



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

**CARRERA DE INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES Y
REDES**

**“DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO DE SEGURIDAD
PARA PARQUEADERO DE BICICLETAS, BASADO EN LA RED
CELULAR”**

Trabajo de Titulación

Tipo: Propuesta Tecnológica

Presentado para optar al grado académico de:

**INGENIERO EN ELECTRÓNICA, TELECOMUNICACIONES Y
REDES**

AUTOR: ANGEL ORLANDO CAIZA CHIMBORAZO

DIRECTORA: Ing. Mayra Alejandra Pacheco Cunduri

Riobamba – Ecuador

2020

©2020, Angel Orlando Caiza Chimborazo

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, Angel Orlando Caiza Chimborazo declaro que el presente trabajo de titulación es de mi autoría y que los resultados del mismo son auténticos y originales. Los textos que constan en el documento y provienen de otras fuentes están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación; El patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 15 de diciembre del 2020

ANGEL ORLANDO CAIZA CHIMBORAZO

C.I. 180421217-1

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA DE INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES Y
REDES

El Tribunal del trabajo de titulación certifica que: El trabajo de titulación: Tipo: Propuesta Tecnológica, **DISEÑO Y CONSTRUCCIÓN DE UN PROTOTIPO DE SEGURIDAD PARA PARQUEADERO DE BICICLETAS, BASADO EN LA RED CELULAR**, realizado por el señor: **ANGEL ORLANDO CAIZA CHIMBORAZO**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

	FIRMA	FECHA
Dr. Geovanny Estuardo Vallejo Vallejo. PRESIDENTE DEL TRIBUNAL	GEOVANNY ESTUARDO VALLEJO VALLEJO Firmado digitalmente por GEOVANNY ESTUARDO VALLEJO VALLEJO Fecha: 2021.01.28 11:06:34 -05'00'2021-01-27.....
Ing. Mayra Alejandra Pacheco Cunduri. DIRECTORA DEL TRABAJO DE TITULACION	MAYRA ALEJANDRA PACHECO CUNDURI Firmado digitalmente por MAYRA ALEJANDRA PACHECO CUNDURI Fecha: 2021.01.27 19:45:38 -05'00'2021-01-27.....
Ing. Franklin Geovanni Moreno Montenegro. MIEMBRO DEL TRIBUNAL	FRANKLIN GEOVANNI MORENO MONTENEGRO Firmado digitalmente por FRANKLIN GEOVANNI MORENO MONTENEGRO DN: cn=FRANKLIN GEOVANNI MORENO MONTENEGRO, o=SECURITY DATA S.A., c=ENTIDAD DE CERTIFICACION DE INFORMACION Motivo: Soy el autor de este documento Fecha: 2021-01-27 19:55:05.002021-01-27.....

DEDICATORIA

El presente proyecto es producto del esfuerzo en todo mi camino, empiezo agradeciendo primero a Dios por darme sus bendiciones, la paciencia y sabiduría para culminar mi carrera, a mis padres quienes me ha dado la vida, así también me han apoyado en el proceso de educación, pero con la constancia y dedicación he logrado uno de mis anhelados metas. Mi madre Etelvina y mi padre Juan Luis, pilares fundamentales, fuente de valores e incansables virtudes que levantaron mi entusiasmo luego de varias caídas, supieron demostrar que a pesar de las dificultades siempre debemos proyectar a un futuro exitoso lleno de metas cumplidas. A mis hermanos, Efraín, Olga, Hilda, Alciviades, Martha, Melida, que a pesar de la situación económica que vivimos me han contribuido con un granito de arena, para llegar a concluir mis metas, que siempre me has sabido brindar su consejo su apoyo en todo momento de mi carrera, sin importar la distancia han estado conmigo en todo momento. A mis amigos únicos en vida he irremplazables Byron, Abigail, Oscar, Marlon con quienes hemos formado un equipo de trabajo optimo, en frente de nosotros hemos tenido múltiples obstáculos, pero hemos sabidos superar trabajando de forma coordinada con una sola visión cumplir el propósito que hemos planteado al inicio de nuestras carreras.

Angel

AGRADECIMIENTO

Gracias a la Escuela Superior Politécnica de Chimborazo, por haberme acogido en sus aulas para formar un profesional de éxito en Electrónica en Telecomunicaciones y Redes, y haber cumplido con unos de tantos anhelados sueños de convertirme en un Ingeniero Electrónico, con capacidades profesionales, virtudes éticos y morales para que en el futuro cuente con los instrumentos necesarios para el desempeño en mi vida profesional.

A mis maestros, Freddy, Mónica, George, Ruth, Natalia y Verónica que más que mis maestros ha sido mis amigos, por haber formado un punto clave en la formación de mis conocimientos, sabidurías y experiencias, tanto académicas como profesionales, siempre proyectando y orientando a trabajar con honestidad, solvencia, impartiendo virtudes de trascendencia para el ser humano y la sociedad.

Angel

TABLA DE CONTENIDO

INDICE DE TABLAS.....	x
INDICE DE FIGURAS.....	xi
INDICE DE GRAFICOS.....	xiii
RESUMEN.....	xv
ABSTRACT.....	xvi
1. INTRODUCCION.....	1
2. MARCO TEÓRICO.....	7
2.1 Ciclo Parquadero.....	7
2.1.1. <i>Ciclo parqueaderos en el mundo</i>.....	7
2.1.2. <i>Tipo de Ciclo Parquaderos en el Ecuador</i>.....	8
2.1.3. <i>Requerimientos de los ciclos parqueaderos</i>.....	11
2.2 Tecnología de transmisión de datos.....	12
2.2.1. <i>GSM</i>.....	12
2.3 Tecnología GSM.....	12
2.3.1. <i>Gestión de Seguridad</i>.....	12
2.4 Tarjetas de desarrollo.....	13
2.4.1. <i>Arduino Mega</i>.....	13
2.4.2. <i>Raspberry PI</i>.....	14
2.4.3. <i>Intel Galileo</i>.....	14
2.4.4. <i>Sensores</i>.....	15
2.5 Registro de datos en la nube.....	16
2.5.1. <i>Ubidots</i>.....	16
2.5.2. <i>Phant</i>.....	16
2.5.3. <i>Thingspeak</i>.....	16

2.6	Virtuino	17
2.7	Wireshark	17
3.	METODOLOGÍA	18
3.1.	Sistematización del prototipo	18
3.2.	Análisis comparativo de las tarjetas de desarrollo	19
3.2.1.	<i>comparación de las tarjetas de desarrollo</i>	20
3.2.2.	<i>Opciones de almacenamiento en la nube</i>	20
3.3.	Arquitectura y requerimientos hardware del Sistema	21
3.3.1.1.	Sensor de proximidad	22
3.3.1.2.	Sensor de movimiento	23
3.3.1.3.	Cerradura electrónica.....	23
3.3.1.4.	Alarma sonora.....	23
3.3.2.	<i>Bloque de control</i>	25
3.3.3.	<i>Bloque de almacenamiento</i>	25
3.3.4.	<i>Bloque Monitoreo</i>	26
3.4.	Requerimientos de hardware	27
3.4.1.	<i>Shield SIM 900 GSM/GPRS</i>	27
3.4.2.	<i>Arduino Mega</i>	28
3.4.3.	<i>Sensores</i>	29
3.4.3.1.	<i>Ultrasónico HC-SR04(Proximidad)</i>	29
3.4.3.2.	<i>Sensor de movimiento PIR detector de presencia HC-SR501</i>	30
3.4.3.3.	<i>Zumbador</i>	31
3.4.4.	<i>Puente H L298N</i>	31
3.4.5.	<i>LCD</i>	32
3.4.6.	<i>Fuente de alimentación</i>	33
3.5.	Diagramas de conexión	35
3.5.1.	<i>Arduino mega y módulo SIM900</i>	35
3.5.2.	<i>Arduino mega y el sensor Ultrasónico HC-SR04</i>	35
3.5.3.	<i>Arduino mega y el PIR detector de presencia HC-SR501</i>	36

3.5.4.	<i>Arduino mega y el actuador electrónico</i>	36
3.6.	Software utilizado el ciclo parqueadero	36
3.6.1.	<i>Thingspeak</i>	37
3.6.2.	<i>IDE Arduino</i>	37
3.6.3.	<i>Wireshark</i>	37
3.7.	Diagrama de flujo	37
3.7.1.	<i>Aparcar</i>	37
3.7.2.	<i>Desaparcar</i>	38
3.7.3.	<i>Disponibilidad</i>	39
3.7.4.	<i>Aparcar sin móvil</i>	40
3.7.5.	<i>Desaparcar sin móvil</i>	41
3.8.	Diagrama de conexiones de elementos	42
4.	GESTION DEL PROYECTO	44
4.1.	Desarrollo de Pruebas	44
4.1.1.	<i>Análisis de funcionabilidad</i>	44
4.1.2.	<i>Menú principal</i>	45
4.1.3.	<i>Thingspeak</i>	52
4.1.4.	<i>Desarrollo de la aplicación móvil</i>	54
4.2.	Prueba de capacidad de acceso de red	56
4.3.	Análisis de tráfico de la conectividad	58
4.4.	Precisión y Rango de alcance de los sensores.	58
4.5.	Costo del proyecto	59
4.6.	Análisis de aceptabilidad	61
4.3.1.	Tabulación	62
5.	CONCLUSIONES	65
6.	RECOMENDACIONES	66

GLOSARIO

BIBLIOGRAFIA

ANEXOS

INDICE DE TABLAS

Tabla 1-3:	Comparativa de las diferentes tarjetas de desarrollo.....	20
Tabla 2-3:	Comparativa de las diferentes plataformas IoT.....	21
Tabla 3-3:	Tabla de verdad.....	24
Tabla 4-3:	Datos técnicos shield GSM SIM900.....	27
Tabla 5-3:	Tabla de características técnicas Arduino Mega.....	28
Tabla 6-3:	Tabla de características técnicas de sensor ultrasónico HC-SR04.....	30
Tabla 7-3:	Características de la fuente de alimentación.....	33
Tabla 8-3:	Conexión de pines entre Arduino Mega y SIM900.....	35
Tabla 9-3:	Conexión de pines entre Arduino Mega y el sensor Ultrasónico HC-SR04.....	35
Tabla 10-3:	Conexión de pines entre Arduino Mega y el PIR detector HC-SR501.....	36
Tabla 11-3:	Conexión de pines entre Arduino Mega y el Actuador electrónico.....	36
Tabla 1-4:	Tiempo de respuesta del servidor Thingspeak.....	56
Tabla 2-4:	Mediciones funcionalidad cinta métrica vs sensor ultrasónico.....	58
Tabla 3-4:	<i>Obtención del costo de los dispositivos electrónicos</i>	59
Tabla 4-4:	presupuesto del trabajo de ingeniería.....	60
Tabla 5-4:	Costo total del ciclo parqueadero implementado.....	61
Tabla 6-4:	Datos necesarios para determinar la muestra.....	61

INDICE DE FIGURAS

Figura 1-1:	Esquema del sistema.....	4
Figura 1-2:	Ciclo parqueadero	7
Figura 2-2:	Plata forma web BiciQuito.....	9
Figura 3-2:	Ciclo parqueadero Cuenca.....	10
Figura 4-2:	Ciclo parqueadero FADE-ESPOCH.....	10
Figura 5-2:	Formas de parquear improvisada en la FIE.	11
Figura 6-2:	Placa Raspberry Pi 2	14
Figura 7-2:	Placa Intel Galileo	15
Figura 1-3:	Diseño del sistema	19
Figura 2-3:	Bloque de sensores y actuadores.....	22
Figura 3-3:	Bloque de control	25
Figura 4-3:	Bloque de Almacenamiento.....	26
Figura 5-3:	Bloque de monitoreo	26
Figura 6-3:	Tarjeta de desarrollo Shield GSM SIM900.....	27
Figura 7-3:	Presentación física tarjeta de desarrollo Arduino MEGA.....	28
Figura 8-3:	Sensor ultrasónico HC-SR04	29
Figura 9-3:	Sensor de movimiento PIR detector de presencia HC-SR501	30
Figura 10-3:	Zumbador piezoeléctrico de 5v.....	31
Figura 11-3:	Puente H L298N	32
Figura 12-3:	Estructura de un LCD	33
Figura 13-3:	Batería de litio.....	33
Figura 14-3:	Diagrama de flujo de aparcamiento	38
Figura 15-3:	Diagrama de flujo del proceso de desaparcar	39
Figura 16-3:	Diagrama de flujo de disponibilidad.....	40
Figura 17-3:	Diagrama de flujo de aparcamiento sin móvil	41
Figura 18-3:	Diagrama de flujo del proceso desaparcar sin móvil	42
Figura 19-3:	Diagrama de conexión del dispositivo final	43
Figura 1-4:	Pantalla de inicio del dispositivo	44
Figura 2-4:	Iniciando sensores y GSM para mediciones.	45
Figura 2-4:	Menú Principal.....	45
Figura 4-4:	Slot's para Aparcar.	46
Figura 5-4:	Generación de clave para marcar el inicio de recepción en 20s.....	46
Figura 6-4:	Envío del código a Thingspeak.....	47

Figura 7-4:	Lectura del mensaje.	47
Figura 8-4:	Caso de coincidencia de código.	48
Figura 9-4:	Caso de no coincidencia de código.	48
Figura 10-4:	Slot's para desaparecer.	48
Figura 11-4:	Generación de clave para marcar el inicio de recepción en 20s.....	49
Figura 12-4:	Envío del código a Thingspeak.....	49
Figura 13-4:	Lectura del mensaje.	50
Figura 14-4:	Caso de coincidencia de código.	50
Figura 15-4:	mensaje de que no coincide el código.....	50
Figura 16-4:	Disponibilidad del Slot 1.	51
Figura 17-4:	Disponibilidad del slot 2.	51
Figura 18-4:	Opciones de manejo sin celular.	52
Figura 19-4:	Ingreso del código por teclado.	52
Figura 20-4:	Plataforma online Thingspeak.	53
Figura 21-4:	Pantalla principal.	54
Figura 22-4:	Pantalla Disponibilidad.....	55
Figura 23-4:	Pantalla Grafica de los eventos.	56
Figura 24-4:	Prueba de la medición del sensor.....	59

INDICE DE GRAFICOS

Gráfico 1-4:	¿Qué tipo de notificación en caso de intento de robo le parece segura?	62
Gráfico 2-4:	¿Le parece muy sencillo el manejo por teclado de la interfaz?	62
Gráfico 3-4:	¿qué tan seguro le parece el ciclo-parqueadero inteligente?	63
Gráfico 4-4:	¿Cómo estuvo su interacción con el ciclo-parqueadero?.....	63

ÍNDICE DE ANEXOS

ANEXO A. DATA SHEET SIM900

ANEXO B. DATA SHEET ARDUINO MEGA

ANEXO B. CÓDIGO FUENTE DEL SOFTWARE DE APLICACIÓN

RESUMEN

El presente trabajo de titulación tuvo por objetivo diseñar y construir un prototipo de seguridad para parqueadero de bicicletas, basado en la red celular. El prototipo está constituido por cinco bloques con funciones específicas: bloque de sensores, bloque de actuadores, bloque de control, bloque de almacenamiento y monitoreo. En primera instancia, en el bloque de sensores fueron implementados el sensor ultrasónico HC-SR04 y sensor de movimiento HC-SR501. El bloque de actuadores fue conformado por una alarma sonora, y una cerradura electrónica, elementos encargados de leer información sobre los eventos que se producen en el ciclo parqueadero y entregar esta información al bloque de control. En el bloque de control, se incorporó el arduino mega y el módulo GSM SIM900, en el arduino mega se realizó toda la programación del presente trabajo. El bloque de almacenamiento fue constituido por la Plataforma Thingspeak en la nube de servicio público y gratuito. En el bloque de monitoreo, constituido por el terminal inteligente con plataforma Android, se ejecuta la aplicación móvil en comunicación con la plataforma IoT, en el cual fueron programadas todas las funciones para la interacción de los usuarios con el ciclo parqueadero. Una vez diseñado, instalado y probado el sistema en el ciclo-parqueadero, se concluye que mediante la red celular con el uso de un módulo SIM900, los tiempos de respuesta promedio del servidor thingspeak son de 47,97ms prácticos y 3,32ms teóricos, con lo cual se da cumplimiento a los requerimientos establecidos. Se recomienda ampliar las funcionalidades del prototipo incorporando nuevas opciones como cámara ip, actuadores específicos y paneles solares.

Palabras clave: <TELECOMUNICACIONES>, <COMUNICACIONES MOVILES>, <PARQUEADERO INTELIGENTE>, <BICICLETAS >, <SEGURIDAD >, <INTERNET DE LAS COSAS (IoT) >, <RED CELULAR>.



0522-DBRAI-UPT-2020

ABSTRACT

The current research was aimed to design and build a security prototype for bicycle parking based on the cellular network. The prototype includes five blocks with specific functions: sensor block, actuator block, control block, storage, and monitoring block. Firstly, the HC-SR04 ultrasonic sensor and HC-SR501 motion sensor were implemented in the sensor block. The actuator block was made up of an audible alarm and an electronic lock; these elements are in charge of reading information about the parking cycle events and delivering this information to the control block. In the control block, the Arduino mega and the GSM SIM900 module were incorporated, all the programming was carried out in the Arduino mega. The Thingspeak Platform established the storage block in the free public service cloud. The mobile application is carried out in the monitoring block, which was made up of the intelligent terminal with the Android platform; this app is in communication with the IoT platform where all the functions were programmed to interact with the users with the parking cycle. Once the system was designed, installed, and tested in the cycle-parking lot, it is concluded that through the cellular network by using a SIM900 module, the average response times of the Thingspeak server are 47.97ms practical and 3.32ms theoretical, with which the established requirements are met. It is recommended to expand the prototype's functionalities by incorporating new options such as an IP camera, specific actuators, and solar panels.

Keywords: <TELECOMMUNICATIONS>, <MOBILE COMMUNICATIONS>, <INTELLIGENT PARKING>, <BICYCLE>, <SECURITY>, <INTERNET OF THINGS>, <CELLULAR NETWORK>.

1. INTRODUCCION

La definición de Smart Cities, es un tema que cada vez toma más fuerza y poder, en los temas de sostenibilidad y calidad de vida, debido al evidente crecimiento demográfico. En la actualidad el uso de la bicicleta cada vez es mayor, debido a los problemas de movilización (comodidad, tiempo de viaje, seguridad, etc.), a la conciencia de la ciudadanía y la conciencia del cuidado ambiental, cada vez más personas son conscientes del daño que se le está causando al planeta con los altos índices de contaminación.

Considerando el uso de la bicicleta como una gran opción de ayuda para el desarrollo sostenible y el auge las ciudades inteligentes, es importante la implementación de ciclo parqueaderos inteligentes, los cuales consisten en el almacenamiento de las bicicletas garantizando la seguridad de las mismas, y esto lo realizan mediante el uso de dispositivos electrónicos que permitan el reconocimiento de usuarios y de esta manera se evita el robo de las bicicletas.

En la Escuela Superior Politécnica de Chimborazo, se ha presentado un incremento en el nivel de robos de bicicletas en la institución, por lo que el diseño y construcción de un prototipo de seguridad para parqueaderos de bicicletas, es una alternativa que ayudará a dar solución a este problema. Existe la necesidad de ofrecer una alternativa de seguridad a los usuarios: estudiantes, docentes, personal administrativo y de servicio que hacen uso a diario de las bicicletas y de los ciclos parqueaderos. Al brindar la posibilidad de poder implementar un sistema de seguridad en un ciclo parqueadero, existe la posibilidad de mejorar opciones para almacenar una bicicleta y mejorar la seguridad de esta al ser respaldada por la tecnología GSM

El desarrollo de este prototipo se realiza en cinco etapas que se describe a continuación: Identificación del problema, identificar los requisitos, hardware y software, diseño del ciclo parqueadero, Implementación del prototipo, evaluación y validación de los resultados obtenidos.

En primeras instancias contiene la introducción, antecedentes y la formulación de la problemática. Luego, se desarrolla la investigación y redacción del marco teórico, el cual contiene información acerca de las tecnologías y topologías existentes para llevar a cabo la implementación del prototipo de seguridad del ciclo parqueadero. Se documenta además las características de los elementos, los mismos que son analizados para una posterior selección, tal es el caso de las tarjetas de desarrollo, y otros elementos que intervienen en el trabajo. En la tercera parte, se desarrolla el marco metodológico, en el que se detalla el diseño en software, la implementación del prototipo para lo

cual se considera los requerimientos necesarios para un ciclo parqueadero inteligente, además se considera las especificaciones técnicas de la tarjeta de desarrollo SIM900 y los sensores necesarios. Y finalmente, se muestra el análisis de los resultados obtenidos al realizar las pruebas de prototipo, los cuales fueron obtenidos a partir de pruebas en las instalaciones de la FIE-ESPOCH. La primera de ellas correspondiente al análisis de funcionalidad, la segunda correspondiente a pruebas de capacidad de acceso, análisis de tráfico de la conectividad, funcionamiento de los sensores, análisis de costos y finalmente la aceptabilidad del prototipo por parte de los usuarios.

ANTECEDENTES

En Bogotá, la universidad de Santo Tomas diseña un Sistema IoT de ciclo parqueadero inteligente, que permite el monitoreo de variables ambientales que afectan la movilidad en bicicleta y por el uso de un sistema de control de acceso, permite identificar las credenciales de los usuarios y asociarlos con la bicicleta aparcada. (Angulo, 2017, p. 66)

Se analizó el uso durante nueve días con resultados de sobre las 09:00 y las 14:00 horas, hay mayores índices de aparcamiento de bicicletas. Mientras que, a partir de las 14:00 hasta las 20:00 horas los usuarios suelen desaparecer sus bicicletas del ciclo parqueadero. (Angulo, 2017, p. 86)

En la Universidad Distrital Francisco José de Caldas, se diseñó un ciclo parqueadero inteligente constituido por tecnología RFID/NFC y aplicación móvil, con el estudio del entorno y el análisis de los grupos interesados se puede evidenciar que existe una organización, tiene un cubrimiento del 6% de la demanda, y que se presentan 4 incidentes de seguridad al mes, es decir, 48 al año. (Cuartas K, Cruzado J, 2016, p. 18)

En la Universidad Politécnica Salesiana en Guayaquil, se diseñó e implementó un parqueadero inteligente basado en Internet of Things (IoT) y constituido por procesadores arduinos, sensores, plataforma web y aplicación móvil, brindando información en tiempo real de los espacios disponibles en el parqueadero (Rosales L. 2016)

En la ESPOCH existe ciclo parqueaderos en la facultad de ciencias, rectorado y FIE, sin embargo, no cuentan con ningún tipo de sistema de seguridad física para las bicicletas de los usuarios. En este contexto específicamente en la facultad de informática y electrónica existe un ciclo parqueadero desde 2017, el cual cuenta con 6 slots, y son utilizados por los estudiantes, docentes y empleados

públicos. A diferencia de los ciclos parqueaderos existentes la contribución es la seguridad autónoma, implantada en el prototipo, podemos mencionar que cuenta con: notificaciones y registros mediante la tecnología GSM

FORMULACIÓN DEL PROBLEMA

¿Cómo diseñar y construir un prototipo de seguridad para parqueadero de bicicletas, basado en la red celular?

SISTEMATIZACIÓN DEL PROBLEMA

¿Cuáles son las características de los sistemas para seguridad en ciclo parqueaderos?

¿Cuáles son requerimientos del prototipo del ciclo parqueadero?

¿Cómo diseñar e implementar el prototipo y la APP para supervisar el ciclo parqueadero?

¿Cómo evaluar y validar el prototipo del ciclo parqueo, con los requerimientos planteados?

JUSTIFICACIÓN TEÓRICA DE LA INVESTIGACIÓN

En la actualidad existen muchos robos de vehículos, motocicletas y bicicletas en nuestro país siendo así necesario incrementar la seguridad a través de dispositivos electrónicos que permitan prevenir estos tipos de incidentes.

En el caso de la ESPOCH, el número de usuarios de medios de transporte es alto, y a pesar de contar con seguridad privada no es suficiente por la extensión de la institución. Parte de los estudiantes utilizan como medio de transporte las bicicletas, y al utilizar los ciclos parqueaderos disponibles en diversos puntos de la institución, deben llevar consigo su propio medio de bloqueo, como una cadena que es colocada para asegurar la bicicleta en los parqueaderos existentes.

Actualmente existe propuestas de ciclo parqueaderos que alerten a cada usuario registrado sobre la violación de la seguridad en los mismos, a través de comunicación en tiempo real utilizando mensajería instantánea, todo esto mediante el uso y manejo de herramientas tecnológicas que en

este caso permiten solucionar este tipo de problemas reales existentes en la sociedad específicamente en escenarios comunes como ciudades, instituciones educativas, etc. A pesar de ello dentro de la institución no existe actualmente implementado al momento del planteamiento de este trabajo ningún sistema de este tipo.

Por lo antes expuesto se propone una solución tecnológica, ante estos acontecimientos de inseguridad en ciclo parqueaderos, y se plantea el trabajo de titulación: “Diseño y Construcción de un prototipo de seguridad para el parqueadero de bicicletas basados en la red Celular” con la finalidad de evitar robos de bicicletas dentro de la institución. Con ello se busca reducir el índice de robos, brindar al propietario de la bicicleta un servicio ágil e inteligente; cabe recalcar que el presente proyecto está planteado acorde a las líneas de Investigación de la Escuela Superior Politécnica de Chimborazo, Todo esto basado en el cumplimiento del Objetivo 5 “Impulsar la productividad y competitividad para el crecimiento económico sostenible de manera redistributiva y solidaria” del Plan Nacional del Buen Vivir 2017-2021.

JUSTIFICACIÓN APLICATIVA DE LA INVESTIGACIÓN

La presente propuesta de titulación consiste, en el diseño y construcción de un prototipo de seguridad para el parqueadero de bicicletas basado en la red celular, mediante el uso de la tecnología GSM (Global System for Mobile communications), se estructura en cinco bloques como se detalla a continuación:

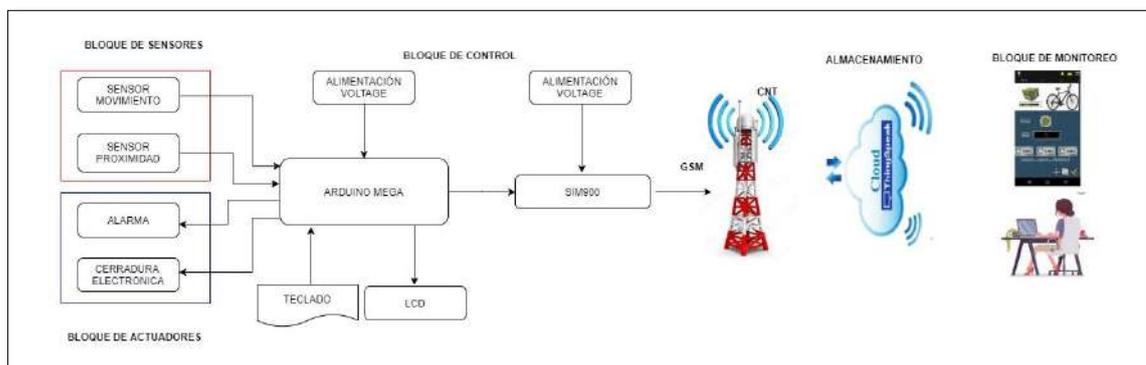


Figura 1-1: Esquema del sistema
Realizado por: Caiza Angel, 2020

Bloque de sensores

Está constituido por dos sensores: el sensor de movimiento, sensor de proximidad, estos dispositivos son los encargados de leer la información, para que los bloques siguientes

Bloque de actuadores

Está conformado por: una alarma sonora y un actuador electrónico.

Bloque de control

Está constituido por: el módulo GSM900 y la tarjeta de desarrollo. La lectura del sensor de proximidad identifica la disponibilidad del slot en el ciclo parqueadero, y la activación del actuador electrónico mediante el uso de mensajería, en caso de que exista movimiento, este ocupado y no exista la recepción de ningún mensaje, de darse el caso anterior se activará la alarma y se enviara un mensaje de notificación al usuario.

Bloque de almacenamiento

Está constituido por la plataforma Thingspeak, este es un servidor de servicio público y gratuito en la nube para investigadores y tesisistas con fines educativos. Se encarga del almacenamiento y los visualiza de manera gráfica al administrador. Los datos que van a ser mostrados son datos para el administrador y datos para el usuario, que en este caso será; la disponibilidad del slot del ciclo parqueadero, movimientos detectados, activación de la alarma y del actuador electrónico de seguridad.

Bloque de monitoreo

Se accede a la plataforma a través del teléfono celular o computadora el usuario podrá verificar la disponibilidad de un slot en el ciclo parqueadero. Mediante el envío de un mensaje, el usuario realizara la activación/desactivación del actuador electrónico. El administrador obtendrá registros de la plataforma como: tiempo de uso del parqueadero, intentos de robo, hora de activación de la alarma, bloqueo/desbloqueo del actuador electrónico de seguridad.

OBJETIVOS

OBJETIVO GENERAL:

Diseñar y Construir un prototipo de seguridad para parqueadero de bicicletas, basado en la red Celular

OBJETIVOS ESPECÍFICOS:

- Investigar los sistemas para seguridad en ciclo parqueaderos
- Estudiar requerimientos del prototipo del ciclo parqueo
- Diseñar e implementar el prototipo y la APP para supervisar el ciclo parqueadero
- Evaluar y validar el prototipo del ciclo parqueo, con los requerimientos planteados

2. MARCO TEÓRICO

El presente capítulo contiene información acerca de las tecnologías y topologías existentes para llevar a cabo la implementación del prototipo de seguridad del ciclo parqueadero, de acuerdo con los objetivos planteados. Se documenta las características de los elementos, los mismos que son analizados para una posterior selección.

2.1 Ciclo Parqueadero

Los estacionamientos de bicicletas, también conocidos como ciclo parqueaderos incluyen la infraestructura y equipamiento que consta de: la estructura física, candados de bicicleta y cadena, para habilitar el estacionamiento seguro, en especial en el ciclismo urbano. Es necesario que exista sistemas de parqueo manuales, automáticos y techos de protección. Como se ve en la figura 1-2.



Figura 1-2: Ciclo parqueadero

Fuente: <https://www.urbadep.com/wp-content/uploads/2014/06/Aparca-Bicicletas-Universal.jpg>

2.1.1. *Ciclo parqueaderos en el mundo*

Existen muchos parqueaderos en Europa que ofrecen seguridad al usuarios usando tecnología, entre estos sistemas de parqueaderos podemos hablar de algunos, biceberg es un aparcamiento automático y subterráneo para bicicletas, recibe y devuelve la bicicleta a pie de calle y además permite guardar otros elementos necesarios para su uso, como la mochila o el casco, el usuario realiza las operaciones mediante una tarjeta chip con código personal secreto, siguiendo un proceso tan sencillo como el de los cajeros automáticos (Garzón y Gómez, 2019, p. 19)

Bikeep es un portabicicletas que actualmente se usa en Bélgica, Islandia, Hungría, Suecia, Finlandia, Suiza, Holanda y Dinamarca, es una combinación de acero de grado industrial con tecnología

inteligente para bloquear de forma segura la rueda delantera y el marco de la bicicleta. Permite bloquear la bicicleta con un teléfono móvil o una tarjeta RFID. El portabicicletas ayuda a las empresas y áreas públicas a ser más amigables con los usuarios y por lo tanto atraen a más ciclistas. Puede configurarse con acceso restringido mediante una aplicación o una tarjeta de acceso, de modo que sólo personas específicas puedan utilizarlo(Garzón y Gómez, 2019, p. 66)

Se tomó como referencia estas empresas que están en uso en varios países de Europa y Asia, ya que cuentan con una estructura sólida y bien constituida

2.1.2. Tipo de Ciclo Parqueaderos en el Ecuador

BICIQUITO es una alternativa económica y ecológica para transportarnos en la ciudad: La Bicicleta Pública de Quito está disposición de los Quiteños y Quiteñas. Aprovechando de experiencias exitosas en otros países, el Municipio del Distrito Metropolitano de Quito, decidió ofrecer a sus ciudadanos y visitantes de un sistema de préstamo de bicicletas sin costo, denominado comúnmente como la bicicleta pública para promover el uso cotidiano de la bicicleta, BiciQuito nos brinda un servicio complementario de fácil y rápido transporte, para trayectos cortos dentro del perímetro urbano ubicado entre la Estación Norte del Trolebús, en La Y, hasta la Plaza de Santo Domingo, en el Centro Histórico. La Bicicleta Pública conocida en la ciudad como BiciQuito, es un sistema de servicio de movilidad no motorizado que opera en la capital desde agosto del 2012, y a partir del 24 de abril del 2017 la Agencia Metropolitana de Tránsito (AMT) opera este servicio. Además, no puedes quedarte con la bicicleta más allá de los 45 minutos de uso libre, si lo haces, como medida cautelar se bloqueará la Tarjeta BiciQuito y no podrás acceder al servicio. Si te sucede algún percance como: fallo mecánico, Robo, o retraso de entrega. Debes detallar en el formulario de constancia de documentos extraviados como referencia el carnet de BiciQuito bajo la categoría “Tarjeta Magnética de identificación de Instituciones Públicas y Privadas.(BiciQuito Bicicleta Publica 2012) como se ve en la figura 2-2



Figura 2-2: Plata forma web BiciQuito
Realizado por: Caiza Angel, 2020

Desde este lunes 1 de abril del 2019, los cuencanos podrán ingresar al sitio Web www.bicicuenca.com para adquirir su membresía, las bicicletas del nuevo sistema público de estacionamiento. El inicio de las operaciones está previsto para la segunda semana de abril. Así lo confirmó Adrián Castro, gerente de la empresa de Movilidad, Tránsito y Transporte (Emov EP). “Ya están instaladas las 20 estaciones, tenemos a disposición el aplicativo móvil y las 240 bicicletas”. La Emov EP y la empresa Bici Cuenca S.A. proveerá a la ciudad de 240 bicicletas en 20 estaciones. El uso será a través de un software de operación que permitirá el acceso a los ciudadanos a través de una aplicación móvil, página Web. En cada estación, el ciudadano puede solicitar la liberación de una bicicleta a través de su teléfono móvil o la tarjeta de proximidad. El sistema de auto atención libera la bicicleta, una vez que se activa una luz verde destellante. El uso es de 30 minutos, como máximo. La devolución de la ‘bici’ puede hacerla en cualquier estación. (240 bicicletas públicas entrarán en operación en 20 estaciones de Cuenca | El Comercio 2019) se observa en la figura 3-2



Figura 3-2: Ciclo parqueadero Cuenca
Fuente: Manuel Quizhpe/EL COMERCIO, 2019

En la actualidad existen muchos tipos de parqueaderos entre los que podemos mencionar los siguientes: ciclo parqueaderos con tubos cuadrados incrustados en una base la cual esta sujeta al piso mediante tacos y tornillos de cemento; en este tipo de parqueadero la bicicleta está asegurado mediante un candado manual y el usuario lleva la llave.



Figura 4-2: Ciclo parqueadero FADE-ESPOCH.
Realizado por: Caiza Angel, 2020

La bicicleta se ubica de forma horizontal, en este tipo de parqueadero, está suspendida a la llanta delantera y la seguridad va colocada en la de atrás,

En caso de no existir la estructura de ciclo parqueadero los usuarios buscan lugares en el cual se pueda sujetar ya sea una señalética de tránsito, un pasamano, un árbol hasta inclusive una banca con la finalidad de proteger la bicicleta. Como se observa en la figura 5-2



Figura 5-2: Formas de parquear improvisada en la FIE.
Realizado por: Caiza Angel, 2020

2.1.3. *Requerimientos de los ciclos parqueaderos*

Muchos son los requerimientos que un usuario necesita en lo que se refiere a la seguridad, en un ciclo parqueadero; específicamente hablando de los ciclos parqueaderos universitarios como en el presente estudio, es que los mismos no tienen ningún tipo de seguridad. Los usuarios deben llevar consigo una cadena o un candado para proteger del robo de su bicicleta. En vista de que los índices de robo han crecido en este tipo de ciclos parqueadero, lo que se requiere es satisfacer las necesidades de seguridad que el usuario necesite, pudiendo garantizarla de forma eficiente, para lo cual se considera las siguientes alternativas de solución:

Tecnología de comunicaciones: se considera una infraestructura de red celular, ya que tiene una estabilidad y cobertura eficiente, capaz de satisfacer la necesidad para la solución planteada (Garzon y Gómez, 2019 p. 36)

Sistemas de notificaciones: se considera necesaria la generación de notificaciones de alerta al usuario con respecto al estado de su bicicleta, la tecnología de comunicación que se escoja estará directamente relacionada con el tipo de notificaciones que se enviarán, debido a que el propósito que se desea es alertar al usuario lo más pronto posible, la notificación que se genere debe tener la capacidad de visualizarse tan pronto como la bicicleta se vea comprometido, en este orden de ideas, se debe plantear un mecanismo de notificación que presente el más mínimo número de requerimientos por parte del usuario (conectividad a internet, envío de SMS, recepción de llamadas) (Garzón y Gómez, 2019 p. 36)

Sistema de sensores: este sistema está constituido por varios dispositivos que cuando detecte cualquier alteración que se presente en el ciclo parqueadero y que este sea el indicador al sistema general que active el envío de alertas al usuario (Garzón y Gómez, 2019 p. 36)

Sistema de actuadores: se instalará un sistema de actuadores para garantizar la seguridad en el parqueadero, ya que servirá para aparcar y desaparcar la bicicleta.

2.2 Tecnología de transmisión de datos

2.2.1. GSM

El Sistema global de comunicaciones móviles (GSM) nace, a comienzos del siglo XXI, el estándar más usado de Europa. Se denomina estándar "de segunda generación" (2G) porque, a diferencia de la primera generación de teléfonos portátiles, las comunicaciones se producen de un modo completamente digital. En 1982, cuando fue estandarizado por primera vez, fue denominado "Groupe Spécial Mobile" y en 1991 se convirtió en un estándar internacional llamado "Sistema Global de Comunicaciones Móviles". En Europa, el estándar GSM usa las bandas de frecuencia de 900MHz y 1800 MHz. Sin embargo, en los Estados Unidos se usa la banda de frecuencia de 1900 MHz. Por esa razón, los teléfonos portátiles que funcionan tanto en Europa como en los Estados Unidos se llaman tribanda y aquellos que funcionan sólo en Europa se denominan bibanda. El estándar GSM permite un rendimiento máximo de 9,6 kbps, que permite transmisiones de voz y de datos digitales de volumen bajo, por ejemplo, mensajes de texto (SMS, Servicio de mensajes cortos) o mensajes multimedia (MMS, Servicio de mensajes multimedia) (Chamba, 2017, p. 25)

2.3 Tecnología GSM

La tecnología GSM (Global System for Mobile communications) es un sistema estándar referente a la telefonía móvil digital, y nos permite conectar dispositivos que cuenten con esta tecnología a nivel mundial. GSM se considera un estándar de segunda generación, debido a su velocidad de transmisión, ya que su extensión de tercera generación denominada UMTS ofrece mayor velocidad de transmisión, lo que hace posible la aplicación de internet de banda ancha (mejor conocido como internet 3G), además de emplear diferentes protocolos de radio como el W-CDMA. La onda de radio de GSM se encuentra disponible en diversas bandas dentro de los rangos de 824 MHz hasta 1,990 MHz. Una de las características más notables de esta tecnología es la utilización de tarjetas SIM, la cual es una tarjeta desmontable donde se almacenan datos referentes a la suscripción del usuario con la compañía telefónica, parámetros de la red y directorio telefónico. Por este medio se puede obtener el número de teléfono del usuario, y la compañía a la cual el usuario está activado (Padilla, Quintero y Díaz, 2016, p. 3)

2.3.1. Gestión de Seguridad

GSM mantiene la seguridad de extremo a extremo al mantener la confidencialidad de las llamadas y el anonimato del abonado de GSM. Con el fin de garantizar la privacidad de las comunicaciones GSM emplea mecanismos de autenticación y cifrados.

La autenticación GSM presenta una doble vertiente. Por un lado, para utilizar la terminal de usuario es necesario introducir un código de identificación personal PIN (Personal Identification Number) sin el cual es imposible el acceso a la SIM. Además, existe una autenticación a nivel de red entre esta y la SIM del cliente.

En cuanto al cifrado, su misión es evitar que un usuario no autorizado que haya interceptado la comunicación pueda acceder a la información.

Por otra parte, para evitar que la identidad del usuario viaje por el aire, GSM implementa un método de asignación de identidades temporales TMSI (Temporary Mobile Subscriber Identity), que es lo realmente se transmite. (Roldan, 2005)

2.4 Tarjetas de desarrollo

Son placas electrónicas que resuelven necesidades y facilitan soluciones para la creación en cualquier tipo de aplicaciones útiles, facilitando una compatibilidad muy alta con diferentes dispositivos y sensores. Con la evolución de tecnología se desarrolla distintos tipos de tarjetas de desarrollo, actualmente las más utilizadas son Arduino, Libelium-Waspote, Raspberry Pi, Galileo. (Noboa, 2019, p.33)

2.4.1. *Arduino Mega*

Arduino es una plataforma electrónica de código abierto basado en hardware y software fácil de usar. Las placas Arduino son capaces de leer las entradas, la luz en un sensor, un dedo sobre un botón o un mensaje de Twitter y convertirlo en una salida, la activación de un motor, encender un LED, publicar algo en línea. Se puede decir que su tablero qué hacer mediante el envío de un conjunto de instrucciones al microcontrolador en el tablero. Para ello se utiliza el lenguaje de programación de Arduino (basado en el cableado), y el software de Arduino (IDE), basadas en el procesamiento. (Chamba, 2017, p.28)

A través de los años Arduino ha sido el cerebro de miles de proyectos, a partir de objetos cotidianos a los instrumentos científicos complejos. Una comunidad mundial de los fabricantes, estudiantes,

aficionados, artistas, programadores y profesionales ha reunido en torno a esta plataforma de código abierto, sus contribuciones han añadido hasta una increíble cantidad de conocimiento accesible que puede ser de gran ayuda para los principiantes como para expertos. (Chamba, 2017, p.28)

2.4.2. *Raspberry PI*

Raspberry Pi2 se muestra en la figura 6-2, es una computadora de una sola tarjeta del tamaño de una tarjeta de crédito desarrollada en el Reino Unido por Raspberry Pi, fundación con la intención de estimular la enseñanza de informática básica en las escuelas. Tiene dos modelos; Modelo A tiene 256Mb de RAM, un puerto USB y no tiene conexión de red. El modelo B tiene 512Mb de RAM, 2 puertos USB y un puerto Ethernet. Eso tiene un sistema Broadcom BCM2835 en un chip que incluye un ARMI176JZF -S Procesador de 700 MHz, GPU Video Core IV y Una tarjeta SD. La GPU es capaz de reproducir con calidad de Blu-ray, usando H.264 a 40MBits / s. Tiene un núcleo 3D rápido al que se accede usando las bibliotecas OpenGL ES2.0 y OpenVG suministradas. El chip específicamente proporciona HDMI y no hay soporte VGA. (Jain, et al., 2014).

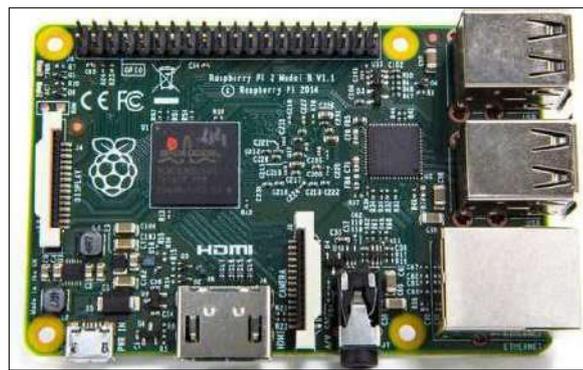


Figura 6-2: Placa Raspberry Pi 2

Fuente: (J. Pomeyrol, 2015, <http://www.muylinux.com/2015/02/02/raspberrypi2>)

2.4.3. *Intel Galileo*

Trabajan bajo la arquitectura Intel, es una herramienta muy interesante en el campo de la Domótica, tiene un procesador de bajo consumo y de bajo costo. Ejecuta un sistema operativo Linux, que contiene librerías de Arduino, es escalable y con un software reutilizable. Puede ser programada desde sistemas operativos Mac OS, Microsoft Windows y Linux. En la figura 7-2 se visualiza un ejemplo de placa Intel Galileo. (Noboa, 2019, p.35)

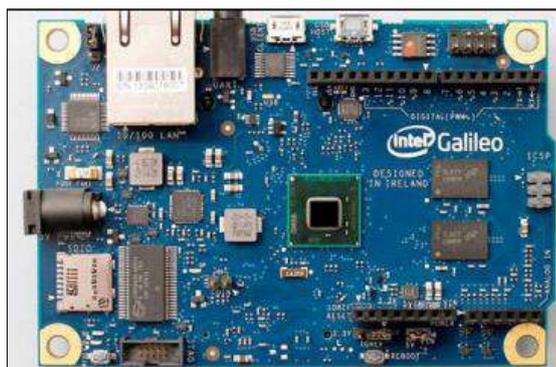


Figura 7-2: Placa Intel Galileo
Fuente: (GARCÍA, 2015, pág. 20)

2.4.4. *Sensores*

Sensores de movimiento

Los sensores de movimiento son dispositivos electrónicos que actúan cuando detectan movimiento en un área bajo supervisión, existiendo dos tipos según su forma de trabajo Sensor de movimiento infrarrojo y de ondas ultrasónicas para captar movimientos en tiempo real, estos sensores de movimientos por lo regular asociados a cámaras de seguridad, puertas en almacenes y centros comerciales son dispositivos que sobresalen en la seguridad electrónica teniendo en cuenta dos aspectos fundamentales tal como el tamaño como su funcionalidad.(«Sensores de movimiento: cómo funcionan y aplicaciones frecuentes | S&P» 2018)

Sensores de proximidad

Existen en el mercado sensores de presencia ultrasónicos, capaces de captar variaciones en el espacio a través de una onda ultrasónica que recorre el área y vuelve al detector rebotando en cada objeto que se encuentra en el ambiente; si el dispositivo detecta un nuevo objeto se activa. Los sensores por infrarrojos, por su parte, detectan la presencia cuando un cuerpo corta el haz que proyecta o a través de la variación que se produce en la temperatura. Un tercer tipo de sensores, serían los llamados sensores duales que son aquellos que combinan las dos tecnologías anteriores, es decir, los infrarrojos y ultrasonidos. Se utilizan en espacios donde es necesario un elevado nivel de detección.(«Aplicaciones más frecuentes de los sensores de movimiento» 2019)

Los detectores PIR Pasivo Infrarrojo, reaccionan sólo ante determinadas fuentes de energía tales como el calor del cuerpo humano o animales. Básicamente reciben la variación de las radiaciones infrarrojas del medio ambiente que cubre. Es llamado pasivo debido a que no emite radiaciones,

sino que las recibe. Estos captan la presencia detectando la diferencia entre el calor emitido por el cuerpo humano y el espacio alrededor. («¿Qué es un detector de movimiento pasivo?» [sin fecha])

2.5 Registro de datos en la nube

Hoy en día, las plataformas son un elemento importante en el desarrollo y la implementación de medios interconectados con la Internet. Las plataformas IoT facilitan el registro de datos recolectados por sensores o dispositivos electrónicos, el enrutamiento en tiempo real de la información y la monitorización de dispositivos. Cabe mencionar que cuenta con una interfaz en la que los usuarios pueden interactuar y visualizar en el sistema IoT, gracias a estas características el prototipo será estimado en una de estas tres plataformas IoT (Cantuña y Satian, 2019, p.37)

2.5.1. Ubidots

Es un servicio en la nube que permite recopilar y almacenar información extraída de sensores que pueden ser enviados desde Raspberry Pi, Arduino y HW, la misma que puede ser visualizada en tiempo real mediante una página web o aplicación móvil (UbiMaker, 2015). A más de eso tiene la capacidad de permitir a los administradores del sistema elaborar una imagen en tiempo real de los sistemas (Cantuña y Satian, 2019, p.37)

2.5.2. Phant

Desarrollada por SparkFun Electronics es una herramienta de código libre diseñada para la recopilación de datos de IoT. Para comenzar con el proceso de almacenamiento de datos se requiere de una clave privada y una pública para poder registrar la información en el servidor mediante cualquier servidor web. Pese a que presta un servicio gratuito tiene como restricción la capacidad máxima de 50MB. (Cantuña y Satian, 2019, p.37)

2.5.3. Thingspeak

Es una plataforma de análisis de IoT que tiene la capacidad de añadir y representar datos en la nube en tiempo real. Thingspeak provee graficas en vivo de los datos recopilados por dispositivos electrónicos. Con la habilidad de ejecutarse en la herramienta MaTLab, en el cual se puede realizar el procesamiento de la información. Las principales características de esta plataforma son las siguientes: Utiliza los principales protocolos de IoT, Visualiza datos en tiempo real, El análisis de IoT es automático por medio de horarios. (Cantuña y Satian, 2019, p.37)

2.6 Virtuino

Virtuino es una plataforma HMI para servidores IoT, Arduino ESP y tableros similares, corredores MQTT, PLC y servidores Modbus. Es compatible con la plataforma Thingspeak y no hay la necesidad de realizar códigos, su configuración se realiza a través de una interfaz visual y fácil manejo. Crea pantallas virtuales increíbles en el teléfono o tableta para controlar cada sistema automata a través del Bluetooth, WiFi o Internet Crea interfaces visuales con widgets como diodos leds, botones, interruptores, pantallas de valores, reguladores, instrumentos, etc. (Noboa, 2019, p.75)

2.7 Wireshark

Es una herramienta de mucha importancia para los administradores de sistemas o profesional de seguridad. Se trata de un software gratuito que permite analizar el tráfico red en tiempo real. Pero su particularidad es que a menudo es la mejor herramienta para solucionar los problemas de Red como la latencia o actividad maliciosa como intentos de piratería. No obstante, hablamos de una aplicación técnica, que requiere conocimientos de los conceptos básicos de redes (DEL POZO, 2019)

3. METODOLOGÍA

El presente capítulo detalla el diseño del software y hardware del prototipo, considerando los requerimientos para un ciclo parqueadero inteligente, se considera las especificaciones de la tarjeta de desarrollo y el módulo GSM SIM900. (Rodríguez, 2012) plantea que la metodología de la investigación se considera y se define como la disciplina que elabora, sistematiza y evalúa el conjunto del aparato técnico procedimental del que dispone la Ciencia, para la búsqueda de datos y la construcción del conocimiento científico. Considerando esta definición se utilizó el método comparativo para varios de los dispositivos utilizados en la implementación ya que se hizo un planteo de varias alternativas y de esto se seleccionó una que cuente con las características necesaria para solventar el requerimiento del prototipo a construir.

Se considero el método deductivo porque se debe considerar el análisis de factores como el consumo de energía, tamaño y disponibilidad de la red GSM, para elegir los equipos de manera correcta y así lograr el funcionamiento al momento de poner en marcha el prototipo.

Cabe indicar que el dispositivo es parte del ciclo parqueadero y la App funcionar en cualquier dispositivo móvil inteligente y con un entorno amigable para el usuario.

3.1. Sistematización del prototipo

El presente prototipo cuenta con cinco bloques, los cuales son: bloque de sensores, actuadores, control, almacenamiento y monitoreo, se explica a continuación.

El bloque de sensores y actuadores está conectado al bloque de control, y este a su vez se conecta a la tarjeta de desarrollo y al módulo GSM, se envía información a una base de datos en la nube. El dispositivo diseñado se aprecia en la figura 1-3 donde se observa la conexión de cada uno de los Bloques mediante GSM.

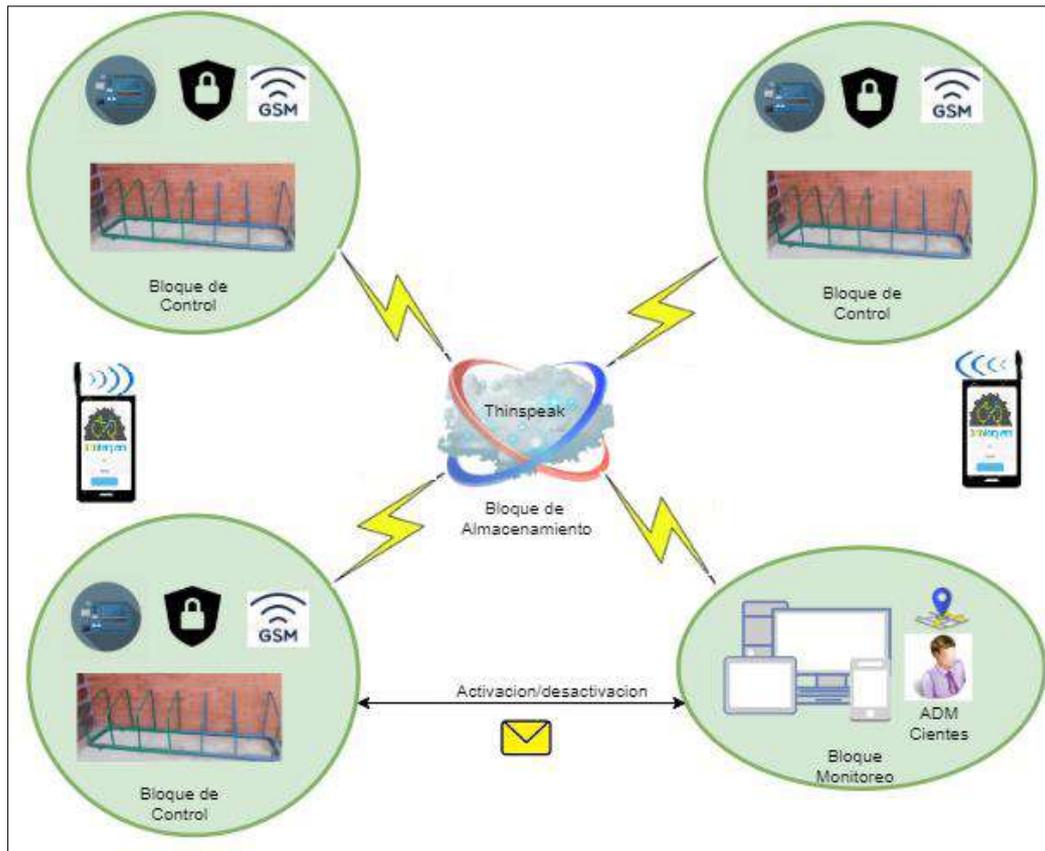


Figura 1-3: Diseño del sistema

Realizado por: Caiza Angel, 2020

Para el bloque de sensores y actuadores, se utilizó un sensor de proximidad Ultrasonico HC-SR0, este dispositivo se encarga de sensar la presencia o la ausencia de la bicicleta en cada slot del ciclo parqueadero, entonces se le asigna un uno de que la bicicleta este en el slot del ciclo parqueadero y un cero de que el slot del ciclo parqueo este vacío. También se utiliza un sensor de movimiento este se encarga de detectar la presencia de personas, quienes estén cerca del ciclo parqueadero. La obtención de estos datos para envío/recepción de la información del prototipo se utilizó la red celular GSM, con la finalidad de tener conectividad permanente ya que si optamos por otras tecnologías se presentó varios inconvenientes de conexión. GSM se encarga de realizar tres funciones:

- a.- **Mensajería:** establecemos el envío de código de seguridad, notificación de robo.
- b.- **Llamada de voz:** hace una llamada al usuario en caso de emergencia o intento de robo.
- c.- **Conexión de Datos:** Envío de paquetes de información recolectados por los sensores y activación/desactivación de los actuadores (alarma y candado).

3.2. Análisis comparativo de las tarjetas de desarrollo

3.2.1. comparación de las tarjetas de desarrollo

se realizó el análisis comparativo de Intel Galileo, Raspberry Pi2 y Arduino Mega, se muestra en la tabla 1-3.

Tabla 1- 3: Comparativa de las diferentes tarjetas de desarrollo.

	Intel Galileo	Raspberry Pi2	Arduino Mega
Procesador	SoC Quark X100	Broadcom BCM2836 ARM Cortex-A7	ATMega328
Memoria RAM	512 KB	1 GB	256 KB
Voltaje de entrada	5 V	5 V	7 – 12 V
Voltaje de Operación	3.3 V / 5 V	3.3 V / 5 V	5 V
E/S Analógicas	6	-	16
E/S Digitales	14	8	54
Memoria Flash	8 MB	-	32 KB
USB	2	4	1
Ethernet	10/100	10/100	No contiene
Precio	\$90	\$50	\$30
Entorno de desarrollo	Arduino IDE	Linux, IDLE, OpenEmbedded, QEMU, Scratchbox, Eclipse, Windows.	Arduino IDE
Velocidad	400 MHz	900 MHz	16 MHz
Sistema Operativo	Linux, Windows	Distribuciones de Linux, Windows 10.	Ninguno
Compatible sim900	N/A	Modular	Modular

Realizado por: Caiza Angel, 2020

De la comparativa realizada se optó por trabajar con el arduino mega por el requerimiento de puertos digitales y el costo económico.

3.2.2. Opciones de almacenamiento en la nube

En la tabla 2-3 se realiza una comparativa entre las plataformas revisadas, se detalla características de cada una como: compatibilidad con los distintos protocolos y lenguajes de programación, capacidad de integración con equipos de hardware y aplicaciones para la implementación del sistema de monitoreo.

Tabla 2-3: Comparativa de las diferentes plataformas IoT.

Parámetros	Thingspeak	Ubidots	Phant
Protocolos	HTTP	HTTP, MQTT, TCP, UDP o Parse	HTTP
Lenguajes de programación	HTML, CSS y JavaScript	HTML, JavaScript	Java, C++, Python, IOS, Android, Javascript
Integración de hardware	Arduino, Spark, Raspberry Pi, Electronic Imp	Arduino Raspberry Pi NodeMCU	Arduino

Realizado por: Caiza Angel, 2020

Del análisis de la tabla 2-3 se optó por trabajar con Thingspeak por su versatilidad de trabajo con arduino

3.3. Arquitectura y requerimientos hardware del Sistema

Habiendo establecido los bloques que conforman en este proyecto, se los identifican como indispensables, es así que no pueden funcionar de manera individual, un bloque depende de los otros y son los siguientes: sensores, actuadores, control, almacenamiento y monitoreo.

3.3.1. *Bloque de sensores y actuadores*

Este bloque consta de cuatro dispositivos, dos sensores y dos actuadores, las cuales fueron elegido según los requerimientos del sistema que se implementó.

Para que los usuarios tengan confianza en la seguridad ofrecida, se utilizó dispositivos capaces de sentir, de una manera ágil. las mediciones de los sensores, estos dispositivos emiten la medida de voltaje mínima y máxima respectivamente.

En la figura 2-3 de boques se presenta los sensores implantados en el bloque de control.

- Sensor de proximidad
- Sensor de movimiento

- Actuador electrónico (cerradura)
- Alarma (alerta de seguridad sonora)

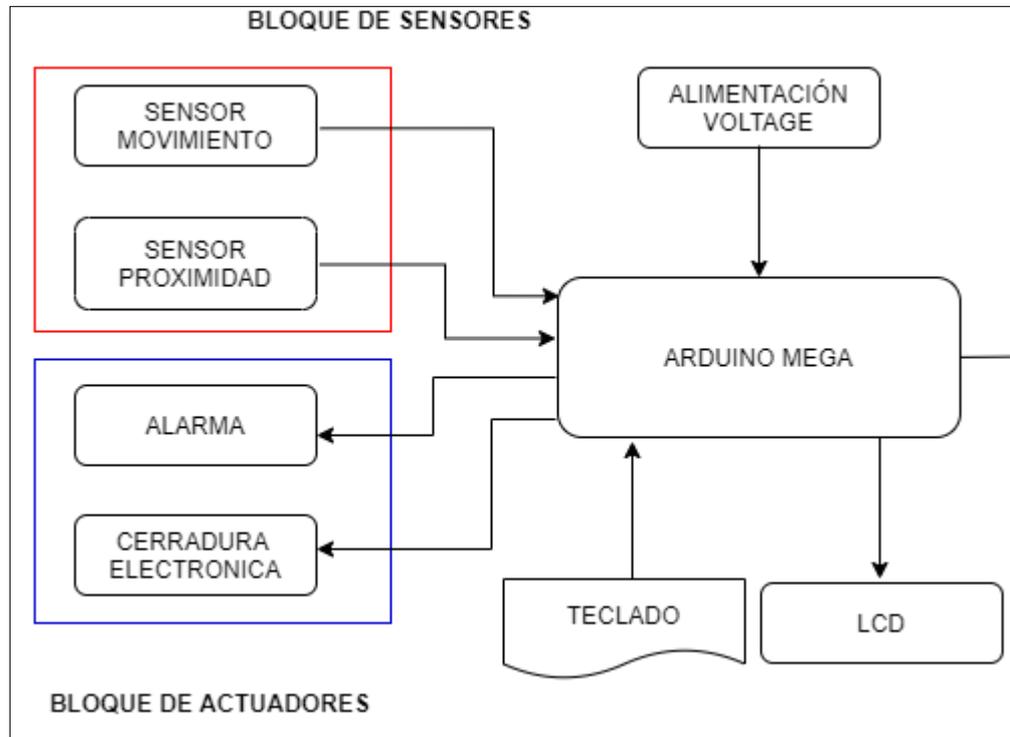


Figura 2-3: Bloque de sensores y actuadores
Realizado por: Caiza Angel, 2020

3.3.1.1. *Sensor de proximidad*

El sensor Ultrasónico HC-SR04, se utiliza para medir distancias o superar obstáculos sin entrar en contacto con estos, entre otras aplicaciones. Esto se consigue enviando un ultrasonido desde uno de sus cilindros que compone el sensor y el otro cilindro recoge el rebote del sonido sobre un objeto, este sensor tiene un rango de distancias de 3cm a 3m con una precisión de 3mm. (Barbus, 2014).

Este sensor se programó para que cuando la bicicleta este a una distancia de 0.5 cm, se le dé un aviso al sistema, que el slot este ocupado y si no existe a esta distancia el slot se considere disponible Aprovechando que la velocidad en el aire de este ultrasonido es de $340 \frac{m}{s}$, o $0,034 \frac{cm}{u-seg}$. Siendo la velocidad igual a la distancia recorrida en un determinado tiempo tenemos que $v = \frac{d}{t}$; donde obtenemos $d = v \cdot t$, siendo v constante y t el tiempo devuelto por el sensor a la placa Arduino, es necesario dividir el resultado entre *dos* dado que el tiempo recibido es de ida y vuelta. (Barbus, 2014)

3.3.1.2. *Sensor de movimiento*

El HC-SR501 es un sensor de movimiento, tiene un rango de detección de movimiento con un alcance de 3 hasta 7 metros de distancia, y aperturas del PIR de hasta 90 y 110°. Eso es un buen rango, permitiendo que se pueda instalar donde lo necesite, como un muro, techo y suelo, el sensor PIR está cubierto por una especie de cúpula blanca, eso es lo que se conoce como lente de Fresnel. Se llama así por el inventor, físico francés Augustin-Jean Fresnel. Gracias a él se permite construir lentes de gran apertura y corta distancia focal sin el peso y volumen del material que debería usarse con una lente convencional.

La función principal de este sensor es monitorear la presencia de cualquier individuo que esté cerca del ciclo parqueadero

3.3.1.3. *Cerradura electrónica*

Los actuadores son dispositivos que incorporado un motor eléctrico y un reductor que permite accionar cualquier dispositivo para llevar a cabo determinada acción o movimiento. Este dispositivo funcionará con el envío/recepción de un SMS utilizando la red celular GSM, esta cerradura se abrirá cuando, el código generado sea el correcto, se reciba un SMS, no haya proximidad y no exista movimiento, o el código generado sea el correcto, se reciba un SMS, no haya proximidad y exista movimiento, o el código generado sea el correcto, se reciba un SMS, haya proximidad y exista movimiento. Esto implica que si no cumple con los parámetros establecidos la cerradura permanecerá cerrada, para un mejor entendimiento se observa en la tabla 3 -3

3.3.1.4. *Alarma sonora*

Una alarma es un elemento muy empleado en la seguridad, y aunque si bien el mismo no evita por completo que se suceda alguna situación anormal resulta muy útil para prevenirla, Su funcionamiento consiste en el envío de una señal de alerta al bloque de control, luego, ésta emitirá la comunicación pertinente con el lugar y de no hallar respuesta enviará al personal correspondiente de seguridad.

Este dispositivo emitirá alerta sonora con la finalidad de alertar la presencia de antisociales, personas ajenas que estén por alrededores, intento de abrir la seguridad del bloqueo, para evitar robos en esta localidad.

Tabla 3-3: Tabla de verdad

	A	B	C	D	S1	S2
0	0	0	0	0	0	0
1	0	0	0	1	0	0
2	0	0	1	0	0	0
3	0	0	1	1	1	0
4	0	1	0	0	0	0
5	0	1	0	1	1	0
6	0	1	1	0	0	0
7	0	1	1	1	1	0
8	1	0	0	0	0	0
9	1	0	0	1	0	0
10	1	0	1	0	0	0
11	1	0	1	1	1	0
12	1	1	0	0	0	1
13	1	1	0	1	0	1
14	1	1	1	0	0	0
15	1	1	1	1	0	1

Realizado por: Caiza Angel, 2020

En la tabla 3-3 la tabla de verdad que consta de 16 estados, se presenta todos los posibles escenarios que puede ocurrir, se consideró como variables: el código generado = A, SMS receptado = B, sensor proximidad = C, sensor movimiento = D y para las salidas S1 es la alarma y S2 la cerradura (actuador)

Para obtener la ecuación de activación/desactivación de la alarma, se consideró varios escenarios utilizando los mapas de karnaugh se simplificó y se obtuvo la siguiente ecuación $S1 = \bar{B}CD + \bar{A}BD$

S1 funcionara únicamente cuando no se recepta la señal, hay presencia de proximidad y hay movimiento o no coincide el código generado, se recepta la señal de proximidad, y existe movimiento. Esta es la función simplificada que se consideró, para la programación de los Bloques de control.

Para la función del cantado $S2 = ABC\bar{C} + ABD$ el funcionamiento es de la siguiente manera: el código generado es correcto, se recepta un SMS y no existe proximidad, o el código generado es correcto, se recepta un SMS y existe movimiento.

3.3.2. *Bloque de control*

Este bloque está constituido por el arduino mega y el módulo GSM, este bloque lee el estado de los sensores y dependiendo de ello activar determinado actuador. A través del sensor de proximidad se establece la disponibilidad del slot del parqueadero y la activación del candado mediante el uso de mensajería,

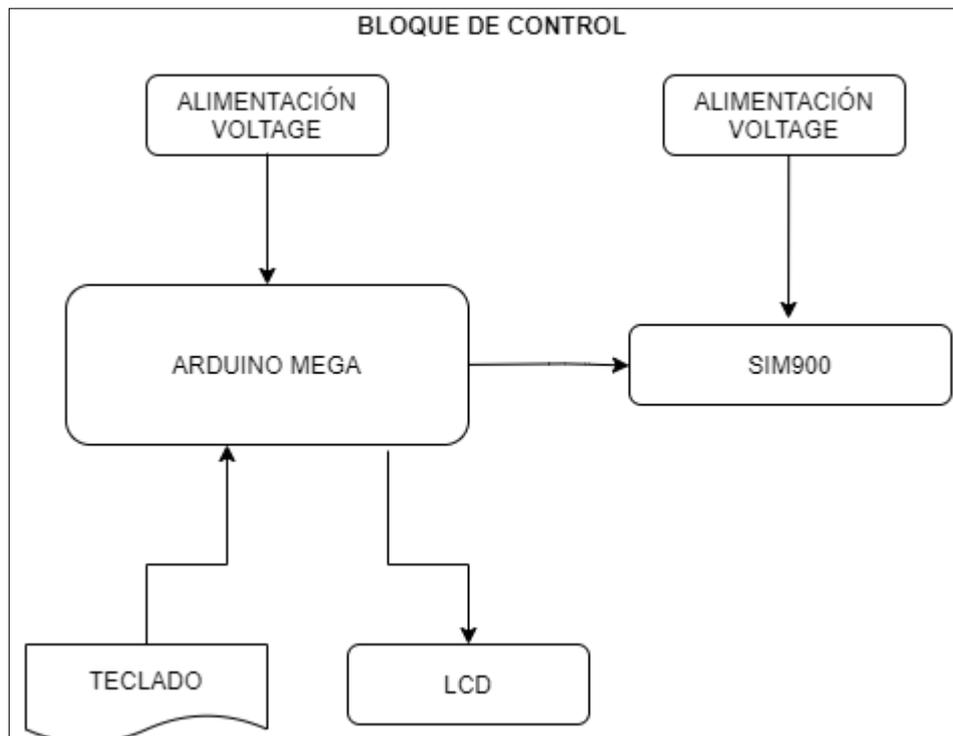


Figura 3-3: Bloque de control
Realizado por: Caiza Angel, 2020

3.3.3. *Bloque de almacenamiento*

Está constituido por la plataforma Thingspeak en la nube de servicio público y gratuito, se encarga del almacenamiento de datos, los registros que se almacenan son: tiempo, estado de disponibilidad, movimiento, activación del alarma y candado del parqueadero como se muestra en la figura 4-3



Figura 4-3: Bloque de Almacenamiento
 Realizado por: Caiza, Angel. 2020

3.3.4. *Bloque Monitoreo*

A través del acceso a la plataforma desde un celular o computador, el usuario podrá verificar la disponibilidad de un espacio en el ciclo parqueadero, mediante el envío de un mensaje realizar la activación/desactivación del candado. El administrador obtendrá registros de la plataforma como: intento de hurto, hora de activación de la alarma, bloqueo/desbloqueo del candado. Se ha especificado la configuración de la plataforma para usuario clientes y administrador. estudiantes, docentes y empleados, se consideran usuarios clientes. Como se observa en la figura 5-3



Figura 5-3: Bloque de monitoreo
 Fuente: Caiza, Angel. 2020

3.4. Requerimientos de hardware

3.4.1. *Shield SIM 900 GSM/GPRS*

Este módulo de desarrollo como el SIM900 es capaz de transmitir datos, a través de la red celular, ya que, para este trabajo de titulación, fue seleccionado por sus características técnicas, su cómodo costo, mayor alcance. Además, cuenta con los servicios que se requiere: mensajería, llamada de voz y llamada de datos. Se muestra físicamente en la Figura 6-3

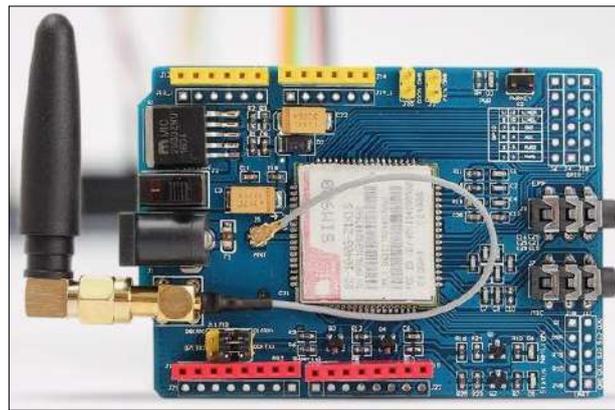


Figura 6-3: Tarjeta de desarrollo Shield GSM SIM900
Realizado por: Caiza, Angel. 2020

Características técnicas del módulo SIM900. Tabla 4-3

Tabla 4-3: Datos técnicos shield GSM SIM900

Fabricante	Simcom
Banda	GSM de 850, 900, 1800 y 1900 MHz
Potencia de Transmisión	2W @ 850 / 900 MHz
Programación	Mediante Comandos AT
Consumo de corriente	15 mA en modo sleep
Fuente de alimentación	3,4 a 4,5V
Temperatura de operación	-40°C a 85°C
Soporta TCP/IP	Si
GPRS	Se controla por UART con comandos AT
Compatibilidad	Arduino, Raspberry Pi, PC

Peso	11,1g
Dimensiones	77 x 55 x 17mm (con pines de montaje)

Fuente: (Montesdeoca, 2012)

Realizado por: Caiza, Angel. 2020.

3.4.2. *Arduino Mega*

Se optó por utilizar este modelo porque es muy completo y tiene varios puertos digitales, que resulta fácil al conectar los dispositivos de entrada y salida. El precio es moderado y compatible con el módulo SIM900, en la placa de desarrollo se realiza toda la programación, se recolecta información de los sensores y también del módulo GSM. cómo se observa en la figura 7-3

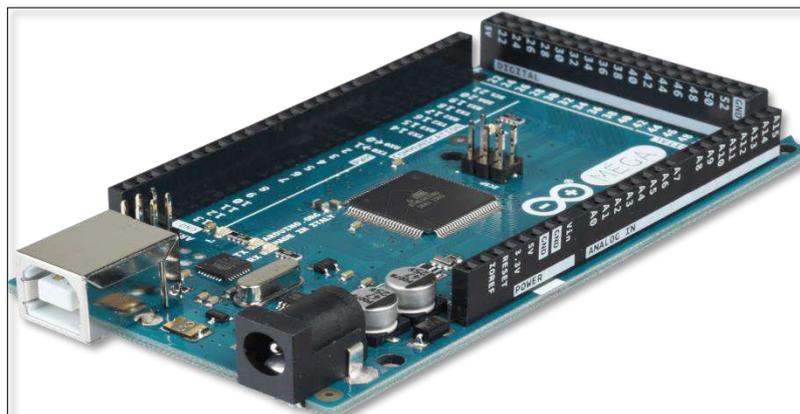


Figura 7-3: Presentación física tarjeta de desarrollo Arduino MEGA

Fuente: <https://www.circuitstoday.com/wp-content/uploads/2018/02/Arduino-Mega-2560-e1517542378961.png>

Las características técnicas se muestran en la Tabla 5-3

Tabla 5-3: Tabla de características técnicas Arduino Mega

Arduino Mega	Características
Memoria flash	128 KB
Pines de E/S analógicas	Dieciséis
Velocidad de reloj	16 MHz (configurado de fábrica a 1Mhz)
Pines de E/S digitales	54
Fuente de alimentación	7-12 V (regulador incorporado para el controlador)
E / S digital total	70 (digital + analógico)
SRAM	8 KB

Microcontrolador	AVR ATmega 2560 (8 bits)
Comunicación	USB (Programación con ATmega 8), ICSP (programación), SPI, I2C y USART
Minutero	2 (8 bits) + 4 (16 bits) = 6 temporizadores
PWM	12 (2-16 bit)
ADC	16 (10 bits)
USART	44
Interrupción de cambio de pin	24

Fuente: (Montesdeoca, 2012)

Realizado por: Caiza, Angel. 2020

3.4.3. Sensores

Se detalla las características más relevantes de los sensores utilizados para el bloque de monitoreo.

3.4.3.1. Ultrasónico HC-SR04(Proximidad)

Su función es el envío de pulsos ultrasónicos en intervalos de tiempo y la percepción del eco y a través de esto se puede conocer la distancia en que se encuentra frente a la presencia de un obstáculo, se observa el en la siguiente figura 8-3



Figura 8-3: Sensor ultrasónico HC-SR04

Realizado por: Caiza, Angel. 2020

Las características técnicas se muestran en la Tabla 6-3.

Tabla 6-3: Tabla de características técnicas de sensor ultrasónico HC-SR04

HC-SR04	Características
Alimentación	5 V
Rango de medición	2 cm a 400 cm
Corriente de alimentación	15 mA
Frecuencia	40 Khz
Apertura de pulsos ultrasónico	15 grados
Señal de disparo	10uS

Fuente: (Montesdeoca, 2012)

Realizado por: Caiza, Angel; 2020.

3.4.3.2. Sensor de movimiento PIR detector de presencia HC-SR501

El sensor PIR detecta el movimiento midiendo los cambios en los niveles de infrarrojos (calor) que emiten los objetos que lo rodean. Cuando se detecta el movimiento del sensor PIR, emite una señal de alto en su pin de salida. Rango de detección de 7 metros y ángulo operativo de 120°. Imprescindible en cualquier alarma y proyecto demótico (Purcachi y Nacimba, 2016), como se observa en la figura 9-3



Figura 9-3: Sensor de movimiento PIR detector de presencia HC-SR501

Realizado por: Caiza Angel, 2020

Características técnicas (Purcachi y Nacimba, 2016)

- Rango de detección de 7 metros
- Ángulo operativo 120°
- Sensor infrarrojo con circuito de control
- Sensibilidad y tiempo ajustables de forma sencilla
- Voltaje de funcionamiento: 4.5V- 20V
- Consumo en estado inactivo: <60uA

- Voltaje de salida: 3.3V TTL output
- Intervalo de tiempo: 0,5-200s (ajustable, por defecto 5s \pm 3%)
- Tiempo de bloque: 2.5s (por defecto)
- Temperatura de trabajo: -20°C a 70°C
- Dimensiones: 3,2 x 2,4 x 1,8cm

3.4.3.3. Zumbador

Es un transductor electroacústico que puede producir un sonido continuo o intermitente generalmente agudo. Se utiliza como mecanismo de señalización acústica. La figura 10-3 muestra la forma física de un zumbador piezoeléctrico.



Figura 10-3: Zumbador piezoeléctrico de 5v
Fuente: Caiza, Angel. 2020

Características técnicas (Electrónica SMD, 2017)

- Tipo: Activo
- Diámetro: 12mm
- Alto: 9.5mm
- Rango de Voltaje 2.7- 3.5 VCD
- Voltaje de funcionamiento: 4.5V- 20V
- Corriente: 32mA
- Salida sonido: 85dB
- Frecuencia: 2300Hz

3.4.4. Puente H L298N

El módulo puente H L298N es una tarjeta para el control de motores de corriente directa, motores a pasos, solenoides y en general cualquier otra carga inductiva. La tarjeta esta construida en torno al

circuito integrado L298N, el cual dispone en su interior de 2 puentes H independientes con capacidad de conducir 2 amperios constantes o 4 amperios en picos no repetitivos. La tarjeta expone las conexiones hacia el motor a través de bloques de terminales (clemas), mientras que las entradas de control y habilitación del puente H se exponen a través de headers macho estándar para facilitar todas las conexiones.

Se utilizó un módulo L298N para controlar el sentido de giro del actuador electrónico se muestra en la figura 11-3

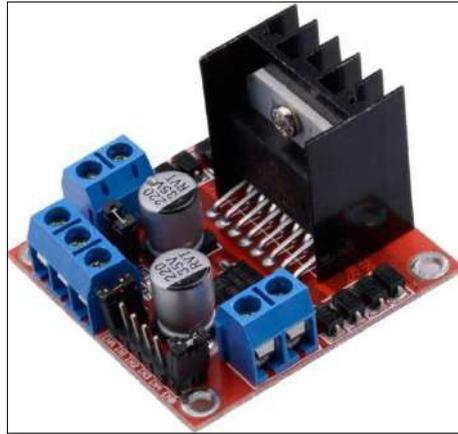


Figura 11-3: Puente H L298N
Realizado por: Caiza Angel, 2020

Características técnicas (Tostatronic, 2019)

- Voltaje de alimentación hasta 46V
- Voltaje de alimentación de la sección lógica max: 7V
- Corriente DC total hasta 4A
- Nivel lógico "0" hasta 1.5V
- Corriente contaste de operación: 2 Amperios

3.4.5. LCD

TFT-LCD (Pantalla de cristal líquido de transistores de película fina) es una variante de pantalla de cristal líquido para mejorar su calidad de imagen. Las LCD de TFT son un tipo de LCD de matriz activa, están disponibles comúnmente en tamaños de 12 a 30 pulgadas. El chip controlador de esta pantalla es extremadamente común y el código necesario se encuentra disponible libremente en internet. Se puede utilizar fácilmente con cualquier microcontrolador que tenga al menos 6 pines disponibles, se observa en la figura 12-3

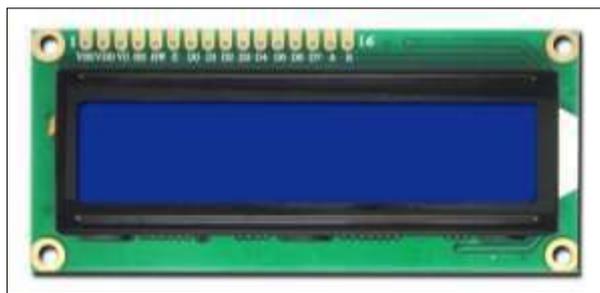


Figura 12-3: Estructura de un LCD

Fuente: (Greek factory, 2019)

En este LCD se aprecia las diferentes opciones que tiene el usuario dependiendo de los requerimientos

Características de pantalla LCD de 16x2 HD44780 (Greek factory, 2019)

- Controlador HD44780
- Color: Fondo azul, letras blancas
- Modo de Operación: 4 y 8 bits
- Voltaje de operación: 4.5 – 5.5V

3.4.6. Fuente de alimentación

Se utilizó un UPS de **110/120VAC** recargable con el propósito de abastecer de energía al prototipo en el caso que el suministro de energía falle o exista corte de energía por mantenimiento. Este dispositivo cuenta con una autonomía de 4 horas y 5 minutos, este tiempo es suficiente para tomar acciones de abastecimiento, en caso de que el corte de energía se prolongue por más tiempo. Como se observa en figura 13-3



Figura 13-3: Batería de litio

Fuente: (Lelong, 2017)

Las características de lo observa en la tabla 7-3

Tabla 7-3: Características de la fuente de alimentación

Detalles	Características
Voltaje entrada	110/120VAC
Frecuencia	60+/-5%
Autonomía	4 horas 5 minutos
Peso	5Kg
Voltaje de salida	120VAC
Regulación de Voltaje	+/-10%
Batería	12V4,5Ah*1

Fuente: (Power, 2019)

Realizado por: Caiza, Angel; 2020.

Para calcular el tiempo de duración de ups se utilizó la siguientes formulas

$$Vb * Ib = Wb$$

$$Vb * Ic = Wc$$

$$\frac{Wb}{Wc} = Horas$$

Vb = Voltaje batería

Ib = Intensidad batería

Wb = Potencia batería

Ic = Intensidad circuito

Wc = Potencia circuito

Ah = Amperios hora

En la tabla 7-3 se observa un extracto de la data del dispositivo lo cual consideramos para realizar los cálculos siguientes:

$$Vb = 12V$$

$$Ib = 4.5Ah$$

$$Ic = 3A$$

$$12 * 4.5 = 54W$$

$$12 * 3 = 36W$$

$$\frac{54}{36} = 1.5 Horas$$

Con la ayuda de las fórmulas se calculó la capacidad del UPS de suministrar energía a nuestro circuito, que es de 1.5 horas, según el registro EERSA, de los últimos cuatro años no existe cortes de energía prolongados en la ciudad de Riobamba. (EERSA, 2019)

3.5. Diagramas de conexión

3.5.1. Arduino mega y módulo SIM900

Las conexiones entre los módulos SIM900 y la placa de desarrollo Arduino mega, los pines TXD y RXD, de la tarjeta SIM900 son los encargados de enviar y recibir toda la información que en desde el arduino y enviarlos a la nube thingspeak.

La forma de conexión entre los dispositivos se aprecia en la Tabla 8-3, donde se detalla cada uno de los pines

Tabla 8-3: Conexión de pines entre Arduino Mega y el módulo SIM900

Arduino Mega	SIM900
Tx1(Ping18)	RXD
Rx1(Ping19)	TXD

Realizado por: Caiza Angel, 2020

La alimentación de estos módulos de lo realiza mediante una fuente propia que viene en el kit al momento de la adquisición del dispositivo las cuales suministra **12V** a **4.5A** respectivamente

3.5.2. Arduino mega y el sensor Ultrasónico HC-SR04

El sensor ultrasónico está conectado con una fuente de alimentación externa, el trigger y el echo son líneas de control mediante estas líneas, pasa los datos de las lecturas tomadas de este sensor, y esta conectados al ping 10 y 11 del arduino respectivamente

La forma de conexión entre los dispositivos se muestra en la Tabla 9-3, donde se detalla cada uno de los pines

Tabla 9-3: Conexión de pines entre Arduino Mega y el sensor Ultrasónico HC-SR04

Arduino Mega	Sensor Ultrasónico HC-SR04
PB4(Ping11)	Trigger
PB5(Ping10)	Echo
VCC	+5v
GND	Gnd -

Realizado por: Caiza Angel, 2020

Para alimentar el dispositivo ultrasónico se lo realiza mediante una fuente externa de **12V** a **4.5A** con la finalidad de proporcionar un funcionamiento adecuado.

3.5.3. *Arduino mega y el PIR detector de presencia HC-SR501*

Consta de tres pines dos de alimentación y un ping de salida, el otro extremo conectado al ping 12 y la alimentación se lo hace con una fuente externa.

La forma de conexión entre los dispositivos se aprecia en la Tabla 10-3, donde se detalla cada uno de los pines.

Tabla 10-3: Conexión de pines entre Arduino Mega y el PIR detector HC-SR501

Arduino Mega	PIR detector de presencia HC-SR501
PB6(Ping12)	OUT
VCC	+5v
GND	Gnd -

Realizado por: Caiza Angel, 2020

3.5.4. *Arduino mega y el actuador electrónico*

Para el correcto funcionamiento es necesario un módulo Puente H L298N, con este dispositivo se puede controlar la salida/entrada del pistillo del actuador, este módulo está conectado al ping 22 y 24 respectivamente, además de esto se debe manifestar que el Puente H L298N, tiene una fuente particular independiente, que suministra energía suficiente, para controlar los estados que envía el bloque de control.

La forma de conexión entre los dispositivos se aprecia en la Tabla 11-3, donde se detalla cada uno de los pines.

Tabla 11-3: Conexión de pines entre Arduino Mega y el Actuador electrónico

Arduino Mega	Actuador Electrónico
PA0(Ping22)	IN1
PA2(Ping24)	IN2
VCC	Vcc (Ping 9)
GND	Gnd (Ping 8)

Realizado por: Caiza Angel, 2020

3.6. **Software utilizado el ciclo parqueadero**

3.6.1. *Thingspeak*

La presente plataforma tiene múltiples prestaciones entre las más destacadas son el almacenamiento, consumo de datos y lo importante el monitoreo de información, los sensores pueden enviar información a esta plataforma y esta tiene la capacidad de procesarla y presentar al usuario para un análisis minucioso de los datos obtenidos.

Esta plataforma recibirá datos de los sensores siguientes: ultrasónico sensor de proximidad, el sensor de movimiento, y del actuador electrónico que es el candado de seguridad del parqueadero.

Esta plataforma admite reconocer hasta 8.219 datos diarios de manera gratuita; si se pretende registrar una mayor cantidad de datos al mes, se puede elegir un plan de servicio pagado que oferta la plataforma según sean las necesidades del usuario (Cantuña y Satian, 2019, p.55)

3.6.2. *IDE Arduino*

Es un lenguaje de programación que permite desarrollar múltiples aplicaciones e interactuar con múltiples sensores y plataformas de entorno gráfico

3.6.3. *Wireshark*

Es un software que sirve para medir el tráfico que produce un usuario o aplicación

3.7. Diagrama de flujo

Es la representación gráfica o secuencia de rutinas simples, tiene la ventaja de indicar la secuencia del proceso a realizar en un determinado trabajo.

3.7.1. *Aparcar*

En el presente diagrama de bloques están los procesos para realizar el aparcamiento de la bicicleta, en la fase inicial lee todas las variables y los inicializa, siguiendo con el algoritmo preguntamos, cuál de los slot elegimos por la opción se identifica al usuario, genera el código de seguridad, estos datos están preparados para subir a la plataforma, y son encaminados mediante el SIM900, esta tarjeta tiene conexión con el internet, luego de haber generado el código aleatorio el dispositivo queda esperando el SMS, por la opción SI, abre el actuador y queda un lapso de 20 segundos, hasta que el usuario coloque la bicicleta en el parqueadero, habiendo transcurrido este tiempo, el actuador

se cierra, acto seguido termina el proceso, si el SMS no coincide con el código generado, luego el LCD del dispositivo visualiza un SMS con el mensaje no coincide, entonces realiza un envío de alerta, continua con una llamada al móvil del propietario, ya que lo considera un intento de robo, también si se genera el código en dispositivo y no recibe ningún código en el bloque de control, este considera como disponible. En el caso de elegir el slot2 será el mismo procedimiento, este proceso se realizará para N slots. Como se ve en la figura de 14-3

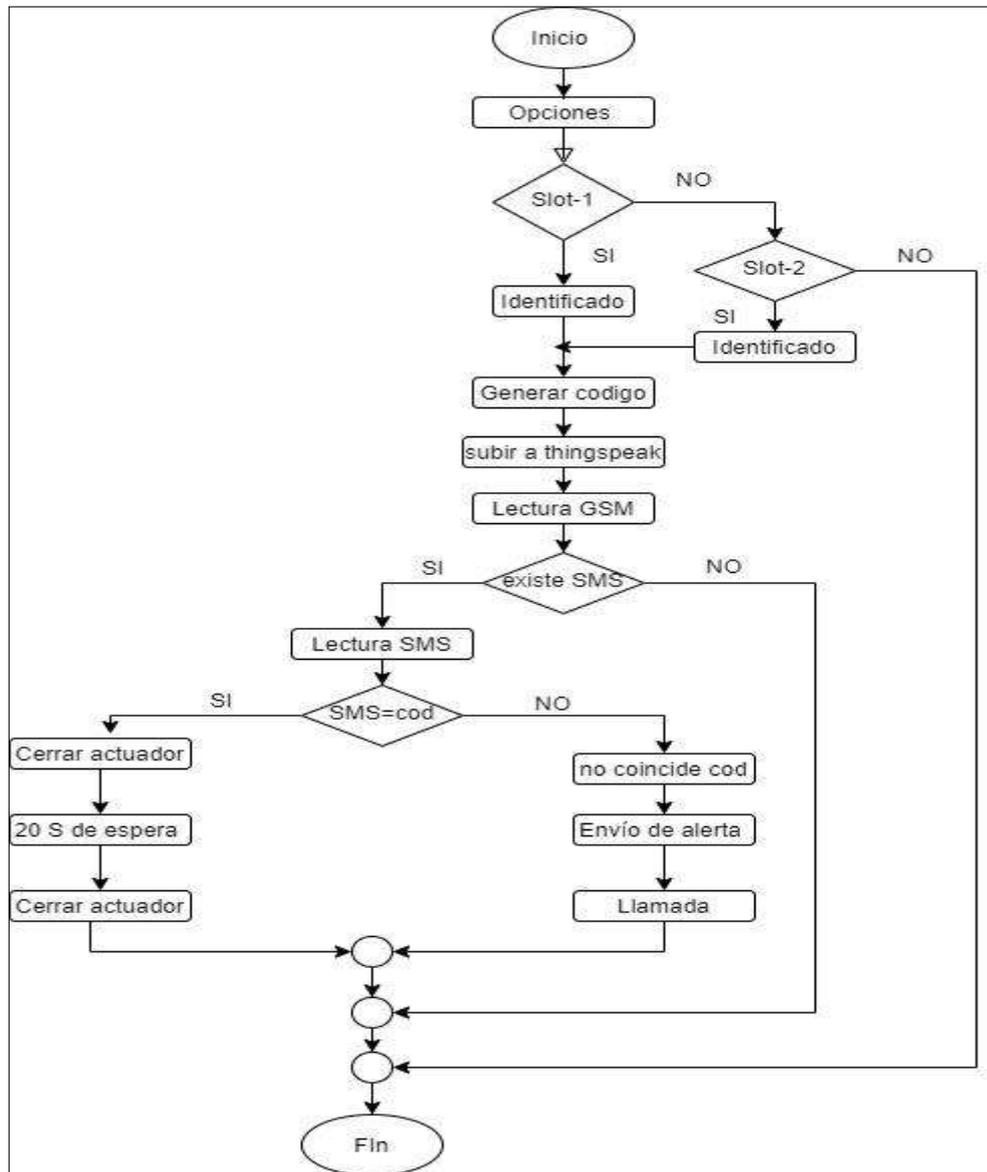


Figura 14-3: Diagrama de flujo de aparcamiento
 Realizado por: Caiza Angel, 2020

3.7.2. Desaparcar

Para realizar la opción desahcar, el usuario realiza un similar procedimiento, con la única diferencia que el código que llega al dispositivo debe ser comparado con el número de teléfono que realizo el aparcamiento, si esto coincide abre el actuador y puede retirar la bicicleta del ciclo parqueadero, caso contrario estas empezaran a enviar avisos de alerta al usuario registrado con ese slot. Como se ve en la figura de 15-3

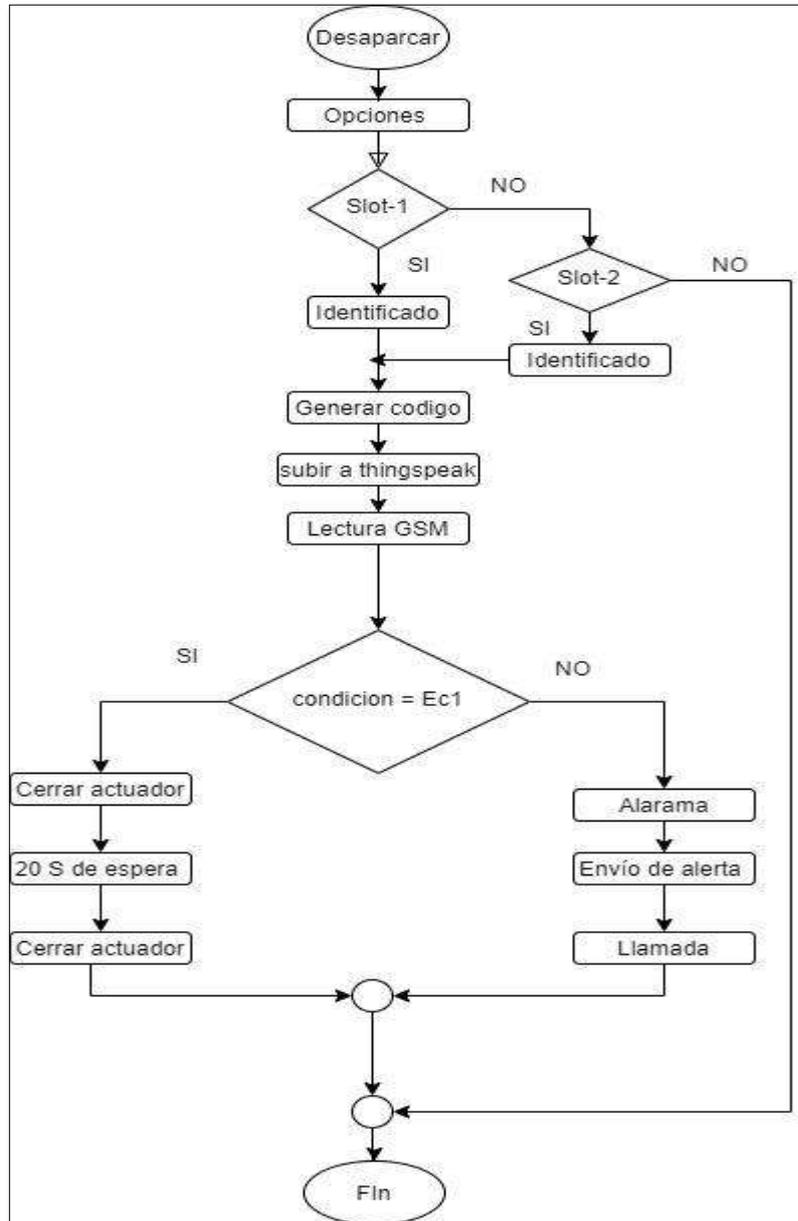


Figura 15-3: Diagrama de flujo del proceso de desahcar
 Realizado por: Caiza Angel, 2020

3.7.3. Disponibilidad

En este diagrama el usuario puede comprobar la disponibilidad de lo slot del ciclo parqueadero, el sensor de proximidad lee los datos, si encuentra un objeto a menos de 5cm, este tomara como, No

Disponible, y lo considera a la variable, $S1 = 1$. Si encuentra a más de 6cm lo considera como Disponible, este procedimiento lo realizara de la misma forma para los demás slots siempre y cuando el usuario necesite hacer el uso de estas opciones. Como se ve en la figura de 16-3

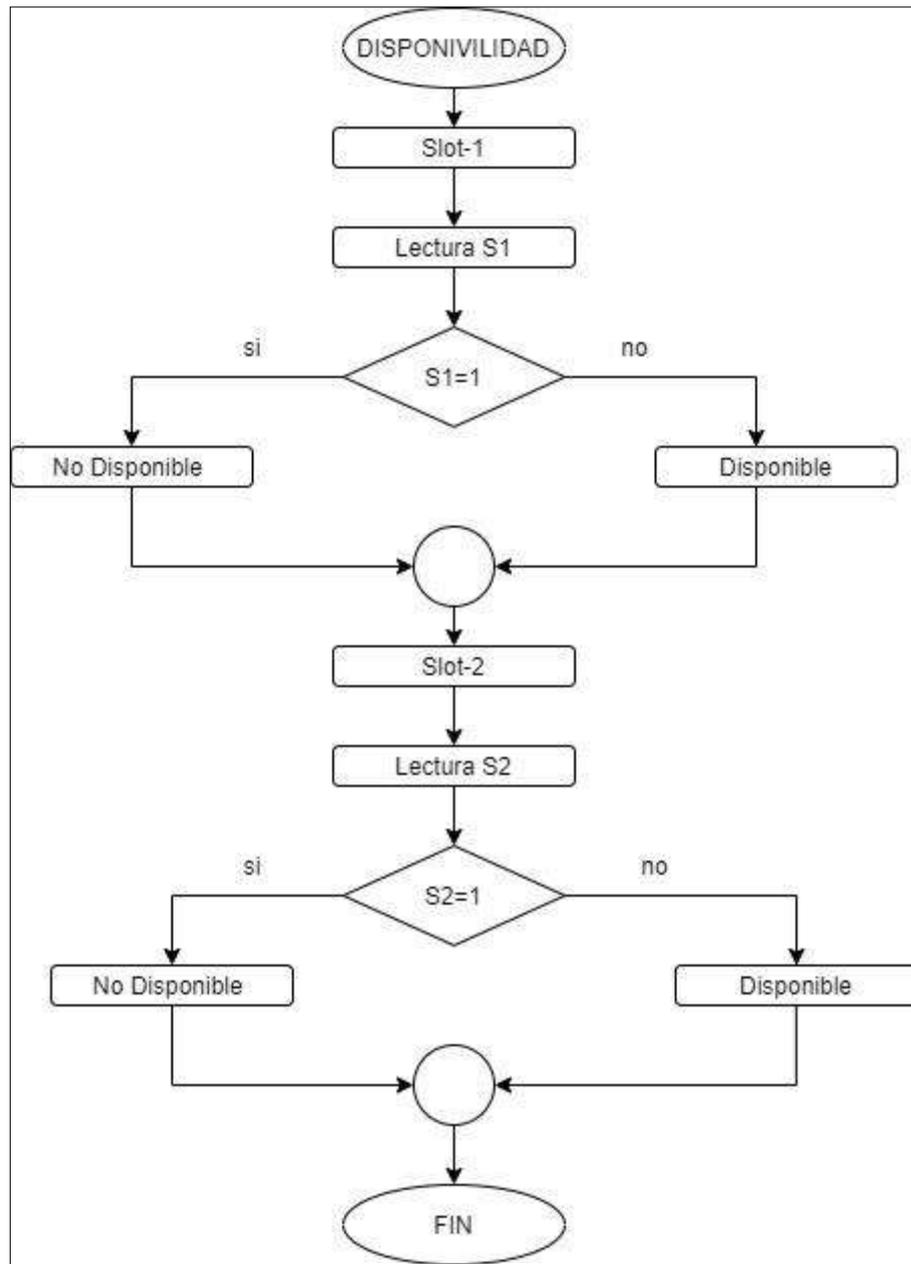


Figura 16-3: Diagrama de flujo de disponibilidad
Realizado por: Caiza Angel, 2020

3.7.4. *Aparcar sin móvil*

El cuarto caso se da, cuando el usuario no dispone de su móvil (terminó su batería, extravió su móvil o se encuentra defectuoso), se podrá utilizar de forma manual, el dispositivo generara el código, esta se podrá visualizar en la cuenta personal de thingspeak, en un dispositivo (computador, table, PDA,

smartphone) que tenga acceso a internet, el usuario entra a la plataforma e ingresa a su cuenta, al ingresar el usuario podrá visualizar la clave después de haber generado el código de seguridad en su cuenta el usuario tendrá un tiempo de 120 segundos, para ingresar en el prototipo, en caso de haber insertado correctamente se procede a abrir el actuador para des aparcar la bicicleta. Como se ve en la figura de 17-3

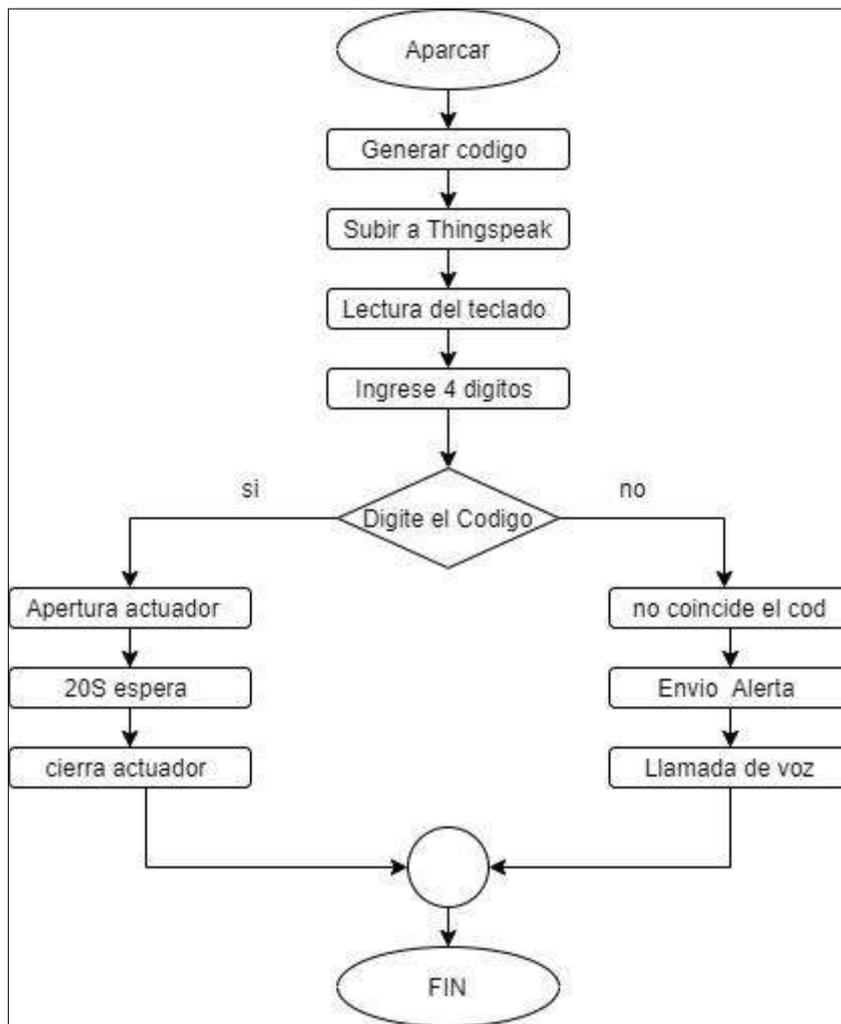


Figura 17-3: Diagrama de flujo de aparcamiento sin móvil
Realizado por: Caiza Angel, 2020

3.7.5. Desaparcamiento sin móvil

Para desaparcamiento la bicicleta de forma manual, el procedimiento es igual al anterior precisando, que las condiciones, establecidas deben ser igual a la condición, de ahí se realiza la apertura del actuador en caso de ir por la opción negativa se activará la alarma sonora, envío de alerta al número de la compañía privada de guardias de seguridad, en caso de hacer caso omiso a esta alerta, esta llamara al ecu 911 como medida de seguridad. Como se ve en la figura de 18-3

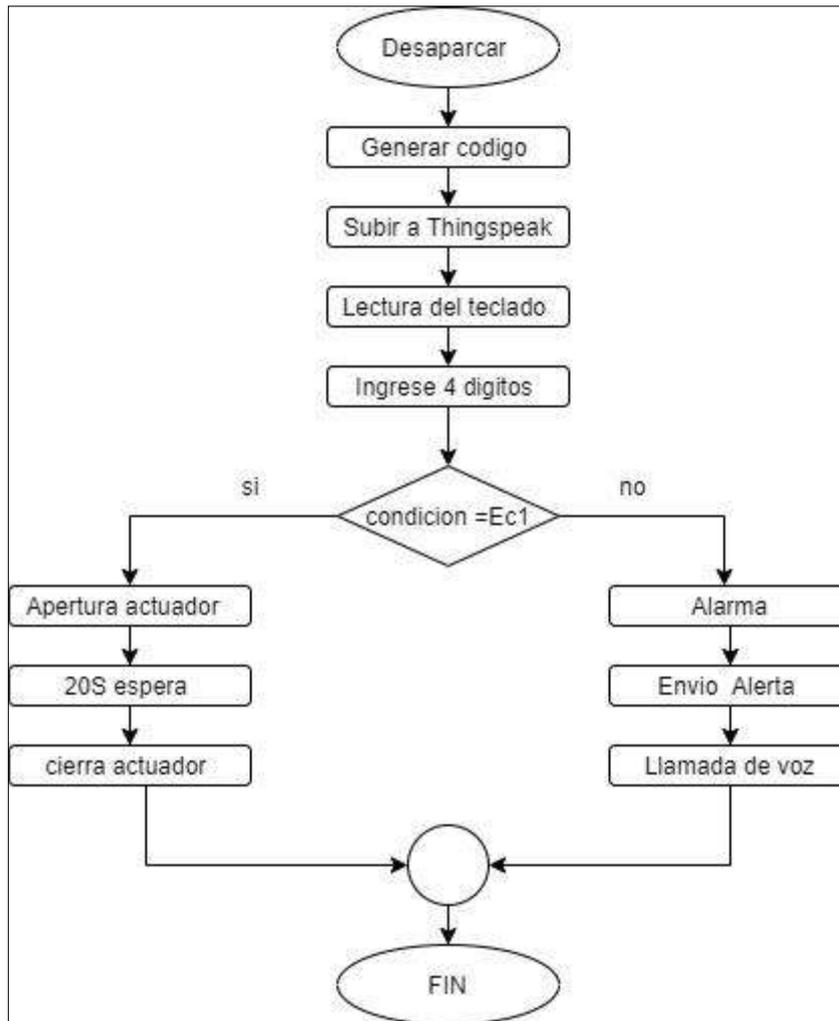


Figura 18-3: Diagrama de flujo del proceso desaparcar sin móvil
 Realizado por: Caiza Angel, 2020

3.8. Diagrama de conexiones de elementos

Se observa cada una de las conexiones cables que se conectan, desde el arduino hacia los módulos que integran el alimentador autónomo, y su conexión de VCC y GND, como se aprecia en la figura 19-3

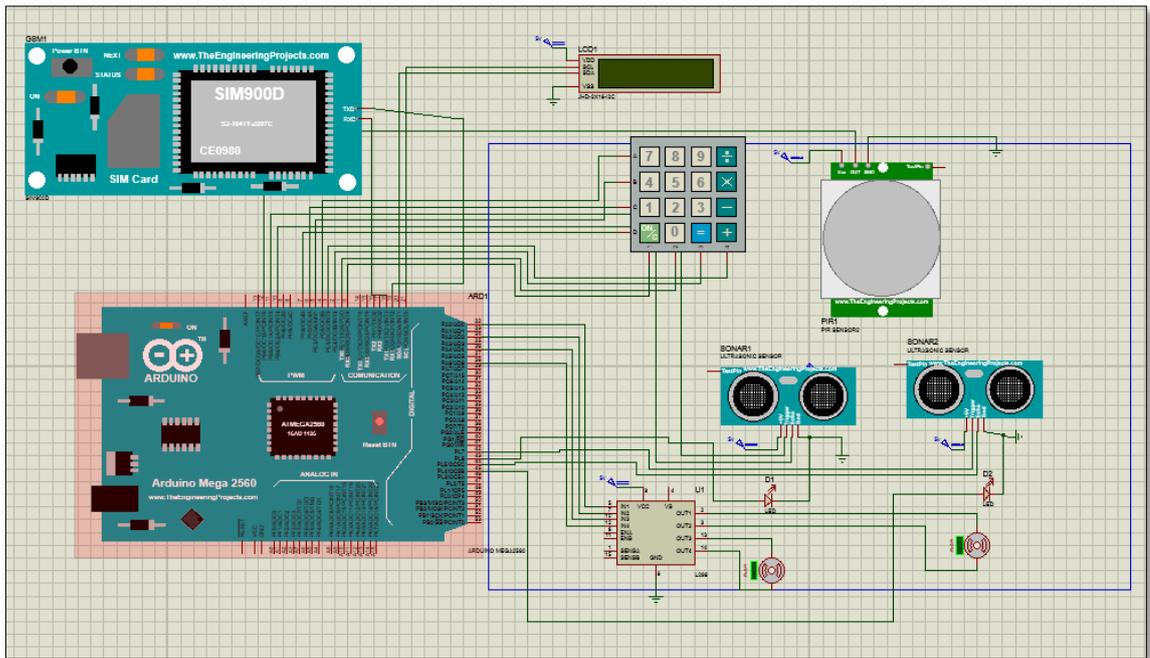


Figura 19-3: Diagrama de conexión del dispositivo
 Realizado por: Caiza Angel, 2020

4. GESTION DEL PROYECTO

En el presente capítulo se muestra el análisis de los resultados obtenidos al realizar las pruebas de prototipo, los cuales fueron obtenidos a partir del desarrollo de tres pruebas en las instalaciones de la FIE. La primera de ellas correspondiente a la funcionalidad, la segunda correspondiente a un análisis de costos y tercera de referente a la aceptabilidad del prototipo por parte de los usuarios.

4.1. Desarrollo de Pruebas

La prueba inicia desde el arranque del prototipo, indicando cada actividad que realiza con el manejo.

4.1.1. Análisis de funcionalidad

El dispositivo entra en funcionamiento y visualización el mensaje ***“Iniciando...”*** en la pantalla LCD, durante este proceso los sensores y puertos inicialicen para la ejecución del sistema como se muestra en la Figura 1-4.



Figura 1-4: Pantalla de inicio del dispositivo
Realizado por: Caiza Angel, 2020

Segundos después de este proceso se imprime el inicio de la ejecución del interfaz con el mensaje ***“Ver niveles”*** que se muestra en la Figura 2-4, iniciando el censado de los dispositivos y enviando hacia la plataforma.



Figura 2-4: Iniciando sensores y GSM para mediciones.

Realizado por: Caiza Angel, 2020

4.1.2. *Menú principal*

Mediante la interfaz de la configuración por teclado, se permite la selección de diferentes funciones: 1=Aparcar, 2=Desaparcar, 3=Disponibilidad y 4=Configuración sin celular; mediante el ingreso de la numeración correspondiente como se muestra en la Figura 3-4.



Figura 3-4: Menú Principal.

Realizado por: Caiza Angel, 2020

4.1.2.1. *Aparcar*

Esta opción permite seleccionar el número del slot que constituye el ciclo-parqueadero, permitiendo elegir mediante el ingreso por teclado, en la Figura 4-4 se visualiza los distintos slots con su correspondiente numeración en la LCD



Figura 4-4: Slot's para Aparcar.

Realizado por: Caiza Angel, 2020

Después de la selección del slot permite recorrer el siguiente proceso para el control del aparcamiento de la bicicleta. Visualización del inicio: se genera una clave randomico, que posteriormente se envía a la plataforma, permitiendo que el usuario envíe en un mensaje la clave, mientras tanto se visualiza "20s para recibir SMS" y se muestra en la Figura 5-4



Figura 5-4: Generación de clave para marcar el inicio de recepción en 20s.

Realizado por: Caiza Angel, 2020

Verifica el código en Thingspeak, el software constata la llegada del mensaje Figura 6-4.



Figura 6-4: Envío del código a Thinkspeak.

Realizado por: Caiza Angel, 2020

Se inicia la lectura del módulo GSM SIM900, se obtiene el número en conjunto con el contenido del mensaje por separado, se verifica que provenga de Ecuador con su código (+593), mediante la visualización “Leyendo SMS” de la Figura 7-4.

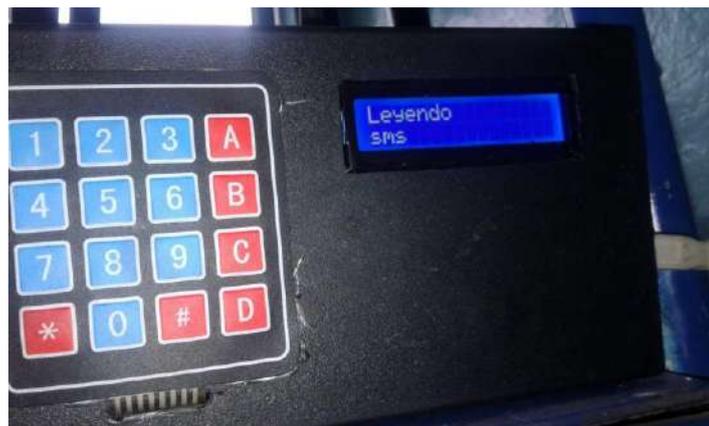


Figura 7-4: Lectura del mensaje.

Realizado por: Caiza Angel, 2020

Mediante la condición por el cumplimiento del sí, se inicia a ingresar el pistón habilitando el slot seleccionado para la llanta de la bicicleta, se continua con una pausa de 5s mostrando el mensaje de la Figura 7-4, al finalizar se extiende el pistón de la cerradura una vez ingresada. En caso de no cumplirse se regresa al menú principal indicando que no coincide el código Figura 9 -4.



Figura 8-4: Caso de coincidencia de código.
Realizado por: Caiza Angel, 2020



Figura 9-4: Caso de no coincidencia de código.
Realizado por: Caiza Angel, 2020

4.1.2.2. *Desaparcar*

Esta opción permite seleccionar un slot que contiene la respectiva bicicleta, la ejecución se realiza cuando se detecta la bicicleta aparcada. Figura 10-4.



Figura 10-4: Slot's para desaparcar.
Realizado por: Caiza Angel, 2020

Mediante la selección del slot se realiza el siguiente proceso: Visualización del inicio, se genera una clave randomico, que posteriormente se envía a la plataforma, permitiendo que el usuario envíe en un mensaje la clave, mientras tanto se visualiza “20s para recibir sms ” se muestra en la Figura 11-4



Figura 11-4: Generación de clave para marcar el inicio de recepción en 20s.
Realizado por: Caiza Angel, 2020

Se visualizó el texto de verificación del código en Thingspeak para el envío del mensaje dentro de 10s. Figura 12-4.



Figura 12-4: Envío del código a Thingspeak.
Realizado por: Caiza Angel, 2020

Se inicia la lectura del módulo GSM SIM900 y la obtención del número en conjunto con el contenido del mensaje por separado, se verifica que provenga de Ecuador con su código (+593), mediante la visualización “Leyendo mensaje” de la Figura 13-4.



Figura 13-4: Lectura del mensaje.

Realizado por: Caiza Angel, 2020

Mediante la condición por el cumplimiento del sí, se inicia a ingresar el pistón habilitando el slot seleccionado para la llanta de la bicicleta, se continua con una pausa de 5s mostrando el mensaje de la Figura 14-4, al finalizar se extiende el pistón de la cerradura una vez ingresada. En caso de no cumplirse se regresa al menú principal indicando que no coincide el código Figura 15-4.



Figura 14-4: Caso de coincidencia de código.

Realizado por: Caiza Angel, 2020



Figura 15-4: mensaje de que no coincide el código.

Realizado por: Caiza Angel, 2020

4.1.2.3. Disponibilidad

Mediante la lectura de los sensores de proximidad se realiza la visualización de la disponibilidad de los slots como se muestra en la Figura 16-4 y Figura 17-4.



Figura 16-4: Disponibilidad del Slot 1.
Realizado por: Caiza Angel, 2020



Figura 17-4: Disponibilidad del slot 2.
Realizado por: Caiza Angel, 2020

4.1.2.4. Control de dispositivo sin celular

El menú consta de dos opciones 1=Aparcar y 2=Desaparcar como se muestra en la Figura 18-4.



Figura 18-4: Opciones de manejo sin celular.
Realizado por: Caiza Angel, 2020

Se realiza el mismo proceso aparcar y desapparcar con el cambio de la recepción del mensaje por el ingreso por teclado, visualizando los números marcados, controlados con un tiempo para el regreso al menú principal, como se muestra en la Figura 19-4.



Figura 19-4: Ingreso del código por teclado.
Realizado por: Caiza Angel, 2020

4.1.3. *Thingspeak*

Mediante el envío de datos del módulo GSM a la plataforma, se registra en la base de datos y visualiza en cada campo específico de las mediciones como se muestra en la Figura 20-4.

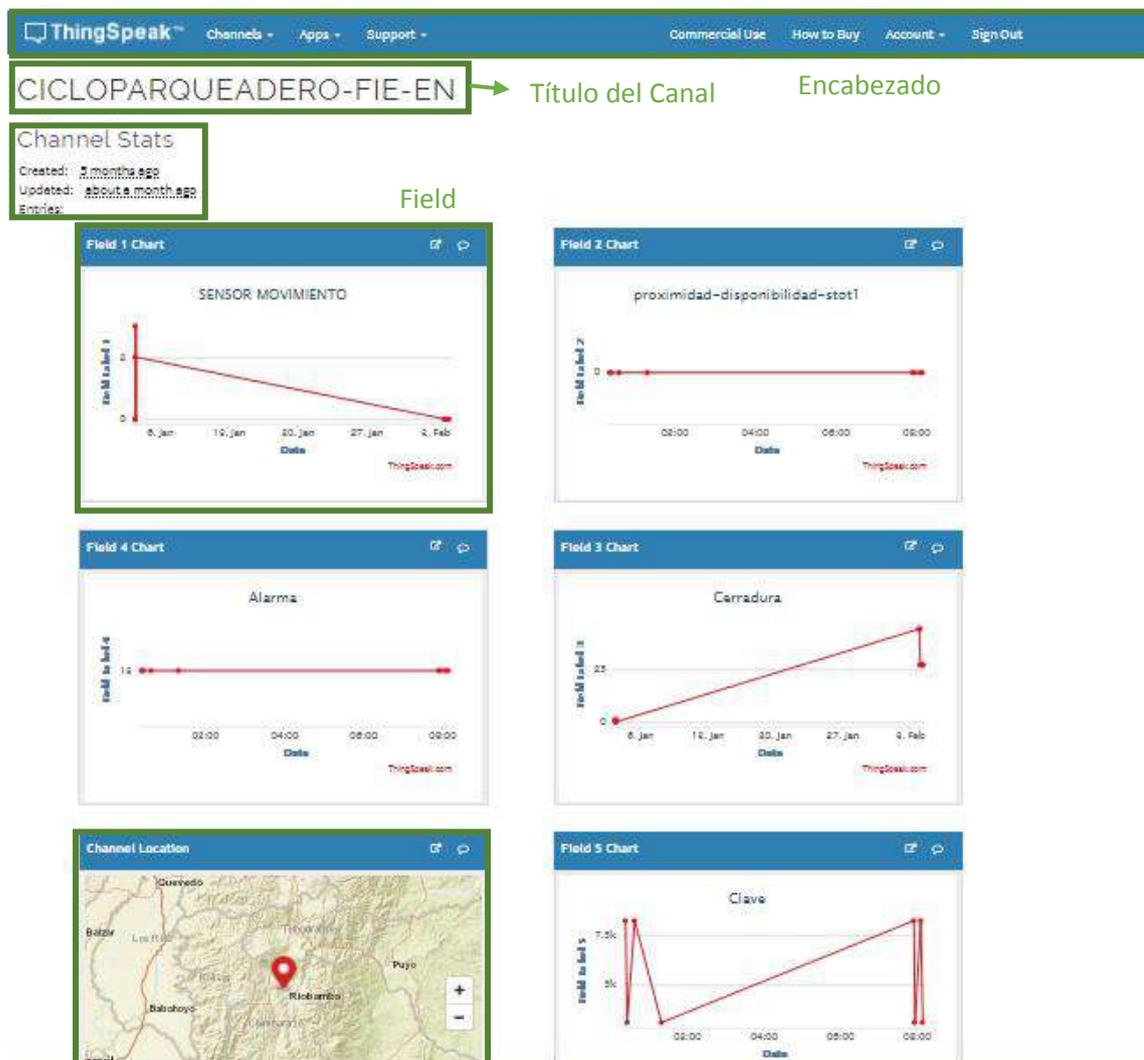


Figura 20-4: Plataforma online Thingspeak.
Realizado por: Caiza Angel, 2020

La interfaz de la plataforma de la Figura 20-4 está constituida por los siguientes elementos:

Field (o campo): herramienta grafica que permite la visualización de los datos de manera gráfica, con el título de la medida, elemento o magnitud correspondiente.

Channel Location: al recibir datos geográficos permite la ubicar la fuente de información.

Título del Canal: denominación para diferenciarse de los demás que tiene a disposición.

Channels: permite encontrar los canales creados o compartidos con el usuario

Chanel stas: indica la fecha de creación y última modificación que se ha realizado.

Apps: permite administrar la visualización de Apps creadas con 53

upport: Da información en documentos, ejemplos, tutoriales y soporte de un asistente.

Comercial Use: suministra información del uso de thingspeak.

How to Build: permite comprar mejores tipos de servicio por una paga.

Account: Administra y configura la cuenta del usuario.

Sigs Out/In: permite ingresar o cerrar la cuenta.

4.1.4. *Desarrollo de la aplicación móvil*

Virtuino es la aplicación que permite la conexión con la plataforma de forma rápida con el uso de identificador (ID) y Api key's,

En la pantalla de inicio se visualiza varios controles que permite realizar las siguientes acciones: la activación de la alarma, generar el código, realizar el aparcamiento, des aparcamiento y reservar un slot, para que este no sea ocupado hasta que el usuario quien reservó llegue y lo utilice, se muestra en la Figura 21-4.



Figura 21-4: Pantalla principal.

Realizado por: Caiza Angel, 2020

La pantalla de Disponibilidad (Figura 22-4) permite al usuario ver que slot esta libre mediante símbolos numéricos: 1: No Disponible, 0: Disponible.

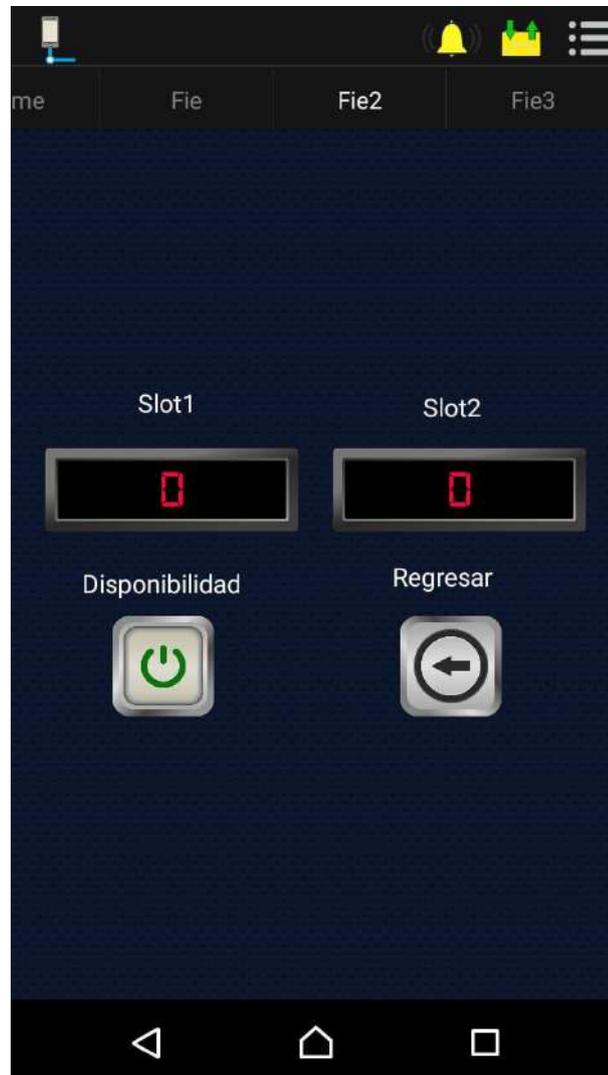


Figura 22-4: Pantalla Disponibilidad.
Realizado por: Caiza Angel, 2020

Al oprimir en el botón de disponibilidad esta buscara en la base de datos los registros y mostrara al usuario esa información acerca del estado en la que se encuentra cada uno de los slots.

La siguiente pantalla es exclusiva para el administrador, permitiendo obtener la información de manera gráfica como se muestra en la Figura 23-4.



Figura 23-4: Pantalla Grafica de los eventos.
Realizado por: Caiza Angel, 2020

4.2. Prueba de capacidad de acceso de red

Se realizó esta prueba para verificar el tiempo que tarda el sistema en enviar la información desde el prototipo hacia thingspeak. mediante el uso del comando PING en el PC conectado a datos móviles CNT. Se obtiene como resultados los datos presentados en la Tabla 1-4. Se tomaron los promedios de respuesta con paquetes ICMP de 32 bytes. Se realizo estas pruebas en 3 diferentes horarios.

Tabla 1-4: Tiempo de respuesta del servidor thingspeak

No	09:00 am	12:00 am	16:00 pm
1	39 ms	34 ms	40 ms
2	37 ms	38 ms	66 ms

3	41 ms	48 ms	69 ms
4	53 ms	60 ms	38 ms
5	45 ms	39 ms	23 ms
6	54 ms	45 ms	63 ms
7	65 ms	47 ms	43 ms
8	32 ms	80 ms	95 ms
9	25 ms	76 ms	51 ms
10	28 ms	34 ms	31 ms
Promedio	41,90 ms	50,10 ms	51,9 ms

Realizado por: Caiza Angel, 2020

La tasa de transferencia de GPRS va desde 52 Kbps a 115Kbps por lo que, para el análisis teórico, se toma las dos tasas de transferencia y aplicando la fórmula de tiempo de transmisión de trama se obtiene los siguientes resultados:

$$\tau = \frac{Lt}{C}$$

Donde:

τ = Tiempo de transmisión de la trama hacia el servidor.

Lt = Longitud de la trama.

C = Tasa de transmisión del canal móvil.

Para tasas de transmisión de trama de 56Kbps se obtiene un tiempo de respuesta de 3,32ms con un encabezado IP de 20 bytes y 26 bits de información.

Tiempo de Transmisión de trama a 115Kbps = 1,62ms

Tomando en cuenta que una trama completa es de 1500 Bytes y el tiempo de respuesta máximo de 250 ms, se determina que la comunicación entre el módulo SIM900 y el servidor es de 41,90 ms para el mejor caso en la mañana y 51,9 ms para el peor caso en la tarde. Se concluye que los tiempos calculados son similares a los obtenidos en forma práctica y mostrados en la Tabla 1-4.

4.3. Análisis de tráfico de la conectividad

Por medio de obtención de la tasa de entrega se obtendrá el porcentaje de recepción de paquetes, con relación con los paquetes enviados por el prototipo hacia thingspeak, y de thingspeak hacia la aplicación.

$$Tasa\ de\ Entrega = \frac{paquetes\ recibidos}{paquete\ enviado} * 100\%$$

Tasa de entrega entre el prototipo de ciclo parqueadero y el thingspeak:

Para definir la tasa de entrega de paquetes, se realiza la relación existente entre paquetes enviados y paquetes recibidos obteniendo el porcentaje de fiabilidad en la entrega de paquetes que tiene.

$$Tasa\ de\ Entrega = \frac{9\ paquetes\ recibidos}{28\ paquete\ enviado} * 100\% = 32.143\%$$

Se obtuvo que el 32.143% de los paquetes enviados por el prototipo de ciclo parqueadero, son acogidos por thingspeak considerando que cada 15s se realiza la apertura de recepción, el cual los paquetes rechazados no avanzan a ser registrados.

Tasa de entrega entre thingspeak y Virtuino:

$$Tasa\ de\ Entrega = \frac{9\ paquetes\ recibidos}{9\ paquete\ enviado} * 100\% = 100\%$$

Se obtuvo que el 100% de los paquetes enviados desde la plataforma, son completamente receptados por la aplicación

4.4. Precisión y Rango de alcance de los sensores.

Prueba de medición del sensor ultrasónico HC-SR04. Para verificar la funcionabilidad de sensor ultrasónico se realizó la siguiente prueba, el ultrasónico fue colocado a distancias variable de 10 a 400 cm de distancia. La herramienta patrón que se utiliza para verificar cada dato obtenido es una cinta métrica de 500cm. El error absoluto que presenta es de ± 1 cm has los 50cm a partir de ello ± 2 cm como se aprecia en la tabla 2-4 Los resultados experimentales obtenidos verifican que el dispositivo no incorpora error al sensor.

Tabla 2-4: Mediciones funcionalidad cinta métrica vs sensor ultrasónico

Nº	Cinta métrica advanced 500 (cm)	Sensor Ultrasónico HC- SR04 (cm)	Error Absoluto (cm)

1	10	10	0
2	20	20	0
3	30	29	-1
4	40	41	+1
5	50	51	+1
6	100	102	+2
7	150	102	+2
8	200	202	+2
9	250	252	+2
10	400	398	-2

Realizado por: Caiza Angel, 2020

Se verificó las medidas que genera el sensor a través del monitor puerto serial de la plataforma arduino presenta errores a partir de los 30 cm como se puede apreciar en la figura 24-4

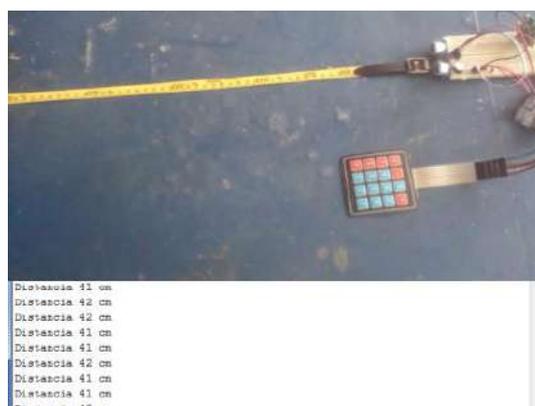


Figura 24-4: Prueba de la medición del sensor.

Realizado por: Caiza Angel, 2020

4.5. Costo del proyecto

Los componentes que se utilizaron para el desarrollo del prototipo Tabla 3-4, se observa la cantidad, costo por unidad y el costo total.

Tabla 3-4: Obtención del costo de los dispositivos electrónicos

N	Descripción	Cantidad	V. Unitario	V. total
1	Arduino Mega 2550	1	15,00	15,00
2	SHIELD SIM 900 GSM/GPRS	1	33,00	33,00

	Sensores			
3	Ultrasónico HC-SR04(Proximidad)	2	2,50	5,00
4	Sensor de movimiento PIR detector de presencia HC-SR501	1	2,50	2,50
5	Zumbador	1	1,50	1,50
6	PUENTE H L298N	1	3,50	3,50
7	LCD de 16 x2 con módulo I2c	1	4,00	4,00
9	FUENTE DE ALIMENTACION	1	4,00	4,00
10	Clave de pines macho- hembra	10	0.1	1,00
11	Tabla perforada	1	1,00	1,00
12	Teclado matricial	1	2,00	2,00
13	UPS	1	36,00	36,00
14	Metro de cable UTP	5	1,00	5,00
	Costo de estructura	1	25.70	25,70
			TOTAL	139,20

Realizado por: Caiza Angel, 2020.

Para justificar el trabajo de ingeniería se detalla a continuación en la tabla 4-4 se ha considerado estos valores con relación al salario básico unificado (SBU)

Tabla 4-4: presupuesto del trabajo de ingeniería

N	Descripción	Cantidad	V. Unitario	V. total
15	Días de realización de pruebas.	15	20	300
1	Costo del diseño del prototipo	1	100	100
1	Programación del APP para android	1	150	150
1	Diseño de la plataforma thingspeak	1	150	150
			Total	700

Realizado por: Caiza Angel, 2020.

El costo total del sistema incluye los dispositivos utilizado y el trabajo de ingeniería se observa en la tabla 5-4.

Tabla 5-4: Costo total del ciclo parqueadero implementado

Costo parcial	139.20
Trabajo de ingeniería	700
Costo Total	839,20

Realizado por: Caiza Angel, 2020.

4.6. Análisis de aceptabilidad

Para el análisis de aceptividad del prototipo por parte de los usuarios, se aplicó una encuesta online a los estudiantes, docentes y trabajadores, mediante el planteamiento de preguntas abiertas, de tipo informativa, para conocer el grado de aceptación del ciclo parqueadero realizado.

Para sacar la muestra se confirmó con las autoridades de la secretaria académica de la ESPOCH el número de alumnos matriculados, los cuales nos manifestaron que existen 1651 miembros. Para el cálculo de la muestra necesaria, se utilizó la siguiente fórmula:

$$n = \frac{N * Z_{\alpha}^2 * P * Q}{e^2 * (N - 1) + Z_{\alpha}^2 * P * Q}$$

Donde los datos utilizados se detallan en la tabla 5-4.

Tabla 6-4: Datos necesarios para determinar la muestra

Símbolo	Significado	Valor
N	Tamaño de la población	1651
Z α	Parámetro estadístico que depende del Nivel de Confianza (NC)	1,96
P	Probabilidad de que si ocurra el evento	50%
Q	Probabilidad de que no ocurra el evento	50%
e	Margen de error	5%

Realizado por: Caiza Angel, 2020.

$$n = \frac{1651 * 1.96^2 * 50 * 50}{5^2 * (1651 - 1) + 1.96^2 * 50 * 50}$$

$$n = \frac{15856202}{50854}$$

$$n = 311 \text{ encuestas}$$

En el anexo se adjunta la información de las preguntas realizadas y se obtiene los siguientes resultados.

4.3.1. Tabulación

A través de la tabulación de las respuestas obtenidas en las encuestas, se realiza un análisis mediante diagramas de pastel considerando el porcentaje que corresponde a cada respuesta.

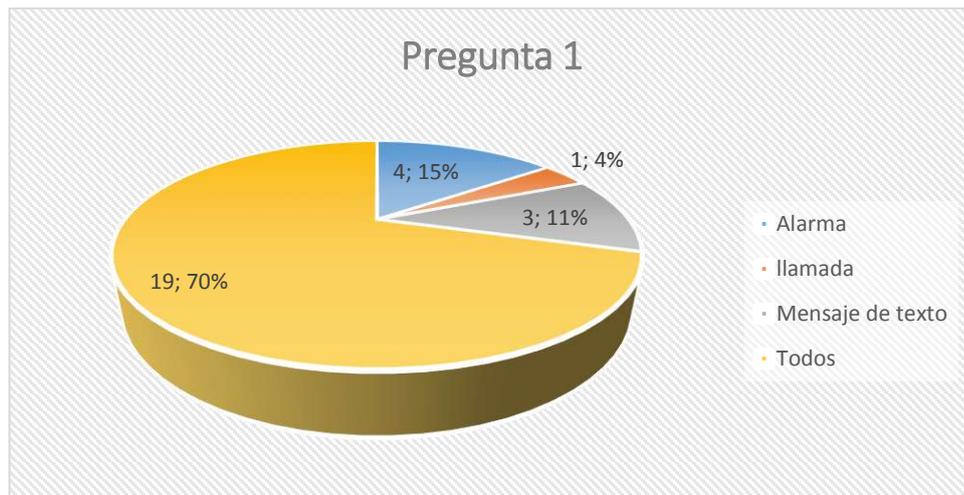


Gráfico 1-4: ¿Qué tipo de notificación en caso de intento de robo le parece segura?
Realizado por: Caiza Angel, 2020

Mediante el análisis del Grafico 1-4 se concluye que la opción de mayor porcentaje tuvo, es del 70%, indica que la gente prefiere que se realice una llamada, mensajería y aviso sonoro cuando exista un intento de vulnerar la seguridad en el ciclo parqueadero.

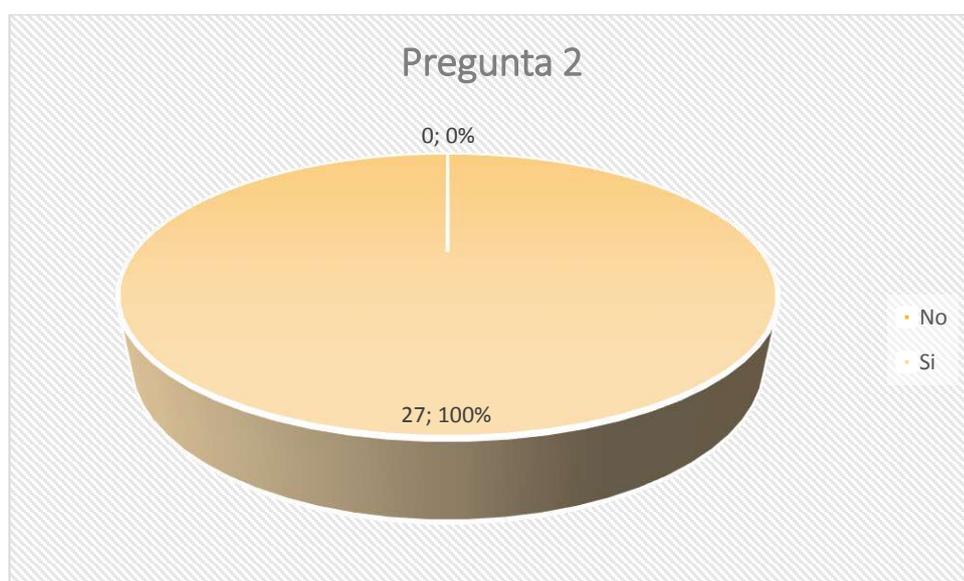


Gráfico 2-4: ¿Le parece muy sencillo el manejo por teclado de la interfaz?

Realizado por: Caiza Angel, 2020

Mediante el análisis del Grafico 2-4 se concluye que el manejo del prototipo fue muy sencillo completamente para la gente.

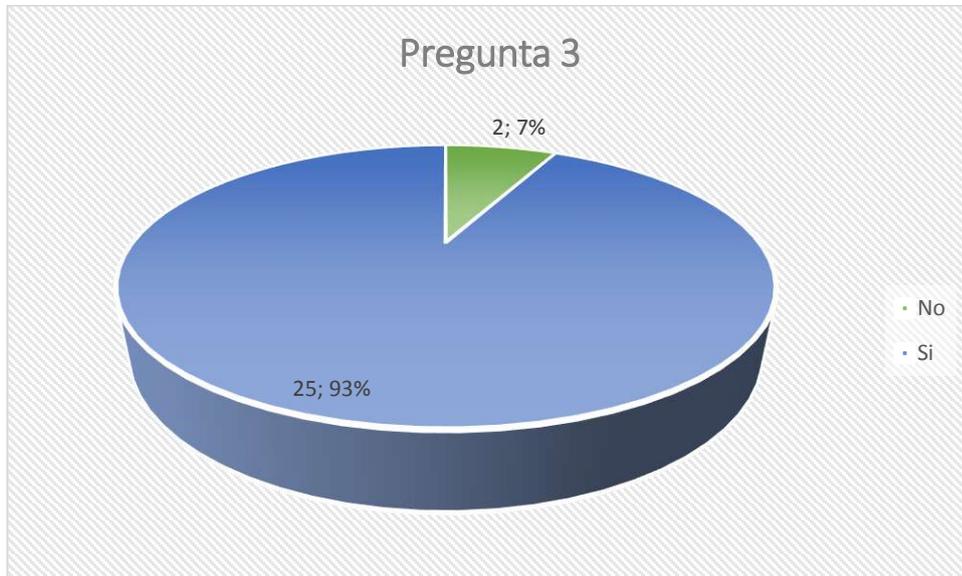


Gráfico 3-4: ¿qué tan seguro le parece el ciclo-parqueadero inteligente?
Realizado por: Caiza Angel, 2020

Mediante el análisis del Grafico 3-4 se concluye que la opción que tuvo mayor porcentaje es de 93%, indica que de la gente encuestada acepta como idóneo el trabajo realizado, Mientras que el 7%, respondieron de forma negativa, esto es debido a que anteriormente los candados eran de un material diferente al metal.

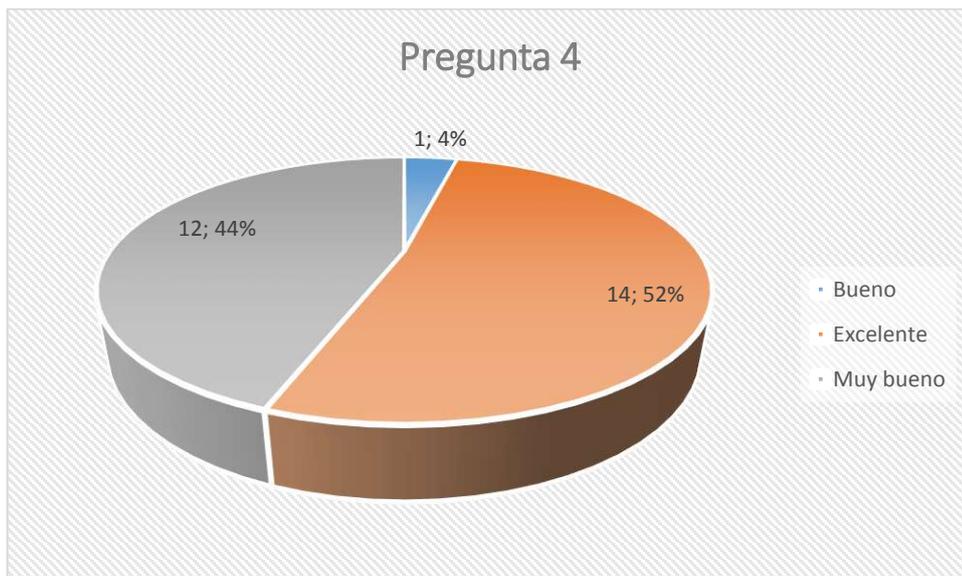


Gráfico 4-4: ¿Cómo estuvo su interacción con el ciclo-parqueadero?
Realizado por: Caiza Angel, 2020

Mediante el análisis del Grafico 4-4 se concluye que, según la valoración cualitativa de las personas encuestada, el 4% le parece buena, 44% muy bueno y el 52% excelente. Estas denominaciones son aceptables debido a que no hubo respuestas negativas en las personas encuestadas.

5. CONCLUSIONES

- Con la investigación se demostró que se pueden diseñar e implementar sistemas de ciclo parqueaderos inteligentes con aplicaciones celulares App.
- A través de la utilización de una red tecnología de comunicaciones, un sistema de notificaciones, un sistema de sensores y un sistema de actuadores, se desarrolló un ciclo parqueadero inteligente.
- Se aprovechó la red datos GSM que se utiliza para la transmisión móvil de voz para el almacenamiento de datos en thingspeak, son herramientas muy útiles en el diseño de prototipo de ciclo parqueadero.
- Thingspeak es una plataforma de Internet of Things (IoT) que permite recoger y almacenar datos de sensores en la nube y desarrollar aplicaciones IoT.
- Virtuino es una app gratuita para Android donde se puede desarrollar proyectos para conectarse via Wi-Fi, Bluetooth, Ethernet o GSM, permite crear una interfaz gráfica personalizada para el acceso, control e interacción con el ciclo parqueadero,
- Al validar el prototipo del ciclo parqueadero se pudo evidenciar que el tiempo promedio del servidor thingspeak es de 47,97ms.

6. RECOMENDACIONES

- Considerar normas de autenticación, para incrementar la seguridad e integridad de datos.
- Para un mejor almacenamiento de datos se debe contratar un hosting, para minimizar el tiempo de respuesta en la comunicación con la base de datos thingspeak.
- Se debería aplicar las normas internacional IEC 60529 para evitar daños.
- Se recomienda diseñar prototipos utilizando un módulo Wi-Fi ESP8266.
- Se recomienda ampliar las funcionalidades del prototipo, incorporando otras opciones como cámara Ip, actuadores electrónicos, paneles solares, entre otras.
- Es recomendable que el mantenimiento preventivo se realice cada 6 meses.

GLOSARIO

- GSM.** Sistema Global para las comunicaciones Móviles
- RTC.** Reloj en tiempo real
- LCD.** Pantallas de cristal líquido
- GND.** Tierra del sistema.
- VCC.** Voltaje colector común
- DC.** Corriente directa.
- Hz.** Hercios
- ION.** Átomo o un grupo de átomos que tiene una carga neta positiva o negativa
- I.** Intensidad de corriente
- A.** Amperio.
- mA.** miliamperio.
- s.** segundo.
- ms.** milisegundo
- W.** Vatio.
- h.** Hora.
- °.** Grado.
- I2C.** Es un bus con múltiples maestros, lo que significa que se pueden conectar varios chips al mismo bus y que todos ellos pueden actuar como maestro, sólo con iniciar la transferencia de datos.
- SPI.** (Serial Peripheral Interface) es un protocolo de datos en serie síncrono utilizado por los microcontroladores para comunicarse con uno o más dispositivos periféricos rápidamente en distancias cortas. También se puede utilizar para la comunicación entre dos microcontroladores.
- MISO.** (Master in Slave Out) - La línea de esclavo para el envío de datos al maestro,
- MOSI.** (Maestro Fuera Esclavo dentro) - La línea principal para el envío de datos a los periféricos,
- SCK.** (Reloj serie) - Los impulsos de reloj que sincronizan la transmisión de datos generado por el maestro y una línea específica para cada dispositivo.
- UART.** Puertos serial por hardware

BIBLIOGRAFIA

CALDERON, M., *Implementación de un prototipo de gafas y bastón electrónico creando una red inalámbrica de comunicación y alerta para personas con discapacidad visual*. [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2018. [Consulta: 13 de diciembre del 2019]. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/11018>

CEVALLOS, A. y CHILLAN, E., *Construcción de un control electrónico de una cerradura eléctrica de 4 dígitos con alarma de seguridad para un departamento* [en línea]. (pregrado). Escuela Politécnica Nacional, 2019. [Consulta: 9 de septiembre del 2019]. Disponible en: <http://bibdigital.epn.edu.ec/handle/15000/2099>.

CALI, E., y TASIGCHANA, V., *Propuesta de diseño de un sistema de bicicletas alternativo sostenible para el cantón Riobamba* [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2019. [Consulta: 23 noviembre 2019]. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/11487>

EBNER, R., *Keep My Bike : plataforma digital para guardar bicicletas de forma segura* [en línea]. (Trabajo de titulación) (Maestría). Universidad de Chile, Santiago, 2018 [Consulta: 24 de septiembre del 2019]. Disponible en: <http://repositorio.uchile.cl/handle/2250/168010>

FERNANDEZ, N., *SISTEMA DE ALARMAS, TRANSMISIÓN Y MONITOREO DE DATOS APLICADO A LA MEDICIÓN DE VARIABLES CON REDES GSM/GPRS*. [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2019. [Consulta: 22 noviembre 2019]. Disponible en: <http://repositorio.ucm.edu.co:8080/jspui/bitstream/handle/10839/1495/Neson%20Fernando%20Martinez%20F.pdf?sequence=1&isAllowed=y>

GARZÓN, I. y GÓMEZ, N., *Desarrollo de un sistema de comunicación funcional para un ciclo parqueadero inteligente*. [en línea], (pregrado). Universidad Piloto De Colombia, 2018 [Consulta: 2 septiembre 2019]. Disponible en: <http://repository.unipiloto.edu.co/handle/20.500.12277/4818>.

GRANIZO, C., SANTILLAN, M. y CONDE, A., *Diseño e implementación de dos módulos para el control y envío de notificaciones de ingreso de usuarios a laboratorios en tiempo real, con tarjetas ARDUINO por medio de la RED GSM*, [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2015. [Consulta: 22 agosto 2019]. Disponible en:

<http://bibdigital.epn.edu.ec/handle/15000/2099w>

HUIDROBO, H., *Comunicaciones móviles: sistemas GSM, UMTS y LTE*. [Consulta: 22 septiembre 2019]. Madrid, SPAIN: Grupo Editorial RA-MA, 2014 Disponible en: <http://ebookcentral.proquest.com/lib/epochsp/detail.action?docID=5758921>

HERNANDEZ, A., *Propuesta de diseño de una ciclovía interconectada al sistema de transporte público urbano para el casco central de la ciudad de Ambato*. [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2019. [Consulta: 22 noviembre 2019]. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/10097>

NOBOA, E., *Implementación de una red Inalámbrica con Bloques Inteligentes para la supervisión en tiempo real y prevención de incendios en la planta de INDUFRICK*. [en línea]. (Pregrado). Escuela Superior Politécnica de Chimborazo, 2019. [Consulta: 7 agosto 2019]. Disponible en: <http://dspace.espoch.edu.ec/bitstream/123456789/2694/1/85T00253.pdf>.

OLAYA, J., y PINTOR, N., *Parqueadero de bicicletas automatizado*. [Consulta: 21 de septiembre del 2019]. Compilador CCCS y simulador PROTEUS para microcontroladores PIC. Barcelona, España, Alfaomega Grupo Editor S.A, 2008. 263 p. Disponible en: <https://repository.uniminuto.edu/handle/10656/1405>

PROAÑO, J., y SUAREZ, M., *Implementación de un prototipo de sistema electrónico mediante comunicación inalámbrica para supervisión y detección de inundaciones*. [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2016. [Consulta: 13 diciembre 2019]. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/6446>

QUICALIQUIN, D., *Implementación de un prototipo: sistema automático de censado inalámbrico basado en RFID para estaciones de control de buses urbanos de Riobamba*. [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2016 [Consulta: 24 de septiembre del 2019]. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/9146>

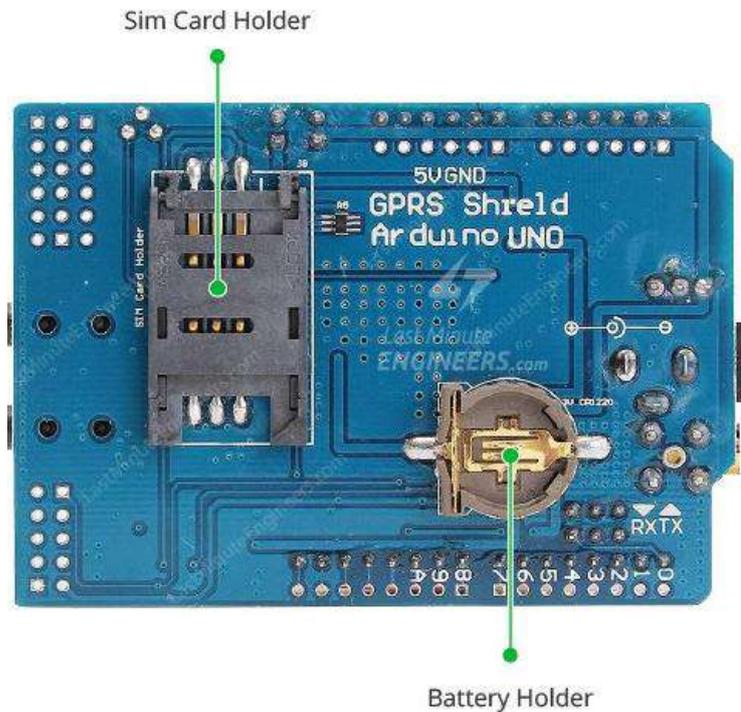
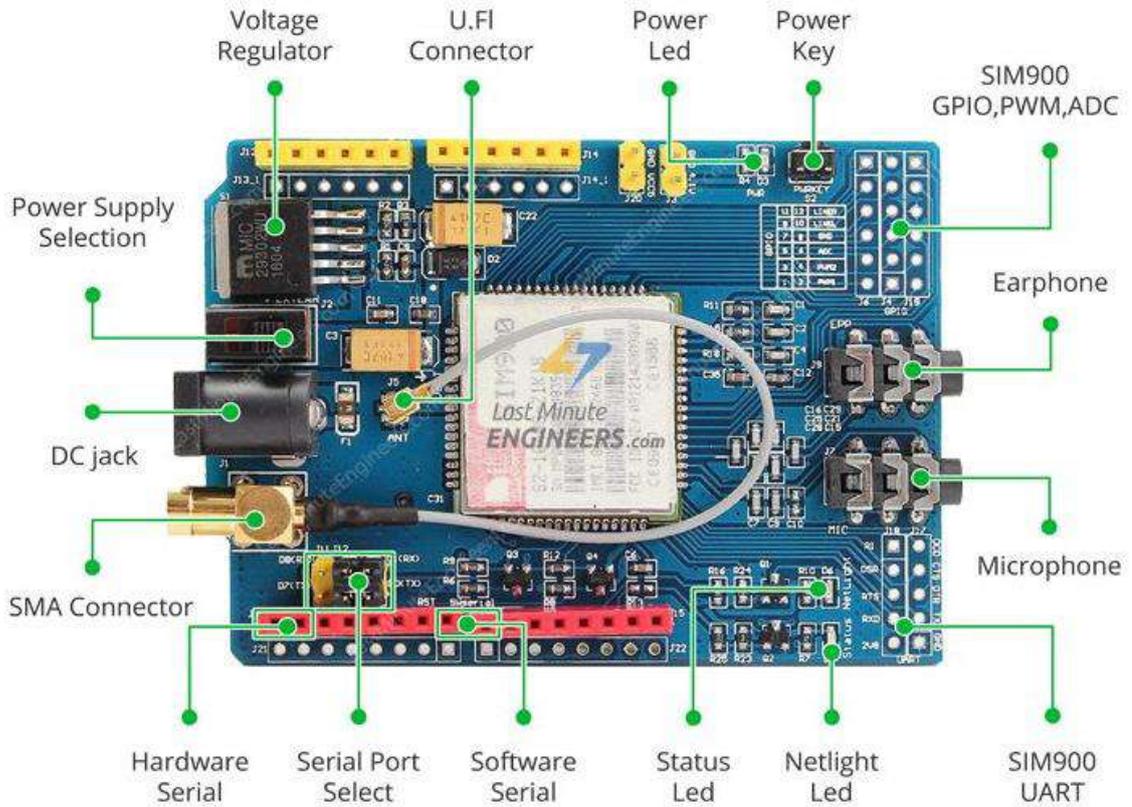
RODRIGUEZ, A., *Diseño de un sistema dispensador de alimento para los perros desamparados de la ESPOCH mediante tecnología inalámbrica integrada a una plataforma IoT*. [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2018. [Consulta: 15 de septiembre del 2019]. Disponible en: <http://dspace.espoch.edu.ec/handle/123456789/10926>

VILAÑEZ, A., *Implementación de un prototipo de sistema de seguridad doméstico basado en WPAN para una red IoT*. [en línea]. (pregrado). Escuela Superior Politécnica de Chimborazo, 2019. [Consulta: 29 agosto 2019]. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/13493>

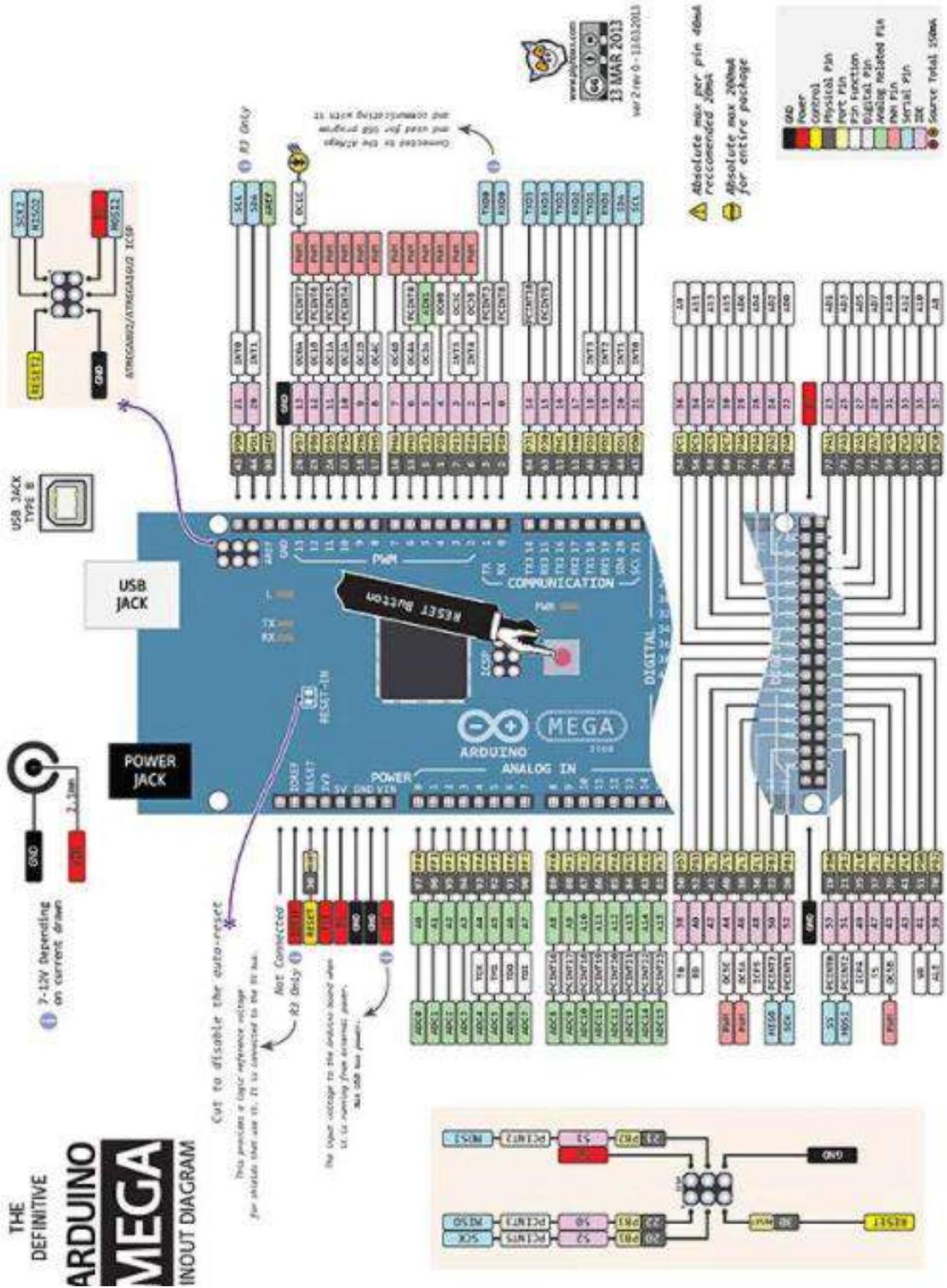
VILLA, A., *El uso de la bicicleta como alternativa de transporte sostenible e inclusivo para Lima Metropolitana: recomendaciones desde un enfoque de movilidad* [en línea]. (Trabajo de titulación) (Maestría). Pontificia Universidad Católica del Perú, 2015 [Consulta: 21 de septiembre del 2019]. Disponible en: <http://tesis.pucp.edu.pe/repositorio/handle/20.500.12404/6484>

ANEXOS

Anexo A: Data sheet SIM900



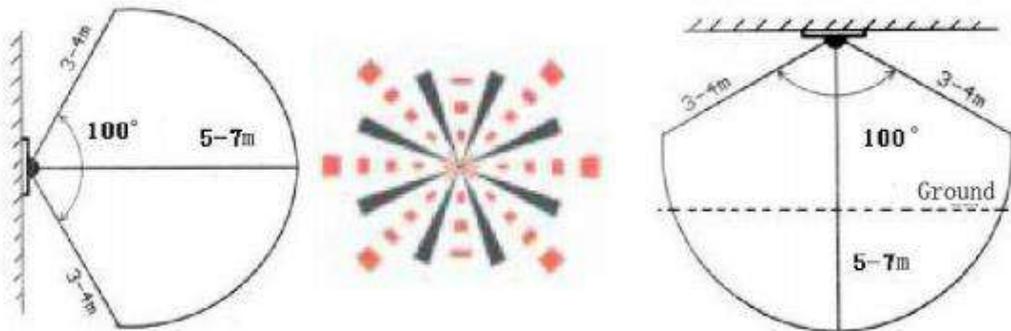
Anexo B: Data sheet Arduino Mega



Anexo C: Data sheet HC-SR501

Product Type	HC--SR501 Body Sensor Module
Operating voltage range	DC 4.5-20V
Quiescent Current	<50uA
Level output	High 3.3 V /Low 0V
Trigger	L can not be repeated trigger/H can be repeated trigger(Default repeated trigger)
Delay time	5-200S(adjustable) the range is (0.xx second to tens of second)
Block time	2.5S(default)Can be made a range(0.xx to tens of seconds)
Board Dimensions	32mm*24mm
Angle Sensor	<100 ° cone angle
Operation Temp.	-15-+70 degrees
Lens size sensor	Diameter:23mm(Default)

Induction Range:



Anexo D: Código fuente del software de aplicación

```

//librerias
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <SoftwareSerial.h>
#include <String.h>
//sim 900
//SoftwareSerial SIM900(23, 25); // Configura el
puerto serial para el SIM GSM
//--teclado
#include <Keypad.h> //llama a la librería
Keypad.h
//I2C pins declaration-----
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3,
POSITIVE);
//--pin de sensor pir
#define pirPin 12
//--pin de alarma
#define ledPin 13
//--pin de candado1 pin 27-- candado2 pin 29--
int PinIN1 = 31;
int PinIN2 = 33;
int PinIN3 = 35;
int PinIN4 = 37;
//--leds
#define candado 27
#define candado1 29
//sensor de proximidad1
const int Trigger = 10; //Pin digital 2 para el
Trigger del sensor
const int Echo = 11; //Pin digital 3 para el Echo
del sensor
//sensor de proximidad2
const int Trigger1 = 42;
const int Echo1 = 43;
// Create variables:
int val = 0;
bool motionState = false; // We start with no
motion detected.
int mov=0;
int cod=0;
int cod1=0;
int coda=0;
int cand=0;

// obtencion de 1 mensaje
int trama=0;
String p="";
char incoming_char=0;

char inicio_clave;
char clave1;
String clave = "";

char num;
char country_code_1;
char country_code_2;
char country_code_3;
//--codigo
int randomNumber;
char contrasena[4]="2222"; //aquí escribimos la
contraseña de 4 dígitos
char codigo[4]; //Cadena donde se
guardaran los caracteres de las teclas
presionadas
int cont=0; //variable que se incrementara
al presionar las teclas
int pass1;
int pass2;
int pass3;
int pass4;
String numero;
String numero1;
char customKey;
//matrices del teclado
const byte ROWS = 4; //Numero de filas del
teclado que se esta usando
const byte COLS = 4; //Numero de columnas del
teclado que se esta usando
//sim900
//char incoming_char=0; //Variable que guarda
los caracteres que envia el SIM GSM
//int salir = 0;
String Textsmsn="";

//Menu
char opcion='0';
int cont1=0; //imreso o no menu principal
int cont2=0;
//sensores
long t; //timepo que demora en llegar el eco
long d; //distancia en centimetros
long t1; //timepo que demora en llegar el eco
long d1; //distancia en centimetros

char hexaKeys[ROWS][COLS] = //Aquí
pondremos la disposición de los caracteres tal
cual están en nuestro teclado
{
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};

byte rowPins[ROWS] = {9, 8, 7, 6};
//Seleccionamos los pines en el arduino donde
iran conectadas las filas
byte colPins[COLS] = {5, 4, 3, 2};
//Seleccionamos los pines en el arduino donde
iran conectadas las filas

Keypad customKeypad =
Keypad(makeKeymap(hexaKeys), rowPins,
colPins, ROWS, COLS); //inicializa el teclado

void setup()
{
  Serial.begin(19200); //inicializar puerto serie
  Serial1.begin(19200); //Configura velocidad
  serial para el SIM
  lcd.begin(16, 2); // medidas de LCD 16 x 2
  lcd.print("Inicializando...");
  delay(5000);
  Serial.println("Inicializando...");
  delay(25000); //Retardo para que encuentra a
  una RED
  pinMode(Trigger, OUTPUT); //pin como
  salida
  pinMode(Echo, INPUT); //pin como entrada
  digitalWrite(Trigger, LOW); //Inicializamos el
  pin con 0
  pinMode(Trigger1, OUTPUT); //pin como
  salida
  pinMode(Echo1, INPUT); //pin como entrada
  digitalWrite(Trigger1, LOW); //Inicializamos
  el pin con 0
  // Configure the pins as input or output:
  pinMode(ledPin, OUTPUT);
  pinMode(candado, OUTPUT);
  pinMode(candado1, OUTPUT);
  pinMode(pirPin, INPUT);
  pinMode(PinIN1, OUTPUT);
  pinMode(PinIN2, OUTPUT);
  pinMode(PinIN3, OUTPUT);
  pinMode(PinIN4, OUTPUT);
  Serial.println("OK"); //Mensaje OK en el
  arduino, para saber que todo va bien.
  //pantalla de inicio
  lcd.setCursor(0, 1);
  lcd.print("Ver niveles");

  delay(3000);
}

void loop()
{
  //--
  opcion=customKeypad.getKey();
  if (opcion){

    switch (opcion) {

      case '1':
        menu1();
        char opcion1;

        do{
          opcion1 = customKeypad.getKey();
          if (d>4 && opcion1=='1'){
            MotorHorario();// meter
            piston//desabilitar
            MotorStop();//parar//desabilitar
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("20s para recibir");
            lcd.setCursor(0, 1);
            lcd.print("sms");
            digitalWrite(candado, HIGH);
          } //desactivar
          gencod();
          Serial.println("Clave:
          "+pass1+pass2+pass3+pass4);
          comandosAT_2();
          lcd.clear();
          lcd.setCursor(0, 0);
          lcd.print("Verifique el cod");
          lcd.setCursor(0, 1);
          lcd.print("en Thinkspeak");
          delay(10000);
          lcd.clear();
        }

        gencod();
        Serial.println("Clave:
        "+pass1+pass2+pass3+pass4);
        comandosAT_2();
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Verifique el cod");
        lcd.setCursor(0, 1);
        lcd.print("en Thinkspeak");
        delay(10000);
        lcd.clear();
      }
    }
  }
}

```



```

digitalWrite(Triple1, HIGH);
delayMicroseconds(10); //Enviamos un
pulso de 10us
digitalWrite(Triple1, LOW);
t1 = pulseIn(Echo1, HIGH); //obtenemos el
ancho del pulso
d1 = t1/59; //escalamos el tiempo a una
distancia en cm
Serial.print("Distancia1: ");
Serial.print(d1); //Enviamos serialmente el
valor de la distancia
Serial.print("cm ");
//Serial.println();
delay(100); //Hacemos una pausa de
100ms

// Read out the pirPin and store as val:
val = digitalRead(pirPin);
// If motion is detected (pirPin = HIGH), do
the following:
if (val == HIGH) {
  //digitalWrite(ledPin, HIGH); // Turn on the
on-board LED.
  // Change the motion state to true (motion
detected):
  //if (motionState == false) {
  mov=1;
  Serial.println("Motion detected:-----
-----");
  // motionState = true;
  //}
}
// If no motion is detected (pirPin = LOW), do
the following:
else {
  //digitalWrite(ledPin, LOW); // Turn off the
on-board LED.
  // Change the motion state to false (no
motion):
  //if (motionState == true) {
  mov=0;
  Serial.println("Motion
ended-----");
  // motionState = false;
  //}
}

if
(((d<=4)&&d1<4)&&(mov==1)&&(coda==0)&&cod1==0))
||((numero!=0)&&numero!=0)&&(d<=4)&&(mo
v==1))){
  digitalWrite(ledPin, HIGH);
  /* abilitar
  mensaje_sms("Peligro de hurto");
  llamar();
  */
}
else{
  digitalWrite(ledPin, LOW);
}
/*
if
(((d>4)&&(1)&&(cod==1))||1)&&(cod==1)&&(mov==1
))){
  digitalWrite(candado, HIGH);
  //llamar(); //Llama
}
else{
  digitalWrite(candado, LOW);
}
*/
//comandosAT_2(); //Llama a la función
comandosAT

/*
if(codigo[0]==0'&&codigo[1]==0'&&codigo[2]
==0'&&codigo[3]==0'){
  digitalWrite(candado, LOW);
}
}
*/

}

//--generar-codigo--
void genCod()
{
  pass1=random(1,9);
  pass2=random(1,9);
  pass3=random(1,9);
  pass4=random(1,9);
  contrasena[0]=(char) pass1;
  contrasena[1]=(char) pass2;
  contrasena[2]=(char) pass3;
  contrasena[3]=(char) pass4;
}

//-- ingreso de codigo de cicloparquedero--
teclado
void codigopar()
{
  customKey = customKeyPad.getKey(); //se
guarda en la variable customKey el caracter de
la tecla presionada
  if (customKey != NO_KEY) //se evalúa si se
presionó una tecla
  {
    codigo[cont]=customKey; //se guarda
caracter por caracter en el arreglo codigo[]
    Serial.println(codigo[cont]); //se imprime en
el puerto serie la tecla presionada

    lcd.setCursor(0, 0);
    lcd.println("Clave:");
    lcd.setCursor(cont+7, 0);
    lcd.println(codigo[cont]);
    cont=cont+1; //incrementamos la variable
cont
    if(cont==4) //si ya fueron presionadas 4 teclas
se evalúa si la contraseña es correcta
    {
      if(codigo[0]==contrasena[0]&&codigo[1]==cont
rasena[1]&&codigo[2]==contrasena[2]&&codig
o[3]==contrasena[3]){
        cod=1;
      }
      else{
        cod=0;
      }
      cont=0; //resetear a 0 la variable cont
    }
  }
}

//-- ingreso de codigo de cicloparquedero--
teclado
void codigopar1()
{
  if(clave[0]==contrasena[0]&&clave[1]==contras
ena[1]&&clave[2]==contrasena[2]&&clave[3]=
=contrasena[3]){
    cod=1;
  }
  else{
    cod = 0;
  }
}

//-----gsm-----
void llamar()
// Función que permite llamar a un celular local
{
  Serial1.println("ATD +593996428962");
  //Celular
  //Serial1.println("ATD "+ numero); //Celular
numero o numero1
  delay(100);
  Serial1.println();
  delay(20000); // wait for 30 seconds...
  Serial1.println("ATH"); // Cuelta el telefono
delay(1000);
}

void mensaje_sms(String Textomsn)
//Funcion para mandar mensaje de texto
{
  Serial1.print("AT+CMGF=1\r"); // AT
command to send SMS message
  delay(100);

  Serial1.println("AT+CMGS=\""+593996428962\
""); // recipient's mobile number, in
international format

  //Serial1.println("AT+CMGS=\""+numero+"\"");
  ;
  delay(100);
  Serial1.println(Textomsn); // message to send
delay(100);
  Serial1.println((char)26); // End AT command
with a ^Z, ASCII code 26 //Comando de
finalizacion
  delay(100);
  Serial1.println();
  delay(5000); // Tiempo para que se envíe el
mensaje
  Serial.println("SMS sent successfully");
}

void obtenermsn()
  Serial1.print("AT+CMGF=1\r"); //
Configuración del módulo GSM en modo texto.
  delay(100);
  Serial1.print("AT+CNMI=2,2,0,0,0\r"); /* El
modulo enviará los mensaje
directamente al puerto serial del
dispositivo. */
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.println("Esperando");
  lcd.setCursor(0, 1);
  lcd.println("mensaje");
  delay(10000);
  if (Serial1.available())>0{
    incoming_char=Serial1.read(); //Lectura de
datos del puerto serial.
    if((trama == 0) && (incoming_char == 'C')){
      trama = 1;
      p="";
      clave="";
    }
    if((trama == 1) && (incoming_char == 'M')){
      trama = 2;
    }
    if((trama == 2) && (incoming_char == 'T')){
      trama = 3;
    }
    if((trama == 3) && (incoming_char == ':')){
      trama = 4;
    }
    if((trama == 4) && (incoming_char == ' ')){
      trama = 5;
    }
    if((trama == 5) && (incoming_char == "")){
      trama=6;
      for(int i=0;i<13;i++){
        num = Serial1.read(); //Captura del
número remitente. ej: +593982832027
        if (num == "") break;
        else
          p += num;
      }
      Serial.println("Numero telefonico: ");
      Serial.println(p); // Imprimo en el puerto
serial el número telefónico.
      country_code_1 = (char)p[1];
      country_code_2 = (char)p[2];
      country_code_3 = (char)p[3];
      // código país 593 para Ecuador
      if
      ((country_code_1=='5')&&(country_code_2
=='9')&&(country_code_3=='3'))
        Serial.println("Numero correcto"); /* Si el
número telefónico corresponde

```



```

    }
    numero=p;
}

else if (d1>5){
    mensaje_sms("recibido");
    gencod();
    comandosAT_2();
    mensaje_sms("envie la clave");//sms al dueño
del numero
    delay(5000);
    obtenermsn();
    codigopar1();
    cod1=cod;
    if
    (((d1>5)&(p==numero1)&(cod1==1))||((p==nu
mero1)&(cod1==1)&(mov==1)))){
        MotorHorario1();// meter piston
        MotorStop1();//parar
        digitalWrite(candado1, HIGH);
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("5s restantes");
        lcd.setCursor(0, 1);
        lcd.print("para aparcar");
        delay(5000);
        //MotorAntihorario1();//cerrar piston
        //MotorStop1();//parar
        digitalWrite(candado1, LOW);
    }
    else{
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("No coinside");
        lcd.setCursor(0, 1);
        lcd.print("el codigo");
    }
    numero1=p;
}
}
}

```

Anexo D: Preguntas de la encuesta realizada y obtención de resultados

¿Qué tipo de notificación en caso de intento de robo le parece segura?			
R\Opciones	No	%	
Alarma		4	14,81%
Llamada		1	3,70%
Mensaje de texto		3	11,11%
Todos		19	70,37%
Total general		27	100,00%

¿Le parece muy dócil el manejo por teclado de la interfaz?			
R\Opciones	No	%	
No		0	0,00
Si		27	100,00%
Total general		27	100,00%

¿Le parece tan seguro el ciclo-parqueadero inteligente?			
R\Opciones	No	%	
No		2	7,41%
Si		25	92,59%
Total general		27	100,00%

¿Como estuvo su interacción con el ciclo-parqueadero?			
R\Opciones	No	%	
Bueno		1	3,70%
Excelente		14	51,85%
Muy bueno		12	44,44%
Total general		27	100,00%

Realizado por: Angel Caiza, 2020.



**ESCUELA SUPERIOR POLITÉCNICA DE
CHIMBORAZO**



**DIRECCIÓN DE BIBLIOTECAS Y RECURSOS
PARA EL APRENDIZAJE Y LA INVESTIGACIÓN**

UNIDAD DE PROCESOS TÉCNICOS
REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 19 / 01 / 2021

INFORMACIÓN DEL AUTOR	
Nombres – Apellidos: ANGEL ORLANDO CAIZA CHIMBORAZO	
INFORMACIÓN INSTITUCIONAL	
Facultad: INFORMÁTICA Y ELECTRÓNICA	
Carrera: INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES Y REDES	
Título a optar: INGENIERO EN ELECTRÓNICA, TELECOMUNICACIONES Y REDES	
f. Analista de Biblioteca responsable:	 Firmado electrónicamente por: ELIZABETH FERNANDA AREVALO MEDINA

