

SmartParkUdeA: Sistema IoT para el estacionamiento inteligente de vehículos en ciudad universitaria

Daniel Rivera Arroyave, Juan Jaramillo Tobón, Raúl Arcila Rodríguez and Danny Múnera Ramírez

Facultad de Ingeniería, Universidad de Antioquia, Medellín, Colombia

Email: {daniel.riveraa} {juan.jaramillo62} {randres.arcila} {danny.munera}@udea.edu.co

Resumen—La gestión de estacionamiento es un problema cada vez más frecuente en las grandes ciudades. En este artículo se presenta una solución de estacionamiento inteligente basada en tecnologías del Internet de las Cosas. Nuestro sistema propone un algoritmo de detección de vehículos con un magnetómetro, acompañado de un sistema de comunicaciones usando la tecnología LoRaWAN y una aplicación web completa. Un prototipo funcional fue diseñado e implementado para el estacionamiento de la Universidad de Antioquia.

Index Terms—Estacionamiento Inteligente (Smart Parking), LoRaWAN, Internet de las Cosas (IoT)

I. INTRODUCCIÓN

Debido al aumento de vehículos en el área metropolitana y particularmente la ciudad de Medellín, encontrar un estacionamiento se ha convertido en un problema cada vez más común y la ciudadela universitaria de la Universidad de Antioquia (UdeA) no es ajena a este problema. El campus de la Universidad cuenta con 949 celdas de estacionamiento distribuidas en 3 sectores principales. La Universidad de Antioquia presenta un ingreso diario de 3.315 vehículos en promedio [1], que si bien entran y salen constantemente, terminan por ocupar gran parte de las zonas de estacionamiento. Esto provoca que algunos vehículos demoren en encontrar una celda disponible, generando un impacto negativo en la disposición de tiempo de los ocupantes, aumentando el tráfico dentro de la universidad e incrementando la emisión de gases causada por estos vehículos [2].

El problema de gestión de los sitios de estacionamiento se ha abordado haciendo uso de sistemas tecnológicos. Los sistemas de estacionamiento inteligente ayudan a los conductores a encontrar un lugar vacante, usando sensores en cada espacio de estacionamiento, los cuales detectan la presencia o ausencia de un vehículo. En sitios de interés, como centros comerciales, comúnmente se implementan estos sistemas utilizando cámaras de video que enfocan cada celda de estacionamiento. El sistema utiliza elementos como luces o pantallas para indicar el estado de las celdas (libre/ocupado) y permite que los usuarios encuentren de forma eficiente un lugar libre [3]. La implementación de una solución basada en cámaras de video puede llegar a ser muy costosa para la Universidad, debido a que la infraestructura de estacionamiento del campus universitario es muy amplia y a que dicha solución está diseñada específicamente para espacios cerrados.

En este artículo se propone un sistema de estacionamiento inteligente para el campus de la UdeA, llamado SmartParkUdeA. Este sistema ofrece una solución de bajo costo y gran

cobertura, la cual está basada en las tecnologías del Internet de las Cosas (IoT, por sus siglas en inglés, *Internet of Things*). SmartParkUdeA propone la implementación de nodos con sensores capacitados para ser instalados a la intemperie, los cuales permiten la detección de automóviles. Estos sensores se instalan en cada una de las celdas de estacionamiento y se comunican a través de LoRaWAN (*Long-Range Low-Power Wide Area Network*) [4].

LoRaWAN es una pila de red que presenta una velocidad de datos máxima de 27 kb/s, la cual afirma que un solo *Gateway* puede recopilar datos de miles de nodos desplegados a kilómetros de distancia. Estas características posicionan esta tecnología como habilitadora de conectividad para aplicaciones IoT [5].

Las tecnologías utilizadas en SmartParkUdeA permiten que la infraestructura necesaria para desplegar el sistema sea de bajo costo, además, gracias a la alta eficiencia energética de las comunicaciones, el costo de mantenimiento del sistema es también bajo. El sistema propuesto cuenta adicionalmente con una aplicación de software, la cual permite recolectar los datos del estado de las celdas en tiempo real y presentar dichos datos a los usuarios interesados a través de una interfaz web. El sistema ha sido validado a través de una prueba funcional demostrando el cumplimiento de las características principales del prototipo.

En el resto de este artículo presentamos un análisis de la arquitectura de la solución en la Sección II. La Sección III muestra un análisis de los posibles sensores para la detección de automóviles. La Sección IV expone el algoritmo de detección de vehículos propuesto en este trabajo. Finalmente, la Sección V presenta la implementación de un prototipo funcional del sistema. Las conclusiones son presentadas en la Sección VI.

II. ARQUITECTURA DEL SISTEMA

Para el desarrollo inicial de este proyecto se realizó una etapa inicial de análisis de requisitos, a partir de la cual determinamos que el sistema debe tener las siguientes funcionalidades principales: Presentar en tiempo real el estado (libre/ocupado) de las celdas de estacionamiento, mostrar el número de celdas disponibles por zona, capturar datos estadísticos sobre el uso del estacionamiento y permitir cambiar el perfil energético de los dispositivos.

En la Figura 1 se pueden ver los componentes de la arquitectura principal del sistema planteada para cumplir las funciones requeridas. Adicionalmente se presentan los protocolos de

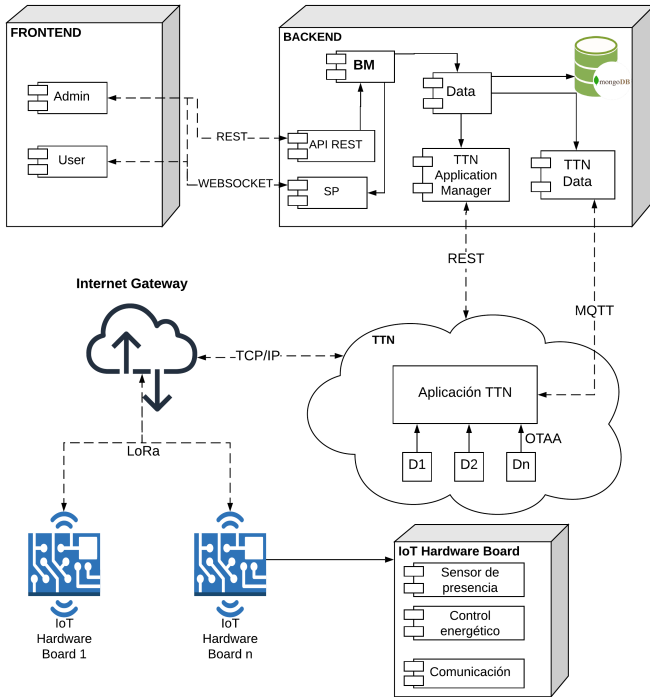


Figura 1: Arquitectura general del sistema.

comunicación identificados para establecer la comunicación entre los diferentes componentes. La arquitectura propuesta posee cuatro módulos principales: *Frontend*, *Backend*, *Servicio externo The Things Network* (en adelante abreviado como TTN) y el *IoT Hardware Board*. Cada uno de estos módulos serán descritos a continuación.

En cada una de las celdas de estacionamiento se implementa un *IoT Hardware Board*, el cual determina el estado de la celda. A través del protocolo LoRaWAN los dispositivos envían el estado de cada celda por Internet a través de un *Gateway LoRaWAN* ubicado frente a una de las zonas de estacionamiento. La tecnología LoRaWAN permite un bajo costo de implementación, un bajo consumo energético en las comunicaciones y, adicionalmente, un despliegue económico de la red, debido al gran alcance de la comunicación [4].

El *Gateway* reenvía los paquetes recibidos al servicio externo de TTN [6] el cual es un servicio open-source y descentralizado que facilita el uso de la tecnología LoRaWAN para el desarrollo de sistemas IoT. El sistema implementa un módulo *Backend* el cual se conecta a la aplicación TTN y almacena la información relevante sobre las celdas de estacionamiento, asimismo provee un conjunto de servicios web que son consumidos por el módulo *Frontend*. Este módulo permite presentar a los usuarios del sistema el estado de las celdas en tiempo real, esta funcionalidad se apoya en el uso de *websockets* para subscribirse al canal donde se actualiza el estado de los dispositivos.

III. SENSORES PARA LA DETECCIÓN DE VEHÍCULOS

La detección eficiente de vehículos en una celda de estacionamiento es una funcionalidad crítica para este proyecto, por lo que se realizó una evaluación del desempeño de tres tipos de sensores: mecánico, ultrasónico y magnético. Esta evaluación se realizó basada en tres criterios principales: la resistencia del sensor a la intemperie, su consumo energético y su precisión.

Sensor mecánico: Un sensor mecánico detecta la presencia de un automóvil por medio del movimiento de un mecanismo que acciona un interruptor. La implementación de este sensor podría implicar altos costos en instalación. Respecto a la *resistencia*, la implementación del sensor no requiere el uso de elementos electrónicos complejos, pero los elementos mecánicos deben ser de buena calidad, ya que son partes móviles que estarán en constante uso. Respecto al *consumo*, el de un interruptor es nulo, no consume energía en reposo (sin actividad), dado que el sistema detecta la entrada o salida de un vehículo con el cambio de estado del interruptor. Por último, respecto a la *precisión*, este sensor proporciona información muy básica acerca del objeto que esta sobre la celda de estacionamiento, por lo tanto la ubicación del sensor es crítica para poder sacar la mayor cantidad de información posible.

Sensor de ultrasonido: El sensor de ultrasonido usa ondas mecánicas por encima del espectro audible para determinar la distancia entre el sensor y el objeto con el cual la onda choca y se refleja. Durante la fase de evaluación se utilizó un sensor ultrasónico de referencia HC SR04. Este sensor está ubicado en el suelo con sus elementos de detección orientados hacia arriba para detectar los vehículos. Respecto a la *resistencia*, el sensor debe estar a la intemperie y no se puede cubrir. Esta referencia no puede ser expuesta al agua, aunque existen alternativas resistentes, estas son muy costosas. Respecto al *consumo*, en reposo el consumo del sensor es menor a 2mA, el consumo en trabajo, es de 15 mA. En el caso de nuestro proyecto el sensor debería estar en constante trabajo. Por último, respecto a la *precisión*, con este sensor se puede tener gran precisión a la hora de determinar si hay un objeto sobre la celda, pero el sensor no es capaz de discernir cuál es el tipo de objeto (un vehículo, una persona, un animal, etc.).

Sensor Magnético (magnetómetro): El magnetómetro es capaz de medir la magnitud del campo magnético que lo cruza. El sensor entrega la magnitud del campo en los tres ejes cartesianos, X, Y y Z. Este sensor puede detectar un vehículo, ya que un automóvil perturba el campo magnético circundante debido al alto contenido metálico que lo constituye. Durante la fase de evaluación se utilizó un magnetómetro de referencia HMC5883L. Respecto a la *resistencia*, este sensor funciona correctamente incluso estando protegido de la intemperie, utilizando para su protección materiales no conductores. Respecto al *consumo*, el sensor, en estado de reposo, consume solo 2 μ A y, en medición continua, 100 μ A. En el caso de nuestro proyecto el sensor debería estar en constante trabajo. Por último, respecto a la *precisión*, la detección de un vehículo se realiza implementando un algoritmo más complejo (con relación al de los sensores anteriores). Sin embargo, la detección podría

ser más confiable, debido a que el sensor no reacciona con otros elementos (personas, animales, etc.).

En el Cuadro I se resume la evaluación desarrollada. El símbolo de ✓ indica si el sensor sí cumple el criterio y el símbolo ✗ indica si no lo cumple. El magnetómetro fue elegido para el uso en este proyecto debido a que tuvo una mejor evaluación con respecto a los otros dos sensores.

Cuadro I: Comparación de los sensores evaluados

	Resistencia	Consumo	Precisión
Mecánico	✓	✓✓	✓
Ultrasonido	✗✗	✓	✗
Magnetómetro	✓✓	✓	✓✓

IV. ALGORITMO PARA LA DETECCIÓN DE VEHÍCULOS

Para utilizar el magnetómetro como un dispositivo para la detección de automóviles en una celda, se requiere analizar las medidas del campo magnético en cada una de las componentes espaciales X, Y y Z. Tras un análisis preliminar, se concluye que si se instala el dispositivo de manera que el eje Z del sensor sea perpendicular al suelo, su medida será independiente a la orientación del campo magnético de la tierra. Por lo tanto el algoritmo de detección de automóviles procesa solo la información proveniente de este componente. La componente en Z siempre tendrá el mismo patrón de cambio sin importar cual sea la orientación del dispositivo.

Algorithm 1 Algoritmo de detección magnetómetro

Require: $base, margen, prom, c_z$
Ensure: $estado$

```

if ( $prom < base - margen$ ) &  $estado = libre$  then
   $espera(2s)$ 
  if  $c_z < base - margen$  then
     $estado \leftarrow ocupado$ 
     $espera(2s)$ 
     $nc_z \leftarrow c_z$ 
  end if
end if
if ( $prom > nc_z + margen$ ) &  $estado = ocupado$  then
   $espera(1s)$ 
  if  $base - prom < c_z < base + margen$  then
     $estado \leftarrow libre$ 
  end if
end if

```

El algoritmo 1 presenta la forma como se procesa la entrada y salida de un vehículo: primero, en una etapa de calibración, el dispositivo toma el valor base del campo ($base$) cuando el dispositivo inicia (se requiere que la calibración se realice en el lugar final de funcionamiento del dispositivo, sin ningún vehículo en la celda). El dispositivo pasa a estado en funcionamiento, donde se capturan y se promedian los últimos valores medidos del campo ($prom$), si este valor sobrepasa los márgenes preestablecidos ($margen$), se pasa a la segunda fase de la detección, después de una espera. Si el campo (c_z) se estabiliza fuera del rango determinado como base, se determina que un vehículo se ha estacionado

($estado \leftarrow ocupado$), y se establece un nuevo valor de base para el campo “ocupado” ($nc_z \leftarrow c_z$). Para determinar si el vehículo abandona la celda se usa la misma lógica anterior, si el campo no regresa a las márgenes iniciales es que aún hay un objeto (un vehículo) ocupando la celda.

La Figura 2 muestra la captura de datos del magnetómetro en cuatro momentos (A,B,C,D) que corresponden respectivamente a celda vacía (A), vehículo entrando (B), celda ocupada (C) y vehículo saliendo (D).

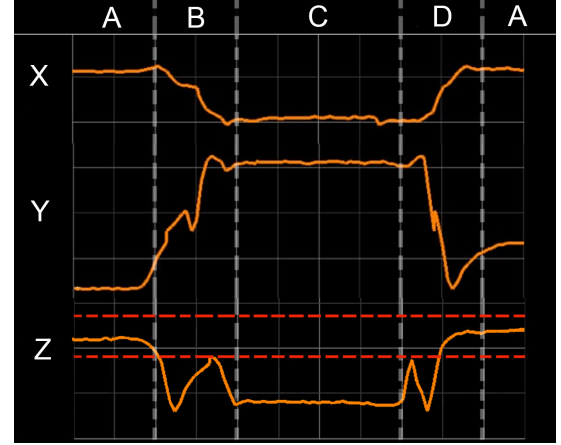


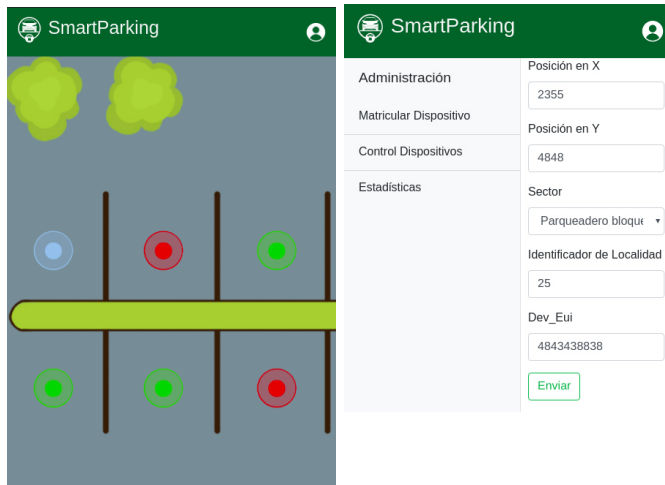
Figura 2: Cambio del campo magnético en el tiempo dada la interacción de un vehículo (se representan los márgenes de detección en el campo Z)

V. IMPLEMENTACIÓN SMARTPARKUDEA

En esta sección se describen los detalles de implementación del sistema SmartParkUdeA. Inicialmente, el sistema usa el servicio en la nube *The Things Network* (TTN) [6], el cual permite que nuestro nodo sensor se comunice con una red LoRaWAN. En el servicio de TTN se crea una aplicación y se inscriben los dispositivos que se van a comunicar con ella, este proceso se puede realizar también usando un servicio desde el *Backend*.

El *FrontEnd* fue desarrollado en *Angular*, un framework para aplicaciones web desarrollado en *typescript*, destinado a la creación de aplicaciones del tipo *Single Page Application*. Nuestra aplicación soporta dos tipos de usuarios: usuario común y administrador. El *usuario común* es un usuario del estacionamiento que, al ingresar a la página, visualiza las diferentes zonas de estacionamiento con su respectivo número de celdas libres. El *administrador* tiene acceso a la gestión de dispositivos, donde podrá registrar, eliminar y cambiar el estado de los mismos, además puede revisar las estadísticas de uso del estacionamiento en un sector en específico.

El *Backend* consta de un servidor de aplicación desarrollado en *Javascript* el cual se ejecuta en un entorno de ejecución *NodeJS*, se usó *ExpressJS* [7] como framework que facilita la creación de aplicaciones web con arquitectura Modelo, Vista, Controlador (MVC) [8]. Este framework nos permite crear los puntos finales de la API que es consumida por el administrador y por el usuario. Para la Base de datos se utilizó *MongoDB* [9],



(a) Estacionamiento (b) Administración

Figura 3: Vistas principales del *FrontEnd*

nos apoyamos en uso de la librería *Mongoose* para el mapeo objeto documento (ODM). En la base de datos los esquemas principales corresponden a la información de las celdas de estacionamiento, el histórico de ocupación, además de las estadísticas de uso que se generan semanalmente.

Como se mencionó anteriormente nuestra aplicación hace uso del servicio TTN, el cual se encarga de recibir los paquetes provenientes del *Gateway* y enviarlos a nuestro *Backend*. TTN provee una librería *Client* para establecer comunicación a través protocolo MQTT (Message Queuing Telemetry Transport), el cual es un protocolo de publicación-subscripción, que nos permite subscribir al *broker* expuesto por TTN, utilizando el tópic adecuado.

Nuestro sistema también implementa un servicio de notificación en tiempo real. La conexión con TTN permite informar al servidor inmediatamente cualquier cambio de estado en las celdas de estacionamiento. El sistema usa la librería *Socket IO* la cual permite implementar comunicación mediante Web-Sockets sobre el servidor de aplicación, creando una sala para cada zona de estacionamiento, donde se emiten los cambios en las celdas de dicha zona.

El prototipo del dispositivo fue desarrollado en una placa TTGO-T-Beam [10] dotada de el microcontrolador de bajo costo ESP32 que además soporta el protocolo LoRa. Por este medio se modulan ondas de radio a una frecuencia de 915 MHz, codificando una mínima cantidad de datos que son recibidos por el *Gateway* con conectividad a TTN.

En cada celda de estacionamiento se debe instalar un dispositivo protegido con una carcasa resistente a la intemperie y equipado con un sensor magnético capaz de detectar un vehículo entrando o saliendo de la celda. El dispositivo emite un mensaje de *keep alive* cada cinco minutos para indicar al sistema que sigue funcionando, también envía información cuando el sensor detecta un cambio de estado en la celda. Estas funcionalidades fueron desarrolladas usando la librería LMIC de IBM [11].

El sistema fue validado a través de una prueba funcional en un ambiente controlado, donde se pudo verificar el cumplimiento de las funcionalidades principales del sistema¹.

VI. CONCLUSIONES

En este artículo se presentó SmartParkUdeA, sistema IoT para el estacionamiento inteligente de vehículos en el campus de la UdeA. SmartParkUdeA fue desarrollado utilizando tecnologías del Internet de las Cosas (IoT), lo cual permite que el sistema tenga un bajo costo y sea fácilmente adaptable a cualquier entorno. SmartParkUdeA presenta una solución amigable para el usuario, el cual puede consultar el estado de las plazas de estacionamiento. Adicionalmente el sistema es capaz de recolectar información que puede ser utilizada por la administrador para mejorar la gestión del estacionamiento.

El sistema fue validado a través de una prueba funcional donde se evidenció la efectividad del sensor magnético como sistema de detección de vehículos. Adicionalmente, se puede concluir que la tecnología de comunicación LoRaWAN es adecuada para este tipo de sistemas, en los cuales se transmite poca información. Se hace notar que la falta de documentación y soporte para el dispositivo utilizado fue un obstáculo para el desarrollo (la implementación se desarrolló guiada por foros y proyectos publicados en línea).

Como trabajo futuro se espera validar el sistema en el entorno real, evaluando otras características del protocolo de comunicación como el consumo de potencia y la cobertura de las estaciones base, además de medir la resistencia a la intemperie del dispositivo.

REFERENCIAS

- [1] D. Zambrano, "Así se impulsa la movilidad sostenible en Medellín," Medellín, mar 2017. [Online]. Available: <https://www.elcolombiano.com/antioquia/estrategias-por-la-movilidad-sostenible-en-medellin-BI6138726>
- [2] J. Rico, J. Sancho, B. Cendon, and M. Camus, "Parking Easier by Using Context Information of a Smart City: Enabling Fast Search and Management of Parking Resources," in *International Conference on Advanced Information Networking and Applications Workshops*. IEEE, 2013, pp. 1380–1385.
- [3] V. Paidi, H. Fleyeh, J. Håkansson, and R. G. Nyberg, "Smart parking sensors, technologies and applications for open parking lots: A review," *IET Intelligent Transport Systems*, vol. 12, no. 8, pp. 735–741, 2018.
- [4] A. J. Wixted, P. Kinnaird, H. Larjani, A. Tait, A. Ahmadinia, and N. Strachan, "Evaluation of LoRa and LoRaWAN for wireless sensor networks," in *2016 IEEE SENSORS*, vol. 53, no. 9. IEEE, oct 2016, pp. 1–3.
- [5] N. Sornin, M. Luis, T. Eirich, T. Kramp, and O. Hersent, "LoRaWAN™ Specification," Tech. Rep., 2015.
- [6] The Things Network, "The Things Network Documentation," 2020. [Online]. Available: <https://www.thethingsnetwork.org/docs/>
- [7] OpenJS Foundation, "Express JS," 2020. [Online]. Available: <https://expressjs.com/>
- [8] C. Larman and B. M. Valle, *UML y patrones: una introducción al análisis y diseño orientado a objetos y al proceso unificado*, ser. Fuera de colección Out of series. Pearson Educación, 2003.
- [9] MongoDB Inc., "mongoDB," 2020. [Online]. Available: <https://www.mongodb.com/es>
- [10] Lilygo, "LILYGO® TTGO T-Beam V0.7," 2020. [Online]. Available: <http://www.lilygo.cn/>
- [11] IBM, "IBM LoRaWAN in C (LMiC) Technical Specification," *IBM Research Labs(LMiC)*, no. LMiC, 2015.

¹El video con la evidencia de la pruebas funcionales puede ser consultado en este enlace: <https://youtu.be/RsmKUX-2e1g>