Characteristics. *International Journal of Bioscience, Biochemistry and Bioinformatics*, 9(1), 1-8. <https://doi.org/10.17706/ijbbb.2019.9.1.1-8>
- [54] Tekscan. (s. f.). *FlexiForce A201 Sensor*. <https://www.tekscan.com/products-solutions/force-sensors/a201?tab=electronics>
- [55] Texas Instruments. (2013, mayo). *LM79 data sheet, product information and support*. <https://www.ti.com/product/LM79?keyMatch=LM79>
- [56] Texas Instruments. (2018, enero). *ADS111x Ultra-Small*. https://www.ti.com/lit/ds/symlink/ads1115.pdf?ts=1665963146191&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FADS1115
- [57] Texas Instruments. (2021, abril). *LM2596 SIMPLE SWITCHER*. <https://www.ti.com/lit/ds/symlink/lm2596.pdf>
- [58] The Qt Company. (s. f.). *Qt Documentation*. Qt.io. <https://doc.qt.io/>

- [59] UNIT ELECTRONICS. (s. f.-a). *Fuente de Alimentación 5V 10A*. <https://uelectronics.com/producto/fuente-de-alimentacion-5v-10a/>
- [60] UNIT ELECTRONICS. (s. f.-b). *LM2596 Regulador Step Down 25W 3A*. <https://uelectronics.com/producto/modulo-regulador-ajustable-lm2596-dc-dc-step-down-3a-1-25-30v/>
- [61] UNIT ELECTRONICS. (s. f.-c). *Módulo ADS1115 ADC Amplificador de Ganancia Programable*. <https://uelectronics.com/producto/modulo-ads1115-adc/>
- [62] UNIT ELECTRONICS. (s. f.-d). *Nano 3.0 con Cable / Sin cable / Shield Compatible con Arduino*. <https://uelectronics.com/producto/arduino-nano-3-0-con-cable-sin-cable-shield/>
- [63] Universidad Nacional Mayor de San Marcos (Lima). Facultad de Medicina. Escuela Académico Profesional de Medicina Humana. Departamento Académico de Cirugía, & Poggi Machuca, L. (1999). Cirugía Laparoscópica. En *Cirugía: I cirugía general*. UNMSM. https://sisbib.unmsm.edu.pe/bibvirtual/libros/medicina/cirugia/Tomo_I/Cap_07_cirug%C3%ADa%20Laparosc%C3%B3pica.htm
- [64] Vilos, G. A. (2008, 15 julio). *Three simple steps during closed laparoscopic entry may minimize major injuries*. SpringerLink. <https://link.springer.com/article/10.1007/s00464-008-0060-4>