



**FACULTAD DE CIENCIAS E INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA
ELECTRÓNICA CON MENCIÓN EN
TELECOMUNICACIONES**

TESIS

Diseño e implementación de un sistema de medición y pronóstico de radiación ultravioleta utilizando Internet de las cosas y Machine Learning

PRESENTADO POR

Mujaico Mariano, Anderson Abel

ASESOR

Soto Cordova, Martin Moises

Los Olivos, 2020

Dedicatoria

Dedico este trabajo con mucho esmero a mis padres y amistades que de una forma u otra brindaron su gran apoyo durante mi formación profesional.

Resumen

La radiación ultravioleta (UV) que incide sobre la superficie terrestre es un componente importante para la formación de la fotosíntesis en las plantas y contribuye en la formación de la vitamina D en los seres humanos. Sin embargo, elevados niveles de esta radiación y sobre todo una exposición prolongada, sin protección apropiada, puede llevar a un riesgo para la salud humana; generando cáncer de piel, alteraciones oculares y otros. Por ello, con la finalidad de asociar los riesgos sea establecido rangos o niveles del denominado Índice de Radiación Ultravioleta (IUV). La presente tesis consiste en el diseño y la implementación de un sistema electrónico para realizar la medición y el pronóstico de la evolución de los niveles de la radiación ultravioleta, utilizando para ello la tecnología de Internet de las cosas (IoT) y los algoritmos de *Machine Learning* (ML). Para ello, se utiliza un sensor especializado para medir la intensidad de la radiación UV y entregar una señal analógica correspondiente a la intensidad a un microcontrolador con capacidad de comunicaciones de datos. Contando así con una plataforma de adquisición y transmisión de datos por Internet a un servidor web de IoT en la nube donde se almacena los datos en determinados periodos prefijados y se analizan mediante un algoritmo de modelado predictivo. Finalmente, cabe señalar que bajo este proyecto se obtiene un pronóstico a corto plazo sobre el índice de radiación ultravioleta, para alertar y tomar medidas de prevención y así evitar o reducir algunos problemas de salud ocasionados por esta radiación.

Palabras claves- Índice de Radiación Ultravioleta (IUV), Internet de las cosas (IoT), WeMos D1 mini ESP8266, *Machine Learning* (ML), algoritmo de modelado predictivo.

Abstract

Ultraviolet radiation (UV) incident on the earth's surface is a significant component for the formation of photosynthesis in plants and contributes to the formation of vitamin D in humans. However, high levels of this radiation and particularly prolonged exposure, without proper protection, can lead a risk to human health. generating skin cancer, eye disorders, and others. Therefore, to associate the risks, ranges, or levels of the so-called Ultraviolet Radiation Index are established. The present thesis consists of the design and implementation of an electronic system to measure and forecast the evolution of ultraviolet radiation levels, using the Internet of Things (IoT) technology and Machine Learning (ML). For it, a specialized sensor is used to measure the intensity of UV radiation and deliver an analog signal corresponding to the intensity to a microcontroller with data communications capability. Thus, having a platform for the acquisition and transmission of data over the Internet to an IoT web server in the cloud where the data is stored in certain predetermined periods and analyzed using a predictive modeling algorithm. Finally, it should be noted that this project short term forecast is obtained on the ultraviolet radiation index to alert and take preventive measures and thus avoid or reduce some health problems caused by this radiation

Keywords- *Ultraviolet Radiation Index (UVI), Internet of Things (IoT), WeMos D1 mini ESP8266, Machine Learning (ML), algorithm of predictive modeling.*

Contenido

Dedicatoria.....	ii
Abstract.....	iv
Contenido.....	v
Lista de Figuras	ix
Lista de Tablas	xii
Introducción	1
CAPÍTULO I: PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA.....	3
1.1. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA DE INVESTIGACIÓN	4
1.1.1. Planteamiento y descripción del problema	4
1.1.2. Formulación del problema	4
1.1.3. Formulación de los problemas específicos	5
1.2. DEFINICIÓN DE LOS OBJETIVOS DE INVESTIGACIÓN	5
1.2.1. Objetivo general.....	5
1.2.2. Objetivos específicos	5
1.3. JUSTIFICACIÓN DE LA INVESTIGACIÓN.....	5
1.3.1. Justificación técnica	5
1.3.2. Justificación económica.....	6
1.3.3. Justificación social	6
1.4. ALCANCES Y LIMITACIONES DE LA INVESTIGACIÓN	7
1.4.1. Alcances	7
1.4.2. Limitaciones	7
CAPÍTULO II: FUNDAMENTO TEÓRICO	8
2.1. ANTECEDENTES	9
2.1.1. Internacionales	9

2.1.2. Nacionales	10
2.2. MARCO TEÓRICO	11
2.2.1. Radiación solar	11
2.2.2. Radiación ultravioleta	12
2.2.3. Índice de radiación ultravioleta	13
2.2.4. Influencia de radiación UV sobre la salud humana	14
2.2.5. Internet de las Cosas (IoT)	17
2.2.6. Machine Learning (ML)	20
2.2.7. Algoritmo de Regresión Lineal	20
2.2.8. Sensor y Hardware	21
2.3. MARCO METODOLÓGICO	28
2.3.1. Tipo de investigación	28
2.3.2. Metodología de la investigación	28
2.4. MARCO LEGAL	28
2.4.1. IEEE 802.11 b/n/g	28
2.4.2. Ley N° 30102	29
2.5. ARQUITECTURA DEL SISTEMA	29
CAPÍTULO III: DESARROLLO DEL SISTEMA	31
3.1. DESCRIPCIÓN DE LA SITUACIÓN ACTUAL DE LA MEDICIÓN DE LA RADIACIÓN UV.	32
3.1.1. Situación actual de la medición de la radiación UV en el Perú. 32	
3.1.2. Instrumentos de medición de radiación UV.	33
3.2. DIAGRAMA DE LAS ETAPAS DEL SISTEMA DE MEDICIÓN Y PRONOSTICACIÓN DE RADIACIÓN ULTRAVIOLETA.	36
3.2.1. Etapas del sistema	36
3.2.2. Características del sistema	37
3.2.3. Diseño del sistema	37

3.2.4.	Diagrama de flujo	41
3.2.5.	Programación en Arduino IDE	41
3.3.	IDENTIFICACIÓN DE UNA PLATAFORMA IoT CON SOPORTE DE APRENDIZAJE AUTOMÁTICO.	46
3.3.1.	Plataformas IoT	46
3.3.2.	Protocolo de comunicación HTTP.	49
3.3.3.	Configuración de la plataforma IoT ThingSpeak.	49
3.3.4.	Configuración del código de Arduino IDE para ThingSpeak ..	53
3.3.5.	Implementación del sistema	54
3.3.6.	Visualización de datos	57
3.3.7.	ThingSpeak – MATLAB	57
3.4.	DESARROLLO DEL ALGORITMO PREDICTIVO PARA EL SISTEMA DE PRONÓSTICO DE RADIACIÓN ULTRAVIOLETA.	59
3.4.1.	Algoritmos de Machine Learning	59
3.4.2.	Regresión Lineal.....	60
3.4.3.	Desarrollo del Algoritmo de regresión lineal en MATLAB.....	62
3.4.4.	Implementación de análisis en la nube	66
3.5.	VERIFICACIÓN DE LOS RESULTADOS DEL SISTEMA DE MEDICIÓN Y PRONÓSTICO DE RADIACIÓN ULTRAVIOLETA	68
3.5.1.	Pasos para usar el dispositivo	68
3.5.2.	Análisis de envío de paquetes de datos a ThingSpeak	69
3.5.3.	Análisis de funcionamiento de medición UV	71
3.5.4.	Análisis del funcionamiento del Pronostico índice UV	75
CAPÍTULO IV: ANÁLISIS DE COSTO Y BENEFICIOS		91
4.1.	ANÁLISIS DE COSTOS	92
4.1.1.	Recursos Humanos	92
4.1.2.	Recursos Materiales.....	92
4.1.3.	Recursos de Software.....	93

4.1.4. Gastos Extra.....	93
4.2. ANÁLISIS DE BENEFICIOS	94
4.2.1. Beneficios tangibles.....	94
4.2.2. Beneficios intangibles	94
4.3. ANÁLISIS DE SENSIBILIDAD	95
4.3.1. Desarrollo del flujo de caja	95
4.3.2. Análisis de VAN	96
4.3.3. Análisis de TIR	96
Conclusiones.....	98
Recomendaciones	99
Glosario.....	100
Referencias	102
Anexos.....	107
Anexo A: Hoja técnica de componentes y sensores.....	108
Anexo B: Comparación de medición UV entre el sensor ML8511 y el radiómetro UV-BIOMETER.....	112
Anexo C: Programación en Arduino IDE	113
Anexo D: Programación en MATLAB a ThingSpeak.....	117
Anexo E: Estadística de casos por enfermedades de radiación UV en Perú	119
Anexo F: Diagrama de Gantt	121
Anexo G: Matriz de consistencia	122
Anexo H: Conference Paper	123

Lista de Figuras

Figura 1. Espectro de la radiación solar electromagnética.	12
Figura 2. Tipos rayos UV que índice sobre la piel.	13
Figura 3. Nivel de riesgo de radiación ultravioleta.	14
Figura 4. Melanoma maligno	15
Figura 5. Carcinoma de células basales.....	15
Figura 6. Carcinoma de células escamosas.	15
Figura 7. Catarata.....	16
Figura 8. Pterigiión.	16
Figura 9. Arquitectura IoT	18
Figura 10. Proceso para la regresión lineal.	20
Figura 11. Gráfica de la Regresión Lineal.	21
Figura 12. Diagrama eléctrico del sensor ML8511.	21
Figura 13. Forma física del sensor UV ML8511.	22
Figura 14. Relación entre el índice UV y el voltaje de salida.....	22
Figura 15. Forma física del WeMos D1 mini ESP8266.	23
Figura 16. Forma física y pines de conexión del display OLED 0.96".....	25
Figura 17. Arduino IDE	26
Figura 18. Diagrama del sistema IoT ThingSpeak.....	27
Figura 19. Diagrama de bloques del sistema de medición.	29
Figura 20. Datos promedios sobre la radiación UV en el año 2015 - 2017.	32
Figura 21. Índice de radiación ultravioleta en Lima 2016.....	33
Figura 22. Radiación UV en Lima Metropolitana, 2017-2018.	33
Figura 23. Luxómetro medidor de luz UV YK-35UV.	34
Figura 24. Medidor digital de radiación para rayos UVA/UVB UV 340B	34
Figura 25. Medidor de radiación solar PCE-SPM 1.	35
Figura 26. Diagrama de bloques del sistema.....	36
Figura 27. Diagrama de conexión del sistema.....	38
Figura 28. Diseño de un modelo base.	40
Figura 29. Modelo del case para el sistema de radiación UV.	40
Figura 30. Diagrama de flujo para el sistema de medición y pronóstico UV.....	41

Figura 31. Esquema de solicitud cliente-servidor.....	49
Figura 32. Registro de usuario en ThingSpeak.	50
Figura 33. Entorno gráfico de ThingSpeak.	50
Figura 34. Configuración del canal.	51
Figura 35. Configuración del campo.	52
Figura 36. API Key del Campo de ThingSpeak.	52
Figura 37. Incorporación del sensor de radiación UV.	55
Figura 38. Incorporación del módulo de carga.....	55
Figura 39. Incorporación del WeMos D1.	56
Figura 40. Implementación del sistema de medición y monitoreo UV.	56
Figura 41. Visualización de datos por ThingSpeak.....	57
Figura 42. MATLAB Visualization programación.	58
Figura 43. MATLAB Visualization gráfica.	58
Figura 44. Gráfica de índice UV con respecto al tiempo.	63
Figura 45. Gráfica del índice UV con respecto al tiempo y la función lineal.	65
Figura 46. Programación en MATLAB Visualization	66
Figura 47. Canal de Pronóstico de índice UV.	67
Figura 48. Esquema de conexión a un punto de acceso	69
Figura 49. Lista de tramas HTTP	69
Figura 50. Trama desde el cliente al servidor.....	70
Figura 51. Trama desde el servidor al cliente.....	71
Figura 52. Ubicación de la toma de muestra 1	72
Figura 53. Fotografía del lugar para la toma de datos.....	72
Figura 54. Gráfica de lectura de IUV en ThingView.....	73
Figura 55. Ubicación de la toma de muestra 2	74
Figura 56. Fotografía del lugar para la toma de datos.....	74
Figura 57. Gráfica de lectura de IUV en ThingView.....	75
Figura 58. Visualización del índice UV a las 12:30 horas.....	76
Figura 59. Visualización del pronóstico IUV siendo las 12:30 horas.	76
Figura 60. Visualización del índice UV a las 12:35 horas.....	77
Figura 61. Visualización del pronóstico IUV siendo las 12:35 horas.	77
Figura 62. Visualización del índice UV a las 12:40 horas.....	78
Figura 63. Visualización del pronóstico IUV siendo las 12:40 horas.	78

Figura 64. Visualización del índice UV a las 12:45 horas.....	79
Figura 65. Visualización del pronóstico IUV siendo las 12:45 horas.	79
Figura 66. Visualización del índice UV a las 12:50 horas.....	80
Figura 67. Visualización del pronóstico IUV siendo las 12:50 horas.	80
Figura 68. Visualización del índice UV a las 12:55 horas.....	81
Figura 69. Visualización del pronóstico IUV siendo las 12:55 horas.	81
Figura 70. Visualización del índice UV a las 13:00 horas.....	82
Figura 71. Visualización del índice UV a las 13:05 horas.....	82
Figura 72. Visualización del pronóstico IUV siendo las 13:05 horas.	83
Figura 73. Visualización del índice UV a las 13:10 horas.....	83
Figura 74. Visualización del pronóstico IUV siendo las 13:10 horas.	84
Figura 75. Visualización del índice UV a las 13:15 horas.....	84
Figura 76. Visualización del pronóstico IUV siendo las 13:15 horas.	85
Figura 77. Visualización del índice UV a las 13:20 horas.....	85
Figura 78. Visualización del pronóstico IUV siendo las 13:20 horas.	86
Figura 79. Visualización del índice UV a las 13:25 horas.....	86
Figura 80. Visualización del pronóstico IUV siendo las 13:25 horas.	87
Figura 81. Visualización del índice UV a las 13:30 horas.....	87
Figura 82. Gráfica comparativa entre la medición y el pronóstico de IUV para las 13 horas.....	88
Figura 83. Gráfica comparativa entre la medición y el pronóstico de IUV para las 13:30 horas.	89
Figura 84. Flujo de caja del primer año.	96

Lista de Tablas

Tabla 1. Protocolos de comunicación en apoyo a IoT.	19
Tabla 2. Especificaciones Técnicas del sensor ML8511	22
Tabla 3. Especificaciones Técnicas del WeMos D1 mini ESP8266.....	24
Tabla 4. Especificaciones Técnicas del display OLED 0.96".....	25
Tabla 5. Consumo eléctrico de los componentes.....	39
Tabla 6. Dimensiones del sistema.....	40
Tabla 7. Plataformas de IoT.....	48
Tabla 8. Resultados obtenidos en la prueba 1.	73
Tabla 9. Resultados obtenidos en la prueba 2.	75
Tabla 10. Resultados para el pronóstico del IUV para las 13:00 horas.....	88
Tabla 11. Resultados para el pronóstico del IUV para las 13:30 horas.....	89
Tabla 12. Resultado y comparación de errores.....	90
Tabla 13. Recursos Humanos.	92
Tabla 14. Recursos Materiales.....	92
Tabla 15. Recursos de Software.....	93
Tabla 16. Gastos Extra.....	93
Tabla 17. Gasto total del proyecto.....	93
Tabla 18. Flujo de caja del proyecto.....	95

Introducción

La radiación ultravioleta constituye un factor importante y beneficioso en algunos seres vivos como las plantas, permitiendo la realización de la fotosíntesis. Asimismo, permite la producción de vitamina D en el ser humano, sobre todo cuando la radiación que incide sobre la piel humana se encuentra en intensidades mínimas, y por tanto no riesgosas para la salud.

Sin embargo, estamos viviendo un cambio climático por efecto del denominado calentamiento global perjudicando la calidad de vida de muchas especies, esto es causado por múltiples factores como exceso de emisión de dióxido de carbono (CO₂) de las fábricas, parque automotor, aerosoles, etc. Esto trae como una de las consecuencias la elevada temperatura terrestre, incremento del nivel del mar, y el daño más importante, el debilitamiento de la capa de ozono hasta el extremo de la formación de agujeros de ozono ubicados en el hemisferio sur terrestre. Cabe indicar que la capa de ozono cumple una función importante en la absorción de la radiación UV, con el debilitamiento de la capa la absorción es menor; por ello la radiación UV incide sobre la superficie terrestre con mayor intensidad pudiendo afectar a la salud de las personas generando inflamación en la piel por quemaduras solares, cáncer de piel como melanoma maligno, enfermedades oculares y otros.

Los rangos o niveles de la intensidad de la radiación UV que incide sobre la superficie terrestre, está caracterizado el Índice de Radiación Ultravioleta (IUV). Existen factores que intensifican el IUV como la latitud y altitud, así habrá mayor intensidad cuando un país se encuentre más cercano a la línea ecuatorial tal como es el caso de Perú. De acuerdo a la OMS a mayor altitud aumenta en 10 - 12 % la intensidad de radiación; esto se debe porque la atmósfera es de menor espesor.

La presente tesis tiene por finalidad el diseño y la implementación de un sistema de electrónico para realizar la medición y el pronóstico de los niveles de la radiación ultravioleta utilizando para ello la tecnología de Internet de las cosas (IoT) y los algoritmos de *Machine Learning (ML)*. La medición de los niveles de radiación UV se realiza capturando los datos procedentes de un sensor de

radiación UV, luego se procesan y los IUV se transmiten a un servicio web de IoT en la nube, donde se almacena los datos en determinados periodos prefijados y se analizan mediante un algoritmo de modelado predictivo; permitiendo de este modo que el sistema brinde un pronóstico a corto plazo del IUV. El sistema bajo estudio permitirá alertar y tomar medidas de prevención evitando algunos problemas de salud y más aún en niños y jóvenes que en un futuro es probable que presenten las consecuentes enfermedades.

En el capítulo I se realiza el planteamiento y la formulación del problema, describiendo la problemática en si sobre las fuentes y datos estadísticos recopilados para la realización de la investigación.

En el capítulo II se describe los antecedentes tanto nacionales como internacionales, los conceptos, teorías y la metodología necesaria para el cumplimiento de la realización de la tesis.

En el capítulo III se desarrolla la tesis cumpliendo con los objetivos planteados. Presentando una descripción de la situación actual, equipos de medición UV, estadísticas sobre los índices UV, diseño del modelo 3d para la estructura, diseño e implementación en la parte electrónica, desarrollo de la programación Arduino y del algoritmo ML y verificación de resultados mediante las pruebas correspondientes.

En el capítulo IV, se realiza el análisis de costos y los beneficios correspondiente a los aspectos administrativos para el desarrollo de la tesis como recursos humanos, materiales, software y gasto; además, se explica sobre la sensibilidad con respecto a la elaboración de la tesis.

Finalmente, se describirá las conclusiones, recomendaciones y los anexos correspondientes en el desarrollo de la presente investigación.

CAPÍTULO I: PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA

1.1. PLANTEAMIENTO Y FORMULACIÓN DEL PROBLEMA DE INVESTIGACIÓN

1.1.1. Planteamiento y descripción del problema

En Perú, según el Servicio Nacional de Meteorología e Hidrología (SENAMHI), anunció que el año 2016 el IUV alcanzó un valor de 12 siendo considerado este número en el margen de nivel de riesgo muy alto para la salud (Lima Como Vamos, 2017). En el año 2019, el Ministerio de Salud (MINSA) ha reportado 8549 casos de pacientes que padecen enfermedades en la piel debido a las elevadas exposiciones prolongadas a la radiación ultravioleta, ver Anexo E (MINSA, 2020).

Principalmente, los IUV son más altos en la estación de verano, temporada en que las familias y sus niños, ambulantes que se encuentran en las vías públicas, obreros hasta los policías que dirigen el tránsito vehicular y otros casos; que están expuestos constantemente a la radiación UV, siendo esto un riesgo muy importante que en ocasiones extremas conlleva a diversos tipos de cáncer de la piel, enfermedades a la vista y otras enfermedades.

Actualmente, existen equipos de medición de radiación ultravioleta con precios altos; sin embargo, este equipo solamente muestra el indicador o valor numérico mas no una interpretación que visualizaría un posible riesgo. (Valiometro, 2020)

Es por ese motivo, que se desea diseñar e implementar un sistema de medición y pronóstico de radiación ultravioleta utilizando Internet de las cosas (IoT) y *Machine Learning* (ML)

Este proyecto tendrá la facilidad de ser útil para los usuarios en general brindando un pronóstico a corto plazo sobre el índice de radiación ultravioleta, para alertar y tomar medidas de prevención evitando algunos problemas de salud y más aún en niños y jóvenes que en un futuro es probable que presenten alguna de estas enfermedades.

1.1.2. Formulación del problema

¿Cómo medir y pronosticar los niveles de radiación ultravioleta mediante Internet de las cosas (IoT) y *Machine Learning* (ML)?

1.1.3. Formulación de los problemas específicos

P.E.1: ¿De qué manera se realiza la medición de los índices de radiación ultravioleta en la actualidad?

P.E.2: ¿Cuáles son las etapas para el diseño e implementación del sistema?

P.E.3: ¿Qué tipo de plataforma permite almacenar y procesar los datos en la nube para el diseño e implementación del sistema?

P.E.4: ¿Qué tipo de análisis se necesita en el sistema de pronóstico de radiación ultravioleta?

P.E.5: ¿De qué manera se validará el sistema de medición y pronóstico de radiación ultravioleta?

1.2. DEFINICIÓN DE LOS OBJETIVOS DE INVESTIGACIÓN

1.2.1. Objetivo general

Diseñar e implementar de un sistema de medición y pronóstico de radiación ultravioleta utilizando Internet de las Cosas y *Machine Learning*.

1.2.2. Objetivos específicos

O.E.1: Describir la situación actual de la medición de la radiación UV.

O.E.2: Diseñar el diagrama de las etapas del sistema de medición y pronóstico de radiación ultravioleta.

O.E.3: Identificar la plataforma de Internet de las cosas con soporte de *Machine Learning*.

O.E.4: Desarrollar el algoritmo predictivo para el sistema de pronóstico de radiación ultravioleta.

O.E.5: Verificar los resultados del sistema de medición y pronóstico de radiación ultravioleta.

1.3. JUSTIFICACIÓN DE LA INVESTIGACIÓN

1.3.1. Justificación técnica

Actualmente, los estudios de los índices de la radiación ultravioleta UV es realizado, mínimamente por entidades públicas que monitorean los índices de estudio de este factor, cabe mencionar que dichos sistemas o equipos de medición solamente detectan y muestran el valor del índice de radiación ultravioleta (ServoVendi, 2020). Por lo que se necesita un sistema que permita captar y

analizar la intensidad de radiación ultravioleta, siendo este proyecto que involucra la utilización de tecnologías como IoT utilizando un hardware abierto como WeMos D1 mini ESP8266 para la transmisión de datos por una red wifi, un sensor para captar los índices de radiación ultravioleta, el software IDE de Arduino para la programación y procesamiento de los datos. Los datos de los IUV son recibidos en una plataforma de servicio de IoT denominada ThingSpeak, donde este permite la compatibilidad con el software Matlab para analizar y desarrollar el algoritmo de regresión lineal de ML para predecir el IUV en los próximos minutos u horas.

1.3.2. Justificación económica

Existen aparatos que permiten medir la radiación ultravioleta pero simplemente muestran el valor del IUV actual. Además, estos equipos disponen de un precio muy elevado para una simple función de mostrar el IUV. (Valiometro, 2020) La realización de este proyecto es medir la radiación ultravioleta y mostrar un pronóstico para prevenir daños siendo de beneficio para la salud de todas las personas evitando las posibles enfermedades que perjudicarían.

1.3.3. Justificación social

Muchas personas se encuentran vulnerables a los accidentes o diferentes tipos de enfermedades que podrían hasta causar la muerte. Principalmente, en épocas de verano las personas están expuestas directamente a la radiación ultravioleta sin darse cuenta que esto podría generarle un daño irreversible y más aún que los índices de radiación se van incrementando con el paso de los años. En muchas situaciones, las personas no toman la importancia de su salud, salen a la calle o pasan mucho tiempo en lugares públicos expuestos directamente al sol sin usar alguna protección contra la radiación UV. Este proyecto se enfocará en ayudar a prevenir informando sobre los altos IUV y los riesgos se encuentran expuestos que pueden afectar a la salud, además muestra la tendencia de exposición a corto tiempo con actualizaciones instantáneas aplicando algoritmos de ML para la realización del pronóstico.

1.4. ALCANCES Y LIMITACIONES DE LA INVESTIGACIÓN

1.4.1. Alcances

- Este sistema estará diseñado para medir los IUV y transmitirlos utilizando IoT a un servicio web de ThingSpeak que además permite realizar un pronóstico mediante algoritmos de ML desarrollado en la nube con la finalidad de visualizar la tendencia de los IUV con respecto al tiempo con actualización en tiempo real.
- La presente tesis sienta la base para posteriores investigaciones que permita agregar otros dispositivos o módulos electrónicos al sistema; además permita resolver problemas de salud o de otra índole.

1.4.2. Limitaciones

- Escasa oferta formativa especializada para conocer a más detalle la tecnología sobre IoT y la relación con ML. Por lo que es necesario asistir a diversas conferencias, charlas, talleres y cursos para comprender todo lo relacionado a esta tecnología.
- Poca disponibilidad en el mercado laboral de dispositivos electrónicos de alto rendimiento, especialmente el sensor de radiación para la obtener resultados precisos y alta fiabilidad.

CAPÍTULO II: FUNDAMENTO TEÓRICO

2.1. ANTECEDENTES

2.1.1. Internacionales

A. Pérez (2014) en su tesis titulada “**Diseño de un radiómetro ultravioleta, para su aplicación en modelos de radiación UV**” de la Universidad Nacional Autónoma de México, menciona que: “Una etapa de amplificación es primordial para lograr el funcionamiento del Radiómetro UV. Así también, es necesario transformar en primera instancia la fotocorriente del sensor UV a voltaje para poder realizar una posterior caracterización”.

La utilización de amplificadores operacionales como parte fundamental para transformar la señal analógica según menciona el autor es recomendable especialmente de amplificadores con mejores características (menores niveles de ruido, baja tensión, alta velocidad, etc.) para obtener mejores resultados de muestreo en conversión análogo- digital.

B. Iribarren (2016) en su tesis titulada “**Modelo predictivo *Machine Learning* aplicado al análisis de datos climáticos capturados por una placa Sparkfun**” de la Universidad Pontificia Comillas, menciona que los modelos predictivos son de gran complejidad debido a la cantidad constante de corrección de errores, el tiempo de realización y la creación durante el proceso.

Es conveniente prestar especial atención al comentario de esta tesis para estar al tanto del desarrollo algorítmico de *Machine Learning* y reducir los posibles errores al realizar las pruebas en su desarrollo.

C. Perea (2016) en su tesis titulada “**Sistema de monitoreo, registro y control de temperatura y humedad para el cultivo de invernadero**” de la Universidad tecnológica de Pereira, durante su desarrollo del proyecto “Estableció conectar un microcontrolador de bajo costo (Arduino Mega) con un servidor web para el almacenamiento de medidas en base de datos en MySQL. Además, recomienda reemplazar la conexión internet por un sistema inalámbrico”.

En esta tesis es interesante al transmitir los datos de información en una red de internet y almacenarla en una base de datos como MySQL esta técnica de base de datos se podría considerar en el caso del índice UV y además seguir la recomendación del uso de sistema inalámbrico como sería a través de una red Wifi.

- D. Arias (2016) en su investigación titulada “**Demostrador IoT-Cloud en tiempo real**” de la Universitat Oberta de Catalunya, concluye que durante el desarrollo de este proyecto ha realizado un análisis de situación actual de IoT con respecto a los servicios en la nube sin embargo esta interacción tiene un alto coste y de ahí que se ha utilizado un sistema de computación en la niebla o *fog computing*.

Cabe señalar que el servicio en la nube se ha vuelto muy popular en los últimos años casi todos los dispositivos se conectan a internet, los datos son procesados y almacenados hacia un ordenador o usuario para luego aplicarlos en un actuador u otras tareas.

2.1.2. Nacionales

- A. Rodríguez (2015) en su tesis titulada “**Influencia de la radiación solar en la salud de las personas en la ciudad de Moquegua, 2001-2010**”, concluye que la radiación solar ultravioleta que incidió en la ciudad de Moquegua en el periodo estudiado, registró un índice en un nivel de riesgo entre moderado y alto, según estándares dados a conocer a través del portal de servicio nacional de meteorología e hidrología del Perú-SENAMHI. El estudio en esta tesis indica que niveles de riesgo afectan totalmente a nuestra salud que pueden causar diversas enfermedades de piel y de la vista. Por ello, se debe concientizar y prevenir de estos riesgos a las autoridades gubernamentales y la población a fin de reducir los posibles daños a la salud.
- B. Paredes (2017) en su tesis titulada “**Diseño de un sistema de señalización utilizando sensores fotovoltaicos para la prevención de los efectos de la radiación solar para los estudiantes de la EPIE**” de la Universidad Nacional del

Altiplano, recomienda que: “Se debe implementar proyectos de señalización de radiación ultravioleta de tal manera que se pueda evitar lesiones y enfermedades producidas por los altos niveles de radiación UV”.

Es muy importante tener en cuenta esta recomendación debido a que muchas personas no saben que significa cada número del índice UV, pero si se muestra en un sistema de señalización por medio de rango de colores indicaría cual es el daño al estar expuesto a determinado índice UV.

- C. En Medina, Mujaico y Soto (2018), indica que las personas cuando está expuesta en lugares con altas concentración de gases contaminantes y sumandos con altos niveles de índice UV; el daño en la salud es 3 veces más que el daño por separado. Además, en este trabajo tiene el objetivo de implementar un sistema para evaluar los factores ambientales en diferentes lugares midiendo la radiación ultravioleta y los principales gases contaminantes.

En este trabajo se realizó una recolección de datos en diferentes distritos para determinar los principales lugares con altos niveles de contaminantes e índice UV.

2.2. MARCO TEÓRICO

2.2.1. Radiación solar

La radiación solar comprende en un segmento de longitudes de onda abarcan desde los rayos gamma hasta las ondas de radio, comprendiendo por rayos X, ultravioleta (UV), visible, infrarrojo (IR) y microondas este que es aplicado en las telecomunicaciones. La radiación conlleva una gran variedad de energía y partículas como los fotones que van directamente proporcional e inversamente proporcional a la longitud de onda con una constante a la velocidad de la luz (De La Morena, Sorribas, Adame, Vilaplana, Gil, Balairón... Lopez, 2010).

En la Figura 1, se visualiza la parte comprendida con longitudes de onda inferiores a 400 nm que son ultravioleta (UV) y que se subdivide en otras regiones como UV-A, UV-B y UV-C. La región del visible

está situada en el rango 400 nm -740 nm se denomina así porque es la región del espectro electromagnético sensible al ojo humano. La parte del infrarrojo del espectro solar son mayores a 780 nm de longitud de onda.

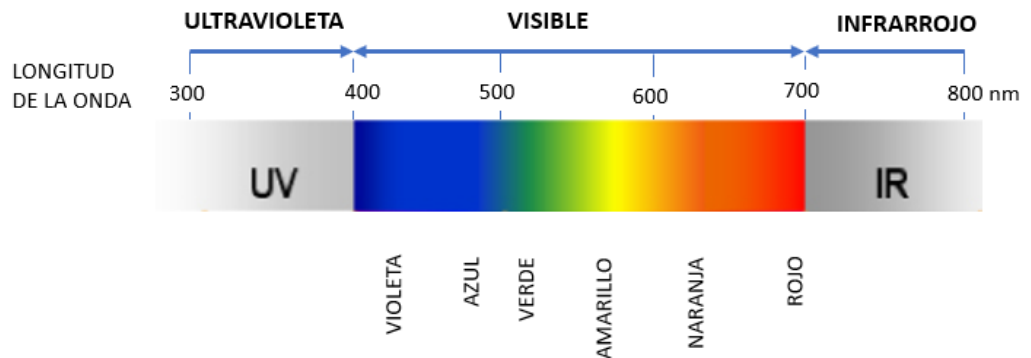


Figura 1. Espectro de la radiación solar electromagnética.

2.2.2. Radiación ultravioleta

La radiación ultravioleta (UV) es una parte de la radiación solar comprendida entre con longitudes de onda entre 100 - 400nm. La radiación ultravioleta es beneficiosa para la salud y desempeñan una función esencial en la producción de vitamina D siempre y cuando sea la radiación en menores cantidades. Sin embargo, una larga exposición provoca diferentes tipos de cáncer cutáneo, quemaduras de sol, envejecimiento acelerado de la piel, cataratas y otras enfermedades oculares (Organismo Mundial de la Salud, 2003).

Existen tres tipos de rayos UV que dependen en intervalos de longitudes de onda. Los rayos UV-A abarcan longitudes de onda entre 315nm a 400 nm y normalmente llega a la superficie terrestre y que pueden hasta atravesar las ventanas y hasta en prendas. Estos rayos penetran la piel livianamente hasta las células de la piel alterando la estructura del ADN que causando enrojecimiento, manchas, resequedad y arrugas a temprana edad (Aguayo, 2017). Los rayos UV-B que abarcan longitudes de onda entre 280nm a 315 algunos de ellos son absorbidos por la capa de ozono y hasta también en las nubes, pero de todas maneras pueden llegar a la superficie terrestre. Estos rayos al tener menor longitud de onda que

el UV-A, poseen mayor energía que son realmente dañinos. Por ello, pueden provocar el aumento de grosor cutáneo y melanoma (Aguayo, 2017).

Los rayos UV-C que abarcan longitudes de onda entre 100nm a 280nm estos rayos son mayormente absorbidos por la capa de ozono mucho antes de llegar a la tierra, debido a que la atmosfera terrestre lo bloquea (Aguayo, 2017).

De la Figura 2, se visualiza la radiación UV-A como el UV-B afecta la naturaleza de la piel dañando toda la zona de la epidermis, dermis y puede llegar hasta la hipodermis afectando las células de la piel alterando la estructura del ADN y pudiendo causar cáncer de piel.

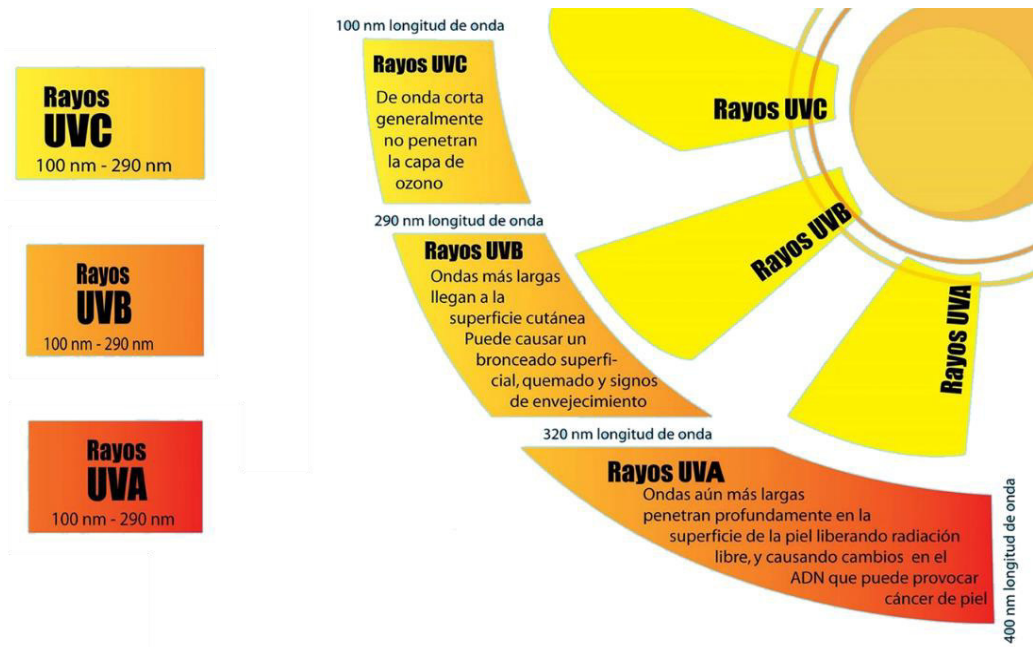


Figura 2. Tipos rayos UV que índice sobre la piel.

2.2.3. Índice de radiación ultravioleta

El IUV es un indicador de medición de la intensidad UV que índice en la plataforma terrestre y que influye en los efectos sobre la piel humana (Organización Mundial de la Salud, 2003).

El IUV es un coeficiente que se adquiere mediante el producto de la irradiancia eritemática por una constante que equivale $k_{er} = 40 \text{ m}^2/\text{W}$, (Servicio Nacional de Meteorología e Hidrología, 2016), como se expresa en la siguiente ecuación:

$$IUV = k_{er} * \int_{280}^{400} I(\lambda) * \varepsilon(\lambda) d\lambda \quad (1)$$

En la Figura 3, se muestra los niveles de radiación UV en Perú son considerados por el nivel bajo hasta un índice UV de 5 en condiciones normales, no se necesitan medidas de protección. Sin embargo, a partir del índice UV 6 es necesario protegerse tomando las medidas preventivas para así reducir el daño.







Índice UV-B	Nivel de riesgo
 1-2	Mínimo
 3-5	Bajo
 6-8	Moderado
 9-11	Alto
 12-14	Muy alto
 >14	Extremo

Figura 3. Nivel de riesgo de radiación ultravioleta.

Fuente: Lima como vamos.

2.2.4. Influencia de radiación UV sobre la salud humana

La piel cubre gran área del cuerpo humano por eso es considerado el órgano más grande. La función principal de la piel es proteger contra sustancias químicas, radiación, infecciones y agentes externos. La piel contiene un pigmento químico llamado melanina que sirve como escudo contra la radiación UV (MedlinePlus, 2019). La radiación UV en reducidas proporciones desempeñan una ocupación esencial en la salud humana para la productividad de calciferol o vitamina D. Un exceso de esta radiación puede provocar riesgos en la salud ocular y cutánea, que son los más comunes (Organización Mundial de la Salud, 2003).

A. Enfermedades Cutáneas

Existen varios casos de enfermedades de piel, como: quemaduras, envejecimiento temprano y cáncer.

- a. **Melanoma maligno:** Es la principal causa de muerte por cáncer de piel (Figura 4).

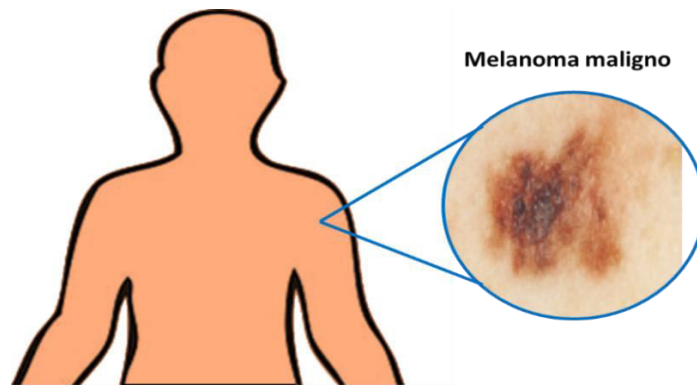


Figura 4. Melanoma maligno

- b. **Carcinoma de células basales:** Cáncer de piel con un crecimiento desacelerado siendo afectado en su mayoría en adultos mayores (Figura 5).

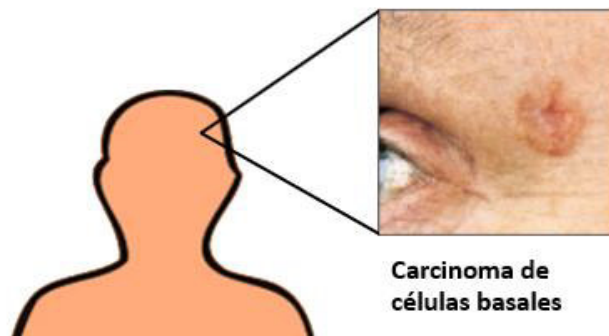


Figura 5. Carcinoma de células basales.

- c. **Carcinoma de células escamosas:** Cáncer de piel maligno muy parecido al melanoma siendo la segunda causa de muerte (Figura 6).



Figura 6. Carcinoma de células escamosas.

Fuente: Organización Mundial de la Salud.

B. Enfermedades Oculares

La exposición excesiva a la radiación UV provocan diversos problemas visuales desde visión borrosa, irritación y enrojecimiento de ojos, pérdida temporal de la visión y, en caso extremo, la ceguera y se calcula que tres millones de personas por año se quedan ciegas. (Oftalvist, 2020)

Estos efectos pueden prevenirse usando gafas protectoras anti UV y realizándose chequeo médico tempranamente.

- a. **Catarata:** El cristalino (parte de ojo) comienza lentamente opacarse parcial o totalmente, reduciendo la visión progresivamente (Figura 7).

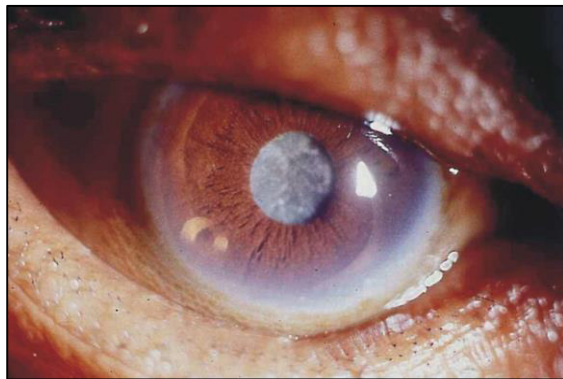


Figura 7. Catarata.

Fuente: Oftalvist.

- b. **Pterigión:** Patología ocular que se presenta como una carnosidad generando molestia, ardor, sequedad y enrojecimiento (Figura 8).

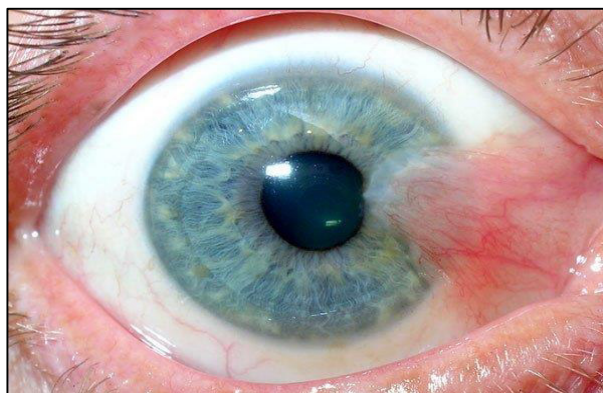


Figura 8. Pterigión.

Fuente: Oftalvist.

2.2.5. Internet de las Cosas (IoT)

Internet de las cosas, IoT por su abreviación en inglés (*Internet of Things*), es puntualmente, que cualquier objeto pueda tener conectividad a internet ya sea por una red inalámbrica o alámbrica, en cualquier lugar y momento, a un bajo costo y bajo consumo energético, permitiendo conectar el mundo físico y el mundo digital (Tecsup, 2018).

Según investigadores de Cisco, estima que en cada mes más de 3 millones de dispositivos nuevos se conectan a Internet. Además, dentro de 4 años habrá más de 30 mil millones de dispositivos en conectividad en todo el mundo. La tercera parte de los dispositivos serán computadoras, smartphones, Tablet, Smart Tv y los dos tercios restantes serán ocupados por “cosas” como los sensores, actuadores y dispositivos recién fabricados (Cisco Networking Academy Build IT & Education, 2019).

Existen varios módulos para realizar proyectos con internet de las cosas, pero depende de la complejidad a cuál se desea realizar. El Raspberry Pi es uno de ellos y se puede decir que es más completo para proyectos altamente complejos. Existen hardware que permite la accesibilidad de trabajar con IoT, como en NodeMCU, Arduino MKR1000, WeMos, módulos Wi-Fi, Shield Ethernet, etc. y que se programan con el software IDE de Arduino.

A. Arquitectura

Existen algunos proyectos aplicados con IoT que tratan de diseñar una arquitectura según lo requerido para la necesidad del investigador (Al-Fuqaha, Guizani, Mohammadi, Aledhari y Ayyash, 2015).

Existen 4 modelos de arquitectura propuestos, el más sencillo y simplificado consta de 3 capas (percepción, red y aplicación) como se muestra en la Figura 9 (a) y otras arquitecturas comunes como el modelo de 5 capas. Estos modelos de capas son diferentes con las TCP / IP en la parte de redes (Al-Fuqaha, Guizani, Mohammadi, Aledhari y Ayyash, 2015)

En la Figura 9, en (a) las tres capas, en (b) basado en middleware, (c) basado en SOA y en (d) las 5 capas.

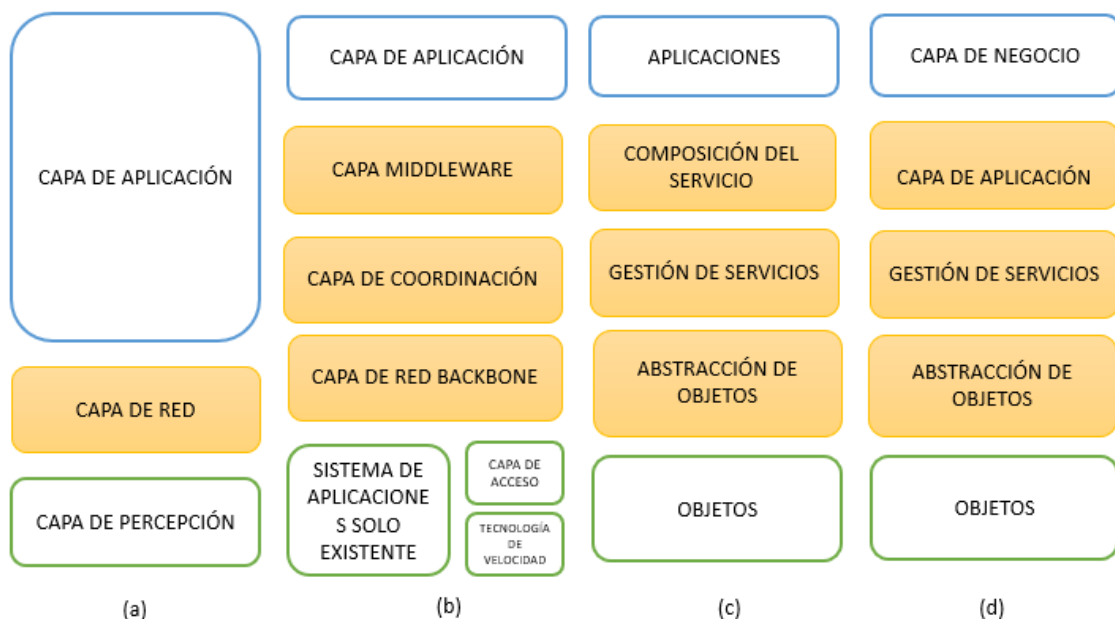


Figura 9. Arquitectura IoT

B. Protocolos IoT

La comunicación entre los sensores, hardware aplicados al IoT se requiere que los protocolos sean ligeros, teniendo en cuenta el bajo consumo energético, el consumo de ancho de banda.

Existen muchos estándares de IoT que permite dar la facilidad en el desarrollo de aplicaciones e integridad en los servicios de los proveedores. Por lo que se ha organizado grupos que facilite los protocolos como soporte al IoT como son *World Wide Web (W3C)*, *Internet Engineering Task Force (IETF)*, *Institute of Electrical and Electronics Engineers (IEEE)*, *European Telecommunications Standards Institute (ETSI)*, entre otros. En la siguiente Tabla, se muestra un resumen de los protocolos más destacados utilizado en IoT (Al-Fuqaha, Guizani, Mohammadi, Aledhari y Ayyash, 2015).

En la Tabla 1, muestra los protocolos más usado con sus respectivas diferencias aplicados en IoT. Teniendo en cuenta, que aun la arquitectura de IoT no está totalmente unificada y no hay definición concreta en los estándares y protocolos para el IoT (Al-Fuqaha, Guizani, Mohammadi, Aledhari y Ayyash, 2015).

Tabla 1. Protocolos de comunicación en apoyo a IoT.

PROTOCOLO DE APLICACIÓN		DDS	CoAP	AMQP	MQTT	MQTT_SN	XMPP	HTTP REST
DESCUBRIMIENTO DE SERVICIOS		mDNS				DNS-SD		
PROTOCOLOS DE INFRAESTRUCTURA	Protocolo de enrutamiento	RPL						
	Capa de red	6LoWPAN			IPv4/IPv6			
	Capa de enlace	IEEE 802.15.4						
	Capa física	LTE-A	EPC global	IEEE 802.15.4	Z-Wave			

Fuente: Al-Fuqaha, Guizani, Mohammadi, Aledhari y Ayyash

C. Componentes de una aplicación IoT

Es importante comprender los componentes que se requiere para una aplicación en IoT, entendiendo desde un punto de vista más simplificado y conformado por tres elementos (Tecsup, 2018):

- **Hardware:** Abarca a los sensores o dispositivos que se caracteriza en capturar la información del entorno físico a medir de acuerdo a su aplicación (sensor de temperatura, humedad, distancia, entre otros).

En este proyecto se requiere el uso de un sensor de radiación UV.

- **Telecomunicaciones:** Es el elemento responsable de transmitir la información y originar los datos, utilizando una red de comunicación como el módulo Wifi, Bluetooth, 3G, 4G, etc.
- **Software en la nube:** En este elemento permite recolectar, almacenar, organizar e integrar la información; de modo, que se puede realizar el análisis de los datos para originar conocimientos, detectando patrones a través diversos mecanismos de la Inteligencia Artificial como: *Machine Learning*, etc.

2.2.6. Machine Learning (ML)

El Aprendizaje de Máquina o Aprendizaje automático. Es una rama de la inteligencia artificial que se encarga del análisis de los sistemas del estudio de las herramientas informáticas utilizando las experiencias pasadas que puedan aprender datos con la finalidad de automatizar. El aspecto iterativo del ML es importante porque a medida que los modelos son expuestos a nuevos datos, éstos pueden adaptarse de forma independiente. Aprenden de cálculos previos para producir decisiones y resultados confiables y repetibles (Software y Soluciones de Analítica, 2019).

2.2.7. Algoritmo de Regresión Lineal

Es uno de los métodos de amplio uso para predecir los valores correctos correspondientes a cada dato de entrada. De otro modo, se busca una relación determinada entre las variables de entrada y salida, representada por una ecuación lineal. Este enfoque es similar al dominio físico y se puede usar para predecir la temperatura o la humedad o en este caso la radiación ultravioleta, conociendo de antemano una serie de datos registrados o históricos (Bonnin, 2016). Así como se muestra en la Figura 10 el proceso para la regresión lineal, donde el valor de “x” representaría los datos de entrada, “h” representa la función de hipótesis y el valor de “y” representaría un valor determinado para la salida dependiendo del valor de entrada.



Figura 10. Proceso para la regresión lineal.

De acuerdo con la tesis, el valor de “x” representaría el tiempo o la hora y el valor de “y” representa el índice de radiación ultravioleta (IUV) que depende de la hora.

Otra forma de definir a la regresión lineal es que genera partir de pares de valores numéricos llamadas ejemplo de entrenamiento una función de hipótesis $h(x)$, mostrado en la Figura 11.

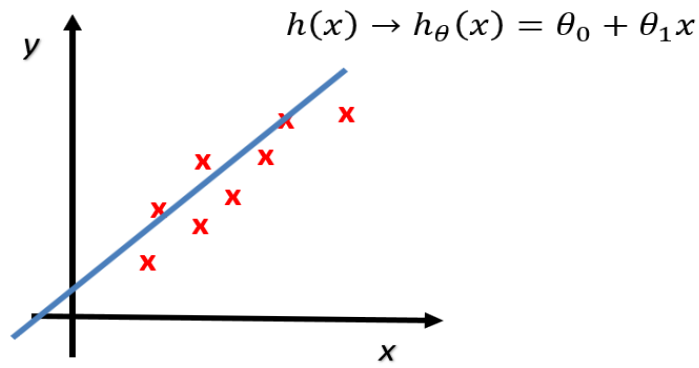


Figura 11. Gráfica de la Regresión Lineal.

2.2.8. Sensor y Hardware

A. Sensor de radiación ultravioleta

El ML8511 es un sensor UV que es adecuado para adquirir intensidad UV. Este sensor consta internamente de un amplificador operacional (OPAM) como se observa en la Figura 12, que convierte la corriente fotoeléctrica a voltaje según la intensidad UV siendo esto muy eficaz para la interfaz con circuitos. En el modo de *power off*, la corriente de *standby* es 0.1 μA , dando una larga duración de la batería (Lapis Semiconductor, 2013).

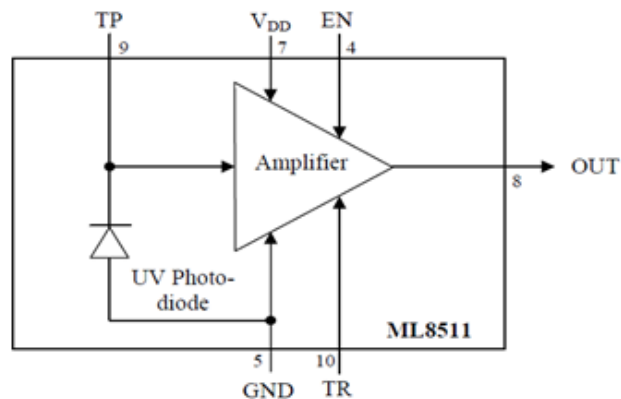


Figura 12. Diagrama eléctrico del sensor ML8511.

Fuente: Lapis Semiconductor.

El sensor ML8511 tiene la capacidad de detectar la luz en un margen del espectro UV-A y UV-B comprendido con un longitud de onda entre 280nm-390nm. Su tamaño físico reducible,

permite la facilidad de utilizar en diferentes proyectos ambientales (Figura 13).

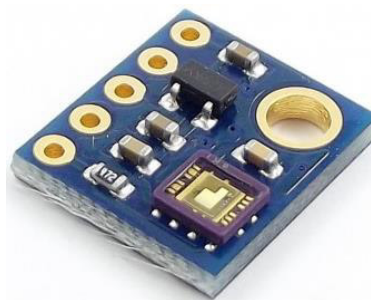


Figura 13. Forma física del sensor UV ML8511.

Fuente: Naylamp Mechatronics

En la Tabla 2, se explica a detalle las especificaciones técnicas del sensor UV ML8511.

Tabla 2. Especificaciones Técnicas del sensor ML8511

ESPECIFICACIONES TÉCNICAS	DESCRIPCIÓN
Voltaje de operación	3.3 – 5.0 VDC
Salida analógica	A0
Fotodiodo sensible	UV-A y UV-B
Longitud de onda	280-390nm
Amplificador operacional	Incorporado
Baja corriente de alimentación	300 μ A
Baja corriente <i>standby</i>	0.1 μ A
Dimensiones	4.0 *3.7*0.73 mm

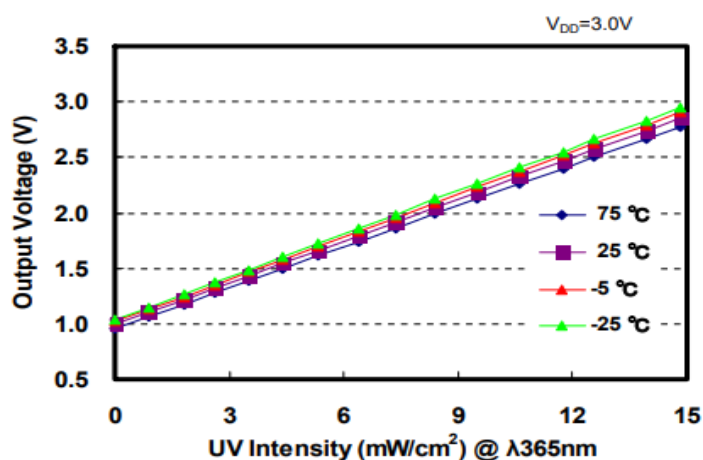


Figura 14. Relación entre el índice UV y el voltaje de salida.

Fuente: Lapis Semiconductor.

En la Figura 14, se muestra la relación del índice de radiación UV en unidades de mili watts por cada centímetro cuadrado (mW/cm^2) con respecto a la salida del voltaje analógico. Mientras mayor es el índice UV que captura el sensor mayor será el voltaje y lo mismo sería en el proceso inverso. Este sensor ML8511 tiene una alta precisión en la medición de índice UV y lo puede revisar en el Anexo B.

B. WeMos D1 mini ESP8266

Es una plataforma de desarrollo electrónico orientado para aplicaciones con el Internet de las Cosas. Esta placa es una versión alterna al Node MCU ESP8266, más simplificada por la cantidad de pines GPIO e integra un microcontrolador con arquitectura de 32 bits, siendo una capacidad superior a un Arduino, e incorpora un módulo Wi-Fi para conectividad inalámbrica (Naylamp Mechatronics, 2020).

En la Figura 15, se muestra el WeMos D1 mini ESP8266 en su forma física de tamaño pequeño flexible y es más reducible semejante al Arduino Pro mini.

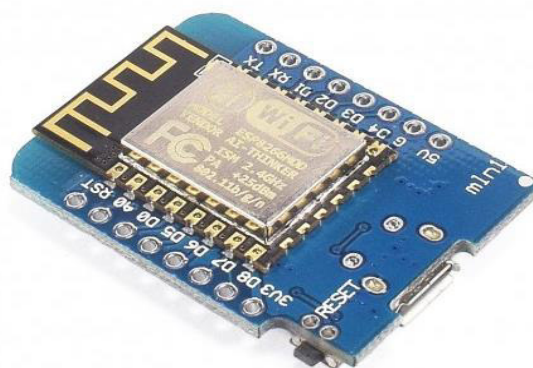


Figura 15. Forma física del WeMos D1 mini ESP8266.

Fuente: Naylamp Mechatronics

WeMos D1 dispone de un conector micro USB incorporado. Posee únicamente un pin analógico, suficiente para trabajar con el sensor de radiación UV y además en el ahorro de espacio por sus dimensiones de la plataforma (Naylamp Mechatronics, 2020).

En la Tabla 3, se explica a detalle las especificaciones técnicas de la plataforma WeMos D1 mini ESP8266.

Tabla 3. Especificaciones Técnicas del WeMos D1 mini ESP8266.

ESPECIFICACIONES TÉCNICAS	DESCRIPCIÓN
Voltaje de Alimentación	5V DC
Voltaje de salida	3.3V DC
System on a Chip	Módulo ESP-12E
Normativa	Wi-Fi Direct (P2P)
Consumo de corriente	10uA ~ 170mA.
Memoria Flash conectable	16MB máximo
Pila de protocolos	TCP / IP
CPU	Tensilica Xtensa LX3 (32 bits).
Frecuencia de Reloj	80 MHz ~ 160MHz.
Pines Digitales GPIO	11
Pin Analógico ADC:	1 de 10 bits
Antena	PCB
Potencia de salida	+ 19.5dBm en modo 802.11b
Estándar	802.11: b/g/n.
Máximo de conexiones TCP:	5
Dimensiones	35mm*21mm*4mm

C. Display Oled 0.96”

Una pantalla o display OLED (*Organic Light-emitting diode*) es una variante de diodo LED basado en una capa de electroluminiscente formado por un compuesto orgánico que emiten luz al ser aplicado un determinado voltaje (Llamas, 2016). En la Figura 16, se muestra la forma física del display OLED 0.96”, dispone de 4 pines de los cuales 2 son de alimentación GND y 5v. Los pines SDA (señal de datos) y SCL (la señal de reloj) corresponde a un protocolo síncrono de comunicación I2C (interfaz de 2 cables), por lo que facilita la conexión con el microcontrolador.

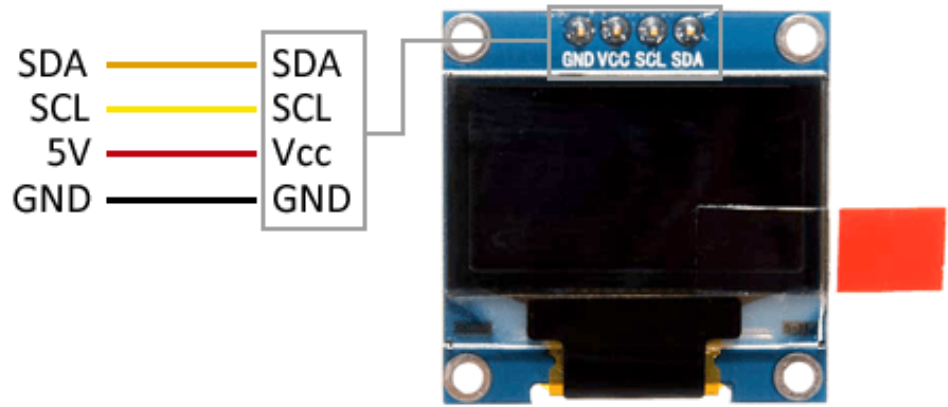


Figura 16. Forma física y pines de conexión del display OLED 0.96".

Fuente: Llamas.

La ventaja de usar estas pantallas, es por el bajo consumo energético, alrededor de 20 mA, debido a que los pixeles se encienden cuando se requiere; y no necesita del *backlight* como las pantallas LCD (Llamas, 2016).

En la Tabla 4, se explica a detalle las especificaciones técnicas sobre el display OLED 0.96".

Tabla 4. Especificaciones Técnicas del display OLED 0.96".

ESPECIFICACIONES TÉCNICAS	DESCRIPCIÓN
Voltaje de Operación	3.3 – 5.0 VDC
Driver	SSD1306
Interfaz	I ² C
Resolución	128*64 píxeles
Color Pixeles	Amarillo y Cian
Angulo de visión	160°
Dimensiones	27*27*4.1 mm

2.2.9. Software y Plataforma

A. Arduino IDE

El Entorno de Desarrollo Integrado es un editor de código abierto en el software Arduino, dirigido a reducir la complejidad de programación y comunicación con la placa. Es compatible su ejecución en sistemas operativos como Windows, Mac y Linux (Arduino, 2019).

Arduino IDE está basado en lenguaje de programación en Java, el procesamiento de datos y compatibilidad con software de código abierto. Este software se puede usar con cualquier placa Arduino, NodeMCU, WeMos D1 mini ESP8266 y otras tarjetas programables (Arduino, 2019).

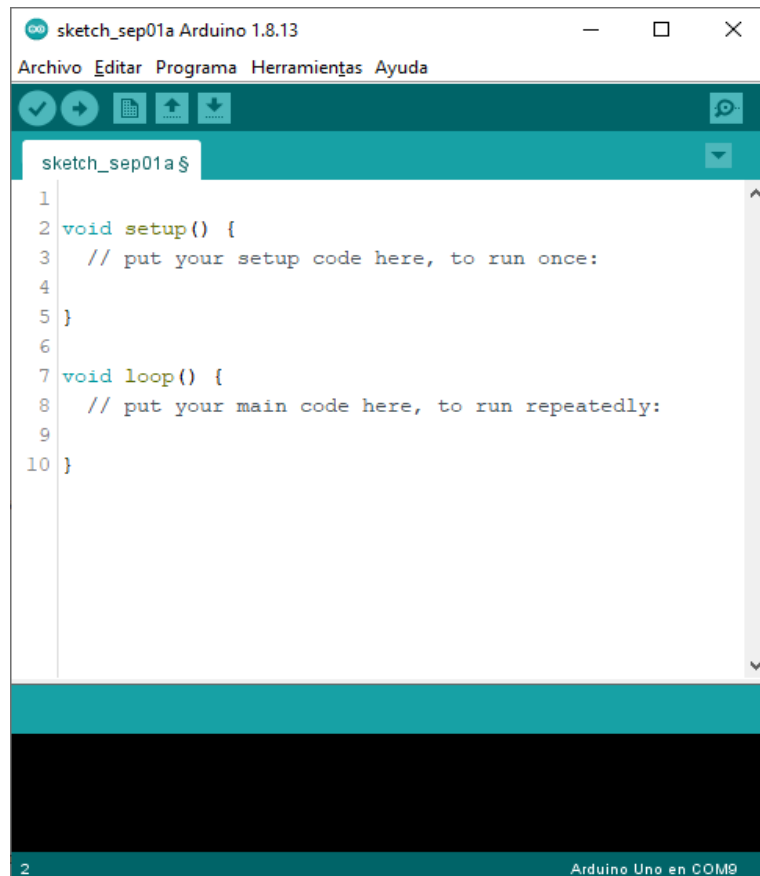


Figura 17. Arduino IDE

En la Figura 17, representa la interfaz del Arduino IDE o el propio software para realizar la programación, donde cada programa abierto se le llaman sketch o bocetos. Estos sketches disponen de un editor de texto que se escribe en el área de programación (fondo de color blanco); al guardar un sketch este genera una extensión de archivo (ino).

B. ThingSpeak

Es un servicio de plataforma web para el desarrollo de IoT que facilita adquirir los datos de los sensores para recopilar, almacenar en un canal privado o público en la nube proporcionando un fácil acceso a los datos, además analizar el

flujo de datos en tiempo real, visualizar mediante el acceso con Matlab (ThingSpeak, 2020).

En la Figura 18, se muestra el diagrama de un sistema de IoT donde incluye estos elementos como los sensores inteligentes, la nube (como ThingSpeak) donde se almacenan los datos y finalmente el desarrollo de un algoritmo asociado a las aplicaciones de IoT (ThingSpeak, 2020).

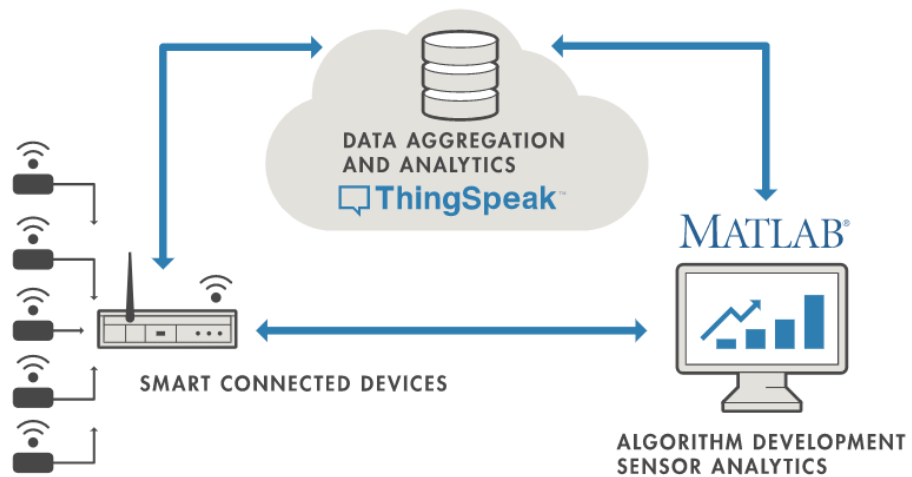


Figura 18. Diagrama del sistema IoT ThingSpeak.

Fuente: ThingSpeak

Se debe señalar que se cuenta con una biblioteca llamada Mathworks/ThingSpeak-Arduino que permite la compatibilidad con Arduino y WeMos D1 mini ESP8266 con ThingSpeak (GitHub, 2019).

Los datos de los sensores pueden enviarse a la nube de ThingSpeak mediante una Interfaz de Programa de Aplicación (API) que pueda comunicarse a cualquier Hardware, que este caso es el WeMos D1 mini ESP8266.

Según ThingSpeak posee algunas capacidades como:

- Visualizar los datos a partir de los sensores en tiempo real.
- Ajustes en los dispositivos para enviar datos a ThingSpeak
- Utilizar Matlab para generar análisis en los datos.
- Crear sistemas IoT sin el requerimiento de servidores ni desarrollo de páginas web.

- Compartir o actuar sobre los datos enviando a servicios de terceros como Twitter.

2.3. MARCO METODOLÓGICO

2.3.1. Tipo de investigación

El tipo de investigación que se está realizando en este caso es una investigación aplicada tecnológica porque se hace uso de la tecnología para interactuar con las personas. En este caso, la prolongada exposición a diario de la radiación ultravioleta que afecta negativamente en la salud a las personas y además proporcionar una predicción en los siguientes días sobre esos niveles mediante sistema de medición y pronóstico de radiación ultravioleta.

2.3.2. Metodología de la investigación

- Las pruebas del sistema de medición UV es de modo general, en cualquier ámbito, podría aplicarse en sitios públicos como en las avenidas donde hay multitud de transeúntes, ambulantes o hasta los policías de tránsito que se encuentran expuestos continuamente a la radiación UV.
- Los valores de IUV se van acumulando en ThingSpeak con actualización constante, solamente se va tomar como muestra intervalos de 5 minuto con su respectivo IUV y para aplicar con el algoritmo de *Machine Learning* con el fin de determinar un pronóstico.
- La presente tesis, se basa en un diseño de investigación teórico-práctico, que se basa en las definiciones de la tecnología de IoT y ML, así como también el sensor y microcontrolador.

2.4. MARCO LEGAL

2.4.1. IEEE 802.11 b/n/g

Son un conjunto de estándares para el sistema de comunicación inalámbrica en el caso de Wi-Fi.

A. IEEE 802.11b

Funciona con conexiones de hasta 11mbps, opera en la banda de 2.4GHz.

B. IEEE 802.11g

Funciona con conexiones de hasta 54mbps, opera en la banda de 2.4GHz.

C. IEEE 802.11n

Funciona con conexiones de hasta 600mbps, opera en la banda de 2.4GHz y 5GHz.

2.4.2. Ley N° 30102

A. Artículo 2

Se entiende que las entidades e instituciones públicas y privadas deben informar, acreditar de instrumentos de medición e informar sobre los riesgos a la exposición ultravioleta.

B. Artículo 7

Declara que el SENAMHI debe difundir diariamente los niveles de radiación solar en todo el país, así como los efectos nocivos para la salud.

2.5. ARQUITECTURA DEL SISTEMA

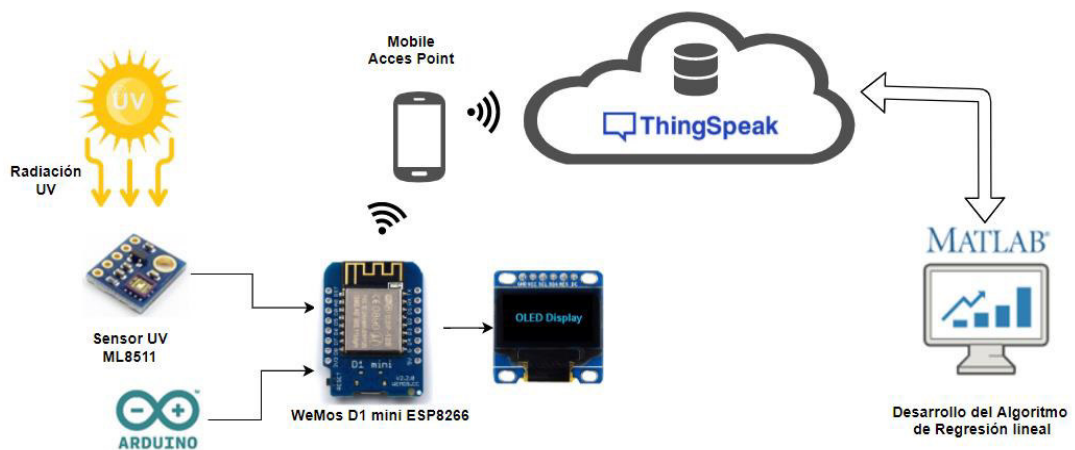


Figura 19. Diagrama de bloques del sistema de medición.

En la Figura 19, se muestra la arquitectura del sistema de medición y predicción de radiación por lo que utiliza un sensor de radiación UV ML8511 que captura la intensidad de radiación UV y entrega un nivel de señal analógica dependiendo de la radiación que detecta. La señal analógica será conectada a la plataforma WeMos D1 mini ESP8266 que viene integrada un convertor analógico-digital (ADC). Una vez que se obtiene la señal analógica, en el WeMos D1 mini ESP8266 se realiza el programa para

realizar el procesamiento los datos utilizando el software Arduino IDE que es compatible.

WeMos D1 mini ESP8266 tiene incorporada un módulo Wi-Fi, en que los datos procesados son transmitidos hacia una plataforma de IoT ThingSpeak, por lo que es necesario incluir esto se declara en la programación respectiva del Arduino asignando comandos para conectarse hacia una red conociendo el nombre y la contraseña de la red Wi-Fi. Los valores de IUV se van acumulando en ThingSpeak con actualización constante donde se puede visualizar gráfico y un indicador numérico en tiempo real. En ThingSpeak se puede realizar análisis con el software Matlab permitiendo desarrollar cualquier tipo de algoritmo e interactuar con los datos en la nube, de este modo se desarrolla un algoritmo de regresión lineal con tendencia de actualizaciones instantáneas para un pronóstico con menos errores.

CAPÍTULO III: DESARROLLO DEL SISTEMA

3.1. DESCRIPCIÓN DE LA SITUACIÓN ACTUAL DE LA MEDICIÓN DE LA RADIACIÓN UV.

3.1.1. Situación actual de la medición de la radiación UV en el Perú.

En el Perú, según el Instituto Nacional de Estadísticas e Informática (INEI) y el Servicio Nacional de Meteorología e Hidrología (SENAMHI), el año 2015 se registró un índice promedio de radiación UV de 11, para el año 2016 este valor aumento a 12 y para el año 2017 este valor desciende en un 33.3% alcanzando un nivel promedio de radiación UV 9, siendo estos índices considerados muy altos, perjudicales para los seres humanos. Esto resume en la Figura 20, donde se muestra los datos promedios de la radiación (Instituto Nacional de Estadística e Informática, 2018).

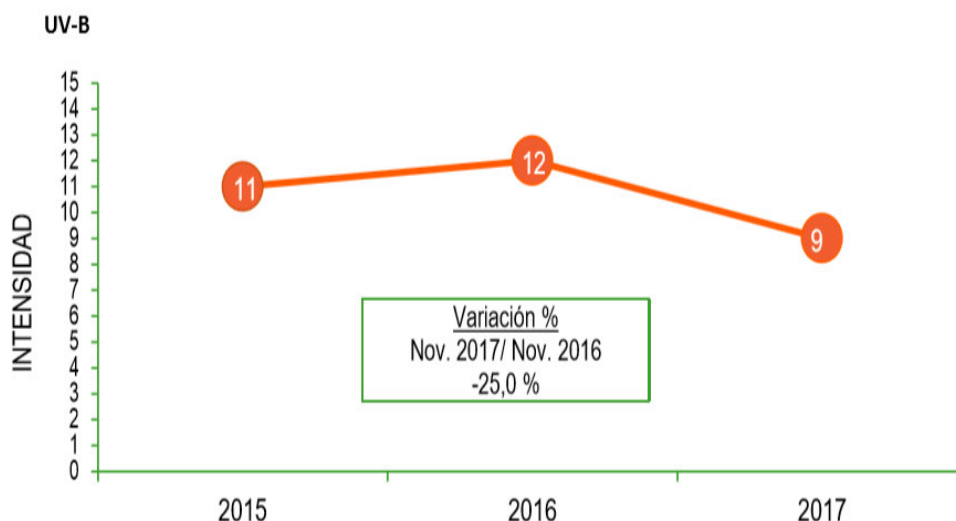


Figura 20. Datos promedios sobre la radiación UV en el año 2015 - 2017.

Fuente: Instituto Nacional de Estadística e Informática.

En la región de Lima, durante el año 2016 se registró que el índice máximo de radiación ultravioleta ocurre en el mes de febrero (Figura 21). Mientras tanto, en la Figura 22, se interpreta que en el mes de febrero se dio el pico más alto de radiación mientras que en los otros meses fue de nivel bajo (Lima Como Vamos, 2018).

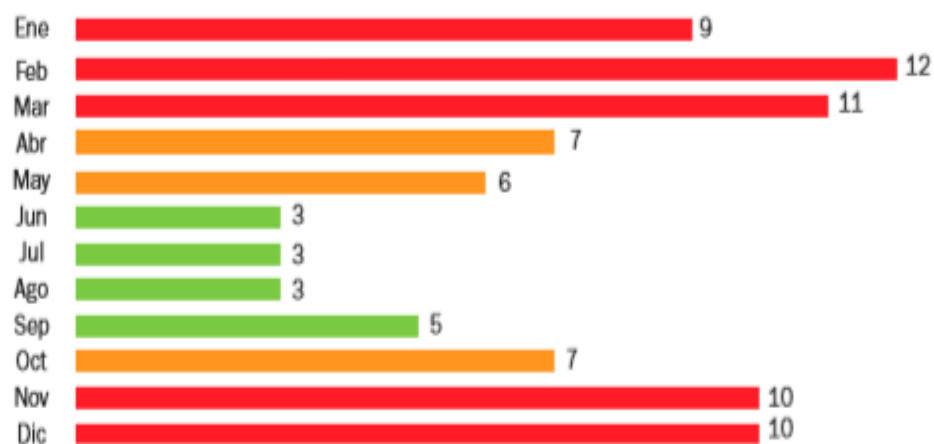


Figura 21. Índice de radiación ultravioleta en Lima 2016.

Fuente: Lima Como Vamos.

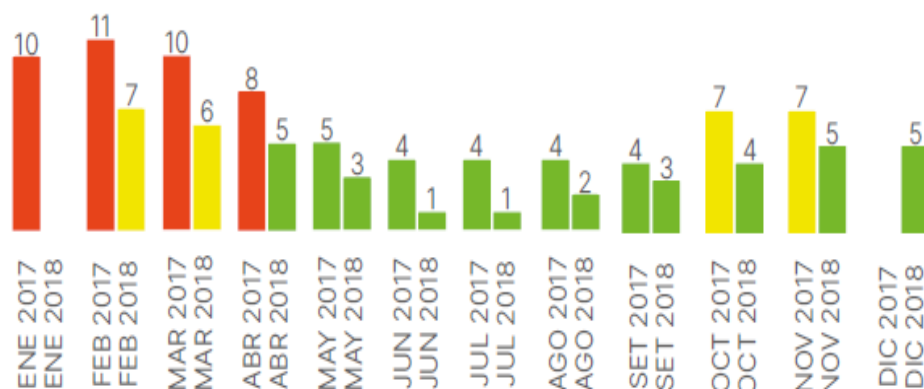


Figura 22. Radiación UV en Lima Metropolitana, 2017-2018.

Fuente: Lima Como Vamos.

3.1.2. Instrumentos de medición de radiación UV.

A. Luxómetro medidor de luz UV

Este equipo se usa para la medición de radiación UVA y UVB con una peculiaridad de lectura a nivel profesional. Posee un rango de medición de 290nm a 390nm del espectro de radiación. El sensor incorpora un diseño de un filtro coseno para la corregir posibles errores con un tiempo de muestreo de 1 segundo en tiempo de muestreo. (Figura 23).



Figura 23. Luxómetro medidor de luz UV YK-35UV.

Fuente: Valiometro

B. Medidor digital de luz/ radiación ultravioleta rayos UV-A y UV-B.

Este equipo de medición de luz UVA-UVB (Figura 24) posee aplicaciones para medir frecuencias de UVA proveniente del sol, industrias con radiaciones UVA elevadas, laboratorios de virología, esterilización, etc.



Figura 24. Medidor digital de radiación para rayos UVA/UVB UV 340B

Fuente: ServoVendi

C. Medidor de radiación solar

Este equipo incorpora una celda solar de silicio monocristalina que permite captar la intensidad de la luz solar e inmediatamente llevarlo al procesador incorporado, el cual se encarga de la corrección automática del resultado manteniendo la calibración después de cada prueba (Figura 25).



Figura 25. Medidor de radiación solar PCE-SPM 1.

Fuente: PCE-Instruments.

3.2. DIAGRAMA DE LAS ETAPAS DEL SISTEMA DE MEDICIÓN Y PRONOSTICACIÓN DE RADIACIÓN ULTRAVIOLETA.

3.2.1. Etapas del sistema

Para comprender el funcionamiento del sistema de medición de radiación UV, la conectividad a Internet de las Cosas y realizar el algoritmo de *Machine Learning* para el pronóstico de radiación UV; se presentará el siguiente diagrama de bloques en la Figura 26.

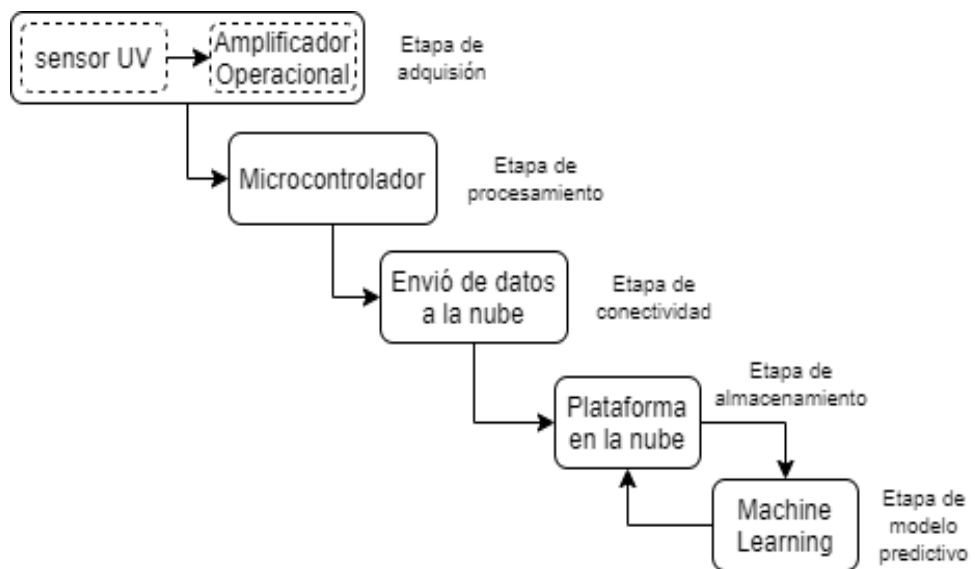


Figura 26. Diagrama de bloques del sistema.

A. Etapa de adquisición

La etapa de adquisición de datos está enfocada a la medición de la radiación ultravioleta a través del sensor UV ML8511, el cual incorpora un amplificador operacional, así convierte la fotocorriente en voltaje según la intensidad de radiación UV en determinado lugar.

B. Etapa de procesamiento

La variación de voltaje entregada por el sensor UV, es una señal analógica por lo que el núcleo de cualquier procesador solo trabaja con señales analógicas. Afortunadamente, la mayoría de microcontroladores o en el caso del WeMos D1 mini ESP8266 viene incorporada un convertidor análogo-digital (ADC) con una resolución de 10 bits. El procesamiento para realizar la

conversión de voltaje a intensidad de radiación UV es mediante un algoritmo desarrollado en la programación Arduino IDE.

C. Etapa de conectividad

Es necesario que los datos de la intensidad de radiación UV sean transmitidos por una comunicación inalámbrica, como la red Wi-Fi enviando los datos a Internet hacia un servidor. El dispositivo que permite enviar los datos procesados del microcontrolador hacia el internet es el módulo Wi-Fi incorporado en dicho microcontrolador.

D. Etapa de almacenamiento

Una vez que los datos sean procesados es necesario almacenar para posibles futuras aplicaciones. Por lo que, en la etapa de almacenamiento, los datos son registrados en una plataforma en la nube de Internet.

E. Etapa de modelo predictivo

Con los datos almacenados se procede a realizar la ciencia de datos como algoritmos de aprendizaje automático o *Machine Learning* para desarrollar modelamiento predictivo que guía a un pronóstico en el comportamiento de los datos.

3.2.2. Características del sistema

El dispositivo deberá cumplir las siguientes características:

- Portátil: Fácil manera para el desplazamiento.
- Peso liviano: Para el fácil transporte es necesario que el peso sea lo más mínimo posible.
- Visualización: El sistema deberá mostrar los índices de radiación UV en cada instante de tiempo.

3.2.3. Diseño del sistema

A. Diagrama de conexión del sistema

En la Figura 27 se representa las conexiones entre los dispositivos del sistema de radiación UV. El sistema es alimentado energéticamente por una batería de litio de 3.7v - 450 mAh hasta un determinado tiempo de descarga. El módulo de carga TIP4056 suministra un control en la carga de la batería para que continúe el funcionamiento del sistema. El

microcontrolador WemosD1 está alimentado directamente con un voltaje de 5v, la pantalla OLED y el sensor UV esta alimentado con un voltaje de 3.3v suministrado por el regulador de voltaje integrado en el WeMos D1 mini ESP8266. Adicionalmente, el pin En (*enable*) del sensor UV está conectado al voltaje de 3.3v; el pin “out” conectado a la entrada analógica del WemosD1. La interfaz I2C (SDA y SCL) de la pantalla OLED se conecta en los pines correspondiente I2C del WeMos D1 mini ESP8266 (D2 y D1).

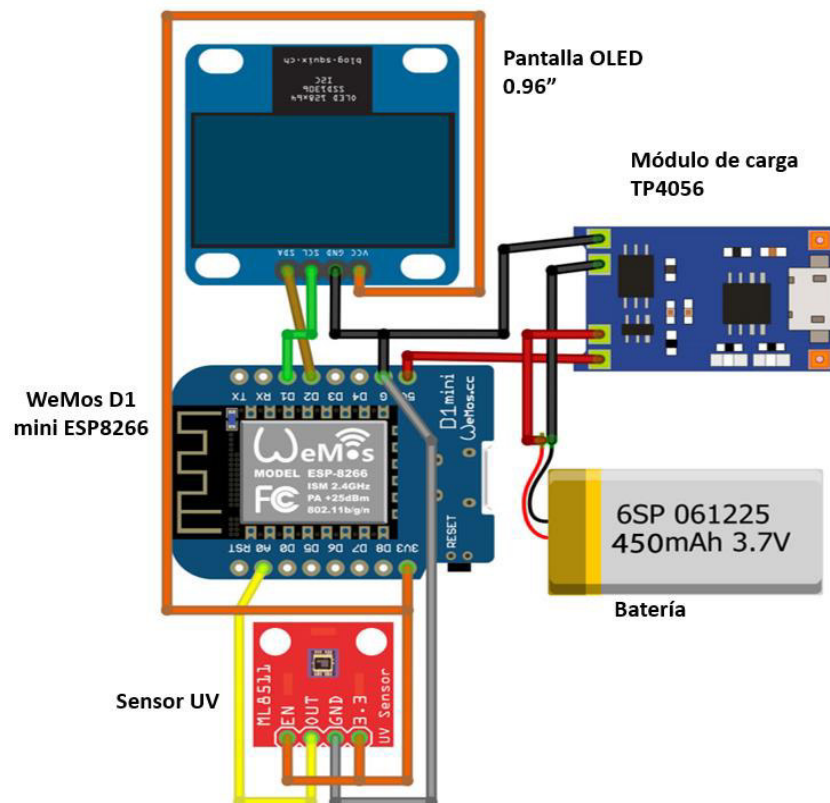


Figura 27. Diagrama de conexión del sistema.

B. Consumo de energético

El consumo de corriente del sistema está determinado por la cantidad de energía requerida promedio de los dispositivos conectados. El consumo total de corriente promedio se calcula como la suma algebraica de cada componente interconectado, tal como se menciona en la Tabla 5.

Tabla 5. Consumo eléctrico de los componentes.

COMPONENTES	CONSUMO DE CORRIENTE
WeMos D1 mini ESP8266	66.58 mA
Sensor UV ML8511	223 uA
Pantalla OLED 0.96	15 mA
Módulo de carga TP4056	8.2 mA
TOTAL	90 mA

C. Autonomía estimada del sistema

El tiempo de duración depende del consumo energético por parte de los dispositivos del sistema y de la capacidad de la batería instalada de 3.7voltios/450mAH. El cálculo para determinar el tiempo de duración o descarga, se realiza mediante la siguiente ecuación:

$$\text{Tiempo de descarga } (t) = \frac{\text{Capacidad de la batería}}{\text{Consumo de corriente total}} \quad (2)$$

$$\text{Tiempo de descarga}(t) = \frac{450 \text{ mAh}}{90 \text{ mA}}$$

$$\text{Tiempo de descarga}(t) = 5 \text{ horas}$$

D. Modelos 3D

Los dispositivos electrónicos que permite el funcionamiento del sistema de radiación UV deben estar ubicados y ordenado en un solo material o producto. Para esto se realizado un diseño en 3D mediante el software FreeCAD, con las dimensiones exactas de los dispositivos.

En la Figura 28, en (a) se muestra el modelo base de un prototipo donde abarca una espacio limitado y suficiente para la instalación de la batería, el módulo de carga y el WeMos D1 mini ESP8266. En la (b), se muestra la tapa para cerrar la base del prototipo, en la tapa se encuentra instalado la pantalla OLED 0.96" y el sensor UV.

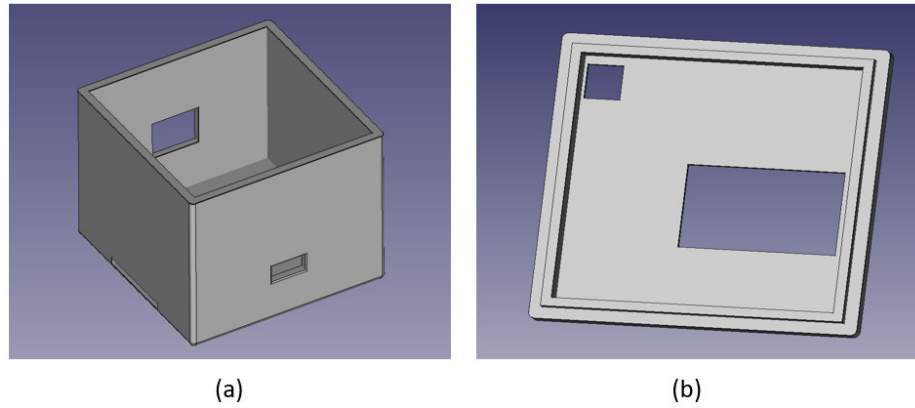


Figura 28. Diseño de un modelo base.

En la Figura 29, se muestra el modelo del prototipo total, uniendo el modelo base y la tapa de la Figura 28.

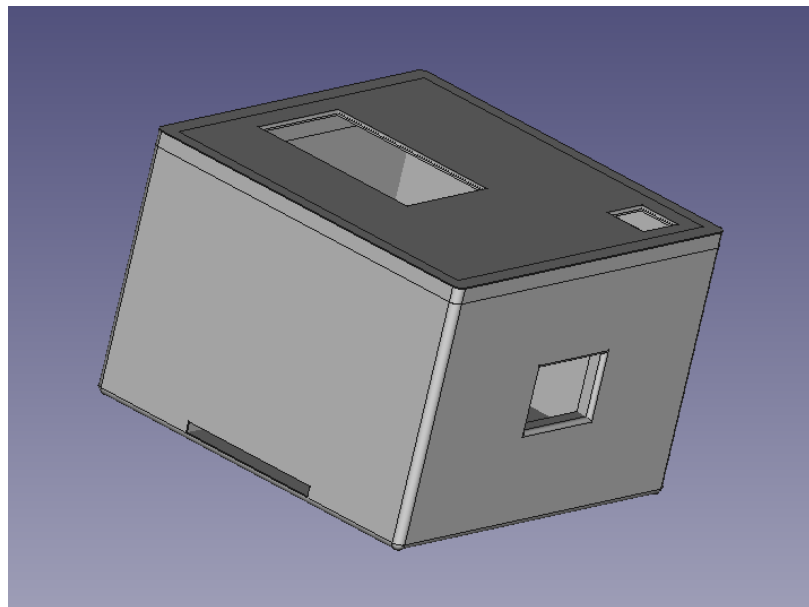


Figura 29. Modelo del case para el sistema de radiación UV.

En la Tabla 6, se muestra las dimensiones del case para el sistema de medición y pronóstico UV.

Tabla 6. Dimensiones del sistema

DIMENSIONES	VALOR
Largo	5.0 cm
Ancho	5.0 cm
Altura	3.5 cm
Grosor	2.0 mm

3.2.4. Diagrama de flujo

En la Figura 30, se muestra el diagrama de flujo del proceso de funcionamiento para el sistema de medición y pronóstico UV.

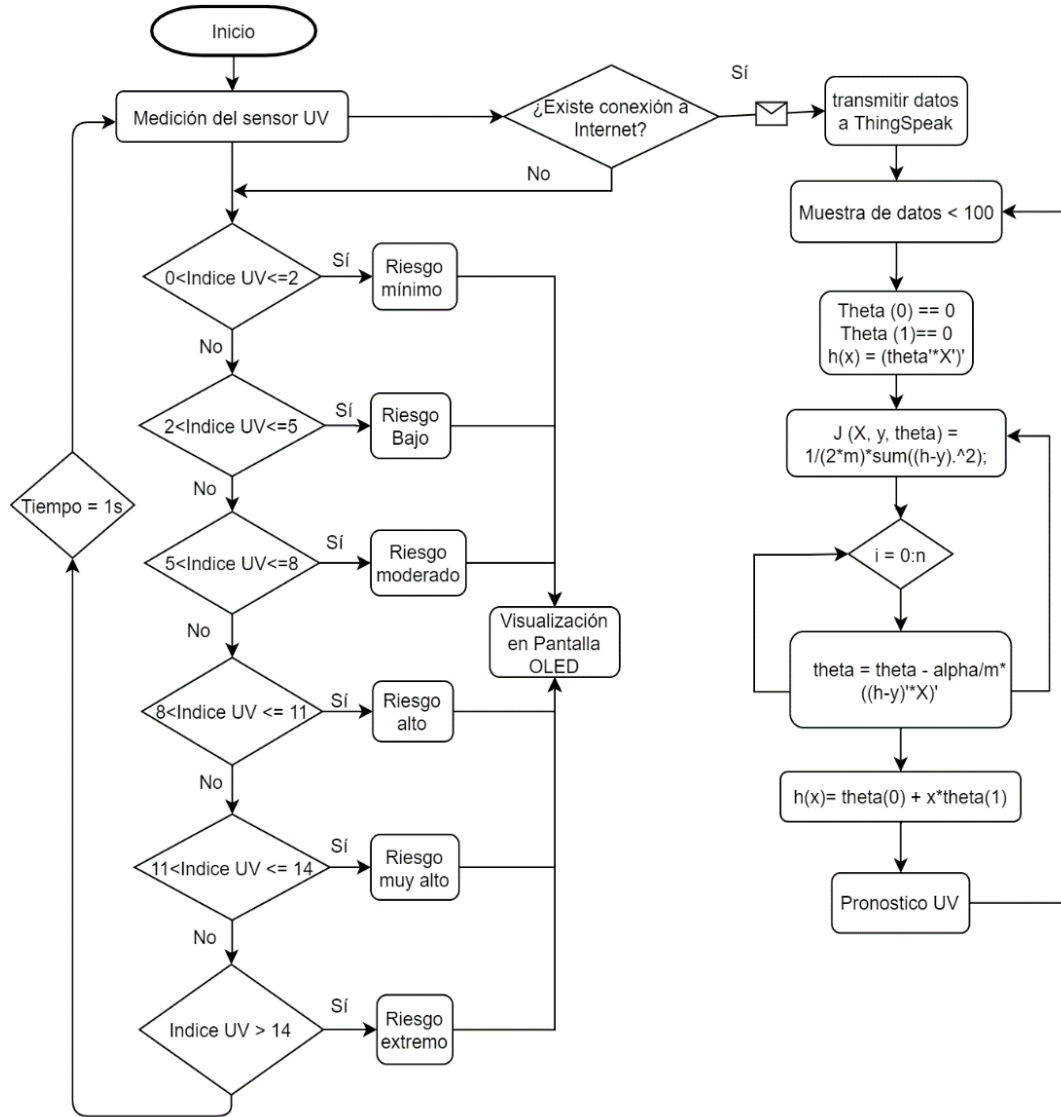


Figura 30. Diagrama de flujo para el sistema de medición y pronóstico UV.

3.2.5. Programación en Arduino IDE

Teniendo la parte de los dispositivos electrónicos como el sensor de radiación UV, la pantalla OLED 0.96" y la plataforma WeMos D1 Mini ESP8266 conectados en los respectivos pines que se ha visto en la Figura 27.

El siguiente paso es programar, permitiendo dar la funcionalidad al proyecto implementado; utilizando el software IDE de Arduino, se programó lo siguiente:

A. Insertar librerías

Como se está usando un WeMos D1 mini ESP8266, se requiere insertar la librería correspondiente.

```
#include <ESP8266WiFi.h>
```

La pantalla OLED 0.96" necesita 2 librería para su funcionamiento, la primera es para pantallas gráficas y la otra, para el controlador SSD1306.

```
#include <Adafruit_GFX.h>
```

```
#include <Adafruit_SSD1306.h>
```

Por último, la librería **Wire** que permite controlar dispositivos que trabajan con el protocolo de comunicación I2C.

```
#include <Wire.h>
```

B. Declarar variables

Se declara una variable UVOUT de tipo entero que almacenará la lectura en el pin analógico A0 donde está conectado el pin de datos del sensor UV.

```
int UVOUT = A0;
```

Se declara 2 variables de tipo *long* para el tiempo de actualización.

```
unsigned long previousMillis = 0;
```

```
const long interval = 1000;
```

Se declara la variable de ANCHO y ALTO con valores numérico de 128 y 64 respectivamente, que representa los pixeles de la pantalla OLED 0.96" y además un variable para el RESET.

```
#define ANCHO 128
```

```
#define ALTO 64
```

```
#define OLED_RESET D4
```

```
Adafruit_SSD1306 oled (ANCHO, ALTO, &Wire, OLED_RESET);
```

C. Configuración de las variables *void setup*

Dentro del campo del *void setup*, se procede a configurar las variables que permitan funcionar el programa. Se agregan los comandos como, la inicialización de la comunicación serial entre la plataforma WeMos y la computadora a una velocidad de 115200 baudios.

```
Serial.begin(115200);
```

Inicializar el protocolo de comunicación I²C.

```
Wire.begin();
```

Inicializa la pantalla OLED 0.96" con una dirección 0x3C.

```
oled.begin(SSD1306_SWITCHCAPVCC, 0x3C);
```

D. Creación de funciones definidas por el usuario

Se crean 2 funciones en la programación para segmentar parte del código especificando tareas determinadas, que luego son declaradas en el *void loop*. Se crea una función **averageAnalogRead** que permite realizar un promedio aritmético de 8 lecturas de datos del sensor UV.

```
int averageAnalogRead (int pinToRead)
{
    byte numberOfReadings = 8;
    unsigned int runningValue = 0;

    for(int x = 0; x < numberOfReadings ; x++)
        runningValue += analogRead(pinToRead);
    runningValue /= numberOfReadings;

    return(runningValue);
}
```

La segunda función es **map** del tipo flotante para valores del mismo tipo, sirve para mapear los valores o realizar un cambio en la escala de los valores.

```
float mapfloat(float x, float in_min, float in_max, float out_min, float
out_max)
{
  return (x - in_min)*(out_max - out_min)/(in_max - in_min) + out_min;
}
```

E. Función principal *void loop*

La función principal es la parte importante del código, establecemos un tiempo de actualización.

```
unsigned long currentMillis = millis ();
if (currentMillis - previousMillis >= interval) {
  previousMillis = currentMillis;
```

Se declara una variable del tipo entero llamado uvLevel que almacena función **averageAnalogRead**.

```
int uvLevel = averageAnalogRead (UVOUT);
```

La siguiente sentencia es para convertir los valores de lectura analógica en voltaje.

```
float outputVoltage = (3.3 / 1023.0) * uvLevel;
```

Agregando una nueva variable de tipo flotante **uvIntensity** que almacena la función **mapfloat** que permitirá convertir la relación del voltaje a niveles de intensidad UV, de acuerdo como se visualizó en la Figura 14.

```
float uvIntensity = mapfloat (outputVoltage, 0.99, 2.8, 0.0,15,0);
```

Establecemos los siguientes comandos para imprimir los textos y valores del sensor UV.

```
oled.clearDisplay();
oled.setTextColor(WHITE);
oled.setCursor(0, 0);
oled.setTextSize(1);
oled.print (" Sistema de medicion");
oled.print(" de radiacion UV ");
oled.setCursor (8, 25);
oled.print("Indice UV: ");
oled.print(intensity);
```



```

Serial.print(" ML8511 output: ");
Serial.print(uvLevel);

Serial.print(" / ML8511 voltage: ");
Serial.print(outputVoltage);

Serial.print(" / UV Intensity (mW/cm^2): ");
Serial.println(uvIntensity);

```

Asignamos las siguientes estructuras de control para determinar el nivel de riesgo con respecto la intensidad UV, como se visualizó en la Figura 8.

```

if (uvIntensity >= 0 && uvIntensity <=2){ //1-2
  oled.setCursor (0, 40);
  oled.print("Riesgo: Minimo");
}
if (uvIntensity > 2 && uvIntensity <=5){
  oled.setCursor (0, 40);
  oled.print("Riesgo: Bajo");
}
if (uvIntensity > 6 && uvIntensity <=8){
  oled.setCursor (0, 40);
  oled.print("Riesgo: Moderado");
}
if (uvIntensity > 8 && uvIntensity <=11){
  oled.setCursor (0, 40);
  oled.print("Riesgo: Alto");
}
if (uvIntensity >11 && uvIntensity <=14){
  oled.setCursor (0, 40);
  oled.print("Riesgo: Muy alto");
}
if (uvIntensity > 14){
  oled.setCursor (0, 40);
  oled.print("Riesgo: Extremo");
}

```

3.3. IDENTIFICACIÓN DE UNA PLATAFORMA IoT CON SOPORTE DE APRENDIZAJE AUTOMÁTICO.

3.3.1. Plataformas IoT

Existe diversas cantidades disponible de plataformas para el desarrollo de proyectos en IoT, que se caracteriza en el almacenamiento de los datos de los sensores en la nube principalmente. En caso, de realizar un análisis con los datos almacenados en la nube.

A. ThingSpeak

ThingSpeak es un servidor web de análisis de IoT permite asignar, monitorear y analizar los datos en tiempo real en la nube. Desde cualquier dispositivo se puede crear una visualización con actualización instantáneas y monitorear desde una aplicación llamada ThingView en un dispositivo móvil. (ThingSpeak, 2020).

Características:

- Facilidad en la configuración de la programación para enviar datos a ThingSpeak mediante protocolos de IoT.
- Monitorea y visualiza en tiempo real los datos por parte de los sensores.
- Capacidad de enlazar con MATLAB para dar utilidad a los datos de IoT.
- Crear sistemas de IoT sin la necesidad de crear servidores ni software web.

B. Thinger IO

Es una plataforma de código abierto para Internet de las cosas, proporciona una infraestructura de nube escalable lista para usar para conectar cosas. Los fabricantes y las empresas pueden comenzar a controlar sus dispositivos desde Internet en minutos, sin preocuparse por la infraestructura de nube requerida (Thinger IO, 2020).

Características:

- Codificación fácil en la programación del microcontrolador.

- Agnóstico de hardware: conecta con Arduino, ESP8266, Raspberry pi, Intel Edison, etc.
- Plataforma en la nube, proporciona una infraestructura de nube escalable para conectar sensores.
- Fuente Abierta, crear servidores en la propia nube.

C. Google Cloud

Es una plataforma de desarrollo de aplicaciones mediante el uso de software libre y de otros dispositivos. Gestiona los datos de manera segura en la nube, además una ventaja que caracteriza es de poseer una nube abierta con tecnología abierta y flexible. (Google Cloud, 2020)

Características:

- Estadísticas inteligentes en una plataforma de sin servidores.
- Seguridad en los datos para la protección de la empresa.
- Disposición de Inteligencia Artificial (IA) y ML.
- Requiere un pago para adquirir estos servicios.

D. Ubidots

Ubidots IoT plataforma desarrollada para aplicaciones en la nube para la fabricación de dispositivos, computación en la nube e implementación. (Ubidots, 2020)

Características:

- Conecta fácilmente hardware y fuente de datos en la nube.
- Desarrolla aplicaciones en la plataforma en la nube.
- Permite implementar aplicaciones personalizadas para usuarios finales.

E. Microsoft Azure

Es un servicio de informática en la nube permitiendo en el soporte para la organización y en los retos de negocios empresariales. Azure posee una flexibilidad en generar, administrar y desarrollar aplicaciones en una red global con múltiples herramientas y plataformas. (Microsoft Azure, 2020)

Características:

- Permite crear aplicaciones personalizadas con IA.
- Moderniza las aplicaciones sin uso del servidor.
- Implementa máquinas virtuales como Windows y Linux en la nube.

F. Adafruit IO

Adafruit.io es un servicio en la nube. Puede conectarse a través de Internet. Está destinado principalmente para almacenar y luego recuperar datos. (Adafruit IO, 2018).

Características:

- Permite visualizar los datos en tiempo real.
- Facilidad de conectar un proyecto a otros dispositivos para Internet.
- Permite conectar los datos a un servicio web como Twitter, feed, etc.

G. Otras plataformas

En la Tabla 7, se describe una lista adicional de plataformas de IoT que son los más comunes que se puede trabajar, se presenta con sus principales características.

Tabla 7. Plataformas de IoT.

NOMBRE	CARACTERÍSTICAS	CÓDIGO ABIERTO
<i>Nimbits</i>	Registro de datos en la nube.	Sí
<i>Idigi</i>	Plataforma en la nube para dispositivos.	No
<i>Sensorcloud</i>	Visualiza y almacena datos de Sensores en línea.	No
<i>Exosite</i>	Plataforma y portales de datos basados en la nube y gestión de dispositivos.	No
<i>Everything</i>	Motor de software y plataforma de la nube.	No
<i>Paraimpu</i>	Herramientas sociales para las cosas.	No
<i>Lelylan</i>	Orientado en automatización del hogar y monitoreo.	No

Fuente: Doukas

3.3.2. Protocolo de comunicación HTTP.

El protocolo de Transferencia de Hipertexto (HTTP), es un protocolo de comunicación de red para el intercambio de información en diferentes formatos. HTTP está diseñado para habilitar las comunicaciones en un modelo cliente/servidor, funcionando como protocolo de petición (*request*) y respuesta (*response*). La plataforma WeMos D1 mini ESP8266 tiene como función principal la comunicación usando las redes Wi-Fi y se implementará en modo cliente HTTP y solicitar el servicio de IoT ThingSpeak, como se muestra en la Figura 31.

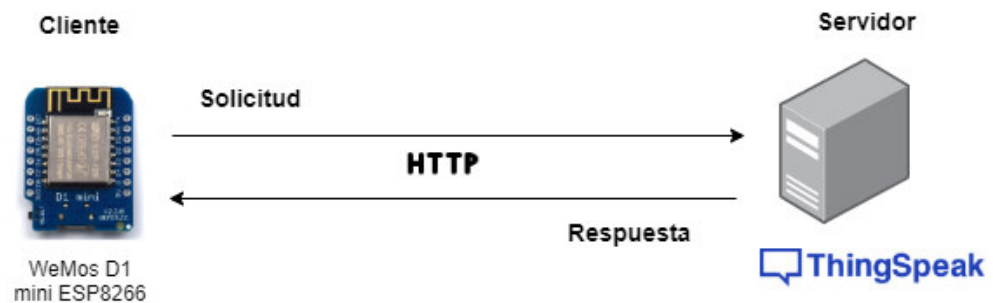


Figura 31. Esquema de solicitud cliente-servidor

En una comunicación HTTP existen 2 métodos comunes en solicitud y respuesta entre cliente – servidor, que son GET y POST.

3.3.3. Configuración de la plataforma IoT ThingSpeak.

A. Registro de cuenta

Para usar la plataforma ThingSpeak es necesario el registro de cuenta inscribiendo los datos personales (nombres y apellidos), correo electrónico, así como se muestra en la Figura 32.

Figura 32. Registro de usuario en ThingSpeak.

B. Entorno de monitorización

Se caracteriza por poseer gráficos denominados campos y opciones de agregar tableros o *dashboard* e indicadores numéricos mostrando la lectura del sensor en tiempo real, ver Figura 33. Los usuarios externos o usuarios que no registran una cuenta en la plataforma, pueden acceder a visualizar el comportamiento de los datos del sensor.

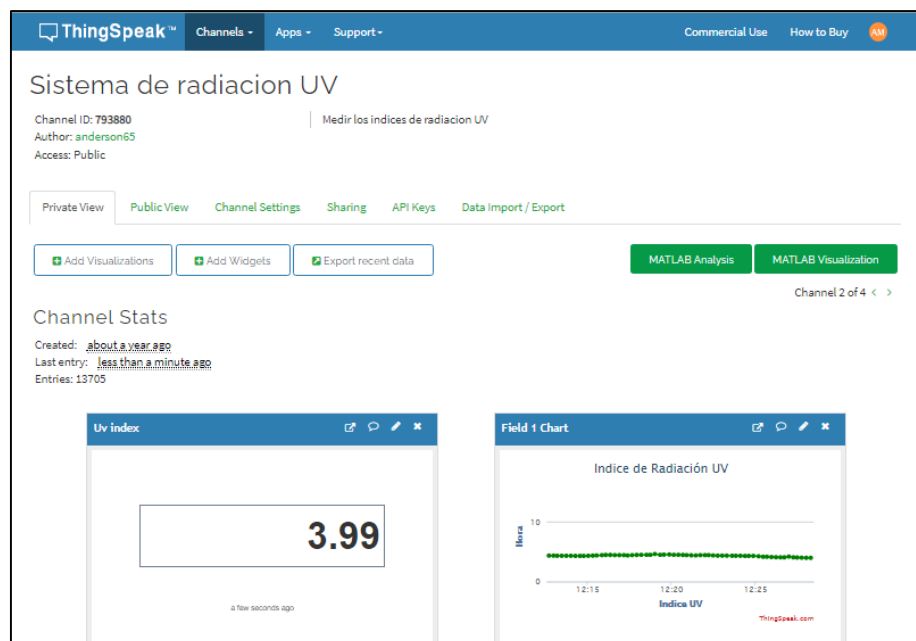
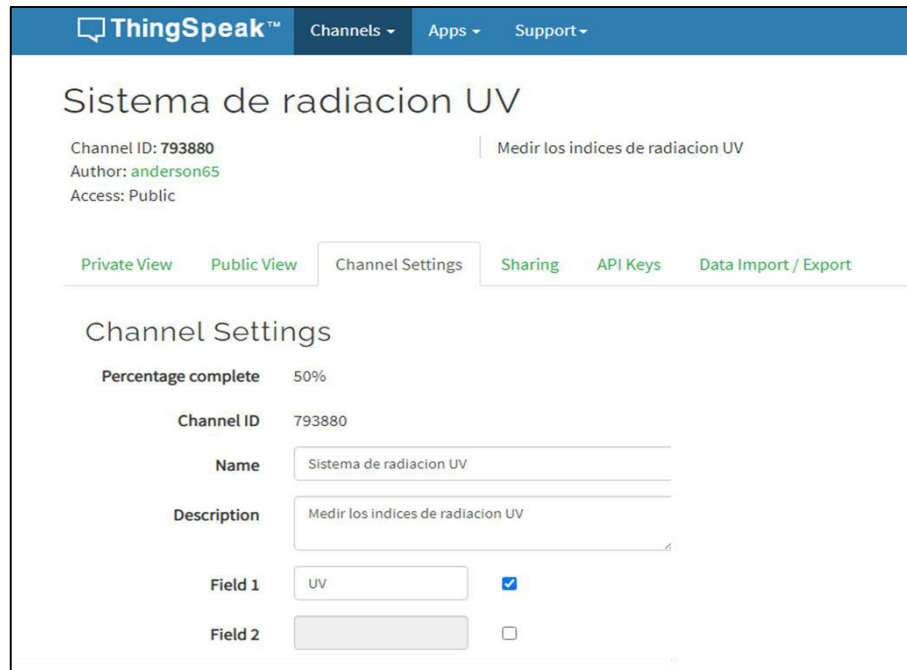


Figura 33. Entorno gráfico de ThingSpeak.

C. Configuración del canal de la cuenta

Los datos van estar almacenados en el canal de ThingSpeak, cada canal dispone de 8 campos para diversos datos como 8 sensores que serán visualizados. Rellenar textualmente los espacios en blanco como, nombre del canal, alguna descripción y algunos campos. Solo se selecciona un campo para el índice de radiación UV que se necesita, se muestra en la Figura 34.



The screenshot shows the 'Channel Settings' page for a ThingSpeak channel titled 'Sistema de radiacion UV'. The channel ID is 793880, the author is 'anderson65', and the access is 'Public'. The page has tabs for 'Private View', 'Public View', 'Channel Settings', 'Sharing', 'API Keys', and 'Data Import / Export'. The 'Channel Settings' tab is active, showing a 'Percentage complete' of 50%. The 'Name' field is 'Sistema de radiacion UV' and the 'Description' is 'Medir los indices de radiacion UV'. There are two 'Field' settings: 'Field 1' is 'UV' with a checked checkbox, and 'Field 2' is empty with an unchecked checkbox.

Figura 34. Configuración del canal.

D. Registro del parámetro del sensor UV

Se requiere proceder un registro de sensor en la plataforma ThingSpeak mediante la configuración del campo 1 que ha sido habilitado en la configuración del canal de la cuenta. Se tiene que rellenar los espacios obligatorios como título de ejes, unidad de medición y tiempo de muestro (Figura 35).

The image shows a dialog box titled "Field 1 Chart Options" with a close button (X) in the top right corner. The dialog is organized into two columns of fields:

- Left Column:**
 - Title: Indice de Radiación UV
 - X-Axis: Hora
 - Y-Axis: Indice UV
 - Color: #008000
 - Background: #ffffff
 - Type: line (dropdown)
 - Dynamic?: true (dropdown)
 - Days: (empty)
 - Results: 60
- Right Column:**
 - Timescale: 10 (dropdown)
 - Average: (dropdown)
 - Median: (dropdown)
 - Sum: (dropdown)
 - Rounding: (empty)
 - Data Min: (empty)
 - Data Max: (empty)
 - Y-Axis Min: 0
 - Y-Axis Max: 15

At the bottom right, there are two buttons: "Save" (green) and "Cancel" (white with blue border).

Figura 35. Configuración del campo.

E. Registro de datos

Los datos deberán ser almacenados en la plataforma de ThingSpeak; sin embargo, para que esto cumpla en cada cuenta creada dispone un código de identificación de API Key, como en la Figura 36. En donde, este código deberá ser escrito durante la programación en el sketch de Arduino que luego es grabado en el WeMos D1 mini ESP8266.

The image shows a simple web interface for writing an API key. It has the following elements:

- Title: "Write API Key"
- Label: "Key" next to a text input field.
- Input: The text "B*****" is entered into the input field.
- Button: An orange button with the text "Generate New Write API Key" is positioned below the input field.

Figura 36. API Key del Campo de ThingSpeak.

3.3.4. Configuración del código de Arduino IDE para ThingSpeak

Para que el sistema de medición de radiación UV envíe, almacene y analice los datos en vivo es de suma importancia adquirir un servicio de plataforma de análisis de IoT como ThingSpeak. Para modificar el sistema de medición UV a un sistema IoT tenemos que agregar comando en la programación Arduino y que será grabado en el WeMos D1 mini ESP8266.

A. Declarar variables

Asignar las variables de tipo **string** para el código de identificación de API Key y variables de tipo **char** (caracteres) para el identificador **SSID** y la contraseña de la red Wi-Fi, el servidor ThingSpeak y además una variable cliente para enviar los datos al servidor.

```
String apiWritekey = "B*****";  
const char ssid [] = "R*****";  
const char password [] = "1*****";  
const char* server = "api.thingspeak.com";  
WiFiClient client;
```

B. Configurar las variables en *void setup*

Inicializa el módulo WeMos D1 mini ESP8266 para buscar la red Wi-Fi asignada con su respectivo SSID y contraseña.

```
WiFi.begin(ssid, password);
```

Establece una estructura de control condicional, si en caso el módulo WeMos D1 mini ESP8266 encuentra y se conectada a dicha red asignada.

```
if (WiFi.status() != WL_CONNECTED) {  
    delay(500);  
    Serial.print(".");  
}
```

Finalmente, se mostrará un mensaje en el monitor serie del Arduino indicando que el WeMos D1 mini ESP8266 se ha conectado a la red Wi-Fi.

C. Función principal *void loop*

Agregar la siguiente estructura de control condicional, si en caso el WeMos D1 mini ESP8266 (como cliente) se conecta al servidor ThingSpeak por el puerto 80 (HTTP) del protocolo TCP, envía los datos de intensidad UV y se almacena en el campo 1 del canal de ThingSpeak.

```
if (client.connect(server,80))
{
    String tsData = apiWritekey;
    tsData += "&field1=";
    tsData += String(uvIntensity);
    tsData += "\r\n\r\n";
    client.print("POST /update HTTP/1.1\n");
    client.print("Host: api.thingspeak.com\n");
    client.print("Connection: close\n");
    client.print("X-THINGSPEAKAPIKEY: "+apiWritekey+"\n");
    client.print ("Content-Type: application/x-www-form-
        urlencoded\n");
    client.print ("Content-Length: ");
    client.print(tsData.length());
    client.print("\n\n");
    client.print(tsData);
}
```

Adicionalmente, se agrega los siguientes comandos para notificar en la pantalla OLED 0.96" que el WeMos D1 mini ESP8266 se ha conectado a la red Wi-Fi.

3.3.5. Implementación del sistema

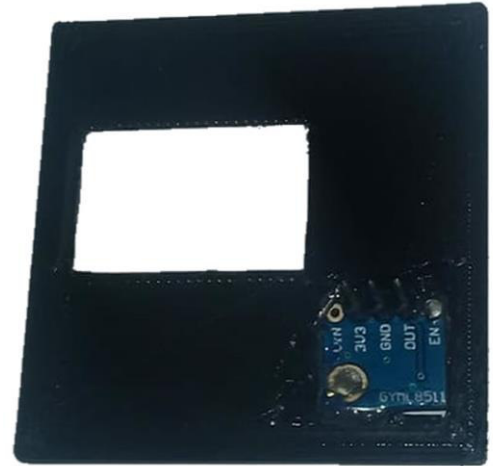
En esta parte, se realiza la implementación con los componentes electrónicos requeridos para el funcionamiento del sistema y así mismo, el posicionamiento dentro del case para la protección de los componentes que se incorporarán internamente.

Inicialmente se graba la programación a realizada en el IDE de Arduino en el WeMos D1 Mini ESP8266.

En el case, se incorpora el sensor de radiación UV en un orificio para la medición del índice UV en el ambiente (Figura 37).



Vista Superior



Vista Inferior

Figura 37. Incorporación del sensor de radiación UV.

Además, se incorpora el módulo de carga y del interruptor para el encendido del sistema de medición y pronóstico UV (Figura 38).



Figura 38. Incorporación del módulo de carga.

Posteriormente, se incorpora el WeMos d1 mini ESP 8266 y la pantalla oled al sistema para la visualización por parte del usuario (Figura 39).

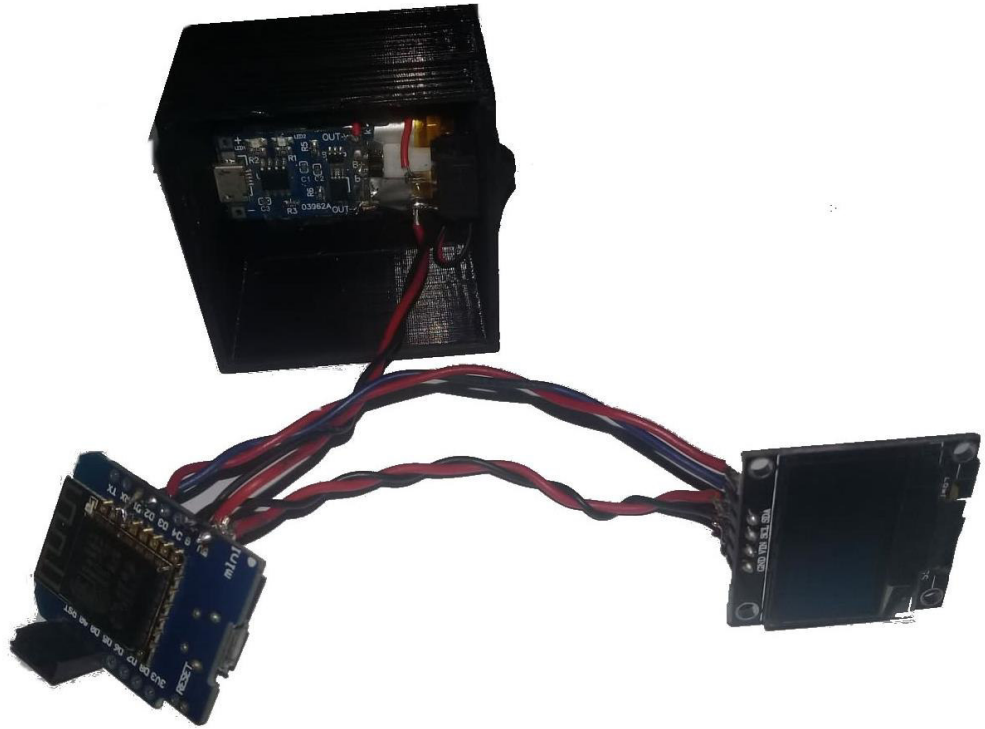


Figura 39. Incorporación del WeMos D1.

Finalmente, se realiza las pruebas de conexión de los componentes que se encuentran incorporados y el cerrado del case (Figura 40).



Figura 40. Implementación del sistema de medición y monitoreo UV.

3.3.6. Visualización de datos

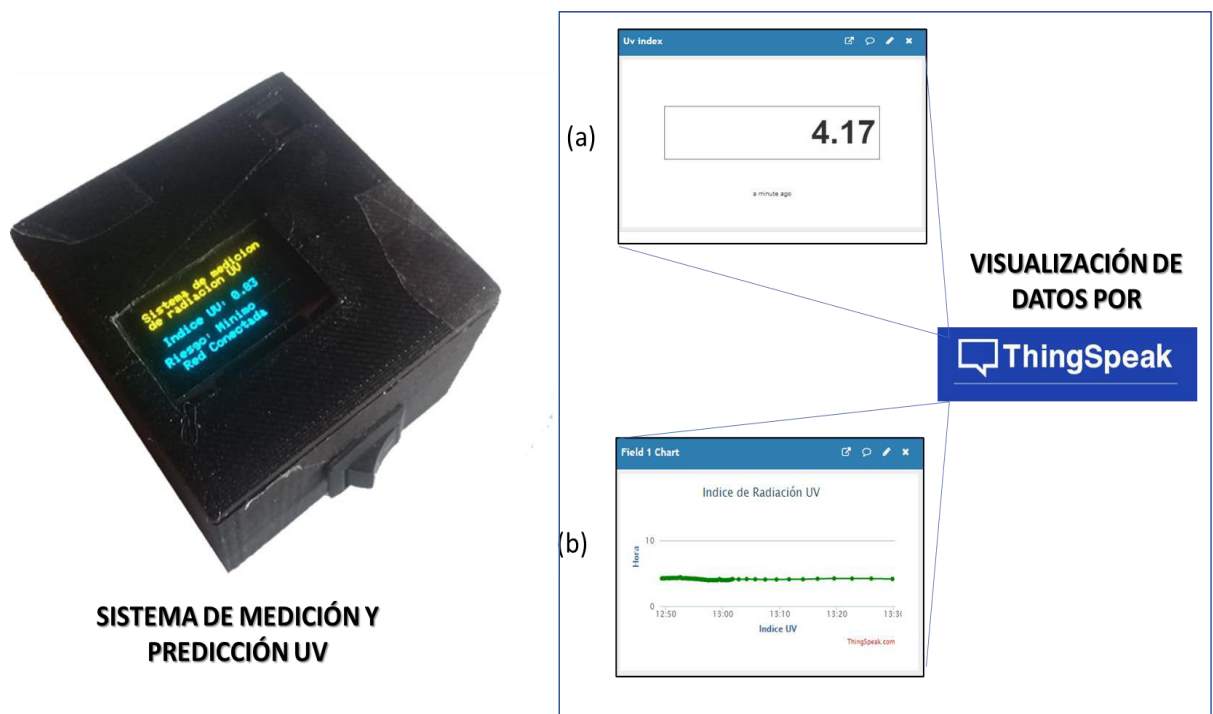


Figura 41. Visualización de datos por ThingSpeak.

En la Figura 41, se muestra el sistema físico para la medición de radiación ultravioleta y la visualización de datos que se realiza en la plataforma de ThingSpeak, en (a) se encuentra el valor numérico del valor obtenido y en (b) se encuentra la variación del índice UV en tiempo real.

3.3.7. ThingSpeak – MATLAB

En la ventana del entorno de monitorización de ThingSpeak, está disponible la aplicación "MATLAB Visualization", el cual permite crear plantillas de visualización para el trazo de graficas interactivas mediante códigos Matlab utilizando los datos recopilados por las lecturas del sistema de medición y predicción UV.

En la Figura 42, se muestra la programación realizada en *Matlab Visualization* en donde se agrega el ID del canal con el *field 1* (parámetro del sensor UV), para trabajar con los datos recolectados. Seguidamente, en la Figura 43, se visualiza la gráfica de la radiación ultravioleta con respecto al tiempo.

Apps / MATLAB Visualizations / Custom (no starter code) 1 / Edit

Name

MATLAB Code

```

1 readChannelID = 793880;
2 fieldID1 = 1
3
4 readAPIKey = 'JUS6OPL8YESNGK1L';
5
6 %% Read Data %%% Read Data %%
7 [data, time] = thingSpeakRead(readChannelID, 'Field', fieldID1, 'NumPoints', 10, 'ReadKey', re
8 %% Visualize Data %%% Visualize Data %%
9 plot(time, data);

```

Save and Run Save

Figura 42. MATLAB *Visualization* programación.

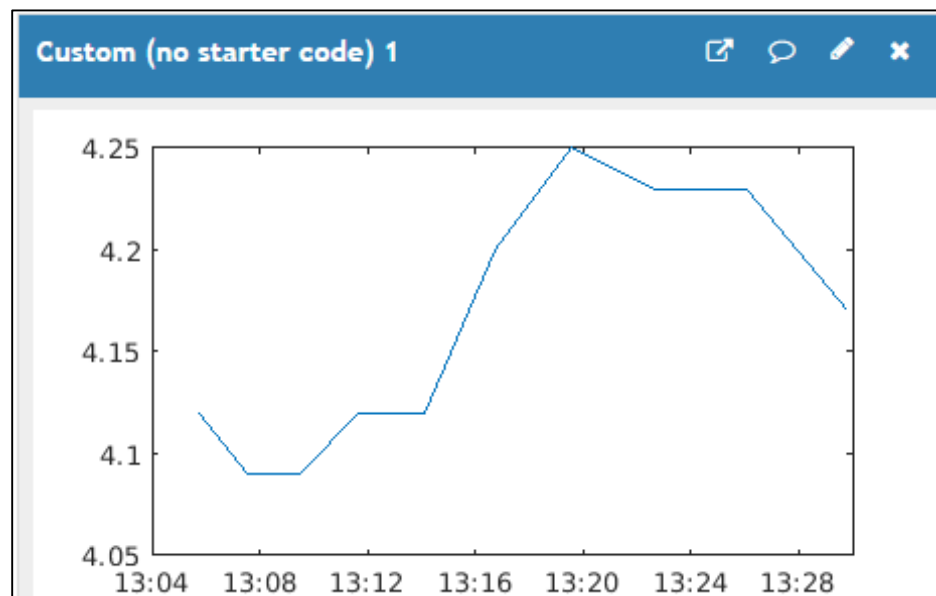


Figura 43. MATLAB *Visualization* gráfica.

De esta manera, se puede realizar el algoritmo de regresión lineal en el script del software MATLAB y se implementa en la plataforma de ThingSpeak para el pronóstico de radiación UV.

3.4. DESARROLLO DEL ALGORITMO PREDICTIVO PARA EL SISTEMA DE PRONÓSTICO DE RADIACIÓN ULTRAVIOLETA.

3.4.1. Algoritmos de Machine Learning

Los algoritmos de *Machine Learning* están comprendidos principalmente por aprendizaje supervisado y aprendizaje no supervisado, según los datos que se van estudiar.

A. Aprendizaje supervisado

En este tipo de aprendizaje se necesita una serie de datos previos, para que el sistema pueda aprender, realizar su trabajo automáticamente y resolver problemas predictivos (Inteligencia Artificial, 2018).

Existen dos clasificaciones en aprendizaje supervisado:

- Regresión (*Regression*): Compuesto por varias técnicas de algoritmo como regresión lineal, polinomial y Ridge/Lasso.
- Clasificación (*Classification*): Se subdivide en múltiples técnicas como regresión logística, árbol de decisiones, Naive Bayes, etc.

B. Aprendizaje no supervisado

A diferencia de aprendizaje supervisado, no es de importancia asignar datos previos; no obstante, se requiere de indicaciones previas para que así el sistema comprenda y analiza la información proporcionada (Inteligencia Artificial, 2018).

Existen tres clasificaciones en aprendizaje no supervisado:

- Agrupamiento (*Clustering*).
- Búsqueda de patrones (*Pattern Search*).
- Reducción de dimensiones (*Dimension Reduction*).

Solamente esta mencionado las técnicas de algoritmo tanto del aprendizaje supervisado como del aprendizaje no supervisado, de manera que se trata evitar muchas definiciones y que solamente se utiliza la técnica de algoritmo de regresión lineal durante el desarrollo de la tesis.

3.4.2. Regresión Lineal

Los modelos lineales son los métodos paramétricos más sencillos porque fácilmente puede resolver muchos problemas que no son lineales (Bonnin, 2016).

La regresión lineal es una técnica de *Machine Learning* y se usa para predicciones donde las aplicaciones son múltiples. Para ello es muy importante que los datos se ajusten en el modelo lineal (Bonnin, 2016).

La regresión lineal se adecua a un modelo de datos solamente que cumpla con una tendencia lineal en los coeficientes de un modelo que describe una relación de variables de respuesta o predicción como “y” y una variable independiente o predictora “x”, como se muestra en la siguiente ecuación (MathWorks, 2020):

$$h_{\theta}(x) = y = \theta_0 + \theta_1 * x \quad (2)$$

Donde θ_0 es la intersección con el eje de las ordenadas (y) y θ_1 es la pendiente o coeficiente de regresión.

Al trabajar con mucha cantidad de datos o conjunto de datos se forman en un sistema de ecuaciones lineales representados por matrices como se muestra (MathWorks, 2020):

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix} \quad (3)$$

A cada matriz los reemplazaremos por otras variables, para un mejor entendimiento.

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}, \theta = \begin{bmatrix} \theta_0 \\ \theta_1 \end{bmatrix} \quad (4)$$

Finalmente, la relación quedaría expresado así:

$$Y = X\theta \quad (5)$$

A. Función de costo

La función de costo es identificada como $J(\theta)$, también llamado función de error, donde el objetivo es minimizar u optimizar el error de predicción para un modelo de datos (Rodríguez, 2018). Teniendo en cuenta la función lineal de la ecuación (2), el objetivo es estimar los mejores valores para los coeficientes (θ) y así ajustar los datos de entrenamiento.

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x_{(i)}) - y_{(i)})^2 \quad (6)$$

Donde m es la longitud o el conjunto de datos, $h_{\theta}(x_{(i)})$ son los datos estimados y, $y_{(i)}$ son los datos reales obtenidos del sensor UV.

B. Método del gradiente descendente

En todo proyecto aplicando ML lo primordial es entrenar los modelos, para ello se requiere identificar los parámetros de un modelo de ML para optimizar el rendimiento del conjunto de datos a entrenar. En el campo de ML existen varios tipos de algoritmos de optimización, siendo el más conocido el método de gradiente descendente. (Rodríguez, 2018)

El método de gradiente descendente es la parte central para el aprendizaje en algoritmos de ML, además es muy aplicado en regresión lineal, permitiendo optimizar numéricamente los coeficientes de la función lineal. (Heras, 2020)

El proceso de funcionamiento para el método gradiente descendente es: (Rodríguez, 2018)

- a. Inicializar los valores para $\theta_0 = 0, \theta_1 = 0$.
- b. Indicar el coeficiente de aprendizaje (α) del algoritmo.
- c. Modificar los valores de θ_0, θ_1 para reducir la función de costo $J(\theta_0, \theta_1)$ hasta encontrar un mínimo.
- d. Actualizar el valor de θ , mediante el algoritmo de gradiente descendente mediante la ecuación (6).

$$\theta_j = \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1) \quad (6)$$

3.4.3. Desarrollo del Algoritmo de regresión lineal en MATLAB

El software MATLAB es muy usado en ingeniería para análisis, diseño y desarrollo de modelamiento numéricos muy complejos.

MATLAB dispone de un Entorno de Desarrollo Integrado (IDE) para programar textualmente mediante el lenguaje de programación llamado **M**, de acuerdo a las necesidades que se requiera. (MathWorks, 2020)

Como a modo de prueba, primero se va desarrollar el algoritmo de regresión lineal en el *script* del programa MATLAB. Una vez que el algoritmo y las pruebas hayan sido satisfactorias, se procederá a implementar dicho algoritmo en la plataforma ThingSpeak.

A. Adquirir los datos de ThingSpeak

En ThingSpeak se encuentra almacenado y se visualiza en tiempo real los datos de índice UV en función del tiempo. Entonces para realizar el pronóstico de índice UV, se empieza creando un *script* en MATLAB y se procede a recopilar los datos de índice UV desde ThingSpeak.

```
%% ===== Datos del ThingSpeak =====
readChId = 793880; % Lectura de datos del canal de ThingSpeak

%Lea los últimos 100 puntos de datos de indice UV con marcas de tiempo
[data, time] = thingSpeakRead (readChId, 'Fields' , 1, 'NumPoints' , 100);

y = data; [H,M,S] = hms(time); %devuelve los valores de hora, min y seg
x = H + M/60 + S/3600; %%pasamos el tiempo en función de hora
%x = datenum(time);
m = length(y); % número de muestras de entrenamiento

plot(x, y, 'bo');
ylabel('Indice UV'); % Etiqueta eje Y
xlabel('Tiempo'); % Etiqueta eje X
title('Meidición de indice UV');
grid on
```

Ejecutado esta parte del código, en la Figura 44 se muestra la lectura de los 100 últimos registros de índice UV en el rango de las 15 horas.

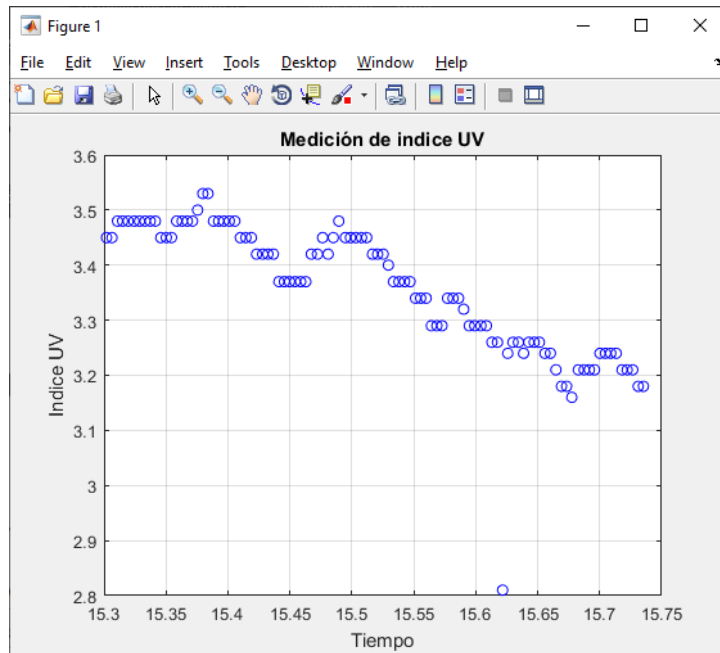


Figura 44. Gráfica de índice UV con respecto al tiempo.

B. Creación de funciones definidas por el usuario

Como en todo software que trabaje mediante un lenguaje de programación, se puede crear funciones personalizadas para estructurar ordenadamente el algoritmo.

Se crea 2 funciones definidas como la función de costo y el método de gradiente descendente.

a. Función de costo

Se procede a crear un nuevo *script* y programar la función de costo llamada **computeCost** que depende de las variables de X, y, theta.

```
function J = computeCost(X, y, theta)
%COMPUTECOST Calcula el costo para una regresión lineal
% J = COMPUTECOST(X, y, theta) calcula el costo usando theta como
% parámetro para ajustar los puntos X e y

m = length(y); % número de muestras de entrenamiento

h = (theta'*X)';
J = 1/(2*m)*sum((h-y).^2);

end
```

b. Función Gradiente descendente

En otro nuevo *script* se crea la función **gradienteDescent** que depende de las variables de X, y, theta, Alpha y el número de repeticiones.

```
function [theta, J_history] = gradientDescent(X, y, theta, alpha, num_iters)
%GRADIENDESCENT Ejecuta Gradient Descent para aprender los valores óptimos
%de theta
% theta = GRADIENDESCENT(X, y, theta, alpha, num_iters) actualiza theta
% tomando num_iters iteraciones con coeficiente de aprendizaje alpha

% Inicializa valores
m = length(y); % número de ejemplos de aprendizaje
J_history = zeros(num_iters, 1);
|
for iter = 1:num_iters

    % ===== MODIFICAR CÓDIGO =====
    % Instrucciones: Ejecuta una iteración de Gradient Descent
    %

    h = (theta'*X)';
    theta = theta - alpha/m*((h-y)'*X)';

    % =====

    % Guarda el costo J de cada iteracion
    J_history(iter) = computeCost(X, y, theta);

end
% ejex=(1:num_iters)';
% figure
% plot(ejex,J_history)

end
```

C. Procedimiento del método gradiente descendente

Regresando al primer *script*, después de la adquisición de datos desde ThingSpeak, Se muestra la parte de programación del método gradiente descendente.

```
%% ===== Parte 2: Gradient descent =====
X = [ones(m, 1),x];
theta = zeros(2, 1); % inicializa vector de parámetros

% Parámetros de Gradient Descent
iterations = 1500;
alpha = 0.01;

% Calcula y muestra costo inicial
J = computeCost(X, y, theta);

% Ejecuta Gradient Descent
theta = gradientDescent(X, y, theta, alpha, iterations);
```

```

% Muestra valores de Theta
fprintf('Theta calculado por gradient descent: ');
fprintf('%f %f \n', theta(1), theta(2));

% Dibuja el ajuste lineal encontrado
hold on;
plot(X(:,2), X*theta, 'r--');
legend('Data de entrenamiento', 'Regresión lineal')

```

Al

ejecutar el código completo el *script* principal, como resultado se visualiza en la Figura 45 la recta de regresión lineal representado por una línea de color rojo.

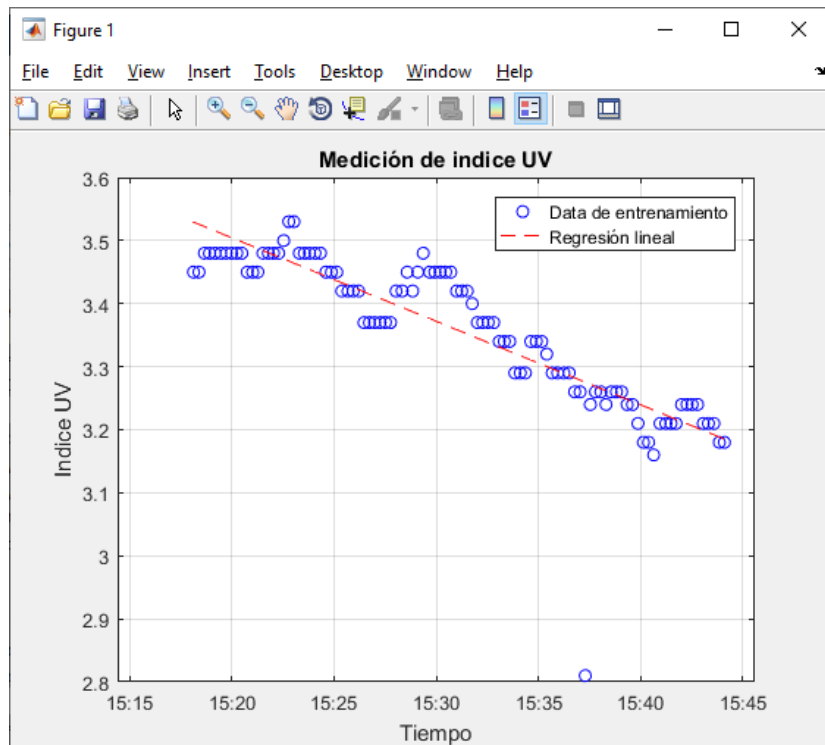


Figura 45. Gráfica del índice UV con respecto al tiempo y la función lineal.

Con respecto a los valores de θ_0 y θ_1 de la función lineal, son muy variables que dependerán de los nuevos datos de índice UV que mide el sensor.

D. Predicción UV

Para estimar un pronóstico de radiación UV con los datos que se analizan desde ThingSpeak podemos agregar algunos comandos en la programación que permita insertar la variable tiempo expresado en horas.

```
%% =====Predicción UV =====
```

```
time_prediccion = 15.0; %% Colocar la hora para pronosticar el indice UV  
predict1 = [1, time_prediccion] * theta;
```

3.4.4. Implementación de análisis en la nube

Teniendo probado el algoritmo en el entorno de software MATLAB, se procede a generar todo el código desarrollado en un nuevo script usando la aplicación “*MATLAB Visualization*” en la plataforma ThingSpeak, como se muestra en la Figura 46.

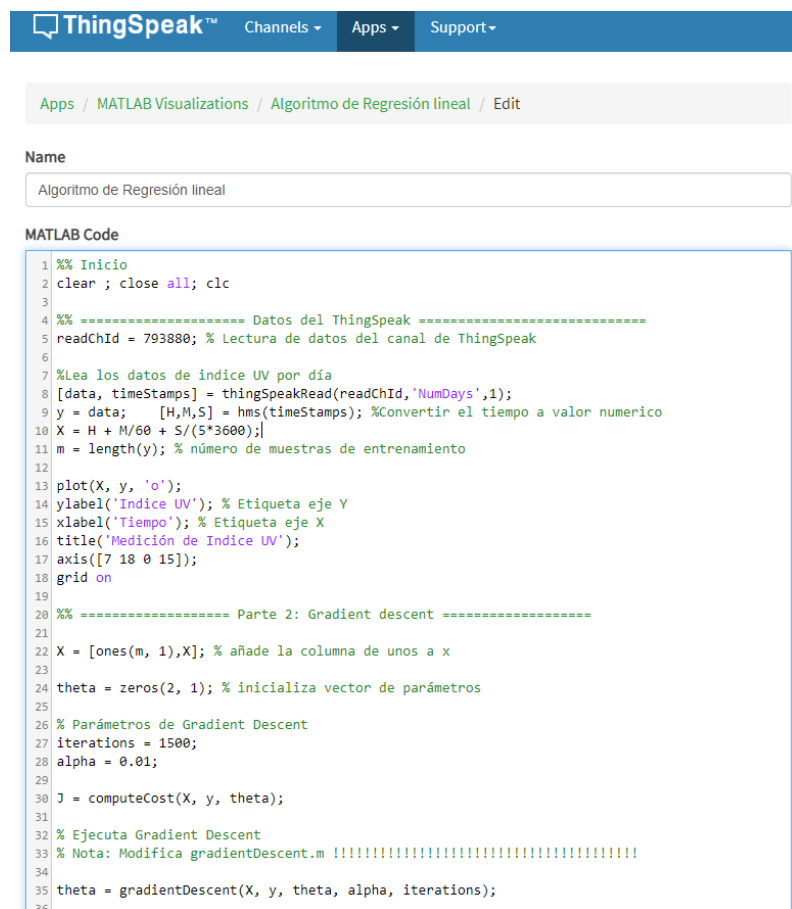


Figura 46. Programación en *MATLAB Visualization*

Para ver más a detalle sobre las variables del algoritmo de regresión lineal, se crea un nuevo canal en ThingSpeak donde se escribirá los variables de dicho algoritmo y el pronóstico.

```
%% ===== Escribir la predicción en ThingSpeak =====  
%La thingSpeakWritefunción escribe el resultado en un canal ThingSpeak  
  
writeChID = 608390;  
writeAPIKey = 'HTBT3VO3CBX95THW' ;  
thingSpeakWrite(writeChID,[theta(1),theta(2), predict1],...  
    'Fields',[1,2,3],'WriteKey',writeAPIKey);
```

En la Figura 47, se muestra el nuevo canal creado donde se visualiza un gráfico e indicadores números, donde (a) es la gráfica del índice UV con respecto al tiempo y la recta de la función lineal, en (b) y (c) son un indicador numérico que representa el valor de θ_0 y θ_1 , respectivamente, y en (d) el último indicador que entregará el pronóstico de índice UV a una hora solicitada. El valor de predicción se irá actualizando con los nuevos datos ingresado a medida que transcurre el tiempo y se acerque a hora solicitada.

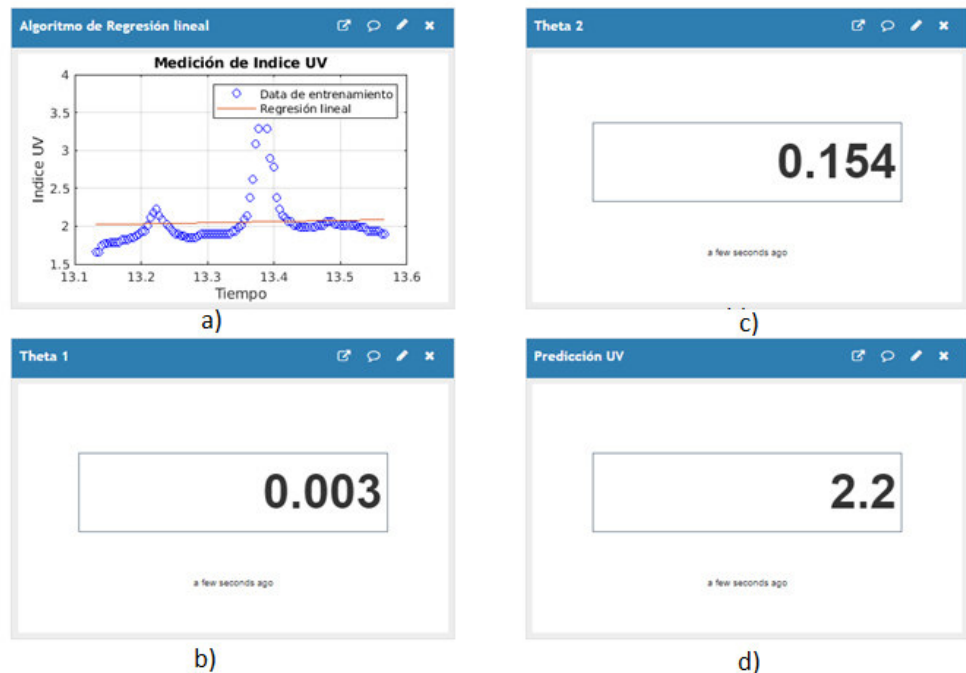


Figura 47. Canal de Pronóstico de índice UV.

3.5. VERIFICACIÓN DE LOS RESULTADOS DEL SISTEMA DE MEDICIÓN Y PRONÓSTICO DE RADIACIÓN ULTRAVIOLETA

Una vez implementado la parte electrónica, desarrollado el algoritmo de regresión lineal e implementado en la nube de ThingSpeak. Se procede a realizar las pruebas correspondientes sobre su funcionamiento.

3.5.1. Pasos para usar el dispositivo

Los pasos que se requiere para usar este dispositivo son los siguientes:

- A. Primeramente, la batería del dispositivo debe estar totalmente cargado y esto se puede verificar al enchufar con un cargador de celular debe emitir una luz en el interior del dispositivo. Si la luz es de color azul, indica que está cargado; y si la luz es color rojo, dejar cargando el dispositivo hasta terminar.
- B. El tiempo de carga para el dispositivo, depende de la cantidad de corriente que entregue la fuente de alimentación de 5v (cargador de celular), un tiempo aproximado demoraría 30 minutos.
- C. Encender manualmente el interruptor del dispositivo.
- D. El sensor UV comenzará a detectar los niveles de radiación UV, la pantalla OLED visualizará el valor del índice UV y el nivel de riesgo a que está expuesto en intervalos de 1 segundo, dependiendo del lugar donde se encuentre ya sea en un parque, vía pública, paraderos de transporte, etc.
- E. Activar el punto de acceso a Internet desde el smartphone. En la pantalla OLED mostrará un texto indicando que el dispositivo se ha conectado a la red Wi-Fi.
- F. Los datos de índice UV se registrarán en la plataforma ThingSpeak, en intervalos de 1 segundo.
- G. Durante el encendido del dispositivo, se dispone de una duración máxima de 4:40 horas.
- H. Proteger contra las caídas o golpes sobre el dispositivo.

3.5.2. Análisis de envío de paquetes de datos a ThingSpeak

Para realizar este tipo de análisis usaremos el analizador de protocolos de redes de datos **WireShark** para visualizar e identificar los detalles del flujo de paquetes de datos. Para ello, se requiere hacer el siguiente esquema como se muestra en la Figura 48.



Figura 48. Esquema de conexión a un punto de acceso

El WeMos D1 mini ESP8266 se conectará a por medio de la red Wifi a una laptop, como punto de acceso, donde este tiene conectividad a internet a través de cable Ethernet. Al tener instalado el programa WireShark en la laptop, se puede visualizar los paquetes de datos que se envía y recibe entre el WeMos D1 mini ESP8266 y el servidor IoT ThingSpeak.

No.	Time	Source	Destination	Protocol	Length	Info
5	0.109682	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
11	0.261126	54.210.227.170	192.168.137.2	HTTP	158	HTTP/1.1 200 OK (text/plain)
26	1.096883	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
30	1.232808	54.210.227.170	192.168.137.2	HTTP	158	HTTP/1.1 200 OK (text/plain)
44	2.071730	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
48	2.201115	54.210.227.170	192.168.137.2	HTTP	158	HTTP/1.1 200 OK (text/plain)
63	3.128401	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
68	3.304880	54.210.227.170	192.168.137.2	HTTP	158	HTTP/1.1 200 OK (text/plain)
81	4.160129	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
85	4.346750	54.210.227.170	192.168.137.2	HTTP	158	HTTP/1.1 200 OK (text/plain)
98	5.109182	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
102	5.290284	54.210.227.170	192.168.137.2	HTTP	162	HTTP/1.1 200 OK (text/plain)
117	6.104503	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
121	6.330461	54.210.227.170	192.168.137.2	HTTP	158	HTTP/1.1 200 OK (text/plain)
136	7.106244	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
140	7.240695	54.210.227.170	192.168.137.2	HTTP	158	HTTP/1.1 200 OK (text/plain)
153	8.131514	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
160	8.279330	54.210.227.170	192.168.137.2	HTTP	55	HTTP/1.1 200 OK (text/plain)
175	9.125559	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur
179	9.267275	54.210.227.170	192.168.137.2	HTTP	158	HTTP/1.1 200 OK (text/plain)
193	10.125799	192.168.137.2	54.210.227.170	HTTP	234	POST /update HTTP/1.1 (application/x-www-form-ur

Figura 49. Lista de tramas HTTP

En la Figura 49 se muestra el programa Wireshark asignado a un filtro de comunicación HTTP, donde se visualiza una lista de tramas HTTP entre el WeMos D1 mini ESP8266 (cliente) y ThingSpeak (servidor). El cliente y el servidor tienen asignadas las direcciones IP 192.168.137.2 y 54.210.227.170, respectivamente.

De la lista, escogeremos solamente 2 tramas (*Frames*) consecutivas, como las tramas 26 y 30. La trama 26, contiene información del origen que es el WeMos D1 mini ESP8266 y el destino es el servidor ThingSpeak, como se muestra en la Figura 50, en la parte de HTTP contiene información del método de respuesta POST, el valor de API KEY del canal del ThingSpeak que está programado en el código Arduino y además el valor del índice UV (0.80).

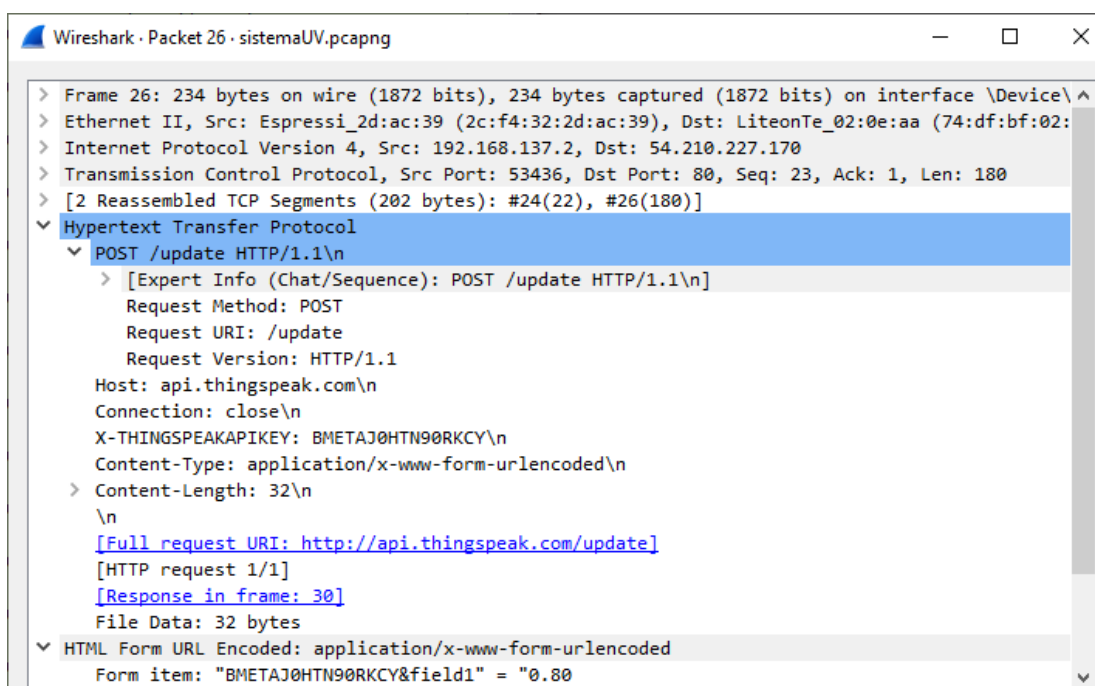


Figura 50. Trama desde el cliente al servidor.

En la Figura 51 representa la información de la trama 30 donde el origen es el servidor ThingSpeak y el destino es el WeMos D1 mini ESP8266. En la parte HTTP, contiene información como el tiempo desde la petición aproximadamente 136 ms.

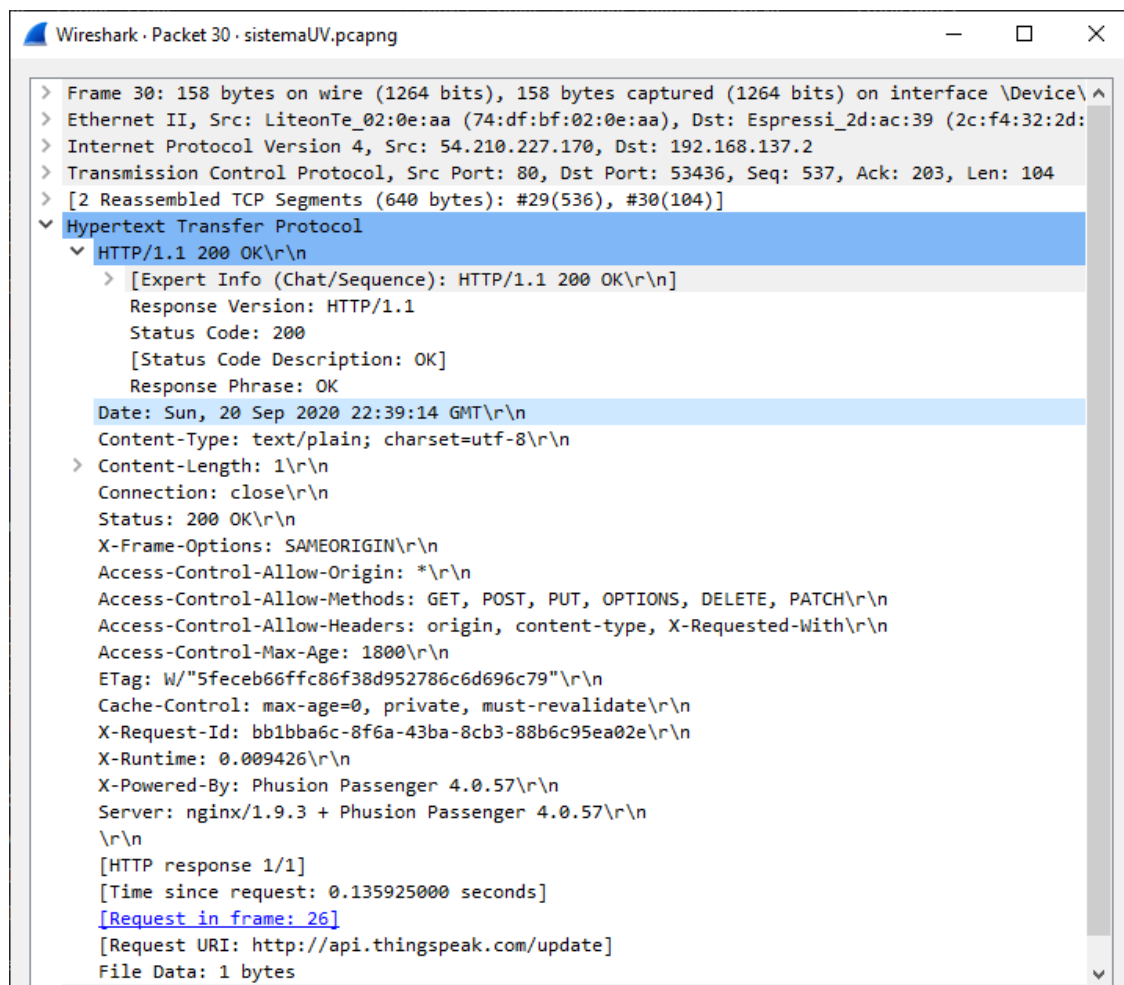


Figura 51. Trama desde el servidor al cliente.

3.5.3. Análisis de funcionamiento de medición UV

Teniendo el dispositivo operativo, se plantea realizar pruebas experimentales sobre la medición del IUV en lugares o zonas donde la población se encuentra con mayor tiempo prolongado a la exposición solar.

A. Prueba en el distrito de Comas

Comenzamos la primera prueba en el distrito de Comas entre el cruce de la Av. México y Av. Tupac Amaru (ver Figura 52), donde existe mucho comercio ambulatorio y el público que transita mucho tiempo expuesto a la radiación solar.

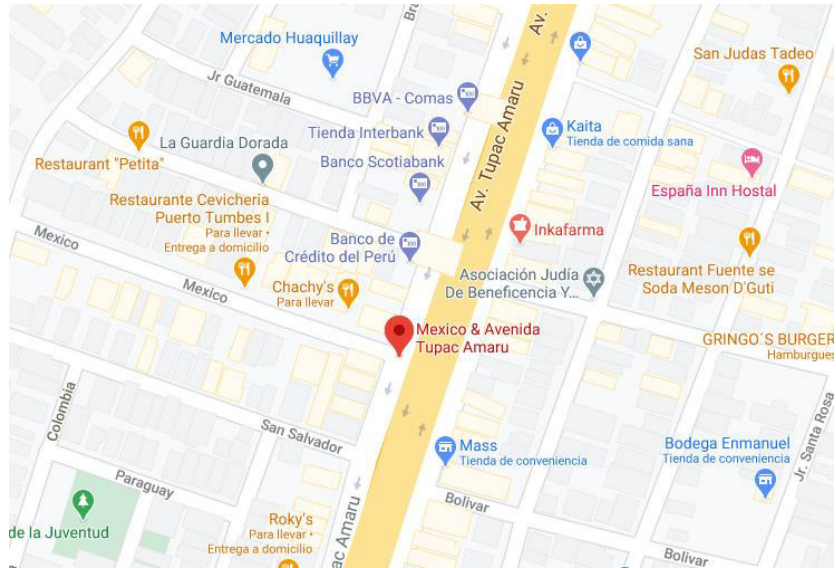


Figura 52. Ubicación de la toma de muestra 1

Fuente: (Google Maps, 2020)

En la figura 53, se visualiza el uso del sistema en el lugar establecido para realizar la medición de IUUV. La toma de muestras de la radiación UV se inicia a las 14:50 horas con una duración de 20 minutos.



(a)

(b)

Figura 53. Fotografía del lugar para la toma de datos.

Para visualizar los datos registrados en ThingSpeak, desde el dispositivo móvil se puede descargar una aplicación llamada ThingView que permite visualizar los canales de ThingSpeak de una manera más sencilla, y así ver la gráfica de los datos de IUUV con respecto al tiempo, como se muestra en la Figura 54; además

entrega como resultado el mínimo, máximo y el ultimo valor de IUV.



Figura 54. Gráfica de lectura de IUV en ThingView

En la tabla 8, se resume los resultados de la prueba de medición en un rango de 25 minutos, obteniendo una IUV promedio de 1.84, mínimo de 1.02 y máximo de 3.85.

Tabla 8. Resultados obtenidos en la prueba 1.

Indicador	Promedio	Mínimo	Máximo
IUV	1.84	1.02	3.85

B. Pruebas en el Puente Alipio Ponce

Comenzamos la primera prueba en el distrito de San Juan de Miraflores exactamente en el puente Alipio Ponce (ver Figura 55), donde existe mucho comercio ambulatorio y el público que transita mucho tiempo expuesto a la radiación solar.

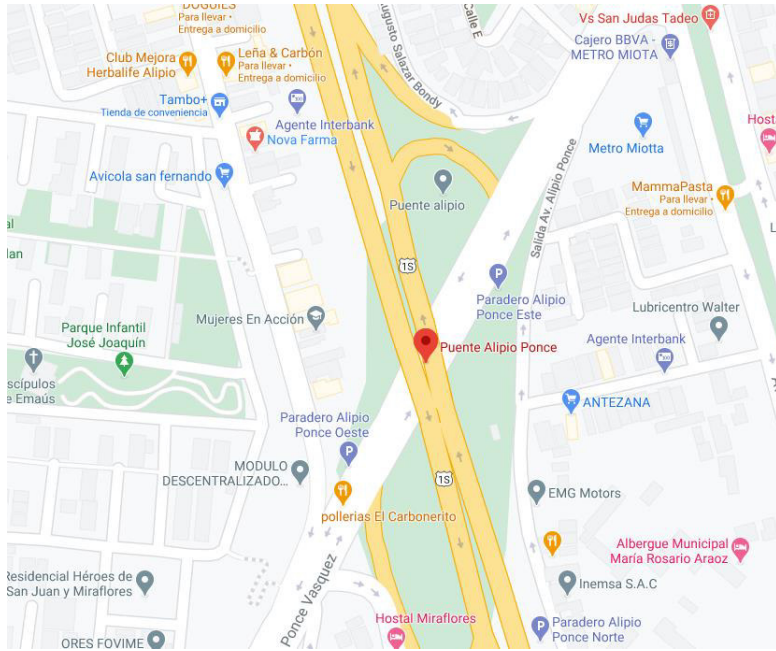


Figura 55. Ubicación de la toma de muestra 2

Fuente: (Google Maps, 2020)

En la Figura 56, se visualiza el uso del sistema en el lugar establecido para realizar la medición de IUUV. Se inicia la toma de datos en horas donde



(a)

(b)

Figura 56. Fotografía del lugar para la toma de datos.

La toma de muestra se inicia a las 14:50 horas con una duración de 20 minutos usando la aplicación ThingView como se muestra en la Figura 57; además entrega como resultado el mínimo, máximo y el ultimo valor de IUUV.



Figura 57. Gráfica de lectura de IUV en ThingView.

En la tabla 9, se resume los resultados de la prueba de medición en un rango de 25 minutos, obteniendo una IUV promedio de 1.84, mínimo de 1.02 y máximo de 3.85.

Tabla 9. Resultados obtenidos en la prueba 2.

Indicador	Promedio	Mínimo	Máximo
IUV	1.68	0.86	2.22

3.5.4. Análisis del funcionamiento del Pronostico índice UV

Una vez que el sistema de medición ha sido probado con las respectivas pruebas, procedemos analizar el pronóstico de IUV a un determinado tiempo (hora). El algoritmo de regresión lineal deberá estimar o pronosticar un valor aproximado de IUV, estando el sistema de medición encendido y tomando la lectura del sensor en tiempo real en horas del día en el mes de septiembre, en el distrito de Comas.

A. Pronosticar la radiación UV para las 13:00 horas

Para pronosticar el índice UV a la 13:00, se empezará a evaluar con media hora antes en intervalos de 5 minutos y con una muestra de los últimos 50 datos.

- a. Siendo las 12:30 horas, se visualiza el IUV de 2.76 en la Figura 58.

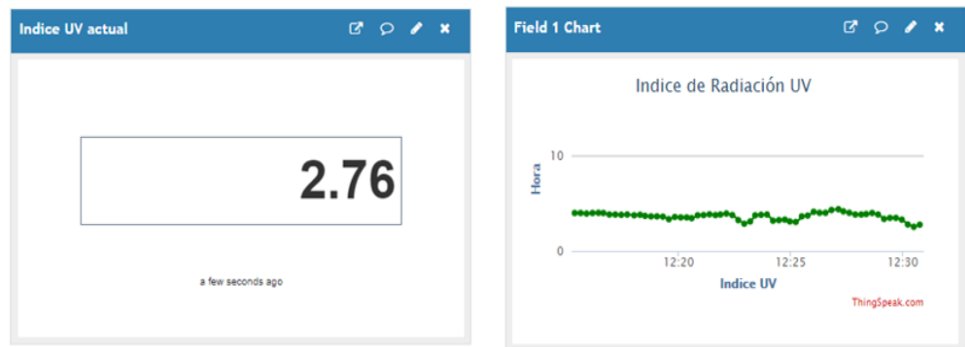


Figura 58. Visualización del índice UV a las 12:30 horas.

En la Figura 59 se visualiza el valor pronóstico para IUV de 3.8 y además los coeficientes de la función lineal siendo para $\theta_0 = 0.027$ y $\theta_1 = 0.290$.

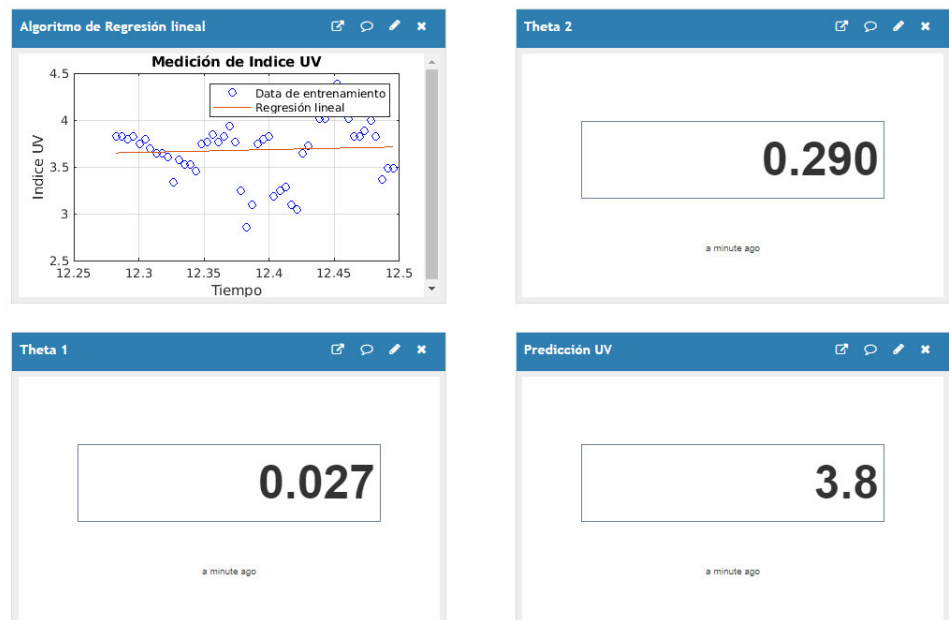


Figura 59. Visualización del pronóstico IUV siendo las 12:30 horas.

- b. Siendo las 12:35 horas, se visualiza el IUV de 4.04 en la Figura 60.

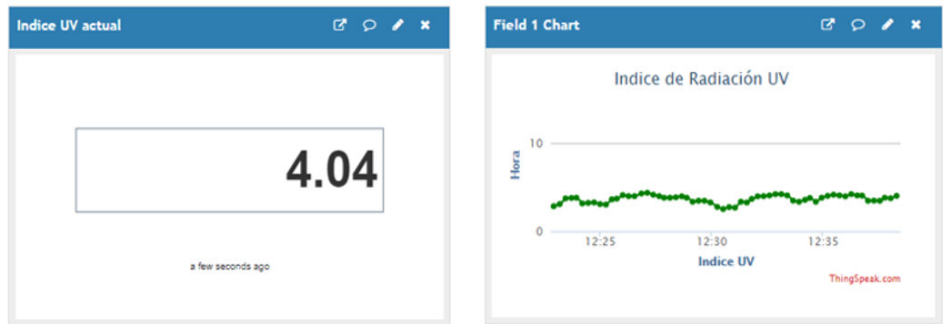


Figura 60. Visualización del índice UV a las 12:35 horas.

En la Figura 61, se visualiza el valor pronóstico para IUV de 3.8 y, además, los coeficientes de la función lineal siendo para $\theta_0 = 0.021$ y $\theta_1 = 0.294$. Manteniéndose el valor del pronóstico IUV como en el caso anterior.

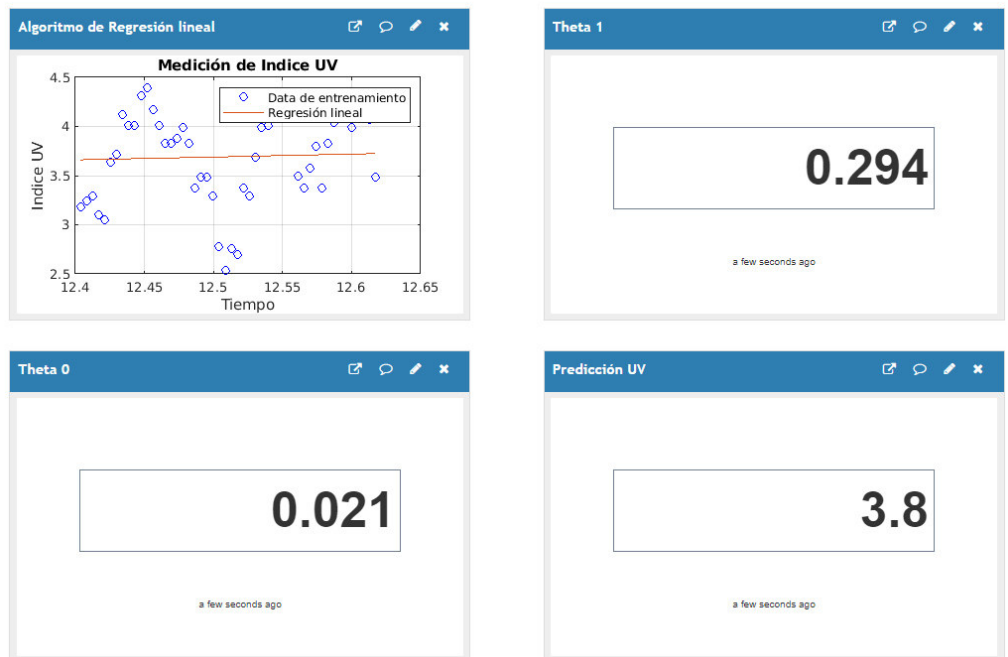


Figura 61. Visualización del pronóstico IUV siendo las 12:35 horas.

- c. Siendo las 12:40 horas, se visualiza el IUV de 4.09 en la Figura 62.

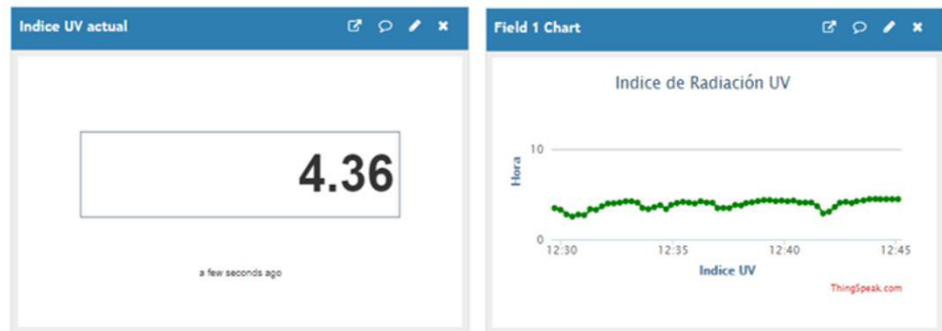


Figura 62. Visualización del índice UV a las 12:40 horas.

En la Figura 63, se visualiza el valor pronóstico para IUV de 3.9, habiendo un mínimo incremento que, en el caso anterior; y además se visualiza los coeficientes de la función lineal siendo para $\theta_0 = 0.006$ y $\theta_1 = 0.301$.

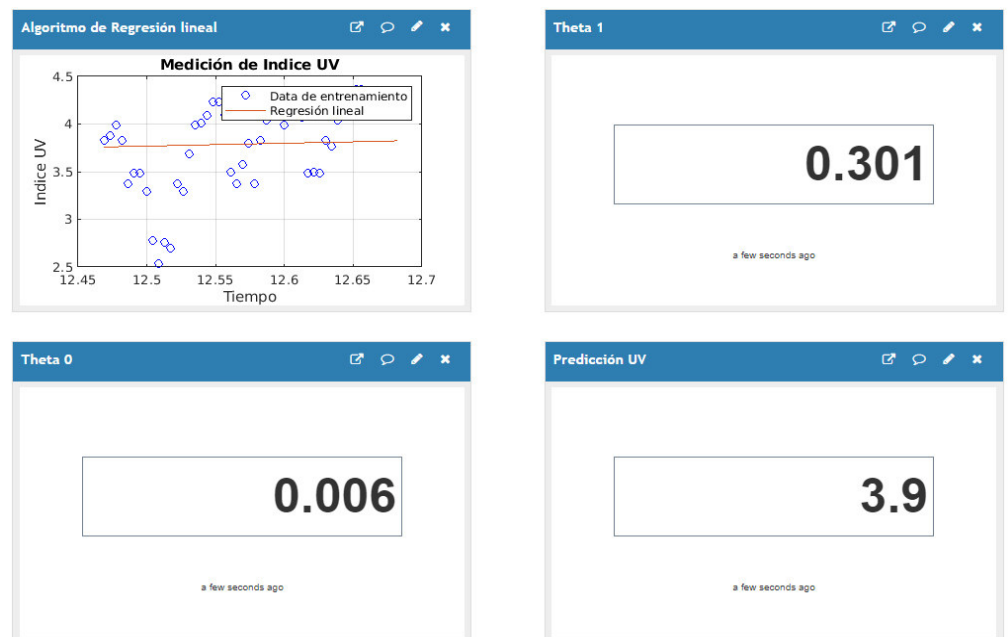


Figura 63. Visualización del pronóstico IUV siendo las 12:40 horas.

d. Siendo las 12:45 horas, se visualiza el IUV de 4.36 en la Figura 64.



Figura 64. Visualización del índice UV a las 12:45 horas.

En la Figura 65, se visualiza el valor pronóstico para IUV de 4.1, habiendo un incremento de 0.2 que, en el caso anterior; y se visualiza además los coeficientes de la función lineal siendo para $\theta_0 = 0.015$ y $\theta_1 = 0.316$.

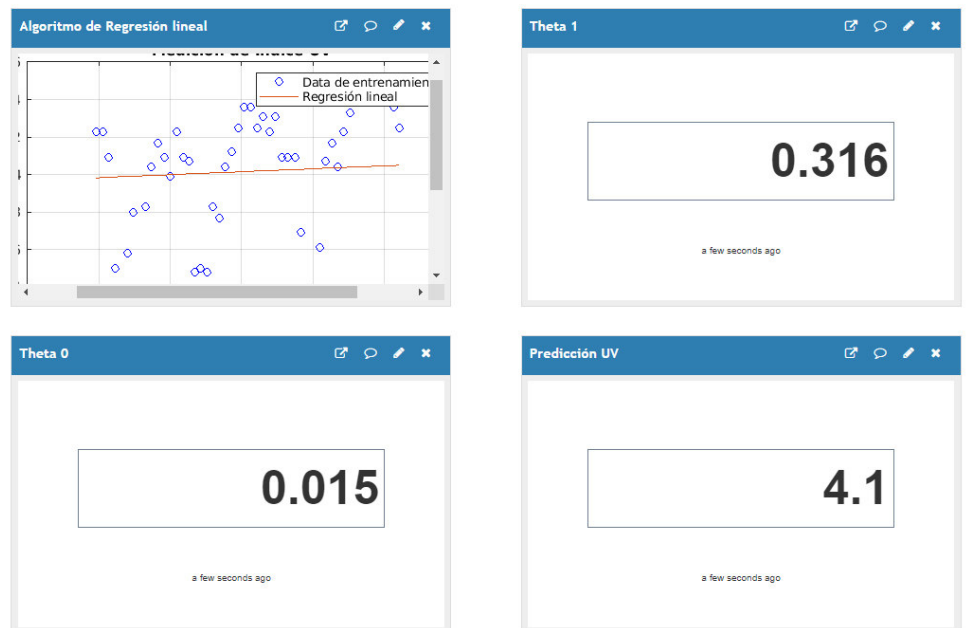


Figura 65. Visualización del pronóstico IUV siendo las 12:45 horas.

- e. Siendo las 12:50 horas, se visualiza el IUV de 2.84 en la Figura 66, el cielo se nublo por unos segundos y nuevamente el cielo se despejo.

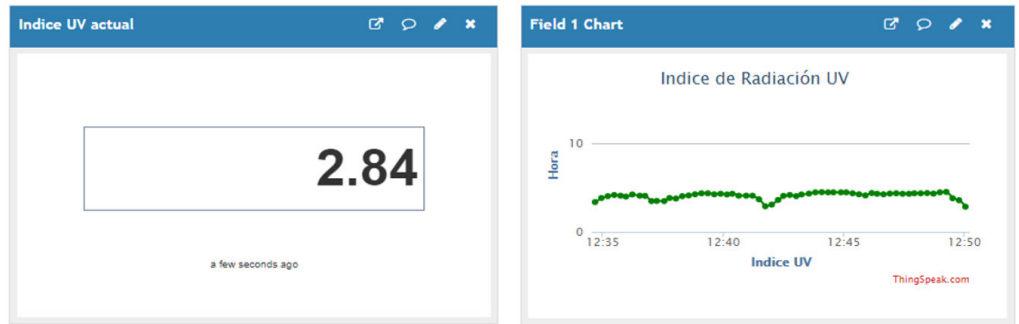


Figura 66. Visualización del índice UV a las 12:50 horas.

En la Figura 62, se visualiza el valor pronóstico para IUV de 4.1, a pesar que hubo una lectura actual más baja, el pronóstico mantiene su valor; y además visualiza los coeficientes de la función lineal siendo para $\theta_0 = 0.034$ y $\theta_1 = 0.317$.

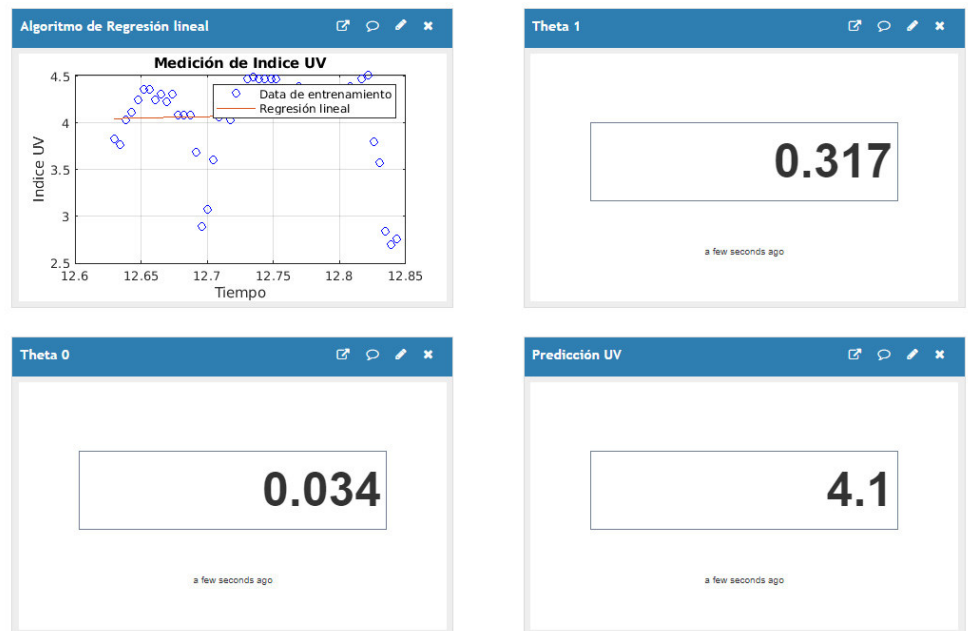


Figura 67. Visualización del pronóstico IUV siendo las 12:50 horas.

- f. Siendo las 12:55 horas, se visualiza el IUV de 4.57 en la Figura 68, un mayor incremento que en los anteriores casos.

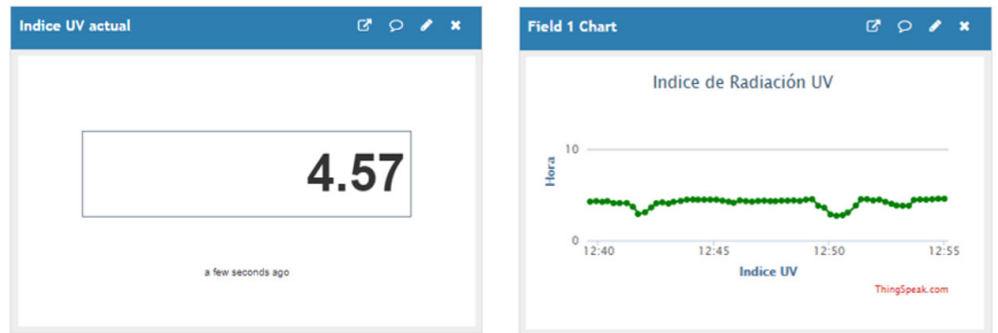


Figura 68. Visualización del índice UV a las 12:55 horas.

En la Figura 69, se visualiza el valor pronóstico para IUV de 4.3; y además visualiza los coeficientes de la función lineal siendo para $\theta_0 = 0.029$ y $\theta_1 = 0.325$.

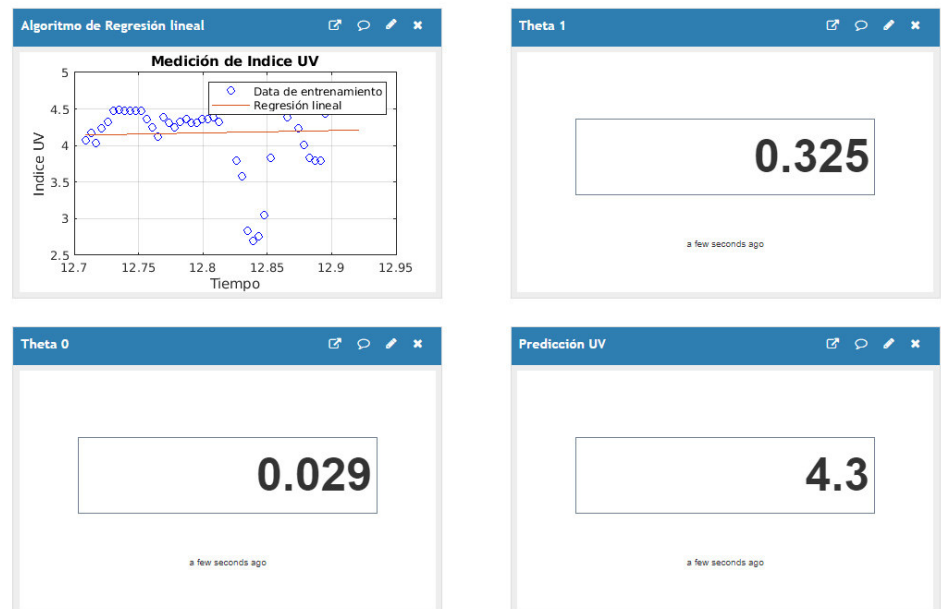


Figura 69. Visualización del pronóstico IUV siendo las 12:55 horas.

- g. Llegando a las 13:00 horas se visualiza el IUV de 4.76 en la Figura 70, y en el pronóstico de hace 5 minutos del caso anterior indicaba el IUV de 4.3.

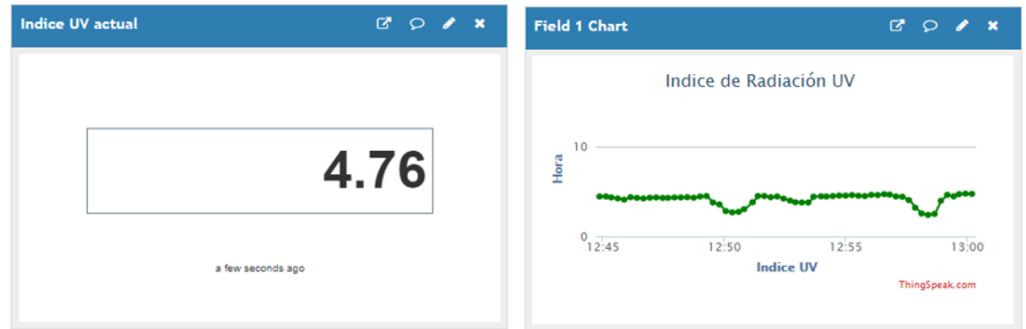


Figura 70. Visualización del índice UV a las 13:00 horas.

B. Pronosticar la radiación UV para las 13:30 horas

Para pronosticar el índice UV a la 13:30, se empezará a evaluar con media hora antes en intervalos de 5 minutos y con una muestra de los últimos 100 datos.

- a. Siendo las 13:05 horas, se visualiza el IUV de 4.87 en la Figura 71.



Figura 71. Visualización del índice UV a las 13:05 horas.

En la Figura 72 se visualiza el valor pronóstico para IUV de 4.5; y además visualiza los coeficientes de la función lineal siendo para $\theta_0 = 0.005$ y $\theta_1 = 0.330$.

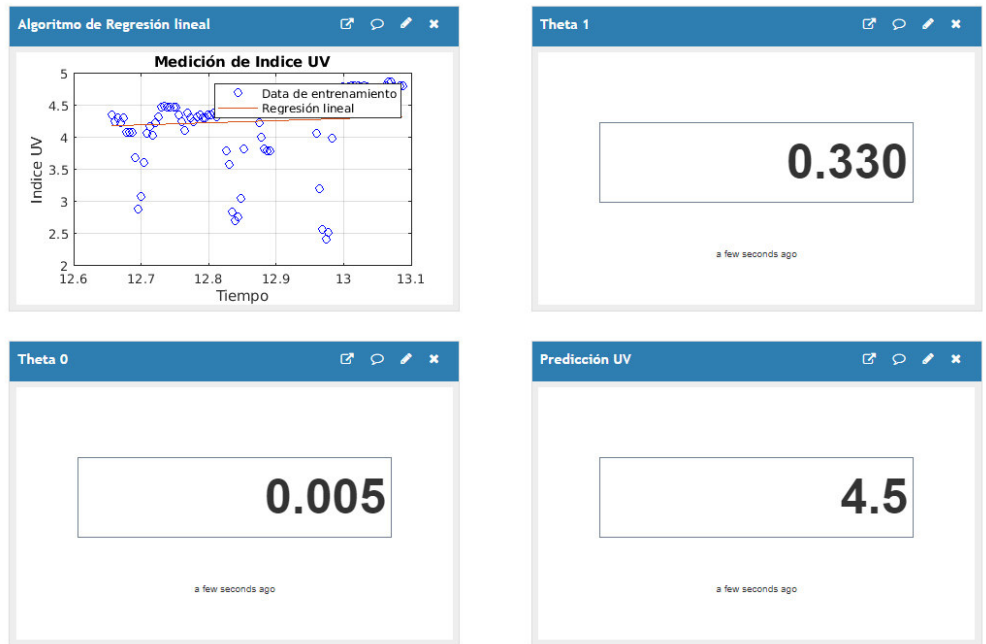


Figura 72. Visualización del pronóstico IUV siendo las 13:05 horas.

b. Siendo las 13:10 horas, se visualiza el IUV de 4.95 en la Figura 73.

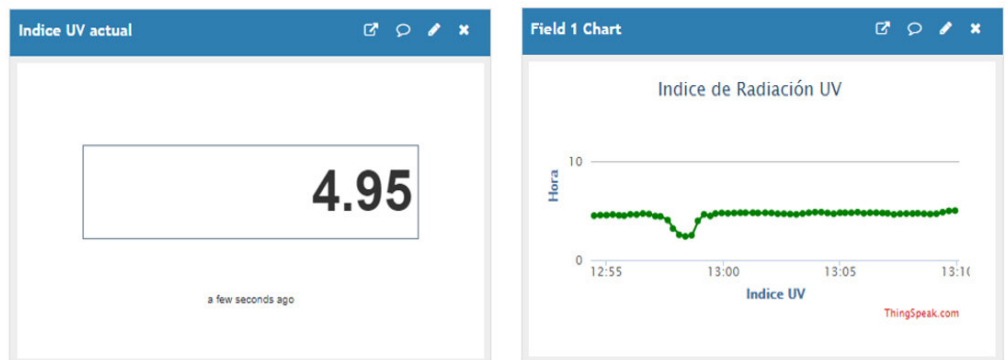


Figura 73. Visualización del índice UV a las 13:10 horas.

En la Figura 74 se visualiza el valor pronóstico para IUV de 4.6; y además visualiza los coeficientes de la función lineal siendo para $\theta_0 = -0.009$ y $\theta_1 = 0.341$.

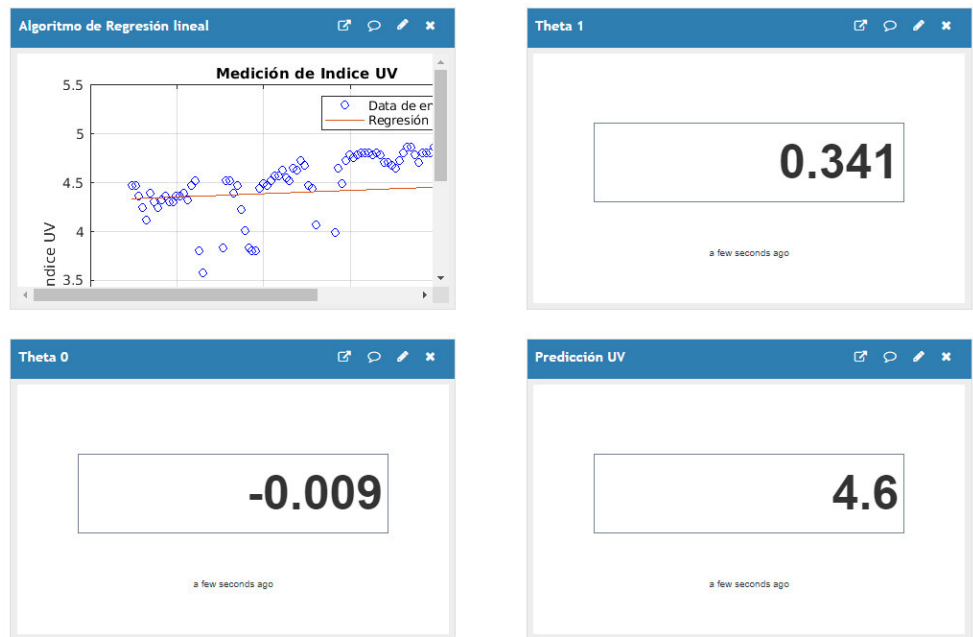


Figura 74. Visualización del pronóstico IUV siendo las 13:10 horas.

c. Siendo las 13:15 horas, se visualiza el IUV de 4.81 en la Figura 75.

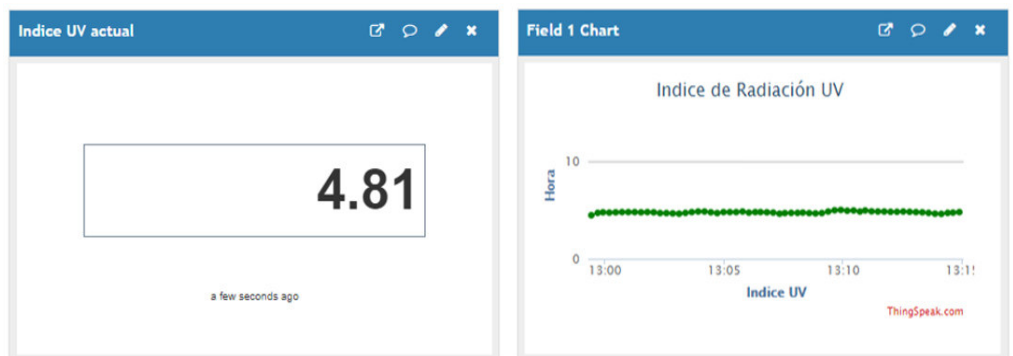


Figura 75. Visualización del índice UV a las 13:15 horas.

En la Figura 76 se visualiza el valor pronóstico para IUV de 4.6; y además visualiza los coeficientes de la función lineal siendo para $\theta_0 = -0.016$ y $\theta_1 = 0.345$.

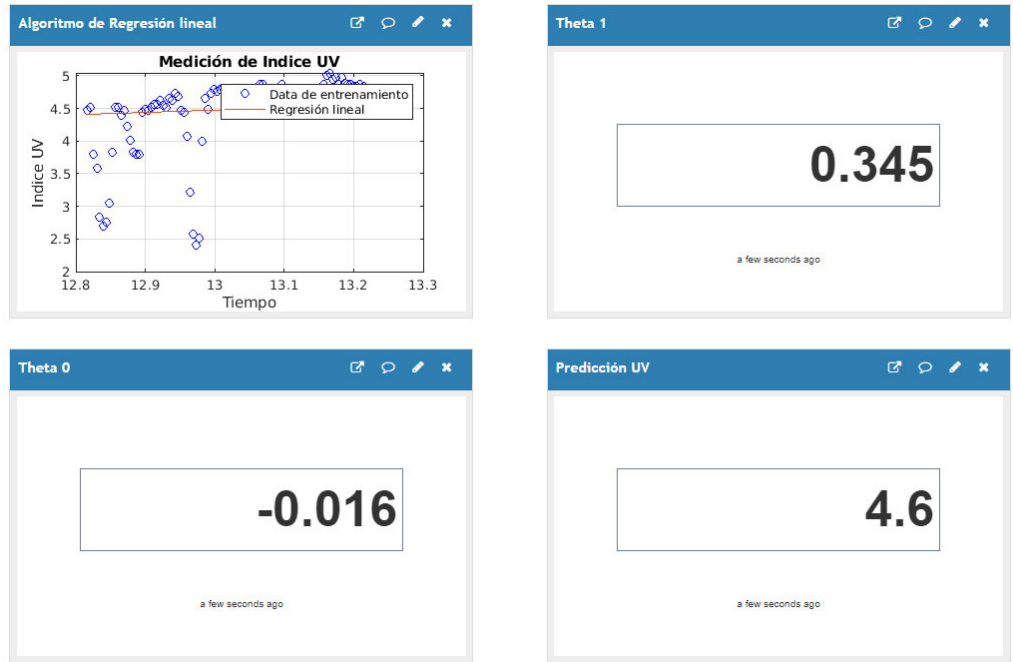


Figura 76. Visualización del pronóstico IUV siendo las 13:15 horas.

d. Siendo las 13:20 horas, se visualiza el IUV de 4.79 en la Figura 77.

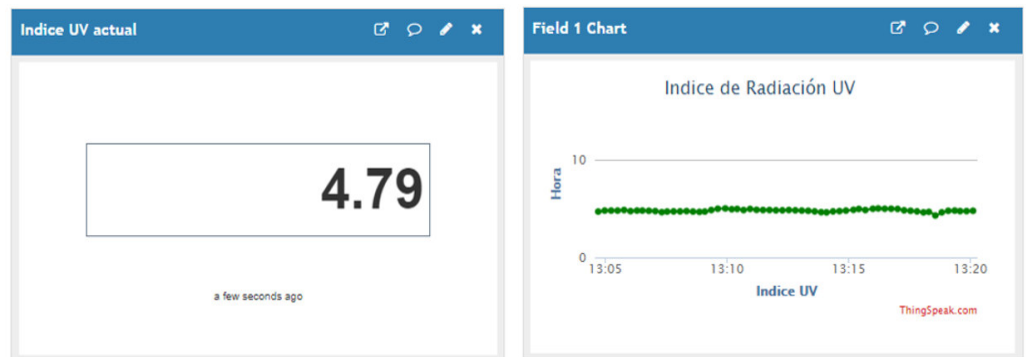


Figura 77. Visualización del índice UV a las 13:20 horas.

En la Figura 78 se visualiza el valor pronóstico para IUV de 4.8; y además visualiza los coeficientes de la función lineal siendo para $\theta_0 = -0.009$ y $\theta_1 = 0.345$.

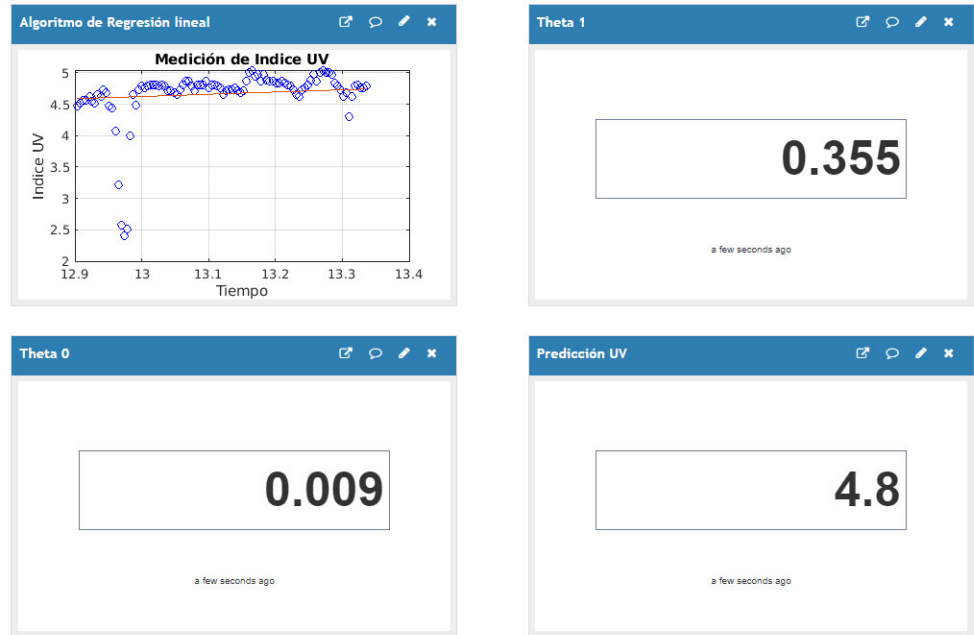


Figura 78. Visualización del pronóstico IUV siendo las 13:20 horas.

e. Siendo las 13:25 horas, se visualiza el IUV de 4.63 en la Figura 79.

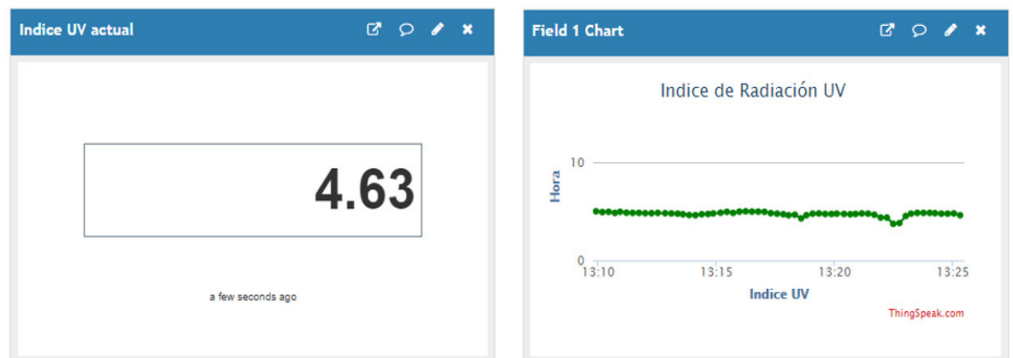


Figura 79. Visualización del índice UV a las 13:25 horas.

En la Figura 80 se visualiza el valor pronóstico para IUV de 4.8; y además visualiza los coeficientes de la función lineal siendo para $\theta_0 = -0.009$ y $\theta_1 = 0.345$.

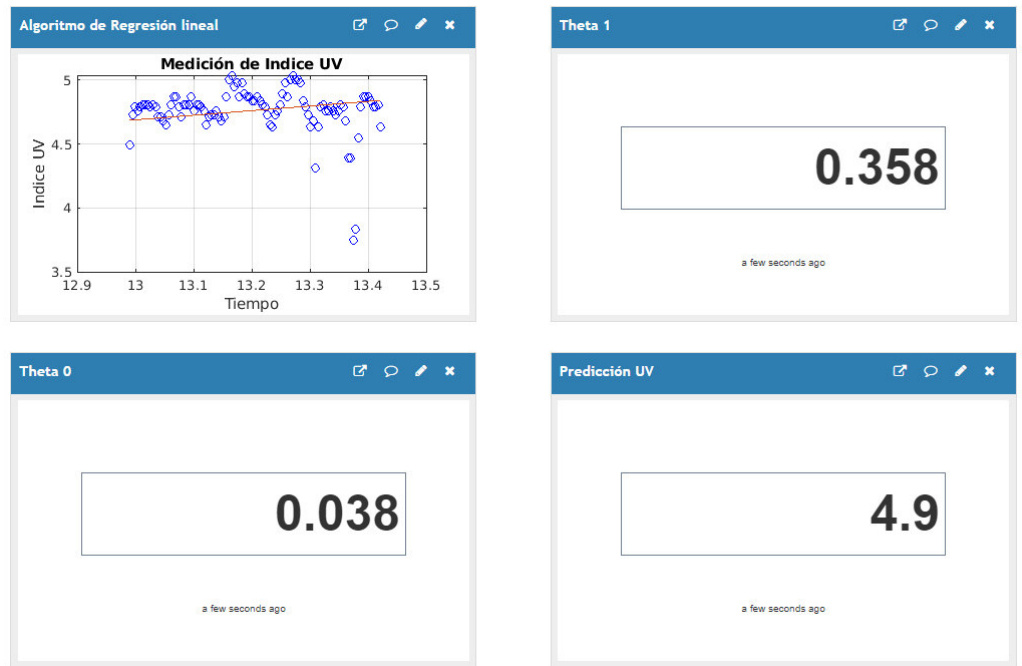


Figura 80. Visualización del pronóstico IUV siendo las 13:25 horas.

- f. Llegando a las 13:30 horas, se visualiza el IUV de 4.76 en la Figura 81; y en el pronóstico de hace 5 minutos del caso anterior indicaba el IUV de 4.9.

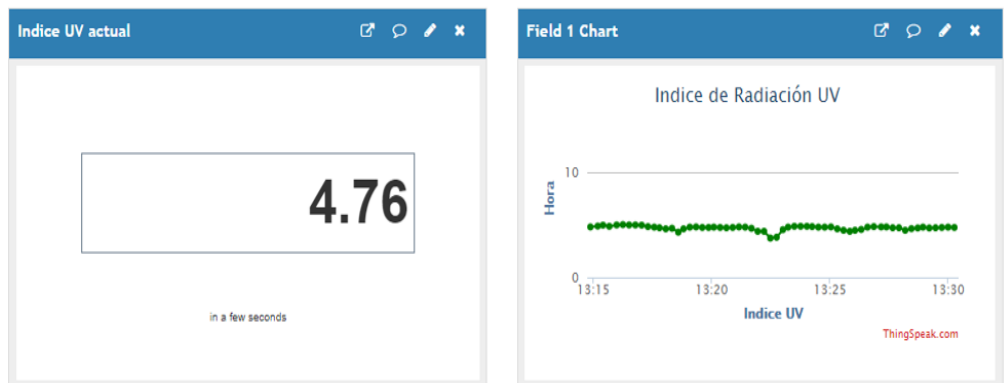


Figura 81. Visualización del índice UV a las 13:30 horas.

C. Resultados sobre las pruebas de pronóstico UV

En la tabla 10, se muestra de manera resumida la información obtenida para pronosticar la IUV cuando sea las 13:00 horas, como se vio en caso A, indicando la medición, pronóstico y el error de IUV respectivamente al tiempo.

Tabla 10. Resultados para el pronóstico del IUV para las 13:00 horas.

Hora	IUV	Pronóstico	Error_1
12:30	2.76	3.8	1.04
12:35	4.04	3.8	0.24
12:40	4.09	3.9	0.19
12:45	4.36	4.1	0.26
12:50	2.84	4.1	1.26
12:55	4.57	4.3	0.27
13:00	4.76	4.3	0.46

En la Figura 82, se puede visualizar claramente el comportamiento de la línea de pronóstico trata de mantener en un rango cercano a la línea de IUV actual, con un rango máximo de separación de 1.26.

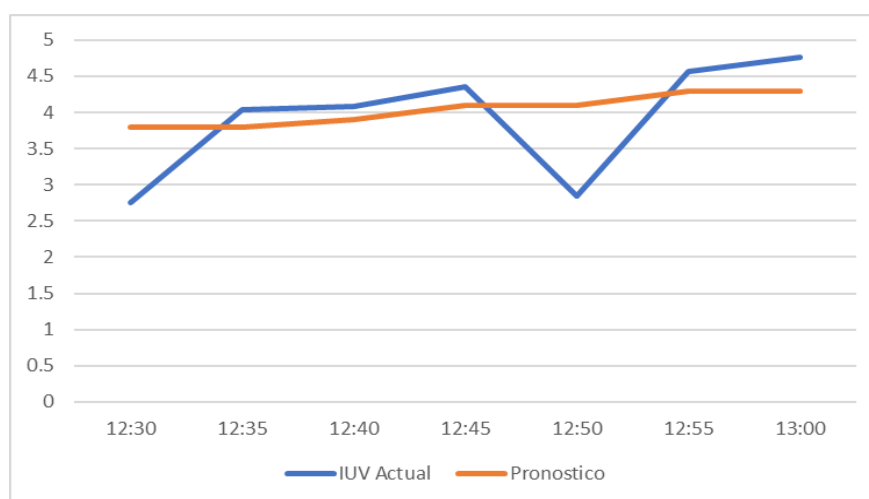


Figura 82. Gráfica comparativa entre la medición y el pronóstico de IUV para las 13 horas.

En la tabla 11, se muestra de manera resumida la información obtenida para pronosticar la IUV cuando sea las 13:30 horas, como se vio en caso B, indicando la medición, pronóstico y el error de IUV respectivamente al tiempo.

Tabla 11. Resultados para el pronóstico del IUV para las 13:30 horas.

Hora	IUV	Pronóstico	Error_2
13:05	4.87	4.5	0.37
13:10	4.95	4.6	0.35
13:15	4.81	4.6	0.21
13:20	4.79	4.8	0.01
13:25	4.63	4.9	0.27
13:30	4.76	4.8	0.04
13:05	4.87	4.5	0.37

En la Figura 83, se puede visualizar el comportamiento de la línea de pronóstico trata de acercarse lo más próximo a la línea de IUV actual, con un rango máximo de separación de 0.37.

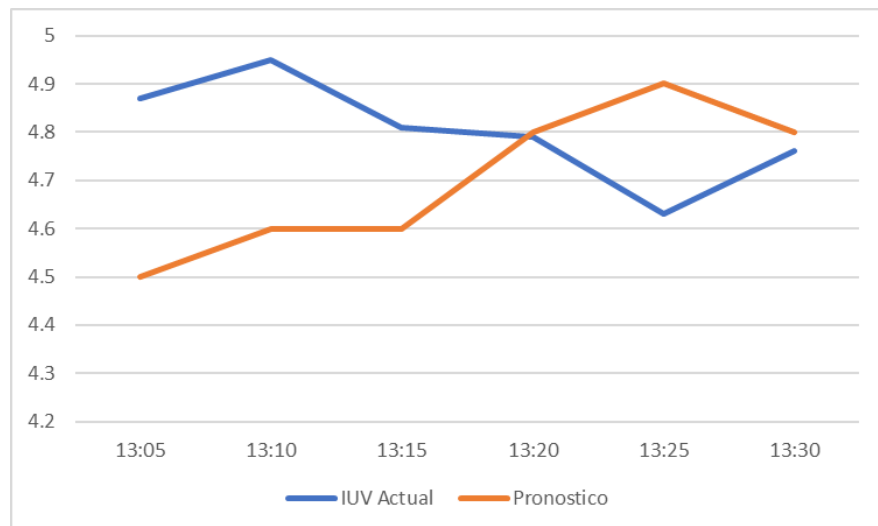


Figura 83. Gráfica comparativa entre la medición y el pronóstico de IUV para las 13:30 horas.

En la tabla 12, se muestra la primera columna del Error_1 que corresponde a los errores, diferencia entre IUV actual y el pronóstico correspondiente al tiempo, resultando un error promedio de 0.53. Del mismo modo, para la columna Error_2 resultando un error promedio 0.21, siendo este valor más preciso o más cercano a cero.

Tabla 12. Resultado y comparación de errores.

Error_1	Error_2
1.04	0.37
0.24	0.35
0.19	0.21
0.26	0.01
1.26	0.27
0.27	0.04
0.46	0.37
0.53	0.21

En el caso del valor promedio de Error_1, fue el resultado de haber entrenado con una muestra de 50 datos; en cambio, el valor promedio de Error_2 fue con una muestra de 100 datos. Por lo tanto, se puede deducir que, a mayor cantidad de muestras de datos de entrenamiento habrá un mejor resultado en el pronóstico.

CAPÍTULO IV: ANÁLISIS DE COSTO Y BENEFICIOS

4.1. ANÁLISIS DE COSTOS

4.1.1. Recursos Humanos

En la Tabla 13, se aprecia los gastos que se requirió para diseñar e implementar el sistema de medición y pronóstico UV.

Tabla 13. Recursos Humanos.

DESCRIPCIÓN	PERSONAL	MESES	PRECIO	PRECIO TOTAL
Jefe de proyecto	1	3	S/1,200.00	S/3,600.00
Ingeniero Ambiental	1	2	S/1,300.00	S/2,600.00
Programador en Arduino	1	2	S/700.00	S/1,400.00
Programador en <i>Machine Learning</i>	1	2	S/1,000.00	S/2,000.00
COSTO TOTAL EN RECURSOS HUMANOS				S/9,600.00

4.1.2. Recursos Materiales

En la Tabla 14, se muestra los recursos materiales en donde se contemplan los dispositivos y componentes electrónicos para la implementación del sistema de medición y pronóstico UV.

Tabla 14. Recursos Materiales.

N°	ACTIVIDADES	CAN.	MATERIALES/OTROS	PRECIO TOTAL
1	FASE: BÚSQUEDA Y ADQUISICIÓN DE COMPONENTES	1	WeMos D1 mini	S/ 100.00
		1	Arduino Uno R3	S/ 85.00
		1	Sensor ML8511	S/ 50.00
		1	Display Oled 0.96" 128*64	S/ 40.00
		1	Batería 3.7- 450 mAH	S/ 25.00
		1	Módulo de carga TP4056	S/ 30.00
		1	Interruptor	S/ 25.00
		1	Protoboard	S/ 25.00
2	FASE: IMPLEMENTACIÓN ELECTRÓNICA	100	Jumpers	S/ 30.00
		1	Cautil Regulable	S/ 35.00
		1	Estaño para soldar	S/ 15.00
		1	Pasta para soldar	S/ 20.00
3	FASE: ESTRUCTURA DEL CASE	1	Filamento	S/ 70.00
		1	Pegamento	S/ 10.00
		1	Impresión 3d	S/ 150.00
COSTO TOTAL EN R. MATERIALES				S/ 710.00

4.1.3. Recursos de Software

En la Tabla 15, se muestra los gastos realizados para la adquisición de programas o software requeridos para la programación de Arduino y *Machine Learning*.

Tabla 15. Recursos de Software.

DESCRIPCIÓN	CANT.	PRECIO UNITARIO	PRECIO TOTAL
Arduino IDE	1	S/0.00	S/0.00
Matlab	1	S/230.00	S/230.00
COSTO TOTAL EN SOFTWARE			S/230.00

4.1.4. Gastos Extra

En la Tabla 16, se detalla los gastos extra realizados contemplando los materiales de información y las herramientas para la implementación del sistema de medición y pronóstico UV.

Tabla 16. Gastos Extra.

MOTIVO DE GASTO	CANT.	DESCRIPCION	PRECIO
Búsqueda de información	-	Asistencia en congresos de tecnología	S/100.00
	-	Adquisición de material referente a Internet de las Cosas y <i>Machine Learning</i>	S/100.00
Herramientas	1	Computadora	S/2,500.00
Pasajes	-	Adquisición de materiales	S/75.00
	-	Muestras en espacios públicos	S/75.00
COSTO TOTAL DE GASTOS ESTRAS			S/2,850.00

Finalmente, en la Tabla 17, se muestra el gasto total realizado teniendo en cuenta los recursos humanos, materiales, software, entre otros desde el inicio hasta las pruebas finales.

Tabla 17. Gasto total del proyecto.

DESCRIPCIÓN	COSTO TOTAL
Costo de Recursos Humanos	S/9,600.00
Costo de Recursos Materiales	S/710.00
Costo de Recursos de Software	S/230.00

Costo de Gastos Extra	S/2,850.00
GASTO TOTAL DEL PROYECTO	S/13,390.00

4.2. ANÁLISIS DE BENEFICIOS

4.2.1. Beneficios tangibles

El proyecto se encuentra enfocado a la medición y pronóstico de radiación ultravioleta en los ambientes públicos, donde las personas se encuentran expuestas frecuentemente.

Se cobrará una utilidad del 20% del gasto total generado por el desarrollo del sistema, teniendo en cuenta la siguiente fórmula para el cálculo del precio de venta:

$$Precio_{venta} = Gasto_{Total} + Utilidad \quad (7)$$

Entonces:

$$Precio_{venta} = 13\,390.00 + 20\%(13\,390.00)$$

$$Precio_{venta} = 16\,068.00 \text{ soles}$$

Cálculo del precio de venta agregando el Impuesto General de Venta (IGV) del 18%:

$$Precio_{venta \text{ con IGV}} = Precio_{venta} + IGV \quad (8)$$

$$Precio_{venta \text{ con IGV}} = 16\,068.00 + 18\%(16\,068.00)$$

$$Precio_{venta \text{ con IGV}} = 18\,960.00 \text{ soles}$$

El precio de venta estimado del proyecto con Impuesto de Renta General De Ventas (IGV) es 18 960.00 soles.

4.2.2. Beneficios intangibles

Los beneficios que se obtiene para al adquirir este sistema, son:

- Identificar los niveles de riesgo al medir en el momento instantáneo el Índice de radiación UV, de ese modo tomar las medidas de precaución al momento de exponerse prolongadamente al sol.
- Se evitaría o reduciría la posibilidad de padecer alguna enfermedad causada por la radiación UV, que en casos extremos causaría la muerte.
- Capacidad de pronosticar el índice de radiación UV a una hora determinada, y actualizándose el resultado a medida que se va acercando dicha hora.

4.3. ANÁLISIS DE SENSIBILIDAD

4.3.1. Desarrollo del flujo de caja

En la Tabla 18, se visualiza el flujo de caja del primer año, realizando la venta de un producto mensual desde el primer mes proyectando hasta los primeros doce meses, contemplando el gasto de recursos y la inversión inicial.

Tabla 18. Flujo de caja del proyecto.

MES	MES 0	MES 1	MES 2	MES 3	MES 4	MES 5	MES 6	MES 7	MES 8	MES 9	MES 10	MES 11	MES 12
INGRESOS													
INGRESO CON IGV	S/0.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00	S/18,960.00
INGRESO SIN IGV	S/0.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00	S/16,068.00
VENTA DE PRODUCTOS	1	1	1	1	1	1	1	1	1	1	1	1	1
EGRESOS													
INVERSION INICIAL	S/13,390.00												
RECURSOS HUMANOS	S/9,600.00												
RECURSOS MATERIAL	S/710.00												
RECURSOS DE SOFTWARE	S/230.00												
GASTOS EXTRA	S/2,850.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00	S/3,500.00
PRODUCCION	S/0.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00	S/8,890.00
TOTAL DE EGRESO	S/13,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00	S/12,390.00
GANANCIA	S/0.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00	S/3,678.00
FLUJO DE CAJA	-S/13,006.00	-S/9,328.00	-S/5,650.00	-S/1,972.00	S/1,706.00	S/5,384.00	S/9,062.00	S/12,740.00	S/16,418.00	S/20,096.00	S/23,774.00	S/27,452.00	S/31,130.00

En la Figura 84, se muestra la gráfica del flujo de caja proyectado para los primeros doce meses.

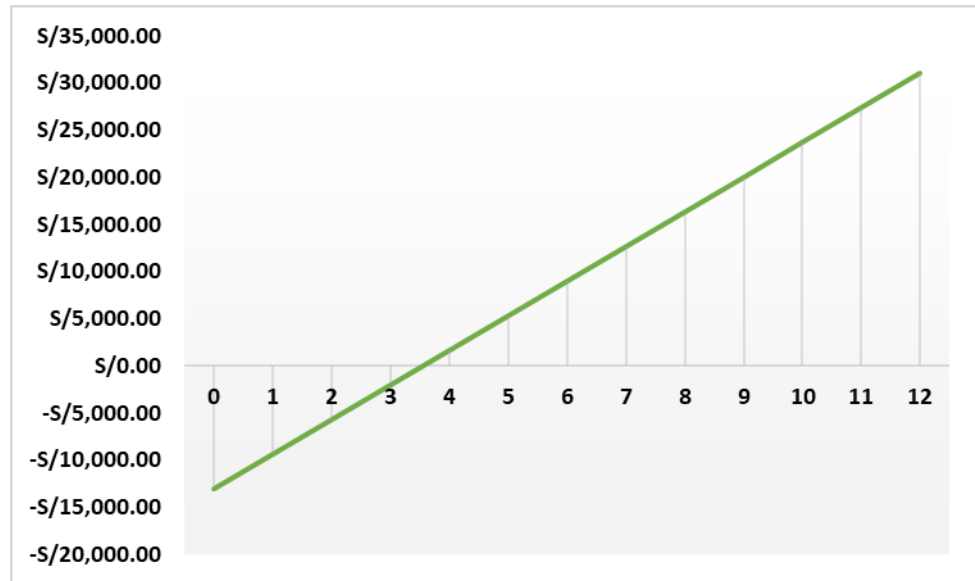


Figura 84. Flujo de caja del primer año.

4.3.2. Análisis de VAN

En esta parte, se calculará el Valor Actual Neto (VAN) del proyecto, por tal motivo se trabajará con los datos obtenidos por el flujo de caja.

Información:

- Tiempo(t)= 12 meses
- Inversión inicial (I_0) = S/ 13 390.00
- Tasa de interés(i)= 10%

La fórmula para el cálculo del análisis de VAN es:

$$VAN = I_0 \sum_{t=1}^n \frac{V_t}{(1+k)^n} \quad (9)$$

$$VAN = S/ 33 412.57$$

$$VAN = S/ 33 412.60$$

4.3.3. Análisis de TIR

En esta parte, se Tasa de Interés de Retorno (TIR) del proyecto, por tal motivo se trabajará con los datos obtenidos por la VAN.

$$VAN = \sum_{t=1}^n \frac{F_t}{(1 + TIR)^t} - I = 0 \quad (10)$$

Posteriormente, se buscará que la tasa de interés (I_0) haga que la VAN sea cero.

La fórmula para el cálculo del análisis de TIR:

$$TIR = \frac{-I + \sum_{i=1}^n F_i}{\sum_{i=1}^n i * F_i} \quad (11)$$

Información:

- Tiempo(t)= 12 meses
- Inversión inicial (I_0) = S/ 13 390.00
- Tasa de interés(i)= 10%

Resultado:

TIR = 0.21

TIR = 21%

Conclusiones

La radiación UV es un componente beneficioso para los seres vivos, en especial para el ser humano en la producción vitamina D. Una exposición prolongada de radiación UV perjudica nuestra salud, desgraciadamente cada año se reportan un incremento de casos por enfermedades de la piel principalmente, el cáncer por melanoma siendo la principal causa de muerte. Además, enfermedades oculares que en el peor de los casos pueden perder la visión.

El sistema de medición UV desarrollado presenta actualización del índice de radiación UV medido cada segundo, con un almacenamiento de datos en la nube de alta capacidad, lo cual permite que se generen alertas ante niveles de riesgo de riesgo a la salud y que los tomadores de decisión puedan generar acciones de protección de la salud de la población.

La implementación del sistema IoT resalta la alta importancia de la operación conjunta IoT – MATLAB – Cloud Computing ThingSpeak, lo que resulta en una plataforma analítica de IoT en la nube de alta capacidad y rendimiento.

El algoritmo de regresión lineal resulta ser muy aplicado en múltiples situaciones para realizar predicciones de tipo lineal y también en no lineales.

Las pruebas realizadas de medición UV resultaron exitosas, donde el valor de error para determinar el pronóstico de índice UV tiende a reducirse, a medida que se aproxima a la hora establecida y además si se agrega mayor cantidad de datos para el entrenamiento en el algoritmo de regresión lineal.

Recomendaciones

Se debe tomar las medidas preventivas para el cuidado de la salud de la piel, aplicándose bloqueadores solares, lentes con protección UV y tratar de no estar prolongadamente expuesto a la radiación UV. De ese modo, se evita posibles enfermedades de piel y oculares.

Para un producto final de uso a gran escala, el sistema electrónico puede reducirse respecto a las dimensiones físicas, realizar en una única placa impresa todo el circuito electrónico como, el microcontrolador, el módulo wifi, el sensor de radiación, pantalla oled y módulo de carga.

Si requiere mayor precisión para el cálculo del pronóstico de índice UV, es preferible que lo realice en la misma plataforma de ThingSpeak y no en el ordenador. ThingSpeak al ser una plataforma en la nube se evita el consumo excesivo de recursos en el ordenador como memoria RAM, CPU, etc.

Con la finalidad de mayor ampliación al estudio de la radiación UV y sus efectos, se debe recopilar datos de los efectos de dichos niveles en la salud y otros datos para buscar correlaciones adecuadas y poder predecir los daños con la mayor precisión.

Glosario

- **AC:** *Alternating Current*, en español Corriente Alterna es una corriente eléctrica que varía su valor en un determinado tiempo.
- **Actuador:** En electrónica, dispositivo electrónico capaz de transformar la energía eléctrica en otro tipo de energía como mecánica, térmica, etc.
- **ADC:** *Analog-to-Digital Converter*, en español Conversor Análogo Digital.
- **ADN:** Ácido desoxirribonucleico.
- **Arduino:** Plataforma de software y hardware abierto para el desarrollo programación y diseño de dispositivos electrónicos.
- **C++:** Lenguaje de programación usado en Arduino y en otras plataformas.
- **DC:** *Direct Current*, en español Corriente directa o continua es una corriente eléctrica que puede tener un valor fijo en un determinado tiempo.
- **GHz:** Giga hercio (10^9 Hz) Múltiplo de unidad de medida del Hertz y utilizado en informática para la cantidad de frecuencia en un cristal de cuarzo.
- **GPIO:** *General Purpose Input/Output*, en español Entrada/Salida de Propósito General, son pines usados en diferentes microcontroladores.
- **IoT:** *Internet of Things*, en español Internet de las cosas.
- **LED:** *Light Emitting Diode*, en español Diodo Emisor de Luz es un componente eléctrico que genera luminosidad cuando se aplica un determinado voltaje DC.
- **mA:** Mili amperio (10^{-3} A), submúltiplo de la unidad de amperio.
- **Mbps:** Megabits por Segundo usado en telecomunicaciones como unidad para velocidad de transferencia de datos.
- **nm:** Nano metro (10^{-9} m), submúltiplo de la unidad del metro.
- **Odroid xu4:** mini ordenador mucho más potente que un Raspberry Pi.

- **OMS:** Organización Mundial de la Salud, encargado en la gestión política preventiva a nivel mundial.
- **PIC:** Es un microcontrolador usado para realizar instrucciones de desarrollo de variedad de funciones y es programado en lenguaje c.
- **RAM:** *Random Access Memory*, en español Memoria de Acceso Aleatorio usado en computadoras y en hardware.
- **Raspberry Pi:** Mini ordenador para el desarrollo de proyectos con amplia gama tecnológica.
- **SENAMHI:** Servicio Nacional de Meteorología e Hidrología del Perú, especializado en otorgar información sobre el clima y pronóstico de tiempo
- **Servidor web:** Programa usado para enviar los datos a páginas web y luego en respuestas a los clientes en las computadoras.
- **TCP / IP:** **TCP** (protocolo de control de transmisión) y **IP** (Protocolo de Internet).
- **μA:** micro- amperio (10^{-6} A), submúltiplo del amperio.
- **USB:** Universal Serial Bus usado para el almacenamiento de datos.
- **UV:** Radiación ultravioleta, parte de la radiación solar
- **Wi-Fi:** Wireless Fidelity, en español fidelidad inalámbrica utilizado en redes de internet para la conectividad inalámbrica.

Referencias

- Adafruit IO (13 de junio de 2018.). *The internet of things for everyone*. Recuperado de <https://io.adafruit.com/>.
- Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M. y Ayyash, M. (2015). Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. In *IEEE Communications Surveys & Tutorials*, 17(4), 2347-2376, doi: 10.1109/COMST.2015.2444095.
- Arduino (29 de octubre del 2019). *Arduino - Software*. Recuperado de <https://www.arduino.cc/en/main/software>.
- Arias, E. (2016). *Demostrador IoT-Cloud en tiempo real* (tesis de maestría). Recuperado de http://openaccess.uoc.edu/webapps/o2/bitstream/10609/53705/7/earias_monTFM0616memoria.pdf
- Aguayo, T. (06 de julio de 2017). *Cuál es la diferencia entre rayos UVA y UVB*. Recuperado de <http://blowver.com/cual-es-la-diferencia-entre-rayos-uva-y-uvb/>.
- Bonnin, R. (2016). *Building Machine Learning Projects with TensorFlow*. Birmingham, United Kingdom.
- Cisco Networking Academy Build IT & Education (2019). *Introducción a Internet de las Cosas*. Recuperado de <https://www.netacad.com/es>
- De La Morena, B. A., Sorribas, M., Adame, J. A., Vilaplana, J. M., Gil, M., Balairón, L... Lopez, F. J. (2010). *La radiación solar: efectos en la salud y el medio ambiente*. Recuperado de http://dspace.unia.es/bitstream/handle/10334/3631/2010_radiacionsolar.pdf
- Díaz, I., Tinajero, E. J. y Escoto, M. D. (2017). *Sistema para la medición, visualización y registro de la radiación solar*. *Research in Computing Science*, 1-10.DOI: 10.13053/rcs-137-1-12.

- Doukas, C. (2012). *Building Internet of Thing with the Arduino*. Recuperado de <https://www.amazon.com/-/es/Charalampos-Doukas/dp/1470023431>
- Google Cloud (3 de agosto de 2020). *Documentation | Google Cloud*. Recuperado de <https://cloud.google.com/docs>.
- Instituto Nacional de Estadística e Informática (2018). *Estadísticas ambientales*. Recuperado de https://www.inei.gob.pe/media/MenuRecursivo/boletines/informe-de-estadisticas-ambientales_1.pdf
- Heras, J. (20 de septiembre de 2020). *Gradiente descendiente para aprendizaje automático*. Recuperado de <https://www.iartificial.net/gradiente-descendiente-para-aprendizaje-automatico/>.
- Inteligencia Artificial. (06 de julio de 2018). *Machine Learning: ¿Qué es, tipos, como funciona y donde es utilizado?* Recuperado de <https://inteligenciaartificial.io/machine-learning/>
- Iribarren D. (2016). *Modelo predictivo Machine Learning aplicado al análisis de datos climáticos capturados por una placa Sparkfun* (tesis de grado). Recuperado de <https://repositorio.comillas.edu/xmlui/handle/11531/14322>
- Lapis Semiconductor (8 de marzo de 2013). *ML8511 UV Sensor with Voltage Output*. Recuperado de https://cdn.sparkfun.com/datasheets/Sensors/LightImaging/ML8511_3-8-13.pdf
- Ley No 30102. Ley que dispone medidas preventivas contra los efectos nocivos para la salud por la exposición prolongada a la radiación solar. Publicada en *El Diario Oficial El Peruano* N° 506529, del 6 de noviembre de 2013.
- Lima Como Vamos (2018). *Evaluando la gestión en Lima y Callao VIII informe de resultados sobre calidad de vida*. Recuperado de <http://www.limacomovamos.org/cm/wp-content/uploads/2018/11/InformeGestion2017.pdf>

- Llamas, L. (4 de noviembre de 2016). *Conectar arduino a una pantalla OLED de 0.96"*. Recuperado de <https://www.luisllamas.es/conectar-arduino-a-una-pantalla-oled-de-0-96/>
- Mathworks (12 de septiembre de 2020). *MATLAB - MathWorks*. Recuperado de <https://www.mathworks.com/products/matlab.html>.
- Medina, M.R. y Mujaico, A.A. (2018). Implementation of a System for the Evaluation of Environmental Factors that Use the Internet of Things. *In 2018 IEEE Sciences and Humanities International Research Conference (SHIRCON)*, 1-4. Doi: 10.1109/SHIRCON.2018.8593015
- MedlinePlus (23 de septiembre de 2019). *Capas de la piel*. Recuperado de https://medlineplus.gov/spanish/ency/esp_imagepages/8912.htm
- Microsoft Azure (3 de septiembre de 2020). *Servicios de informática en la nube*. Recuperado de <https://azure.microsoft.com/es-es/>
- Ministerio de Salud (5 de mayo de 2020). *Solicitud de acceso a la información pública*. Recuperado de <https://www.minsa.gob.pe/portada/transparencia/solicitud/frmFormulario.asp>
- Naylamp Mechatronics (29 octubre de 2019). *Modulo ML8511 Detector UV*. Recuperado de <https://naylampmechatronics.com/sensores-luz-y-sonido/169-modulo-ml8511-detector-uv.html>
- Naylamp Mechatronics (20 de enero de 2020). *WeMos D1 mini ESP8266*. Recuperado de <https://naylampmechatronics.com/espressif-esp/291-wemos-d1-mini-esp8266.html>
- Oftalvist (7 de septiembre del 2020). *Patologías oculares producidas por la excesiva exposición al Sol*. Recuperado de <https://www.oftalvist.es/blog/patologias-oculares-por-exposicion-al-sol/>
- Organización Mundial de la Salud. (2003). *Índice UV Solar Mundial Guía Práctica*. Recuperado de

<http://apps.who.int/iris/bitstream/handle/10665/42633/9243590073.pdf;jsessionid=6CD9FEC664A0EE0A68B36149A7439DA?sequence=1>

Paredes, P. I. (2017). *Diseño de un sistema de señalización utilizando sensores fotovoltaicos para la prevención de los efectos de la radiación solar para los estudiantes de la EPIE* (tesis de grado). Recuperado de http://repositorio.unap.edu.pe/bitstream/handle/UNAP/3601/Paredes_Ccama_Patrick_Ibrahin.pdf?sequence=1&isAllowed=y

Perea, J. W. (2016). *Diseño de un sistema de monitoreo, registro y control de temperatura y humedad para un cultivo de invernadero*. (trabajo de investigación). Recuperado de <http://repositorio.utp.edu.co/dspace/handle/11059/7436>

Pérez, M. (2014). *Diseño de un radiómetro ultravioleta, para su aplicación en modelos de radiación UV* (tesis de grado). Recuperado de <http://www.ptolomeo.unam.mx:8080/xmlui/handle/132.248.52.100/3246>

PCE Instruments (2 septiembre 2020). *Medidor de radiación solar PCE-SPM 1*. Recuperado de https://www.pce-instruments.com/peru/instrumento-medida/medidor/medidor-radiacion-solar-pce-instruments-medidor-de-radiacion-solar-pce-spm-1-det_5907578.htm?_list=kat&_listpos=1

Rodríguez, F. R. (2015). *Influencia de la radiación solar en la salud de las personas en la ciudad de Moquegua, 2001-2010* (tesis de maestría). Recuperado de <http://repositorio.unjbg.edu.pe/handle/UNJBG/1031>

Rodríguez, D. (21 de diciembre de 2018). *Implementación del método descenso del gradiente en Python - Analytics Lane*. Recuperado de <https://www.analyticslane.com/2018/12/21/implementacion-del-metodo-descenso-del-gradiente-en-python/#:~:text=El%20m%C3%A9todo%20del%20descenso%20del,en%20el%20conjunto%20de%20datos.>

Servicio Nacional de Meteorología e Hidrología (2016). *Pronóstico con cobertura nacional del índice de radiación solar ultravioleta*. Recuperado de <http://repositorio.senamhi.gob.pe/handle/20.500.12542/282>

ServoVendi (2 de septiembre de 2020). *Medidor digital de Luz/Radiación ultravioleta Rayos UV/UVA/UVB (UV340B)*. Recuperado de <https://www.servovendi.com/es/medidor-digital-de-luz-radiacion-ultravioleta-rayos-uv-uva-uvb-uv340b.html>

Software y Soluciones de Analítica (10 de agosto de 2020). *Aprendizaje automático: qué es y por qué es importante*. Recuperado de https://www.sas.com/es_pe/insights/analytics/machine-learning.html

Tecsup (2018). *Introducción, conceptos fundamentales y pilares de IoT*. Recuperado de <https://tecsup.instructure.com/login/canvas>

ThingerIO (2 de septiembre de 2020). *Documentation Overview*. Recuperado de <https://docs.thinger.io/>

ThingSpeak (12 de mayo de 2020). *Learn More - ThingSpeak IoT*. Recuperado de https://thingspeak.com/pages/learn_more

Ubidots (10 de junio de 2020). *Data Drives Decisions*. Recuperado de <https://ubidots.com/>

Valiometro (5 de julio de 2020). *Luxómetro medidor de luz UV YK-35UV*. Recuperado de <https://www.valiometro.pe/luxometro-medidor-de-luz-uv-yk-35uv>

Anexos

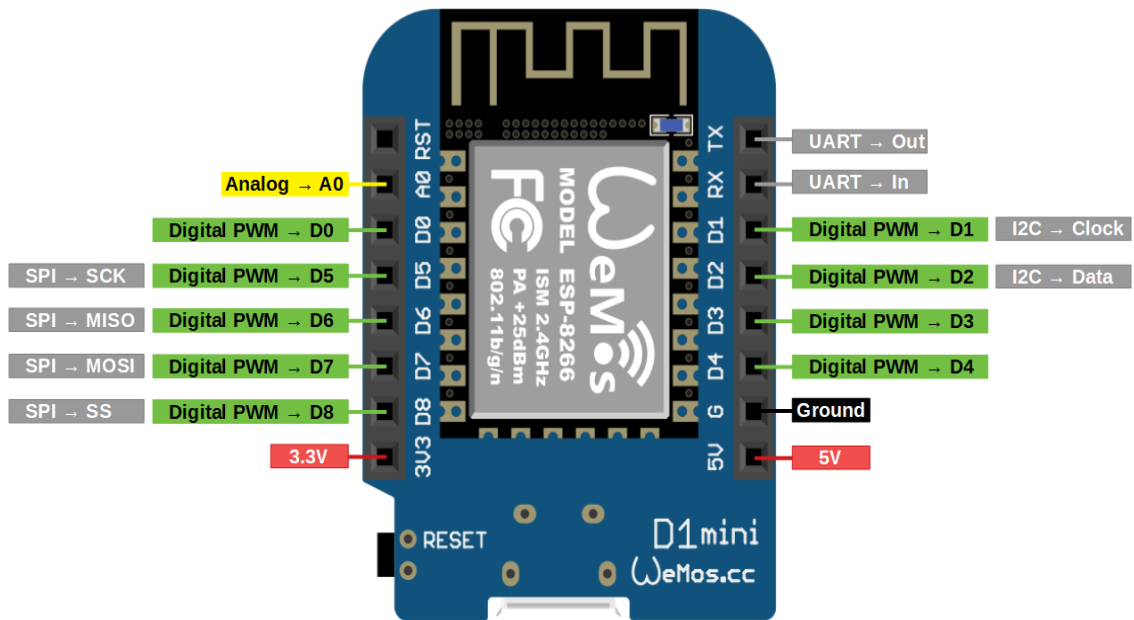
Anexo A: Hoja técnica de componentes y sensores.

A.1: Datasheet del WeMos D1 ESP 8266

1. Specification:

- Voltage:3.3V.
- Wi-Fi Direct (P2P), soft-AP.
- Current consumption: 10uA~170mA.
- Flash memory attachable: 16MB max (512K normal).
- Integrated TCP/IP protocol stack.
- Processor: Tensilica L106 32-bit.
- Processor speed: 80~160MHz.
- RAM: 32K + 80K.
- GPIOs: 17 (multiplexed with other functions).
- Analog to Digital: 1 input with 1024 step resolution.
- +19.5dBm output power in 802.11b mode
- 802.11 support: b/g/n.
- Maximum concurrent TCP connections: 5.

2. Pin Definition:



A.2: Datasheet del sensor UV ML8511

ML8511

UV Sensor with Voltage Output

GENERAL DESCRIPTION

The ML8511 is a UV sensor, which is suitable for acquiring UV intensity indoors or outdoors. The ML8511 is equipped with an internal amplifier, which converts photo-current to voltage depending on the UV intensity. This unique feature offers an easy interface to external circuits such as ADC. In the power down mode, typical standby current is 0.1 μ A, thus enabling a longer battery life.

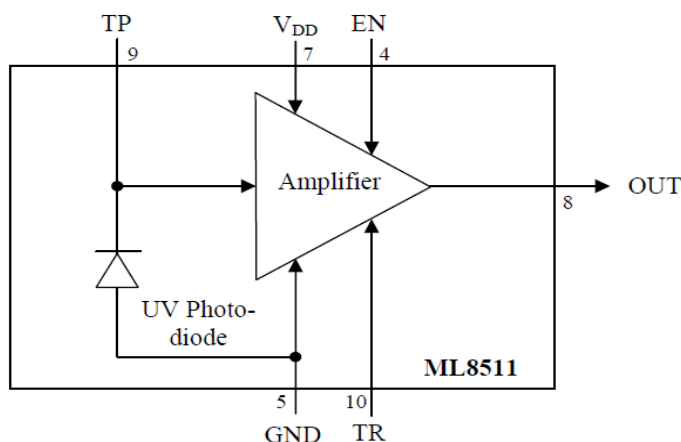
FEATURES

- Photodiode sensitive to UV-A and UV-B
- Embedded operational amplifier
- Analog voltage output
- Low supply current (300 μ A typ.) and low standby current (0.1 μ A typ.)
- Small and thin surface mount package (4.0mm x 3.7mm x 0.73mm, 12-pin ceramic QFN)

APPLICATIONS

- Smart phone, Watch, Weather station, Bicycle navigation, Accessory, Gaming

BLOCK DIAGRAM



PIN CONFIGURATIONS

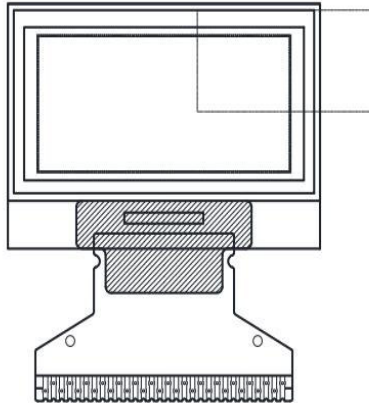
Pin	Symbol	I/O	Function
7	VDD	PW	Supply voltage. Decouple this pin to ground with 0.1 μ F capacitor.
5	GND	PW	Ground
4	EN	I	Active high enable pin. (High: Active mode, Low: Standby mode)
8	OUT	O	Output (Low in power down or standby mode)
9	TP	I/O	Test pin. Do not connect.
10	TR	I/O	Internal reference voltage. Decouple this pin to ground with 1 nF capacitor.
1,2,3, 6,11,12	NC	-	No Connection. Do not connect.

A.3: Datasheet del display OLED 0.96

OLED-128O064D-BPP3N00000

Vishay

128 x 64 Graphic OLED



FEATURES

- Type: graphic
- Display format: 128 x 64 dots
- Built-in controller: SSD1306BZ
- Duty cycle: 1/64
- +3 V power supply
- Interface: 6800, 8080, serial, and I²C
- Material categorization: for definitions of compliance please see www.vishay.com/doc?99912



RoHS
COMPLIANT

MECHANICAL DATA		
ITEM	STANDARD VALUE	UNIT
Module dimension	26.7 x 19.26 x 1.65	mm
Viewing area	23.938 x 12.058	
Active area	21.738 x 10.858	
Dot size	0.148 x 0.148	
Dot pitch	0.17 x 0.17	
Mounting hole	n/a	

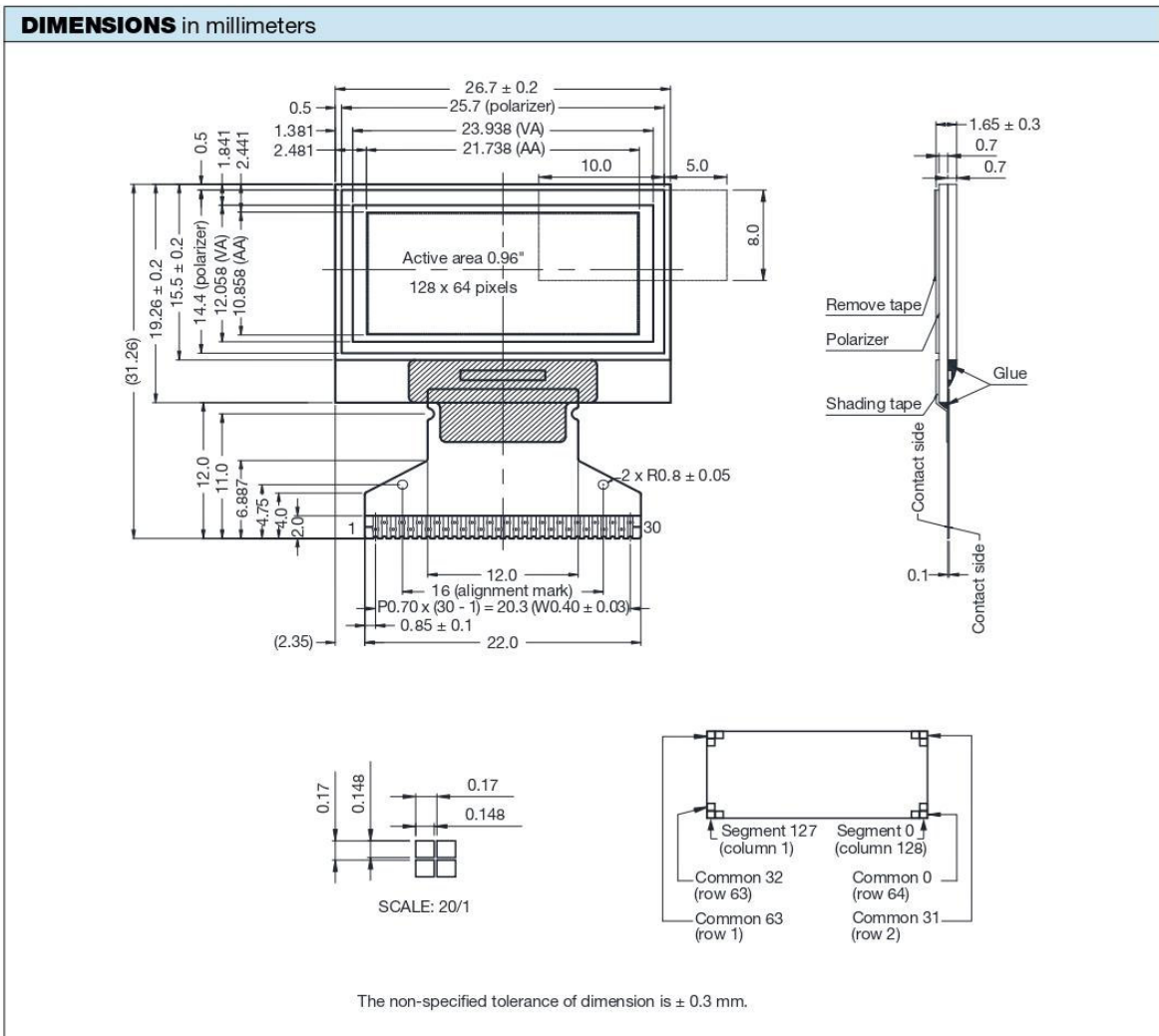
ABSOLUTE MAXIMUM RATINGS				
ITEM	SYMBOL	STANDARD VALUE		UNIT
		MIN.	MAX.	
Supply voltage for logic ⁽¹⁾⁽²⁾	V _{DD}	0	4	V
Supply voltage for display ⁽¹⁾⁽²⁾	V _{CC}	0	15	
Operating temperature	T _{OP}	-40	+80	°C
Storage temperature	T _{STG}	-40	+80	

Notes

- (1) All the above voltages are on the basis of "V_{SS} = 0 V".
- (2) When this module is used beyond the above absolute maximum ratings, permanent breakage of the module may occur. Also, for normal operations, it is desirable to use this module under the conditions according to section 6 "Electrical Characteristics". If this module is used beyond these conditions, malfunctioning of the module can occur and the reliability of the module may deteriorate.

ELECTRICAL CHARACTERISTICS						
ITEM	SYMBOL	CONDITION	STANDARD VALUE			UNIT
			MIN.	TYP.	MAX.	
Supply voltage for logic	V _{DD}	-	2.8	3.0	3.3	V
Supply voltage for display	V _{CC}	-	10	12	15	
Input high voltage	V _{IH}	-	0.8 V _{DD}	-	V _{DDI/O}	
Input low voltage	V _{IL}	-	0	-	0.2 V _{DD}	
Output high voltage	V _{OH}	-	0.9 V _{DD}	-	V _{DDI/O}	
Output low voltage	V _{OL}	-	0	-	0.1 V _{DD}	
50 % check board operating current	I _{CC}	V _{CC} = 12 V	9	10	12	mA

OPTIONS				
EMITTING COLOR				
YELLOW	GREEN	RED	BLUE	WHITE
-	-	-	Yes	-



Anexo B: Comparación de medición UV entre el sensor ML8511 y el radiómetro UV-BIOMETER

En un trabajo de investigación, realizaron el alto nivel de precisión de los sensores UV ML8511. Para ello, hicieron una prueba comparativa con un equipo profesional, UV- BIOMETER, de medición de radiación UV. (Díaz, Tinajero y Escoto, 2017).



Figura: Radiómetro UV-BIOMETER

Fuente: Díaz, Tinajero y Escoto.

En la siguiente Figura, se visualiza una gráfica comparativa entre el radiómetro UV-BIOMETER y el sensor UV ML8511, donde se puede ver claramente la alta semejanza en las mediciones.

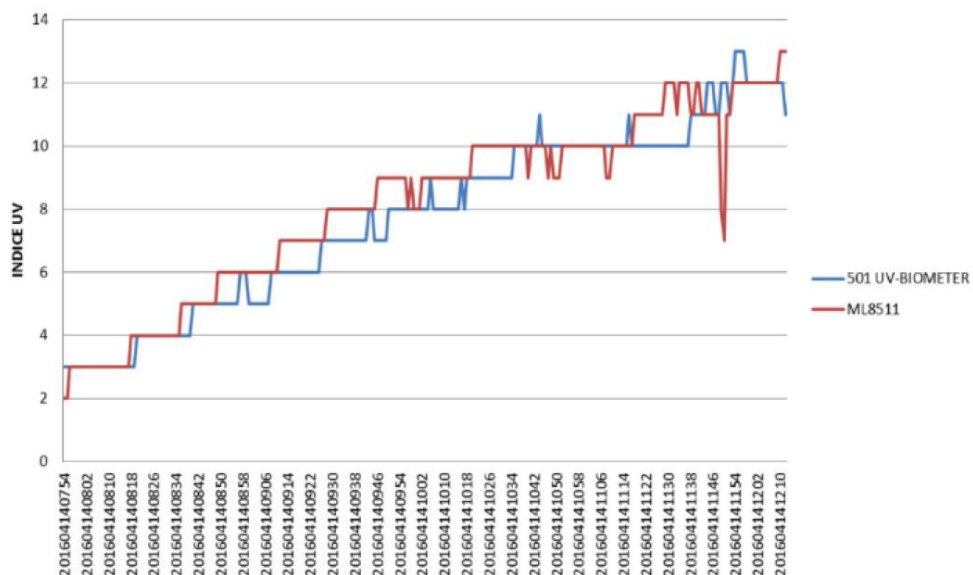


Figura: Comparación de mediciones entre el UV-BIOMETER y el sensor UV8511

Fuente: Díaz, Tinajero y Escoto.

Anexo C: Programación en Arduino IDE

```
1 #include <ESP8266WiFi.h>
2 #include <Wire.h> // libreria para bus I2C
3 #include <Adafruit_GFX.h> // libreria para pantallas graficas
4 #include <Adafruit_SSD1306.h> // libreria para controlador SSD1306
5
6 String apiWritekey = "*****"; // replace with your THINGSPEAK WF
7
8 const char ssid [] = "*****"; // your wifi SSID name
9 const char password [] = "*****"; // wifi pasword
10
11 unsigned long previousMillis = 0;
12 const long interval = 1000;
13
14 const char* server = "api.thingspeak.com";
15
16 WiFiClient client;
17
18 int UVOUT = A0; //Output from the sensor
19
20 #define ANCHO 128 // reemplaza ocurrencia de ANCHO por 128
21 #define ALTO 64 // reemplaza ocurrencia de ALTO por 64
22
23 #define OLED_RESET D8 // necesario por la libreria pero no usado (
24 //mejor revisar el GPIO /Wemos D8
25 Adafruit_SSD1306 oled(ANCHO, ALTO, &Wire, OLED_RESET); // crea objeto
26
27 void setup()
28 {
29 Serial.begin(115200);
30 Wire.begin(); // inicializa bus I2C
31 oled.begin(SSD1306_SWITCHCAPVCC, 0x3C); // inicializa pantalla con di
32
33 WiFi.disconnect();
34 delay(10);
35 WiFi.begin(ssid, password);
36
37 Serial.println();
38 Serial.println();
39 Serial.print("Connecting to ");
40 Serial.println(ssid);
41
42 WiFi.begin(ssid, password);
43
44 if (WiFi.status() != WL_CONNECTED) {
45 delay(500);
46 Serial.print(".");
```

```

47     }
48     Serial.println("");
49     Serial.print("WeMos D1 connected to wifi...");
50     Serial.println(ssid);
51     Serial.println();
52     pinMode(UVOUT, INPUT);
53     pinMode(D0, OUTPUT); //para GND del sensor UV
54     pinMode(D5, OUTPUT); //para VCC del sensor UV
55     digitalWrite(D0, LOW);
56     digitalWrite(D5, HIGH);
57 }
58
59 void loop()
60 {
61     digitalWrite(D0, LOW);
62     digitalWrite(D5, HIGH);
63     unsigned long currentMillis = millis();
64     if (currentMillis - previousMillis >= interval) {
65         previousMillis = currentMillis;
66         int uvLevel = averageAnalogRead(UVOUT);
67
68         //Use the 3.3V power pin as a reference to get a very accurate output
69         float outputVoltage = (3.3 / 1023.0) * uvLevel;
70
71         float uvIntensity = mapfloat(outputVoltage, 0.99, 2.8, 0.0, 15.0);
72         oled.clearDisplay();
73         oled.setTextColor(WHITE); // establece color al unico disponibl
74         oled.setCursor(0, 0); // ubica cursor en inicio de coordenada
75         oled.setTextSize(1); // establece tamano de texto en 1
76         oled.print(" Sistema de medicion"); // escribe en pantalla el t
77         oled.print(" de radiacion UV ");
78         oled.setCursor(8, 25); // ubica cursor en coordenadas 10,30
79         oled.print("Indice UV: ");
80         oled.print(uvIntensity);
81
82
83
84         if (uvIntensity >= 0 && uvIntensity <=2){ //1-2
85             oled.setCursor(0, 40);
86             oled.print("Riesgo: Minimo");
87         }
88         if (uvIntensity > 2 && uvIntensity <=5){
89             oled.setCursor(0, 40);
90             oled.print("Riesgo: Bajo");
91         }

```

```

92▢ if (uvIntensity > 6 && uvIntensity <=8){
93   oled.setCursor (0, 40);
94   oled.print("Riesgo: Moderado");
95   }
96▢ if (uvIntensity > 8 && uvIntensity <=11){
97   oled.setCursor (0, 40);
98   oled.print("Riesgo: Alto");
99   }
100▢ if (uvIntensity >11 && uvIntensity <=14){
101   oled.setCursor (0, 40);
102   oled.print("Riesgo: Muy alto");
103   }
104▢ if (uvIntensity > 14){
105   oled.setCursor (0, 40);
106   oled.print("Riesgo: Extremo");
107   }
108   Serial.print(" ML8511 output: ");
109   Serial.print(uvLevel);
110
111   Serial.print(" / ML8511 voltage: ");
112   Serial.print(outputVoltage);
113
114   Serial.print(" / UV Intensity (mW/cm^2): ");
115   Serial.println(uvIntensity);
116
117   //////////////////////////////////////
118   if (client.connect(server,80)
119▢ {
120     String tsData = apiWritekey;
121     tsData += "&field1=";
122     tsData += String(uvIntensity);
123     tsData += "\r\n\r\n";
124
125     client.print("POST /update HTTP/1.1\n");
126     client.print("Host: api.thingspeak.com\n");
127     client.print("Connection: close\n");
128     client.print("X-THINGSPEAKAPIKEY: "+apiWritekey+"\n");
129     client.print("Content-Type: application/x-www-form-urlencoded\n");
130     client.print("Content-Length: ");
131     client.print(tsData.length());
132     client.print("\n\n"); // the 2 carriage returns indicate closing
133     client.print(tsData);
134
135     oled.setCursor (10,50);
136     oled.print("Red Conectada");

```

```

137
138     Serial.println("uploaded to Thingspeak server...");
139 }
140 client.stop();
141
142 Serial.println("Waiting to upload next reading...");
143 Serial.println();
144 oled.display();    // muestra en pantalla todo lo establecido anteriormente
145 }
146
147 }
148
149 //Takes an average of readings on a given pin
150 //Returns the average
151 int averageAnalogRead(int pinToRead)
152 {
153     byte numberOfReadings = 8;
154     unsigned int runningValue = 0;
155
156     for(int x = 0 ; x < numberOfReadings ; x++)
157         runningValue += analogRead(pinToRead);
158     runningValue /= numberOfReadings;
159
160     return(runningValue);
161 }
162
163 //The Arduino Map function but for floats
164 //From: http://forum.arduino.cc/index.php?topic=3922.0
165 float mapfloat(float x, float in_min, float in_max, float out_min, float out_max)
166 {
167     return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;
168 }

```


Anexo D: Programación en MATLAB a ThingSpeak

```
1 %% Inicio
2 clear ; close all; clc
3
4 %% ===== Datos del ThingSpeak =====
5 readChId = *****; % Lectura de datos del canal de ThingSpeak
6
7
8 %Lea los últimos 100 puntos de datos de índice UV con marcas de tiempo
9 [data, time] = thingSpeakRead (readChId, 'Fields' , 1, 'NumPoints' , 100);
10
11 y = data; [H,M,S] = hms(time); %devuelve los valores de hora, minuto y segundo
12
13 X = H + M/60 + S/3600; %pasamos el tiempo en función de hora
14 m = length(y); % número de muestras de entrenamiento
15
16 plot(X, y, 'bo');
17 ylabel('Índice UV'); % Etiqueta eje Y
18 xlabel('Tiempo'); % Etiqueta eje X
19 title('Medición de Índice UV');
20 grid on
21
22 %% ===== Parte 2: Gradient descent =====
23
24 X = [ones(m, 1),X];
25
26 theta = zeros(2, 1); % inicializa vector de parámetros
27
28 % Parámetros de Gradient Descent
29 iterations = 1500;
30 alpha = 0.01;
31
32 J = computeCost(X, y, theta);
33
34 % Ejecuta Gradient Descent
35
36 theta = gradientDescent(X, y, theta, alpha, iterations);
37
38 % Muestra valores de Theta
39 fprintf('Theta calculado por gradient descent: ');
40 fprintf('%f %f \n', theta(1), theta(2));
41
42 % Dibuja el ajuste lineal encontrado
43 hold on;
44 plot(X(:,2), X*theta, '-')
45 %legend('Data de entrenamiento', 'Regresión lineal');
46
47
48 %%=====Agregar el tiempo que desee predecir el índice UV=====
49
50 time_predicction = 13.5; %% 13:00 hrs.
51 predict1 = [1, time_predicction] * theta;
52
53 %%=====Parte 4: Escribir la predicción en ThingSpeak=====
54 %La thingSpeakwritefunción escribe el resultado en un canal ThingSpeak
55
56 writeChID = *****;
57 writeAPIKey = '*****' ;
58 thingSpeakWrite(writeChID,[theta(1),theta(2), predict1], 'Fields',[1,2,3], 'WriteKey',writeAPIK
59
```

```

60
61 %%=====Funciona definidas por el usuario=====
62 % Calcula y muestra costo inicial :
63     function J = computeCost(X, y, theta)
64     %COMPUTECOST Calcula el costo para una regresión lineal
65     % J = COMPUTECOST(X, y, theta) calcula el costo usando theta como
66     % parámetro para ajustar los puntos X e y
67
68     m = length(y); % número de muestras de entrenamiento
69
70     % ===== MODIFICAR CÓDIGO =====
71     % Instrucciones: Calcular el costo para un set de parámetros "theta".
72
73     h = (theta'*X)';
74     J = 1/(2*m)*sum((h-y).^2);
75
76     % =====
77     end
78
79
80 function [theta, J_history] = gradientDescent(X, y, theta, alpha, num_iters)
81 %GRADIENDESCENT Ejecuta Gradient Descent para aprender los valores óptimos
82 %de theta
83 % theta = GRADIENDESCENT(X, y, theta, alpha, num_iters) actualiza theta
84 % tomando num_iters iteraciones con coeficiente de aprendizaje alpha
85
86 % Inicializa valores
87 m = length(y); % número de ejemplos de aprendizaje
88 J_history = zeros(num_iters, 1);
89
90 for iter = 1:num_iters
91
92     % ===== MODIFICAR CÓDIGO =====
93     % Instrucciones: Ejecuta una iteración de Gradient Descent
94     %
95
96     h = (theta'*X)';
97     theta = theta - alpha/m*((h-y)'*X)';
98
99     % =====
100
101     % Guarda el costo J de cada iteracion
102     J_history(iter) = computeCost(X, y, theta);
103
104 end
105 % ejex=(1:num_iters)';
106 % figure
107 % plot(ejex,J_history)
108
109 end

```

Anexo E: Estadística de casos por enfermedades de radiación UV en Perú

En la Figura se muestra el número de pacientes que padecen enfermedades producidas por radiación UV por etapas de vida, en los últimos 5 años a nivel nacional (Ministerio de Salud, 2020). En el año 2015 se registró un total de 6452 de pacientes, incrementándose a 6817 en el año siguiente; en el 2017 se registró descendió a 5199, para el año 2018 ascendió nuevamente a 6373 y en 2019 incremento bruscamente a 8549 casos. Donde se registró elevados números de casos en adultos mayores, por delante de los adultos a partir de 30 años.

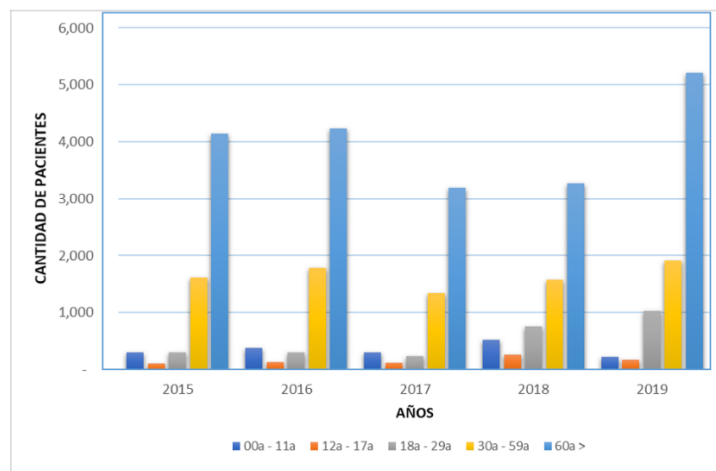


Figura: Cantidad de pacientes por etapas de vida.

Fuente: Ministerio de Salud.

En la siguiente Figura, se visualiza el porcentaje de pacientes que padecen cáncer de piel de Melanoma, a nivel nacional, en el año 2019 (Ministerio de Salud, 2020). Donde se registró un mayor porcentaje en pacientes adultos mayores, segundo lugar están los adultos a partir de 30 años y en tercer lugar los jóvenes a partir de 18 años. Recordar que el melanoma es un cáncer de piel maligno que puede ocasionar la muerte, sino sigue un tratamiento y los cuidados adecuados.

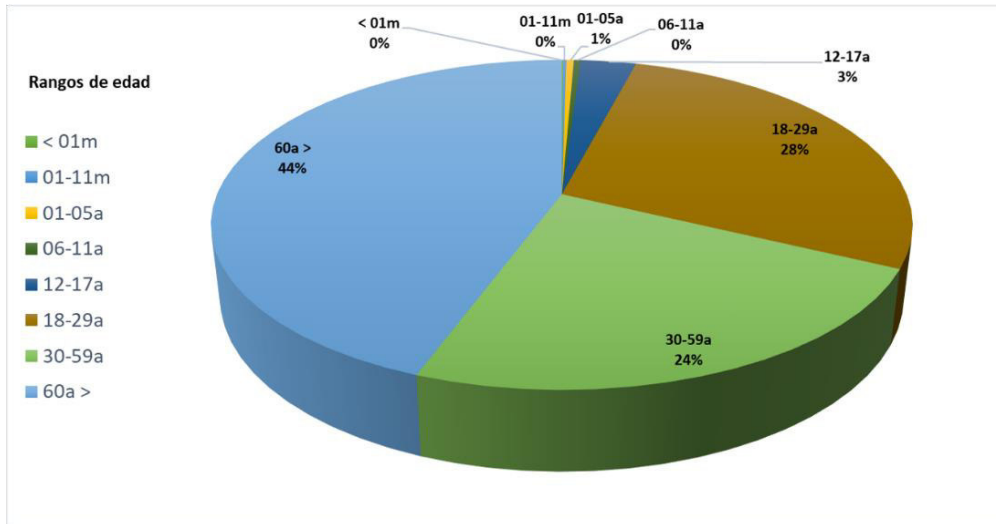


Figura: Porcentaje de pacientes que padecen cáncer piel melanoma en el año 2019

Fuente: Ministerio de Salud.

En la siguiente Figura, la barra de color verde y color azul representa al número de casos de pacientes que padecen melanoma y tumores malignos de la piel, respectivamente. El mayor número de casos por estas enfermedades se registra en el departamento de Lima, por delante de Lambayeque y Arequipa. (Ministerio de Salud, 2020)

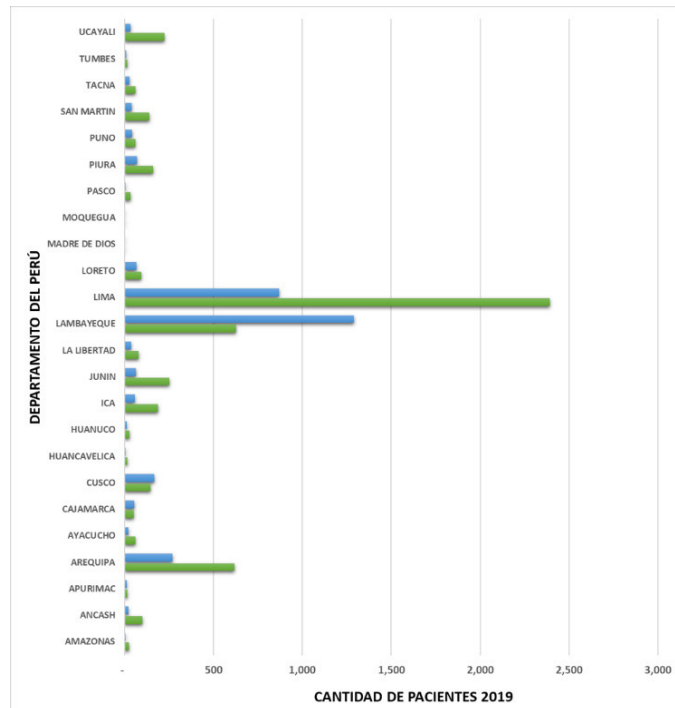
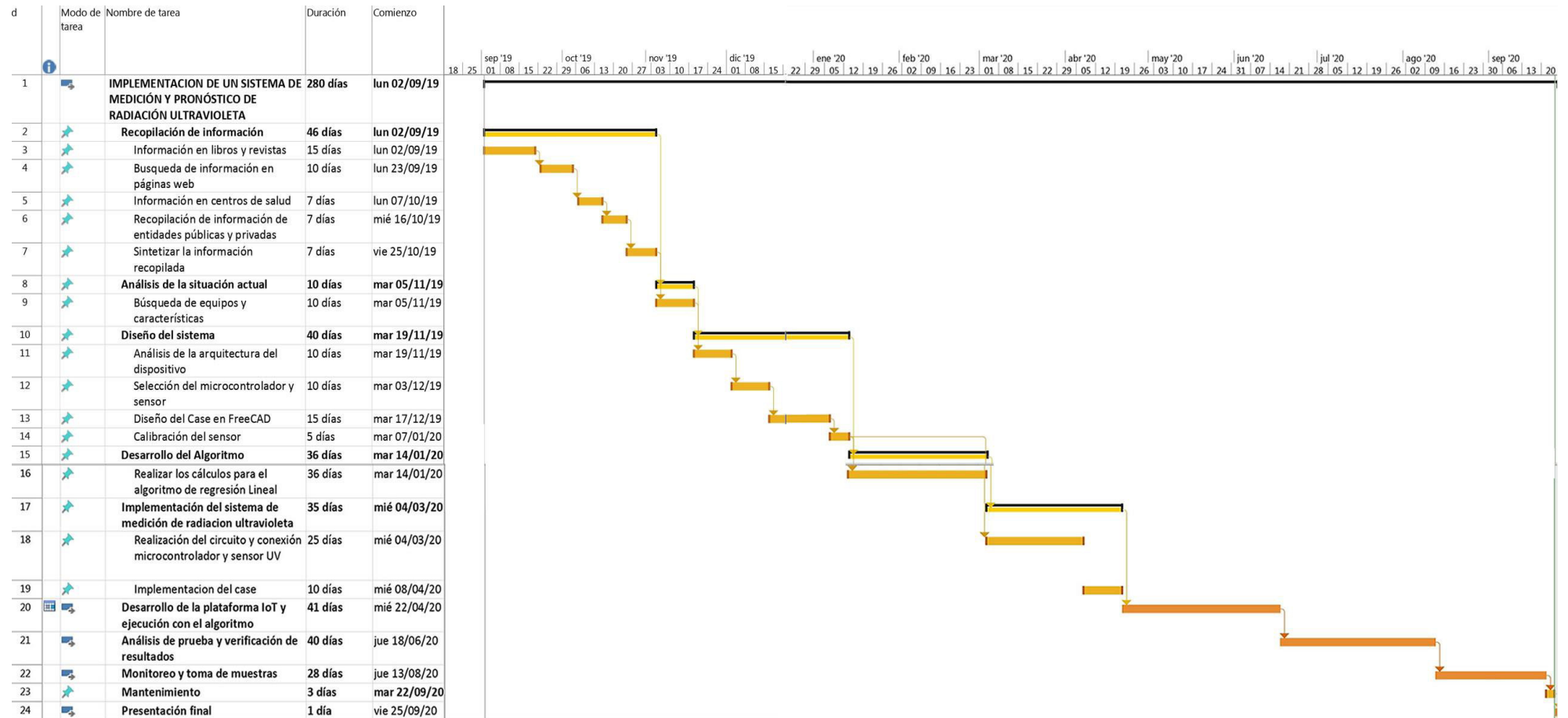


Figura: Cantidad de pacientes por departamentos.

Fuente: Ministerio de Salud.

Anexo F: Diagrama de Gantt



Anexo G: Matriz de consistencia

Título: Diseño e implementación de un sistema de medición y pronóstico de radiación ultravioleta utilizando Internet de las Cosas y Machine Learning

PROBLEMÁTICA	PREGUNTA GENERAL	OBJETIVO GENERAL	PREGUNTAS ESPECÍFICAS	OBJETIVOS ESPECÍFICOS	ACCIONES	MARCO TEÓRICO	DESARROLLO	EVALUACIÓN	CONCLUSIONES	RECOMENDACIONES
<p>En Perú, según el Servicio Nacional de Meteorología e Hidrología (SENAMHI), el año 2016 el índice de radiación Ultravioleta alcanzó un valor de 12 siendo considerado este número en el margen de nivel de riesgo muy alto para la salud. En el año 2019, el Ministerio de Salud (MINSA) ha reportado 8549 casos de pacientes que padecen enfermedades en la piel en todo el país, debido a las altas exposiciones prolongadas a la radiación ultravioleta. Es por este motivo, que se desea implementar un sistema que permita medir y pronosticar la radiación ultravioleta utilizando Internet de las Cosas y Machine Learning. Este proyecto proporcionará un pronóstico a corto plazo sobre el índice de radiación ultravioleta, para alertar y tomar medidas de prevención evitando algunos problemas de salud.</p>	<p>¿Cómo medir y pronosticar los niveles de radiación ultravioleta mediante Internet de las Cosas (IoT) y Machine Learning?</p>	<p>Diseñar e implementar un sistema de medición y pronóstico de radiación ultravioleta utilizando Internet de las Cosas y Machine Learning.</p>	<p>¿De qué manera se realiza la medición de los índices de radiación ultravioleta en la actualidad?</p>	<p>Describir la situación actual de la medición de la radiación UV.</p>	<p>Investigar sobre los niveles de riesgos de índice UV y que consecuencias negativas afectarían en la salud humana.</p>	<p>Radiación UV y su influencia de sobre la salud humana</p>	<p>Instrumentos de medición UV, estadísticas de datos de índice UV promedio y casos de pacientes en Perú.</p>	<p>Se realizó tomas de pruebas en 2 lugares diferentes para medir el índice de radiación UV.</p>	<p>Una exposición prolongada de radiación UV perjudica nuestra salud, desgraciadamente cada año se reportan un incremento de casos por enfermedades de la piel principalmente, el cáncer por melanoma siendo la principal causa de muerte. Además, enfermedades oculares que en el peor de los casos pueden perder la visión.</p>	<p>Tomar las medidas preventivas para el cuidado de la salud de la piel, aplicándose bloqueadores solares, lentes con protección UV y tratar de no estar prolongadamente expuesto a la radiación UV. De ese modo, estaremos evitando posibles enfermedades de piel y oculares, en un futuro.</p>
			<p>¿Cuáles son las etapas para el diseño e implementación del sistema?</p>	<p>Diseñar el diagrama de las etapas del sistema de medición y pronóstico de radiación ultravioleta.</p>	<p>Determinar los pasos para el desarrollo del sistema de medición y pronóstico.</p>	<p>Internet de las Cosas, componentes de una aplicación IoT, sensor, hardware y software</p>	<p>Etapas de adquisición, procesamiento de datos, conectividad, almacenamiento y modelo predictivo. Diagrama de conexión y diseño del case en 3D.</p>	<p>Visualización del índice UV en la pantalla OLED</p>	<p>Con este sistema de medición UV permite detectar el índice de radiación UV y visualizar los niveles de riesgos que representa, con actualización de 1 segundos y además dispone la capacidad de almacenar los datos en una plataforma en la nube con acceso a internet mediante la red Wi-Fi.</p>	<p>El sistema electrónico podría reducirse respecto en las dimensiones físicas, realizar en una única placa impresa todo el circuito electrónico como, el microcontrolador, el módulo wifi, el sensor de radiación, display oled y módulo de carga.</p>
			<p>¿Qué tipo de plataforma permite almacenar y procesar los datos en la nube para el diseño e implementación del sistema?</p>	<p>Identificar la plataforma de Internet de las cosas y que disponga compatibilidad para realizar Machine Learning.</p>	<p>Investigar las plataformas IoT y la compatibilidad con Machine Learning.</p>	<p>plataformas y protocolos IoT</p>	<p>Lista de plataformas IoT y configuración de ThingSpeak</p>	<p>Monitoreo y visualización de los datos de índice UV en la plataforma ThingSpeak</p>	<p>La importancia de la implementación de un sistema IoT con realizar análisis de datos sin desarrollar software o servidores web personalizados. El flujo de trabajo entre en MATLAB y ThingSpeak, hace una plataforma analítica de IoT que puede ejecutar en la propia nube.</p>	<p>El Internet de Las Cosas se encuentra en todo su apogeo y podemos aprovechar tecnológicamente con el uso de múltiples plataformas y adaptar nuestro proyecto con conectividad a Internet para almacenar, analizar y visualizar los datos.</p>
			<p>¿Qué tipo de análisis se necesita en el sistema de pronóstico de radiación ultravioleta?</p>	<p>Desarrollar el algoritmo predictivo para el sistema de pronóstico de radiación ultravioleta.</p>	<p>Investigar sobre Machine Learning y sus algoritmos.</p>	<p>Algoritmos de Machine Learning</p>	<p>Algoritmo de regresión lineal, desarrollo del Algoritmo en MATLAB Implementación de análisis en la nube</p>	<p>Recopilar los datos de índice UV de ThingSpeak programando en el software MATLAB</p>	<p>El algoritmo de regresión lineal resulta ser muy aplicado en múltiples situaciones para realizar predicciones de tipo lineal y también en no lineales.</p>	<p>En el momento de realizar el entrenamiento de datos usando cualquier algoritmo de ML, es importante identificar qué tipos de datos van manejar. En caso de que los datos a evaluar fuesen fecha y hora, se tendrá que hacer una conversión a dato tipo numérico.</p>
			<p>¿De qué manera se validará el sistema de medición y pronóstico de radiación ultravioleta?</p>	<p>Verificar los resultados del sistema de medición y pronóstico de radiación ultravioleta.</p>	<p>Obtener los resultados óptimo durante el funcionamiento de sistema.</p>	<p>Monitoreo en Arduino IDE, Plataforma IoT ThingSpeak y el software MATLAB</p>	<p>Pasos para usar el dispositivo, análisis de funcionamiento de medición y pronóstico de índice UV.</p>	<p>Uso del App ThingView para visualizar el índice UV en un smartphone y asignar en ThingSpeak una hora específica para pronosticar el índice UV.</p>	<p>Las pruebas realizadas de medición UV resultaron de manera eficiente con el correcto funcionamiento en lugares. Una ventaja muy esencial, es el resultado del valor de error para determinar el pronóstico de índice UV que será mínimo, a medida que se aproxima a la hora establecida y además si se agrega mayor cantidad de datos para el entrenamiento en el algoritmo de regresión lineal.</p>	<p>Si requiere mayor precisión para el cálculo del pronóstico de índice UV, es preferible que lo realice en la misma plataforma de ThingSpeak y no en el ordenador. ThingSpeak al ser una plataforma en la nube se evita el consumo excesivo de recursos en el ordenador como memoria RAM, CPU, etc.</p>

Anexo H: Conference Paper

Publicado en: 2018 IEEE *Sciences and Humanities International Research Conference* (SHIRCON)

Implementation of a System for the Evaluation of Environmental Factors that Use the Internet of Things

Publisher: IEEE

Cite This

PDF

Martha Medina-De-La-Cruz ; Anderson Mujaico-Mariano [All Authors](#)

84
Full
Text Views



Abstract

Document Sections

- I. Introduction
- II. The Skin
- III. Methodology
- IV. Results
- V. Future Work

[Show Full Outline](#)

Authors

Figures

References

Keywords

Metrics

Abstract:

Environmental factors influence the deterioration of human health, causing various diseases and depend on prolonged exposure in environmental environments where these factors are found in extremely harmful indexes for people. In recent years, the most worrisome environmental factors are air pollution due to the different concentrations of polluting gases and the high rates of **ultraviolet radiation**. If the time of exposure to the factors is greater, then the damage to health will be harmful. It is estimated that 24% of annual deaths worldwide and 33% of diseases in children under 5 years of age are due to exposure to risks due to environmental factors that could be prevented. For this reason, a measurement system was implemented to evaluate environmental factors and prevent existing diseases. The research project uses the NodeMCU ESP8266 platform for the development of the Internet of Things (IoT), which will allow sending the data to the server using the internet technology of things, where the data acquired by measuring the electrochemical sensors (MQ), temperature and humidity (DHT-22) and **ultraviolet radiation** sensor (GYML8515). This system works using a Blynk server allowing to monitor the data in real time.

Published in: 2018 IEEE Sciences and Humanities International Research Conference (SHIRCON)

Date of Conference: 20-22 Nov. 2018

INSPEC Accession Number: 18355987

Date Added to IEEE Xplore: 31 December 2018

DOI: 10.1109/SHIRCON.2018.8593015

► ISBN Information:

Publisher: IEEE

Conference Location: Lima, Peru

Figura: Publicación de Paper

Fuente: IEEE