**QUINDIO, TERRITORIO INTELIGENTE**

INGtelligent Group

**ESPACIO ACADÉMICO**

PROYECTO CDIO III

**DOCENTE**

JORGE IVAN MARIN HURTADO

**PROJECT MANAGER**

JOHAN SEBASTIAN MOSQUERA CAICEDO

**ESTUDIANTES:**

HEMERSON LEANDRO ALVAREZ RODRIGUEZ

YEIKOOL ANDRES BALLESTEROS PINEDA

SANTIAGO BENJUMEA HURTADO

JUAN JOSE BERMUDEZ LOPERA

ARMENIA, QUINDÍO

12/12/2021

**TABLA DE CONTENIDO**

[**PRUEBAS DE HARDWARE**](#_heading=h.eszuigeelob5) **3**

[AUTONOMÍA DE ALIMENTACIÓN DEL CIRCUITO EN LAS ZONAS AZULES.](#_heading=h.6ths528ifz3n) 4

[CORRECTO FUNCIONAMIENTO EN DIFERENTES ESCENARIOS.](#_heading=h.bnrt4cxqbvuu) 5

[FALLOS AL MANDAR LA INFORMACIÓN A LA BASE DE DATOS.](#_heading=h.ffb55uewzn1v) 6

[**PRUEBAS DE SOFTWARE**](#_heading=h.e5yw6royhzy0) **7**

[APLICACIÓN PARA ADMINISTRADORES DE PARQUEADEROS](#_heading=h.9xjzyjsp5t7n) 7

[APLICACIÓN PARA LOS USUARIOS](#_heading=h.bcp95apxtl41) 9

[**PRUEBAS DE INTEGRACIÓN DE LOS SISTEMAS**](#_heading=h.khe7t697aoc1) **10**

# 

A partir de los hitos planteados en la EDT, se puede concluir que se cumplió satisfactoriamente con el apartado de prototipo del hardware y software, en el cual se llevó a cabo el diligenciamiento de los requisitos funcionales y no funcionales para las diferentes partes de la solución planteada (tanto el circuito como las aplicaciones), a su vez se realizó una lluvia de ideas con el fin de obtener una idea concreta en cuanto al modelado de la aplicación se refiere, con lo que a partir de la idea elegida se realizó una especie de prototipo y se comenzó con el desarrollo de aplicaciones de prueba. Paralelo a esto se fue implementando el montaje del circuito (parte del hardware) empleado tanto para obtener información de la cantidad de cupos disponibles como para comunicarse esté dato a la base de datos Thingspeak.

Comenzando con el desarrollo de la pruebas, se han ido realizando pruebas en paralelo, ya que se ha probado obtener una comunicación estable tanto de parte del circuito como de la aplicación del administrador del parqueadero con la base de datos; y a su vez se ha probado obtener la información en tiempo real de la base de datos en la aplicación de los usuarios, obteniendo como resultado relevante que satisfactoriamente se obtienen ambos datos (el de los cupos censados por el circuito y los cupos totales determinados con la entrada o salida de vehículos).

# PRUEBAS DE HARDWARE

En la Figura 1 se observa el circuito implementado con el arduino uno, el módulo WiFi y el sensor infrarrojo.

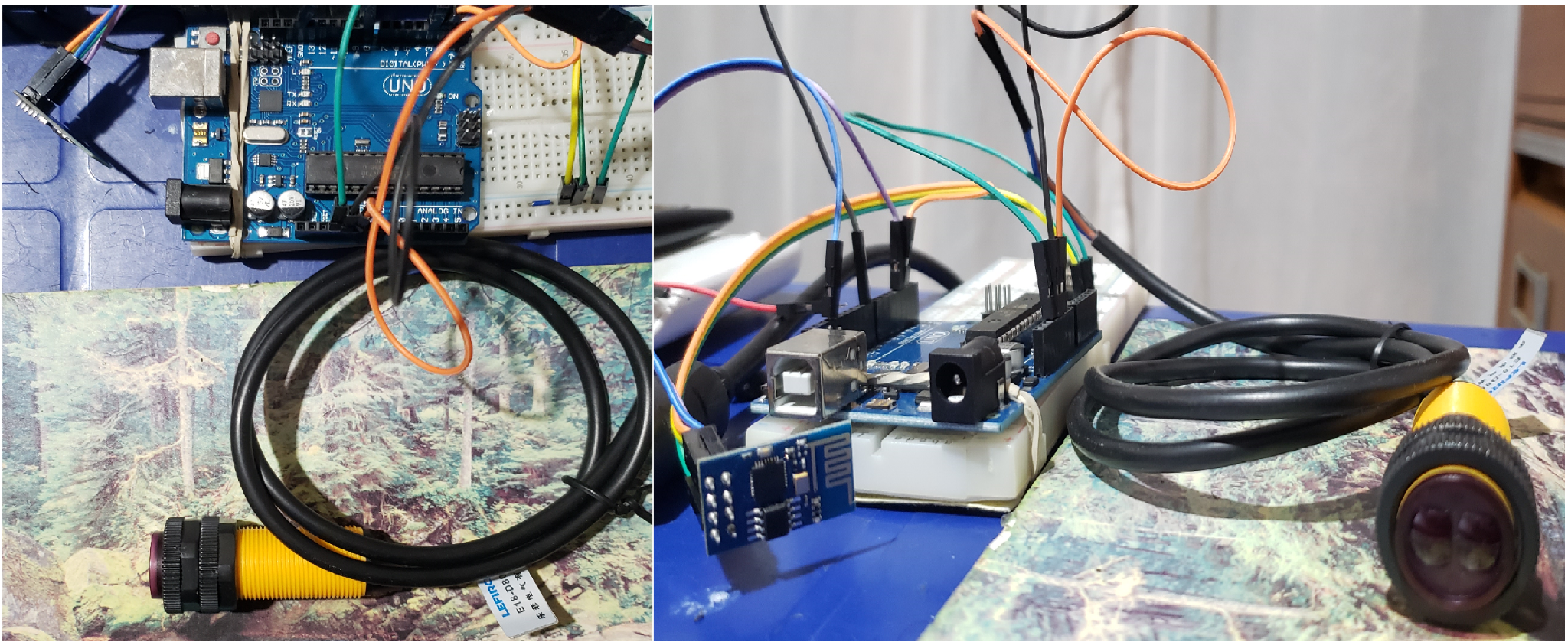


Figura 1. Circuito para las zonas azules.

En la Figura 2 se observa los datos enviados a la base de datos Thingspeak, la cual permite ser visualizada en una gráfica o el dígito con el número de cupos disponibles.

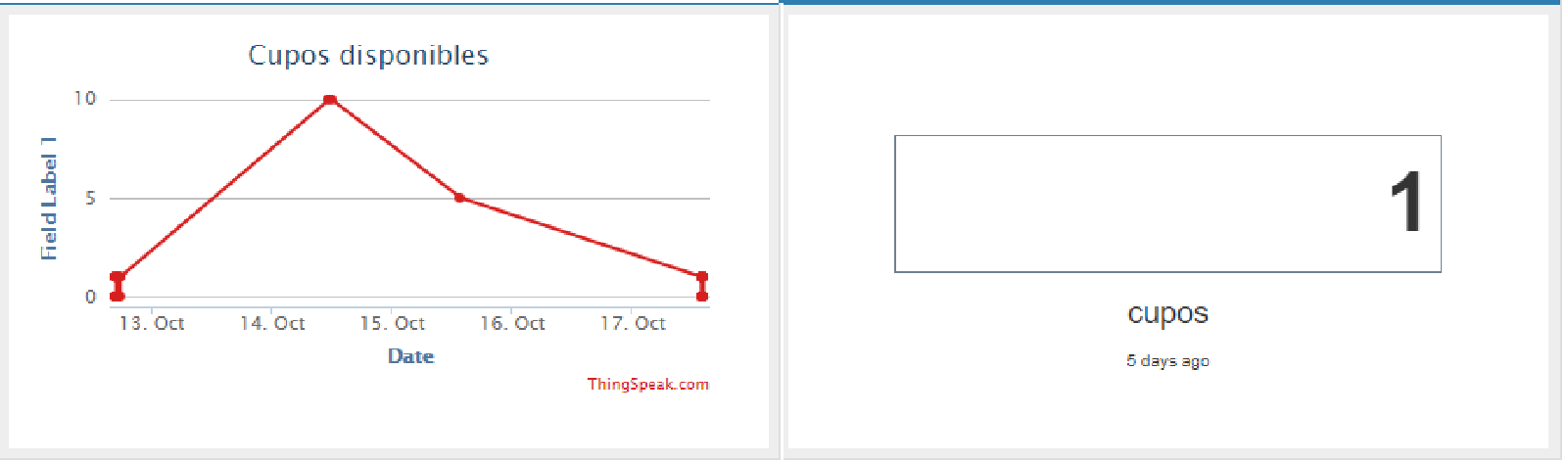


Figura 2. Información en Thingspeak.

## AUTONOMÍA DE ALIMENTACIÓN DEL CIRCUITO EN LAS ZONAS AZULES.

El circuito debe tener una autonomía suficiente para que garantice una duración de las baterías de más de 6 horas, con el fin de que el hardware funcione correctamente en la oscuridad, ya que los sensores y el módulo WiFi, deben seguir mandando información a la base de datos ThingSpeak.

Por este motivo, se implementó el uso de las baterías recargables, para garantizar un periodo de tiempo mayor al esperado. Además, los paneles solares se encargan de cargar las baterías para tener un correcto funcionamiento durante todo el día. En la Figura 3 se observa el montaje con los sensores y las pilas para verificar cuántas horas de autonomía tiene el circuito.



Figura 3. Prueba de duración de baterías.

Se tiene que las dos pilas cada una de 3.7 V con 8800 mAh garantizan 11 horas de funcionamiento, lo que permite que el circuito funcione correctamente por la noche o días nublados, cuando los paneles no estén cargando las baterías.

Por otro lado, se tiene que cuando las baterías están conectadas al panel, se tiene un funcionamiento del circuito de todo el día, lo que permitirá garantizar el correcto funcionamiento de todo el hardware para conocer el número de espacios disponibles en las zonas azules.

## CORRECTO FUNCIONAMIENTO EN DIFERENTES ESCENARIOS.

Se probó el circuito cuando se tienen diferentes condiciones climáticas (sol, lluvia) para asegurar que los sensores siguen funcionando correctamente, mandando una información correcta a la base de datos ThingSpeak. Se realizó las pruebas teniendo solo un sensor en el que se verificó teniendo un clima soleado que el sensor no mandara un dato incorrecto a la base de datos y que cuando detectara un vehículo si se mandara los datos a thingSpeak como se observa en la Figura 4.



Figura 4. Pruebas en día soleado.

Por otro lado, se verificó el funcionamiento teniendo un día lluvioso, por lo que se colocaron unas gotas de agua en el sensor para analizar si afectaba en la lectura del sensor o si este seguía funcionando correctamente, por lo que se observó que cuando el sensor tiene gotas de agua en el lente este no lo afecta mandando correctamente la lectura a la base de datos. Como se observa en la Figura 5.



Figura 5. Pruebas con gotas de agua en el sensor.

Se tiene que el circuito funciona correctamente con los dos climas, mandando correctamente la información a la base de datos ThingSpeak, por lo que se tiene que en días lluviosos o soleados los sensores siguen mandando correctamente la información.

## FALLOS AL MANDAR LA INFORMACIÓN A LA BASE DE DATOS.

El módulo WiFi permite mandar el número de espacios disponibles que se tienen en una zona azul a la base de datos ThingSpeak, pero se puede presentar que al momento de mandar los datos se presente una falla como se observa en la Figura 6.

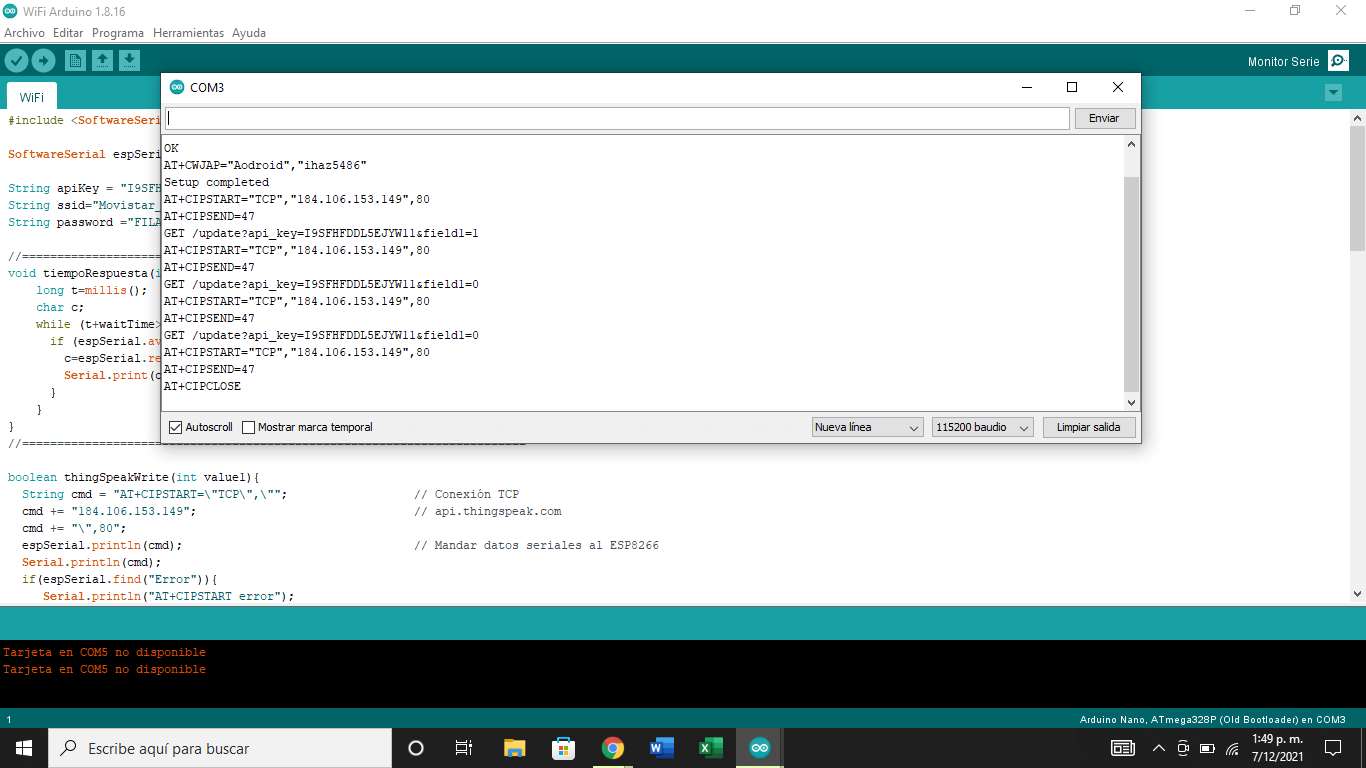


Figura 6. Fallos del envío de datos.

Se puede observar que cuando hubo un error al momento del envío de datos se muestra un mensaje “AT+CIPCLOSE” el cual nos indica que hubo un error al mandar los datos a ThingSpeak. Por lo que fue necesario la implementación de unas líneas de código como se observa en la Figura 7, para que cuando se presenta un fallo en el envío de la información a la base de datos, este vuelva a repetir el proceso de mandar los datos que se obtienen de los sensores después de 15 segundos, lo que permite garantizar que cuando se presente una falla al enviar los datos a ThingSpeak este vuelva a repetir este procedimiento después de 15 segundos y así tener una información correcta en la aplicación para los usuarios.



Figura 7. Código en arduino.

Al enviar un dato correctamente a ThingSpeak se tiene que en la variable “x” se guarda un estado “true”, pero si al mandar los datos a la base de datos hay un fallo se envía un estado “false”, por lo cual no entraría en el ciclo while, si no que entraría en la función if para esperar 15 segundos para volver a mandar la información a ThingSpeak, obteniendo correctamente el envío de datos como se observa en la Figura 8.

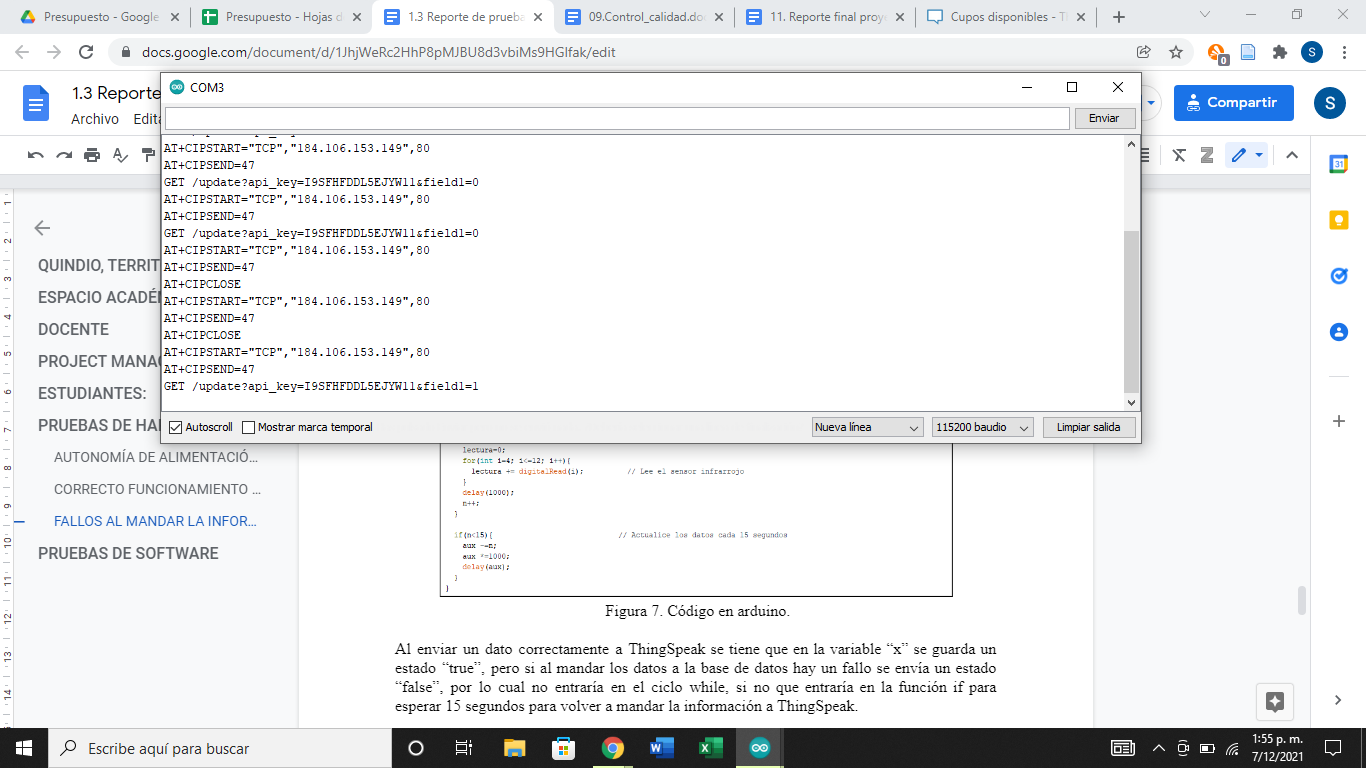


Figura 8. Correcto envío de datos.

Cuando se envía correctamente la información a la base de datos, se observa que muestra un mensaje con el API Keys de ThingSpeak seguido de un signo igual (=) y del valor que se está enviando.

# PRUEBAS DE SOFTWARE

## 2.1 APLICACIÓN PARA ADMINISTRADORES DE PARQUEADEROS

En la Figura 9 se observa el prototipo de la aplicación destinada a ser empleada por los administradores de los parqueaderos, en donde se encuentran dos botones los cuales permiten sumar o restar la cantidad de cupos disponibles totales en el establecimiento, dependiendo de la entrada o salida de un vehículo.



Figura 9. Prototipo de aplicación de los administradores

Se realizaron las respectivas pruebas y se encontró que al iniciar nuevamente la aplicación el valor de cupos disponibles se reiniciaba en el valor por defecto, por lo que se realizó un codigo por medio del cual al momento de reiniciar la aplicación se obtenga el último valor enviado a la base de datos el cual será el valor actual, sin que se reinicie el conteo. Por otro lado se implementó un botón cuya función es la de reiniciar el conteo definiendo el valor de cupos totales disponibles del establecimiento como el valor de cupos disponibles actuales. Además, de implementar un botón el cual indique las instrucciones para el adecuado uso de la aplicación por parte de los administradores de los parqueaderos. En la Figura 10 se evidencia el producto final de la aplicación destinada a los administradores.

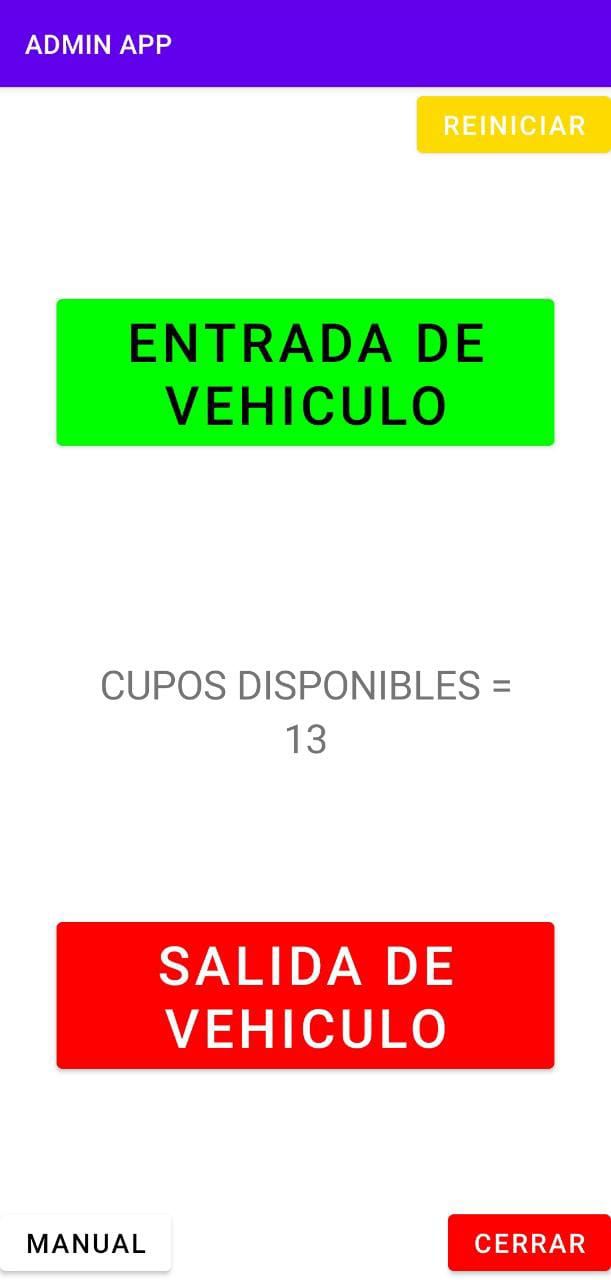


Figura 10. Producto final de la aplicación para administradores.

En este punto se empleó una condición en la cual el valor máximo de cupos disponibles no supere el valor de cupos totales del establecimiento, ni que sea menor a cero.

Ya que se implementó la aplicación dentro de la Universidad del Quindío, se realizó el conteo de cupos para vehículos, más específicamente carros, en donde se llegó a que la universidad cuenta en la zona norte con 178 cupos disponibles para parqueo de carros; cabe resaltar que contando hasta el bloque de economía, ya que el bloque de ingeniería alimentos está cerrado, por lo cual no hay paso de vehículos. Este número total de cupos disponibles está distribuido de la siguiente manera.

TABLA I

CUPOS TOTALES SECTOR NORTE UNIVERSIDAD DEL QUINDÍO

| LUGAR | CUPOS DISPONIBLES |
| --- | --- |
| Sotano B/. Economía | 15 |
| Entrada B/. Educación | 6 |
| Parqueadero contiguo B/. Educación | 61 |
| Parte posterior B/. Ingeniería | 25 |
| Costado Biblioteca | 17 |
| Frente entrada Capilla | 9 |
| Costado derecho entrada 4 (Sector Norte) | 45 |
| TOTAL | 178 |

## 2.2 APLICACIÓN PARA LOS USUARIOS

En la Figura 11 se observa el prototipo de la aplicación destinada para los usuarios, en donde se obtiene tanto información de los cupos disponibles en el lugar donde se encuentran los sensores como la información de los cupos totales disponibles otorgado por la aplicación del administrador del parqueadero.

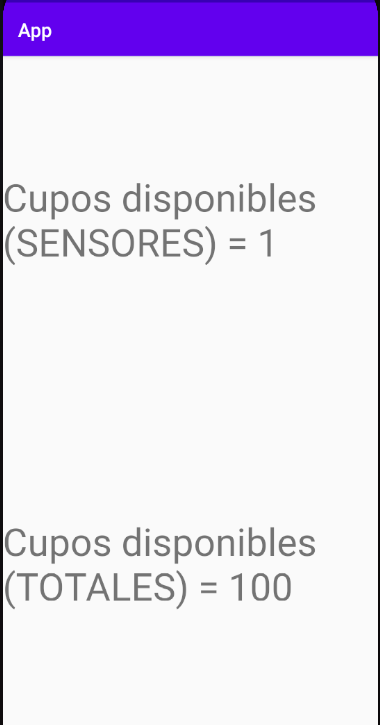


Figura 11. Prototipo de aplicación de los usuarios.

Una vez realizada la lectura correctamente de los datos de la base de datos, se procedió a implementar el mapa dentro de la aplicación para los usuarios.

En el mapa se debe colocar un marcador en la ubicación ya sea del establecimiento de los parqueaderos o del lugar en donde se encuentren implementados los sensores, esté marcador al ser presionado mostrará información relevante del respectivo lugar, como el nombre del establecimiento, o donde se encuentran los sensores (dirección) así como lo principal como lo es la cantidad de cupos disponibles en el momento de ser consultado. En las Figuras 12 y 13 se muestra la interfaz de la aplicación al iniciar y la interfaz del mapa con los diferentes marcadores ya sea de parqueaderos o de zonas azules (sensores).

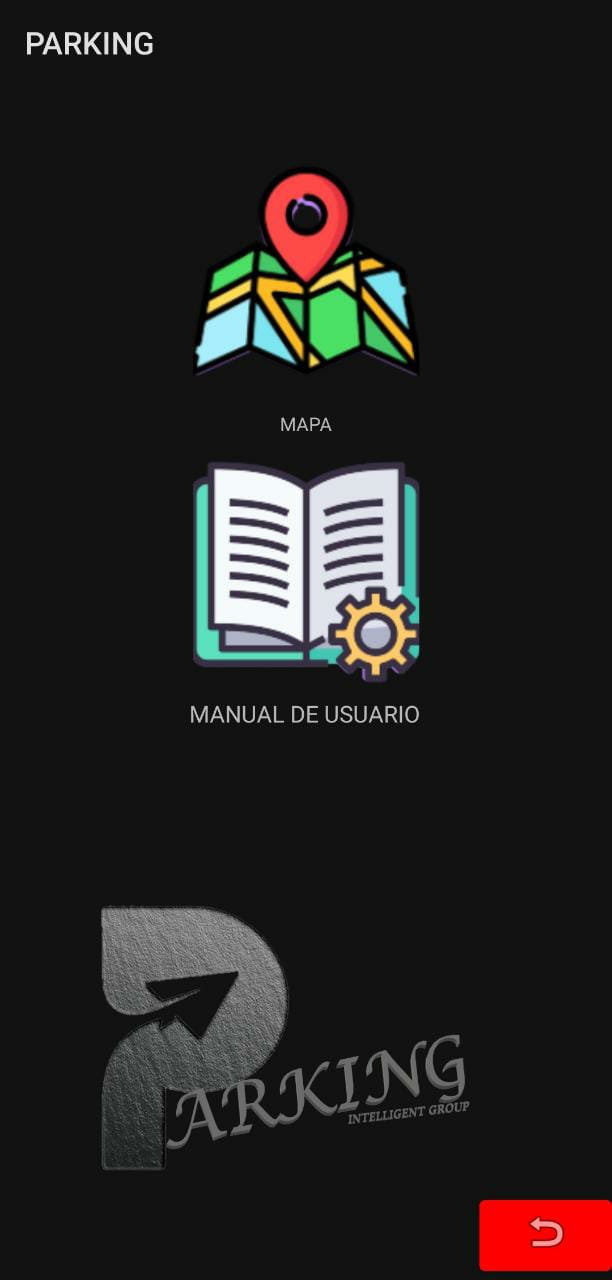


Figura 12. Interfaz inicial de la aplicación de los usuarios.



Figura 13. Interfaz del mapa con los marcadores.

# PRUEBAS DE INTEGRACIÓN DE LOS SISTEMAS

Como siguiente fase en la etapa de pruebas, se realizó la integración de los sistemas, implementando el circuito en la zona de parqueaderos al costado de la biblioteca en la Universidad del Quindío, en donde en cada espacio de parqueo de carros se debió colocar un sensor, tal y como se muestra en la Figura 14. Estos sensores deben ser puestos de tal forma que al momento en el que un carro tome el lugar del parqueadero, este lo pueda leer correctamente, tal y como se muestra en la Figura 15.



Figura 14. Implementación del circuito (sensores) en zonas de parqueo público.



Figura 15. Ubicación correcta de los sensores.

A partir de la correcta implementación del circuito se procedió a realizar la prueba de la lectura del valor arrojado por los sensores por medio de la aplicación de los usuarios, en donde en la Figura 16 se observa el dato leído de la base de datos.



Figura 16. Datos en la aplicación de los usuarios.

A partir de la Figura 14 se puede determinar que hay 5 espacios ocupados en el momento, y ya que se realizó la prueba con 5 sensores en esos lugares la cantidad de cupos disponible en el lugar es de cero (0) tal y como se muestra en la aplicación de los usuarios en la Figura 16.

Por otro lado, se realizó la prueba con la aplicación de los administradores, la Figura 17 muestra la aplicación de los administradores con su respectivo dato de cupos disponibles en el momento.



Figura 17. Uso de la aplicación de los administradores.

En la Figura 18 se puede observar el dato leído de la base de datos el cual fue enviado por la aplicación de los administradores y será mostrado a los usuarios por medio de la aplicación destinada para ellos.



Figura 18. Datos de la aplicación para los usuarios.

Cabe resaltar que la aplicación a ser empleada por los usuarios se diseñó para ser usada tanto en modo vertical como en modo horizontal, dependiendo del gusto y de cómo los usuarios suelen ubicar sus celulares al momento de manejar su vehículo. En las Figuras 19 y 20 se muestra la aplicación en vertical y horizontal respectivamente.



Figura 19. Aplicación en modo vertical.



Figura 20. Aplicación en modo horizontal.