



Organización
de las Naciones Unidas
para la Educación,
la Ciencia y la Cultura



Cátedra UNESCO
Tecnologías de apoyo para
la Inclusión Educativa



REVISTA

JUVENTUD Y CIENCIA SOLIDARIA:

En el camino de la investigación

PROTOTIPO DE SISTEMA DE FRENADO INTELIGENTE PARA VEHÍCULOS MONITOREADO CON IOT PARA MINIMIZAR ACCIDENTES O EMERGENCIAS EN CARRETERAS

Benjamín Azanza Díaz, Ariel Nicolas Bravo Jurado,
David Alejandro Cruz Palacios, Daniel Alejandro Urbina Artega



Mi nombre es **Benjamín Azanza Díaz**, tengo 17 años y estudio el 3ero de BACH en el Colegio Técnico Salesiano Don Bosco. Me gusta jugar en el PC, salir de fiesta y programar. Quiero estudiar Ingeniería en Software en la universidad.



Mi nombres es **Ariel Nicolas Bravo Jurado**, tengo 18 años y estudio el 3ero de BACH en el Colegio Técnico Salesiano Don Bosco. Me gusta resolver puzles, hacer bicicleta y nadar. Quiero estudiar una carrera referente a la electrónica y programación.



Mi nombre es **David Alejandro Cruz Palacios**, tengo 17 años y estudio el 3ero de BACH BGU en el Colegio Técnico Salesiano Don Bosco. Me gusta jugar a videojuegos, programar y la tecnología. Quiero estudiar Ingeniería en Software en la universidad.



Daniel Alejandro Urbina Artega, tengo 18 años y esttudio el 3ero de BACH BGU en el Colegio Técnico Salesiano Don Bosco. Me gusta la lectura y jugar videojuegos. Quiero estudiar Ingeniería en sistemas en la universidad.

Freno de mano

Según la revista *Car And Driver*, “El freno de emergencia es un dispositivo mecánico que se utiliza para mantener el vehículo en su lugar cuando está estacionado” [6]. El freno de emergencia se activa mediante una palanca ubicada en el habitáculo del vehículo, que está conectada a un sistema de cables que se extienden hasta los frenos traseros [7]. Al tirar de la palanca, se aprietan las pastillas de freno traseras contra los discos para detener el movimiento del vehículo [8, 9]. En la Fig. 2 se muestra un esquema del sistema presente en el freno de mano para su funcionamiento.

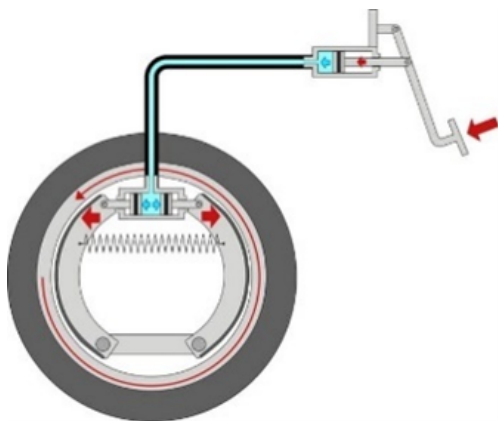


Figura 2. Sistema de frenado Fuente: [6]

Freno de disco

Según la revista *Car and Driver*, “Los frenos de disco se componen de un disco de freno, que es una placa metálica que gira con la rueda, y una pinza, que se ajusta contra el disco de freno para detener la rueda” [6]. Cuando se aplica la presión de frenado, las pastillas de freno se presionan contra el disco de freno, lo que detiene la rueda. Los frenos de disco proporcionan una excelente capacidad de frenado y son menos propensos a sufrir fatiga que los frenos de tambor [10, 11]. En la Fig. 3 localizamos un esquema del sistema presente en el freno de disco para su funcionamiento.

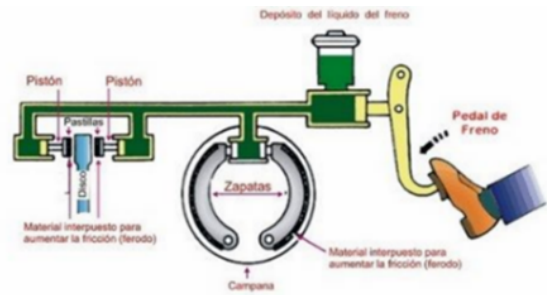


Figura 3. Sistema freno de sistema Fuente: [6]

0.0.1. Freno ABS

Según el sitio web de *HowStuffWorks*, “El sistema de frenos ABS evita que las ruedas se bloqueen al frenar, lo que ayuda a mantener el control del vehículo en situaciones de emergencia” [12]. El ABS funciona mediante la medición de la velocidad de las ruedas y la aplicación de los frenos de manera intermitente para evitar el bloqueo de las ruedas [13, 14, 15]. En la Fig. 4 visualizamos un diagrama explicativo del funcionamiento del ABS.

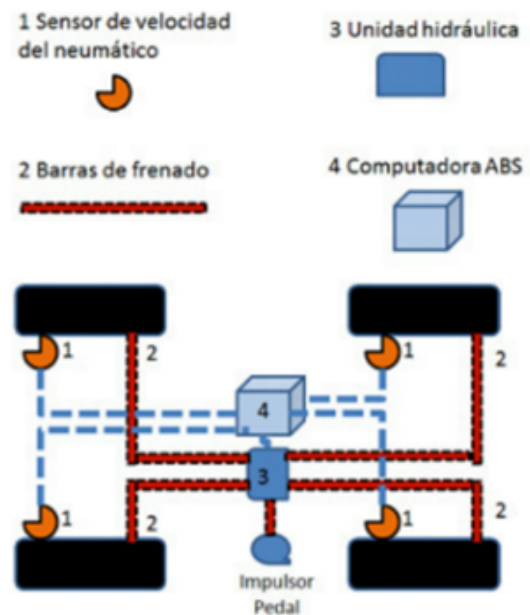


Figura 4. Freno de ABS Fuente: [12]

Node-RED

Herramienta de programación visual que permite a los usuarios crear aplicaciones de Internet de las cosas

(IoT) de manera intuitiva y sin necesidad de escribir código. Node-RED se integra con una amplia variedad de dispositivos IoT, incluidos los dispositivos Arduino. Según palabras de Nick O’Leary, creador de Node-RED, esta “es una herramienta de programación visual

que permite a los usuarios conectar y programar dispositivos IoT de manera sencilla y eficiente” [16]. En la Fig. 5 encontramos una de la interfaz de desarrollo de Node-RED.

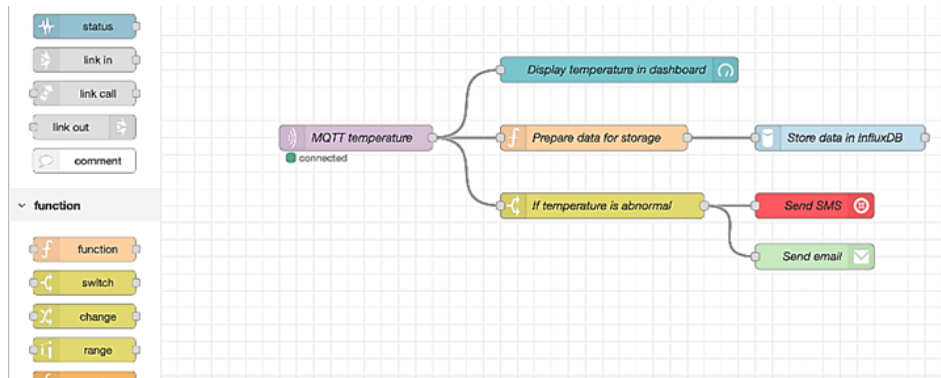


Figura 5. Interfaz de Node-RED
Fuente: [16]

Freeboard

Freeboard.io es una plataforma en línea que permite a los usuarios crear paneles de control interactivos para sus dispositivos IoT. La plataforma se integra con una amplia variedad de dispositivos IoT, incluidos los dispositivos Arduino. Según palabras de Steve Loughran,

ingeniero de software de Adobe, “Freeboard.io es una herramienta excelente para crear paneles de control para dispositivos IoT” [17]. Los usuarios pueden conectar sus dispositivos Arduino a Freeboard.io y crear paneles de control interactivos en cuestión de minutos [18, 19]. En la Fig. 6 encontramos un ejemplo de *dashboard* diseñada en Freeboard.

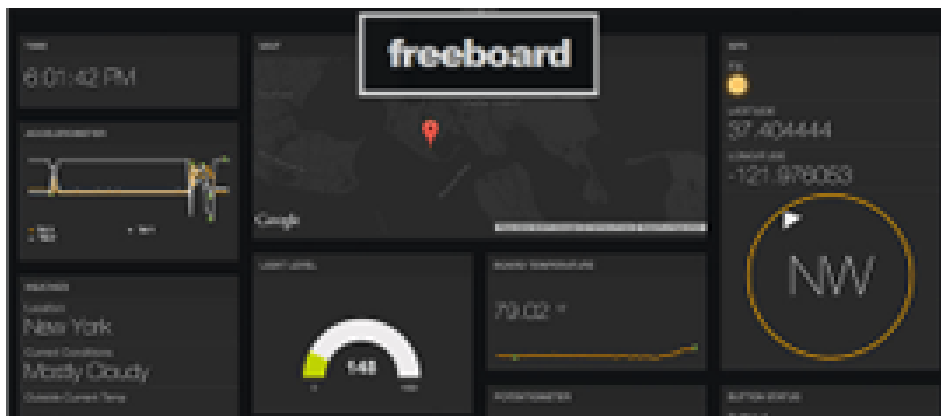


Figura 6. Interfaz de Freeboard
Fuente: [15]

Dashboard

Herramienta personalizable de visualización de datos, que te ayuda a conectar tus archivos, servicios, API o archivos adjuntos, y muestra estos datos como tablas,

tipos de gráficas u otras visualizaciones de datos al espectador y reduce el esfuerzo manual [17]. El seguimiento en tiempo real de este tipo de datos reduce las horas de análisis y las largas líneas de comu-

nicación que antes suponían un reto para las empresas. un *dashboard*.
La Fig. 7 hace referencia a la interfaz ejemplificada de



Figura 7. Interfaz de una Dashboard
Fuente: [17]

Arduino

Es una plataforma de hardware libre y de código abierto basada en microcontroladores, que permite a los usuarios desarrollar y programar proyectos in-

teractivos y autónomos. Según Sedra Smith, Arduino es "una herramienta para crear prototipos electrónicos basados en software y hardware flexibles y fáciles de usar" [20]. En la Fig. 8 presentamos hardware y software diseñado por Arduino.



Figura 8. Hardware y software de Arduino
Fuente: [21]

Sensor de ritmo cardiaco

Según el sitio web oficial de Arduino, “El sensor de ritmo cardíaco para Arduino es un dispositivo que se utiliza para medir la frecuencia cardíaca de una persona” [22]. El sensor se coloca en la punta del dedo y mide la cantidad de luz que se refleja en la sangre mientras el corazón late [23]. Visualizamos, en la Fig. 9, el sensor de ritmo cardiaco presente en el prototipo.



Figura 9. Sensor de ritmo cardiaco para Arduino
Fuente: [20]

Diagrama de bloques del sistema

El prototipo funciona de la siguiente manera: Se abre NODE-RED y Freeboard en el computador, y se conecta al microcontrolador mediante el cable serial. A su vez, el microcontrolador se conecta a un Shield, que logra una mejor presentación y ayuda a ordenar los distintos cables. De este Shield es que se derivan el resto de sensores y actuadores del prototipo, siendo todos controlados por un *switch* para habilitar el modo automático o el modo manual, según corresponda. El L298 es el único que se encuentra conectado a una fuente de 12V DC, esto para lograr que se muevan todos los motores a velocidad visible. A continuación, en la Fig. 10, observamos una imagen a detalle del diagrama de bloques ya explicado.

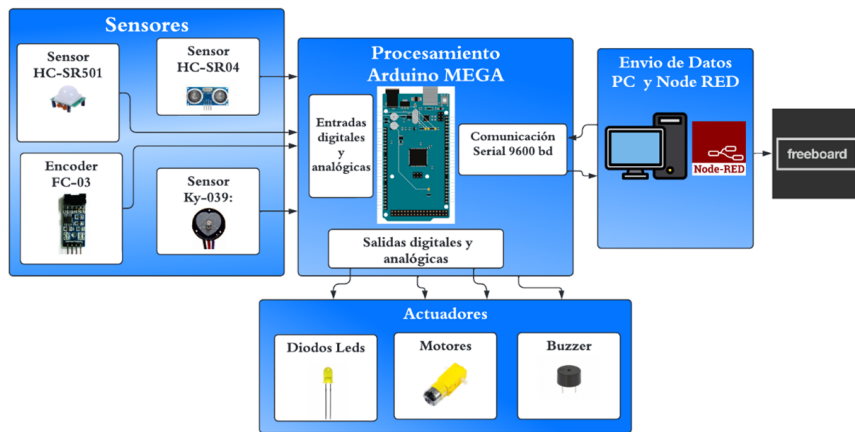


Figura 10. Diagrama de bloques general del sistema
Fuente: Autores

Diagrama de nodos de programación en node red



Figura 11. Diagrama de nodos general
Fuente: Autores

En la configuración de nodos en el entorno de Node Red como se aprecia en la Fig.11, se encuentran implementados varios nodos para el control manual y automático del automóvil. Cada nodo manual está asignado a un dato específico representado por una letra. Esto permite enviar directamente el dato desde el dashboard hacia el Arduino, lo cual controla los leds, motores y también la slider. En la segunda imagen, se pueden observar los nodos que controlan la parte automática del automóvil.

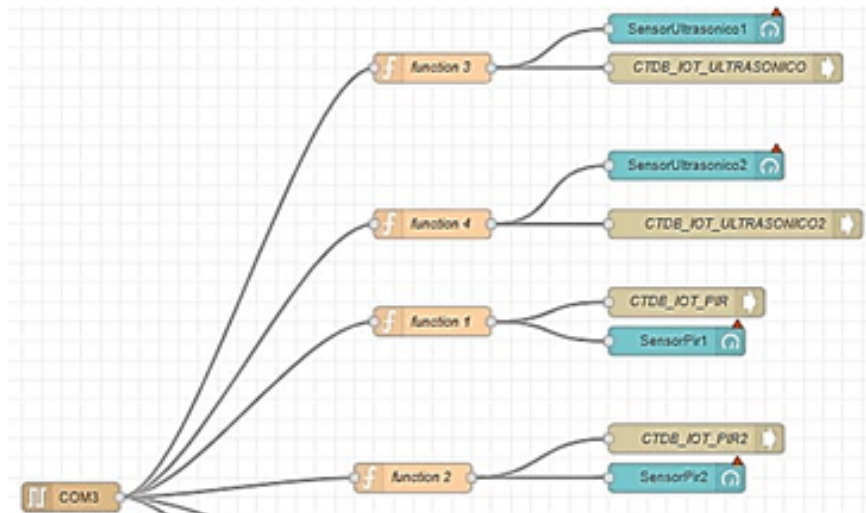


Figura 12. Diagrama de Nodos para Dashboard
Fuente: Autores

Estos nodos envían datos hacia el Freeboard, y cada entrada de datos está controlada por una función con un nombre asignado en el *string* del Arduino. Los nombres de las vías de datos hasta el servidor también

tienen su propio nombre, como se puede observar en la Fig. 12, lo cual asegura una correcta transmisión de los datos hacia el Freeboard.

Ensamblaje del vehículo

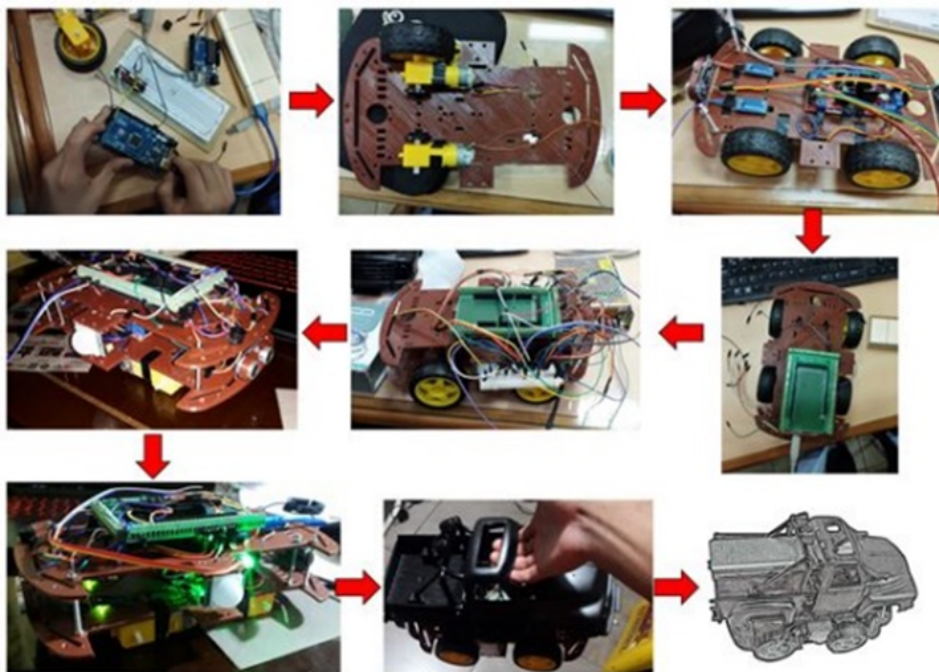


Figura 13. Proceso de montaje general
Fuente: Autores

La Fig. 13 muestra el montaje de cuatro motores y sus correspondientes ruedas en la parte inferior del chasis, junto con la implementación de diversos sensores y drivers para los motores. En la parte superior del chasis, se colocó el microcontrolador junto con su respectivo shield mientras que en la carcasa del prototipo se disponen las luces y el interruptor. Además, se realiza la instalación de las conexiones de acuerdo con la programación y se ajustan al chasis para asegurar un funcionamiento adecuado del sistema en su conjunto.

Enlace con Freeboard y el Dashboard de Node-Red

Se muestran dos ejemplos de enlaces entre Freeboard y el Dashboard de Node-Red. En el primer caso, se integra el proyecto con un Arduino MEGA y se visualizan las mediciones en tiempo real utilizando widgets interactivos en Freeboard. En el segundo ejemplo, un impresionante Dashboard desarrollado en Node-Red controla dispositivos IoT, recolectando datos ambientales y presentándolos mediante gráficas y controles deslizantes personalizados. En la Fig.14 encontramos las interfaces desarrolladas en ambas webs.



Figura 14. Interfaces Freeboard y Dashboard
Fuente: Autores

Prototipo final

En este apartado observamos la primera versión funcional del prototipo, posteriormente se agregaron elementos netamente decorativos para que el proyecto

sea más presentable, pero el funcionamiento seguía siendo el mismo. En la Fig. 15 se detalla el estado del proyecto en ese momento.



Figura 15. Resultado del prototipo final
Fuente: Autores

Pruebas y resultados

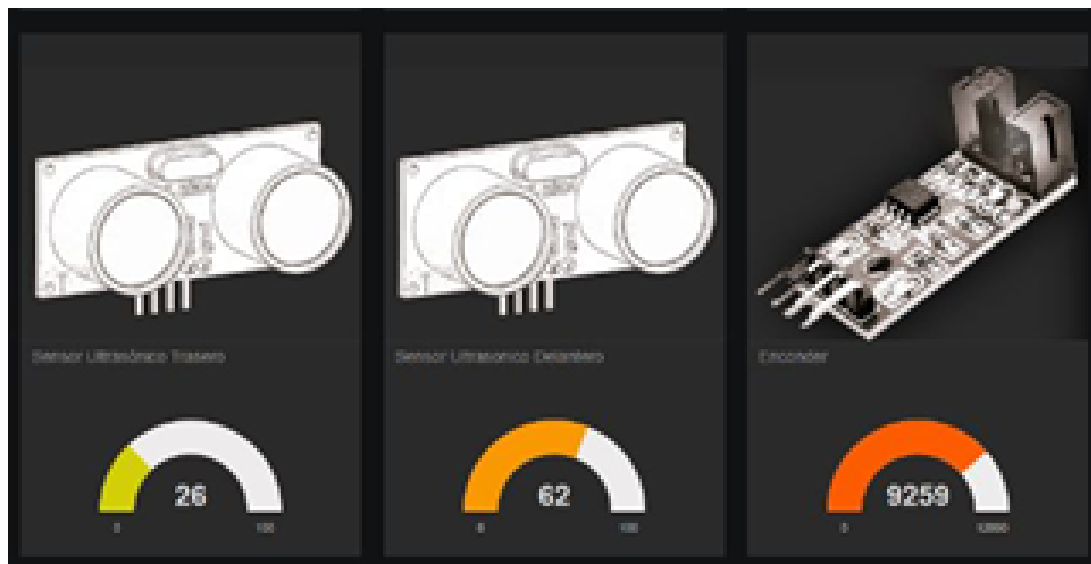


Figura 16. Funcionamiento y registro en Freeboard
Fuente: Autores

En la Fig.16 se visualizan las medidas en tiempo real de los sensores ultrasónicos, el sensor PIR y las revoluciones por minuto (RPM) enviados por el en-

coder mediante la plataforma de Freeboard, con lo que se cumple con la medición de los parámetros del prototipo para usarlos en los frenos del vehículo.

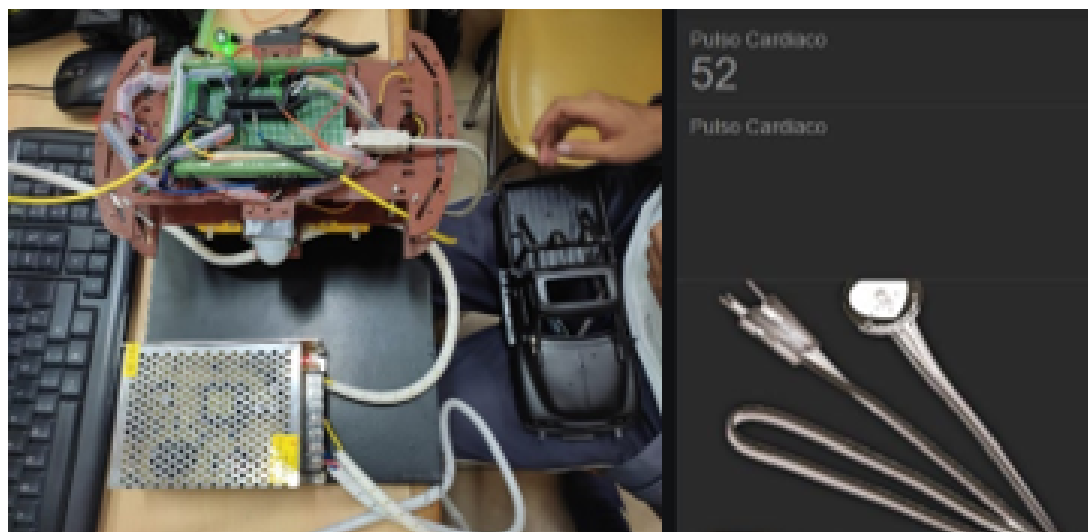


Figura 17. Medidas del ritmo cardiaco en tiempo real a través de la plataforma Freeboard
Fuente: Autores

Por último, en la Fig.17 se observa las medidas del ritmo cardiaco del prototipo desde la plataforma Freeboard en conjunto con el funcionamiento de los frenos, al tener un ritmo cardiaco por debajo del límite progra-

mado se accionan los frenos ABS y las direccionales del vehículo completando la maniobra del freno inteligente propuesto.

Conclusiones

- La implementación de una interrupción por pin en el sistema de frenado automático, junto con el uso de señales PWM y el shield L298, ha logrado una detección precisa y oportuna de las señales del encoder, mejorando significativamente el control, monitoreo y rendimiento del frenado, garantizando así una mayor seguridad en el sistema del vehículo a escala.
- El encoder desempeña un papel fundamental en nuestro proyecto, al ser el mecanismo primario para detectar posibles accidentes y proporcionar información crucial para el funcionamiento del prototipo. La configuración de pines, la captura y transmisión de datos a través del IoT, así como la obtención precisa de mediciones, dependen en gran medida de las lecturas del Encoder. Su correcto desempeño y precisión son vitales para garantizar la seguridad y el rendimiento óptimo del sistema.
- La integración de un sensor de ritmo cardiaco en un sistema de frenado automático monitoreado por IoT ofrece una capacidad adicional para evaluar la condición fisiológica del conductor en tiempo real. Al medir el ritmo cardiaco (BPM), el sistema puede determinar si el conductor está en un estado estable para conducir, lo que permite tomar decisiones más informadas sobre la activación o desactivación del frenado automático. Esto mejora la seguridad y proporciona un control más preciso en situaciones de conducción potencialmente peligrosas.

Agradecimientos

Primeramente, agradecemos a los padres de familia por darnos la oportunidad de estudiar y adquirir los conocimientos necesarios para desarrollar este proyecto. Agradecemos a la institución por permitirnos formar parte de una comunidad salesiana que lleva consigo los valores cristianos inculcados por San Juan Bosco y María Auxiliadora. A nuestros docentes, que nos han formado como verdaderos estudiantes técnicos, nos han ayudado a enfrentar nuestras responsabilidades sin im-

portar los problemas que se nos presenten, logrando llegar a ser profesionales honestos ante la sociedad y el mundo laboral. Finalmente, y en reconocimiento a todo lo que han hecho los profesores por nosotros, deseamos expresar nuestro agradecimiento por su ardua labor en nuestra educación y enriquecimiento de conocimiento. Por último, también agradecemos a la vida por habernos unido como integrantes de este grupo, formando una unidad tan funcional.

Referencias

- [1] S. Anbalagan, P. Srividya, B. Thilaksurya, S. G. Senthivel, G. Suganeshwari, y G. Raja, «Vision-Based Ingenious Lane Departure Warning System for Autonomous Vehicles», *Sustainability*, vol. 15, n.º 4, Art. n.º 4, ene. 2023, doi: 10.3390/su15043535.
- [2] A. Ziębiński, R. Cupek, D. Grzechca, y L. Chruszczyk, «Review of advanced driver assistance systems (ADAS)», presentado en AIP Conference Proceedings, nov. 2017, p. 120002. doi: 10.1063/1.5012394.
- [3] Kaspersky, «¿Qué es la Internet de las cosas? Definición y explicación», latam.kaspersky.com. [En línea]. Disponible en: <https://shorturl.at/ajpsM>
- [4] K. Hiramatsu, «Design Principles for Advanced Driver Assistance System: Keeping Drivers In-the-Loop», *International Harmonized Research Activities (IHRA)*, 2010.
- [5] R. Jabbar, M. Shinoy, M. Kharbeche, K. Al-Khalifa, M. Krichen, y K. Barkaoui, «Urban Traffic Monitoring and Modeling System: An IoT Solution for Enhancing Road Safety», en *2019 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC)*, dic. 2019, pp. 13-18. doi: 10.1109/IINTEC48298.2019.9112118
- [6] CAR AND DRIVER, «Ken Block prueba el nuevo freno de mano electrónico del Focus RS», Car and Driver. [En línea]. Disponible en: <https://shorturl.at/yMR03>
- [7] W. Shi, M. B. Alawieh, X. Li, y H. Yu, «Algorithm and hardware implementation for visual perception system in autonomous vehicle: A survey», *Integration*, vol. 59, pp. 148-156, sep. 2017, doi: 10.1016/j.vlsi.2017.07.007.

- [8] V. Kumar, P. Aravind, S. Pooja, S. Prathyush, S. AngelDeborah, y K. Chandran, «Driver Assistance System using Raspberry Pi and Haar Cascade Classifiers», may 2021, pp. 1729-1735. doi: 10.1109/ICICCS51141.2021.9432361.
- [9] S. Grubmüller, J. Plihal, y P. Nedoma, «Automated Driving from the View of Technical Standards», en *Automated Driving: Safer and More Efficient Future Driving*, D. Watzenig y M. Horn, Eds., Cham: Springer International Publishing, 2017, pp. 29-40. doi: 10.1007/978-3-319-31895-0_3.
- [10] H. Thevendran, A. Nagendran, H. Hydher, A. Bandara, y U. Oruthota, «Deep Learning & Computer Vision for IoT based Intelligent Driver Assistant System», en *2021 10th International Conference on Information and Automation for Sustainability (ICIAfS)*, ago. 2021, pp. 340-345. doi: 10.1109/ICIAfS52090.2021.9605823.
- [11] B. Varma, S. Sam, y L. Shine, «Vision Based Advanced Driver Assistance System Using Deep Learning», en *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, jul. 2019, pp. 1-5. doi: 10.1109/ICCCNT45670.2019.8944842.
- [12] HowStuffWorks, «Could anti-lock brakes detect a flat? | HowStuffWorks». [En línea]. Disponible en: <https://shorturl.at/beAT0>
- [13] D. Yi, J. Su, C. Liu, M. Quddus, y W.-H. Chen, «A machine learning based personalized system for driving state recognition», *Transportation Research Part C: Emerging Technologies*, vol. 105, pp. 241-261, ago. 2019, doi: 10.1016/j.trc.2019.05.042.
- [14] R. Albasrawi, F. F. Fadhil, y M. T. Ghazal, «Driver drowsiness monitoring system based on facial Landmark detection with convolutional neural network for prediction», *Bulletin of Electrical Engineering and Informatics*, vol. 11, n.º 5, Art. n.º 5, oct. 2022, doi: 10.11591/eei.v11i5.3966.
- [15] K. Ferencz y J. Domokos, «Using Node-RED platform in an industrial environment», feb. 2020.
- [16] S. Hind, «Dashboard design and the ‘datafied’ driving experience», *Big Data & Society*, vol. 8, n.º 2, p. 205395172110498, jul. 2021, doi: 10.1177/20539517211049862.
- [17] M. Meena y V. Prakash, «Vehicle to Vehicle Communication for Collision Avoidance», *Engineering, Technology and Applied Science Research*, vol. 6, pp. 1380-1386, may 2018.
- [18] S. Garethiya, L. Ujjainiya, y V. Dudhwadkar, «Predictive vehicle collision avoidance system using Raspberry - pi», *ARPN Journal of Engineering and Applied Sciences*, vol. 10, pp. 3655-3659, ene. 2015.
- [19] Arduino Spain, «[GUÍA] Arduino sensor de pulso cardiaco ky-039 + código, conexión», Arduino Spain. [En línea]. Disponible en: <https://shorturl.at/rwzQZ>
- [20] L. Louis, «Working Principle of Arduino and Using it as a Tool for Study and Research», presentado en *International Journal of Control, Automation, Communication and Systems*, jul. 2018. doi: 10.5121/ijcacs.2016.1203.
- [21] P.-Y. Hsiao, C.-W. Yeh, S.-S. Huang, y L.-C. Fu, «A Portable Vision-Based Real-Time Lane Departure Warning System: Day and Night», *Vehicle Technology, IEEE Transactions on*, vol. 58, pp. 2089-2094, jun. 2009, doi: 10.1109/TVT.2008.2006618.
- [22] D. Parekh *et al.*, «A Review on Autonomous Vehicles: Progress, Methods and Challenges», *Electronics*, vol. 11, n.º 14, Art. n.º 14, ene. 2022, doi: 10.3390/electronics11142162.