

Escuela Superior Politécnica del Litoral

Facultad de Ingeniería en Mecánica y Ciencias de la Producción

Diseño y prototipado de sistemas para teleconsulta usando tarjetas de desarrollo de bajo costo con fotoplestimografía

Proyecto Integrador

Previo la obtención del Título de:

Ingenieros en Mecatrónica

Presentado por:

Benjamín Enrique Escudero Cevallos

Nicole Dayanne Santibáñez Rivera

Guayaquil - Ecuador

Año: 2023

Dedicatoria

Dedicamos este trabajo a nuestras familias, quienes con su apoyo y amor nos han ayudado a llevar adelante este proyecto donde se ven reflejados sus esfuerzos constantes del pasado y presente.

Agradecimientos

Agradecemos a los profesores que han sido nuestra guía en el proceso de aprendizaje y del desarrollo del proyecto.

De la misma manera, extendemos nuestros agradecimientos a los buenos amigos que hemos realizado dentro de nuestro tiempo en la carrera y nos han acompañado durante todo el proceso.

Declaración expresa

“Los derechos de titularidad y explotación, nos corresponde conforme al reglamento de propiedad intelectual de la institución; *Benjamín Enrique Escudero Cevallos* y *Nicole Dayanne Santibáñez Rivera* damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual”.



Benjamín Enrique
Escudero Cevallos



Nicole Dayanne
Santibáñez Rivera

Evaluadores

Efraín Terán, M.Sc.

Profesor de la materia

Ronald Solís, M.Sc.

Profesor tutor

Resumen

En este trabajo, se presenta el desarrollo de un sistema de telemedicina, que emplea el sensor TCRT5000 para medir la frecuencia cardíaca y el sensor LM35 para medir la temperatura corporal, en conjunto con la placa Raspberry Pi Pico W. La metodología involucró la captura de la onda de pulso cardíaco a través de la fotoplestimografía. Para ello fue necesario implementar una etapa de ganancia para amplificar la señal obtenida, además se incorporó un filtro pasa alto y un filtro pasa bajo para optimizar la señal del fototransistor. Se realizó un código en micropython cargado a la Raspberry para procesar los datos y calcular la frecuencia cardíaca. La conexión al servidor se establece para enviar los datos a una plataforma web creada con PHP y XAMPP, conectada a una base de datos MySQL. Para la página web, se diseñó una interfaz para los pacientes y otra para que los médicos puedan visualizar el registro de los datos biométricos de sus pacientes. Este sistema de telemedicina representa un avance en la atención médica remota, fusionando la tecnología de sensores con la plataforma web para brindar monitoreo y acceso a la información de salud a distancia.

Palabras Clave: Fotoplestimografía, MySQL, telemedicina, TCRT5000 y LM35

Abstract

This paper presents the development of a telemedicine system, which uses the TCRT5000 sensor to measure heart rate and the LM35 sensor to measure body temperature, in combination with the Raspberry Pi Pico W. The methodology involved the capture of the cardiac pulse wave through photoplethysmography. To do this, it was necessary to implement a gain stage to amplify the obtained signal, also a high-pass filter and a low-pass filter were incorporated to optimize the phototransistor signal. The Raspberry was programmed using micropython to process the measured data and calculate the heart rate. The connection to the server was established to send the data to a web platform created with PHP and XAMPP that are connected to a MySQL database. For the web page, an interface was designed for patients and another for the doctors to be able to visualize the biometric data record of their patients. This telemedicine system represents a breakthrough in remote medical care, merging sensor technology with a web platform to provide remote monitoring and access to health information.

Keywords: *Photoplethysmography, MySQL, telemedicine, TCRT5000 and LM35*

Índice general

Resumen.....	I
<i>Abstract</i>	II
Índice general.....	III
Abreviaturas.....	VIII
Simbología.....	IX
Índice de figuras.....	X
Índice de tablas.....	XIV
Capítulo 1.....	1
1. Introducción.....	2
1.2 Descripción del problema.....	3
1.3 Justificación del problema.....	3
1.4 Objetivos.....	7
1.4.1 Objetivo general.....	7
1.4.2 Objetivos específicos.....	8
1.5 Marco teórico.....	8
1.5.1 Sistema cardiovascular.....	8
1.5.2 Corazón.....	9
1.5.3 Ciclo cardíaco.....	10

1.5.4 Presión arterial	12
1.5.5 Frecuencia cardíaca.....	12
1.5.6 Fotoplestimografía	14
1.5.8 Desarrollo de frontend.....	16
1.5.9 Desarrollo de backend.....	17
1.5.10 Base de datos.....	17
1.5.11 Base de datos Relacionales	18
1.6 Estado del arte.....	18
1.6.1 PPG EduKit: Un sistema ajustable de evaluación por fotoplestimografía ajustable para actividades educativas.....	18
1.6.2 Dispositivos portátiles.....	19
1.6.3 Kit de sensores de pulso cardíaco	19
1.6.4 Desarrollo de un sistema web de citas médicas usando herramientas open source y sensores biométricos de signos vitales.....	20
1.6.5 Diseño e implementación de un glucómetro no invasivo basado en la ley de Lambert-beer y longitud de onda cercana al infrarrojo (NIR), con interfaz de comunicación bluetooth a dispositivos con sistema operativo Android	21
1.6.7 Pulsera inalámbrica para monitorización del pulso cardíaco.....	22
Capítulo 2.....	24
2. Metodología	25

2.1.1	Requerimientos	25
2.1.2	Selección de la alternativa de solución	25
2.1.2.1	Microcontrolador:	26
2.1.2.2	Interfaz de desarrollo:	26
2.1.4	Criterios para las alternativas del microcontrolador:	28
2.2	Proceso de diseño.....	29
2.2.1	Diseño conceptual.....	30
2.2.2	Diseño electrónico	31
2.2.2.1	Selección de los sensores y módulos electrónicos	31
2.2.2.2	Selección de microcontrolador.....	34
2.2.2.3	Etapas de acondicionamiento	34
2.2.2.3.1	Conexión del sensor TCRT5000.....	35
2.2.2.3.2	Nueva referencia de voltaje más un seguidor de tensión	36
2.2.2.3.3	Amplificador inversor	38
2.2.2.3.4	Etapas de filtrado.....	41
2.2.2.4	Conexión del sensor LM35.....	45
2.2.2.5	Esquemático con las conexiones realizadas del circuito.....	46
2.2.2.5	Etapas de digitalización.....	47
2.2.2.6	Diseño de programación	48

2.2.2.6.1 Activación del circuito.....	48
2.2.2.6.2 Visualización de la onda cardíaca.....	49
2.2.2.6.3 Ingreso del número de cédula del paciente al Rasberry pi pico.....	50
2.2.2.6.4 Cálculo de la frecuencia cardíaca	51
2.2.2.6.5 Cálculo de la temperatura corporal	54
2.2.2.6.6 Interfaz web.....	54
2.2.2.6.5 Servidor web	55
2.2.2.6.6 Comunicación por wifi hacia un servidor web	56
2.2.2.6.7 Base de datos.....	57
Capítulo 3.....	64
3. Resultados y análisis	65
3.1.1 Electrónica.....	65
3.1.1.1 Análisis de resultados	69
3.1.1.2 Diseño PCB.....	72
3.1.2 Programación	73
3.1.2.1 Interfaz Web.....	73
3.1.2.2 Análisis de resultados	84
3.1.3 Cotización de rubros	84
Capítulo 4.....	86

<i>4.1 Conclusiones y recomendaciones</i>	87
4.1.1 Conclusiones	87
4.1.2 Recomendaciones	88
Referencias.....	89
Apéndices.....	93

Abreviaturas

ECV	Enfermedades Cardiovasculares
MSP	Ministerio de Salud Pública
INEC	Instituto Nacional de Estadística y Censos
OMS	Organización Mundial de la Salud
PPG	Fotoplestimografía
IoT	Internet de las cosas
PHP	Preprocesador de hipertexto
HTTP	Protocolo de transferencia de hipertexto
CSS	Hojas de estilo en cascada
MYSQL	Lenguaje estructurado de consulta

Simbología

lpm	Latidos por minuto
mA	Miliamperios
nm	Nanómetro
V	Voltaje
mV	Milivoltios
Hz	Hercio
μ F	Microfaradios
KB	Kilobytes
$^{\circ}$ C	Grados Celsius
ms	Milisegundos

Índice de figuras

Figura 1.1 División territorial de establecimientos médicos en Ecuador	5
Figura 1.2 Sistema cardiovascular del ser humano	9
Figura 1.3 Partes del corazón	10
Figura 1.4 El ciclo cardíaco	11
Figura 1.5 Presión arterial.....	12
Figura 1.6 Representación gráfica del pulso cardíaco	13
Figura 1.7 Fotoplestimografía por transmisión	15
Figura 1.8 Fotoplestimografía por reflexión	15
Figura 1.9 Pulsioxímetro.....	16
Figura 1.10 Modelo físico PPG Edukit.....	19
Figura 1.11 Prototipo de un glucómetro no invasivo realizado	22
Figura 1.12 Pulsera inalámbrica para monitorización del pulso cardiaco	23
Figura 2.1 Diagrama del proceso de diseño.....	29
Figura 2.2 Boceto de la solución.....	30
Figura 2.3 Sensor TCRT5000.....	31
Figura 2.4 Sensor LM35	32
Figura 2.5 Pantalla OLED SSD1306	33
Figura 2.6 Teclado matricial	33
Figura 2.7 Raspberry pi pico w.....	34
Figura 2.8 Diagrama de conexión del sensor TCRT5000 en Proteus.....	35
Figura 2.9 Referencia de voltaje en Proteus	36

Figura 2.10 LM324N Configuración de los pines	37
Figura 2.11 Amplificador Inversor	38
Figura 2.12 Comportamiento obtenido con voltaje de prueba.....	40
Figura 2.13 Osciloscopio	40
Figura 2.14 Filtro pasa altos - pasivo.....	42
Figura 2.15 Filtro pasa alto RC pasivo más seguidor de voltaje en Proteus.....	42
Figura 2.16 Filtro pasa bajos – pasivo	43
Figura 2.17 Filtro pasa bajos - activo.....	44
Figura 2.18 Filtro pasa bajo en Proteus	44
Figura 2.19 Diagrama del sensor LM35	45
Figura 2.20 Conexiones eléctricas del sistema electrónico descrito.....	46
Figura 2.21 Digitalización de los datos analógicos	48
Figura 2.22 Código de activación del circuito	49
Figura 2.23 Código de la pantalla OLED	50
Figura 2.24 Código para registrar el número de cédula.....	50
Figura 2.25 Onda senoidal	51
Figura 2.26 Variables usadas para calcular la frecuencia cardíaca.....	52
Figura 2.27 Código para calcular la frecuencia cardíaca.....	53
Figura 2.28 Diagrama de flujo de la interfaz web	54
Figura 2.29 Base de datos	55
Figura 2.30 Código para realizar el envío de datos al servidor	56
Figura 2.31 Forma del tipo de dato enviado con clave y valor.....	57
Figura 2.32 Código POST.....	57

Figura 2.33 Diagrama de entidad del Paciente	58
Figura 2.34 Diagrama de entidad del Doctor.....	58
Figura 2.35 Diagrama de entidad Sensordata	59
Figura 2.36 Diagrama de entidad Preguntas	60
Figura 2.37 Diagrama de entidad Indicaciones.....	62
Figura 2.38 Diagrama de relación.....	62
Figura 3.1 Prueba del dispositivo electrónico.....	65
Figura 3.2 Toma de datos con el dispositivo realizado.....	66
Figura 3.3 Frecuencia cardíaca obtenida usando un sensor digital.....	67
Figura 3.4 Datos obtenidos usando el dispositivo	68
Figura 3.5 Visualización de datos obtenidos en la página web	69
Figura 3.6 Comparación entre el sensor TCRT5000 y el sensor digital para medir la frecuencia cardíaca en estado de reposo.....	70
Figura 3.7 Comparación entre el sensor TCRT5000 y el sensor digital para medir la frecuencia cardíaca luego de haber realizado una actividad.....	71
Figura 3.8 Modelo 3D del diseño PCB realizado	72
Figura 3.9 Página principal de la interfaz web	73
Figura 3.10 Inicio de sesión Pacientes.....	74
Figura 3.11 Registro del paciente	75
Figura 3.12 Tabla MySQL pacd	75
Figura 3.13 Inicio de sesión Pacientes ya una vez registrado.....	76
Figura 3.14 Sesión ya iniciada del paciente.....	77
Figura 3.15 Formulario médico	78

Figura 3.16	Tabla MySQL preguntas	79
Figura 3.17	Ventana DatosTiemporeal	79
Figura 3.18	Inicio de sesión Doctores.....	80
Figura 3.19	Sesión ya iniciada del doctor	81
Figura 3.20	Vista terminada de la ventana sesión iniciada del doctor.....	82

Índice de tablas

Tabla 1.1 Porcentaje de establecimientos médicos por provincia	5
Tabla 1.2 Porcentaje de consultas de prevención realizadas entre 2018 y 2020	7
Tabla 1.3 Valores de frecuencia cardíaca	13
Tabla 2.1 Matriz de decisión de la interfaz de desarrollo	27
Tabla 2.2 Matriz de decisión de las alternativas del microcontrolador	28
Tabla 3.1 Comparación entre el sensor TCRT5000 y el sensor digital para medir la frecuencia cardíaca en estado de reposo.....	70
Tabla 3.2 Comparación entre el sensor TCRT5000 y el sensor digital para medir la frecuencia cardíaca luego de haber realizado una actividad.....	71
Tabla 3.3 Evaluación de la interfaz web por varios usuarios	83
Tabla 3.4 Cotización de Rubros.....	84

Capítulo 1

1. Introducción

En Ecuador, una de las principales razones de muerte son las enfermedades cardiovasculares (ECV). Estas enfermedades pueden implicar un conjunto de problemas del corazón y vasos sanguíneos, entre las que la hipertensión e insuficiencia cardíaca son las más mortales. La OMS responsabiliza a las ECV de causar altas tasa de mortalidad. En el 2015 murieron 17.7 millones de personas por estos trastornos, lo que representa el 31% de las muertes registrados a nivel global. Es importante señalar que en los países de ingresos bajos y medios son los que muestran una mayor prevalencia de ECV. Las principales causas de muerte en países de escasos recursos son por la falta de atención médica primaria para la detección anticipada y el correcto tratamiento preventivo de usuarios con factores de riesgo. [1]

Según un reporte realizado por el diario El Comercio, en el Ministerio de Salud Pública (MSP) se realiza un promedio anual de 247 000 consultas. Así mismo, se ejecutó 1.5 millones de consultas subsecuentes entre el 2018 y el 2020 en relación con enfermedades cardiovasculares. [2]

La telemedicina es una práctica que aplica la tecnología digital para brindar servicios médicos. Su objetivo es ofrecer alternativas eficientes para la comunicación entre médico y paciente. Dicha práctica se ha desarrollado y aplicado desde el surgimiento de la comunicación electrónica. La OMS define a la telemedicina como un servicio adicional de atención sanitaria, en el que, el factor distancia tiene un valor crítico [3]. Existen diversos beneficios en el uso de la telemedicina. Esto se debe a que brinda una mayor eficiencia, accesibilidad y progreso a la calidad del servicio. Por estos motivos, el MSP busca expandir la aplicación de estas tecnologías a nivel nacional. [4]

1.2 Descripción del problema

En la actualidad, el servicio de consulta médica brindado por el MSP no es eficiente. Es más, muchas muertes son ocasionadas por una mala calidad en la atención y la incapacidad de ofrecer con rapidez la solución del problema. Además, los factores tiempo y distancia son un impedimento en el cuidado de la salud. Existen personas que no cuentan con los medios para acudir a un hospital cercano. En estos casos, se corre el riesgo de que ciertas dolencias se conviertan en enfermedades crónicas, perjudicando su calidad de vida. [5]

En las últimas décadas, el uso de la tecnología en la medicina ha evolucionado para traspasar los límites físicos y anticipar enfermedades crónicas. Siguiendo esta línea, el presente proyecto se enfoca en diseñar un sistema de atención médica que emplea una placa de desarrollo e incluye sensores que miden datos biométricos utilizando fotoplestimografía. Los datos se visualizan a través de una interfaz web, para realizar un chequeo del estado de salud del paciente y anticipar anomalías.

1.3 Justificación del problema

La integración de la fotoplestimografía, la electrónica y la programación brindan una solución a la problemática, innovando en la atención médica a los pacientes. La fotoplestimografía monitorea de forma no invasiva signos vitales, como frecuencia cardíaca, frecuencia respiratoria, nivel de glucosa y saturación de oxígeno. Por lo que, esta tecnología aplicada a la teleconsulta médica beneficia tanto a los médicos como a los pacientes.

Por un lado, los médicos registran los signos de su paciente, analizando su progreso a través del tiempo. Por otro, el paciente se beneficia al realizarse un chequeo médico de forma no invasiva

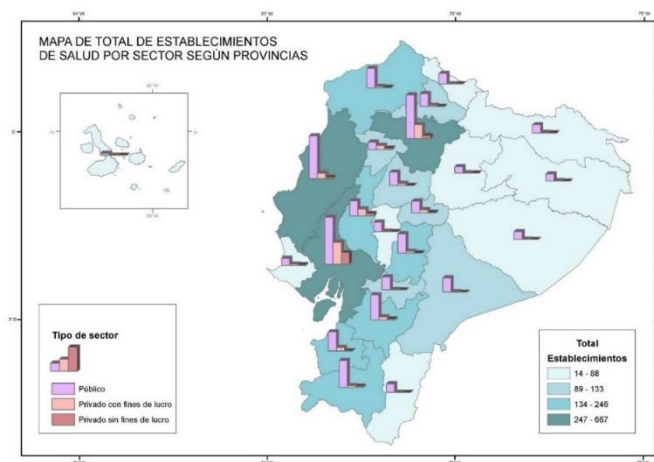
y remota. Muchas personas, especialmente en zonas rurales, no tienen cerca un establecimiento médico. Otras simplemente no tienen el tiempo suficiente para movilizarse. [6]

En Ecuador el sistema de salud se divide en 2 sectores, el público y el privado. El sector privado, a su vez, se divide en sector privado con fines de lucro y sector privado sin fines de lucro. El Instituto Ecuatoriano de Seguridad Social (IEES) brinda servicios de salud a la población asalariada, que registra aportes a este organismo. Además, otras entidades como el MSP y los municipios de cada cantón brindan servicios de salud a la población en general. Es importante señalar que, la población que utiliza servicios privados de salud posee ingresos medios y altos.

Según los datos del 2022, recopilados y publicados por el INEC, de un total de 4136 establecimientos médicos el 79,8% pertenece al sector público, el 15,01% al sector privado con fines de lucro y el 5,17% al sector privado sin fines de lucro. La **Figura 1.1** muestra la división territorial de estos establecimientos. Adicionalmente la **Tabla 1.1** presenta los porcentajes de cada tipo de establecimiento por provincia. Se resalta que existen provincias con amplia extensión territorial que presentan baja presencia de establecimientos médicos. [7]

Figura 1.1

División territorial de establecimientos médicos en Ecuador [7]

**Tabla 1.1**

Porcentaje de establecimientos médicos por provincia

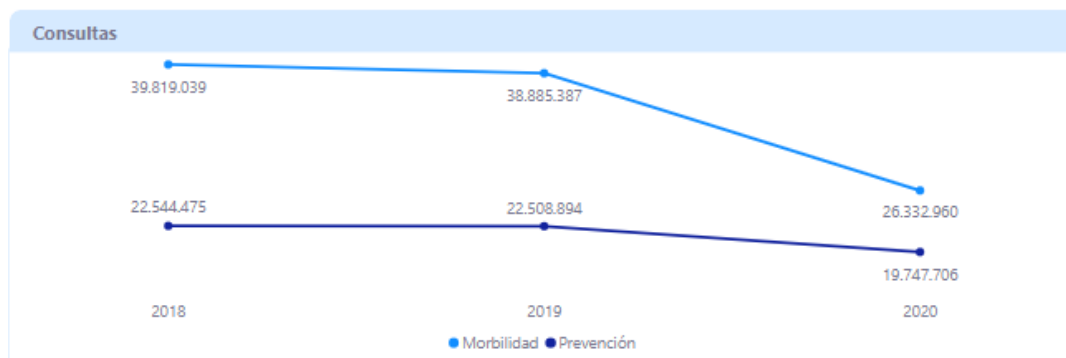
Provincia	Establecimientos
Guayas	16,13%
Pichincha	12,04%
Manabí	10,08%
Loja	5,95%
Azuay	5,92%
Los Ríos	4,79%
El Oro	4,64%

Chimborazo	4,38%
Esmeraldas	4,23%
Cotopaxi	3,22%
Tungurahua	3,05%
Morona Santiago	2,93%
Imbabura	2,90%
Santo Domingo de los Tsáchilas	2,71%
Cañar	2,66%
Carchi	2,13%
Bolívar	2,10%
Santa Elena	1,89%
Zamora Chinchipe	1,74%
Sucumbíos	1,67%
Pastaza	1,64%
Orellana	1,57%
Napo	1,31%
Galápagos	0,34%

Entre los años 2018 y 2020, disminuyó la cantidad de consultas de prevención. Se pasó de 22544475 en 2018 a 19747706 en 2020. De estas consultas de prevención, 14877606 se desarrollaron en el área urbana y el valor restante en el área rural. [7]

Tabla 1.2

Porcentaje de consultas de prevención realizadas entre 2018 y 2020



El INEC establece que, en 2022, el 60,4% de los hogares en Ecuador contaban con acceso a internet y el 69,7% de los individuos utilizaban este servicio. En el área urbana, el 69,7% de la población usa internet y en el área rural, el 50,5%. Esto implica que más de la mitad de la población serían beneficiarios de un sistema de teleconsulta.

Además, el desarrollo de este sistema hará posible la detección temprana de posibles anomalías cardiovasculares, un diagnóstico temprano permite prevenir el agravamiento de la situación del paciente y mejorar su calidad de vida. [7]

1.4 Objetivos

1.4.1 Objetivo general

- Diseñar un sistema de atención médica a distancia que permita hacer uso de un dispositivo electrónico para realizar mediciones de datos biométricos las cuales podrán ser visualizados por el doctor mediante una plataforma web.

1.4.2 Objetivos específicos

- Diseñar un dispositivo electrónico que incorpore fotoplestimografía.
- Crear dos interfaces para paciente y médico que posean conexión a una base de datos para registrar y presentar la información recolectada por cada paciente, además de realizar el agendamiento de cita.
- Realizar una prueba conceptual sobre el sistema de telemedicina involucrando el diseño electrónico con la interfaz web.

1.5 Marco teórico

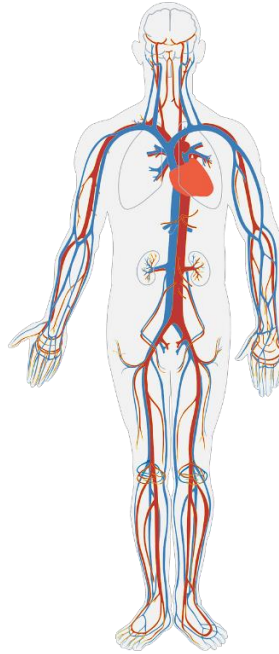
1.5.1 Sistema cardiovascular

Este está conformado por un conjunto de conductores nombrados vasos sanguíneos conocidos como transportadores de sangre, la cual es impulsada por el corazón.

El corazón se encuentra siempre activo, ocasionando que la sangre este periódicamente en circulación a través de los vasos sanguíneos. El sistema cardiovascular se encarga de llevar las sustancias necesarias a las células: oxígeno y nutrientes, además tiene la función de recoger los productos de desecho del metabolismo para luego ser enviadas a sectores que serán procesadas o eliminadas. [8]

Figura 1.2

Sistema cardiovascular del ser humano [9]



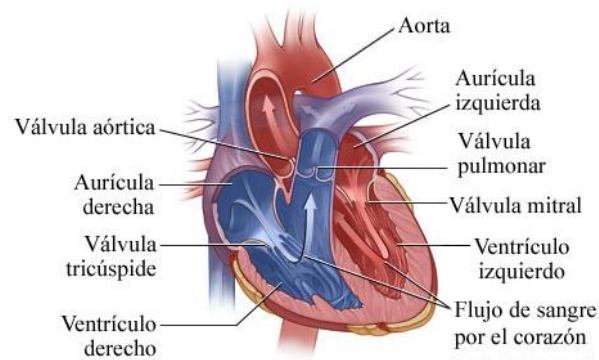
1.5.2 Corazón

Es el órgano más importante de los seres vivos, es denominado el motor del sistema cardiovascular, está separada en 2 hemisferios: derecho e izquierdo. El lado derecho toma la sangre de las venas, la cual tiene pocas cantidades de oxígeno para luego ser enviada a los pulmones, mientras que el lado izquierdo recibe la sangre de los pulmones con un alto grado de oxígeno y la bombea hacia las arterias.

El flujo de movimiento de la sangre va en una sola dirección recorriendo cuatro partes del corazón llamada cámaras, estas son: aurícula derecha, ventrículo derecho, aurícula y ventrículo izquierdos. [10]

Figura 1.3

Partes del corazón [10]



1.5.3 Ciclo cardíaco

Es el evento producido por la contracción y relajación de los ventrículos y aurículas del corazón, cada ciclo representa un latido, está compuesto por dos fases:

- **Diástole:** Ocurre cuando las cámaras del corazón están en estado de relajación y reciben sangre de las venas.
- **Sístoles:** Las cámaras del corazón están en estado de contracción y bombea la sangre hacia las arterias. [11]

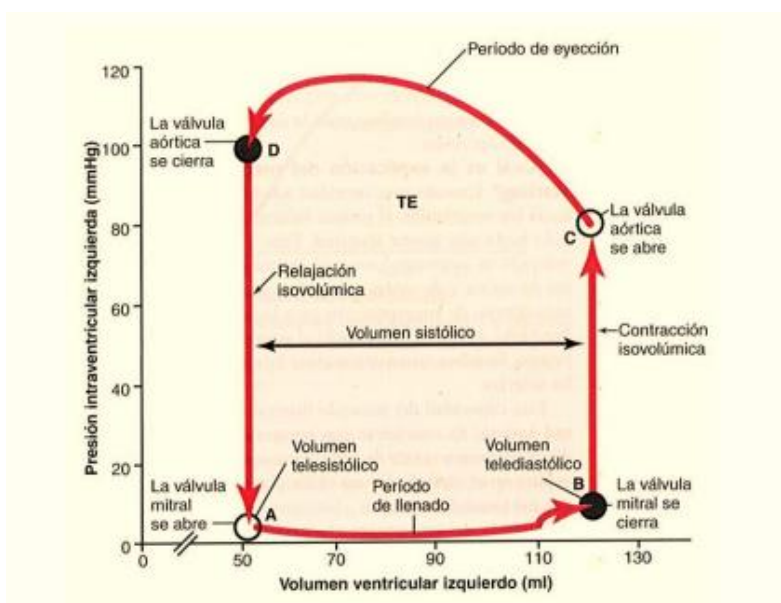
El ciclo cardíaco cuenta con tres principales etapas durante sus dos fases:

- **Sístole auricular:** Al principio del ciclo un estímulo de potencial emitido ocasiona que las aurículas se contraigan, ocasionando que se libere la sangre almacenada en los ventrículos y al ser completamente expulsada, se cierran las válvulas de comunicación entre las aurículas y ventrículos.

- **Sístoles ventriculares:** Primero se realiza la contracción ventricular isovolumétrica, los ventrículos comienzan a contraerse y luego se produce una inyección, por la diferencia de presión entre la arterial y la ventricular, los conductos son abiertos, permitiendo el paso de la sangre hacia el sistema circulatorio.
- **Diástole cardíaca completa:** Se subdivide en dos etapas, la relajación ventricular isovolumétrica y el llenado auricular pasivo. Durante la relajación ventricular los ventrículos se relajan y la presión sometida sobre ellos es descendida, en la siguiente etapa las aurículas estarán ocupadas por sangre, al haber un aumento de presión en las aurículas en relación con los ventrículos relajados, se abrirán las válvulas mitral y tricúspide para finalmente reiniciar el ciclo. [8]

Figura 1.4

El ciclo cardíaco



Nota: Gráfico obtenida de la relación volumen y presión ventricular en cada etapa del ciclo cardíaco. [12]

1.5.4 Presión arterial

La sangre es bombeada por el corazón, dicha fuerza es conocida como presión arterial, empuja la sangre sobre las paredes de las arterias la cual es transmitida a todas las arterias periféricas. [13]

Figura 1.5

Presión arterial



1.5.5 Frecuencia cardíaca

Es la cantidad de veces que el corazón se contrae, está representada por latidos por minutos (lpm), es un valor que se obtiene al medir el pulso arterial durante un intervalo de tiempo.

El promedio de pulsaciones de una persona depende de su edad, un individuo mayor de edad que se encuentra en reposo es de sesenta y cien latidos por minuto, este valor varía según la edad ya que el corazón se va envejeciendo con el paso de tiempo.

En la Tabla 1.3 se observará el rango de frecuencia cardíaca normal en reposo con respecto a la edad. [14]

Tabla 1.3

Valores de frecuencia cardíaca

Edad	Frecuencia cardíaca
0-1 mes	70-190 lpm
1-11 meses	80-160 lpm
1-2 años	80-130 lpm
3-4 años	80-120 lpm
5-6 años	75-115 lpm
7-9 años	70-110 lpm
10 años para adelante	60-100 lpm

Figura 1.6

Representación gráfica del pulso cardíaco



1.5.6 Fotoplestimografía

La fotoplestimografía (PPG) es una técnica médica no invasiva utilizada para obtener mediciones de la función endotelial, fue inventada por Hertzman en 1937 cuando propuso que estas representaban las variaciones del cambio de volumen sanguíneo. El método consiste en el uso de un haz de luz para obtener el volumen de los vasos subcutáneos.

El tejido humano comúnmente no permite el paso de la luz, sin embargo, la mayoría de las zonas blandas son excepciones, estos pueden recibir y reflejar radiación al ser expuestas a una fuente de luz. La PPG utiliza este método para inducir una luz infrarroja que traspasa el tejido, la cual será reflejada y absorbida por la sangre en dicha zona de medición y según esta cantidad de luz se podrá aproximar el volumen sanguíneo. Existen diversos factores que alteran la intensidad de luz variante en el tiempo, el cúmulo de sangre en la zona medida y la longitud de onda de la luz. [15]

La onda obtenida al realizar la PPG es descrita por sus dos componentes:

- **AC:** Señal pulsátil o variable que representa los cambios de luz absorbidas en el tiempo debido a las alteraciones de volumen sanguíneo.
- **CC:** Señal constante que representa el volumen de sangre constante.

La PPG tiene dos tipos de configuraciones: transmisión y reflexión, la diferencia es el cambio de posición del fotodiodo receptor, en el modo de transmisión el fotodiodo emisor y receptor están en posiciones opuestas, mientras que en la reflexión ambos están del mismo lado, por tal razón la señal obtenida usando transmisión es más fuerte en comparación a la reflexión, durante el camino de luz desde el emisor y receptor, se encuentra mayores interferencias

provocando la atenuación de la luz, además se tiende a obtener resultados más imprecisos y con mayor dificultad de acondicionamiento a la señal de salida. [15]

Figura 1.7

Fotoplestimografía por transmisión [9]

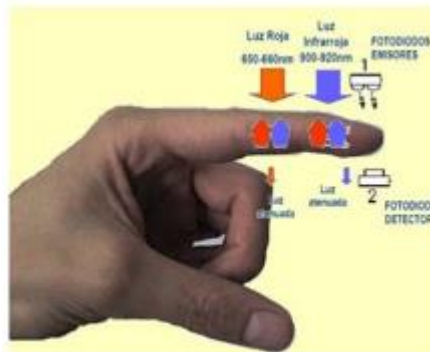
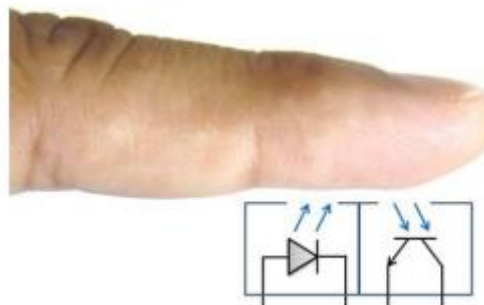


Figura 1.8

Fotoplestimografía por reflexión [16]



Un ejemplo de PPG es el pulsioxímetro, es un dispositivo el cual se coloca sobre un dedo de la mano, en su interior se encuentra un fotodiodo que envía luces rojas e infrarrojas las cuales pasan por un fotodetector y según la longitud de onda provocada por la sangre arterial sin tomar

en consideración la sangre venosa, huesos, piel, músculos y grasa, se calcula la saturación de oxígeno en la sangre. El tipo de configuración es por transmisión y es el más comúnmente usado para esta aplicación. [17]

Figura 1.9

Pulsioxímetro



1.5.8 Desarrollo de *frontend*

El desarrollo de un sistema web se divide en dos grupos, el desarrollo del *frontend* y el desarrollo del *backend*. Se compone de todo lo que es presentado al usuario, en esta sección se brinda estilo a la página y, se organiza la forma en la que será presentada la información al usuario final. Durante el desarrollo del *frontend* se debe tomar en cuenta la estética de los componentes, así como su ubicación que permita un uso fácil e intuitivo para brindar una experiencia de usuario agradable. Las diferentes tecnologías y lenguajes para el desarrollo del *frontend* se pueden englobar en los siguientes elementos HTML, CSS y JavaScript. [18]

- **HTML:** Es un lenguaje de marcado que brinda a los navegadores web las instrucciones para estructurar las páginas web definiendo el texto y demás elementos como imágenes, listas o videos. [18]
- **CSS:** Cuyas siglas en español corresponde a Hojas de Estilo en Cascada (*Cascading Style Sheets*) permite dar estilos al código realizado en HTML como seleccionar la fuente y tamaño del texto, agregar columnas entre otras. Los aspectos visuales y estéticos del sitio web recaen sobre esta herramienta. [18]
- **JavaScript:** Es un lenguaje de programación que sirve para definir el comportamiento lógico de una página web. JavaScript es interpretado en los navegadores sin necesidad de complicación. Uno de sus usos principales es para la creación de páginas webs dinámicas, que puedan ser interactivas y animadas. [18]

1.5.9 Desarrollo de *backend*

El *backend* se encarga de procesar la información que se presentará al usuario en el *frontend* cuando este realice una petición, para poder hacer esto, el *backend* interactúa con bases de datos. Existen varios lenguajes de programación que se pueden utilizar para el desarrollo del *backend*.

Una de herramienta que se puede usar para el desarrollo del *backend* es PHP, que es un lenguaje de programación especializado para crear interfaces web en entorno al servidor. [19]

1.5.10 Base de datos

Una base de datos es una agrupación de datos que se encuentran organizados, estos datos pueden ser alfanuméricos, numéricos, cadenas de texto, fechas, imágenes, entre otras. Un sistema

de gestión de base de datos (DBMS) es el que permite administrar los datos almacenados. De forma general se pueden clasificar las bases de datos en relacionales y no relacionales.

1.5.11 Base de datos Relacionales

Se utilizan cuando los datos están estructurados, estos se almacenan en diferentes tablas que se relacionan entre sí. En una tabla cada registro tiene un identificador único que le permite relacionarse con otras tablas. Algunos ejemplos de gestores (DBMS) para este tipo de base de datos son MySQL, Microsoft SQL Server, PostgreSQL, entre otros que utilizan el lenguaje de consulta estructurada SQL. [18]

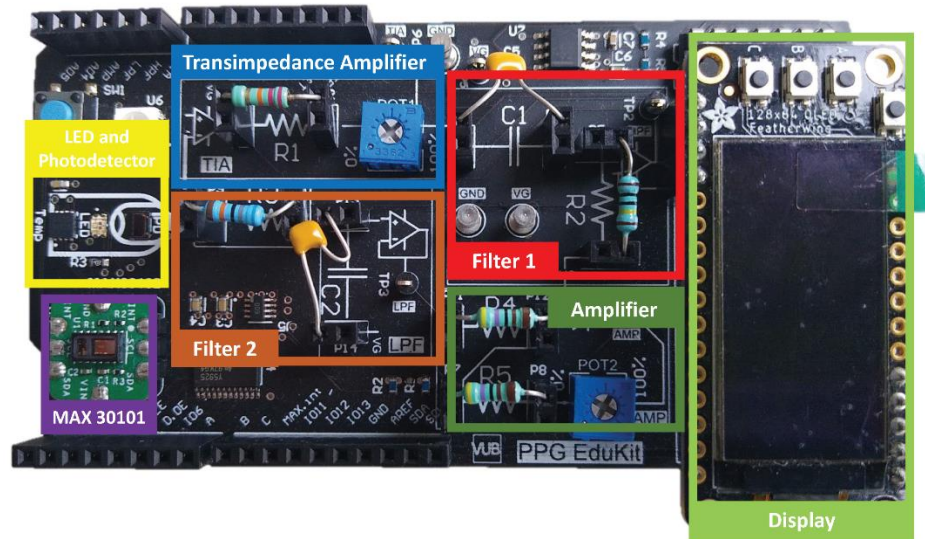
1.6 Estado del arte

1.6.1 PPG EduKit: Un sistema ajustable de evaluación por fotoplestimografía ajustable para actividades educativas.

El dispositivo EduKit se desarrolló con la intención de facilitar el aprendizaje de los métodos PPG con fines médicos y de ingeniería. PPG Edukit hace posible modificar e implementar señales PPG para obtener información fisiológica entre las que se encuentran la frecuencia cardíaca, niveles de oxígeno en la sangre y temperatura corporal. PPG Edukit hace uso de diodos LEDs y un fotorreceptor, la señal de este fotorreceptor se convierte de corriente a voltaje, se filtra, amplifica y se convierte en señal digital para finalmente ser mostrada en una pantalla LCD. Además, cuenta con un sensor MAX30100, el cual mide directamente la frecuencia cardíaca para ser comparada por la obtenida con el fotorreceptor. [20]

Figura 1.10

Modelo físico PPG Edukit [20]



1.6.2 Dispositivos portátiles.

Existen una gran variedad de dispositivos portátiles como relojes de mano o cinturones que el usuario se coloca y estos miden su frecuencia cardíaca y saturación de oxígeno. Estos dispositivos son de uso comercial, se encuentran en el mercado con precios desde aproximadamente \$500 en adelante.

Muchos de estos dispositivos también seguimiento de salud y son conectados a aplicaciones móviles para que el usuario pueda monitorizar de forma constante el progreso de su salud, son generalmente consumidos por personas que realizan deporte o actividad física constante ya que además permiten monitorizar los pasos, alimentación, horas de sueño entre otras.

1.6.3 Kit de sensores de pulso cardiaco

Generalmente se ubican en los dedos del paciente, esto son:

- MAX30120: Es un oxímetro y pulsímetro integrado que necesita una fuente de alimentación de 5 V, este compuesto por 2 leds, un infrarrojo y otro de color rojo y es comunicado mediante I2C al microcontrolador.
- MAX30100: Igual que el MAX30120 también es un oxímetro y pulsímetro integrado con diferencia que es a una menor resolución ya que funciona con niveles de voltaje más pequeños a 5V.

1.6.4 Desarrollo de un sistema web de citas médicas usando herramientas open source y sensores biométricos de signos vitales.

Como parte de un proyecto de titulación de la Universidad de Guayaquil, se llevó a cabo un proyecto cuya finalidad asignar la prioridad de citas médicas según el nivel de riesgo del paciente detectado por mediciones de sensores biométricos y permitir que se pueda recetar al paciente medicamentos para calmar sus síntomas hasta que se presente su turno para una cita médica.

Para el desarrollo de este trabajo Delgado J. y Tumbaco K. (2020) hacen uso de los siguientes sensores biométricos.

- Oxímetro, dispositivo capaz de medir los niveles de saturación de oxígeno en la sangre.
- Sensor de pulso cardíaco para Arduino con alimentación de 5v.
- Sensor de respiración que se debe colocar envolviendo al pecho y estómago.
- Sensor LM35 de temperatura que se encuentra calibrado a grados Celsius y comportamiento lineal.

Para realizar la página web hacen uso de PHP, un lenguaje de programación que es embebido entre el código HTML y se ejecutan en el servidor para poder responder a las peticiones de información de parte del cliente de forma HTML y mediante la herramienta Bootstrap dar formato HTML a su página web. Utilizan MySQL como gestor de la base de datos, y el framework LegoBox para enviar la respuesta adecuada según la petición al servidor. El controlador empleado es el Arduino UNO.

Un aspecto relevante de este trabajo es la investigación primaria que se realizó a una muestra con tamaño de 384 usuarios de servicios de agendamiento de citas en el sistema de salud pública de una zona de la ciudad de Guayaquil, cuyos resultados serán tomados en cuenta en el desarrollo de este proyecto de materia integradora. [21]

1.6.5 Diseño e implementación de un glucómetro no invasivo basado en la ley de Lambert-beer y longitud de onda cercana al infrarrojo (NIR), con interfaz de comunicación bluetooth a dispositivos con sistema operativo Android

Al realizar este proyecto, López V y Oñate W (2014) como parte de su trabajo de tesis para la Universidad Politécnica Salesiana de Quito, hicieron uso de la fotoplestimografía, con un sensor de oximetría que envía luz visible e infrarroja a longitudes de onda de 660 nm y 890 nm respectivamente, se realizaron los circuitos electrónicos necesarios para la conversión de la señal de fotoplestimografía a una señal electrónica, su respectiva amplificación y filtrado. Usando la ley de Lambert – Beer se relaciona la cantidad de luz absorbida medida en forma de voltaje con la concentración de glucosa, los resultados se obtuvieron con un error relativo de 5,226% con relación a productos similares de venta comercial. Se recopilan los datos mediante un teléfono inteligente

Android y se envían a una página web para permitir mejorar el seguimiento médico de los pacientes de diabetes. [22]

Figura 1.11

Prototipo de un glucómetro no invasivo realizado [22]



1.6.7 Pulsera inalámbrica para monitorización del pulso cardiaco

El proyecto está basado en realizar el diseño de una pulsera inteligente con la capacidad de medir el pulso cardiaco de una persona, el cual se podrá ser monitoreada y visualizada en una plataforma online MBED, el dispositivo cuenta con un sensor de pulso basado en fotoplestimografía y la placa Tiny-BLE donde se ubicará el microcontrolador. El sistema inteligente está basado en un ARM CórteX M-0, el cual se comunicará remotamente por bluetooth utilizando el protocolo BLE.

Las herramientas usadas para el correcto funcionamiento de la plataforma fueron C/C++ (SDK), HDK, Compiladores online.

Dentro de las ventajas relacionadas a la pulsera al entorno médico para realizar tele consultas, están:

- Realiza el monitoreo constante de la actividad cardíaca del paciente remotamente.

- Comodidad para el usuario adaptándose al entorno.
- Rapidez y sencillez en el diagnóstico.

Sin embargo, dentro de sus desventajas se encuentran:

- Probabilidad de falla en la comunicación Bluetooth.
- Mal manejo de personas sin conocimientos de tecnología.
- Incapacidad para proveer la información suficiente sobre el estado del

usuario. [23]

Figura 1.12

Pulsera inalámbrica para monitorización del pulso cardíaco [23]



Capítulo 2

2. Metodología

En esta sección se detallará la explicación del diseño electrónico y software del proyecto, se mostrará los requerimientos y la alternativa de solución con su respectivo proceso de diseño para su realización.

2.1.1 Requerimientos

Antes de realizar el diseño electrónico y programación del sistema de atención médica, es necesario definir las especificaciones o directrices que el cliente desea.

Estas son:

- Diseñar un dispositivo electrónico capaz de medir señales bioeléctricas y generar alarmas visuales de forma local y en la nube.
- Creación de una interfaz para el agendamiento de citas y visualización del historial médico por paciente en la nube captado por el dispositivo electrónico.

2.1.2 Selección de la alternativa de solución

La intención del proyecto es diseñar un dispositivo electrónico que sea capaz de medir la frecuencia cardíaca y la temperatura corporal de un individuo, el cual será controlado por un microcontrolador y se comunicara con una base de datos en la nube para posteriormente ser visualizado en una interfaz web por un médico.

Para realizar el sistema de atención médica con lo anteriormente dicho, se han propuesto algunas alternativas de solución referente a la selección de microcontrolador e interfaz de desarrollo para la cita médica.

2.1.2.1 Microcontrolador:

Alternativa 1

Diseñar un dispositivo electrónico utilizando el microcontrolador Raspberry pi picow.

Alternativa 2

Diseñar un dispositivo electrónico utilizando PIC.

2.1.2.2 Interfaz de desarrollo:

Alternativa 1

Crear una página web capaz de conectarse a una base de datos en la nube y agendar citas médicas.

Alternativa 2

Crear una aplicación móvil capaz de conectarse a una base de datos en la nube y agendar citas médicas.

Para escoger la mejor alternativa de solución, se evaluó cada una de las alternativas a diferentes criterios importantes para realizar el proyecto.

2.1.3 Criterios para las alternativas de la interfaz:

- **Funcionalidad:** Cantidad de operaciones que se puedan desarrollar.
- **Facilidad de uso:** Mayor accesibilidad al usuario para manejar la interfaz.

- **Seguridad:** Exposición a menos riesgo por factores externos como usuarios no deseados o filtración de información del programa.
- **Estética:** Mejor aspecto visual de la interfaz.
- **Complejidad de desarrollo:** Cual es más fácil de realizar la programación de la interfaz.

Tabla 2.1*Matriz de decisión de la interfaz de desarrollo*

	Criterio 1	Criterio 2	Criterio 3	Criterio 4	Criterio 5	Total
Descripción	Facilidad de uso	Funcionalidad	Seguridad	Complejidad de desarrollo	Estética	
peso	4	3	2	1.8	1	
porcentaje	34%	25%	17%	15%	8%	100%
Alternativas	Calificación	Calificación	Calificación	Calificación	Calificación	
Página web	8	8	6	9	7	7,6
Aplicación móvil	6	10	8	6	8	7,5

2.1.4 Criterios para las alternativas del microcontrolador:

- **Funcionalidad:** Cantidad de operaciones que se puedan desarrollar.
- **Comunidad de soporte:** Mayor ejemplos y librerías que se puedan consultar en la web.
- **Curva de aprendizaje:** Mayor educación y fundamentos de programación.
- **Comunicación Wireless:** Contar con la capacidad de realizar una conexión por bluetooth o ethernet.

Tabla 2.2

Matriz de decisión de las alternativas del microcontrolador

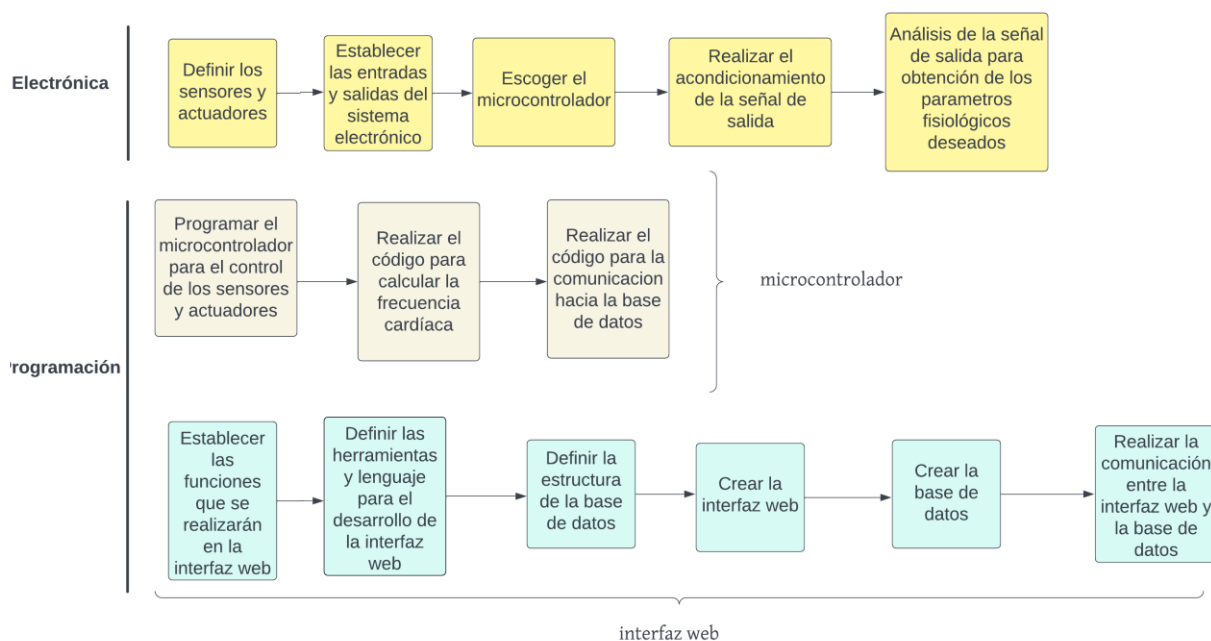
	Criterio 1	Criterio 2	Criterio 3	Criterio 4	Total
Descripción	Funcionalidad	Comunidad de soporte	Curva de aprendizaje	de Comunicación Wireless	
peso	4	3	2	2	
porcentaje	36%	27%	18%	18%	100%
Alternativas	Calificación	Calificación	Calificación	Calificación	
Raspberry pi	7	8	8	2	6.5
pico w					
PIC	3	2	2	7	3.3

La Raspberry pi pico w es el escogido por su comunidad de soporte y ser un microcontrolador básico con bastantes herramientas en la web para su aprendizaje.

2.2 Proceso de diseño

Figura 2.1

Diagrama del proceso de diseño



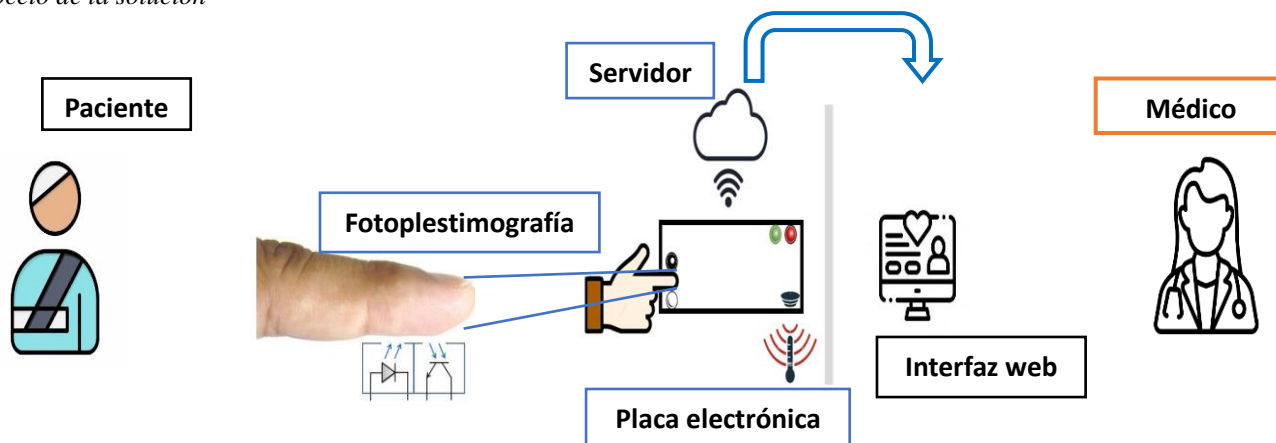
En el proceso de diseño se muestran las etapas a seguir para realizar el presente proyecto, esto claro ya una vez siendo definido las limitaciones y requerimientos que tendrá la solución.

El diseño está dividido en dos fases, el componente electrónico y la programación del microcontrolador e interfaz web para crear el sistema de atención médica a distancia.

2.2.1 Diseño conceptual

Figura 2.2

Boceto de la solución



El diseño conceptual consiste en crear un sistema de atención médica a distancia utilizando una placa de desarrollo y una página web. El dispositivo electrónico estará basado en el principio de fotoplestígrafía para medir la frecuencia cardíaca, además de la temperatura corporal del paciente y registrar el número de cédula mediante un teclado con la finalidad de usar la cédula del usuario para relacionarlo con los datos de los pacientes en la página web. La información captada por el dispositivo será enviada hacia un servidor en la nube para luego ser mostrada en la página web por el médico para observar los datos y un formulario de preguntas médicas contestadas por los pacientes.

2.2.2 Diseño electrónico

2.2.2.1 Selección de los sensores y módulos electrónicos

Para realizar la fotoplestimografía se debe escoger un led rojo de alta intensidad o una luz infrarroja para lograr traspasar la piel, la longitud de onda es un factor crítico al momento de escoger el led. El rango de interés para la selección de la longitud de onda esta entre 600 a 950 nm. El led rojo de alta luminosidad tiene un consumo aproximado de 20mA y una longitud de onda de 660 nm, luego se escogerá un fototransistor, es un transistor bipolar normal que es sensible a la luz, permite el paso de corriente entre el emisor y el colector, mientras a mayor luz este expuesta, mayor será la corriente de salida. El sensor escogido es el TCRT5000, este ya viene incluido el led infrarrojo y el fototransistor, además esta encapsulado en una carcasa que no permite la interferencia de la luz exterior.

Figura 2.3

Sensor TCRT5000



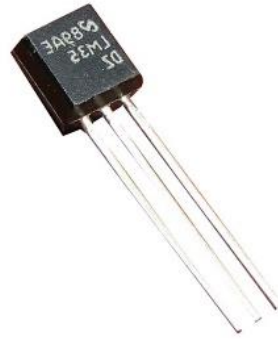
Características:

- Corriente máxima del colector de 10mA.
- Longitud de onda de 950nm.

Para medir la temperatura corporal se seleccionó el sensor LM35, tiene un rango de medición de -5 hasta 150 °C, su salida es una señal analógica proporcional a la temperatura en grados Celsius.

Figura 2.4

Sensor LM35



Se seleccionó el display OLED SSD1306 para la visualización de la onda cardíaca captada por el sensor, tiene un voltaje de operación de 3.3 a 5 V con una comunicación SPI y una resolución de 128 por 4 bits.

Figura 2.5

Pantalla OLED SSD1306



Además, se escogió el teclado matricial 4x4 para que el paciente pueda ingresar su número de cédula que se servirá para identificar sus datos enviados a la página web.

Figura 2.6

Teclado matricial



2.2.2.2 Selección de microcontrolador

La Raspberry pi pico w es el microcontrolador RP2040 que cuenta integrado un módulo de Wifi, está compuesto por 26 pines digitales y 4 analógicos, una memoria flash y RAM de 2Mb y 264 Kb respectivamente, una frecuencia de reloj de 133 Mhz y un voltaje de trabajo de 3.3 V.

Figura 2.7

Raspberry pi pico w



Se seleccionó la Raspberry pi pico w, ya que el microcontrolador es capaz de ser programado con micropython, lo cual facilitará el cálculo de la frecuencia cardíaca para manejar los datos en listas y para realizar la comunicación con el módulo Wifi interno.

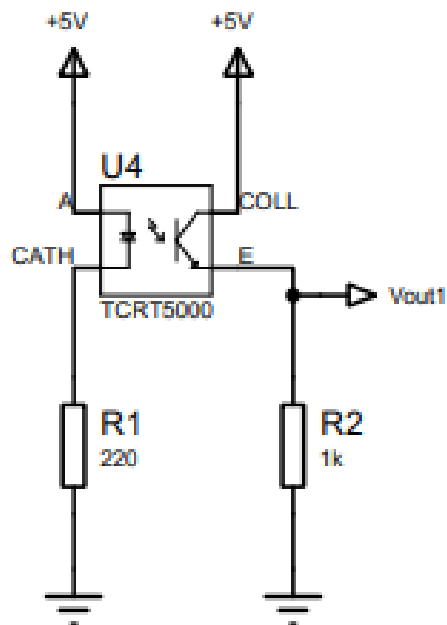
2.2.2.3 Etapa de acondicionamiento

Se realizó una serie de procesos para lograr el respectivo acondicionamiento de la señal obtenida del fototransistor utilizando amplificadores operacionales.

2.2.2.3.1 Conexión del sensor TCRT5000

Figura 2.8

Diagrama de conexión del sensor TCRT5000 en Proteus



El votaje de salida V_{out1} esta dada por:

$$V_{out1} = R2 * I_c \quad (2. 1)$$

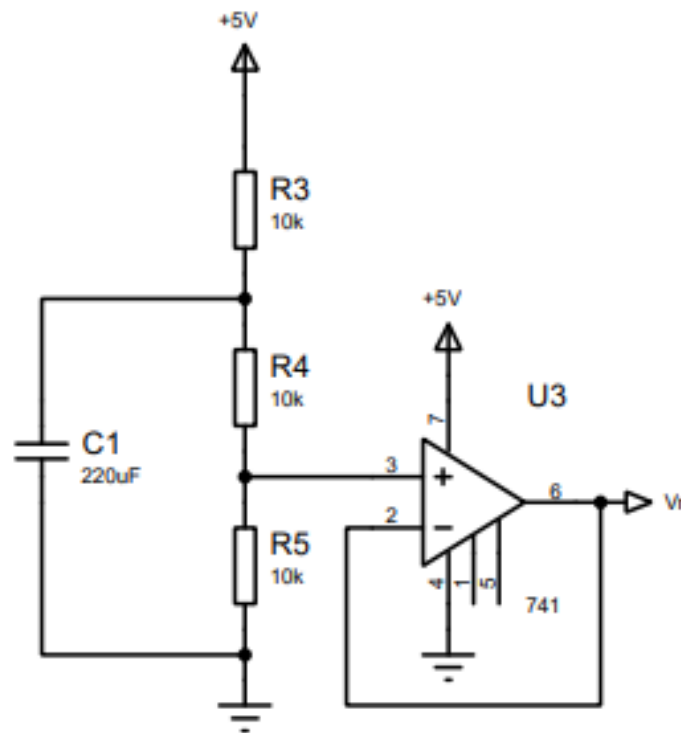
Donde I_c es la corriente que pasa por el fototransistor debido a la presencia de luz y $R2$ es la resistencia.

La corriente I_c se encuentra en constante cambio alterando el voltaje de salida V_{out1}

2.2.2.3.2 Nueva referencia de voltaje más un seguidor de tensión

Figura 2.9

Referencia de voltaje en Proteus



Se construyó un divisor de voltaje con la fuente de alimentación $V_c = 5V$, para obtener un voltaje de referencia (V_r) para la entrada no inversora del amplificador operacional de la siguiente etapa.

$$V_r = \frac{R_5}{R_3 + R_4 + R_5} * V_{cc} \quad (2.2)$$

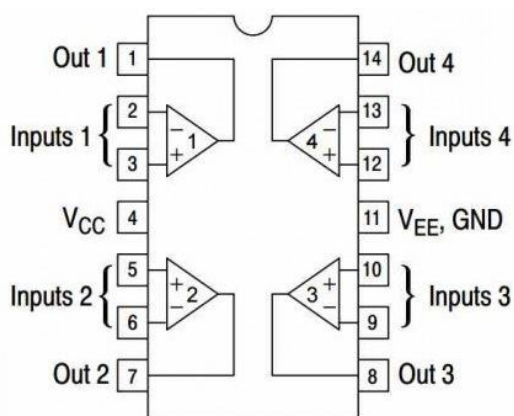
$$V_r = \frac{10k\Omega}{30k\Omega} * 5V$$

$$V_r = 1.67 V$$

La razón principal del divisor de voltaje es para mantener un valor cercano al voltaje de operación del amplificador operacional LM324, que es de 3 a 32 V al solo usar una fuente de alimentación positiva, la cual es detallada en su *datasheet*.

Figura 2.10

LM324N Configuración de los pines



Además, se colocó en paralelo un capacitor al circuito anterior ya que proporciona una mayor estabilidad de voltaje y elimina la señal de ruido provocado por la alimentación de 5V.

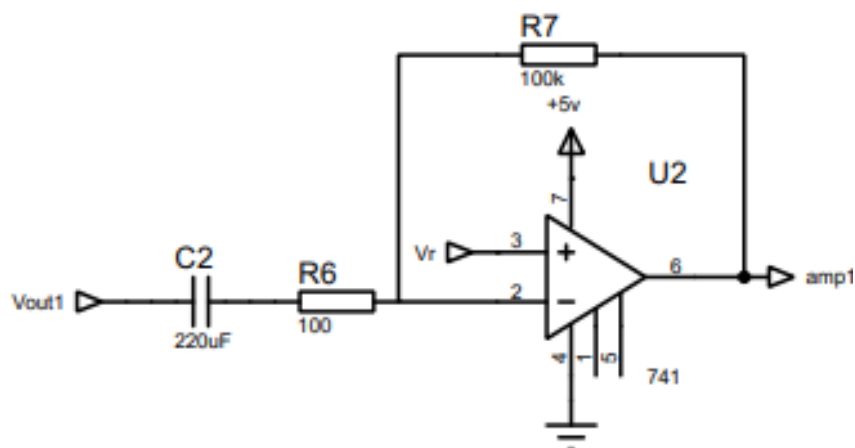
El divisor de voltaje no se ve afectado al conectar en paralelo el capacitor y el ruido concentrado en la resistencia R_3 es limpiada luego para obtener un voltaje de referencia limpio.

El valor del capacitor no provoca ningun cambio al divisor de voltaje, por lo que no es importante de analizar.

Finalmente se añade el seguidor de tensión para mantener el valor de voltaje y poder usar el voltaje de referencia en otros circuitos de los amplificadores operacionales.

2.2.2.3.3 Amplificador inversor

Figura 2.11
Amplificador Inversor



La variación de voltaje recibida del sensor TCR5000 es muy pequeña por lo que debe ser amplificada, está en el rango de milivoltios por lo que es necesario realizar una etapa de amplificación.

La **Figura 2.11** muestra la configuración del amplificador inversor implementado, donde la entrada de Vr en el terminal positivo hace que funcione como amplificador diferencial inversor que amplifica e invierte la diferencia entre el voltaje vout1 y Vr.

La fórmula del voltaje de salida con respecto al voltaje de entrada se halla aplicando la ley de corrientes de Kirchhoff siguiendo el principio de alta impedancia de los amplificadores operacionales el voltaje en del terminal positivo debe ser igual al voltaje en el terminal negativo.

$$\frac{(V_{out1} - V_r)}{R_6} = \frac{(V_r - V_{amp1})}{R_7} \quad (2.3)$$

$$\frac{R_7}{R_6}(V_{out1} - V_r) = (V_r - V_{amp1})$$

$$V_r - V_{amp1} = \frac{R_7}{R_6}(V_{out1} - V_r)$$

$$V_{amp1} = -\frac{R_7}{R_6}(V_{out1} - V_r) + V_r$$

Figura 2.12

Comportamiento obtenido con voltaje de prueba

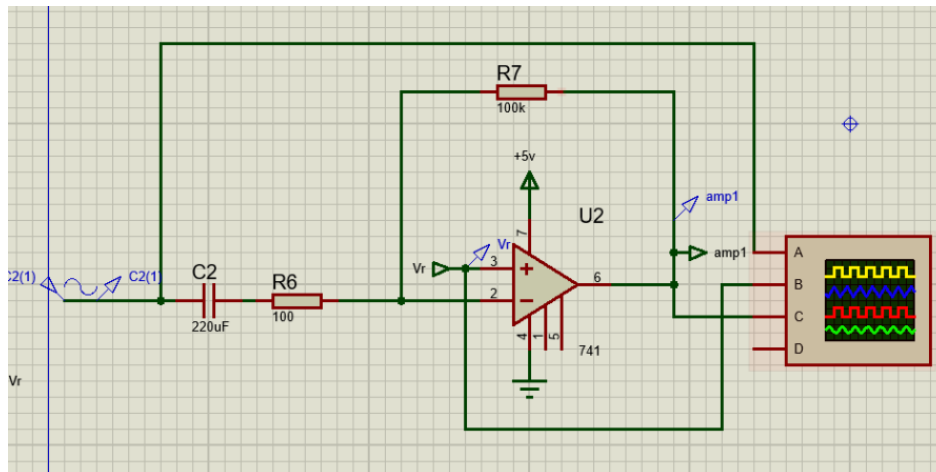
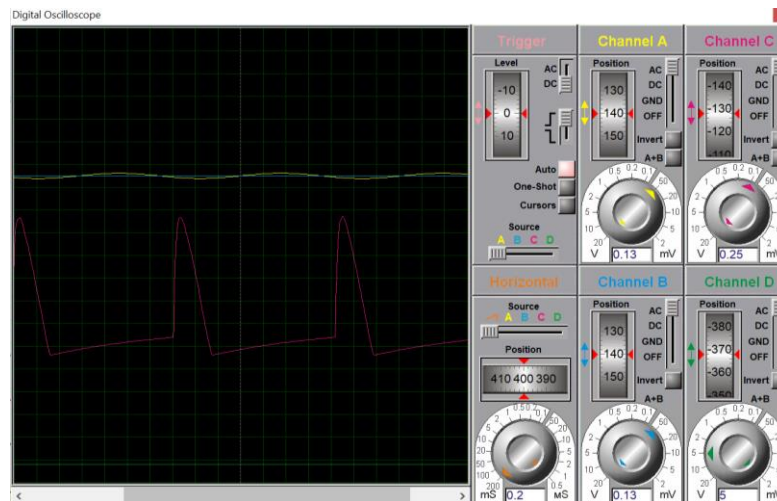


Figura 2.13

Osciloscopio



Para poder realizar pruebas, se colocó un osciloscopio para visualizar el comportamiento del circuito. Se colocó como señal de entrada una señal con una amplitud en rango de milivoltios con un offset del mismo nivel de la referencia V_r . Y logramos ver que se obtiene una amplificación suficiente de la onda, la salida del circuito se muestra de color rosado en la **Figura 2.13**.

2.2.2.3.4 Etapas de filtrado

La señal PPG esta compuesta por dos componentes: AC y CC, se desea obtener la AC, la cual refleja los latidos del corazón, sin embargo al ser esta pequeña en comparacion con la señal en su totalidad, se realizó un filtro pasa banda para mantener la señal obtenida del PPG dentro de un rango de frecuencia deseado.

Los latidos por minutos de una persona dependen de su edad y salud, si se establece un rango entre 30 a 190 lpm, al realizar la conversión de lpm a Hz, se obtiene el rango de frecuencia:

$$Hz = lpm/60min \quad (2.4)$$

$$frecuencia1 = \frac{30lpm}{60min} = 0.5Hz$$

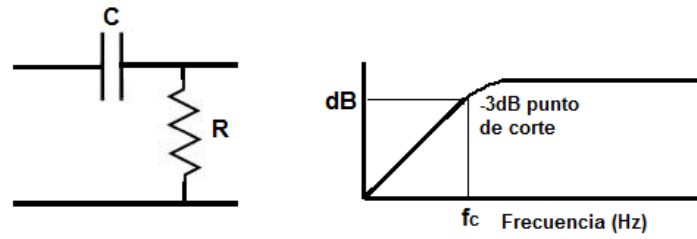
$$frecuencia2 = \frac{190lpm}{60min} = 3.16Hz$$

El rango de interés de frecuencia deseadas será de 0.5 a 3.16 Hz.

Para realizar los límites de frecuencia, se usó un filtro de banda que esta dividido en dos partes: filtro pasa alto y filtro pasa bajo.

Figura 2.14

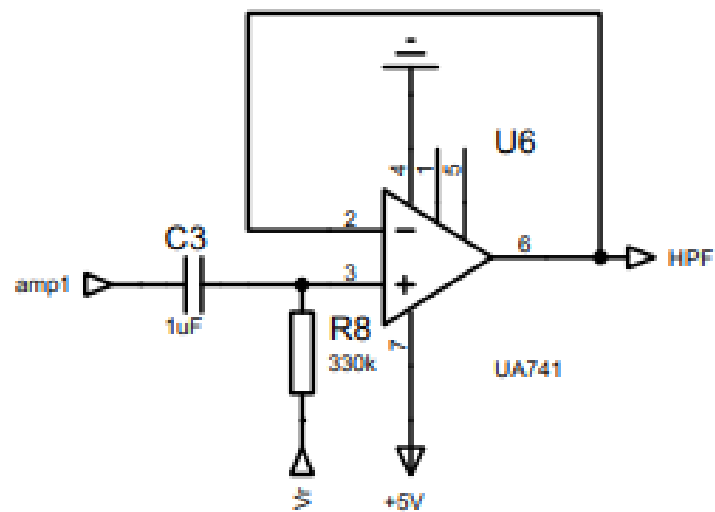
Filtro pasa altos - pasivo



$$f_c = \frac{1}{2\pi RC}$$

Figura 2.15

Filtro pasa alto RC pasivo más seguidor de voltaje en Proteus



Los valores de resistencia y capacitor son:

$$R8 = 330k\Omega$$

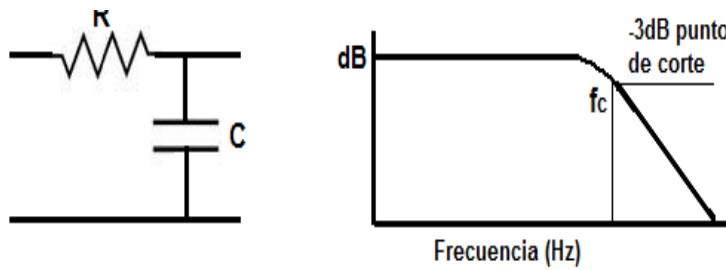
$$C3 = 1\mu f$$

Y el valor de la frecuencia de corte para el filtro pasa alto es:

$$f_c = \frac{1}{2\pi * R * C} = \frac{1}{2\pi * 330k * 1\mu} = 0,48 \text{ Hz} \quad (2.5)$$

Figura 2.16

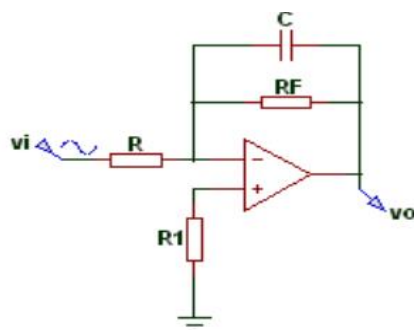
Filtro pasa bajos – pasivo



$$f_c = \frac{1}{2\pi RC}$$

Figura 2.17

Filtro pasa bajos - activo



$$\frac{V_o}{V_i}(s) = -\left(\frac{R_f}{R}\right) \left(\frac{\frac{1}{CR_f}}{s + \frac{1}{CR_f}} \right)$$

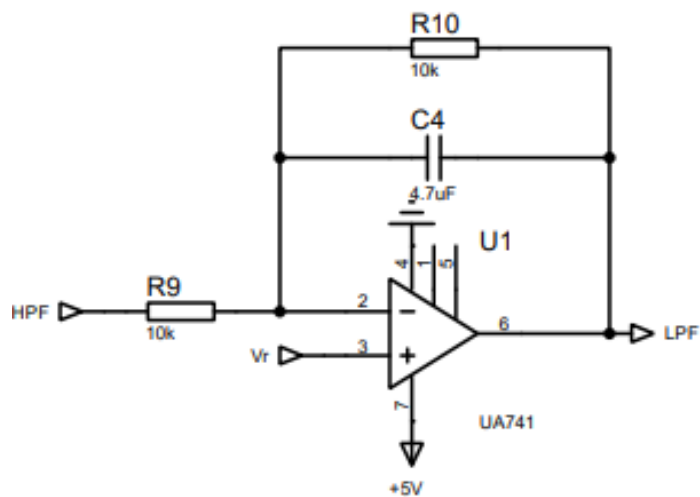
$$R_f = \frac{1}{2\pi f_c C}$$

$$R = \frac{R_f}{A}$$

$$R_1 = R \parallel R_f$$

Figura 2.18

Filtro pasa bajo en Proteus



Los valores de resistencia y capacitor son:

$$R9 = 10k\Omega$$

$$C4 = 4.7\mu f$$

$$R10 = 10k\Omega$$

Y el valor de la frecuencia de corte para el filtro pasa bajo es:

$$f_c = \frac{1}{2\pi * R * C} = 3.38 \text{ Hz} \quad (2.6)$$

Y finalmente le damos una ganancia G de 1 para no afectar al voltaje anterior.

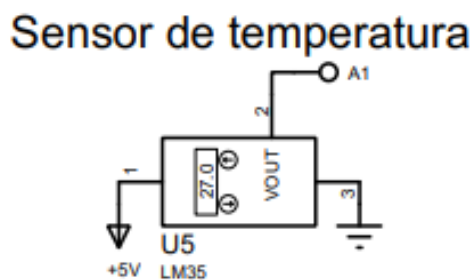
$$G = \frac{R7}{R10} = \frac{10k}{10k} = 1 \quad (2.7)$$

Esto es debido al rango de voltaje medido del fototransistor, el cual fue realizado de manera experimental hasta obtener un nivel de voltaje adecuado.

2.2.2.4 Conexión del sensor LM35

Figura 2.19

Diagrama del sensor LM35

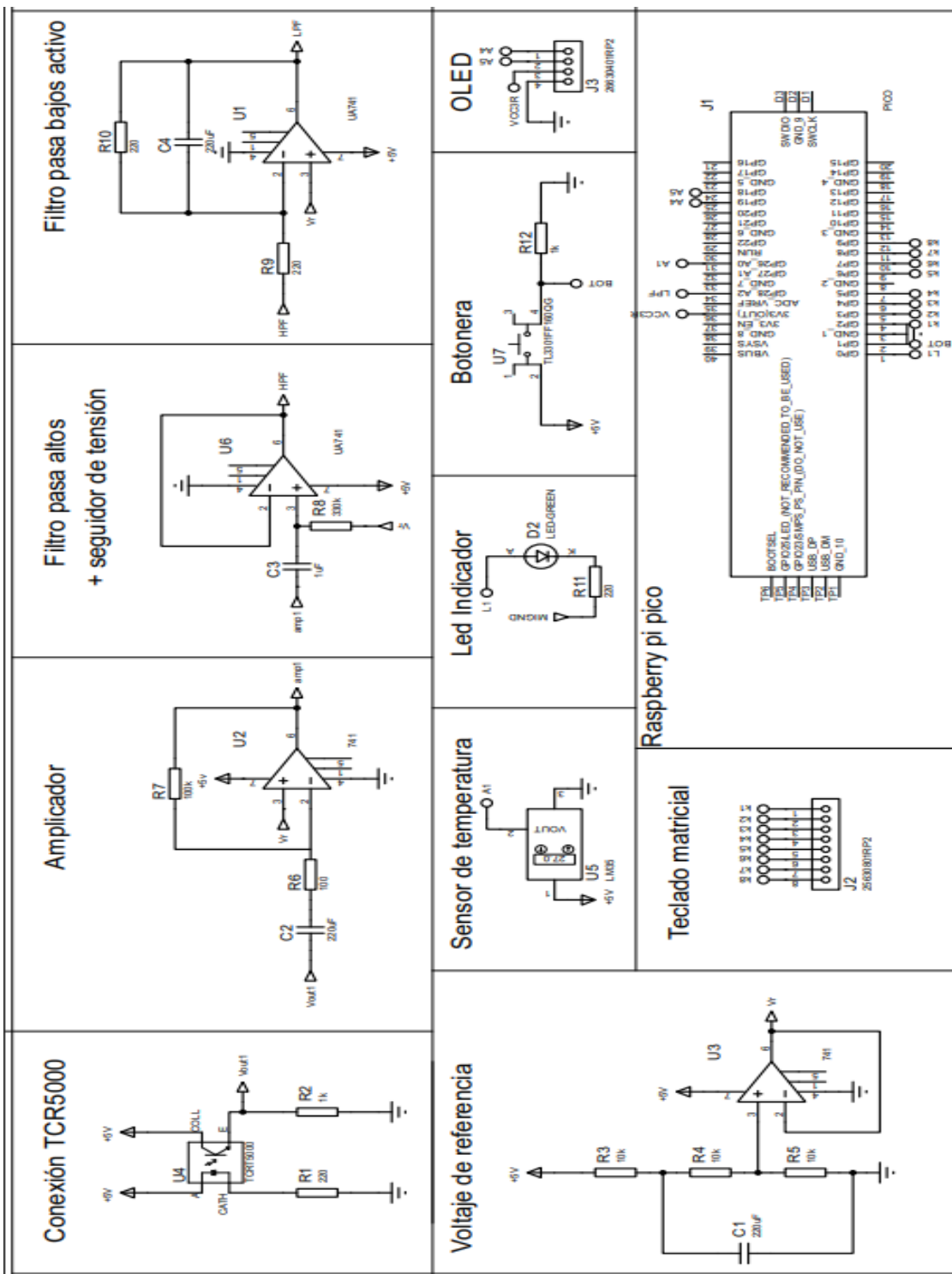


Se demuestra la conexión del sensor LM35, donde se visualiza el pin de alimentación, tierra y la salida de la entrada analógica.

2.2.2.5 Esquemático con las conexiones realizadas del circuito

Figura 2.20

Conexiones eléctricas del sistema electrónico descrito

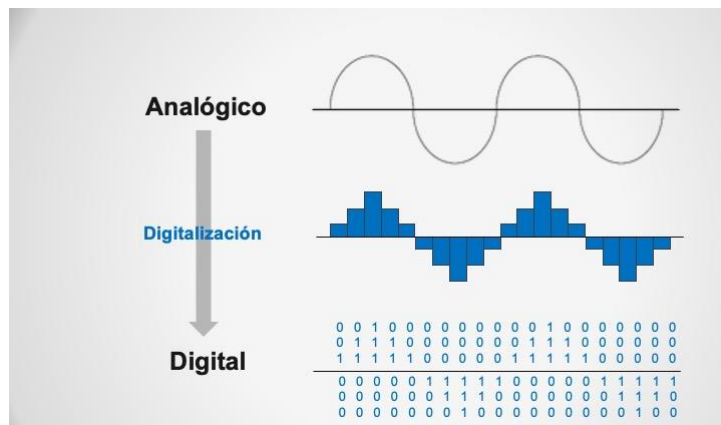


Se detallan las conexiones realizadas de los componentes de este proyecto, donde se detalla la explicación de la sección anterior para el acondicionamiento de señal y además demuestra la conexión de los otros componentes utilizados entre ellos con la ubicación de los pines del Raspberry pi pico que es el microcontrolador que será utilizado para controlar y ejecutar las siguientes acciones:

- Leer las entradas analógicas del sensor de temperatura y pulso cardíaco
- Utilizar un botón y led para iniciar y representar el estado de encendido del dispositivo.
- Calcular la frecuencia cardíaca
- Graficar el pulso de onda utilizando el OLED SSD1306
- Escribir el número de cédula del paciente mediante un teclado matricial para luego asociarlo con la base de datos.

2.2.2.5 Etapa de digitalización

Utilizando los puertos de entrada analógica de la Raspberry pi pico w, se recibe la señal PPG realizada con el sensor TCRT5000 y el sensor de temperatura LM35.

Figura 2.21*Digitalización de los datos analógicos*

La Raspberry pi pico w cuenta con un convertidor interno de analógico a digital, la resolución es de 16 bits, representando valores de 0 a 65535, el máximo valor de voltios (3.3V) equivale a 65535 y 0 bits equivale a (0V).

Para leer las entradas analógicas, se usa la función `read_u16()`.

2.2.2.6 Diseño de programación

2.2.2.6.1 Activación del circuito

Se usó un botón y led para representar el estado de encendido del circuito, al aplastar el botón se prende el led indicando la activación de la lectura de las entradas analógicas del sistema.

Figura 2.22*Código de activación del circuito*

```

buttonValue = button.value()
if (button.value()):
    buttonValue= True
    #buttonState= not buttonState

else:
    buttonValue= False

if buttonValue != lastButtonState:
    lastDebounceTime = time.ticks_ms()
    #if (time.ticks_ms() - lastDebounceTime) > debounceDelay:
    #if buttonValue != buttonState:
    buttonState = buttonValue

    if buttonState == True:
        ledState = True
        med = True
        lista_t2=[]
        lista_d2=[]

led.value(ledState)
lastButtonState = buttonValue

```

El código realizado los cambios de estado del botón para prender y apagar el botón.

2.2.2.6.2 Visualización de la onda cardíaca

Implementando el OLED SSD1306, se realizó la grafica del voltaje leído por el sensor TCRT5000, dicho voltaje esta en el rango de 0 a 2.3 V.

Para realizar el código de la pantalla, se implemento una función plot_time() donde se define el eje x e y, el cual sera representando como tiempo y voltaje respectivamente, esto significara que de manera continua se leerán los datos del sensor que serán visualizados dentro de una escala de 0 a 3.3 V sobre el eje y.

Figura 2.23

Código de la pantalla OLED

```
def plot_time(yp, t, x, y, var = [0.0,3.3], vpts=[25, 16, 40], hpts = [25, 55, 112]):
    #Axis
    oled.vline(vpts[0], vpts[1], vpts[2], 1) #x, y, h
    oled.hline(hpts[0], hpts[1], hpts[2], 1) #x, y, w
    oled.text(str(round(var[0],1)), vpts[0]-25, hpts[1]-5)
    oled.text(str(round(var[1],1)), vpts[0]-25, vpts[1])
    #y - axis
    y[1] = int((yp-var[0])/(var[1]-var[0]) * (vpts[1]-hpts[1]) + hpts[1]) #Interpolation
    if t < hpts[2] - hpts[0]:
        x[1] = x[0]+1
    else:
        x[1] = hpts[2]

    #Plot the line
    oled.line(x[0],y[0],x[1],y[1],1)
    oled.show()

    #Update past values
    y[0] = y[1]
    x[0] = x[1]
```

2.2.2.6.3 Ingreso del número de cédula del paciente al Raspberry pi pico

Mediante una función que permita al usuario escribir su número de cédula hasta lograr que se complete los diez dígitos en el teclado matricial llamada get_cedula()

Figura 2.24

Código para registrar el número de cédula

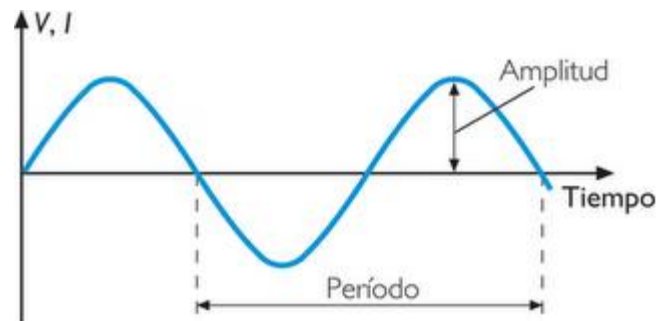
```
def get_cedula():
    global cedula_value, cedula_obtenida # Accedemos a las variables globales
    cedula_digits = []
    while not cedula_obtenida and len(cedula_digits) < 10: # Mientras no se haya obtenido una cédula
        key = Keypad4x4Read(col_list, row_list)
        if key is not None and key.isdigit():
            cedula_digits.append(key)
            print("Cédula parcial:", "".join(cedula_digits))
            time.sleep(0.3)
        if len(cedula_digits) == 10:
            cedula_value = "".join(cedula_digits)
            cedula_obtenida = True
            return cedula_value
```

2.2.2.6.4 Cálculo de la frecuencia cardíaca

A partir de la señal PPG digitalizada en el puerto analógico de la Raspberry pi pico w, obtenemos el valor de los latidos por minuto.

La señal obtenida puede ser interpretada como una onda senoidal, la cual está caracterizada principalmente por tener una amplitud, período y frecuencia.

Figura 2.25
Onda senoidal



El período T está directamente relacionado con la frecuencia f :

$$T = \frac{1}{f} \quad (2.8)$$

A partir de los valores recibidos del dispositivo, se obtuvo el período realizando un conteo del tiempo durante el transcurso en alcanzar dos valores máximos que representan el pico de la onda.

Para realizar el cálculo, se utilizó el lenguaje microPython mediante la Raspberry pi pico w, se creó cuatro listas vacías para almacenar los valores obtenidos y calculados del programa, tal y como se muestra en la siguiente imagen:

Figura 2.26

VARIABLES USADAS PARA CALCULAR LA FRECUENCIA CARDÍACA

```
lista_t2 = []  
lista_d2 = []  
picos_t2 = []  
picos_d2 = []
```

- La lista_t2 representa los valores de voltaje obtenidos del sensor TCRT5000.
- La lista_d2 representa el tiempo en milisegundos de los valores leídos del sensor TCRT5000.
- La lista picos_t2 almacena los valores de voltaje máximo que se obtienen al momento que se producen los picos de la onda cardíaca.
- La lista picos_d2 almacena el tiempo en milisegundos de los voltajes máximos leídos.

Además, se crea una variable donde se guardará el valor promedio de frecuencia cardíaca.

Figura 2.27

Código para calcular la frecuencia cardíaca

```

if(len(lista_t2)>0):
    for i in range(1, len(lista_t2)-1):
        if lista_t2[i] > lista_t2[i - 1] and lista_t2[i] > lista_t2[i + 1]:|
            picos_d2.append(lista_d2[i])
            picos_t2.append(lista_t2[i])

    for i in range(1,len(picos_d2)-1):
        if i > 0:
            restas.append(picos_d2[i]- picos_d2[i - 1])
            prom_resto = sum(restas) / len(restas)
            prom_resto_seg = prom_resto / 1000
            prom_resto_min=60 / prom_resto_seg
            v1=prom_resto_min

print("Frecuencia:", 60 / prom_resto_seg)
EnviarDatos()
time.sleep(1)
prom_resto_min=0
picos_t2=[]
picos_d2=[]
restas = []

```

El código recorre la lista de voltajes obtenidos del sensor TCRT5000 para luego reconocer y almacenar los valores de voltaje máximo y tiempo de dicho momento en la lista picos. Luego se calcula la diferencia de los tiempos entre dos picos de toda la lista, se transforma el tiempo de milisegundos a segundos y finalmente se lo convierte a minutos, pero de forma inversa tal y como se lo muestra en la **ecuación (2. 8)**.

2.2.2.6.5 Cálculo de la temperatura corporal

Se realizó la lectura de la entrada analógica del sensor LM35, posterior para poder obtener la temperatura en grados Celsius se usó las siguiente formula:

$$\text{Temperatura Celsius LM35} = (\text{pinLM35.read_u16} * (0.0005) * 100) \quad (2.9)$$

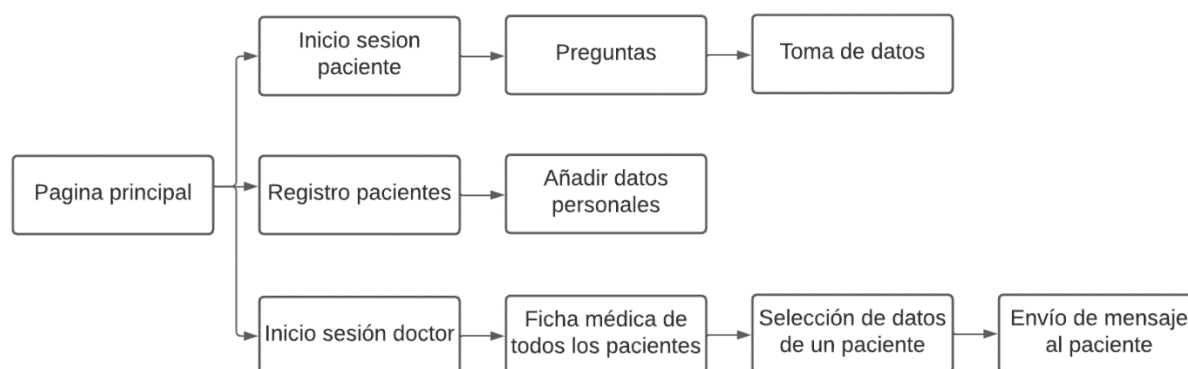
Según el *datasheet* del sensor LM35, cada grado Celsius representa 10mV. Primero se realiza una conversión del sensor analógico a voltaje y luego se multiplica por un factor de 100 para obtener la temperatura en grados Celsius.

2.2.2.6.6 Interfaz web

Para cumplir con el objetivo de visualización de los datos del paciente en una plataforma web, se desarrolló un código basado en lenguaje PHP con el cual se generan todos los componentes, funciones y características de la interfaz web.

Figura 2.28

Diagrama de flujo de la interfaz web



En el diagrama de flujo se muestran las diferentes secciones y acciones que se llevaron a cabo en la interfaz web. Este contara con un inicio de sesión tanto para el paciente y el médico, sin embargo, el paciente deberá crear un registro donde completara información básica necesaria para la consulta médica mientras que el doctor tendrá una cuenta administrativa genérica, una vez que el paciente haya iniciado sesión, contestara una serie de preguntas relacionada a su salud y finalmente se tomaran los datos del dispositivo electrónico.

Además, se utilizó el lenguaje CSS para brindarle un estilo personalizado al texto, tablas y cuadros de la página web para aumentar el aspecto visual de la interfaz.

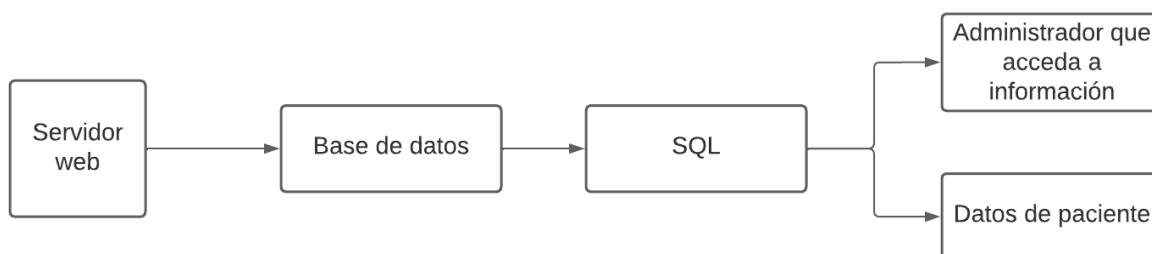
2.2.2.6.5 Servidor web

Con el objetivo de enviar la información del dispositivo electrónico a la web, se decidió hacer uso del servicio de XAMPP, es un software abierto para todo público que permite relacionar la base de datos MySQL y el servidor Apache.

Sin embargo, para el alojamiento web y poder visualizar la interfaz web en internet, se usó webhost.

Figura 2.29

Base de datos



El médico tendrá acceso a la información clínica de cada paciente registrado en la interfaz web, la cual podrá ser visualizada mediante una página web.

2.2.2.6.6 Comunicación por wifi hacia un servidor web

Los latidos por minutos y temperatura corporal obtenidos de la Raspberry pi pico w son llevados hacia un servidor web para guardar el registro de la información en una base de datos.

Para enviar los datos mediante Wifi, se usó la librería `urquests`, `network` y `time` para la comunicación del servidor. Primero se logra la conexión con el nombre y contraseña de Wifi local, posterior se usa la petición **POST** para solicitar enviar los datos hacia el servidor

Figura 2.30 Código para realizar el envío de datos al servidor

```
def EnviarDatos():
    data= "&value1=" + str(v1) + "&value2=" + str(v2)

    headers = {
        "Content-Type": "application/x-www-form-urlencoded"
    }
    response=urquests.post("http://192.168.0.104/tconsulta/post-esp-data.php", data=data, json=None, headers=headers)
# Check the response status code
if response.status_code == 200:
# The data was sent successfully
    print("Data sent successfully")
    print(response.text)
else:
# The data was not sent successfully
    print("Error sending data:", response.status_code)
    print(response.text)
```

Se definió una función para realizar la comunicación hasta el servidor, donde se define la url que representa el dominio de servidor que realizara la captura de valores enviados del método **POST**.

Los datos son enviados en una forma de formulario, declarados como tipo String, la cual está compuesta por una clave y valor. La clave contiene el nombre de la variable y el valor su respectiva medida, tal y como se muestra en la siguiente imagen:

Figura 2.31

Forma del tipo de dato enviado con clave y valor

```
Array  
(  
    [value1] => 15  
    [value2] => 30  
)
```

El código **POST** usado es visualizado por el siguiente código:

Figura2.32

Código POST

```
response=urequests.post("http://192.168.0.104/tconsulta/post-esp-data.php", data=data, json=None, headers=headers)
```

Es importante especificar el url de la extensión donde se realiza la petición.

2.2.2.6.7 Base de datos

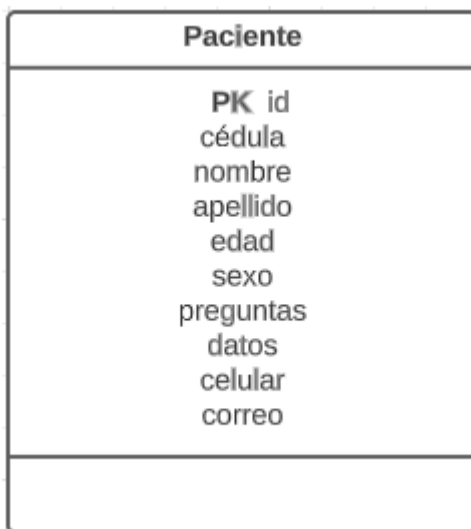
Se utilizó el gestor de base de datos relacional MYSQL, para permitir crear una relación entre la información clínica del paciente en formas de tablas, es una estructura organizada donde una de sus variables está identificada como una clave principal, esta sirve para identificar los datos relacionados al objeto o entidad.

Creamos cinco tablas SQL, estas son: paciente, doctor, sensorData, preguntas e indicaciones, las cuales harán la base de datos llamada basepac.

Se detallará la arquitectura de cada tabla:

Figura 2.33

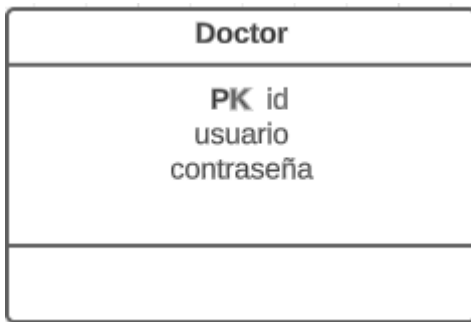
Diagrama de entidad del Paciente



La tabla paciente está compuesta por ID, cédula, nombre, apellido, edad, sexo, preguntas y datos. La cédula es la que se usará para la verificación y relación con las demás tablas de la base de datos y poder identificar los diferentes tipos de pacientes que habrá con su respectiva información.

Figura 2.34

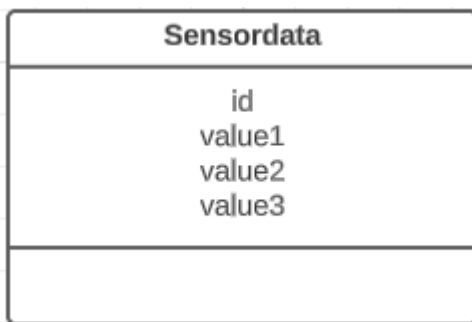
Diagrama de entidad del Doctor



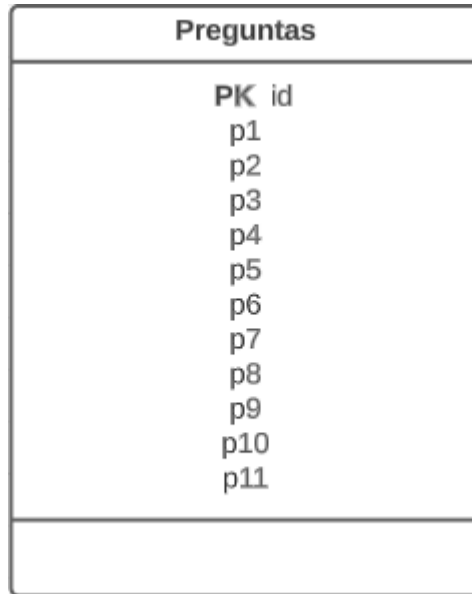
La entidad doctor está compuesta por las variables: ID, usuario y contraseña. Esta es una cuenta administrativa con la capacidad de poder leer la información de los pacientes registrados a través de su cédula.

Figura 2.35

Diagrama de entidad Sensordata



En sensordata se guardarán los valores recibidos del microcontrolador NodeMCU, que son: frecuencia cardíaca, temperatura del paciente y cédula del paciente, los cuales serán asignados a value1, value2 y value3 respectivamente.

Figura 2.36*Diagrama de entidad Preguntas*

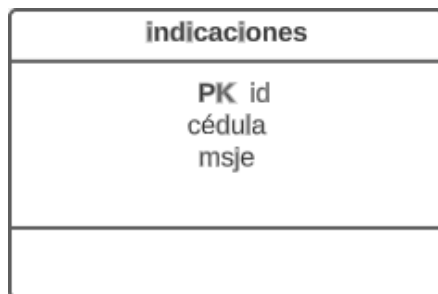
La tabla preguntas cuenta con once preguntas acerca del estado de salud del paciente, otorgando información importante para el análisis del doctor para la consulta.

Las preguntas son:

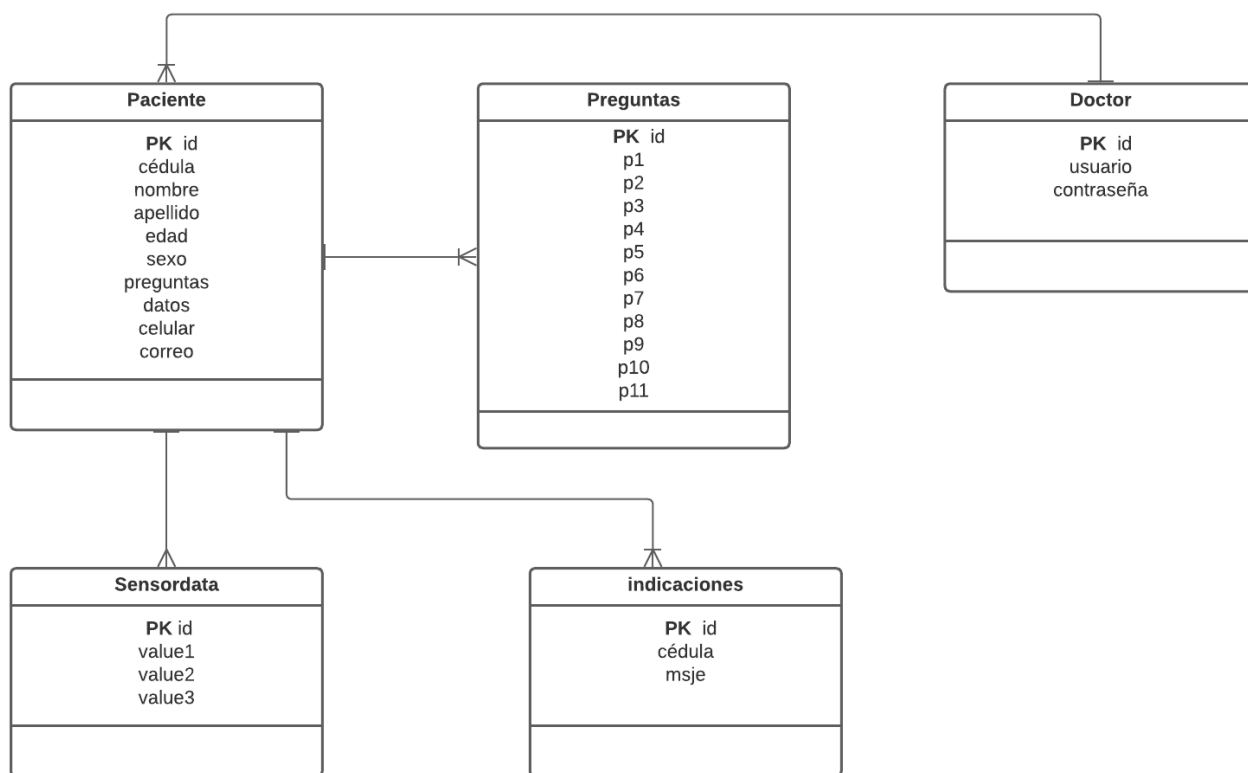
- Antecedentes:
 - 1) ¿Sufre de alguna enfermedad tal como hipertensión y diabetes? Y cual
 - 2) ¿Sufre de alguna alergia? Y cual
 - 3) ¿Algún familiar sufre de algunas de las enfermades anteriormente dichas?
 - 4) ¿Ha sido operado alguna vez?
 - 5) En caso de haber sido operado, ¿Cuánto tiempo ha sido desde que se ha operado?
 - 6) ¿Toma algún medicamento?

- Motivo de consulta:
 - 7) ¿Presenta algún tipo de dolor?
 - Pulsátil
 - Opresivo
 - Cólico
 - 8) ¿En qué parte se encuentra el dolor?
 - Cabeza
 - Tórax
 - Abdomen
 - Extremidades superiores
 - Extremidades inferiores
 - 9) ¿Cuál es la intensidad del dolor?
 - Bajo
 - Medio
 - Fuerte
 - 10) ¿Además del dolor, presenta algún otro síntoma? ¿Y cuál?
 - 11) De una descripción de su problema:

Las preguntas fueron elaboradas por la asistencia de un doctor para poder realizar una consulta médica general.

Figura 2.37*Diagrama de entidad Indicaciones*

La tabla indicaciones guardará la cédula y el mensaje que desea enviar el doctor al paciente para el agendamiento de una cita.

Figura 2.38*Diagrama de relación*

En el diagrama de relación se establecen la conexión de las tablas entre sí, dando a entender que teniendo un doctor recibirá muchos pacientes y un paciente tendrá muchas preguntas, indicaciones y Sensordata de realizar.

Capítulo 3

3. Resultados y análisis

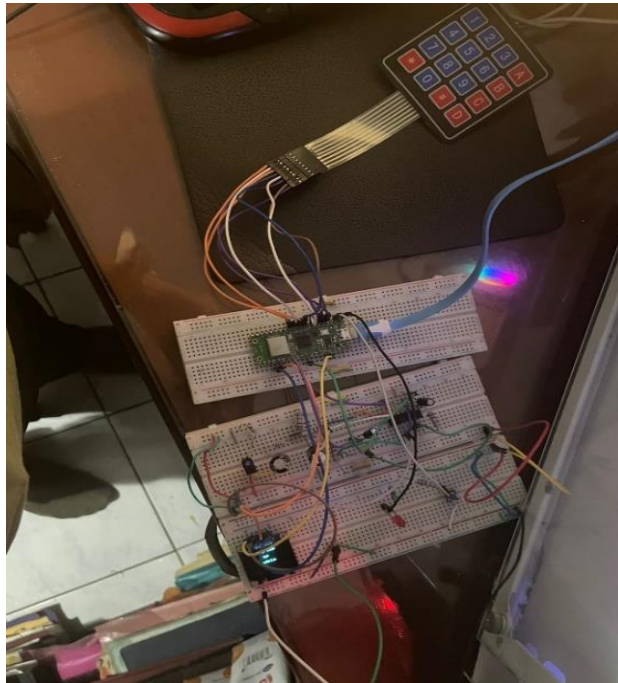
En esta sección se demostrarán los resultados y análisis realizados del sistema de consulta médica a distancia, donde se detallarán la funcionalidad del sistema electrónico conectado con la interfaz web desarrollada, tal y como se explicaron en la metodología.

Está dividida por dos partes: electrónica y programación. En la parte electrónica se podrá visualizar la comparación de los resultados obtenidos del sensor de frecuencia cardíaca (TCRT5000) con un sensor digital y el diseño del PCB del sistema electrónico integrado la placa de desarrollo Raspberry pi pico w. En programación se podrá observar las distintas interfaces creadas para el paciente y médico.

3.1.1 Electrónica

Figura 3.1

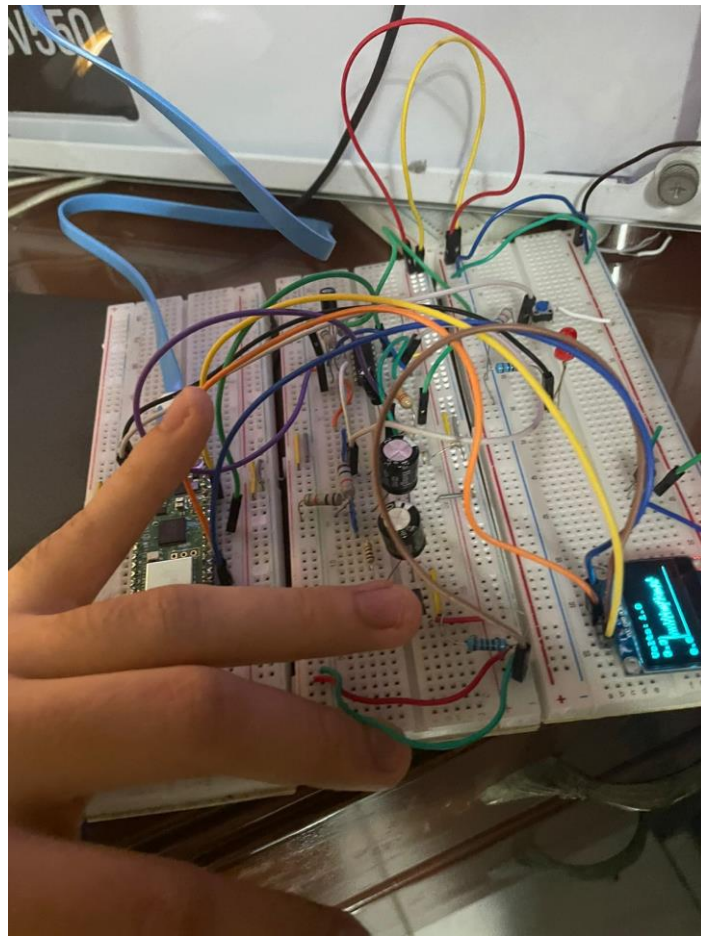
Prueba del dispositivo electrónico



El dispositivo está conectado y listo para usar, en la electrónico se puede observar todos los componentes descritos en la sección de metodología de forma física.

Figura 3.2

Toma de datos con el dispositivo realizado



Se realizó una prueba del dispositivo, una vez que ya se registró el número de cédula del paciente, se coloca el dedo sobre el sensor TCRT5000 y se activa el circuito al aplastar el botón, luego se calcula la frecuencia cardíaca del individuo y se visualiza a tiempo real en la pantalla OLED el pulso obtenido.

Figura 3.3

Frecuencia cardíaca obtenida usando un sensor digital



Se obtiene la frecuencia cardíaca del paciente utilizando un sensor digital para luego compararla con el obtenido en el dispositivo.

Figura 3.4

Datos obtenidos usando el dispositivo

```

pruebaaa.py
298     led.value(ledState)
299
300     if(showw):
301         showw=0
302         if(len(lista_t2)>0):
303             for i in range(1, len(lista_t2)-1):
304                 if lista_t2[i] > lista_t2[i - 1] and lista_t2[i] > lista_t2[i + 1]:
305                     #picos.append((lista_d2[i], lista_t2[i]))
306                     picos_d2.append(lista_d2[i])
307                     picos_t2.append(lista_t2[i])
308                     #print((str(lista_d2[i])+";" +str((lista_t2[i]))))
309
310
311             for i in range(1,len(picos_d2)-1):
312                 if i > 0:
313                     restas.append(picos_d2[i]- picos_d2[i - 1])
314                     prom_resto = sum(restas) / len(restas)
315                     prom_resto_seg = prom_resto / 1000
316                     prom_resto_min=60 / prom_resto_seg
317                     vl=prom_resto_min
318
319             print("Frecuencia:", 60 / prom_resto_seg)
320             EnviarDatos()
321             time.sleep(1)
322             prom_resto_min=0
323             picos_t2=[]
324             picos_d2=[]

```

```

Shell
>>> $!run -c $EDITOR_CONTENT

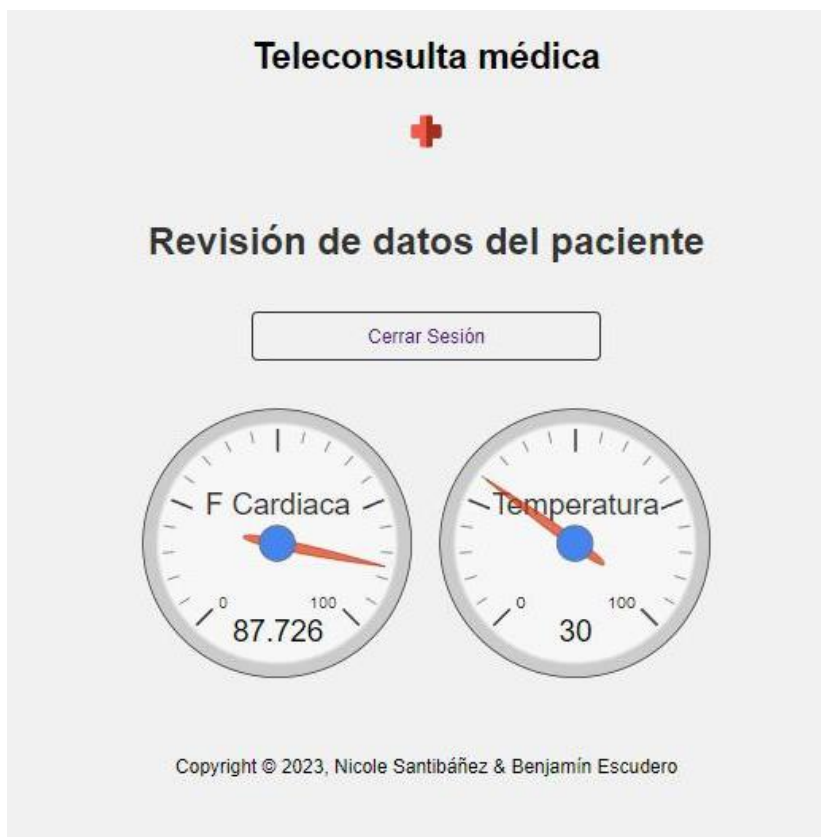
MPY: soft reboot
192.168.0.100
Frecuencia: 87.72642
Data sent successfully
Array
{
  [value1] => 87.72642
  [value2] => 30
}
New record created successfully

```

Se obtuvo una frecuencia cardíaca de 87.72 lpm, un valor muy cercano al obtenido usando el sensor digital.

Figura 3.5

Visualización de datos obtenidos en la página web



Finalmente, los datos son enviados directamente a la página web para ser visualizados por el paciente.

3.1.1.1 Análisis de resultados

Se usó un sensor digital que calcula directamente la frecuencia cardíaca para luego comparar con el valor calculado del sensor TCRT5000.

A partir de la **Tabla 3.1**, se puede observar los valores obtenidos por cada paciente en estado de reposo con su respectivo valor de frecuencia cardíaca y temperatura utilizando distintos sensores. Los pacientes: uno, dos y tres, corresponden a adultos de 23, 22 y 60 años

respectivamente, el sexo del paciente uno es masculino mientras que los otros dos pacientes son mujeres. El valor de frecuencia cardíaca de un adulto debe oscilar entre los 60 a 100 lpm, los valores según la tabla están dentro del rango. Al comparar el error relativo de la frecuencia cardíaca de todos los pacientes, este no supera el 5%, lo cual representa un gran margen de error considerablemente bajo por lo que se puede concluir que la frecuencia cardíaca obtenida por el sensor TCRT5000 tiene una exactitud adecuada, sin embargo, ese pequeño error es debido a los factores externos que alteran la sensibilidad del sensor, el movimiento involuntario y la exposición a luces externas a pesar de la protección del sensor pueden ocasionar una perturbación en la señal.

Además, al observar varios comportamientos de movimientos inusuales sobre el sensor, este entra en estado de saturación por los amplificadores operacionales, por tal razón es necesario dejar en reposo el dedo veinte segundos con anticipación antes de tomar datos erróneos.

Tabla 3.1

Comparación entre el sensor TCRT5000 y el sensor digital para medir la frecuencia cardíaca en estado de reposo

Paciente	Sensor LM35	Sensor TCRT500 (lpm)	Sensor digital de frecuencia cardíaca (lpm)	Error relativo entre la frecuencia cardíaca (%)
1	27	86	87	1.16
2	30	65	63	3.07
3	40	69	72	4.34

Tabla 3.2

Comparación entre el sensor TCRT5000 y el sensor digital para medir la frecuencia cardíaca luego de haber realizado una actividad

Paciente	Sensor LM35	Sensor TCRT500 (lpm)	Sensor digital de frecuencia cardíaca (lpm)	Error relativo entre la frecuencia cardíaca (%)
1	27	125	140	10.71
2	30	113	120	3.07
3	40	100	107	4.82

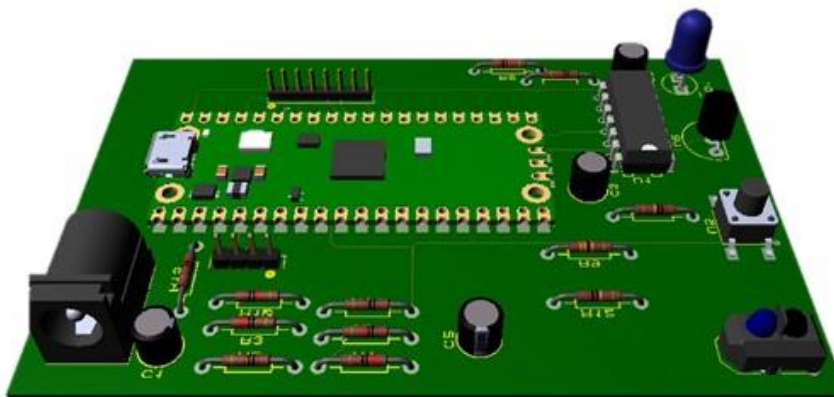
Nota. Se midió la frecuencia cardíaca y temperatura de 3 pacientes luego de un estado de actividad física

Se realizó la misma medición de los pacientes luego de haber realizado alguna actividad física produciendo un aumento en su pulso cardíaco. El error relativo del paciente uno es de 10.71%, lo cual significa un error alto comparado con los demás que resultaron ser menor al 5%, esto es ocasionado por los factores mencionados anteriormente y además es porque el paciente uno al realizar su medición, la posición y postura de brazo no eran similares al igual que cuando fue medido en el sensor digital. El brazo debe estar sobre una superficie como soporte, extendiendo el dedo sobre el sensor TCRT5000 y estar sentado sin realizar alguna actividad, cumpliendo con estos requisitos el valor medido se acerca al valor real, tal y como ocurrió con los demás pacientes.

3.1.1.2 Diseño PCB

Figura 3.6

Modelo 3D del diseño PCB realizado



Para el desarrollo del PCB se utilizó el software Proteus, para lo cual fue necesario obtener el archivo que contiene el modelo CAD de los componentes que no se encontraban incluidos en la librería de Proteus. Se colocaron los componentes como el sensor TCR5000, la botonera, el sensor de temperatura y el Raspberry en la ubicación más conveniente y para los demás componentes se utilizó la herramienta de auto posicionamiento de Proteus, luego se generaron las rutas mediante el auto ruteo para lo cual fue necesario utilizar más de una capa en la placa ya que con una sola placa no fue posible generar todas las rutas.

Adicionalmente, se ha añadido un conector de alimentación para alimentar el circuito de la PCB con 5 VDC, y se han colocado los pines sobre los que se conectará el OLED y un teclado matricial en caso de que sea necesario su uso para ampliar la funcionalidad del proyecto.

3.1.2 Programación

3.1.2.1 Interfaz Web

Se muestran las diferentes interfaces realizadas para el paciente y médico cumpliendo con las funciones mencionadas en la sección de Metodología:

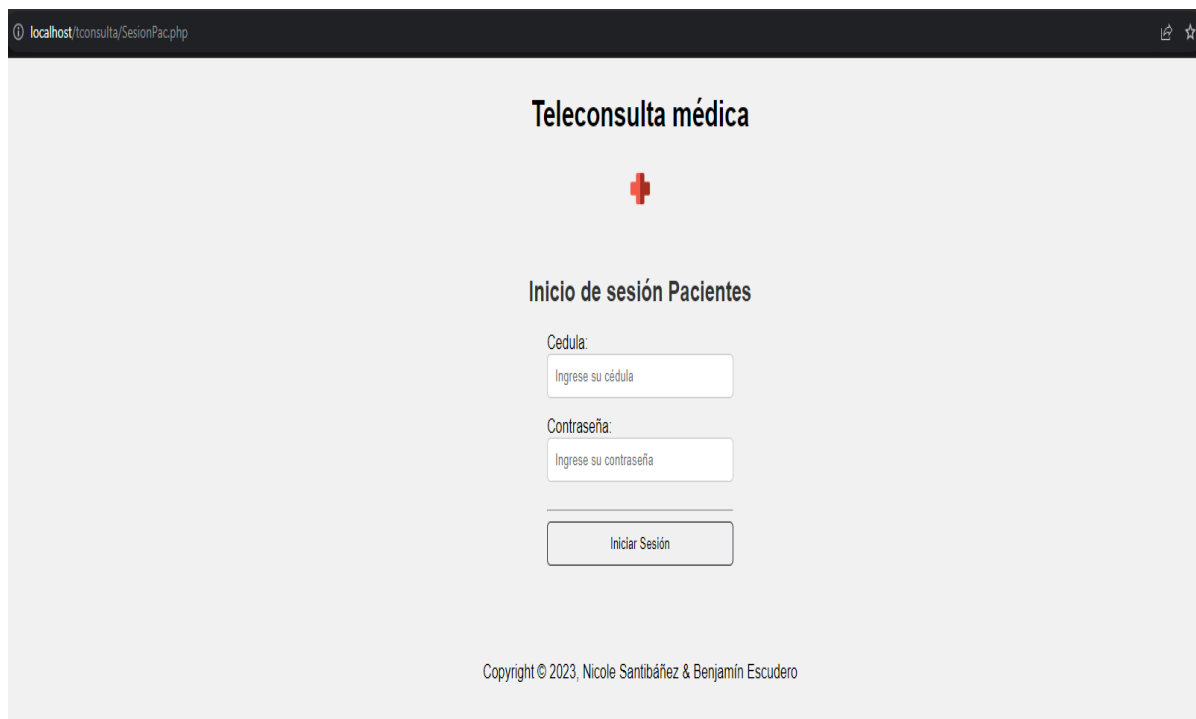
Figura 3.7

Página principal de la interfaz web




La pantalla principal de la página web, la cual el usuario podrá ingresar como paciente o médico, sin embargo, es necesario haberse creado una cuenta registrada como paciente, en el caso de ser un doctor, este ya viene con un usuario y contraseña por defecto para poder administrarla.

El icono de la cruz de ambulancia tiene la función de poder regresar siempre a la página principal.

Figura 3.8*Inicio de sesión Pacientes*


The screenshot shows a web browser window with the address bar displaying 'localhost/tconsulta/SesionPac.php'. The page content is centered and features the following elements:

- Teleconsulta médica**: The main title of the page.
- : A red cross icon representing a medical or healthcare service.
- Inicio de sesión Pacientes**: The heading for the patient login section.
- Cedula:** A label for the first input field.
- : A text input field for the patient's ID card number.
- Contraseña:** A label for the second input field.
- : A text input field for the patient's password.
- : A button to submit the login information.
- Copyright © 2023, Nicole Santibáñez & Benjamín Escudero

Para el inicio de sesión Pacientes, se requiere colocar la cédula y la contraseña creada en el registro.

Figura 3.9*Registro del paciente*

Teleconsulta médica



Registro

Cédula:

Contraseña:

Confirmar contraseña:

Nombres:

Apellidos:

Correo:

Celular:

Edad:

Copyright © 2023, Nicole Santibáñez & Benjamín Escudero

Figura 3.10*Tabla MySQL pacd*

<input type="checkbox"/>	 Editar	 Copiar	 Borrar	17	0913456789	123	Vicky	Campos	15	vicky@gmail.com	0987811221
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	18	1245885591	123	Pepe	Cevallos	20	beescude@gmail.com	0987811221
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	19	0951169275	123	Ivan	Mieles	23	ivan@espol.edu.ec	0988789443
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	20	1203615933	123	Dalila	Mendoza	45	dalilla@espol.edu.ec	0988743937
<input type="checkbox"/>	 Editar	 Copiar	 Borrar	21	09443579191	123	Carlos	Caballero	45	carlos@espol.edu.ec	0912452212

Al momento de ya haber sido registrado el paciente, se guardará dentro de la tabla pacd.

Figura 3.11

Inicio de sesión Pacientes ya una vez registrado



The screenshot shows a web browser window with the address bar displaying 'localhost/tconsulta/SesionPac.php'. The main content area has a light gray background and is titled 'Teleconsulta médica' with a red cross icon below it. Underneath, it says 'Inicio de sesión Pacientes'. There are two input fields: 'Cedula:' with the value '0951688670' and 'Contraseña:' with masked characters '.....'. Below these fields is a button labeled 'Iniciar Sesión'. At the bottom of the page, there is a copyright notice: 'Copyright © 2023, Nicole Santibáñez & Benjamín Escudero'.


Inicia sesión colocando la cédula y contraseña registrada.

Figura 3.12*Sesión ya iniciada del paciente*

Posterior al inicio de sesión, una ventana dando la bienvenida al paciente con su respectivo nombre y dos recuadros para cerrar sesión y realizar el formulario de preguntas relacionadas al estado de salud. Además, un cuadro de texto donde se visualizará si el doctor realizó algún comentario después de haber observado la ficha médica del paciente.

Figura 3.13
Formulario médico

Teleconsulta médica



Bienvenido/a Ivan

Cerrar Sesión

Responda a las siguientes preguntas

¿Sufre de alguna enfermedad como hipertensión o diabetes? ¿Cuál?

¿Sufre de alguna alergia? ¿Cuál?

¿Algún familiar sufre de algunas de las enfermedades anteriormente dichas?

¿Ha sido operado alguna vez?

En caso de haber sido operado, ¿Cuánto tiempo ha sido desde que se ha operado?

¿Toma algún medicamento?

Motivo de consulta:

¿Presenta algún tipo de dolor?

Pulsátil

Opresivo

Cólico

¿En qué parte se encuentra el dolor?

Cabeza

Tórax

Abdomen

Extremidades superior

Extremidades inferior

¿Cuál es la intensidad del dolor?

Bajo

Medio

Fuerte

¿Además del dolor, presenta algún otro síntoma? ¿Cuál?

De una descripción de su problema:

Enviar

Toma de Datos

Copyright © 2023, Nicole Santibáñez & Benjamín Escudero

Al finalizar el formulario y enviarlo, se aplasta el botón Toma de Datos para abrir una nueva pestaña y visualizar los datos obtenidos del microcontrolador a tiempo real.

Figura 3.14*Tabla MySQL preguntas*


id	cedula	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11
6	0951688670	No	No	No	No	no	No	Si	En la cabeza	Bastante fuerte	cabeza	Regularmente me da un dolor de cabeza en las ma...

Se cargan los datos de las respuestas contestadas en el formulario dentro de la tabla preguntas.

Para concluir la interfaz del paciente, se muestran los datos de frecuencia cardíaca y temperatura de forma dinámica obtenidos de la Raspberry pi pico w en la ventana DatosTiemporeal.php.

Figura 3.15*Ventana DatosTiemporeal*

Figura 3.16*Inicio de sesión Doctores*

The screenshot shows a web browser window with the address bar displaying "localhost/tconsulta/SesionDoc.php". The page content is centered and features the following elements:

- Teleconsulta médica**: The main title of the application, positioned at the top center.
- : A small red cross icon located below the title.
- Inicio de sesión Doctores**: The subtitle for the login section.
- Usuario:** A label for the username field, with the value "admindoc" entered.
- Contraseña:** A label for the password field, with masked characters "*****" entered.
- Iniciar Sesión**: A button to submit the login information.
- Copyright © 2023, Nicole Santibáñez & Benjamin Escudero**: A footer note at the bottom center.

El médico ingresa el usuario y contraseña administrativa: `admindocs`, la cual ya está creada en la tabla `docs`.

Figura 3.17*Sesión ya iniciada del doctor*

localhost/consulta/Sesioniniciadoc.php

Cerrar Sesión

Escoja el paciente:

0951169275

Ficha Médica

Información Personal

Nombres: Ivan

Apellidos: Mielles

Número de cédula: 0951169275

Correo: ivan@espol.edu.ec

Celular: 0980709443

Edad: 23

Antecedentes Médicos

Enfermedades: No

Alergias: No

Enfermedades de familiares: Si, mis padres sufren de diabetes

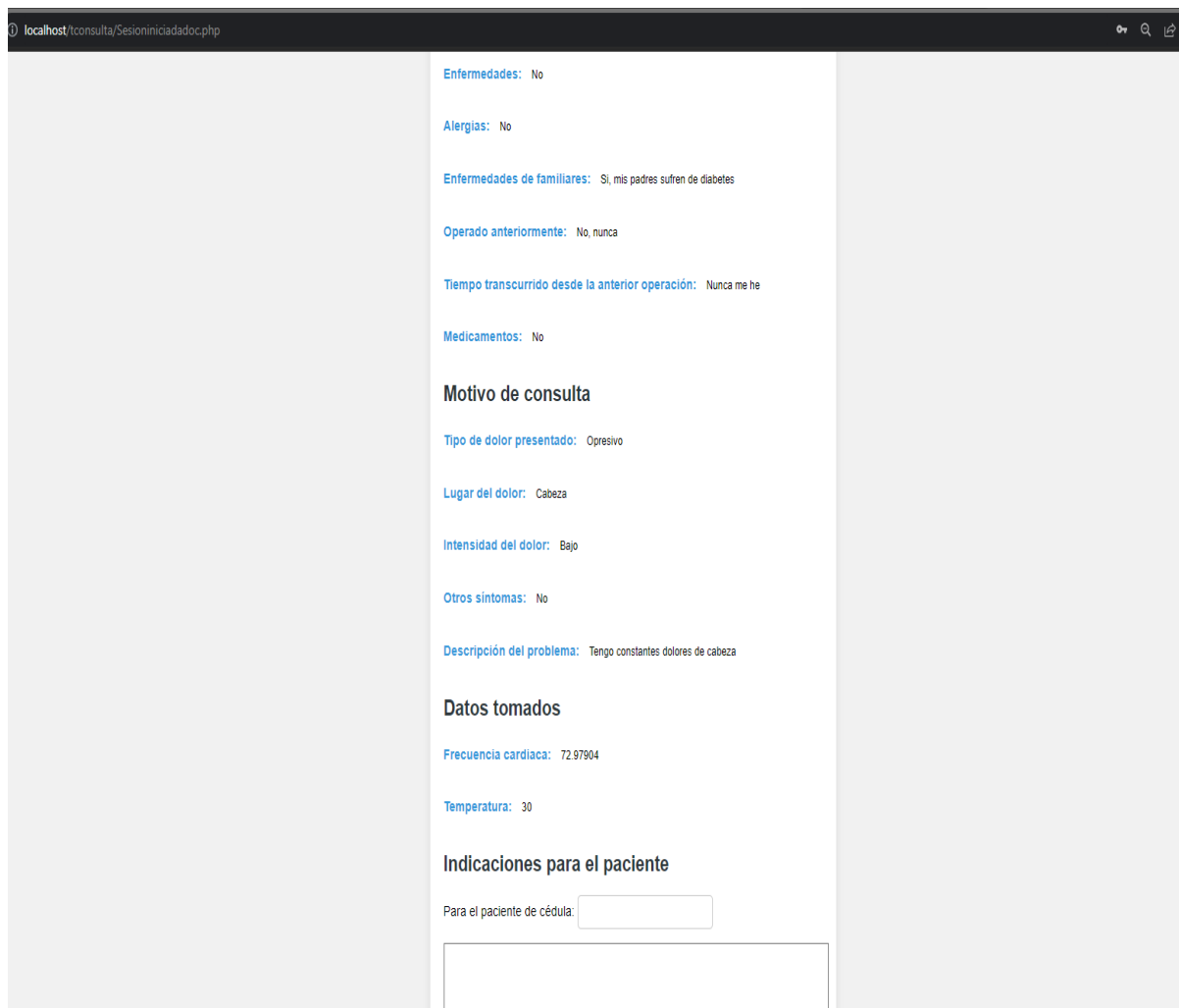
Operado anteriormente: No, nunca

Tiempo transcurrido desde la anterior operación: Nunca me he

Para visualizar la ficha médica, se deberá escoger el número de cédula del paciente deseado, aparecerán todos los datos relacionados a ese paciente en específico en una ficha médica

Figura 3.18

Vista terminada de la ventana sesión iniciada del doctor



The screenshot shows a web browser window with the address bar displaying 'localhost/tconsulta/Sesioniniciadoc.php'. The main content area is a white panel with a light gray background, containing the following information:

- Enfermedades:** No
- Alergias:** No
- Enfermedades de familiares:** Si, mis padres sufren de diabetes
- Operado anteriormente:** No, nunca
- Tiempo transcurrido desde la anterior operación:** Nunca me he
- Medicamentos:** No
- Motivo de consulta**
 - Tipo de dolor presentado:** Opresivo
 - Lugar del dolor:** Cabeza
 - Intensidad del dolor:** Bajo
 - Otros síntomas:** No
 - Descripción del problema:** Tengo constantes dolores de cabeza
- Datos tomados**
 - Frecuencia cardiaca:** 72.97904
 - Temperatura:** 30
- Indicaciones para el paciente**
 - Para el paciente de cédula:
 -

Al acabar de revisar la información del paciente, el doctor podrá enviar alguna indicación para realizar el agendamiento como mensaje de texto.

Tabla 3.3*Evaluación de la interfaz web por varios usuarios*

Usuarios	Criterios de evaluación		
	Estética	Facilidad	Fluidez
1	4	5	4
2	5	4	5
3	4	4	4
4	3	4	5

3.1.2.2 Análisis de resultados

3.1.3 Cotización de rubros

Tabla 3.4

Cotización de Rubros

Componentes	Cantidad	Precio por unidad (\$)	Total (\$)
Impresión PCB	1	15	15
Kit de 140pcs	1	3.50	3.50
LM324N	1	0.50	0.50
Resistencias	12	0.10	1.20
Capacitores	4	0.20	0.8
Fuente de alimentación 5V	1	5	5
Led	1	0.20	0.20
Botón	1	0.30	0.30
Pantalla OLED	1	3.50	3.50
Teclado Matricial	1	5	5

Raspberry pi pico w	1	15	15
Conector Jack de alimentación	1	0.50	0.50
Total			50.5

En la **Tabla** 3.4

Cotización de Rubros el costo para generar una nueva placa es de \$50.5, existen otros rubros como el costo del hosting web, los precios del hosting web y el costo de la base de datos oscilan entre \$ 2 y \$ 500 para todo tipo de aplicación. Estos costos se asumirían añadiendo un porcentaje en el margen de venta o buscando otras soluciones como apoyo mediante publicidad u otras alternativas.

Para poder generar utilidades el margen mínimo de una venta debe ser del 30% del costo, de este modo precio mínimo al que se puede vender la placa es de \$65.65 lo cual puede no ser un precio asequible para un individuo con ingresos básicos, pero se vuelve más razonable si se adquiere para ser usado en una comunidad por un grupo de personas para que puedan tener acceso a un sistema de telemedicina si no existen centros de salud médica cercanos a sus lugares de vivienda. Se debe tomar en cuenta que si se llegase a producir al por mayor estos dispositivos se podría disminuir en un poco porcentaje los costos de fabricación.

Capítulo 4

4.1 Conclusiones y recomendaciones

El proyecto realizado tiene la finalidad de ser utilizado en sectores rurales donde exista escasez de doctores y centros clínicos, por tal razón el sistema de consulta médica a distancia permite la comunicación entre paciente y médico de forma remota sin necesidad de haber un contacto físico entre ellos, brindando al doctor la información necesaria para poder realizar un diagnóstico del paciente con la frecuencia cardíaca, temperatura corporal y un formulario de preguntas contestadas por el usuario, brindando un servicio a personas con problemas médico que buscan realizarse un chequeo médico pero no cuentan por la capacidad de poder viajar a ciudades frecuentemente, perjudicando su salud.

Esto podría ser utilizado en el sector de salud pública ya que la mayoría de los centros médicos se encuentran en áreas urbanas, con esta solución se podría brindar mayor cobertura a áreas rurales además de permitir reducir el tiempo de espera para poder recibir una cita con un especialista.

4.1.1 Conclusiones

- Se logró obtener la forma de onda del pulso cardiaco en base al principio de fotoplestimografía mediante el cual fue posible calcular la frecuencia cardíaca analizando los valores máximos de la onda y el intervalo de tiempo entre ellos.
- Se diseñó una página web con conexión a una base de datos donde se almacenan los datos de los pacientes, para eso fue necesario el uso de una placa de desarrollo (Raspberry pi pico w) que sea capaz de conectarse al servidor vía wifi.

- Al realizar la prueba conceptual se pudo constatar el correcto funcionamiento de la integración entre la parte electrónica y la interfaz web para la visualización de datos.

4.1.2 Recomendaciones

- Se recomienda utilizar un diferente tipo de fotodiodo y fototransistor con menor sensibilidad para no ser alterado por factores externos, un ejemplo serían los usados en los oxímetros digitales, ya que para una aplicación médica siempre se deberá conseguir la mayor exactitud de resultados.
- Es necesario implementar un modelo físico para restringir el movimiento de dedo del paciente al momento de obtener su frecuencia cardíaca debido a que la mayoría de las personas de edad avanzada tiende a sufrir de problemas nerviosos, produciendo movimientos involuntarios que afectan el resultado calculado por el dispositivo electrónico.
- Se recomienda editar el mapa de entidad-relación de la base de datos agregando una nueva entidad “citas” donde se pueda vincular el paciente, doctor, e información emitida en la cita de forma que el paciente y el doctor puedan revisar un historial de todas las citas pasadas.

REFERENCIAS

- [1] Caeme, "Enfermedades Cardiovasculares," Cámara Argentina de Especialidades Medicinales, Julio 2020. [En línea]. Available: <https://www.caeme.org.ar/archivo-weber/enfermedades-cardiovasculares/#:~:text=Seg%C3%BAAn%20la%20OMS%2C%20las%20enfermedades,31%25%20de%20las%20muertes%20registradas.>
- [2] Espol , "Ecuador acumula pacientes con enfermedades cardiovasculares," Centro de estudios Asia-Pacífico, 23 Enero 2023. [En línea]. Available: <https://ceap.espol.edu.ec/es/content/ecuador-acumula-pacientes-con-enfermedades-cardiovasculares/#:~:text=Entre%202018%20y%202021%20el,271%20defunciones%20por%20complicaciones%20cardiovasculares.>
- [3] D. M. Bolaños, "Telemedicina en el Ecuador: un mundo de desafíos y oportunidades," Universidad Politécnica Salesiana, diciembre 2010. [En línea]. Available: <https://lagranja.ups.edu.ec/index.php/granja/article/view/484.>
- [4] Ministerio de Telecomunicaciones y de la Sociedad de la Información, "Expansión del Programa de Telemedicina," Ministerio de Salud Pública, Marzo 2013. [En línea]. Available: <https://www.telecomunicaciones.gob.ec/wp-content/uploads/2015/04/Expansi%C3%B3n-del-Programa-de-Telemedicina-Nivel-Nacional.pdf.>
- [5] J. Machado, "Hasta seis meses de espera por una cita médica en los hospitales públicos," Primicias el periodismo comprometido, Septiembre S 2022. [En línea]. Available: [https://www.primicias.ec/noticias/sociedad/pacientes-citas-médicas-iess-hospitales-publicos/.](https://www.primicias.ec/noticias/sociedad/pacientes-citas-médicas-iess-hospitales-publicos/)
- [6] R. Lucio, N. Villacrés y R. Henríquez, "Sistema de salud de Ecuador," Scielo, Enero 2011. [En línea]. Available: https://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0036-36342011000800013#:~:text=Ecuador%20tiene%20registrados%203%20847,ISSFA%2C%20ISSPOL%20y%20otras%20entidades.
- [7] I. n. d. e. y. censos, "Actividades y recursos de salud," Ecuador en cifras, 2020. [En línea]. Available: [https://www.ecuadorencifras.gob.ec/actividades-y-recursos-de-salud/.](https://www.ecuadorencifras.gob.ec/actividades-y-recursos-de-salud/)

- [8] R. Medrán, "Medición del ritmo cardíaco mediante fotopletismografía," Dpto. Teoría de la Señal y Comunicaciones Escuela Técnica Superior de Ingeniería Universidad de Sevilla , 2018. [En línea]. Available: <https://biblus.us.es/bibing/proyectos/abreproy/12461/fichero/PFC-2461-MEDRAN.pdf>.
- [9] T. Barrientos, "SlidePlayer," 2014. [En línea]. Available: <https://slideplayer.es/slide/143581/>.
- [10] Heatlwise, "Corazón normal," Cigna Healthcare, Septiembre 2022. [En línea]. Available: <https://www.cigna.com/es-us/knowledge-center/hw/corazn-normal-aa54865#:~:text=El%20lado%20derecho%20del%20coraz%C3%B3n%20recibe%20de%20las%20venas%20sangre,hacia%20el%20resto%20del%20cuerpo>
- [11] "Ciclo cardíaco," Kenhub, Julio 2023. [En línea]. Available: <https://www.kenhub.com/es/library/anatomia-es/ciclo-cardiaco-es>.
- [12] Universidad Nacional Autónoma de México, "Facmed," 2019. [En línea]. Available: <https://fisiologia.facmed.unam.mx/wp-content/uploads/2019/10/El-coraz%C3%B3n-como-bomba.pdf>.
- [13] D. J. Massó, "Quironsalud," 12 Abril 2018. [En línea]. Available: <https://www.quironsalud.es/blogs/es/corazon/causas-efectos-hipertension-arterial>.
- [14] M. Julio , "Frecuencia cardíaca," CINFASALUD, Mayo 2020. [En línea]. Available: <https://cinfasalud.cinfa.com/p/frecuencia-cardiaca/>.
- [15] C. Jhon, G. Mark, H.-G. Barrie y M. Nicole, "Fotopletismografía," OFICINA ESPAÑOLA DE PATENTES Y MARCAS ESPAÑA, [En línea]. Available: <https://patents.google.com/patent/ES2556648T3/es..>
- [16] "Diseño e implementación de un glucometro digital basado en la fotoplestimografía," Escuela Superior de Valencia, 2014. [En línea]. Available: https://riunet.upv.es/bitstream/handle/10251/49716/MEMORIAdefinitiva_14047195741583523182364041613508.pdf?sequence=3.
- [17] TECNOMED 2000, S. L., "Pulsioxímetro: Qué es, cómo funciona, uso, resultados y limitaciones," BLOGTECNOMED 2000, 2017 Julio. [En línea]. Available:

<http://blog.tecnomed2000.com/2017/07/03/pulsioximetro-que-es-como-funciona-uso-resultados-y-limitaciones/>.

- [18] P. Susana , Q. Jose, M. Felipe y L. Daniel, "Herramientas y tecnologías para el desarrollo web desde el FrontEnd al BackEnd," SEDICI Repositorio Institucional de la UNLP, Abril 2021. [En línea]. Available: <http://sedici.unlp.edu.ar/handle/10915/120476>.
- [19] Ivan, "Rockcontent," 09 Marzo 2020. [En línea]. Available: <https://rockcontent.com/es/blog/php/>.
- [20] A. Solé, J. Lambert, V.-E. Baciú, B. da Silva, J. C. Garcia-Naranjo y J. Stiens, "PPG-EduKit An Adjustable Photoplethysmography Evaluation System for Educational Activities," MDPI, [En línea]. Available: <https://gitlab.com/etrovub/wearables/publications/PPG-EduKit/-/tree/master>.
- [21] C. Juan Carlos y T. Kevin, "Desarrollo de un sistema web de citas médicas usando herramientas open source y sensores biométricos de signos vitales," Repositorio Universidad de Guayaquil, 2020. [En línea]. Available: <http://repositorio.ug.edu.ec/handle/redug/48834>.
- [22] L. Victor y O. William, "Diseño e implementación de un glucómetro no invasivo basado en la ley de Lambert Beer y longitud de onda cercana al infrarrojo (NIR), con interfaz de comunicación bluetooth a dispositivos con sistema operativo Android," UNIVERSIDAD POLITÉCNICA SALESIANA, 2014. [En línea]. Available: <http://dspace.ups.edu.ec/handle/123456789/6332>.
- [23] N. d. I. N. Rivas Muñoz, Febrero 2019. [En línea]. Available: <https://riuma.uma.es/xmlui/handle/10630/17545?show=full>.
- [24] Departamento de Fisiología, Facultad de Medicina, "El corazón como bomba: fases del ciclo cardíaco. Ecocardiograma," UNAM, [En línea]. Available: <https://fisiologia.facmed.unam.mx/wp-content/uploads/2019/10/El-coraz%C3%B3n-como-bomba.pdf>.
- [25] M. Joseph, "Causas y efectos de la hipertensión arterial," quirónsalud, Abril 2028. [En línea]. Available: <https://www.quironsalud.com/blogs/es/corazon/causas-efectos-hipertensión -arterial>.
- [26] "Pulsímetro y oxímetro con Arduino y MAX30102," Luis Llamas Ingeniería, informática y diseño, 2023. [En línea]. Available: <https://www.luisllamas.es/pulsimetro-y-oximetro-con-arduino-y-max30102/>.

- [27] M. Haque, "Beers-Lambert's law derivation," SlidesShare, 2017. [En línea]. Available: <https://www.slideshare.net/MorshedulHaque/beer-lamberts-law>.
- [28] G. Contreras, L. Rafael y B. Cesar, "Pulse oximeter with Internet data visualization," Sistemas & Telemática, 2018. [En línea]. Available: <https://doi.org/10.18046/syt.v16i45.2746>.
- [29] pixabay, "pixabay," [En línea]. Available: <https://pixabay.com/es/vectors/cuerpo-humano-sistema-circulatorio-311864/>.
- [30] L. Martín, "Automatismos Mar del Plata," 18 Marzo 2016. [En línea]. Available: <https://www.automatismos-mdq.com.ar/blog/2016/03/medicion-del-ritmo-cardiaco.html>.

APÉNDICES

APENDICE A

Código PHP en Visual Studio Code

Figura A.1

Configuración para cerrar sesión de la pagina

```

EXPLORER
  TCONSULTA
    cerrar.php
    config.php
    cruz.png
    Datosensor.php
    Datostemporeal.php
    #estilo.css
    fpac.php
    index.php
    post-esp-data.php
    recupinfo.js
    RegistroPac.php
    SesiónDoc.php
    Sesioniniciadoc.php
    Sesioniniciadpac.php
    SesiónPac.php
    Sesionprincipac.php

cerrar.php
1 <?php
2
3 require 'config.php';
4 $_SESSION = [];
5 session_unset();
6 session_destroy();
7 header("Location: index.php");
  
```

Figura A. 2

Código de la página principal

```

EXPLORER
  TCONSULTA
    cerrar.php
    config.php
    cruz.png
    Datosensor.php
    Datostemporeal.php
    #estilo.css
    fpac.php
    index.php
    post-esp-data.php
    recupinfo.js
    RegistroPac.php
    SesiónDoc.php
    Sesioniniciadoc.php
    Sesioniniciadpac.php
    SesiónPac.php
    Sesionprincipac.php

index.php
1
2 <!DOCTYPE html>
3 <html>
4   <head>
5     <title>Tesis Nicole y Benjamin</title>
6     <meta charset="UTF-8"/>
7     <meta name="viewport" content="width=device-width, initial-scale=1.0">
8     <link href="#estilo.css" rel="stylesheet">
9   </head>
10  <body>
11    <header>
12      <div id="nombre">
13        <p>Teleconsulta médica</p>
14      </div>
15    </header>
16    <h2> Iniciar Sesión </h2><br>
17    <a href="SesionDoc.php" class="button">Inicio de Sesión Doctores</a><br>
18    <a href="SesionPac.php" class="button">Inicio de Sesión Pacientes</a><br>
19    <a href="RegistroPac.php" class="button">Registro de Paciente</a><br><br><br><br>
20  </body>
21  <footer>
22    <p> Copyright &copy; 2023, Nicole Santibáñez & Benjamin Escudero</p>
23  </footer>
24 </html>
  
```

Figura A.3

Código para el registro paciente

```

EXPLORER  ...  index.php  RegistroPac.php  fpac.php  SesionIniciadoc.php  SesionIniciadapac.php
▼ CONSULTA
  cerrares.php
  config.php
  cruz.png
  Datosensor.php
  Datostemporeal.php
  # estilo.css
  fpac.php
  index.php
  post-esp-data.php
  recuipinop.js
  RegistroPac.php
  SesionDoc.php
  SesionIniciadoc.php
  SesionIniciadapac.php
  SesionPac.php
  SesionPinpac.php
  OUTLINE
1  <?php
2
3  require "config.php";
4
5  if(!empty($_SESSION["id"])){
6      header("Location: SesionIniciadapac.php");
7  }
8
9
10 if (isset($_POST["submit"])){
11     $cedula = $_POST["cedula"];
12     $contrasena = $_POST["contrasena"];
13     $ccontrasena = $_POST["ccontrasena"];
14     $nombres = $_POST["nombres"];
15     $apellidos = $_POST["apellidos"];
16     $edad = $_POST["edad"];
17     $correo = $_POST["correo"];
18     $celular = $_POST["celular"];
19
20     $duplicate = mysqli_query($conn, "SELECT * FROM pacd WHERE cedula = '$cedula'");
21     if(mysqli_num_rows($duplicate)>0){
22         echo
23         "<script> alert('Cédula ya registrada'); </script>";
24     }else{
25         if($contrasena == $ccontrasena){
26             $query = "INSERT INTO pacd (cedula, contrasena, nombres, apellidos, edad, correo, celular) VALUES ('$cedula', '$contrasena', '$nombres', '$apellidos', '$edad', '$correo', '$celular')";
27             mysqli_query($conn,$query);
28             echo
29             "<script> alert('Registro exitoso'); </script>";
30         }else{
31             echo
32             "<script> alert('No coinciden las contraseñas'); </script>";
33         }
34     }
35 }
36 }
37 }
38 }
39 }
40 }
41 <!DOCTYPE html>
42 <html>
43 <head>
44 <title>Tesis Nicole y Benja</title>
45 <meta charset="UTF-8"/>
46 <meta name="viewport" content="width=device-width, initial-scale=1.0">
47 <link href="estilo.css" rel="stylesheet">
48
49 </head>
50
51 </html>
52
53 </body>
54 </html>
55
56 </html>
57
58 </html>
59
60 </html>
61
62 </html>
63
64 </html>
65
66 </html>
67
68 </html>
69
70 </html>
71
72 </html>
73
74 </html>
75
76 </html>
77
78 </html>
79
80 </html>
81
82 </html>
83
84 </html>
85
86 </html>
87
88 </html>
89
90 </html>
91
92 </html>
93
94 </html>
95
96 </html>
97
98 </html>
99
100 </html>

```

Figura A.4

Código en PHP para guardar datos de formularios ingresados por el paciente

```

EXPLORER  ...  index.php  SesionIniciadapac.php  fpac.php  SesionIniciadoc.php  SesionIniciadapac.php
▼ CONSULTA
  cerrares.php
  config.php
  cruz.png
  Datosensor.php
  Datostemporeal.php
  # estilo.css
  fpac.php
  index.php
  post-esp-data.php
  recuipinop.js
  RegistroPac.php
  SesionDoc.php
  SesionIniciadoc.php
  SesionIniciadapac.php
  SesionPac.php
  SesionPinpac.php
  OUTLINE
1  <?php
2
3  require "config.php";
4
5  if(!empty($_SESSION["id"])){
6      $id = $_SESSION["id"];
7      $result = mysqli_query($conn, "SELECT * FROM pacd WHERE id = '$id'");
8      $row = mysqli_fetch_assoc($result);
9      header("Location: index.php");
10 }
11
12 if (isset($_POST["submit"])){
13     $cedula = $_POST["cedula"];
14     $p1 = $_POST["p1"];
15     $p2 = $_POST["p2"];
16     $p3 = $_POST["p3"];
17     $p4 = $_POST["p4"];
18     $p5 = $_POST["p5"];
19     $p6 = $_POST["p6"];
20     $p7 = $_POST["p7"];
21     $p8 = $_POST["p8"];
22     $p9 = $_POST["p9"];
23     $p10 = $_POST["p10"];
24     $p11 = $_POST["p11"];
25
26
27     $duplicate = mysqli_query($conn, "SELECT * FROM preguntas WHERE cedula = '$cedula'");
28     if(mysqli_num_rows($duplicate)>0){
29         echo
30         "<script> alert('Preguntas ya respondidas'); </script>";
31     }else{
32         $query = "INSERT INTO preguntas (cedula, p1, p2, p3, p4, p5, p6, p7, p8, p9, p10, p11) VALUES ('$cedula', '$p1', '$p2', '$p3', '$p4', '$p5', '$p6', '$p7', '$p8', '$p9', '$p10', '$p11')";
33         mysqli_query($conn,$query);
34         echo
35         "<script> alert('Eminio exitoso'); </script>";
36     }
37 }
38 }
39 }
40 }
41 }
42 }
43 }
44 <!DOCTYPE html>
45 <html>
46 <head>
47 <title>Tesis Nicole y Benjamin</title>
48 <meta charset="UTF-8"/>
49
50 </head>
51
52 </html>
53
54 </body>
55 </html>
56
57 </html>
58
59 </html>
60
61 </html>
62
63 </html>
64
65 </html>
66
67 </html>
68
69 </html>
70
71 </html>
72
73 </html>
74
75 </html>
76
77 </html>
78
79 </html>
80
81 </html>
82
83 </html>
84
85 </html>
86
87 </html>
88
89 </html>
90
91 </html>
92
93 </html>
94
95 </html>
96
97 </html>
98
99 </html>
100 </html>

```

Figura A.5

Código que maneja el inicio de sesión de los doctores

```

28 }
29 }
30 <!DOCTYPE html>
31 <html>
32 <head>
33 <title>Tesis Nicole y Benja</title>
34 <meta charset="UTF-8">
35 <meta name="viewport" content="width=device-width, initial-scale=1.0">
36 <link href="estilo.css" rel="stylesheet">
37 </head>
38 <body>
39 <header>
40 <div id="nombrepag">
41 <p> Teleconsulta médica </p>
42 <a href="index.php"><center></center></a>
43 </div>
44 </header>
45 <h2>Inicio de sesión Doctores</h2>
46 <form action="" name="formulario" method="POST">
47 <div>
48 <div>
49 <div>
50 <div>
51 <div>
52 <div>
53 <div>
54 <div>
55 <div>
56 <div>
57 <div>
58 <div>
59 <div>
60 <div>
61 <div>
62 <div>
63 <div>
64 <div>
65 <div>
66 <div>
67 <div>
68 <div>
69 <div>
70 <div>
71 <div>
72 <div>
73 </div>

```

Figura A.6

Código para mostrar al doctor la información del paciente seleccionado

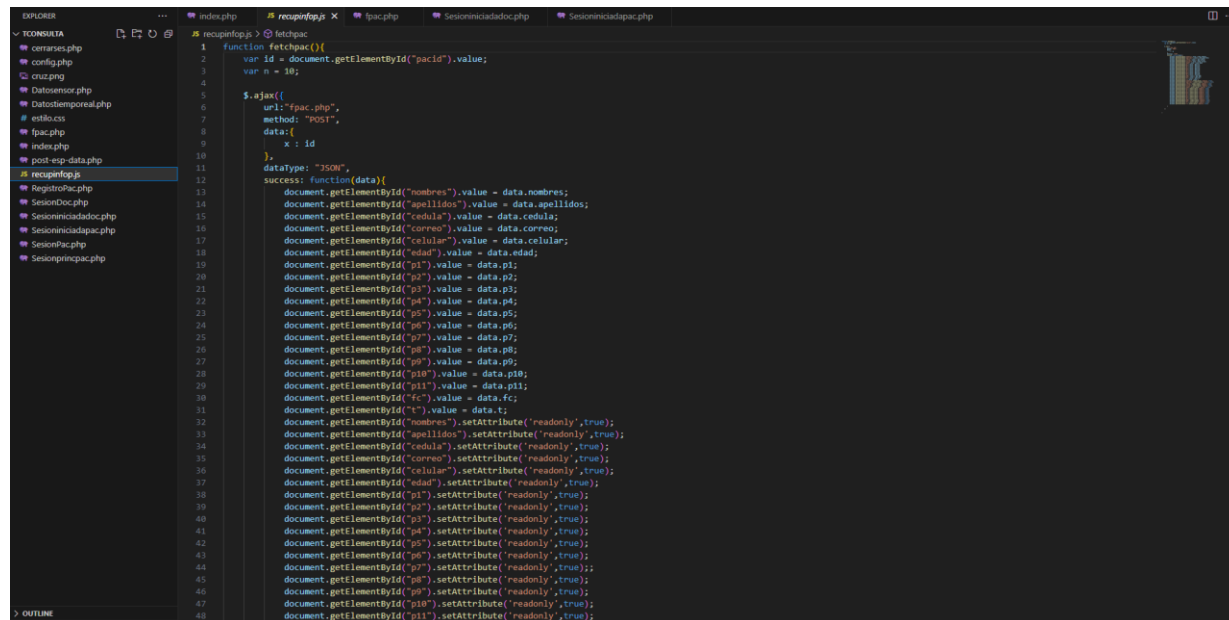
```

1 <?php
2 require 'config.php';
3 $k = $_POST['x'];
4 $sql = "SELECT * from pacid P, preguntas R, sensordata D where P.cedula={\$k} AND R.cedula={\$k} AND D.cedula={\$k}";
5 $res = mysql_query($com, $sql);
6
7
8
9 while($rows = mysql_fetch_array($res)){
10 $data['nombres'] = $rows['nombres'];
11 $data['apellidos'] = $rows['apellidos'];
12 $data['cedula'] = $rows['cedula'];
13 $data['edad'] = $rows['edad'];
14 $data['correo'] = $rows['correo'];
15 $data['celular'] = $rows['celular'];
16 $data['p1'] = $rows['p1'];
17 $data['p2'] = $rows['p2'];
18 $data['p3'] = $rows['p3'];
19 $data['p4'] = $rows['p4'];
20 $data['p5'] = $rows['p5'];
21 $data['p6'] = $rows['p6'];
22 $data['p7'] = $rows['p7'];
23 $data['p8'] = $rows['p8'];
24 $data['p9'] = $rows['p9'];
25 $data['p10'] = $rows['p10'];
26 $data['p11'] = $rows['p11'];
27 $data['fc'] = $rows['value1'];
28 $data['t'] = $rows['value2'];
29
30
31 }
32
33
34
35 echo json_encode($data);
36
37
38 }>

```

Figura A.7

Código para transformar el dato extraído en la figura A7 a formato Json



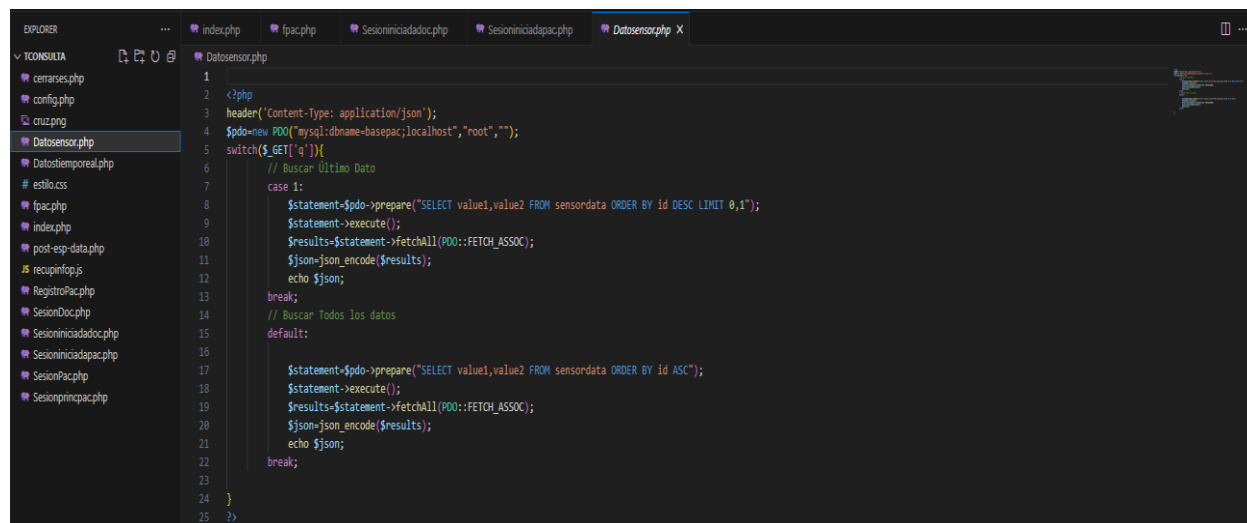
```

1 function fetchpac(){
2     var id = document.getElementById("pacid").value;
3     var n = 10;
4
5     $.ajax({
6         url:"fpac.php",
7         method: "POST",
8         data:{
9             x : id
10        },
11        dataType: "JSON",
12        success: function(data){
13            document.getElementById("nombres").value = data.nombres;
14            document.getElementById("apellidos").value = data.apellidos;
15            document.getElementById("cedula").value = data.cedula;
16            document.getElementById("correo").value = data.correo;
17            document.getElementById("celular").value = data.celular;
18            document.getElementById("edad").value = data.edad;
19            document.getElementById("p1").value = data.p1;
20            document.getElementById("p2").value = data.p2;
21            document.getElementById("p3").value = data.p3;
22            document.getElementById("p4").value = data.p4;
23            document.getElementById("p5").value = data.p5;
24            document.getElementById("p6").value = data.p6;
25            document.getElementById("p7").value = data.p7;
26            document.getElementById("p8").value = data.p8;
27            document.getElementById("p9").value = data.p9;
28            document.getElementById("p10").value = data.p10;
29            document.getElementById("p11").value = data.p11;
30            document.getElementById("fc").value = data.fc;
31            document.getElementById("t").value = data.t;
32            document.getElementById("nombres").setAttribute("readonly",true);
33            document.getElementById("apellidos").setAttribute("readonly",true);
34            document.getElementById("cedula").setAttribute("readonly",true);
35            document.getElementById("correo").setAttribute("readonly",true);
36            document.getElementById("celular").setAttribute("readonly",true);
37            document.getElementById("edad").setAttribute("readonly",true);
38            document.getElementById("p1").setAttribute("readonly",true);
39            document.getElementById("p2").setAttribute("readonly",true);
40            document.getElementById("p3").setAttribute("readonly",true);
41            document.getElementById("p4").setAttribute("readonly",true);
42            document.getElementById("p5").setAttribute("readonly",true);
43            document.getElementById("p6").setAttribute("readonly",true);
44            document.getElementById("p7").setAttribute("readonly",true);
45            document.getElementById("p8").setAttribute("readonly",true);
46            document.getElementById("p9").setAttribute("readonly",true);
47            document.getElementById("p10").setAttribute("readonly",true);
48            document.getElementById("p11").setAttribute("readonly",true);

```

Figura A.8

Código para obtener el último dato de mediciones biométricas registrado en la base de datos.



```

1
2 <?php
3 header("Content-Type: application/json");
4 $pdo=new PDO("mysql:dbname=basepac;localhost","root","");
5 switch($_GET["q"]){
6     // Buscar Último Dato
7     case 1:
8         $statement=$pdo->prepare("SELECT value1,value2 FROM sensordata ORDER BY id DESC LIMIT 0,1");
9         $statement->execute();
10        $results=$statement->fetchAll(PDO::FETCH_ASSOC);
11        $json=json_encode($results);
12        echo $json;
13        break;
14    // Buscar Todos los datos
15    default:
16
17        $statement=$pdo->prepare("SELECT value1,value2 FROM sensordata ORDER BY id ASC");
18        $statement->execute();
19        $results=$statement->fetchAll(PDO::FETCH_ASSOC);
20        $json=json_encode($results);
21        echo $json;
22        break;
23
24 }
25 ?>

```

Figura A.9

Código realizado para mostrar las mediciones recibidas con el código de la figura A 9

```

1 <?php
2 require 'config.php';
3
4 if(!empty($_SESSION['id'])){
5     $id = $_SESSION['id'];
6     $result = mysql_query($conn, "SELECT * FROM pacd WHERE id = '$id'");
7     $row = mysql_fetch_assoc($result);
8 }else{
9     header("Location: index.php");
10 }
11
12 ?>
13 <!DOCTYPE html>
14 <html>
15 <head>
16 <title>Tesis Nicole y Benjamín</title>
17 <meta charset="UTF-8">
18 <meta name="viewport" content="width=device-width, initial-scale=1.0">
19 <link href="estilo.css" rel="stylesheet">
20 <script src="https://ajax.googleapis.com/ajax/libs/jquery/2.2.0/jquery.min.js"></script>
21 <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
22 <script type="text/javascript">
23     google.charts.load({current: {'packages':['gauge']}});
24     google.charts.setOnLoadCallback(drawChart);
25     function drawChart() {
26
27         var data = google.visualization.arrayToDataTable([
28             ['Label', 'Value'],
29             ['Cardiaca', 0],
30             ['Temperatura', 0]
31         ]);
32
33         var options = {
34             min: 0, max:120,
35             width: 500, height: 500,
36             minorTicks: 5
37         };
38
39         var chart = new google.visualization.Gauge(document.getElementById('Medidores'));
40
41         chart.draw(data, options);
42     }
43 }

```

Figura A.10

Código para insertar todos los datos recibidos a la base de datos

```

1 <?php
2 $servername = "localhost";
3 $username = "root";
4 $password = "";
5 $dbname = "basepac";
6
7 $api_key_value = "12345";
8
9 $api_key= $value1 - $value2 - $value3 = "";
10 print_r($_POST);
11
12 if ($_SERVER["REQUEST_METHOD"] == "POST") {
13     $value1 = test_input($_POST["value1"]);
14     $value2 = test_input($_POST["value2"]);
15     $value3 = test_input($_POST["value3"]);
16
17     // Create connection
18     $conn = new mysqli($servername, $username, $password, $dbname);
19     // Check connection
20     if ($conn->connect_error) {
21         die("Connection failed: " . $conn->connect_error);
22     }
23
24     $sql = "INSERT INTO sensordata (value1, value2, value3) VALUES ('" . $value1 . "', '" . $value2 . "', '" . $value3 . "')";
25
26     if ($conn->query($sql) === TRUE) {
27         echo "New record created successfully";
28     } else {
29         echo "Error: " . $sql . "<br>" . $conn->error;
30     }
31     $conn->close();
32 }
33
34
35
36
37 else {
38     echo "No data posted with HTTP POST.";
39 }
40
41 function test_input($data) {
42     $data = trim($data);
43     $data = stripslashes($data);
44     $data = htmlspecialchars($data);
45     return $data;
46 }

```

Figura A.11

Código donde se muestra la interfaz del doctor luego del inicio de sesión, donde este puede seleccionar a un paciente y revisar su información.

```

43 <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.0/jquery.min.js"></script>
44
45 </head>
46
47 <body>
48
49 <header>
50 <div id="nombrepag">
51 <p>Teleconsulta médica</p>
52 <a href="index.php"><center></center></a>
53 </div>
54 </header>
55 <br> Bienvenido Dr </br>
56 <a href="cerrarses.php" class="button">Cerrar Sesión</a><br>
57
58 Elige el paciente:
59 <select name="pacid" id="pacid" onchange="fetchpac()" style="width: 300px">
60 <?php while( $rows = mysqli_fetch_array($res)){
61 >
62 <option value = "<?php echo $rows['cedula']; ?>"><?php echo $rows['cedula']; ?> </option>
63
64 </?php
65 >
66 </select>
67
68
69
70
71
72 <div class="ficha">
73 <div class="header1">
74 <h3>Ficha Médica</h3>
75 </div>
76 <div class="content">
77 <div class="profile">
78 <h4>Información Personal</h4>
79 <div class="profile-details">
80 <div class="detail">
81 <span class="label">Nombres:</span>
82 <span class="value"><input type="text" class="sinborde" name="nombres" id="nombres" style="width: 200px"/></span>
83 </div>
84 <div class="detail">
85 <span class="label">Apellidos:</span>
86 <span class="value"><input type="text" class="sinborde" name="apellidos" id="apellidos" style="width: 200px"/></span>
87 </div>
88 <div class="detail">
89 <span class="label">Número de cédula:</span>
90 <span class="value"><input type="text" class="sinborde" name="cedula" id="cedula" style="width: 200px"/></span>

```

Figura A.12

En esta sección se muestra el código CSS que permite brindar estilo a la página web.

```

1 body {
2   font-family: Arial, sans-serif;
3   margin: 0;
4   padding: 0;
5   display: flex;
6   flex-direction: column;
7   align-items: center;
8   justify-content: center;
9   background-color: #f1f1f1;
10 }
11
12 header {
13   font-size: 30px;
14   font-weight: bold;
15   margin-bottom: 30px;
16 }
17
18 .ficha {
19   width: 600px;
20   margin: 20px auto;
21   background-color: #fff;
22   box-shadow: 0px 0px 10px #000000;
23   border-radius: 10px;
24   overflow: hidden;
25 }
26
27
28 .header1 {
29   background-color: #3498db;
30   color: #fff;
31   padding: 20px;
32   text-align: center;
33 }
34
35 .sinborde {
36   border: 0;
37 }
38
39 .content {
40   padding: 20px;
41 }
42
43 h1, h2 {
44   color: #e333;
45 }

```

APENDICE B

Código del microcontrolador con micropython

Figura B.1

Importación de librerías, declaración de variables

```

1 from machine import Pin, I2C, ADC
2 from utime import sleep_ms
3 from ssd1306 import SSD1306_I2C
4 import time
5 import framebuf
6 import network
7 import urequests
8
9
10 col_list=[2,3,4,5]
11 row_list=[6,7,8,9]
12 led = machine.Pin(0, machine.Pin.OUT)
13 button = machine.Pin(1, machine.Pin.IN)
14 ledState = False
15 buttonState = False
16 buttonValue = False
17 lastButtonState = False
18 lastDebounceTime = 0
19 debounceDelay = 50
20 med = False
21 lista_t2 = []
22 lista_d2=[]
23 picos_t2=[]
24 picos_d2=[]
25 restas = []
26 prom_resto_seg=0
27 alpha = 0.5
28 S=0
29 lastTime=0
30 showw=0
31 cedula_value = ""
32 cedula_obtenida = False
33 for x in range(0,4):
34     row_list[x]=Pin(row_list[x], Pin.OUT)
35     row_list[x].value(1)
36
37
38 for x in range(0,4):
39     col_list[x] = Pin(col_list[x], Pin.IN, Pin.PULL_UP)
40
41 key_map=[["0","#", "0", "*"],\
42          ["C", "9", "8", "7"],\
43          ["B", "6", "5", "4"],\
44          ["A", "3", "2", "1"]]
45
46 def Keypad4x4Read(cols,rows):
47     for r in rows:
48         r.value(0)

```

Figura B.2

Definición de funciones para el teclado matricial

```
pruebas.py *
45
46 def Keypad4x4Read(cols,rows):
47     for r in rows:
48         r.value(0)
49         result=[cols[0].value(),cols[1].value(),cols[2].value(),cols[3].value()]
50         if min(result)!=0:
51             key=key_map[int(rows.index(r))][int(result.index(0))]
52             r.value(1) # manages key kept pressed
53             return(key)
54             r.value(1)
55
56 def get_cedula():
57     global cedula_value, cedula_obtenida # Accedemos a las variables globales
58     cedula_digits = []
59     while not cedula_obtenida and len(cedula_digits) < 10: # Mientras no se haya obtenido una cedula
60         key = Keypad4x4Read(col_list, row_list)
61         if key is not None and key.isdigit():
62             cedula_digits.append(key)
63             print("Cedula parcial:", "".join(cedula_digits))
64             time.sleep(0.3)
65             if len(cedula_digits) == 10:
66                 cedula_value = "".join(cedula_digits)
67                 cedula_obtenida = True
68                 return cedula_value
69 # Get the network interface
70 interface = network.WLAN(network.STA_IF)
71
72 # Connect to the network
73 interface.active(True)
74 interface.connect("Benja", "Beec2000")
75
76 # Wait until connected
77 while not interface.isconnected():
78     pass
79
80 # Print the IP address
81 print(interface.ifconfig()[0])
82
83 # Get the current time
84 v1=0
85 v2=30
86 v3=0
87 # Create a JSON payload
88
89
90 def EnviarDatos():
91     data= "&value1=" + str(v1) + "&value2=" + str(v2) + "&value3=" + str(v3)
92
```

Figura B.3

Definición de función para enviar datos obtenidos a la base de datos y graficar en el OLED

```
pruebas.py *
89
90 def EnviarDatos():
91     data= "&value1=" + str(v1) + "&value2=" + str(v2) + "&value3=" + str(v3)
92
93     headers = {
94         "Content-Type": "application/x-www-form-urlencoded" }
95     response=urequests.post("http://192.168.0.104/tconsulta/post-esp-data.php", data=data, json=None, headers=headers)
96 # Check the response status code
97 if response.status_code == 200:
98     # The data was sent successfully
99     print("Data sent successfully")
100     print(response.text)
101 else:
102     # The data was not sent successfully
103     print("Error sending data:", response.status_code)
104     print(response.text)
105
106 def button_pressed(arg):
107     global buttonState
108     buttonState = not buttonState
109
110 def plot_time(yt, t, x, y, var = [0.0,3.3], vpts=[25, 16, 40], hpts = [25, 55, 112]):
111     #Axis
112     oled.vline(vpts[0], vpts[1], vpts[2], 1) #x, y, h
113     oled.hline(hpts[0], hpts[1], hpts[2], 1) #x, y, w
114     oled.text(str(round(var[0],1)), vpts[0]-25, hpts[1]-5)
115     oled.text(str(round(var[1],1)), vpts[0]-25, vpts[1])
116     #y - axis
117     y[1] = int((yt-var[0])/(var[1]-var[0]) * (vpts[1]-hpts[1]) + hpts[1]) #Interpolation
118     if t < hpts[2] - hpts[0]:
119         x[1] = x[0]+1
120     else:
121         x[1] = hpts[2]
122     #Plot the line
123     oled.line(x[0],y[0],x[1],y[1],1)
124     oled.show()
125
126 #Indate past values
127 y[0] = y[1]
128 x[0] = x[1]
129
130
131 #If you have already reached the end of the graph then ...
132 if t > hpts[2] - hpts[0]:
133     #Erases the first few pixels of the graph and the y-axis.
134     oled.fill_rect(vpts[0],vpts[1],2,vpts[2],0)
135     #Clears the entire y-axis scale
136     oled.fill_rect(vpts[0]-25, vpts[1]-5, vpts[0]-vpts[1], vpts[2]-5, 0)
```


Figura B.4

Código que se ejecuta la primera vez que se corre por primera vez el programa

```

Step over [F8]
pruebas.py
144     else:
145         t += 1
146
147     return t,x,y
148     |
149
150 if __name__ == '__main__':
151     led.off()
152     button.irq(trigger=machine.Pin.IRQ_FALLING, handler=button_pressed)
153     WIDTH = 128
154     HEIGHT = 64
155     FACTOR = 3.3 / (65535)
156
157     PLACA = True #True: Raspberry Pi Pico, False: ESP8266
158
159     if PLACA:
160         i2c = I2C(1, scl = Pin(19), sda = Pin(18), freq = 200000)
161         pot = ADC(28)
162     else:
163         i2c = I2C(scl = Pin(5), sda = Pin(4))
164         pot = ADC(0)
165
166     oled = SSD1306_I2C(WIDTH, HEIGHT, i2c)
167
168     #Global Variables
169     t = 0
170     y = [55, 55]
171     x = [25, 25]
172
173     oled.fill(0)
174     oled.text("Ingrese", 35, 0)
175     oled.text("su", 50, 20)
176     oled.text("cedula", 35, 40)
177     oled.show()
178     sleep_ms(600)
179     oled.fill(0)
180
181
182
183
184     while True:
185
186
187         buttonValue = button.value()
188         if (button.value()):
189             buttonValue= True
190             #buttonState= not buttonState

```

Figura B.5

Loop infinito que se ejecuta en el microcontrolador donde se detecta si se ha aplastado el botón para empezar con la medición de datos, se procede a leer los datos del sensor que se muestran en tiempo real a través del OLED

```

pruebas.py
184     while True:
185         buttonValue = button.value()
186         if (button.value()):
187             buttonValue= True
188             #buttonState= not buttonState
189
190         else:
191             buttonValue= False
192
193         if buttonValue != lastButtonState:
194             lastDebounceTime = time.ticks_ms()
195             #if (time.ticks_ms() - lastDebounceTime) > debounceDelay:
196                 #if buttonValue != buttonState:
197                     buttonState = buttonValue
198
199
200             if buttonState == True:
201                 ledState = True
202                 med = True
203                 lista_t2=[]
204                 lista_d2=[]
205
206
207         led.value(ledState)
208         lastButtonState = buttonValue
209
210         while med:
211             value=get_cedula()
212             if len(str(value))==10:
213                 print("Cédula completa:", value)
214                 v3=value
215                 oled.text("cedula:", 35, 0)
216                 oled.text(str(v3), 50, 20)
217                 oled.show()
218                 sleep_ms(800)
219                 oled.fill(0)
220
221                 volts = pot.read_u16() * FACTOR
222                 tiempo=time.ticks_ms()
223                 t,x,y = plot_time(volts,t,x,y)
224                 oled.fill_rect(0,0,120,15,0)
225                 oled.text("Volts: ", 0, 0)
226                 oled.text(str(round(volts,1)), 52, 0)
227                 oled.show()
228                 sleep_ms(100)
229                 S = (alpha * volts) + ((1 - alpha) * S)
230
231                 if(len(lista_t2)<70):
232                     lista_t2.append(S)

```

Figura B.6*Código para calcular la frecuencia cardíaca*

```

229
230     if(len(lista_t2)<70):
231         lista_t2.append(S)
232
233         lista_d2.append(tiempo)
234         ledState = False
235         showw=0;
236     else:
237
238         med=0;
239         showw=1;
240
241
242
243     led.value(ledState)
244
245     if(showw):
246         showw=0
247         if(len(lista_t2)>0):
248             for i in range(1, len(lista_t2)-1):
249                 if lista_t2[i] > lista_t2[i - 1] and lista_t2[i] > lista_t2[i + 1]:
250                     picos_d2.append(lista_d2[i])
251                     picos_t2.append(lista_t2[i])
252
253
254
255         for i in range(1,len(picos_d2)-1):
256             if i > 0:
257                 restas.append(picos_d2[i]- picos_d2[i - 1])
258         prom_resto = sum(restas) / len(restas)
259         prom_resto_seg = prom_resto / 1000
260         prom_resto_min=60 / prom_resto_seg
261         v1=prom_resto_min
262
263         print("Frecuencia:", 60 / prom_resto_seg)
264         oled.fill(0)
265         oled.text("Frecuencia", 30, 0)
266         oled.text("cardiaca", 35, 20)
267         oled.text(str(60 / prom_resto_seg), 25, 40)
268         oled.show()
269         sleep_ms(800)
270         oled.fill(0)
271         EnviarDatos()
272         time.sleep(1)
273         prom_resto_min=0
274         picos_t2=[]
275         picos_d2=[]
276         restas = []

```

APENDICE C

Código del microcontrolador con micropython

Figura C.1

Evidencia de consulta médica y pruebas de funcionamiento del dispositivo electrónico



Figura C.2

Pruebas con pulsómetro digital



Figura C.3

Pruebas con pulsómetro digital



Figura C.4

Pruebas con pulsómetro digital y prototipo

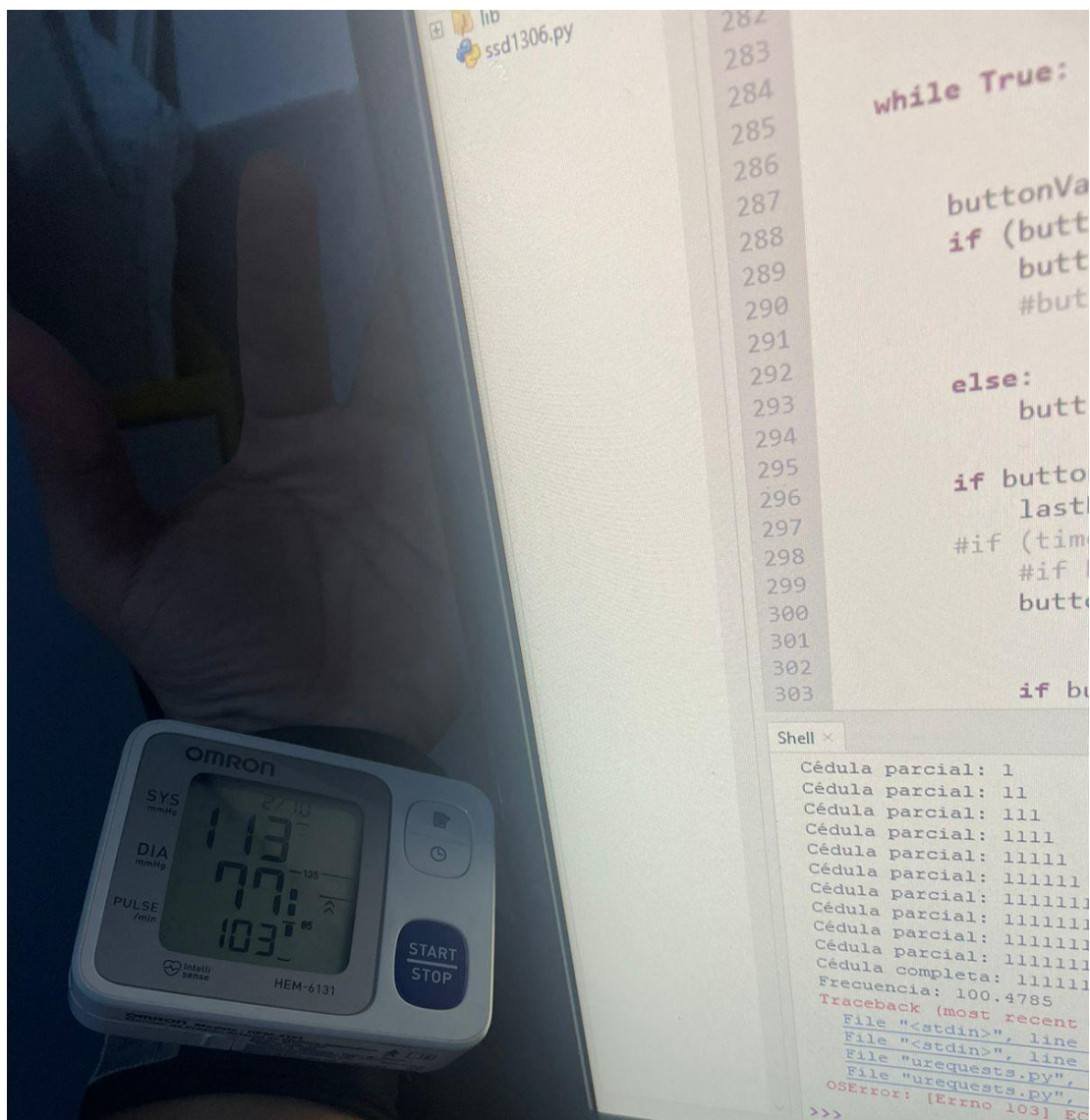


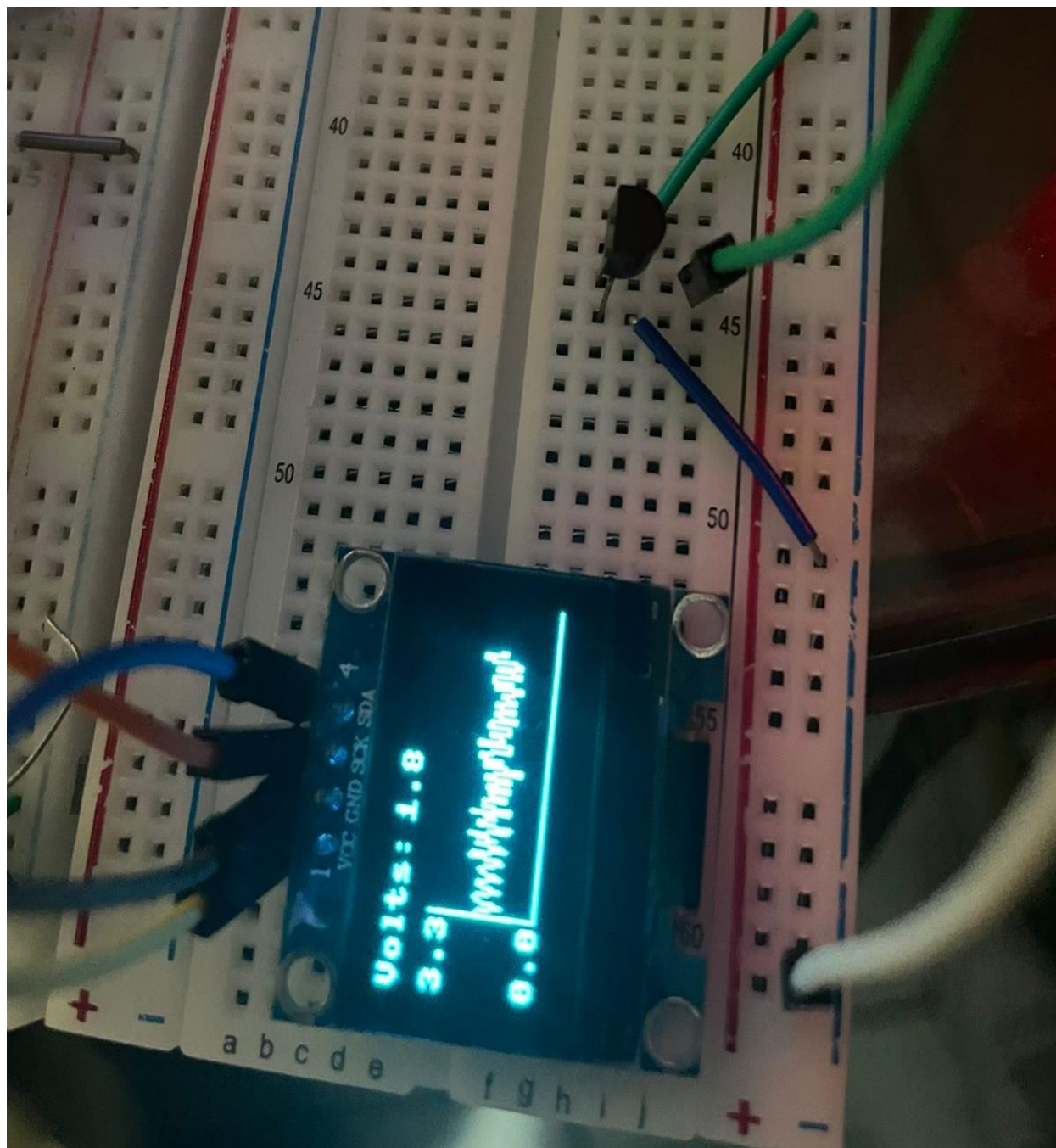
Figura C.5

Pruebas con pulsómetro digital y prototipo



FiguraC.6

Gráfica la onda obtenida en el OLED



APENDICE D

Datasheets

FiguraD.1

Datasheet del sensor LM35

LM35

Precision Centigrade Temperature Sensors

General Description

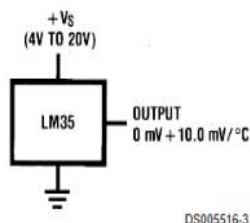
The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of $\pm 1/4^\circ\text{C}$ at room temperature and $\pm 3/4^\circ\text{C}$ over a full -55 to $+150^\circ\text{C}$ temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only $60\ \mu\text{A}$ from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a -55° to $+150^\circ\text{C}$ temperature range, while the LM35C is rated for a -40° to $+110^\circ\text{C}$ range (-10° with improved accuracy). The LM35 series is available pack-

aged in hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-220 package.

Features

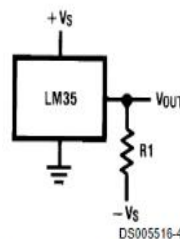
- Calibrated directly in ° Celsius (Centigrade)
- Linear $+10.0\ \text{mV}/^\circ\text{C}$ scale factor
- 0.5°C accuracy guaranteeable (at $+25^\circ\text{C}$)
- Rated for full -55° to $+150^\circ\text{C}$ range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than $60\ \mu\text{A}$ current drain
- Low self-heating, 0.08°C in still air
- Nonlinearity only $\pm 1/4^\circ\text{C}$ typical
- Low impedance output, $0.1\ \Omega$ for 1 mA load

Typical Applications



DS005516-3

FIGURE 1. Basic Centigrade Temperature Sensor
($+2^\circ\text{C}$ to $+150^\circ\text{C}$)



DS005516-4

Choose $R_1 = -V_S/50\ \mu\text{A}$
 $V_{\text{OUT}} = +1,500\ \text{mV}$ at $+150^\circ\text{C}$
 $= +250\ \text{mV}$ at $+25^\circ\text{C}$
 $= -550\ \text{mV}$ at -55°C

FIGURE 2. Full-Range Centigrade Temperature Sensor

Figure D.2

Datasheet LM324n

Philips Semiconductors

Product specification

Low power quad op amps**LM124/224/324/324A/
SA534/LM2902****DESCRIPTION**

The LM124/SA534/LM2902 series consists of four independent, high-gain, internally frequency-compensated operational amplifiers designed specifically to operate from a single power supply over a wide range of voltages.

UNIQUE FEATURES

In the linear mode, the input common-mode voltage range includes ground and the output voltage can also swing to ground, even though operated from only a single power supply voltage.

The unity gain crossover frequency and the input bias current are temperature-compensated.

FEATURES

- Internally frequency-compensated for unity gain
- Large DC voltage gain: 100dB
- Wide bandwidth (unity gain): 1MHz (temperature-compensated)
- Wide power supply range Single supply: $3V_{DC}$ to $30V_{DC}$ or dual supplies: $\pm 1.5V_{DC}$ to $\pm 15V_{DC}$
- Very low supply current drain: essentially independent of supply voltage (1mW/op amp at $+5V_{DC}$)
- Low input biasing current: $45nA_{DC}$ (temperature-compensated)
- Low input offset voltage: $2mV_{DC}$ and offset current: $5nA_{DC}$
- Differential input voltage range equal to the power supply voltage
- Large output voltage: $0V_{DC}$ to $V_{DC}-1.5V_{DC}$ swing

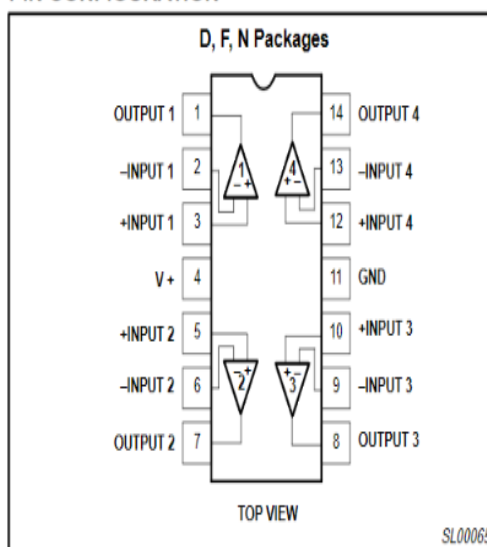
PIN CONFIGURATION

Figure 1. Pin Configuration

Figura D.3

Datasheet del sensor TCR5000



TCRT5000(L)

Vishay Telefunken

Reflective Optical Sensor with Transistor Output

Description

The TCRT5000(L) has a compact construction where the emitting-light source and the detector are arranged in the same direction to sense the presence of an object by using the reflective IR beam from the object. The operating wavelength is 950 nm. The detector consists of a phototransistor.



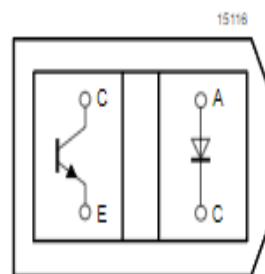
94 9442

Applications

- Position sensor for shaft encoder
- Detection of reflective material such as paper, IBM cards, magnetic tapes etc.
- Limit switch for mechanical motions in VCR
- General purpose – wherever the space is limited

Features

- Snap-in construction for PCB mounting
- Package height: 7 mm
- Plastic polycarbonate housing construction which prevents crosstalk
- L = long leads
- Current Transfer Ratio (CTR) of typical 10%



Top view