



“SISTEMA PARA EL MONITOREO, DETECCIÓN Y ALERTA DE
SOMNOLENCIA DEL CONDUCTOR MEDIANTE VISIÓN ARTIFICIAL,
COMUNICACIÓN INALÁMBRICA Y GEOLOCALIZACIÓN”

Proyecto Terminal 1

Autores:

Alan Eduardo Gamboa Del
Ángel
Maite Paulette Díaz Martínez
Nilsen Alfaro Parra

Asesores:

M.en C. Niels Henrik Navarrete
Manzanilla
Dr. Rodolfo Vera Amaro

Índice

1. Resumen	6
2. Abstract	6
3. Capítulo I: Introducción	7
3.1. Planteamiento del problema	8
3.2. Propuesta de solución	10
3.3. Alcances	12
3.4. Escenario de pruebas	12
3.5. Justificación	13
3.6. Metodología	14
3.7. Objetivos	15
3.7.1. Objetivo General	15
3.8. Objetivos Específicos	15
4. Capítulo II: Marco de Referencia	16
4.1. Marco Teórico	16
4.1.1. Raspberry Pi	16
4.1.2. Visión Artificial	16
4.1.3. Red Neuronal	16
4.1.4. Redes Neuronales Convolucionales	17
4.1.5. <i>Content Delivery Network</i>	19
4.1.6. Estándares y Protocolos de Comunicación Inalámbrica	20
4.1.7. LTE-M	20
4.1.8. Teorema de Shannon-Hartley	21
4.1.9. Geolocalización	22
4.1.10. Funcionamiento	22
4.2. Estado del Arte	23
4.2.1. <i>Driver Drowsiness Detection Using Machine Learning with Visual Behaviour</i>	23
4.2.2. <i>Detection Of Drowsiness And Distraction Of Drivers Using CNN</i>	23
4.2.3. <i>Driver Drowsiness Detection System Using Convolutional Neural Networks</i>	23
4.2.4. Diseño e implementación de sistema de visión artificial para alerta y detección de somnolencia mediante aprendizaje profundo aplicable en conductores de vehículos	23
4.2.5. Sistema de detección de somnolencia mediante inteligencia artificial en conductores de vehículos para alertar la ocurrencia de accidentes de tránsito	24
4.2.6. Diseño de un sistema electrónico para detectar la somnolencia en automovilistas por medio de la actividad ocular	24
4.2.7. Sistema para la detección del estado de somnolencia en seres humanos, con reconocimiento de patrones	24
4.2.8. Sistema de Detección de Somnolencia	24
4.2.9. Desarrollar un prototipo de reconocimiento facial basado en Machine Learning para detectar estado de Somnolencia en conductores de una cooperativa de transporte.	25

4.2.10. Sistema basado en la detección y notificación de somnolencia en conductores de autos	25
4.2.11. Detección de somnolencia para conducción sin accidentes	25
4.2.12. Implementación de un sistema web con geolocalización para medir la transaccionalidad del portal ABC de la democracia para el consejo nacional electoral (CNE)	25
4.2.13. Diseño e implementación de un sistema de geolocalización en interiores para plataforma Android vía la red enterprise WLAN de la PUCP	26
4.2.14. Geolocalización con LoRa mediante multilateración	26
4.2.15. Propuesta de un sistema de geolocalización y monitoreo vía GPS/GSM/GPRS aplicado a un pulsómetro para personas con enfermedades cardiovasculares	26
4.2.16. Influencia de un sistema de geolocalización en el control y monitoreo de vehículos con dispositivos GPS en una empresa logística	27
4.2.17. Diseño de la red de acceso LTE en el distrito de Jesús María	27
5. Capítulo III: Análisis	28
5.1. Análisis de la Zona Geográfica	28
5.2. Análisis del Módulo Central de Procesamiento	30
5.2.1. Análisis del Submódulo Visión Artificial	31
5.2.2. Análisis y Elección del Modelo de Red Neuronal	32
5.3. Análisis del Módulo de Comunicaciones	35
5.4. Análisis del Módulo de Estación Base	36
5.4.1. Análisis de la Aplicación Web	42
6. Capítulo IV: Diseño	52
6.1. Diseño del Módulo Central de Procesamiento	52
6.1.1. Diseño de la Unidad Contenedora de Procesamiento	54
6.1.2. Diseño del Submódulo de Visión Artificial	59
6.2. Diseño del Módulo de Comunicaciones	61
6.2.1. Diseño del Submódulo de Transmisión de Datos	62
6.2.2. Diseño del Submódulo de Telemetría	63
6.2.3. Diseño del Submódulo de Cobertura	64
6.3. Diseño de la Estación Base	65
6.3.1. Diseño de la Aplicación Web	67
6.3.2. Diseño de la Base de Datos	73
7. Conclusiones	77
8. Referencias	78

Índice de figuras

1.	Diagrama general del diseño preliminar de la arquitectura del sistema.	11
2.	Descomposición en la metodología <i>Top-Down</i> . [5]	14
3.	Módulo estándar de una neurona artificial	16
4.	Arquitectura general de una Red Neuronal Convolutiva	18
5.	Ejemplo de Max Pooling	18
6.	Arquitectura de una CDN	19
7.	Mapa de cobertura 3G / 4G / 5G, México. [5]	28
8.	Mapa de cobertura 3G / 4G / 5G, Ciudad de México. [5]	29
9.	Boosting	29
10.	Redes Unicapa	32
11.	Redes Unicapa	32
12.	Redes Unicapa	33
13.	Redes Neuronal con 2 Capas Ocultas de 4 Neuronas cada una	33
14.	Representación Gráfica de la Función ReLu	34
15.	Redes Neuronal con 2 Capas Ocultas de 4 Neuronas cada una utilizando la función de Activación ReLu	34
16.	Representación gráfica de la función Tangente Hiperbólica	35
17.	Redes Neuronal con 2 Capas Ocultas de 4 Neuronas cada una utilizando la función de Activación Tanh	35
18.	Diagrama de Casos de Usos del Módulo de Estación Base	39
19.	Diagrama de Casos de Uso	48
20.	Programa clásico vs Machine Learning	52
21.	Modularidad	52
22.	Especificaciones Físicas	54
23.	Zumbador Pasivo KY-006	54
24.	Cámara Pivariety IMX230	55
25.	Jumpers	55
26.	Micro SD	55
27.	RaspberryPi 3G/4G/ LTE Base HAT	56
28.	LE 910Cx Mini PCIe Series Linux	56
29.	LTE Main Diversity GNSS Triple Port u.FL Antenna de 100mm	57
30.	Unidad Contenedora del Módulo Central de Procesamiento	58
31.	Puntos de referencia	59
32.	<i>Mouth Opening Ratio</i>	59
33.	Arquitectura del Módulo de Estación Base	65
34.	Diagrama de Componentes AWS Amplify	66
35.	Diagrama de Secuencia Detalle Reporte Incidencia	67
36.	Diagrama de Secuencia Consultar Ubicación	67
37.	Diagrama de Secuencia Recuperar Contraseña	68
38.	Diagrama de Secuencia Confirmar Incidencia	68
39.	Diagrama de Secuencia Consultar Perfil	68
40.	Diagrama de Secuencia Registrar Conductor	69
41.	Diagrama de Secuencia Modificar Conductor	69
42.	Diagrama de Secuencia Eliminar Conductor	69

43.	Diagrama de Secuencia Estado de los Periféricos	70
44.	Página Inicio de Sesión	70
45.	Página Principal	71
46.	Página Perfil del Conductor	71
47.	Página Detalle de Incidencia	72
48.	Página Ubicación en Tiempo Real	72
49.	Estructura de un Documento en MongoDB	73
50.	Modelo propuesto para la base de datos en MongoDB	74
51.	Documento de Registro de Conductores	74
52.	Documento de registro de Incidencias	75
53.	Documento de registro de Incidencias	76

Índice de tablas

1.	RF01- Verificar estado funcional de periféricos	30
2.	RF02- Activar Alarma	30
3.	RF01- Capturar Video en Tiempo Real	31
4.	RF02- Delimitar región de Interés	31
5.	RF01- Guardar Incidencia	36
6.	RF02- Guardar Video	37
7.	RF03- Conectar Aplicación Web	37
8.	RF04- Desplegar Aplicación Web	38
9.	RF05- Guardar Credenciales de Usuario Administrador	38
10.	Requerimientos No Funcionales del Módulo de Estación Base	39
11.	Caso de Uso 01	40
12.	Caso de Uso 02	40
13.	Caso de Uso 03	40
14.	Caso de Uso 04	40
15.	Caso de Uso 05	41
16.	RF01- Iniciar Sesión	42
17.	RF02- Mostrar el Historial de Reportes de Incidencia	42
18.	RF02- Mostrar el Historial de Reportes de Incidencia	43
19.	RF03- Visualizar Reporte de Incidencia	43
20.	RF04- Confirmar Incidencia	44
21.	RF05 - Recuperar Contraseña	44
22.	RF06- Mostrar perfil del Conductor	45
23.	RF07- Mostrar ubicación Geográfica	45
24.	RF08- Descartar Incidencia	46
25.	RF09- Registrar Usuario	46
26.	RF10- Mostrar el Historial de Reportes de Incidencia	46
27.	RF11- Mostrar el Historial de Reportes de Incidencia	47
28.	RF12- Generar Mensaje de Alerta	47
29.	Requerimientos No funcionales - Aplicación Web	48
30.	Caso de Uso 01	49
31.	Caso de Uso 02	49
32.	Caso de Uso 03	50
33.	Caso de Uso 04	50
34.	Caso de Uso 05	50
35.	Caso de Uso 06	50
36.	Caso de Uso 07	51
37.	Caso de Uso 08	51
38.	Caso de Uso 08	51
39.	Agrupación de Puntos de Referencia	59

1. Resumen

El presente documento detalla el proceso de diseño del proyecto titulado "Sistema para el monitoreo, detección y alerta de somnolencia del conductor mediante visión artificial, comunicación inalámbrica y geolocalización como Proyecto Terminal, dirigido a conductores de la ciudad de México. Dicho sistema hará uso de técnicas de visión artificial, tales como las funciones Haar para la detección de rostro, ojos y boca del conductor a partir de la entrada de video en tiempo real, así como la propuesta del uso de una red neuronal convolucional entrenada para poder detectar el estado de los ojos. El sistema utilizará puntos de referencia faciales para marcar la posición de la boca y posteriormente se obtendrá la relación de apertura de la boca, con el fin de detectar si se trata de un bostezo. Si el sistema detecta somnolencia, se activará una alarma con intención de alertar al conductor, a su vez, se realizará un reporte de incidencia el cual será enviado a una base de datos. El sistema utilizará una Raspberry Pi para su funcionamiento de manera portátil. Como parte del sistema se analizó y diseñó un sistema de administración que permitirá al usuario verificar y visualizar la información de los reportes de incidencia y la información de cada conductor. Para el cual se diseñó una aplicación web que le permita al usuario acceder a la información y adicionalmente monitorear la ubicación en tiempo real del conductor.

Palabras Clave: Somnolencia, Visión artificial, Geolocalización, Machine Learning, Alerta, Red Neuronal Convolucional.

2. Abstract

This document details the design process of the project entitled "System for the monitoring, detection and alert of driver drowsiness through artificial vision, wireless communication and geolocation." as a Terminal Project, aimed at drivers in Mexico City. Said system will make use of artificial vision techniques, such as the Haar functions for the detection of the driver's face, eyes and mouth from the video input in real time, as well as the proposal of the use of a convolutional neural network trained to be able to detect the state of the eyes. The system will use facial reference points to mark the position of the mouth and then the mouth opening ratio will be obtained, in order to detect if it is a yawn. If the system detects drowsiness, an alarm will be activated with the intention of alerting the driver, in turn, an incident report will be made which will be sent to a database. The system will use a Raspberry Pi for its portable operation. As part of the system, an administration system was analyzed and designed that will allow the user to verify and view the information of the incident reports and the information of each driver. For which a web application was designed that allows the user to access the information and additionally monitor the location in real time of the driver.

Keywords: Drowsiness, Artificial vision, Geolocation, Machine Learning, Alert, Convolutional Neural Network.

3. Capítulo I: Introducción

De acuerdo con el Informe sobre la situación mundial de la seguridad vial 2018, publicado por la Organización Mundial de la Salud (OMS), el número anual de muertes por accidentes de tránsito llegó a los 1,35 millones. Sin embargo, las consecuencias de estos accidentes no solamente afectan al conductor y a los pasajeros del vehículo, sino que también involucran a los peatones, ciclistas y motociclistas, en particular aquellos que residen en países en desarrollo. [1]

La somnolencia al conducir es reconocida como un factor que puede contribuir a los percances viales. De acuerdo a datos de CONAPRA, en México cada año mueren en promedio 16,500 mexicanos por percances de este tipo. Con base en estimaciones hechas por el Secretariado Técnico del Consejo Nacional para Prevención de Accidentes (STCONAPRA), los accidentes viales le cuestan al país alrededor de 150 mil millones de pesos que representan el 1.7 % del Producto Interno Bruto (PIB), sumando costos directos e indirectos. [2]

La somnolencia, al momento de conducir se trata de un fenómeno complejo que implica disminuciones en los niveles de alerta y conciencia por parte del conductor. Esta situación se convierte en un detonador de accidentes pues disminuye en el sujeto la capacidad de identificar situaciones de riesgo y por tanto evitar el peligro.[2]

En el desarrollo logístico de empresas que se encargan de transportar a pasajeros o que se encargan de repartir paquetería, los datos pueden verse de forma más alarmante o preocupante, ya que, usualmente, las personas designadas como conductores se enfrentan a largas horas de jornada sin descanso y en muchos casos, rotan por horarios que pueden ser en el día o en la noche, aumentando el riesgo de accidentarse por el desbalance de tener un horario mixto.

Muchas de estas empresas optan por sistemas de rastreo de vehículos, que hoy es un recurso determinante para la planificación de rutas de distribución al optimizar las mismas y, lo más importante, al estar en un constante monitoreo se puede tener la certeza de que se está siguiendo la ruta marcada correctamente. [3]

El rastreo satelital de vehículos funge como una de las principales opciones de sistemas capaces de indicar la ubicación de los vehículos en cuestión, sin embargo, el costo es elevado y periódico.

Tomando en cuenta el contexto antes planteado, se analizará y diseñará en el presente documento un sistema que alerte a un conductor con somnolencia, mediante el uso de una cámara digital y algoritmos de visión artificial, además de incluir un sistema de administración que permita la geolocalización del sistema en tiempo real, así como la gestión de conductores y reportes de incidencias.

3.1. Planteamiento del problema

La somnolencia es un fenómeno complejo de analizar debido a los factores que pueden intervenir. Algunas de las características más notorias de un estado de somnolencia, se pueden apreciar principalmente en el rostro de las personas: frecuencia de parpadeo, bostezos, movimientos faciales y cabeceos, los cuales son parámetros claves para determinar si una persona está en estado de somnolencia o vigilia.

Gracias a los avances tecnológicos en los últimos años, se han comenzado a desarrollar técnicas de visión artificial y aprendizaje automático que permiten detectar patrones de manera más eficiente. En el área del hardware, se han desarrollado microordenadores capaces de realizar tareas que requieran un nivel moderado de computación de una manera eficaz. Finalmente, en el área de telecomunicaciones, tecnologías como el 4G y avances en el área de *Internet of Things* han permitido mayores velocidades de transmisión de datos, así como mayor cobertura dentro del territorio nacional.

Hoy en día, se están comenzando a utilizar tecnologías para prevenir y detectar síntomas de fatiga y somnolencia en conductores. Donde se han realizado diversos estudios y soluciones que intentan dar respuesta al problema de monitoreo de la somnolencia.

Uno de estos métodos está basado en el comportamiento del vehículo, el cual detecta el estado del conductor mediante el análisis de distintas métricas como son: movimientos del volante, posición del vehículo, la presión del acelerador o del freno, cambio de velocidades, con los cuales se determina la posibilidad de que el conductor se encuentre en estado de somnolencia. El principal problema de dicho método es que las características individuales del vehículo, conductor y carretera repercuten en la eficacia del sistema. [4]

Existen también sistemas que analizan el comportamiento del vehículo y su entorno. Estos sistemas cuentan con cámaras frente al vehículo, donde se digitaliza el entorno del vehículo; posteriormente se analizan parámetros como la reacción del conductor frente a la dirección de su vehículo con respecto a la carretera, velocidad, comportamiento del vehículo frente a baches en la carretera y distancia con respecto al vehículo que esta adelante. Si se detecta un comportamiento anormal en los parámetros anteriormente mencionados, se procede a realizar las alertas necesarias para advertir al conductor. No obstante, estos sistemas suelen ser costosos y no toman en cuenta el estado del conductor.

Por otra parte, se encuentran los métodos que se basan en el análisis de variables fisiológicas, los cuales permiten la detección de somnolencia en sus fases tempranas con una baja tasa de falsos positivos. Se destacan los métodos basados en: electroencefalograma (EEG), electromiograma (EMG), electrocardiograma (ECG) y electrooculograma (EOG). [4]

Cabe mencionar que, entre todos los métodos, el EEG es el más común para la detección de la somnolencia, donde se analizan diferentes bandas de frecuencia. Todas estas señales brindan información adicional al momento de analizar el estado de somnolencia de una persona. Sin embargo, estos métodos requieren contacto con el conductor y el uso excesivo de canales de encefalogramas, lo cual resta comodidad, maniobrabilidad y practicidad al conductor, además de poder llegar a ser

invasivos, lo cual puede llegar a entorpecer el desempeño del conductor. [4]

Finalmente, se encuentra el análisis de características visuales que puede presentar un conductor somnoliento, como los movimientos faciales, parpadeos rápidos y constantes, cabeceos y bostezos frecuentes. Sin embargo, los bostezos se presentan generalmente antes de que el conductor entre en somnolencia, mientras que, los cabeceos normalmente ocurren cuando el conductor se duerme. Por lo que estos métodos no son capaces de detectar con exactitud cuando un conductor está empezando a entrar en un estado de somnolencia, sin embargo, son los métodos visuales más adecuados para dicho propósito. [4] Por tanto, se debe tomar en cuenta las diferencias temporales entre los distintos signos visuales, por lo que realizar la combinación de varias de estas características aumentará la robustez final del sistema, logrando una mejor eficacia en el sistema.

Debido a las dificultades que pueden presentarse en los sistemas a la hora de detectar la somnolencia y la eficacia de este, así como las dificultades que muchos de los sistemas anteriores presentan en cuanto al espacio y potabilidad, se hará uso del método de análisis de características visuales para brindar una mejora y complemento a los sistemas actuales, usando visión artificial, además de integrar un sistema de monitoreo que permita a un tercero seguir la ubicación del conductor y registrar las incidencias del mismo.

Actualmente existen varias opciones para la geolocalización de vehículos en tiempo real, pero con costos elevados, como lo es el rastreo satelital que se tiene que pagar el servicio con una suscripción anual o mensual. Entonces surge esa necesidad de poder encontrar soluciones más asequibles pero que estén a la vanguardia del mercado, poder hacer uso de tecnologías gratuitas o que reduzcan los costos, como es el caso de las redes 4G, y así lograr la correcta comunicación del sistema para rastrear el vehículo en cuestión.

Dadas las posibles soluciones y escenarios antes mencionados se plantea la siguiente pregunta: ¿Cómo desarrollar un sistema portátil que pueda detectar la somnolencia y alertar al conductor, además de monitorear la ubicación en tiempo real y gestionar las incidencias detectadas en el conductor?

3.2. Propuesta de solución

Como respuesta a la problemática planteada en la sección 3, se propone el desarrollo de un sistema portátil que sea capaz de detectar somnolencia en conductores y su vez activar una alarma que permita alertar al conductor. Adicionalmente se desarrollará como parte del sistema, un subsistema que permita la administración y validación de las incidencias reportadas, así como el monitoreo GPS del conductor en tiempo real dentro de la Ciudad de México.

Los problemas que resolverá el sistema serán listados a continuación:

- El sistema será portátil por lo que se hará uso de un microordenador para poder ser instalado y que funcione dentro de un vehículo.
- Mediante una cámara digital conectada al microordenador, el sistema analizará el rostro del conductor en tiempo real, tomando en cuenta parámetros de análisis tales como; rostro, ojos y boca del conductor.
- El sistema hará uso de técnicas de visión artificial para poder detectar si el conductor presenta signos de somnolencia.
- En caso de presentarse un caso de somnolencia, el sistema activará una alarma para alertar al conductor.
- Al mismo tiempo, el sistema realizará y almacenará un reporte de incidencia que contendrá datos como fecha, hora, ubicación geográfica, y un pequeño video clip que muestre el momento en que el conductor presentó signos de somnolencia.
- Posteriormente, el sistema hará uso de redes de telecomunicaciones móviles para enviar el reporte de incidencia previamente generado hacía una estación base que funcionará mediante una aplicación web, dónde un administrador podrá verificar este reporte con el fin de confirmar que se trata de un caso de somnolencia y no un falso positivo.
- A su vez, el administrador podrá consultar la ubicación en tiempo real del conductor desde la misma estación base.
- En la figura 1 se muestra la propuesta del diagrama general de diseño de la arquitectura del sistema.

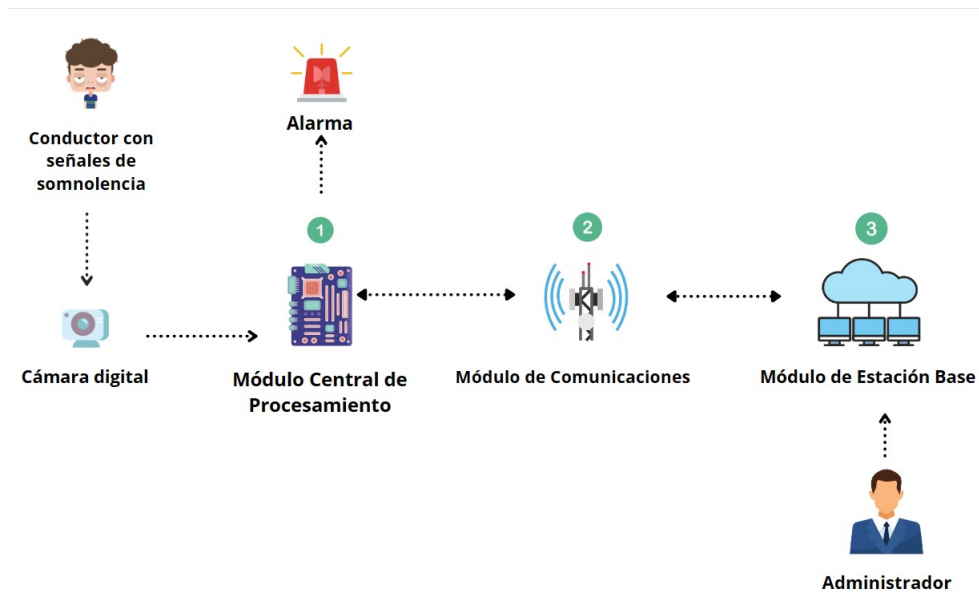


Figura 1: Diagrama general del diseño preliminar de la arquitectura del sistema.

3.3. Alcances

3.4. Escenario de pruebas

Las pruebas se dividirán en dos fases:

Fase 1:

En la primera fase se probará el sistema de somnolencia y el sistema de administración de manera separada, con el propósito de analizar los resultados obtenidos. Para el sistema de detección de somnolencia será colocado dentro del vehículo, sin embargo, dicho vehículo permanecerá estacionado y con el motor apagado, esto con el fin de realizar pruebas de la precisión del sistema y poder ajustar parámetros en caso de ser necesario.

En un primer escenario se examinarán los siguientes signos de somnolencia sin el uso de gafas de sol:

Estado de los ojos:

- Abiertos, cerrados.
- Duración de estos estados.

Bostezos:

- Apertura de la boca.

En un segundo escenario se examinarán los mismos signos de somnolencia con el uso de gafas de sol.

Estas pruebas se realizarán tanto de noche como de día para comprobar la efectividad del sistema. En caso de ser necesario, se realizarán los ajustes correspondientes en los parámetros de análisis para mejorar la efectividad de las pruebas. Así mismo se verificará que la alarma se active correctamente al detectar somnolencia en el conductor.

Para el sistema de administración se probará la aplicación web que incluye el acceso a la geolocalización del sistema en tiempo real, así como pruebas de estrés al servidor de alojamiento, con el fin de conocer la eficacia de respuesta del sistema durante periodos de tráfico de datos elevados.

Fase 2:

En la segunda fase, se probará el sistema completo con el vehículo en movimiento, donde el sistema estará monitoreando el estado de somnolencia del conductor en todos sus trayectos.

Para el sistema de administración se corroborará que en los casos en que el sistema de somnolencia detecte una incidencia, esta sea enviada correctamente y pueda ser visualizada desde el sistema de administración. A su vez, se constará que la información de la ubicación geográfica del sujeto de prueba esté disponible en todo momento.

3.5. Justificación

Actualmente, nos encontramos en la era digital donde se han desarrollado diferentes avances tecnológicos tales como el internet de las cosas (IoT) el cual permite la comunicación con dispositivos y da pauta al desarrollo de sistemas inteligentes que resuelven problemas complejos de manera automática y eficaz en ambientes específicos, la inteligencia artificial, el machine learning, el procesamiento masivo de datos en menor tiempo, entre muchos otros.

El sistema propuesto es capaz de detectar la somnolencia apoyado en estudios donde se analizan los aspectos fisiológicos y donde se delimitan los datos biométricos del estado de somnolencia obtenidos en diferentes pruebas, mediante el uso de visión artificial y machine learning para tener mejores resultados y posteriormente ser enviados a una estación base con ayuda de la tecnología de las redes inalámbricas.

Al haber accidentes de tránsito por somnolencia en México, con este proyecto se pretende prevenirlos e integrar un sistema que permita el monitoreo del vehículo mediante la geolocalización del mismo, el uso de una alarma auditiva para que el conductor se mantenga alerta, así como un sistema interactivo con los usuarios que permita visualizar las incidencias que se puedan presentar por parte de los conductores.

Ofreciendo así un sistema que utilice las tecnologías de la nueva generación que permitan el monitoreo, la detección y la alerta del conductor.

3.6. Metodología

Debido a que el presente proyecto conjunta la parte de software con hardware, se estarán implementando dos metodologías.

- **Top-Down**

La metodología de diseño *Top-Down* consiste en dividir un problema complejo en problemas o partes más pequeñas con mayor facilidad de resolución. A este proceso se le llama *Descomposición*. Estas partes pueden ser llamadas módulos o subrutinas. Cada módulo puede ser probado individualmente. [5]

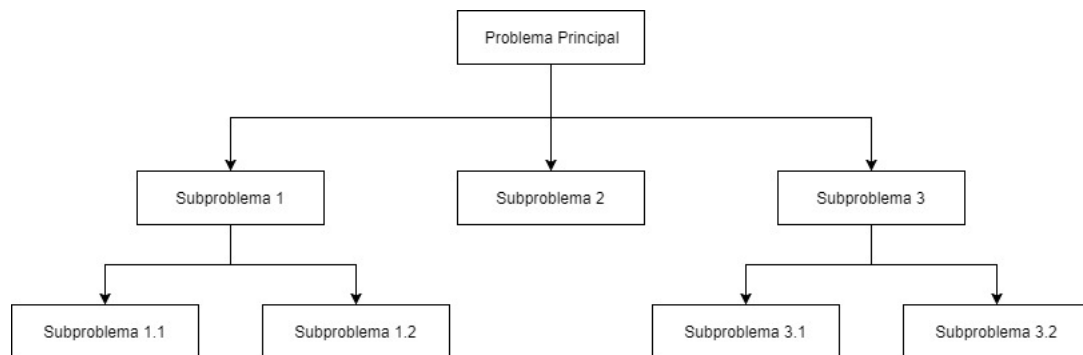


Figura 2: Descomposición en la metodología *Top-Down*. [5]

Utilizando la metodología Top-Down, se puede dividir el sistema en módulos que serán independientes de sí mismo, esto con la finalidad de ahorrar tiempo, así como recursos para su mantenimiento. Por lo tanto, en este proyecto se comenzará entendiendo el problema general para posteriormente poder realizar una subdivisión de problemas más pequeños para ser resueltos de manera individual.

- **UML**

UML es el estándar de la industria para modelar sistemas orientados a objetos [6]. UML incluye un conjunto de herramientas que permite visualizar la construcción de un sistema orientado a objetos. Al trabajar por iteraciones, se aborda de manera cada vez con más detalle el diseño de sistema, hasta que se defina con claridad las relaciones de cada uno de los objetos del sistema. Uno de las principales ventajas de UML es la reutilización de objetos, esto permite reducir costo de desarrollo así como de mantenimiento de los programas.

3.7. Objetivos

3.7.1. Objetivo General

Analizar y diseñar un sistema para la detección de síntomas de somnolencia y alerta del conductor. Además de ser capaz de obtener la ubicación del conductor en tiempo real para ser monitoreada desde una aplicación web que a su vez permita gestionar y visualizar los reportes de incidencias del mismo.

3.8. Objetivos Específicos

Para Trabajo Terminal 1:

- Diseñar un sistema de visión artificial que sea capaz de detectar la somnolencia en conductores.
- Diseñar un sistema de geolocalización en tiempo real usando redes de telecomunicaciones.
- Diseñar un sistema de administración para la gestión de usuarios, almacenamiento de datos y consultas de las incidencias de cada conductor.
- Diseñar un sistema de comunicaciones que permita la interconexión entre el sistema de visión artificial y el sistema de administración.

Para Trabajo Terminal 2:

- Implementar un sistema de visión artificial que sea capaz de detectar la somnolencia en conductores.
- Implementar un sistema de geolocalización en tiempo real usando redes de telecomunicaciones.
- Implementar un sistema de administración para la gestión de usuarios, almacenamiento de datos y consultas de las incidencias de cada conductor.
- Implementar un sistema de comunicaciones que permita la interconexión entre el sistema de visión artificial y el sistema de administración.

4. Capítulo II: Marco de Referencia

4.1. Marco Teórico

4.1.1. Raspberry Pi

La Raspberry Pi es una computadora de bajo costo y con un tamaño compacto, puede ser conectada a un monitor de computador o un TV, y usarse con un mouse y teclado estándar. Es un pequeño computador que corre un sistema operativo linux capaz de permitirle a las personas aprender a programar en lenguajes como Scratch y Python. Es capaz de hacer la mayoría de las tareas típicas de un computador de escritorio, desde navegar en internet, reproducir videos en alta resolución, manipular documentos de ofimática, hasta reproducir juegos. [1]

4.1.2. Visión Artificial

Se puede definir como el conjunto de técnicas para la extracción de información de imágenes digitales. El tipo de información obtenida de una imagen puede ser de identificación, mediciones para navegación, o aplicaciones de realidad aumentada[12]. El campo de visión artificial puede verse como una parte de la informática. La teoría de algoritmos y el aprendizaje automático son esenciales para el desarrollo de algoritmos de visión artificial. La visión artificial está compuesta generalmente por un conjunto de procesos destinados a realizar el análisis de imágenes: captación de imágenes, memorización de la información, proceso e interpretación de los resultados[12].

4.1.3. Red Neuronal

Una red neuronal es un sistema que pretende emular ciertas características propias de los seres humanos, tales como la capacidad de memorizar o y asociar hechos o características. Este sistema está se basa en el concepto de *neurona*.

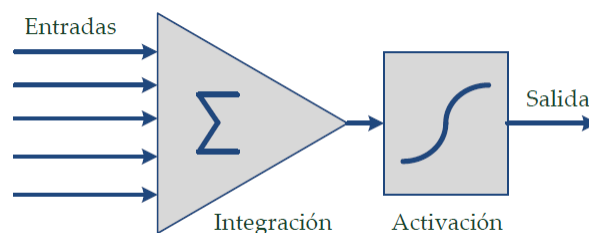


Figura 3: Modelo estándar de una neurona artificial

Un modelo simplificado de una neurona artificial consta de dos etapas. En la primera etapa, las entradas provenientes de otras neuronas son combinadas tomando en cuenta los pesos de las sinapsis. Como resultado de esta etapa surge la entrada neta o *excitación* de la neurona. En la segunda etapa, la entrada neta se utiliza para determinar el valor de salida de la neurona, que posteriormente será propagada a otras neuronas.

En la etapa de integración de las entradas, una neurona combina las distintas entradas x_i con sus pesos para así determinar su entrada neta z_j :

$$z_j = \sum_i w_{ij} x_i \quad (1)$$

Dónde w_{ij} representan los pesos sinápticos asociadas desde la i -ésima neurona hasta la j -ésima. Estos pesos tendrán valores real. Positivos para modelar conexiones excitatorias y negativos para conexiones inhibitorias.

Por otra parte, en la etapa de activación de una neurona, esta utiliza el valor asociado a su entrada neta para generar una salida y_j :

$$y_j(t) = F(y_j(t-1), z_j(t)) = F(y_j(t-1), net_j(t)) \quad (2)$$

4.1.4. Redes Neuronales Convolucionales

Las redes neuronales convolucionales, tambien conocidas como *redes convolutivas*, son redes neuronales artificiales que se utilizan comunmente para resolver problemas que requieren el procesamiento de imágenes. Sus casos de uso más frecuentes van desde la detección de objetos, hasta generar una descripción textual del contenido de una imagen. Particularmente, sus entradas y salidas pueden ser estructuradas. Esto quiere decir, que en lugar de recibir un vector de entradas, se puede recibir un vector (1D), matriz (2D) o tensor ($> 2D$). En el caso de señales bidimensionales, las entradas pueden pertenecer a los pixeles de una imagen capturada por una cámara.

Como su nombre lo indica, esta red neuronal utiliza la operación de convolución. La convolución es una operación matemática que se realiza sobre dos funciones para producir una tercera que se suele interpretar versión modificada (filtrada) de las funciones originales.[14]

La convolución entre las funciones f y g se representa de la siguiente manera:

$$(f \star g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau = \int_{-\infty}^{\infty} f(t - \tau)g(\tau)d\tau \quad (3)$$

En el caso particular de procesamiento digital de imágenes, las variables $[n_1, n_2]$ corresponden a cordenadas $[x, y]$ de los píxeles de una imagen. Además, el signo menos que aparece en la Ecuación 8, se suele sustituir por un signo más, por lo cual la definición de convolución se expresaría como:

$$(x \star h)[x, y] = \sum_{k_1=0}^{K_1-1} \sum_{k_2=0}^{K_2-1} h[k_1, k_2]x[x + k_1, y + k_2] \quad (4)$$

Capas convolutivas

El componente clave de las redes neuronales convolucionales son las capas, que sustituyen la tradicional multiplicación de pesos por entradas. La entrada de una capa convolutiva es una señal, en el caso de imágenes, una señal bidimensional. Dicha señal es procesada realizando una convolución con una máscara o *kernel*. Los pesos correspondientes al *kernel* serán los parámetros de la capa convolutiva que será entrenada. A continuación se presenta la representación más simple de las capas de una red neuronal convolucional.

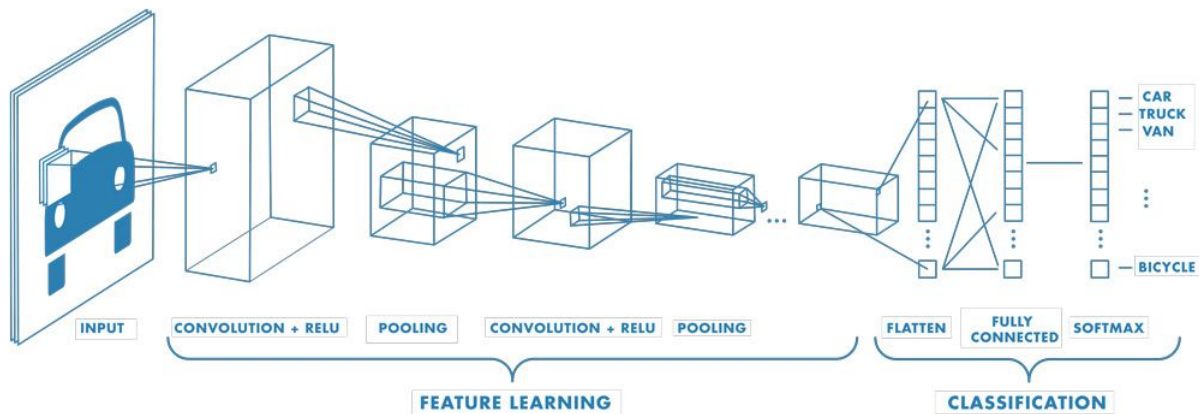


Figura 4: Arquitectura general de una Red Neuronal Convolutiva

■ Capa Convolutiva

Esta capa se utiliza para extraer las diversas características de las imágenes de entrada. En esta capa se realiza la convolución entre la imagen de entrada y un *kernel* de un tamaño particular $M \times M$.

La salida se denomina mapa de características, que nos brinda información sobre la imagen, como las esquinas y los bordes. Posteriormente, este mapa de características se alimenta a otras capas para aprender otras características de la imagen de entrada.

La capa de convolución en CNN entrega el resultado a la siguiente capa una vez que se aplica la en la entrada.

■ Capa de *Pooling*

El objetivo principal de esta capa es disminuir el tamaño del mapa de características convolucionado para reducir los costos computacionales. Esto se realiza disminuyendo las conexiones entre capas y opera de forma independiente en cada mapa de características. Según el método utilizado, existen varios tipos de operaciones de agrupación. Básicamente resume las características generadas por una capa de convolución. El tipo de pooling más utilizado es el *max-pooling*. Este devuelve el valor máximo del fragmento de la imagen filtrada por el kernel.

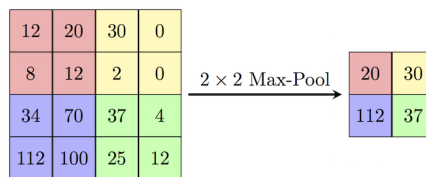


Figura 5: Ejemplo de Max Pooling

■ *Fully Connected Layer*

La capa totalmente conectada consta de los pesos y sesgos junto con las neuronas y se utiliza para conectar las neuronas entre dos capas diferentes. La imagen de entrada de las capas anteriores se aplanada y se alimenta a esta capa. Luego, el vector aplanado pasa por unas pocas capas más donde normalmente tienen lugar las operaciones de las funciones matemáticas. En esta etapa comienza el proceso de clasificación.

- *Output Layer*

Esta capa se encarga de mejorar el rendimiento de un modelo de aprendizaje automático, ya que evita el sobreajuste al simplificar la red. Descarta neuronas no esenciales de la red neuronal durante el entrenamiento.

- *Funciones de activación*

Son funciones matemáticas que determinan la salida de una capa de la red neuronal, y se utilizan en capas convolucionales así como en las capas completamente conectadas. Estas funciones se utilizan para asignar los valores de las salidas de cada capa entre valores definidos, para así facilitar que el modelo se adapte a una gran variedad de datos.

- *Flattening*

Flattening consiste en convertir los datos de salida de la capa convolucional a una matriz unidimensional para ser llevada a la capa siguiente.

4.1.5. *Content Delivery Network*

Una Red de Distribución de Contenido, o CDN por sus siglas en inglés, es de una red de servidores que se encarga de mejorar el rendimiento en la carga y descarga de contenido de la red. Los componentes de una CDN se encuentran distribuidos en diferentes nodos. Estos componentes pueden ser servidores y se encuentran con contenido replicado alrededor del mundo. El trabajo de una CDN consiste en redirigir la solicitudes de los clientes a un servidor ubicado geográficamente cerca de estos. En consecuencia, los clientes obtienen los datos solicitados a una velocidad más rápida.

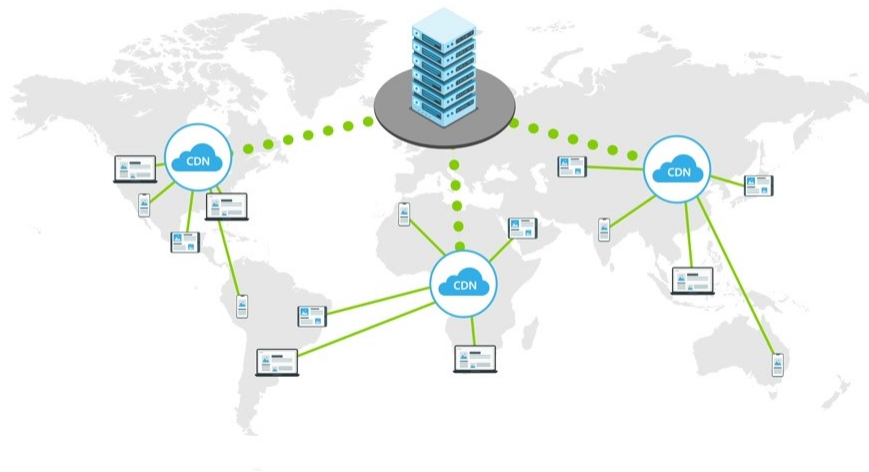


Figura 6: Arquitectura de una CDN

Además de esto, una CDN es capaz de detectar cambios en la información existente, así como detectar la disponibilidad de nuevo contenido en los servidores de origen.

4.1.6. Estándares y Protocolos de Comunicación Inalámbrica

Para transferir datos o información de un punto a otro sin la utilización de cableado o algún medio físico, tenemos las redes inalámbricas que utilizan ondas de radio para conectar a los dispositivos permitiendo así, a los dispositivos remotos, se conecten sin dificultad y sin importar que estos dispositivos estén a unos metros o incluso a varios kilómetros de distancia [?]. Se dividen en 4 tipos dependiendo del alcance requerido y se definen por el estándar 802.11 del IEEE que es el organismo de estandarización internacional [?].

- **Red inalámbrica de area amplia (WWAN)**

Usan ondas de radio pero transmite a uno o varios puntos de acceso inalámbrico donde un usuario inalámbrico puede conectarse a la red, al disponer de un ancho de banda más elevado ofrece una mejor cobertura.

Como ejemplo de estas redes se tienen la tecnologías 4G y 5G. Son conocidas como redes de largo alcance con cobertura de hasta 100km, pueden dar soporte a gran parte del territorio geográfico[24].

En esta red se incluyen:

- **Celulares**
Es conocida como la red de telefonía móvil.
- **LPWAN(*Low Power Wide Area Network*): Red de Área Amplia de Baja Frecuencia**
son redes de área amplia y de baja potencia, es un protocolo de transporte inalámbrico de datos que hoy en día se utiliza como uno de los protocolos básicos para la implementación de IoT. Existen varias implementaciones del protocolo LPWAN, tales como Sigfox, LoRaWAN, NB-IoT y LTE. Hay muchas diferencias entre cada una de ellas en cuanto a los esquemas de modulación, el alcance geográfico, la cantidad de información transmitida y a sus capacidades de encriptación y autenticación[25].

Las interfaces de comunicación inalámbrica LPWAN consideradas son:

4.1.7. LTE-M

LTE-M es el término simplificado de la industria para el estándar tecnológico de Área Amplia de Baja Potencia (LPWA por sus siglas en inglés). Se refiere específicamente a la tecnología LTE CatM1.

Es una tecnología de área amplia de baja potencia que admite IoT a través de dispositivos poco complejos y proporciona una cobertura extendida, al tiempo que permite la reutilización de la base instalada de LTE.

Las redes LTE-M coexistirán con las redes móviles 2G, 3G y 4G y se beneficiarán de todas las características de seguridad y privacidad de las redes móviles, como la confidencialidad

de la identidad del usuario, la autenticación de la entidad, la privacidad, la integridad de los datos y la identificación del equipo móvil[26].

A continuación se describen las ventajas principales de LTE-M:

- **Ahorro de Energía**

LTE-M permite que los dispositivos IoT usen un modo de ahorro de energía, que les permite ir a dormir cuando no están en uso. Y tampoco tienen que despertarse para enviar solo la actualización de ubicación, sino que pueden extender los tiempos de suspensión por un período de tiempo más largo. LTE-M también permite que los dispositivos usen recepción discontinua extendida (eDRX). Cuando el dispositivo está fuera del modo suspendido, verifica periódicamente la información del enlace descendente. Con eDRX, se aumenta el tiempo de verificación de la radio, lo que reduce el consumo de energía.

- **Cobertura**

Las redes LTE-M/Cat-M1/Cat-M2 pueden utilizar la infraestructura 4G LTE existente. Esta es una ventaja significativa, ya que más del 50 por ciento de las conexiones móviles globales se realizan en redes 4G.

Con una pérdida de acoplamiento máxima (MCL) de 156 decibelios (dB), 14 dB más alta que LTE, las redes LTE-M también ofrecen mayor cobertura y mejor penetración en interiores. La señal puede manejar muchas interferencias de edificios y otras estructuras que obstruyen su camino.

- **Velocidad de Transferencia**

Comparado con LTE, LTE-M no es particularmente rápido. Pero 1 megabit por segundo para transmisiones de enlace ascendente y descendente es excepcional en comparación con redes más antiguas como 2G y 3G (UMTS) y otras LPWAN. (Es varias veces más rápido que NB-IoT). Y eso es más que suficiente para la mayoría de las aplicaciones de IoT. Es factible para aplicaciones con necesidades de datos pequeñas e incluso casos de uso que involucran transmisión de video.

En el contexto de IoT, el alto rendimiento de datos significa que los dispositivos que usan LTE-M pueden recibir fácilmente actualizaciones por aire (OTA), y las transmisiones de datos consumirán menos energía, porque el dispositivo puede volver al modo de ahorro de energía más rápido.

4.1.8. Teorema de Shannon-Hartley

Un sistema óptimo es el que cuenta con la capacidad de minimizar la probabilidad de error de bit a la salida del sistema, esto depende de las restricciones de la energía transmitida y del ancho de banda del canal[?].

El teorema de Shannon-Hartley establece la máxima cantidad de información que puede ser transmitida sin error con un ancho de banda específico y que está expuesto a la interferencia de ruido. La ecuación para la capacidad de canal es[?]:

$$C = B \cdot \log\left(1 + \frac{S}{N}\right) \quad (5)$$

Donde C es la capacidad de canal, es decir, la velocidad máxima a la que se puede transmitir la información a lo largo del canal sin error, medida en bits por segundo, B es el ancho de banda en

hertz, S es la potencia de la señal útil en watts y N es la potencia de ruido presente en el canal expresada en watts. Al término S/N se le conoce como relación señal a ruido[?].

4.1.9. Geolocalización

La Geolocalización consiste en la identificación de la posición de un dispositivo móvil en el espacio real. El Sistema de Posicionamiento Global GPS por sus siglas en inglés es la forma más común y precisa en que se realiza la localización geográfica, y es capaz de ubicar el aparato con una precisión de unos pocos metros. El GPS es una red satelital que cuenta con al menos 30 satélites y que se mantienen en órbita alrededor de la tierra. Si bien el sistema en sus inicios tenía un propósito militar, en la actualidad cualquier persona puede ocuparlo[?].

4.1.10. Funcionamiento

Cuando se solicita el posicionamiento por medio del GPS este envía señales de radio que permiten localizar a los satélites, el centro de comando transmite la información de la órbita, el tiempo y la posición de los otros satélites en el mismo sistema GPS. Estos satélites envían simultáneamente la información de tiempo y órbita a la tierra y finaliza cuando el dispositivo GPS utiliza la información recibida para determinar su localización la cual se interpreta mayormente en dos conjuntos: la latitud y longitud[?].

4.2. Estado del Arte

4.2.1. *Driver Drowsiness Detection Using Machine Learning with Visual Behaviour*

Este trabajo de investigación realizado por el *International Journal of Creative Research Thoughts* propone un sistema de detección de signos de somnolencia en conductores utilizando un modelo de Red Neuronal Convolutacional para detectar la posición de los ojos, y OpenCV junto con Dlib para la detección de la boca y realizar el conteo del número de bostezos por minuto. Para alimnetnar a la red Neuronal, se utilizaron el conjunto de datos de NTHU-DDD. También fue utilizado el método PERCLOS para obtener el número de parpadeos del sujeto de estudio. Este trabajo concluye que las mayores dificultades a la hora de detección de rostros fueron el uso de gafas oscuras, así como cambios en la iluminación.

4.2.2. *Detection Of Drowsiness And Distraction Of Drivers Using CNN*

En este trabajo realizado por los estudiantes de la Universidad de Mumbai, se implementó el aprendizaje automático y el paquete Keras para construir un modelo de CNN, el cual, clasifica si el conductor se encuentra somnoliento o distraído, el sistema emite un tono de alerta al detectar correctamente la somnolencia, dando al conductor una alerta temprana. Se utilizó el clasificador Open CV Haar-Cascade, un clasificador en cascada basado en características, usando sus funciones integradas, se detetó el rostro y la región de los ojos.

4.2.3. *Driver Drowsiness Detection System Using Convolutional Neural Networks*

En este trabajo realizado en la Universidad Anurag en el año 2022 se presenta una forma de analizar y anticipar la somnolencia del conductor mediante la aplicación de una red neuronal convolucional sobre la cara del conductor de un marco de secuencia. Se uso un conjunto de datos para dar forma y aprobar el modelo, usando redes convolucionales 3D basadas en modelos de múltiples capas de arquitectura de red neuronal repetitiva para detectar la somnolencia del conductor. Tras una sesión de entrenamiento, se obtuvo una precisión que se acerca al 92 % de aceptación.

4.2.4. **Diseño e implementación de sistema de visión artificial para alerta y detección de somnolencia mediante aprendizaje profundo aplicable en conductores de vehículos**

En este trabajo realizado en la facultad de Ingeniería de la Universidad Nacional de Trujillo, se desarrolló un sistema de extracción de características faciales tales como pestañeo, cabeceo y bostezos. Para la extracción de regiones de interés se utilizaron cascadas Haar, y la clasificación de estas características se realizó utilizando un modelo de red LeNet. Además el trabajo incluye la creación de una base de datos de las regiones de interés de la cara utilizando imágenes propias y también utilizando conjunto de datos externos. Para el desarrollo de este trabajo se utilizó el lenguaje de programación Python, junto con las librerías de OpenCV, Tensorflow y Keras.

4.2.5. Sistema de detección de somnolencia mediante inteligencia artificial en conductores de vehículos para alertar la ocurrencia de accidentes de tránsito

Este proyecto fue realizado por estudiantes de la Escuela Profesional de Ingeniería de Perú y consistió en llevar a cabo un sistema para la detección de la somnolencia y la distracción del conductor. El sistema se desarrolló utilizando C# con EmguCV para detectar la distracción y orientación de los ojos utilizando técnicas de visión artificial. Además, cuenta con un sistema de alarma compuesto por un zumbador de 12v, que se activa a recibir la orden el microcontrolador al procesar el sistema de visión artificial junto a una red neuronal.[12]

4.2.6. Diseño de un sistema electrónico para detectar la somnolencia en automovilistas por medio de la actividad ocular

Este trabajo realizado por alumnos de la ESIME Culhuacán en el año 2019 detalla el diseño y la implementación de un sistema que se basa en la fusión de dos señales. Una de ellas proviene de la detección del estado de los ojos utilizando información proveniente de una cámara digital. Para lo anterior, se realiza una segmentación de las regiones de la piel y posteriormente se obtiene la ubicación y el rastro de los ojos. La segunda señal se obtiene a partir de los datos proveniente de un acelerómetro colocado sobre la cabeza del conductor, cuya función es detectar los cabeceos asociados con somnolencia. El procesamiento de dicha señal del acelerómetro se lleva a cabo con la ayuda de la Transformada Wavelet Discreta. Estas dos señales son correlacionadas para tener como salida dos alarmas secuenciales que son percibidas por el conductor. La primera le alerta sobre un primer estado de posible somnolencia y la segunda acciona un control difuso para el control momentáneo del auto y la corrección del volante para el seguimiento del carril. Los resultados obtenidos por este trabajo demuestran una eficacia para la detección de ojos cerrados del 86 % y para la detección de cabeceo superior al 90 %.

4.2.7. Sistema para la detección del estado de somnolencia en seres humanos, con reconocimiento de patrones

En este artículo se muestra la implementación de un sistema de detección del estado de somnolencia en seres humanos, a través de la identificación de patrones faciales y la frecuencia de parpadeo de los ojos. Utilizando técnicas de inteligencia artificial, visión por computadora y un sistema embebido con cámara integrada para la adquisición de imágenes. El cual permite detectar en tiempo real el estado de fatiga de un conductor automovilístico y su grado de somnolencia, todo con el objetivo de disminuir la tasa de accidentes viales causados precisamente por la somnolencia en México. Se hizo uso del lenguaje de programación Python, bibliotecas como OpenCV, Dlib y Scipy, las cuales, fueron requeridas debido a los modelos predefinidos que establecen una mayor precisión en la detección de puntos faciales específicos, utilizando como referencia el método de predicción de 68 puntos específicos del rostro. El sistema propuesto tiene la característica de funcionar con luz de día en una primera etapa, y la idea es poder implementarlo en cualquier tipo de vehículo automotriz a un costo accesible a la mayoría de los propietarios de vehículos automotrices.

4.2.8. Sistema de Detección de Somnolencia

En este trabajo de fin de grado, realizado por un alumno de la Universidad de La Laguna en el año 2022 se estudió el uso de los modelos de Redes Neuronales para la clasificación de imágenes.

Enfocado en la resolución del problema de la somnolencia en los conductores. En el cual se utilizó el lenguaje de programación Python, junto con diversas librerías que facilitan la integración del modelo, y otras que ayudan en la captura de las imágenes.

4.2.9. Desarrollar un prototipo de reconocimiento facial basado en Machine Learning para detectar estado de Somnolencia en conductores de una cooperativa de transporte.

En este trabajo, realizado por alumnos de la Universidad de GUAYAQUIL en el año 2020-2021 se plantea el desarrollo un prototipo para la detección de la somnolencia del conductor de una cooperativa de transporte el cual dicho conductor se había sobrepasado el límite de horas de trabajo, usando la técnica de reconocimiento facial, basándose específicamente en el estado de los ojos. Para ello se realizó una investigación bibliográfica relacionada con patrones biométricos, inteligencia artificial y programación mediante Machine Learning, así como, las principales variables que permiten identificar un estado de somnolencia. Dentro de esta investigación también se determinan cuáles son los algoritmos a utilizar (CV2, Imutils, etc.) siendo la herramienta Python y su librería principal Visión por Computadora las seleccionadas para el estudio. [16]

4.2.10. Sistema basado en la detección y notificación de somnolencia en conductores de autos

En este trabajo, realizado por alumnos de la Universidad de Córdoba en el año 2015 se plantea el desarrollo que permite alertar a los conductores en estado de somnolencia leve, utilizando la tecnología de reconocimiento de objetos de Kinect y la librería OpenCV con el lenguaje C# para el reconocimiento de imágenes. Además, se diseña una aplicación móvil del sistema operativo Android para la notificación de somnolencia utilizando una conexión socket tipo TCP. Para el procesamiento de imágenes, se utilizó el algoritmo Viola-Jones, el cual se basa en una nueva forma de representación de imágenes llamada ?Integral Image?, permitiendo que las características del detector se conmuten rápidamente. [13]

4.2.11. Detección de somnolencia para conducción sin accidentes

En este trabajo de fin de grado, realizado por un alumno de la Universidad de Alicante en el año 2022 se propuso crear una aplicación real para detectar la somnolencia al volante haciendo énfasis a un bajo coste económico. Donde se realizó una comparativa entre una solución mediante Machine Learning (ML) y Principal Component Analysis (PCA).

4.2.12. Implementación de un sistema web con geolocalización para medir la transaccionalidad del portal ABC de la democracia para el consejo nacional electoral (CNE)

En este proyecto de titulación de la Universidad de Guayaquil del año 2016, se implementa un sistema de geolocalización para poder tener una mejor visión a través de mapas virtuales de las personas y de las zonas geográficas de los productos o servicios que se quieran introducir, apoyado en un cien por ciento de las herramientas que Google Maps ofrece, se utilizó una aplicación popular y gratuita para poder analizar los datos obtenidos con la localización, Google Analytics. En este proyecto, sólo se pretendía visualizar mapas en la web, por lo que sólo fue necesario usar los

servicios que ofrece Google Maps por ser una aplicación gratuita con la ayuda del api Javascript de Google Maps, se tuvieron en cuenta las limitaciones con respecto al código libre, pero éstas no afectaron el desempeño del sistema; los datos obtenidos se recopilaban en una base de datos en MySQL.

4.2.13. Diseño e implementación de un sistema de geolocalización en interiores para plataforma Android vía la red enterprise WLAN de la PUCP

En este proyecto de titulación de la Pontificia Universidad Católica Del Perú del 2016 se desarrolló una aplicación móvil capaz de geolocalizar a un usuario dentro de las instalaciones de la universidad usando la técnica de Huellas de Señal (fingerprinting), que minimiza el error debido a reflexiones y obstáculos, basada en el estimador de máxima verosimilitud (ML por sus siglas en inglés Maximum Likelihood) junto a las mediciones de señal de los Access Points cercanos usando la red Wi-Fi. La tecnología de radiofrecuencia que se usó fue la de redes inalámbricas de área local, Wi-Fi. Que ofrece conectividad por radiofrecuencia, con alcance local a un dispositivo que envíe datos Ethernet desde la ubicación del mismo hasta una conexión a la red fija, que en este caso la universidad contaba con 32 access points que recibirían la señal de datos a través de cobre o fibra. La técnica del fingerprinting, que está dirigida a geolocalización en interiores, consiste en un mapeo de datos que se encuentran en un escenario para luego asociarlos a una localización y almacenarlos en una base de datos, para estimar la localización más probable se utilizó el algoritmo de ML basado en el teorema de Bayes de probabilidad. En las conclusiones señalan que este sistema obtuvo una precisión del 100 % en la estimación del ambiente con un error menor de 2.4m en las pruebas realizadas.

4.2.14. Geolocalización con LoRa mediante multilateración

En esta tesis de la Universidad de la República de Uruguay del 2018 se desarrolló un prototipo para la geolocalización animal en estudios de comportamiento. Para la geolocalización se usó una tecnología nueva emergente llamada LoRa, tecnología de radiofrecuencia de bajo consumo energético y largo alcance; usando un método de multilateración para la geolocalización, técnica de navegación basada en la medición de la diferencia de distancia a dos estaciones en posiciones conocidas, el proyecto se conformó por tres puntos fijos o access points y nodos, que son los dispositivos llevados por los animales de radio frecuencia. Los nodos envían señales de radio frecuencia cada cierto tiempo, la señal emitida se recibe por los puntos fijos que, a su vez, esa señal genera metadatos que se almacenan en una base de datos para posteriormente realizar los cálculos en Geolocator , sistema de geolocalización desarrollado en Python, y así, obtener la posición geográfica del nodo. Se concluyó que el uso de LoRa para la geolocalización tiene gran potencial y que el alcance de las señales fue eficiente, de igual manera se trata de una tecnología de bajo consumo y que mantuvo los dispositivos con energía por meses.

4.2.15. Propuesta de un sistema de geolocalización y monitoreo vía GPS/GSM/GPRS aplicado a un pulsómetro para personas con enfermedades cardiovasculares

Tesis del Instituto Politécnico Nacional de la Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Zacatenco del año 2018, donde se empleó un sensor de pulso cardiaco el cual proporciona información en tiempo real de los latidos del corazón mientras que un microcontrolador procesa los datos y, en caso de que se obtengan los datos de que se está presentando una taquicardia, el

microcontrolador solicita la ubicación al módulo GPS y envía un mensaje de texto a través de GSM/GPRS a la persona designada.

4.2.16. Influencia de un sistema de geolocalización en el control y monitoreo de vehículos con dispositivos GPS en una empresa logística

Tesis del año 2015 de la Universidad César Vallejo de Perú donde se investiga de manera profunda las características y detalles de la tecnología GPS para determinar la influencia de un sistema de geolocalización en el control y monitoreo de vehículos con esta tecnología en una empresa logística.

4.2.17. Diseño de la red de acceso LTE en el distrito de Jesús María

Tesis del año 2017 de la Pontifica Universidad Católica del Perú que tiene por objetivo analizar y diseñar una red usando la tecnología LTE, al desarrollar el proyecto se conocen las ventajas y desventajas de usar esta red así como la posibilidad de implementarla, también se realizan los cálculos de las coberturas y se dan a conocer las velocidades de enlace y el uso de diversos anchos de banda para su despliegue.

5. Capítulo III: Análisis

5.1. Análisis de la Zona Geográfica

El sistema está dirigido principalmente para empresas cuya actividad esté enfocada al transporte de material o personas. Ya que una parte fundamental del proyecto es la portabilidad, el sistema requiere de un servicio de acceso al internet. Debido a lo anterior, se decidió utilizar redes móviles celulares para cumplir lo anterior.



Figura 7: Mapa de cobertura 3G / 4G / 5G, México. [5]

Tomando en cuenta la información del Mapa de cobertura 3G/4G/5G, de redes móviles en México, mostrado en la figura 2, se observa que las zonas con mayor cobertura son principalmente las que cuentan con mayor densidad de población, por tanto, se plantea que su funcionamiento sea principalmente en la Ciudad de México, debido a que cuenta con mayor infraestructura en dichas redes (figura 3).

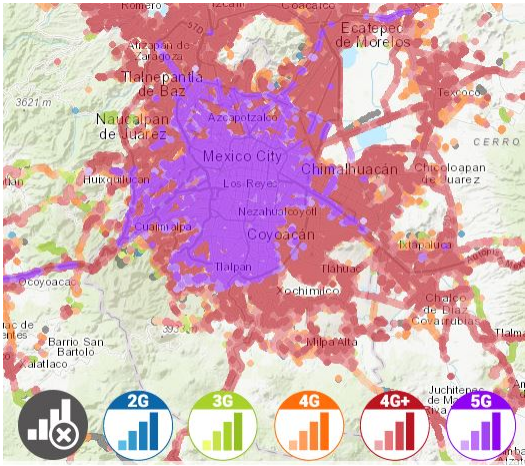


Figura 8: Mapa de cobertura 3G / 4G / 5G, Ciudad de México. [5]

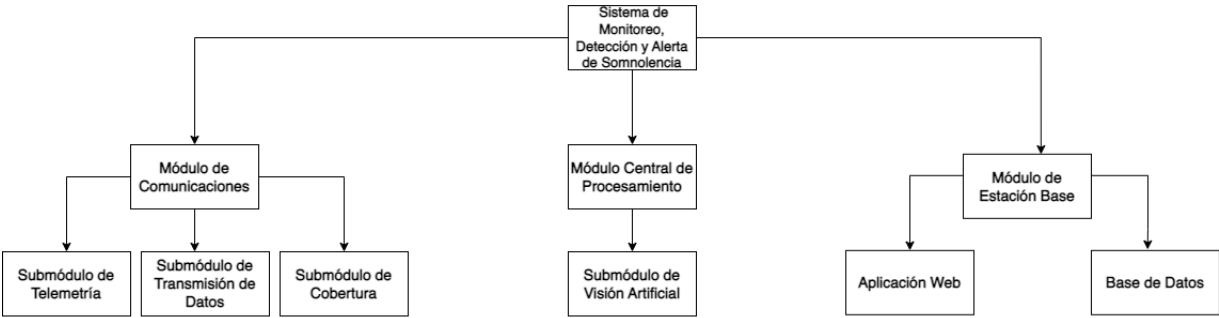


Figura 9: Boosting

5.2. Análisis del Módulo Central de Procesamiento

ID	Nombre corto del Requerimiento
<i>RF01</i>	<i>Verificar estado funcional de periféricos</i>
Descripción: El sistema verificará el estado funcional de los distintos periféricos utilizados en el Módulo Central de Procesamiento	
Elementos: <ul style="list-style-type: none">• Raspberry Pi• Cámara Digital• Zumbador Pasivo• Antena Sixfab	
Solución del Requerimiento: Utilizando métodos de comprobación ofrecidos por cada uno de las librerías de los periféricos, se comprobará que funcionen de manera correcta, en caso contrario se enviará una alerta al módulo de Estación Base.	

Tabla 1: RF01- Verificar estado funcional de periféricos

ID	Nombre corto del Requerimiento
<i>RF02</i>	<i>Activar Alarma</i>
Descripción: El sistema activará el zumbador en caso de que el Submódulo de	
Elementos: <ul style="list-style-type: none">• Raspberry Pi• Zumbador Pasivo	
Solución del Requerimiento: Utilizando métodos de comprobación ofrecidos por cada uno de las librerías de los periféricos, se comprobará que funcionen de manera correcta, en caso contrario se enviará una alerta al módulo de Estación Base.	

Tabla 2: RF02- Activar Alarma

p

5.2.1. Análisis del Submódulo Visión Artificial

■ Requerimientos Funcionales

ID	Nombre corto del Requerimiento
<i>RF01</i>	<i>Capturar video en tiempo real</i>
Descripción: El sistema capturará video en todo momento	
Elementos: <ul style="list-style-type: none">● Conductor● Incidencias● Administrador● MongoDB	
Solución del Requerimiento: Con el uso de una cámara digital, se capturará video en tiempo real.	

Tabla 3: RF01- Capturar Video en Tiempo Real

ID	Nombre corto del Requerimiento
<i>RF02</i>	<i>Delimitar una región de Interés</i>
Descripción: El sistema capturará video en todo momento	
Elementos: <ul style="list-style-type: none">● OpenCV● Cascadas Haar	
Solución del Requerimiento: Utilizando el software de OpenCV y .	

Tabla 4: RF02- Delimitar región de Interés

5.2.2. Análisis y Elección del Modelo de Red Neuronal

Debido a que el presente proyecto se plantea resolver un problema de clasificación, a continuación se analizará que tipos de modelos de redes neuronales son los más aptos para la resolución de los mismos.

- **Redes Neuronales Unicapa** En el caso mas simple de una red neuronal, las neuronas de la capa de entrada se limitan a recibir las señales de entrada. Estas se encargan de redistribuir estas señales a las neuronas de la capa de salida. A esto se le llama una red neuronal unicapa.

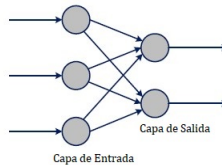


Figura 10: Redes Unicapa

Utilizando al herramienta de *PlayGround Tensorflow* podemos observar la eficacia de este tipo de redes al resolver problemas de clasificación. Si se le pide a esta Red Neuronal que sombree el área del color de los puntos mostrados dentro de la gráfica, se puede observar que es incapaz de realizarlo de manera correcta.

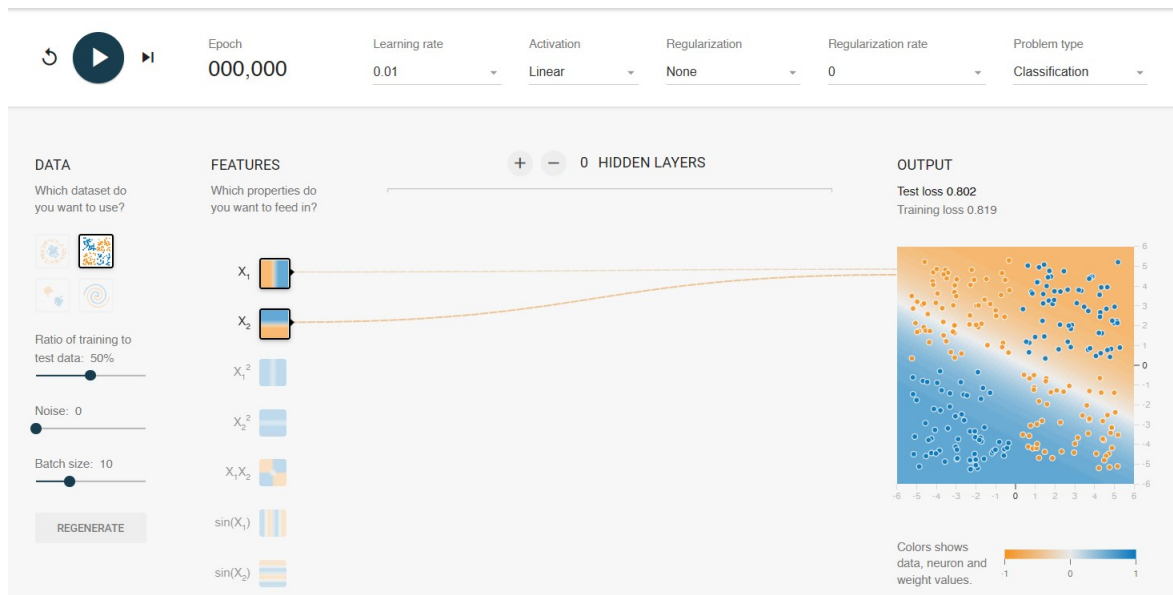


Figura 11: Redes Unicapa

- **Redes Neuronales Multicapa**

Existe otro tipo de modelo de red neuronal que utiliza capas intermedias entre la capa de salida y la capa de entrada, a estas redes se les conoce como redes profundas o redes multicapa.

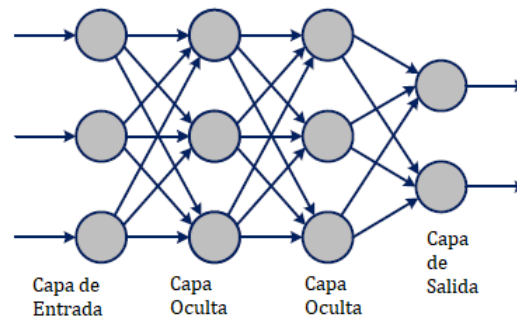


Figura 12: Redes Unicapa

En la siguiente figura se puede apreciar que las redes neuronales multicapa tienen una eficiencia mayor a la hora de realizar el mismo problema de clasificación de puntos. Sin embargo, aún no puede resolver este problema correctamente.

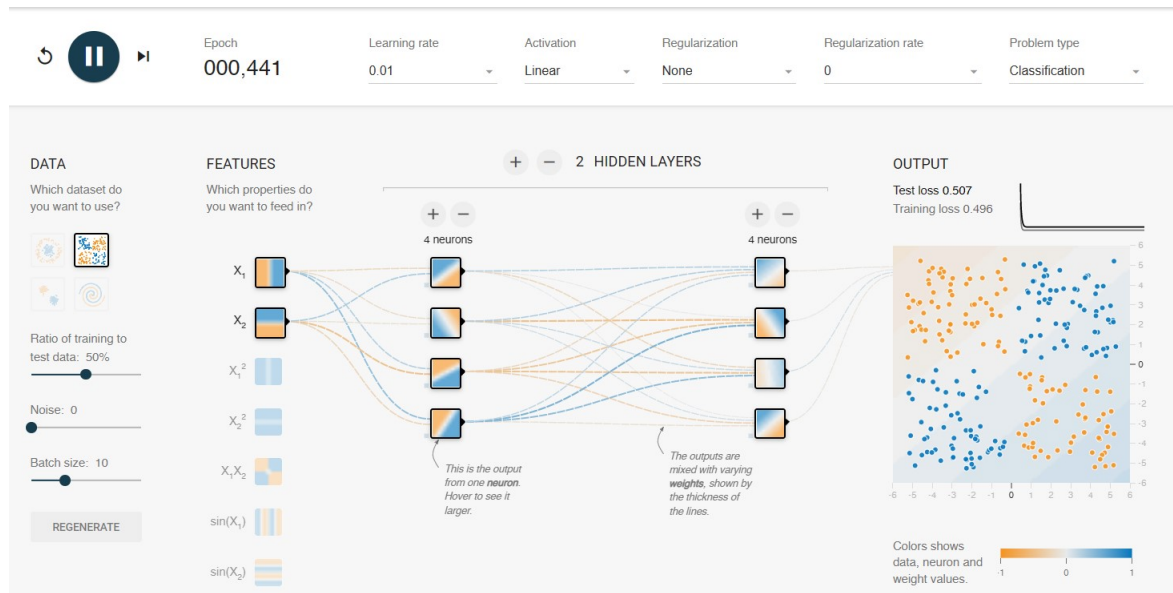


Figura 13: Redes Neuronal con 2 Capas Ocultas de 4 Neuronas cada una

■ Funciones de Activación

Una función de activación es un filtro o umbral que modifica o impone límites en los valores resultantes de una capa en una red neuronal. A continuación se muestran los resultados al resolver el mismo problema de clasificación de puntos de colores utilizando distintas funciones de activación.

- **ReLU** Esta función de activación consiste en anular los valores de entrada negativos y conservar los valores positivos con sus valores originales.

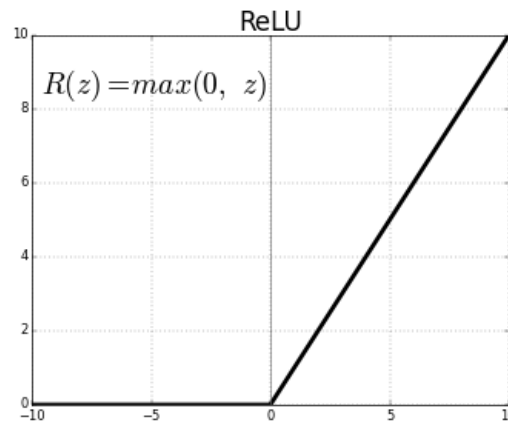


Figura 14: Representación Gráfica de la Función ReLu

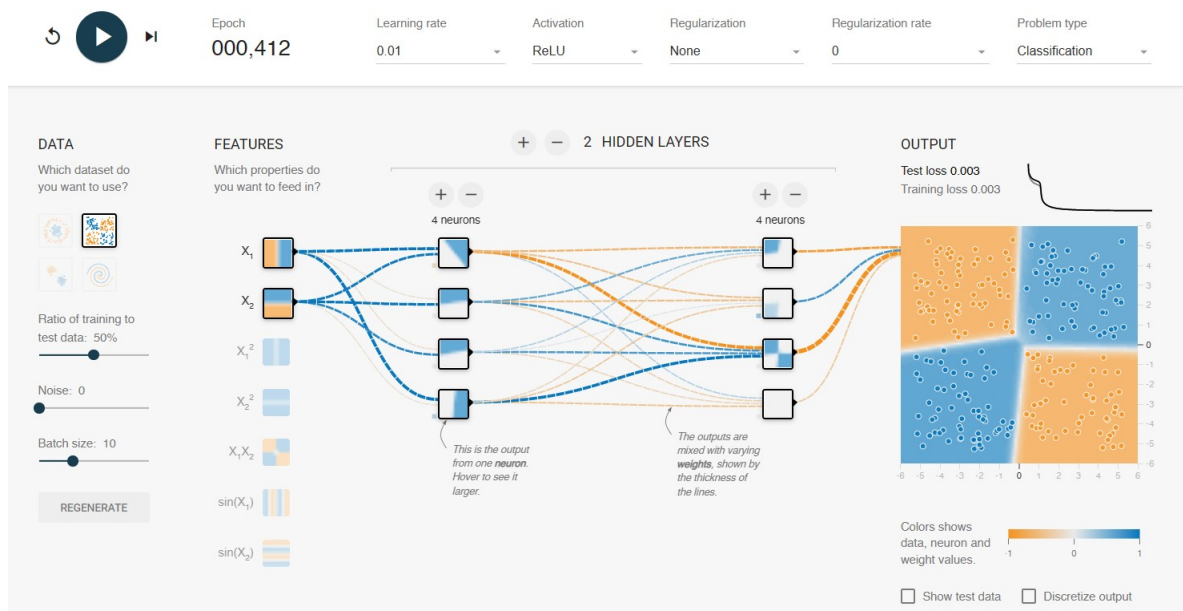


Figura 15: Redes Neuronal con 2 Capas Ocultas de 4 Neuronas cada una utilizando la función de Activación ReLu

- **Tanh** La función de la tangente hiperbólica toma cualquier valor real como valores de entrada y de salida en el rango de -1 a 1. Cuanto mayor sea la entrada, más cerca estará del valor de salida de 1.0. Por otro lado, mientras sea menor sea la entrada, el valor de salida se aproximará mas a -1.0

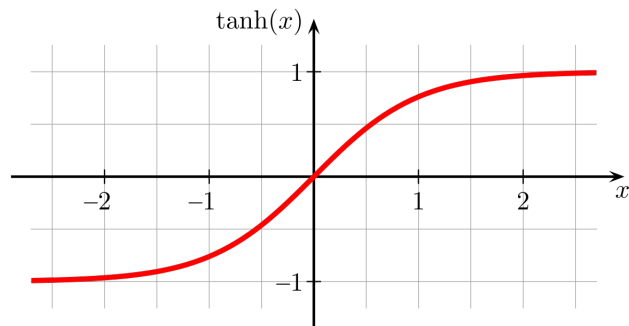


Figura 16: Representación gráfica de la función Tangente Hiperbólica

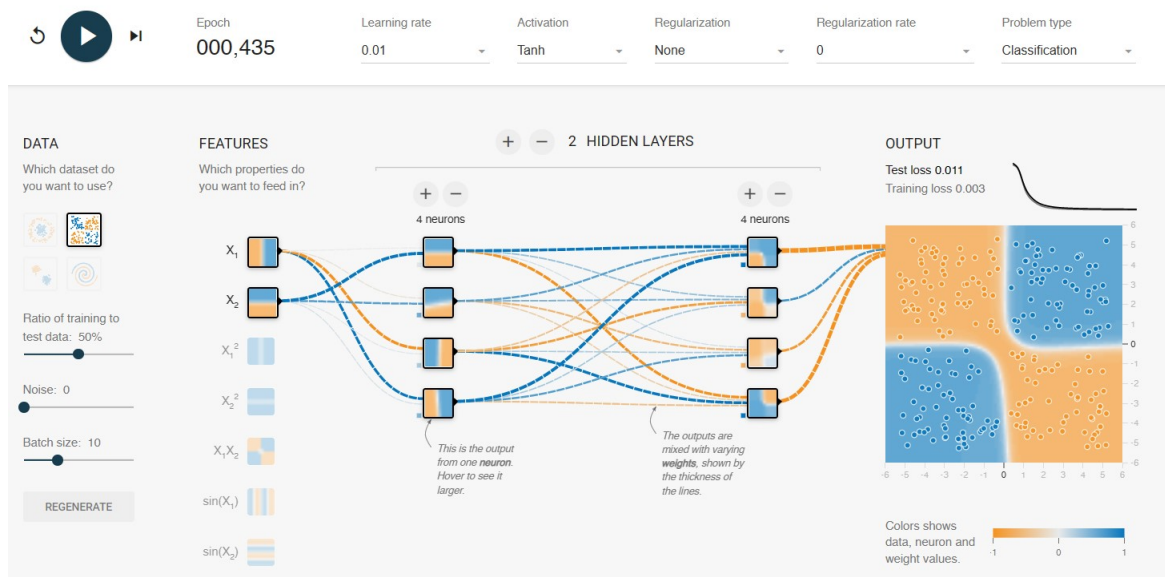


Figura 17: Redes Neuronal con 2 Capas Ocultas de 4 Neuronas cada una utilizando la función de Activación Tanh

5.3. Análisis del Módulo de Comunicaciones

5.4. Análisis del Módulo de Estación Base

El módulo de la estación base tiene como objetivo que el usuario administrador, visualice y confirme el estado de los reportes de incidencia que se hayan presentado por parte del conductor, por tal motivo, se realizará una aplicación web, la cual se conectará a una base de datos NoSQL. En ella se guardarán los reportes de incidencias y se podrán visualizar por medio de la aplicación web. Cabe aclarar que el video de la incidencia se almacenará en la nube, ya que al ser contenido multimedia no se puede guardar en la base de datos, únicamente se guardará el URL para acceder al video. Las credenciales del usuario serán almacenadas en Amazon Cognito, con la cuales podrán iniciar sesión dentro de la Aplicación Web, la cual estará alojada en un servidor web.

El sistema contará con los siguientes requerimientos:

Requerimientos Funcionales del Módulo de Estación Base

ID	Nombre corto del Requerimiento
<i>RF01</i>	<i>Guardar Incidencia</i>
Descripción: La base de datos almacenará la información de cada reporte de incidencia que se envíe desde el Módulo Central de Procesamiento.	
Elementos: <ul style="list-style-type: none">• Id del Conductor• Nombre de Conductor• Apellidos de Conductor• Número de Incidencias• Fecha• Hora• Estado de la Incidencia• Ubicación• URL del video	
Solución del Requerimiento: El Módulo de Procesamiento Central puede insertar datos en la base datos.	

Tabla 5: RF01- Guardar Incidencia

ID	Nombre corto del Requerimiento
<i>RF02</i>	<i>Guardar Video</i>
Descripción: La base de datos almacenará la información de cada reporte de incidencia que se envíe desde el Módulo Central de Procesamiento.	
Elementos: <ul style="list-style-type: none">• Video de incidencia	
Solución del Requerimiento: El Módulo de Procesamiento Central puede insertar el video en el almacenamiento de objetos, mientras que la base de datos obtiene la URL del video guardado.	

Tabla 6: RF02- Guardar Video

ID	Nombre corto del Requerimiento
<i>RF03</i>	<i>Conectar Aplicación Web</i>
Descripción: La base de datos almacenará la información de cada reporte de incidencia que se envíe desde el Módulo Central de Procesamiento.	
Elementos: <ul style="list-style-type: none">• Node.js• React• Express• MongoDB	
Solución del Requerimiento: El usuario administrador puede realizar la inserción, modificación, eliminación y consulta de datos desde la aplicación web.	

Tabla 7: RF03- Conectar Aplicación Web

ID	Nombre corto del Requerimiento
<i>RF04</i>	<i>Desplegar Aplicación Web</i>
Descripción: La aplicación web se alojará en una red de entrega de contenido (CDN), disponible con una URL.	
Elementos: <ul style="list-style-type: none">• AWS Amplify• GitHub• Amazon S3	
Solución del Requerimiento: El usuario administrador puede acceder a la interfaz de la aplicación web, haciendo uso de la URL en el navegador web.	

Tabla 8: RF04- Desplegar Aplicación Web

ID	Nombre corto del Requerimiento
<i>RF05</i>	<i>Guardar Credenciales de Usuario Administrador</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Elementos: <ul style="list-style-type: none">• Amazon Cognito	
Solución del Requerimiento: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 9: RF05- Guardar Credenciales de Usuario Administrador

Requerimientos No Funcionales del Módulo de Estación Base

ID	Nombre del requerimiento	Descripción
<i>RNF01</i>	Disponibilidad	La disponibilidad del sistema será continua, el usuario podrá acceder a la información las 24 horas del día.
<i>RNF02</i>	Interoperabilidad	El sistema será capaz de intercambiar información con el Módulo Central de Procesamiento a través del Módulo de Telemetría.
<i>RNF03</i>	Seguridad	Las credenciales de acceso al sistema podrán ser cambiadas únicamente por el administrador.
<i>RNF04</i>	Usabilidad	El sistema estará enfocado a la visualización de reportes, por lo que este tendrá una interfaz intuitiva y amigable para el usuario. El sistema proporcionará mensajes de advertencia orientados al usuario, en caso de ocurrir un error en el Módulo Central de Procesamiento.

Tabla 10: Requerimientos No Funcionales del Módulo de Estación Base

Análisis de Casos de Uso

Con base en los requerimientos se realizó el siguiente diagrama de casos de uso, el cual muestra las actividades y la interacción con el Módulo de Estación base y el Módulo Central de Procesamiento.

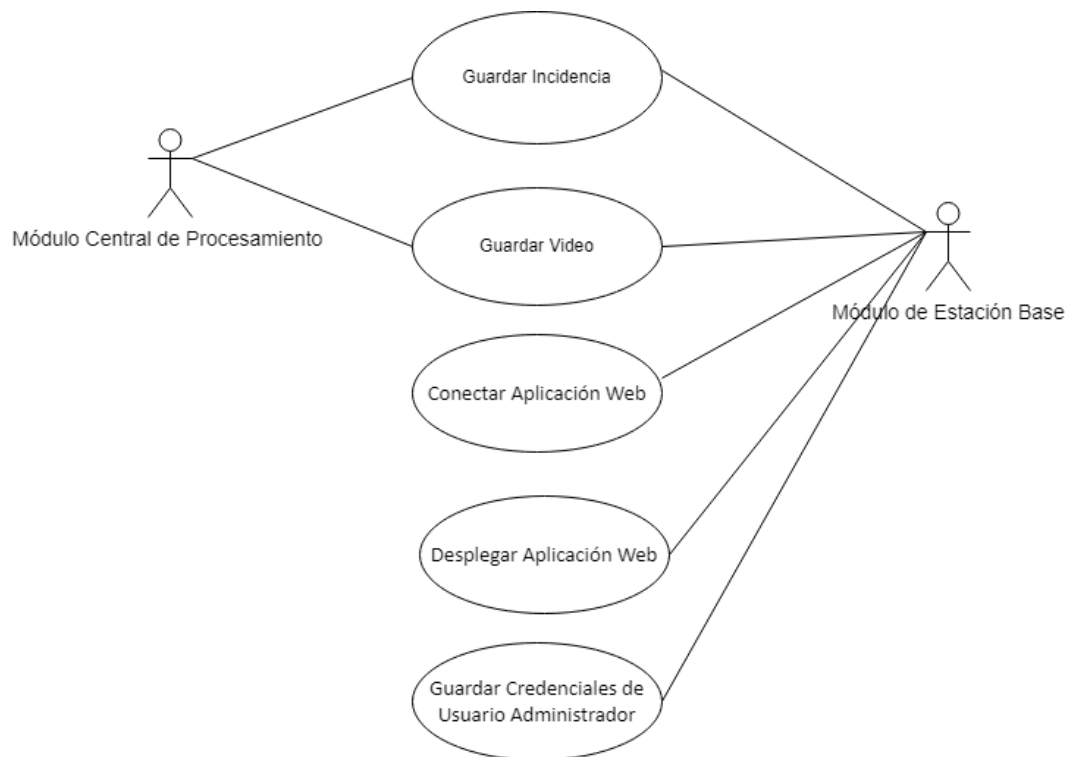


Figura 18: Diagrama de Casos de Usos del Módulo de Estación Base

Requerimientos No Funcionales del Módulo de Estación Base

Nombre del Caso de Uso	<i>Guardar Incidencia</i>
Actores	<i>Módulo Central de Procesamiento, Módulo de Estación Base</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador podrá iniciar sesión en la aplicación web.	

Tabla 11: Caso de Uso 01

Nombre del Caso de Uso	<i>Guardar Video</i>
Actores	<i>Módulo Central de Procesamiento, Módulo de Estación Base</i>
Descripción: Se almacenarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Haber Iniciado Sesión Previamente	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 12: Caso de Uso 02

Nombre del Caso de Uso	<i>Conectar Aplicación Web</i>
Actores	<i>Módulo de Estación Base</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 13: Caso de Uso 03

Nombre del Caso de Uso	<i>Desplegar Aplicación Web</i>
Actores	<i>Módulo Central de Procesamiento, Módulo de Estación Base</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 14: Caso de Uso 04

Nombre del Caso de Uso	<i>Guardar Credenciales de Usuario Administrador</i>
Actores	<i>Módulo Central de Procesamiento</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 15: Caso de Uso 05

5.4.1. Análisis de la Aplicación Web

■ Análisis de Requerimientos Funcionales

ID	Nombre corto del Requerimiento
<i>RF01</i>	<i>Iniciar Sesión</i>
Descripción: El sistema permitirá iniciar sesión en la aplicación web	
Elementos: <ul style="list-style-type: none">• Amazon Cognito• Administrador• Credenciales• MongoDB	
Solución del Requerimiento: <p>El sistema, con la ayuda de los servicios de Amazon Cognito, comprobará que las credenciales ingresadas por parte del usuario administrador se encuentren en la base de datos. En caso contrario, la aplicación web indicará que ese usuario no se encuentra registrado en la base de datos.</p>	

Tabla 16: RF01- Iniciar Sesión

ID	Nombre corto del Requerimiento
<i>RF02</i>	<i>Mostrar el Historial de Reportes de Incidencia</i>
Descripción: El sistema desplegará en forma de lista todas las incidencias que se tengan registradas en la base de datos	
Elementos: <ul style="list-style-type: none">• Conductor• Incidencias• Administrador• MongoDB	
Solución del Requerimiento: <p>El sistema recuperará de la base de datos todas las incidencias que se tengan registradas.</p>	

Tabla 17: RF02- Mostrar el Historial de Reportes de Incidencia

ID	Nombre corto del Requerimiento
<i>RF03</i>	<i>Visualizar Reporte de Incidencia</i>
Descripción: El sistema desplegará los detalles específicos de cada incidencia registrada.	
Elementos: <ul style="list-style-type: none">• Incidencia• Administrador• Conductor	
Solución del Requerimiento: <p>El usuario administrador podrá visualizar los reportes individuales de incidencias de cada conductor al hacer click en cualquiera de las incidencias mostrada en la lista principal. Los reportes contendrán información sobre la fecha, hora, ubicación y un video corto del momento en que fueron detectados síntomas de somnolencia.</p>	

Tabla 18: RF02- Mostrar el Historial de Reportes de Incidencia

ID	Nombre corto del Requerimiento
<i>RF03</i>	<i>Visualizar Reporte de Incidencia</i>
Descripción: El sistema desplegará los detalles específicos de cada incidencia registrada.	
Elementos: <ul style="list-style-type: none">• Incidencia• Administrador• Conductor	
Solución del Requerimiento: <p>El usuario administrador podrá visualizar los reportes individuales de incidencias de cada conductor al hacer click en cualquiera de las incidencias mostrada en la lista principal. Los reportes contendrán información sobre la fecha, hora, ubicación y un video corto del momento en que fueron detectados síntomas de somnolencia.</p>	

Tabla 19: RF03- Visualizar Reporte de Incidencia

ID	Nombre corto del Requerimiento
<i>RF04</i>	<i>Confirmar Incidencia</i>
Descripción: El sistema permitirá al administrador confirmar la incidencia, esto para descartar que se trate de un falso positivo.	
Elementos: <ul style="list-style-type: none">• Administrador• Incidencia• Conductor• MongoDB	
Solución del Requerimiento: <p>El usuario administrador podrá confirmar la incidencia después de haber revisado el videoclip del momento de somnolencia con la intención de descartar falsos positivos. Esto será posible ingresando a una incidencia específica mostrando sus detalles.</p>	

Tabla 20: RF04- Confirmar Incidencia

ID	Nombre corto del Requerimiento
<i>RF05</i>	<i>Recuperar Contraseña</i>
Descripción: El sistema contará con una opción para recuperar la contraseña del administrador en caso de que sea olvidada la contraseña.	
Elementos: <ul style="list-style-type: none">• Amazon Cognito• Email• Administrador• MongoDB	
Solución del Requerimiento: <p>El sistema requerirá que el usuario administrador ingrese el correo con el que fue registrado. Posteriormente se le enviará un código de recuperación de contraseña a ese correo. El administrador ingresará se código en el apartado de recuperar contraseña y así podrá ingresar una nueva contraseña.</p>	

Tabla 21: RF05 - Recuperar Contraseña

ID	Nombre corto del Requerimiento
<i>RF06</i>	<i>Mostrar Perfil Conductor</i>
Descripción: El sistema permitirá al usuario administrador visualizar los datos de cada conductor registrado en la base de datos.	
Elementos: <ul style="list-style-type: none">• Perfil• Conductor	
Solución del Requerimiento: <p>El administrador podrá consultar cada uno de los perfiles de los conductores registrados en la base de datos dando click en el nombre del mismo. En dicho perfil se mostrarán datos como nombre, apellido, así como el número de incidencias de dicho conductor, con sus respectivos detalles.</p>	

Tabla 22: RF06- Mostrar perfil del Conductor

ID	Nombre corto del Requerimiento
<i>RF07</i>	<i>Mostrar Ubicación Geográfica</i>
Descripción: La aplicación web, con ayuda de los datos proporcionados por el módulo de telemetría, mostrará la ubicación en tiempo real de un conductor.	
Elementos: <ul style="list-style-type: none">• Ubicación en Tiempo Real• Módulo de Telemetría	
Solución del Requerimiento: <p>El sistema permitirá al administrador consultar la ubicación de los conductores en tiempo real. Para esto el administrador deberá de ingresar previamente al perfil del conductor del cual desea consultar dicha ubicación.</p>	

Tabla 23: RF07- Mostrar ubicación Geográfica

ID	Nombre corto del Requerimiento
<i>RF08</i>	<i>Descartar Incidencia</i>
Descripción: El sistema permitirá catalogar una incidencia como Falsa si fuera el caso.	
Elementos: <ul style="list-style-type: none">• Incidencia• Administrador	
Solución del Requerimiento: En caso de presentarse una incidencia falsa, después de haber sido revisada por el administrador, esta podrá ser catalogada como falsa incidencia y será eliminada automáticamente de la base de datos.	

Tabla 24: RF08- Descartar Incidencia

ID	Nombre corto del Requerimiento
<i>RF09</i>	<i>Registrar Usuario</i>
Descripción: El sistema le permitirá al administrador registrar nuevos conductores.	
Elementos: <ul style="list-style-type: none">• Incidencia• Administrador	
Solución del Requerimiento: En caso de presentarse una incidencia falsa, después de haber sido revisada por el administrador, esta podrá ser catalogada como falsa incidencia y será eliminada automáticamente de la base de datos.	

Tabla 25: RF09- Registrar Usuario

ID	Nombre corto del Requerimiento
<i>RF10</i>	<i>Modificar Usuario</i>
Descripción: El sistema le permitirá al administrador modificar los datos del conductor.	
Elementos: <ul style="list-style-type: none">• Nombre• Apellido• Conductor• Administrador	
Solución del Requerimiento: El administrador podrá editar los datos de los conductores como nombre o apellido.	

Tabla 26: RF10- Mostrar el Historial de Reportes de Incidencia

ID	Nombre corto del Requerimiento
<i>RF11</i>	<i>Eliminar Usuario</i>
Descripción: El sistema le permitirá al administrador eliminar los datos del conductor. Elementos: <ul style="list-style-type: none">• Conductor• Administrador	
Solución del Requerimiento: El sistema le permitirá al eliminar los datos de conductores de la base de datos.	

Tabla 27: RF11- Mostrar el Historial de Reportes de Incidencia

ID	Nombre corto del Requerimiento
<i>RF12</i>	<i>Generar Mensaje de Alerta</i>
Descripción: El sistema alertará con un mensaje en caso de que algún componente en alguna Unidad Contenedora no funcione correctamente. Elementos: <ul style="list-style-type: none">• Módulo Central de Procesamiento• Administrador• Alerta	
Solución del Requerimiento: El sistema enviará un mensaje de alerta al administrador indicando que Unidad presenta algún desperfecto, esto con la intención de sea reparada posteriormente.	

Tabla 28: RF12- Generar Mensaje de Alerta

■ Análisis de Requerimientos no Funcionales

A continuación se mencionan los requerimientos no funcionales de la aplicación web.

Requerimiento No Funcional	Descripción
Disponibilidad	La aplicación web deberá estar disponible las 24 horas del día
Interoperabilidad	La aplicación deberá ser accesible desde cualquier sistema operativo mientras se cuenta con un navegador web en su versión más actual
Eficiencia	La aplicación tendrá tiempos de respuesta menores a 100ms.
Escalabilidad	La aplicación web deberá ofrecer la oportunidad de agregar nuevas funcionalidades y soportar un mayor número de usuarios a futuro
Estabilidad	La aplicación deberá ofrecer un buen funcionamiento mientras se tenga una conexión a internet mínima de 1Mbps

Tabla 29: Requerimientos No funcionales - Aplicación Web

■ Análisis de Casos de Uso

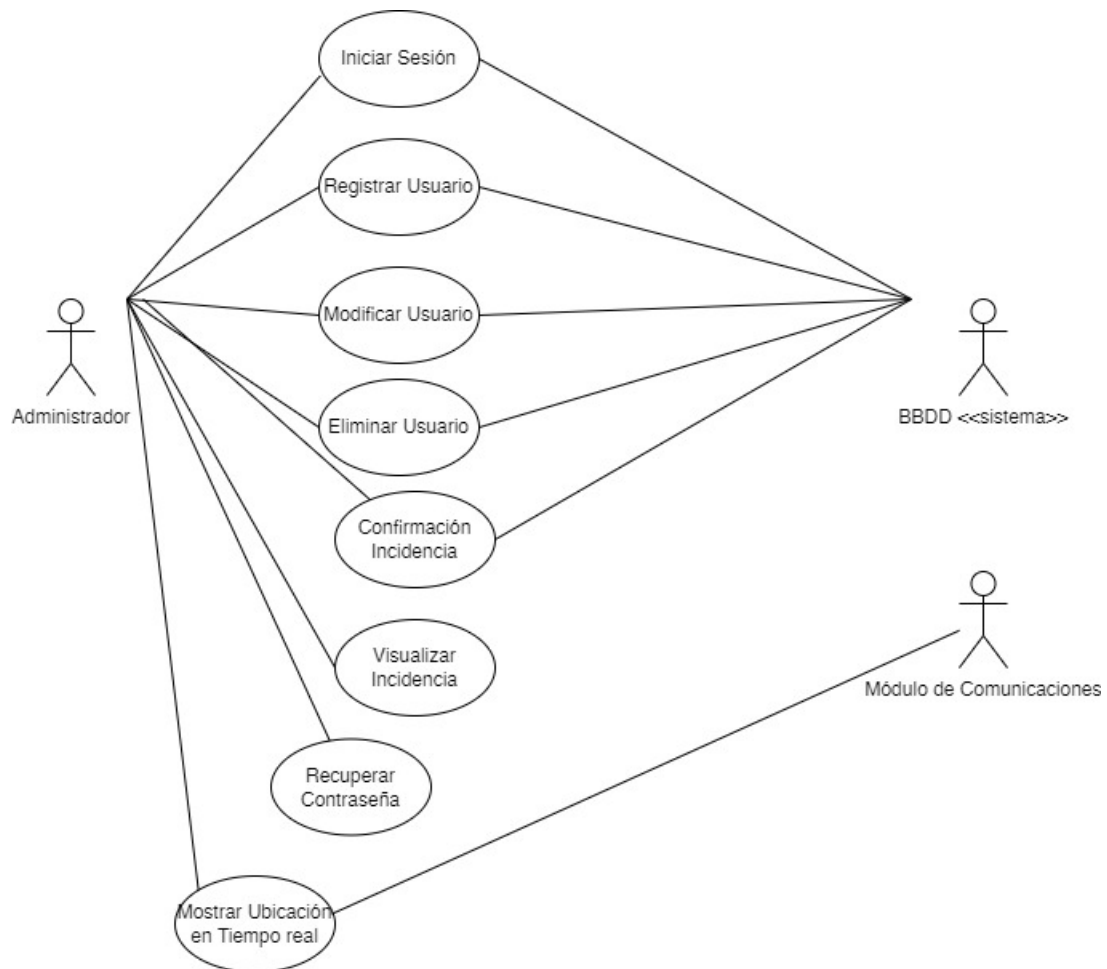


Figura 19: Diagrama de Casos de Uso

Nombre del Caso de Uso	<i>Iniciar Sesión</i>
Actores	<i>Administrador, BBDD</i>
Descripción: El administrador ingresará los datos correspondientes para iniciar sesión en la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• El administrador deberá tener sus credenciales registradas en Amazon Cognito	
Entradas: <ul style="list-style-type: none">• Email• Contraseña	

Tabla 30: Caso de Uso 01

Nombre del Caso de Uso	<i>Registrar Usuario</i>
Actores	<i>Administrador, BBDD</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
Entradas: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 31: Caso de Uso 02

Nombre del Caso de Uso	<i>Modificar Usuario</i>
Actores	<i>Administrador, BBDD</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 32: Caso de Uso 03

Nombre del Caso de Uso	<i>Eliminar Usuario</i>
Actores	<i>Administrador, BBDD</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 33: Caso de Uso 04

Nombre del Caso de Uso	<i>Eliminar Usuario</i>
Actores	<i>Administrador, BBDD</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 34: Caso de Uso 05

Nombre del Caso de Uso	<i>Confirmar Incidencia</i>
Actores	<i>Administrador, BBDD</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 35: Caso de Uso 06

Nombre del Caso de Uso	<i>Visualizar Incidencia</i>
Actores	<i>Administrador, BBDD</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 36: Caso de Uso 07

Nombre del Caso de Uso	<i>Recuperar Contraseña</i>
Actores	<i>Administrador, BBDD</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 37: Caso de Uso 08

Nombre del Caso de Uso	<i>Mostrar Ubicación en Tiempo Real</i>
Actores	<i>Administrador Módulo de Comunicaciones</i>
Descripción: Se guardarán únicamente las credenciales de los usuarios administradores que podrán acceder a la aplicación web.	
Precondiciones: <ul style="list-style-type: none">• Amazon Cognito	
PostCondiciones: El usuario administrador puede iniciar sesión en la aplicación web.	

Tabla 38: Caso de Uso 08

6. Capítulo IV: Diseño

6.1. Diseño del Módulo Central de Procesamiento

■ Elección del Lenguaje de Programación

Debido a que el presente proyecto posee un enfoque orientado a la inteligencia artificial, se debe de tener en claro que esta difiere al paradigma de la programación convencional. En la programación convencional, o estructurada, el usuario le indica a la máquina los datos y las reglas que debe seguir, y la máquina se encarga de obtener los resultados los parámetros mencionados. Sin embargo, en el paradigma de la programación orientada a inteligencia artificial, este enfoque cambia. Ahora se le indican los datos y los resultados, para obtener finalmente las reglas con las que se llegan a dichos resultados.

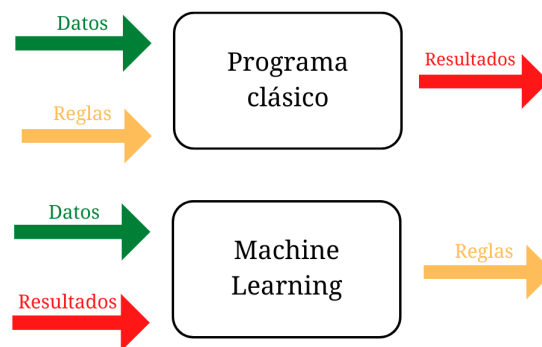


Figura 20: Programa clásico vs Machine Learning

■ Arquitectura

Cómo se mencionó anteriormente en la sección de metodología, a lo largo de este proyecto se hará uso de la metodología *Top-Down*, en dónde uno de sus principios fundamentales radica en la modularidad. Esta no es más que la habilidad de poder dividir programas en partes más pequeñas o sub-programas llamados módulos. Estos módulos deben posser cierto grado de independencia de la aplicación principal, y a su vez poder ser utilizados por otras aplicaciones.

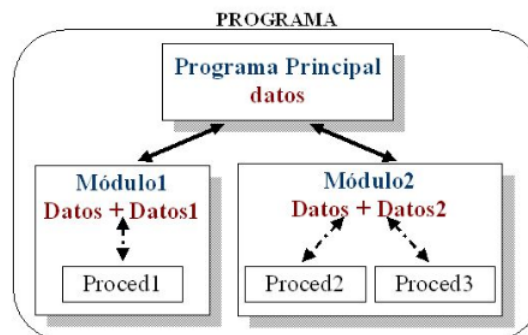


Figura 21: Modularidad

Python es un lenguaje de programación que permite trabajar con más de un paradigma(programación estructurada y orientada a objetos).[\[15\]](#). Normalmente cuando se programa un script sencillo, se aplica el paradigma de programación estructurada. Sin embargo, al tratarse de un problema más complejo, se suele utilizar el paradigma orientado a objetos.

6.1.1. Diseño de la Unidad Contenedora de Procesamiento

Para el diseño de la unidad contenedora del módulo de procesamiento, se tomaron en cuenta los elementos físicos que estarán dentro de la unidad y sus respectivas medidas. Cabe mencionar que los elementos que respectan al modelo del ordenador, el modelo de la cámara, el modelo del zumbador y el modelo de la microSD fueron previamente seleccionados en base a los requerimientos del sistema.

Elementos físicos que contendrá la unidad:

- Raspberry Pi 4 Modelo B

De acuerdo con las especificaciones físicas de la Raspberry Pi 4 Modelo B se tiene las siguientes medidas en milímetros:

Dimensiones 85 x 53mm.

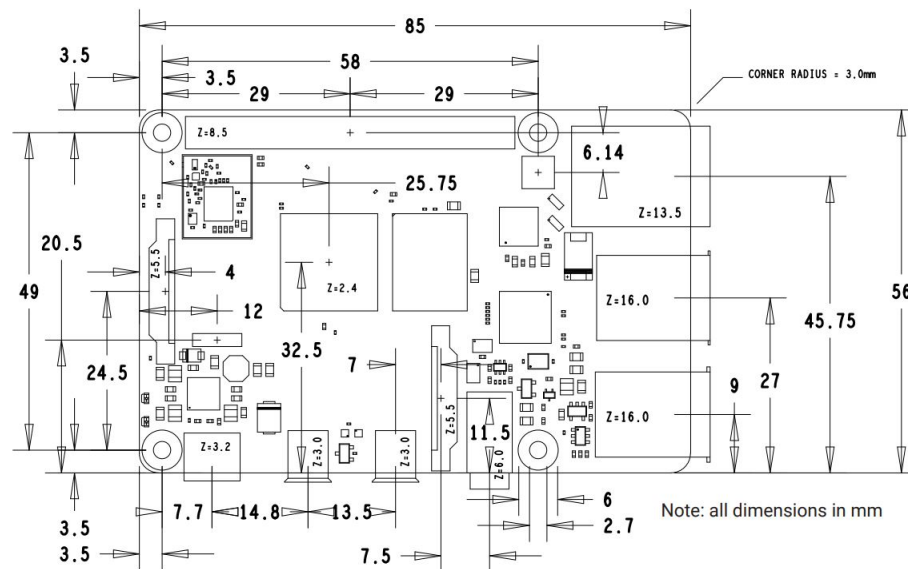


Figura 22: Especificaciones Físicas

- Zumbador Pasivo KY-006

Dimensiones 18 x 15mm.

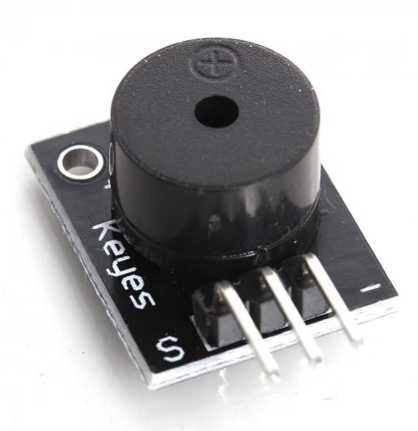


Figura 23: Zumbador Pasivo KY-006

- Cámara Arducam Pivariety IMX230
Dimensiones de la placa: 38 x 38mm.



Figura 24: Cámara Pivariety IMX230

- Cable macho-hembra



Figura 25: Jumpers

Se utilizarán 2 cables macho-hembra para la conexión del zumbador pasivo hacia los pines GND y Vcc del Raspberry Pi 4. Largo 10 cm.

- Micro SD
Dimensiones: 15 x 11 x 1 mm.



Figura 26: Micro SD

- RaspberryPi 3G/4G/ LTE Base HAT



Figura 27: RaspberryPi 3G/4G/ LTE Base HAT

Este HAT celular proporciona una conexión de datos simplificada para proyectos de IoT remotos, en el campo, en todo el mundo, en todas partes. Comience a usar una conexión LTE de alta velocidad con bajo consumo de energía, además de contar con todo el software necesario para desarrollar en una Raspberry Pi.

- LE 910Cx Mini PCIe Series Linux

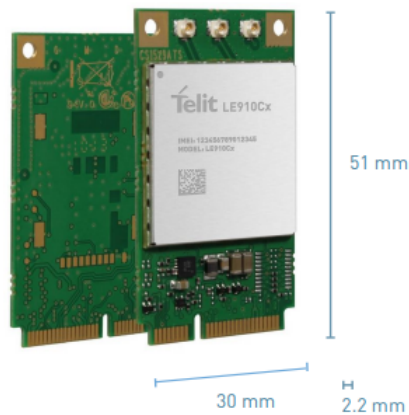


Figura 28: LE 910Cx Mini PCIe Series Linux

Los módulos Mini PCIe (mPCIe) de la serie LE910Cx son optimizadas para redes LTE de baja categoría y están disponibles en modo único de LTE o con opciones de respaldo 3G/2G.

- **LTE Main Diversity GNSS Triple Port u.FL Antenna de 100mm**



Figura 29: LTE Main Diversity GNSS Triple Port u.FL Antenna de 100mm

Antenas LTE principal y Diversity combinadas con una antena GNSS en una sola antena formada. Ajuste directo para el módulo Mini PCIe LTE de Quectel EC25, Telit LE910C1 y Telit LE910C4.

Tomando en cuenta las medidas anteriormente mencionadas, se presenta un bosquejo del diseño modular para la unidad contenedora, cabe aclarar que la siguiente propuesta puede presentar modificaciones durante el desarrollo del proyecto, con el fin de cumplir con las necesidades del sistema y presentar un diseño acorde a los cambios.

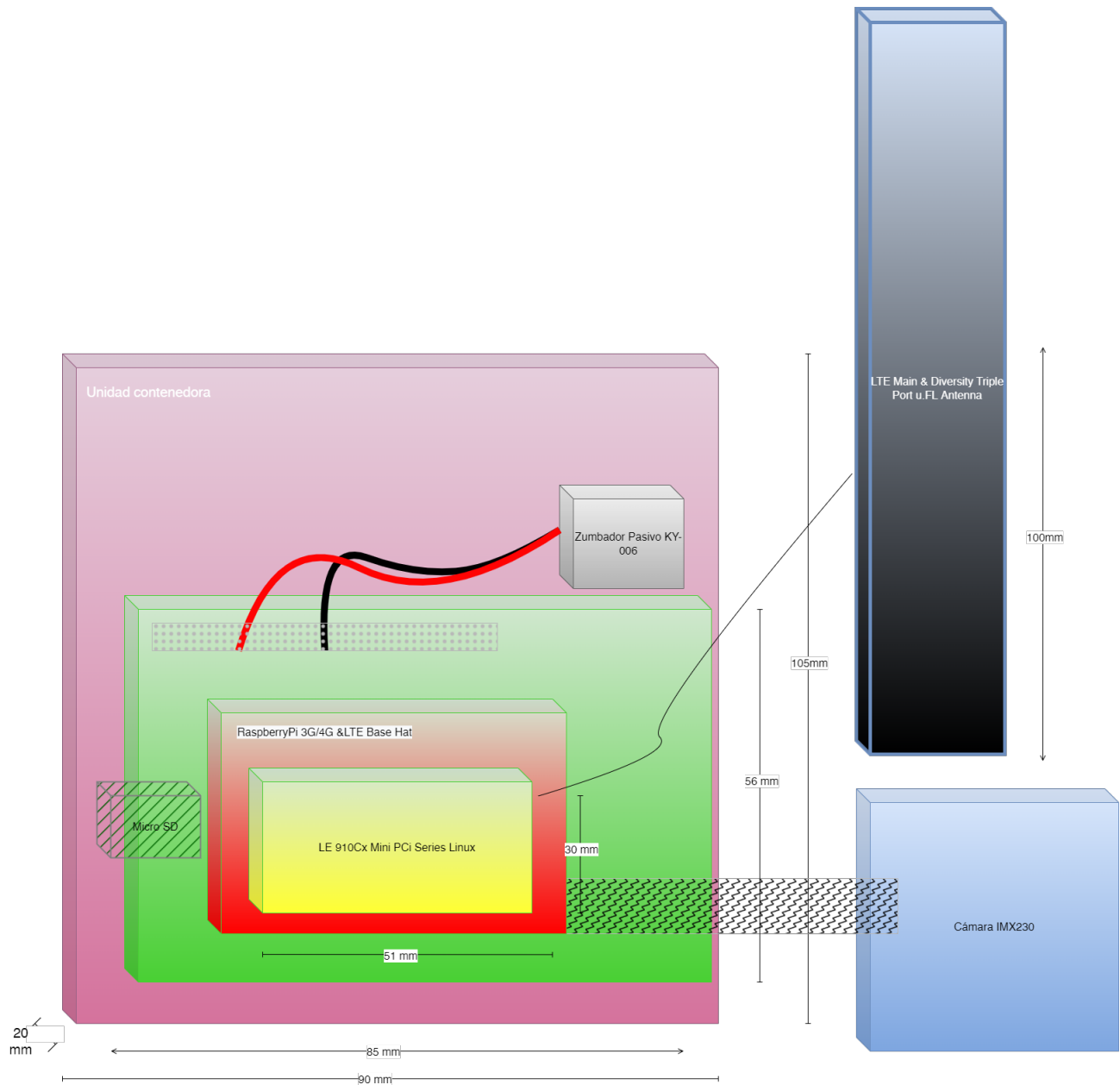


Figura 30: Unidad Contenedora del Módulo Central de Procesamiento

6.1.2. Diseño del Submódulo de Visión Artificial

■ Puntos de Referencia

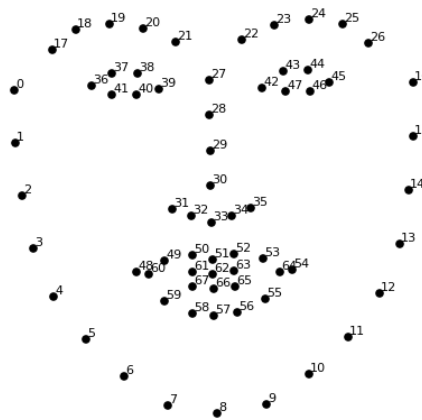


Figura 31: Puntos de referencia

A su vez, se utilizarán puntos de referencia con la ayuda de el software OpenCV y el modelo iBUG 300-W de delimitación de rostro para dividir las características que posteriormente serán extraídas y posteriormente clasificadas.

En la siguiente tabla se muestra la delimitación y agrupación de los puntos de interés, tales como: ojo derecho, ojo izquierdo y boca.

Partes	Puntos de Referencia
Boca	[48-67]
Ojo Derecho	[36-41]
Ojo Izquierdo	[42-46]

Tabla 39: Agrupación de Puntos de Referencia

■ Mouth Opening Ratio

Bostezar es un signo cansancio. El bostezo se caracteriza por una apertura lenta y amplia de la boca. Por lo que se necesita medir el tamaño y la forma de la boca es necesario para identificar un bostezo. Para esto, se utilizara el *Mouth Opening Ratio* que es un método que utiliza puntos de referencia para medir la apertura de la boca. Entre más grande sea este valor, más es la aperatura de la boca, por lo tanto cumple con las características de un bostezo.

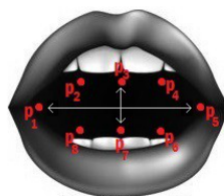


Figura 32: *Mouth Opening Ratio*

La formula general para calular el *MOR* es la siguiente:

$$MOR = \frac{\|p_2 - p_8\| + \|p_3 - p_7\| + \|p_4 - p_6\|}{2 \|p_1 - p_5\|} \quad (6)$$

Utilizando nuestros puntos de referencia, se podrá calcular el MOR de la siguiente manera:

$$MOR = \frac{\|P49 - P59\| + \|P51 - P57\| + \|P53 - P55\|}{2 \|P48 - P54\|} \quad (7)$$

6.2. Diseño del Módulo de Comunicaciones

6.2.1. Diseño del Submódulo de Transmisión de Datos

6.2.2. Diseño del Submódulo de Telemetría

6.2.3. Diseño del Submódulo de Cobertura

6.3. Diseño de la Estación Base

Con base en los requerimientos, se plantea realizar el diseño de la arquitectura del Módulo de Estación Base, integrando el Módulo de Procesamiento Central. El cual hará énfasis en la organización y comunicación de los elementos que lo conforman.

El siguiente diagrama muestra la arquitectura del sistema de Estación base, el cual se compone de las interacciones de los elementos que permitirán al usuario acceder e interactuar con la aplicación web. El diagrama también muestra la participación del Módulo de Procesamiento Central ya que el registro de incidencias y el envío del video de incidencia serán realizados por dicho módulo. La información de los reportes de incidencias y el registro de los conductores serán almacenados en Mongo DB, mientras que los videos de incidencia se almacenarán en Amazon S3. Las credenciales de los Usuarios Administradores que tendrán permitido acceder a la aplicación se almacenarán en Amazon Cognito. Para la aplicación web, el *backend* será desarrollado en node.js mientras que el *frontend* será desarrollado en React, posteriormente los archivos se almacenarán en un repositorio dentro de Github. La aplicación web será alojada y desplegada desde AWS Amplify, lo cual le permitirá al cliente acceder a la aplicación desde una URL.

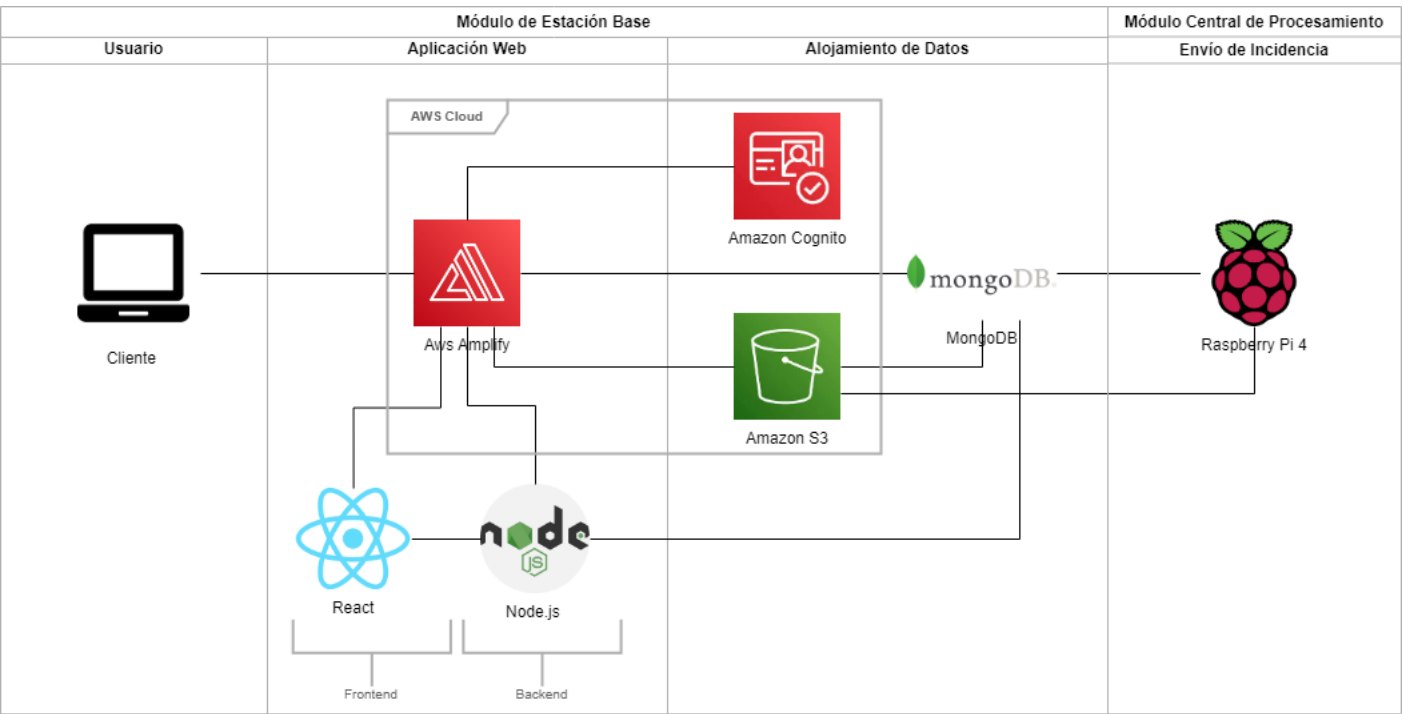


Figura 33: Arquitectura del Módulo de Estación Base

Elección del servidor de alojamiento Después de haber analizado los distintos proveedores de alojamiento así como sus características, llegamos a la conclusión que el que mejor cumple con los requisitos del proyecto es AWS Amplify. Ya que este contiene una suite de herramientas que facilitan la implementación de una aplicación web, como por ejemplo S3, que es un servicio de alojamiento en la nube utilizado para archivos multimedia de gran tamaño. [11]

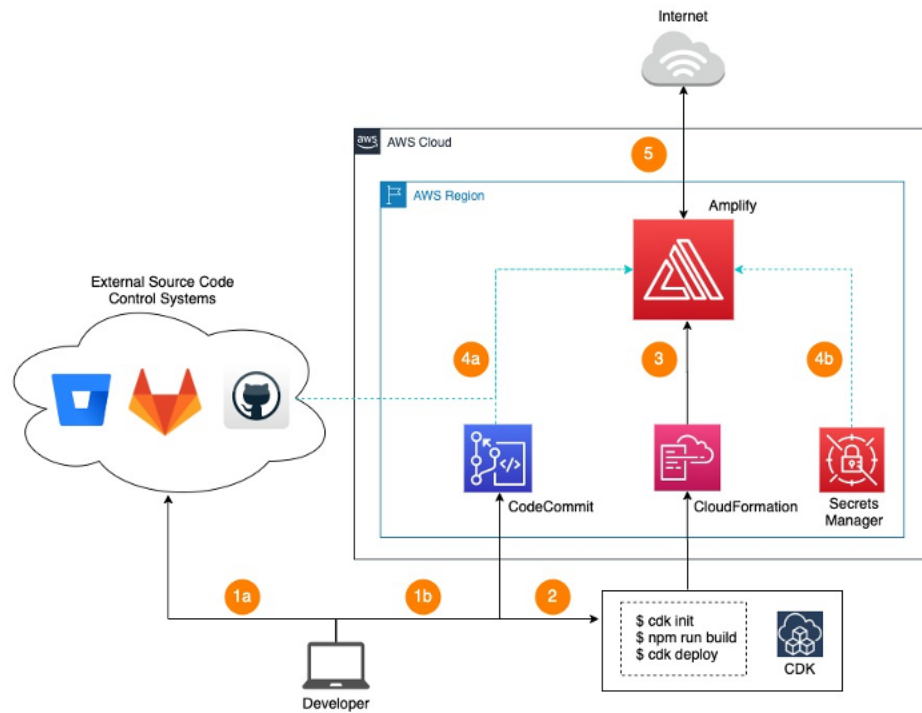


Figura 34: Diagrama de Componentes AWS Amplify

Otro factor fundamental por el cual se decidió utilizar la suite de AWS Amplify, es que esta cuenta con su propia CDN, lo cual significa que los tiempos de respuesta al usar la aplicación web deben de ser mínimos.

6.3.1. Diseño de la Aplicación Web

Para el diseño y desarrollo de la Aplicación web, se decidió utilizar el lenguaje Javascript, ya que este cuenta con las herramientas y/o frameworks que mejor se adaptan a los requisitos previamente definidos. A su vez, se decidió utilizar el NodeJs para el sistema backend.

Node.js, es un entorno en tiempo de ejecución multiplataforma basado en JavaScript. Este es controlado por eventos, permitiendo establecer y gestionar múltiples conexiones al mismo tiempo. Gracias a esta característica, el bloqueo de procesos no existe. NodeJs trabaja fundamentalmente bajo dos características: *asincronía*, que permite la ejecución de varios procesos al mismo tiempo, y *Entrada/salida sin bloqueo*, que significa poder trabajar con múltiples solicitudes sin bloquear un hilo para una sola solicitud. [9]

NodeJs es capaz de manejar distintas peticiones sin que tener que esperar a que una petición sea respondida para continuar con la siguiente petición. De ahí la elección de este entorno de Javascript para el desarrollo de la aplicación web.

A continuación se muestran diversos diagramas de secuencias que describen los procesos de los requerimientos definidos en la sección de Análisis.

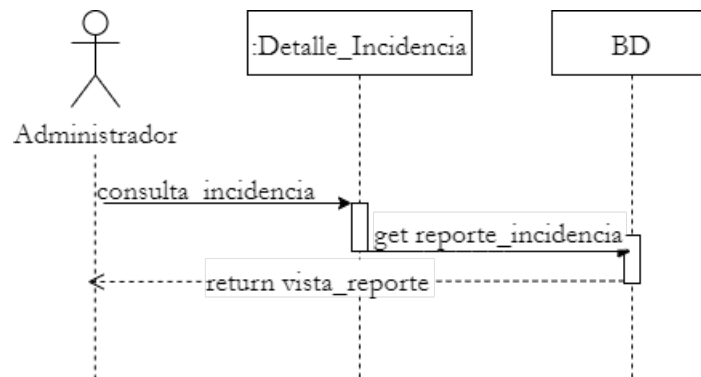


Figura 35: Diagrama de Secuencia Detalle Reporte Incidencia

El Administrador podrá consultar el reporte de incidencia de cada uno de los conductores. Accediendo a la base de de datos de MongoDB

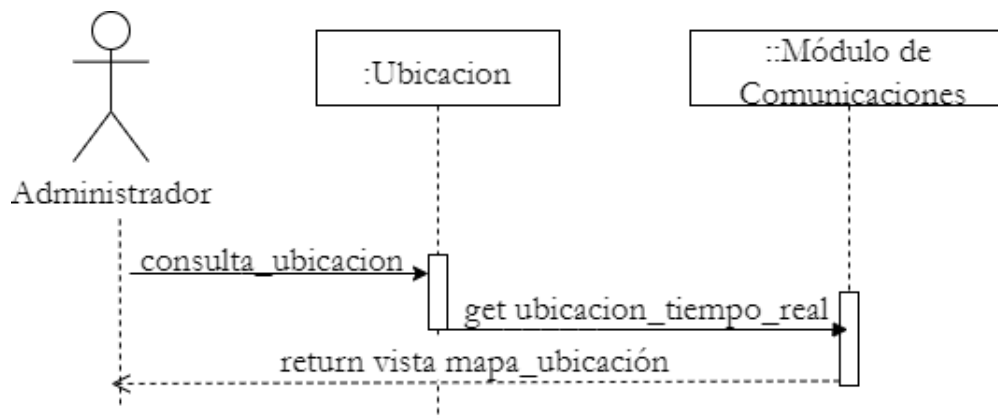


Figura 36: Diagrama de Secuencia Consultar Ubicación

De igual manera, el Administrador podrá consultar la ubicación en tiempo real del conductor.

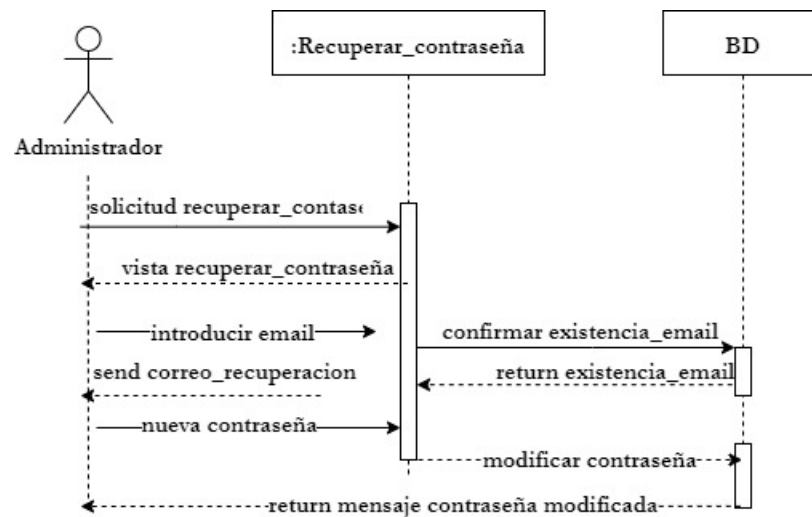


Figura 37: Diagrama de Secuencia Recuperar Contraseña

También podrá recuperar su contraseña en caso de que esta sea olvidada.

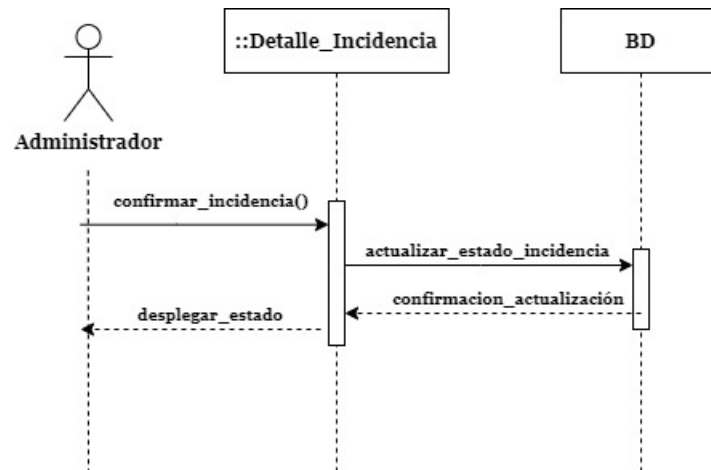


Figura 38: Diagrama de Secuencia Confirmar Incidencia

Para evitar falsos positivos, el Administrador podrá confirmar una incidencia una vez mirando el videoclip del incidente.

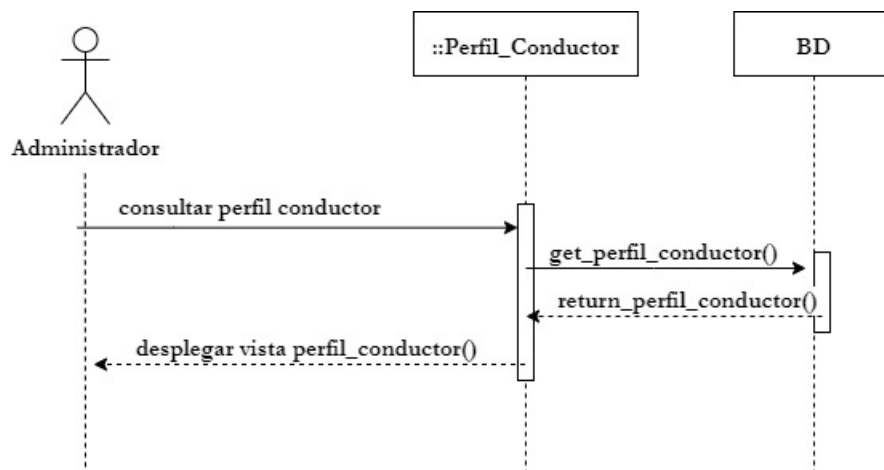


Figura 39: Diagrama de Secuencia Consultar Perfil

El Administrador podrá consultar el perfil de cada conductor.

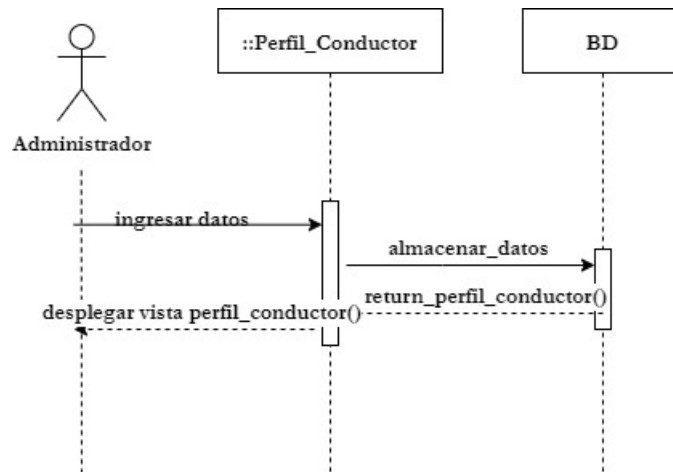


Figura 40: Diagrama de Secuencia Registrar Conductor

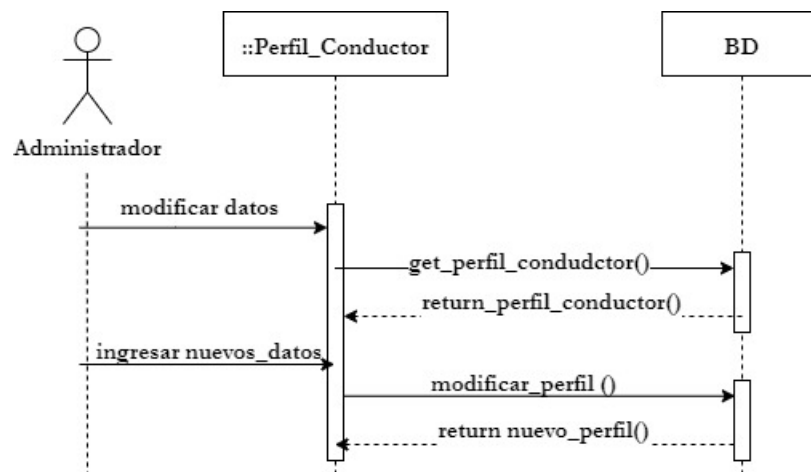


Figura 41: Diagrama de Secuencia Modificar Conductor

En caso de que los datos del conductor sean incorrectos, el administrador podrá modificarlos.

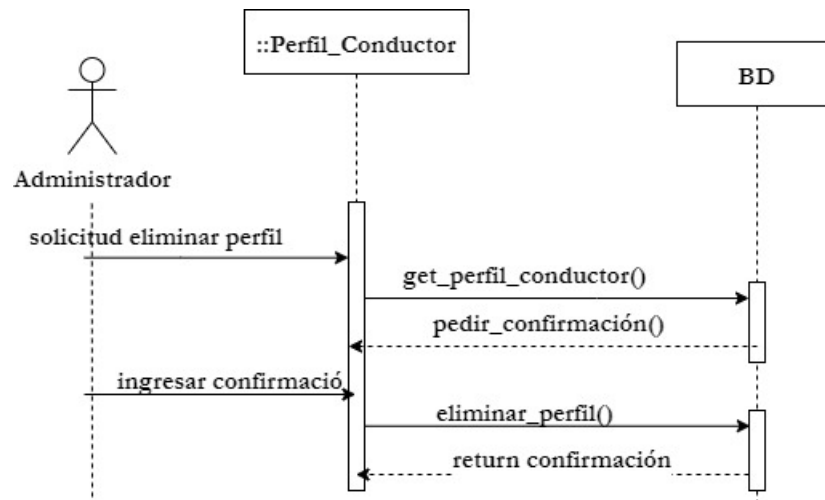


Figura 42: Diagrama de Secuencia Eliminar Conductor

El Administrador también tendrá la opción de dar de baja a un conductor.

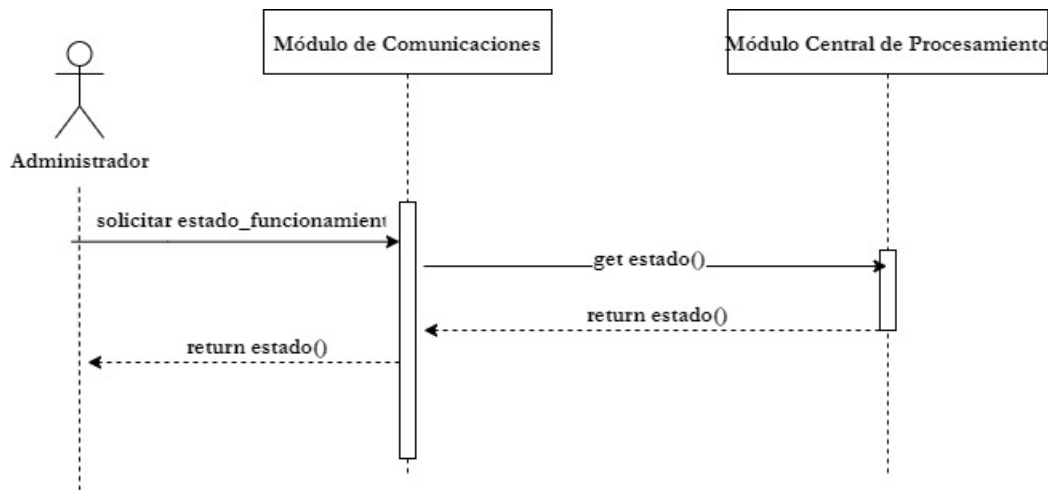


Figura 43: Diagrama de Secuencia Estado de los Periféricos

La aplicación web indicará si alguno de los periféricos no se encuentra en correcto funcionamiento, lanzando una alerta indicando el número de Unidad y a qué conductor pertenece. A continuación se muestra el diseño propuesto de cada una de las vistas que tendrá la aplicación web.

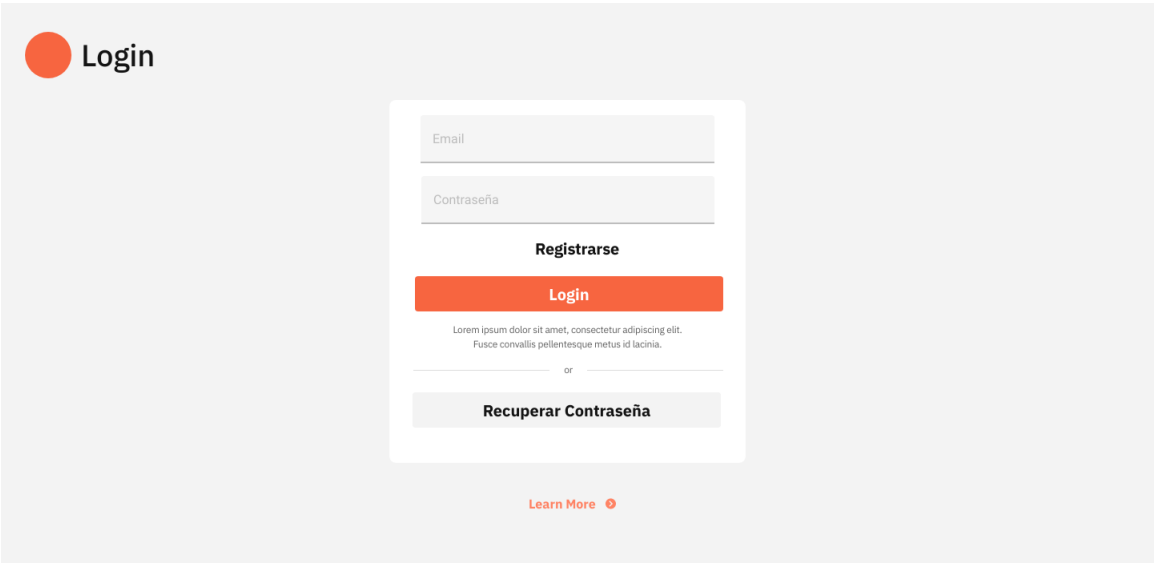


Figura 44: Página Inicio de Sesión



Figura 45: Página Principal

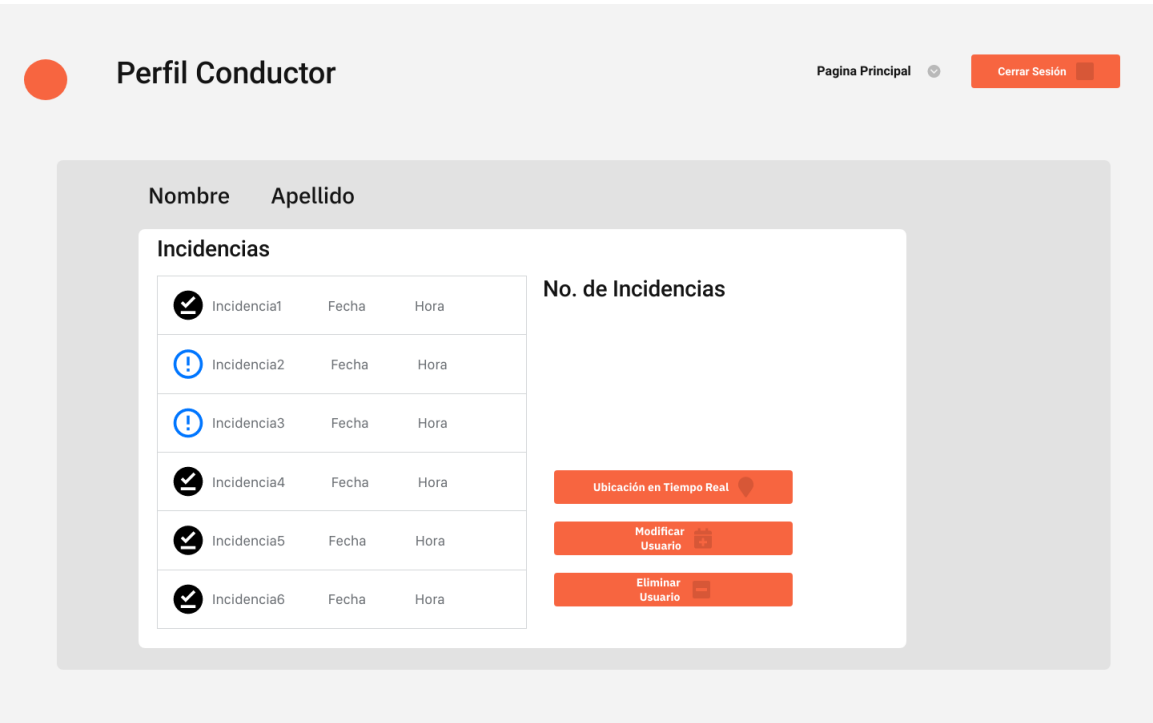
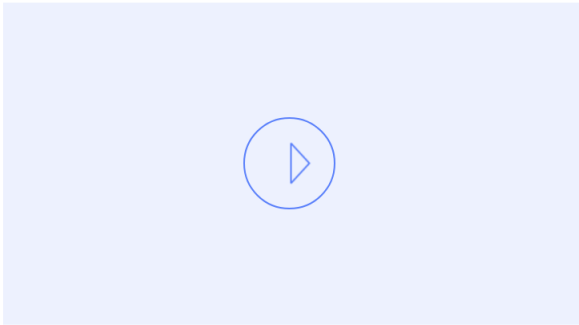


Figura 46: Página Perfil del Conductor

Reporte de Incidencia

Página Principal Cerrar Sesión

ID INCIDENCIA



Nombre Apellido

Fecha Hora

Ubicación (Coordenadas)

UBICACIÓN EN TIEMPO REAL

Confirmar Descartar

Figura 47: Página Detalle de Incidencia

Ubicación en Tiempo Real

Página Principal Cerrar Sesión

Nombre Apellido




Figura 48: Página Ubicación en Tiempo Real

6.3.2. Diseño de la Base de Datos

Elección del manejador de Base de Datos

■ MongoDB

Debido a la escalabilidad horizontal, facilidad de implementación, bajo coste y flexibilidad que ofrece los gestores de bases de datos NoSQL para adaptarse a las necesidades del proyecto, se optó por la elección de un gestor de este tipo. Además, se contemplaron las posibles modificaciones dentro del modelado de la base de datos, las cuales tendrán como motivo, cumplir los requisitos del sistema y presentar la información necesaria en la aplicación web. Estas modificaciones (en caso de ser necesarias) podrán ser fácilmente aplicadas usando un SGBD NoSQL ya que ofrecen una mayor facilidad de realizar cambios dentro del esquema a diferencia de un SGBD SQL.

Dentro de los SGBD NoSQL se eligió MongoDB, ya que la aplicación web será desarrollada utilizando como base el lenguaje JavaScript y el framework Node.js para realizar el *backend* de la aplicación. Por lo cual MongoDB, se integrará de manera funcional en el desarrollo del proyecto. Además, MongoDB es de código abierto, y es una herramienta gratuita hasta cierto punto, por lo cual no se requerirá una licencia para la implementación. Otra característica importante es que es una base de datos multiplataforma, esto nos permitirá registrar información desde el sistema operativo Linux, que se ocupará en la Raspberry Pi y por otro lado se ocupará Windows en la aplicación web para la inserción, modificación, lectura y eliminación de los datos en la base de datos.[13]

MongoDB utiliza "documentos" para registrar los datos, estos documentos son almacenados dentro de las colecciones, las cuales, en una base de datos relacional se asemeja a una tabla. Las bases de datos en MongoDB están conformadas por una o más colecciones de documentos.

MongoDB almacena registros de datos como documentos BSON. BSON es una representación binaria de documentos JSON, aunque contiene más tipos de datos que JSON.

Un registro en MongoDB es un documento, que es una estructura de datos compuesta por pares de campo y valor. Los documentos MongoDB son similares a los objetos JSON. Los valores de los campos pueden incluir otros documentos, matrices y matrices de documentos.

```
{
  name: "sue",
  age: 26,
  status: "A",
  groups: [ "news", "sports" ]
}
```



Figura 49: Estructura de un Documento en MongoDB

Basado en la estructura NOSQL de MongoDB, se propone el siguiente modelo para la base de datos, la cual contará con dos colecciones: Conductores e Incidencias. Cabe mencionar que en dicho modelo no se contempla a los administradores, debido a que se utilizará el sistema

de Amazon Cognito, que ofrece herramientas de autenticación, registro, verificación e inicio de sesión para el usuario.

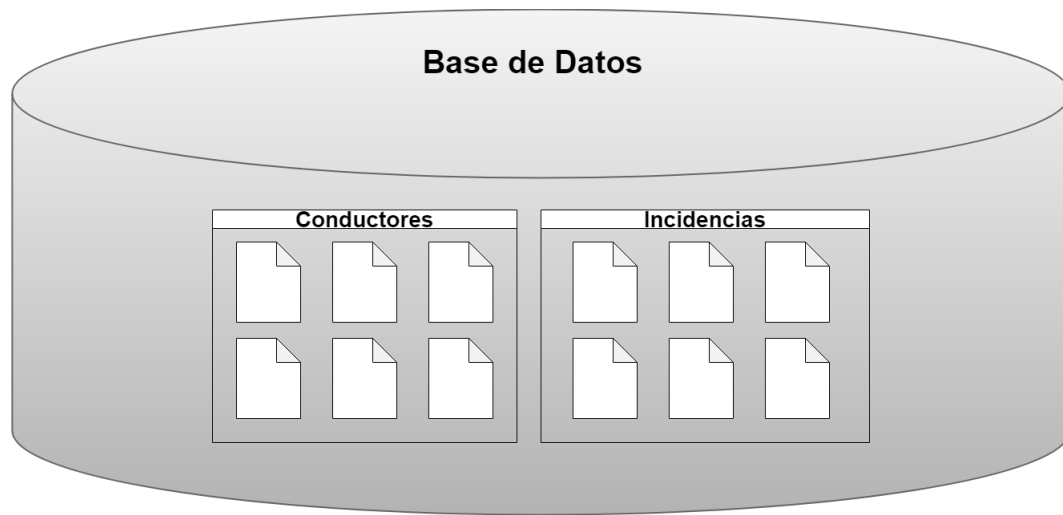


Figura 50: Modelo propuesto para la base de datos en MongoDB

El modelo de documento para la colección Conductores, se usará para registrar el nombre y apellido de cada conductor, así mismo se contabilizará el número de incidencias que presenté cada uno de ellos, el contador del conductor en marcha aumentará cada vez que el Módulo Central de Procesamiento envíe un reporte de incidencia a la base de datos. Si el Usuario Administrador del Módulo de la Estación Base revisa el video de la incidencia y lo cataloga como Descartar entonces se restará la incidencia, si esta es catalogada como Confirmar, el contador permanecerá igual. Es importante mencionar que el registro de cada conductor se realizará desde la Aplicación Web. A continuación, se muestran los campos y un ejemplo de los valores que pueden ser ingresados dentro del documento de registro de conductores.

```
{
  "_id" : 1,
  "NombreConductor" : "Nombre Conductor",
  "ApellidoConductor" : "ApellidoMaterno ApellidoPaterno",
  "NúmeroIncidencias" : 0
}
```

Figura 51: Documento de Registro de Conductores

El modelo de documento para la colección Incidencias, se usará para registrar el Reporte de Incidencia que presente un conductor, el cual será enviado desde el Módulo Central de Procesamiento, por tanto, se realizará una consulta previa a la colección Conductores para obtener el id, nombre y apellido del conductor en marcha. Posteriormente se realizará el reporte de la incidencia, el cual contendrá el Id del conductor, la fecha y hora de la incidencia, el nombre y apellido del conductor, y el estado de incidencia, este último muestra si la incidencia fue catalogada como descartada o confirmada por parte del Usuario Administrador después de revisar el video de la incidencia. Como se mencionó anteriormente, MongoDB permite anidar documentos, por lo que, dentro de este documento se encontrará anidado

un segundo documento llamado Detalles, el cual registrará la Ubicación donde se detectó la incidencia y la URL donde se encontrará almacenado el video.

```
{  "_id" : 1,
    "_idConductor" : "1",
    "NombreConductor" : "Alan",
    "ApellidoConductor" : "Gamboa del Angel",
    "FechaHora" : "2022-11-16T13:30:00.000Z",
    "Estado_Incidencia" : null,
    "Detalles": {
      "Ubicación" : [ -73.856077, 40.848447 ],
      "Video" : "URL"
    }
}
```

Figura 52: Documento de registro de Incidencias

Cabe señalar que los anteriores modelos muestran el campo Id, sin embargo, MongoDB asigna automáticamente un Id, si el id no es especificado, esto lo realiza cada vez que se crea un nuevo documento dentro de una colección, el cual es un valor hexadecimal de 12 bytes representado en 24 caracteres. Por tanto, se utilizará el id que proporcione Mongo DB en los documentos.

■ Amazon Cognito

Es una suite de herramientas que ofrece autenticación, autorización y administración de usuarios para aplicaciones móviles o web. Amazon Cognito utiliza dos componentes principales: los grupos de usuarios y grupos de identidades. Los grupos de usuarios se tratan de directorios que proporcionan a los usuarios de las aplicaciones opciones para inscribirse e iniciar sesión. Por otro lado, los grupos de identidades conceden a los usuarios acceso a otros servicios de *Amazon Web Services*

A continuación se muestra el funcionamiento de Amazon Cognito

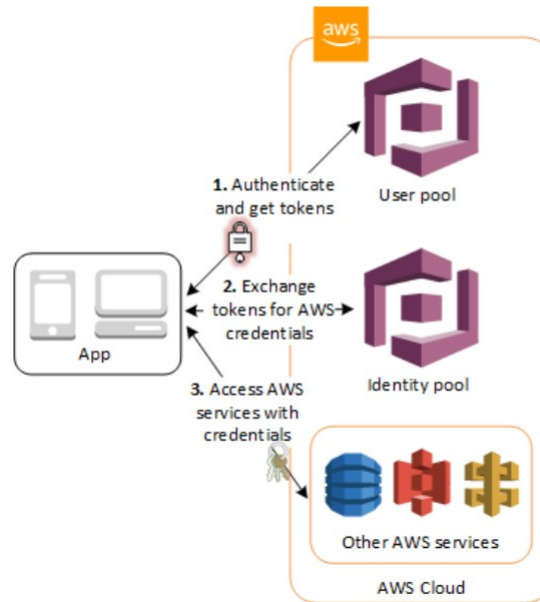


Figura 53: Documento de registro de Incidencias

1. En primer lugar, el usuario inicia sesión a través de su respectivo grupo de usuarios y recibe *tokens* de grupos de usuario después de una autenticación correcta.
2. Después, la aplicación intercambia dichos tokens del grupo de usuarios por las credenciales de AWS mediante un grupo de identidades.
3. Finalmente, el usuario puede utilizar estas credenciales de AWS para obtener acceso a otros servicios como Amazon S3.

Amazon Cognito se encuentra disponible en varias regiones alrededor del mundo.

■ Amazon S3 (*Simple Storage Service*)

Amazon S3, es un servicio de almacenamiento de objetos en la nube. Es utilizado para almacenar datos en la nube de una forma segura, eficiente y escalable. Este servicio utiliza elementos llamados *buckets*, que se encargan de almacenar objetos. Un objeto es un archivo o cualquier metadato que describa dicho archivo. Para almacenar datos en S3, primero se debe especificar el nombre de un bucket y la región dónde se planea que opere la aplicación. Esto con la intención de que el acceso a los datos se realice de manera eficiente.

7. Conclusiones

En un principio, se tenía contemplado utilizar la red LoRaWan para la comunicación y transferencia de archivos. Sin embargo, después de las actividades investigación, nos dimos cuenta de que esta opción no era viable, ya que el *bitrate* soportado por LoRaWan es insuficiente para la transferencia de archivos multimedia, que es una parte fundamental de nuestro sistema. Por tanto, se optó por utilizar la interfaz de LTE-M, una opción más costosa, pero que cumple los requisitos de una alta tasa de transferencia, así como una buena integración con la Raspberry Pi 4.

En el caso del Submódulo de Procesamiento de imágenes, se tenía contemplado desarrollarlo de manera separada del submódulo de *machine learning* pero después de haber realizado un análisis más profundo, se decidió integrarlo dentro de este mismo. Esto para el ahorro de recursos de procesamiento y conseguir una mejor comunicación entre procesos. En un siguiente reporte, se terminará por diseñar la Red Neuronal Convolucional, así como el diseño del Módulo de Telemetría y de la Estación Base.

Pará el diseño de la unidad de almacenamiento se obtuvo un bosquejo prototipo, sin embargo, se tiene contemplado rediseñar el modelo con el fin de mejorar y adaptarla los cambios del sistema a futuro.

8. Referencias

Referencias

- [1] www.who.int/publications/i/item/9789241565684
- [2] imt.mx/resumen-boletines.html?IdArticulo=449&IdBoletin=168
- [3] Rastreo satelital de vehículos: ¿qué es mejor que el GPS tradicional? (s/f). Beetrack.com. Recuperado de <https://www.beetrack.com/es/blog/rastreo-satelital-de-vehiculos>
- [4] "Sistemas de detección de somnolencia en conductores: Inicio, desarrollo y futuro", Revista Ingeniería y Región., vol. 13, n.º 1, pp. 159-168, 2015. Disponible: <https://journalusco.edu.co/index.php/iregion/article/view/717/1372>
- [5]
- [6] Kenneth E. Kendall, Julie E. Kendall, *Análisis y Diseño de Sistemas*, Octava Edición, Pearson Educación de México, 2011
- [7] <https://www.nperf.com/es/map/MX/-/2004799.Telcel-Mobile/signal/>
- [8] M. Paez y Emilio Abarca. *Resumen Boletines*, Instituto Mexicano del Transporte. <https://imt.mx/resumen-boletines.html?IdArticulo=449&IdBoletin=168> (accedido el 18 de marzo de 2022).
- [9] Jonathan Wexler, *Get Programming with Node.js*, Manning, 2019
- [10] Marjin Haverbeke *Eloquent Javascript*, Third Edition, No Starch Press, 2020
- [11] Amazon, *Amazon Amplify*, <https://aws.amazon.com/es/amplify/>
- [12] David A. Forsyth, Jean Ponce, *Computer Vision a Modern Approach*, Second Edition, Pearson, 2012
- [13] Kyle Banker, *MondoDB in Action*, Second Edition, Manning, 2016
- [14] Fernando Berzal, *Redes Neuronales y Deep Learning*, Granada, 2018
- [15] Irv Kalb, *Object-Oriented Python*, Primera Edición, No Starch-Press, 2021
- [16] V. Moret Bonillo, *Fundamentos de Inteligencia Artificial*, Segunda Edición, Santiago de Compostela: Universidad de La Coruña Servicio de Publicaciones, 2005.
- [17] Ian Sommerville, *Ingeniería de Software*, Novena Edición, Pearson Educación de México, 2011
- [18] Aurélien Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and Tensorflow*, Segunda Edición, O'Reilly, 2009
- [19] Kurt Demaagd, Anthony Oliver, Nathan Oostendorp y Katherine Scott *Practical Computer Vision with OpenCV*, Tercera Edición, O'Reilly, 2017

- [20] Cuno Plister, *Getting Started with Internet of Things*, Primera Edición, O'Reilly, 2011
- [21] Vilca Espinoza, R.A, *Influencia de un sistema de geolocalización en el control y monitoreo de vehículos con dispositivos GPS en una empresa logística*, 2007
- [22] S. Pachouly, N. Bhondve, A. Dalvi, V. Dhande y N. Bhamare. "Driver drowsiness detection using machine learning with visual behaviour". <https://ijcrt.org/papers/IJCRT2006408.pdf> (accedido el 2 de noviembre de 2022).
- [23] J. A. Castillo. *WLAN: Qué es, definición, estándar 802.11 y diferencias con LAN*. <https://www.profesionalreview.com/2020/03/07/wlan-que-es/> (accedido el 2 de noviembre de 2022).
- [24] Tarify.Win!Red WWAN ¿Qué es, para qué sirve y cómo funcionan? <https://tarify.win/definiciones/red-wwan/>
- [25] La Rosa, A. (2021, 13 julio). *LPWAN como base de comunicaciones para IoT* Pandora FMS-The Monitoring Blog. Recuperado 8 de mayo de 2022, de <https://pandorafms.com/blog/es/que-es-lpwan/?msclkid=08006fc5cf4611ecbc00aa3b275ed2b5>
- [26] *LTE-M, el despegue esperado para IoT. Hablemos de empresas*. <https://hablemosdeempresas.com/grandes-empresas/lte-m-despliegue-soluciones-iot/>
- [27] Sánchez Rosado, David, *NB-IoT tecnologías celulares narrow-band: análisis práctico de las soluciones de Telefónica y Vodafone*, 2019