



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE QUINTANA ROO

DIVISIÓN DE CIENCIAS, INGENIERÍA Y TECNOLOGÍA

IMPLEMENTACIÓN DE UN SISTEMA DE
ADMINISTRACIÓN DE ACCESO A ESPACIOS DE
LA INGENIERÍA EN REDES EN LA DCIT

TESIS

PARA OBTENER EL GRADO DE
INGENIERA EN REDES

PRESENTA

IDANIA MARTÍNEZ TORRES

DIRECTOR

DR. JAVIER VÁZQUEZ CASTILLO

ASESORES

MTI. VLADIMIR VENIAMÍN CABAÑAS VICTORIA

MTI. MELISSA BLANQUETO ESTRADA

MSI. LAURA YÉSICA DÁVALOS CASTILLA

DR. JAIME SILVERIO ORTEGÓN AGUILAR



CHETUMAL QUINTANA ROO, MÉXICO, MAYO DEL 2024



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE QUINTANA ROO

DIVISIÓN DE CIENCIAS, INGENIERÍA Y TECNOLOGÍA

TRABAJO DE TESIS TITULADO

“IMPLEMENTACIÓN DE UN SISTEMA DE ADMINISTRACIÓN DE ACCESO A
ESPACIOS DE LA INGENIERÍA EN REDES EN LA DCIT”

ELABORADO POR
IDANIA MARTÍNEZ TORRES

BAJO SUPERVISIÓN DEL COMITÉ DEL PROGRAMA DE LICENCIATURA Y
APROBADO COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE:

INGENIERA EN REDES

COMITÉ DE TESIS

DIRECTOR:

DR. JAVIER VÁZQUEZ CASTILLO

ASESOR:

MTI. VLADIMIR VENIAMIN CABAÑAS VICTORIA

ASESORA:

M.T.I. MELISSA BLANQUETO ESTRADA

ASESORA SUPLENTE:

MS.I. LAURA YÉSSICA DAVALOS GARCÍA

ASESOR SUPLENTE:

DR. JAIME SILVERIO ORTEGÓN AGUIAR



CHETUMAL QUINTANA ROO, MÉXICO, MAYO DEL 2024



RESUMEN

El proyecto aquí presentado consiste en la implementación de un sistema de administración de acceso a espacios de laboratorios del departamento de informática y redes de la División de Ciencias e Ingeniería de la Universidad Autónoma del Estado de Quintana Roo (UQROO). El sistema propuesto utiliza una base de datos MariaDB y un dispositivo compuesto por una Raspberry Pi, una cámara que funciona como lector de códigos QR, una pantalla para visualizar la lista de actividades que el usuario requiera y un botón para interactuar con el sistema. También se complementa con una aplicación gratuita descargada desde la AppStore de Android para generar códigos QR e interactuar con el sistema propuesto.

El objetivo principal del sistema es poder monitorear las actividades que se realizan dentro de los espacios de la carrera de Ingeniería en Redes de la UQROO y con ellos realizar diversos análisis de uso a las instalaciones, así como conocer las necesidades de dichas áreas.

El sistema utiliza una base de datos basada en MariaDB, la cual se utiliza para almacenar la información generada por los estudiantes al hacer uso de las instalaciones del plantel educativo, logrando con esto, conocer el uso real de los espacios educativos para gestionar los recursos y orientarlos a mejorar los espacios con mayor uso. Todo lo anterior, con la finalidad de detectar cuales espacios son menos utilizados y poder tomar decisiones que generen un impacto positivo en la comunidad estudiantil. Para ello, un código QR personalizado para cada alumno, invitado o profesor, es asignado con la finalidad de que nos permita la interacción con el sistema.

Para llevar a cabo la lectura de los códigos QR, se emplea un dispositivo compuesto por una Raspberry Pi y una cámara. La Raspberry Pi actúa como el cerebro del sistema, el cual obtiene las imágenes capturadas por la cámara y registra una acción. La cámara, por su parte, se encarga de capturar los códigos QR que cada usuario tenga.

Cuando un usuario acceda a un espacio, presentará el código QR al dispositivo. La cámara captura la imagen del código y la Raspberry Pi envía a la base de datos una solicitud que consiste en una lista de actividades que previamente serán asignadas por el administrador del sistema. El sistema mostrará una lista de las actividades que se realizarán en el área y al seleccionar la actividad está se registrará en la base de datos.

AGRADECIMIENTOS

A Omar Ríos Arreola, por todo su apoyo, por orientarme, motivarme y brindarme sus conocimientos con paciencia, cariño y por creer en mis capacidades y encaminarme a continuar con mis siguientes metas.

A mi familia, en especial, a mi hermanita Gisele Berenice quien me motivó desde el principio a fin del trayecto de la carrera, creer en mí y confiar en que podría lograrlo, sobre todo y con sus mensajes decirme que no importa el tiempo, que no importa la edad en la que se empiece algo ni el tiempo que se lleve, que las circunstancias son distintas para todos y que lo más importante es tener el valor de esforzarse y perseverar.

Gracias a todas aquellas amistades y compañeros quienes participaron en mi trayecto a la meta además de aportar mucho conocimiento y reiterar que el trabajo en equipo es imprescindible, agradezco por toda su amistad, compañerismo, amabilidad, cordialidad y por brindar una hermosa experiencia de vida. Gracias Carlos Alfredo y todos los demás apreciados compañeros.

Finalmente agradezco infinitamente a mis profesores, quienes brindaron su conocimiento, paciencia, amabilidad, excelente trato, y su tiempo y al resolver mis dudas y con ello apoyar a una persona a llegar a la meta. Gracias Dr. Javier Vázquez Castillo, MTI. Vladimir Veniamín Cabañas Victoria, Mtro. Rubén, MTI. Melissa Blanqueto Estrada, Dr. Jaime Silverio Ortegón Aguilar, Dr. Homero, Mtro. Raúl y MSI. Laura Yésica Dávalos Castilla.

DEDICATORIA

A mi familia, principalmente a mi hermanita, quien me ha motivado. A mi novio Omar por estar a mi lado y apoyarme en los momentos difíciles.

Contenido

Tabla de ilustraciones.....	xi
1. Introducción.....	1
1.1 Importancia del control de acceso a laboratorios.	1
1.2 Aplicaciones de control de acceso por tarjeta	3
1.3 Herramientas para desarrollo del software	6
1.4 Justificación	12
1.5 Objetivos	13
1.5.1 Objetivo general	13
1.5.2 Objetivos específicos	13
2 Marco teórico.....	15
2.1 Sistemas de administración de control de acceso.....	17
3 Desarrollo.....	17
3.1 Requerimientos del sistema	17
3.2 Fase de planificación	19
3.2.1 Descripción de interés de elaboración del sistema	19
3.2.2 Historias de usuario	19
3.3 Fase de diseño	20
3.3.1 Elementos utilizados para implementar aplicaciones de desarrollo.....	20
3.3.2 Arquitectura del sistema	23
Hardware.....	24
3.3.3 Diagrama de conexión	27
3.3.4 Pruebas	30
Software	31
2.4.2 Framework	33
Resultados del desarrollo	34
Hardware	34

Software.....	34
Conclusiones.....	37
Referencias.....	38
Anexo A.....	39
Anexo B.....	43
Creación de tablas.....	43
Anexo C.....	46
Anexo D.....	50

TABLA DE ILUSTRACIONES

Figura 1. Artículo de Amazon	5
Figura 2. Artículo de mercado libre	6
Figura 3, Push button, imagenes obtenidas de google.	22
Figura 4. Diagrama de arquitectura del sistema.	23
Figura 5. Raspberry Pi Model B V1.2, 2015, imagen obtenida de Google.	24
Figura 6. Pantalla OLEDQT 0.960D, imagen obtenida de google.	25
Figura 7. Push button, imagen obtenida de google.....	26
Figura 8. Webcam reciclada	26
Figura 9. Conexión en pines de la Raspberry.	27
Figura 10. Conexiones en pines GPIO de Raspberry Pi	28
Figura 11. Sistema de administración de acceso de ARE 1.....	28
Figura 12. Actividades que tiene permitido el usuario dentro del área.	29
Figura 13. Tiempo agotado para el registro de acceso.	29
Figura 14. Entorno Thonny	31
Figura 15. Diagrama E-R de la base de datos.	32
Figura 16. Diagrama UML de funcionamiento de Sistema	33
Figura 17. Pantalla de base de datos.	35
Figura 18. Paso 1	39
Figura 19. Paso 2	40
Figura 20. Paso 3	40
Figura 21. Paso 4	41
Figura 22. . Paso 5	41
Figura 23. Paso 1, inicio de la aplicación.....	42
Figura 24. Paso 2, seleccionar el botón crear, escribir matricula	42
Figura 25. Paso 3, descargar el código QR	42

1. INTRODUCCIÓN

1.1 IMPORTANCIA DEL CONTROL DE ACCESO A LABORATORIOS.

Por lo general en cada edificio o institución en el que múltiples usuarios puedan acceder, es importante contar con un sistema de administración de control de acceso, debido a cuestiones de seguridad, monitoreo, etcétera. Es por ello que en la actualidad existen diversos tipos de sistemas de control de acceso que dependen del método de identificación; por ejemplo, aquellos detallados en el blog (nuoplanet, 2019) a continuación:

Manuales

Estos tipos de control de acceso prescinden de tecnologías y utilizan una persona física que se encarga de la identificación, autenticación y autorización.

Se trata de un sistema actualmente obsoleto por el bajo nivel de seguridad que proporciona, siendo dependiente de los potenciales errores humanos. Las identificaciones falsas y la falta de un registro o histórico fiable son dos de las vulnerabilidades más graves.

Por teclado

En este caso, se emplea un código que, introducido en un teclado, permite el acceso a las instalaciones. De nuevo, se considera uno de los tipos de control de acceso vulnerables y con varios problemas de seguridad:

- Es sencillo que las credenciales sean interceptadas por personas no autorizadas. Esto incluye desde la facilidad para compartirlas a la sencillez de observarlas a cierta distancia para utilizarlas.
- Resulta frecuente que los usuarios olviden los códigos. Frente a esto, suele optarse por credenciales excesivamente simples (“1111”), que son fáciles de memorizar, pero más arriesgadas.
- Es común que los teclados dejen de funcionar con el paso del tiempo, generando una inhabilitación general del sistema.

Huella dactilar

La huella dactilar destaca a día de hoy como el más seguro de todos los tipos de control de acceso. En este caso, se trata de una tecnología biométrica, encargada de convertir una parte física de una persona en su llave de acceso. Siendo la huella dactilar imposible de imitar, clonar y transferir, actualmente se trata del sistema más fiable y seguro.

Reconocimiento facial

Enmarcado entre los tipos de control de acceso biométrico, el reconocimiento facial supone una opción más vulnerable que las huellas dactilares porque resulta más fácil aplicar prácticas fraudulentas.

En concreto, es sencillo utilizar imágenes estáticas (por ejemplo, una imagen de la persona que debe ser identificada) para burlar el sistema en dispositivos de baja o media gama. Con la proliferación de imágenes personales en las redes sociales, este es un peligro especialmente significativo en la actualidad. Por otra parte, los dispositivos de alta gama son actualmente poco competitivos en precio y no están aún desarrollados en su totalidad a nivel seguridad.

Tarjeta identificativa

Este sistema emplea tarjetas identificativas, radiofrecuencia y lectores específicos (hoy, a menudo sin contacto y a distancia) para funcionar.

En este caso, existen diversos niveles de seguridad dependiendo del tipo de tarjeta que se elija, que puede ir desde ser muy segura a nada segura. Así, las tarjetas de proximidad de baja frecuencia (125 kHz) suponen un sistema anticuado y fácilmente clonable, ofreciendo una nula garantía de seguridad.

Las más modernas tarjetas de proximidad de alta frecuencia (13.56 MHz) cuentan, por otro lado, con modelos vulnerables y otros más seguros.

Por teléfono móvil

Estos tipos de sistemas de control de acceso suponen una evolución del sistema de proximidad de alta frecuencia. Mediante esta tecnología, se emplea el teléfono móvil personal de una persona en comunicación con un dispositivo cercano para identificar, autenticar y permitir el acceso.

Entre las ventajas del control de accesos a través del móvil se encuentran el mayor nivel de seguridad que ofrecen, ya que cuentan no solo con la encriptación creada para sistemas de radiofrecuencia, sino también con el sistema de seguridad del propio dispositivo móvil personal.

1.2 APLICACIONES DE CONTROL DE ACCESO POR TARJETA

Existen diversos tipos de sistemas de control de acceso físico por tarjeta, a continuación, se enlista los siguientes que se describen en (Termired, 2020):

Control de acceso por tarjeta magnética

El control de acceso con tarjeta magnética es una solución muy extendida. Esta opción usa lo que se conoce como “emulación de teclado”, mediante la cual se reflejan los datos personales en el software vinculado para el registro. Su introducción es muy parecida a la de los datáfonos, contando con una ranura como la que se usa para el pago con tarjeta de crédito.

Tarjeta Chip.

Con este tipo de productos, no se requiere que tarjeta y dispositivo entren en contacto, ya que, acercándola, puedes ser identificado correctamente gracias al sensor de proximidad. El modo de funcionamiento de la tarjeta chip, es muy similar a las tarjetas de crédito contactless, que se han difundido entre gran parte de la población.

Existen dos subcategorías de tarjetas chip, las cuales cuentan con distintas características: las microprocesadas, muy similares a la CPU y con un funcionamiento parecido al que incorporan los DNI; y las de memoria, que se graban mediante sistemas de lectura y escritura y se pueden acceder de forma libre o mediante un PIN.

Con código de barras.

Este modelo es el más tradicional a la hora de identificar a una persona mediante una tarjeta, ya que su metodología es más sencilla y económica que otras alternativas.

RFID.

La tarjeta RFID es aquella solución que recurre a la radiofrecuencia para identificar al portador del elemento en cuestión, registrando la información almacenada en una especie de etiqueta pegada a la tarjeta.

Entre los grandes beneficios del uso de estos modelos, se puede destacar su carácter multidireccional, lo que supone que no sea necesario que esté en la línea de detección del equipo para obtener la información de registro.

Smart Card.

Las tarjetas inteligentes o de proximidad son las que más han crecido en los últimos años por su comodidad y rapidez de uso, al no tener que hacerle contactar con el detector, incluir más información y ser mucho más precisas que otras alternativas.

Sobre los sistemas de control de accesos con tarjetas

Al igual que ocurre con muchos otros tipos de tecnologías, usar un control de acceso con tarjeta tiene una serie de ventajas y desventajas respecto a otras soluciones.

Las ventajas de usar un control de acceso con tarjeta son muy numerosas y muestra de ello es el extendido uso de estos modelos que se está haciendo de ellos. A continuación, exponemos las más destacadas:

- Seguridad: contar con un sistema de este tipo supone colocar una barrera precisa y eficaz ante el intento de entrada de personal no autorizado al interior de una construcción, por lo

que, de este modo, se mantiene la información sensible y la integridad del personal debidamente protegida.

- Control de la jornada y los descansos: en el entorno empresarial, estos sistemas son la mejor solución para lograr un registro de las horas trabajadas de los trabajadores, así como de las pausas en esa jornada, siempre recogidas de una forma precisa y real.
- Fácil vinculación con sistemas de domótica: las casas u oficinas inteligentes, que se valen de la domótica, ya están aquí y han dejado de ser una idea de ciencia-ficción. Los detectores de tarjetas pueden conectarse y coordinarse de manera sencilla con los softwares que se encargan de gestionar estas construcciones de forma automatizada.
- Adaptación: los productos presentes en el mercado del control de accesos con tarjeta se pueden adaptar a la perfección a cualquier escenario, formato, tamaño o tipo de acceso en el que se quiera instalar.

Por su parte, también pueden darse ciertos impedimentos para el correcto funcionamiento de estas soluciones tecnológicas destinadas a la identificación personal: no llevar la tarjeta encima, por descuido o extravío; problemas con la correcta detección, por contar con restos de suciedad que impiden su lectura; o el riesgo de falsificación de los datos, como cuando alguien, de manera consciente o no, utiliza una tarjeta que no le pertenece, alterando los datos que se registran.

Se realizó una investigación por medio de las tiendas más destacadas o conocidas en México (mercado libre y amazon) y se concluyó que los dispositivos más comprados por los usuarios mexicanos son los equipos con sistema de control de acceso RFID.

A continuación, se muestran los artículos más destacados de dichas tiendas online:

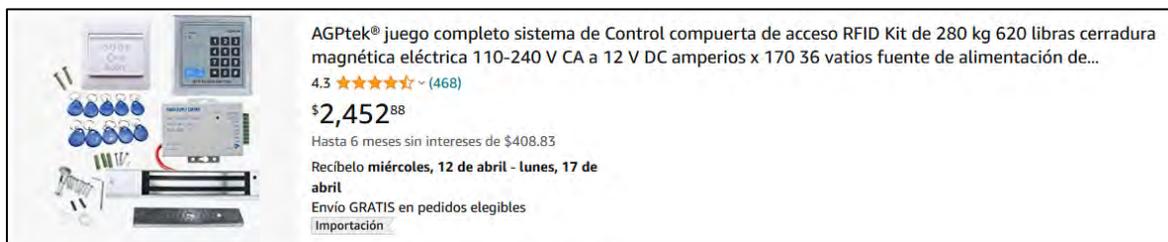


Figura 1. Artículo de Amazon



Figura 2. Artículo de mercado libre

“La RFID es un sistema de identificación de productos que puede parecer similar al código de barras tradicional, pero cuenta con grandes ventajas. A diferencia del código de barras, que utiliza la imagen para identificar una etiqueta colocada en un producto, la RFID utiliza las ondas de radio para comunicarse con un microchip, que puede estar montado sobre gran cantidad de soportes, como por ejemplo un tag o etiqueta RFID, una tarjeta o un transpondedor.” (Tecnipesa, 2021)

Sin embargo a pesar de las cualidades antes mencionadas, debido a las características y requerimientos de una administración de control de acceso a los espacios de ingeniería de redes en el área de DCIT sería poco practica la aplicación de un sistema con etiqueta RFID, ya que requeriría de costos de elaboración de estos microchips, además el objetivo de la implementación de dicho sistema no es de denegar o permitir acceso a dicha área, si no conocer la afluencia de alumnos y personal a dicha área para poder gestionar las necesidades ya sea espacios, optimización de equipos o muebles, entre otros.

1.3 HERRAMIENTAS PARA DESARROLLO DEL SOFTWARE

De acuerdo con (ORACLE, 2024): “Una base de datos es una recopilación organizada de información o datos estructurados, que normalmente se almacena de forma electrónica en un sistema informático. Normalmente, una base de datos está controlada por un sistema de gestión de bases de datos (DBMS). En conjunto, los datos y el DBMS, junto con las aplicaciones asociadas a ellos, reciben el nombre de sistema de bases de datos, abreviado normalmente a simplemente base de datos”.

En relación con los tipos de bases de datos más comunes, Oracle señala que los datos de los tipos más comunes de bases de datos en funcionamiento actualmente se suelen utilizar como estructuras de filas y columnas en una serie de tablas para aumentar la eficacia del procesamiento y la consulta de datos. Así, se puede acceder, gestionar, modificar, actualizar, controlar y organizar fácilmente los datos. La mayoría de las bases de datos utilizan un lenguaje de consulta estructurada (SQL) para escribir y consultar datos.

Las bases de datos han evolucionado drásticamente desde su inicio a principios de la década de 1960. Las bases de datos de navegación, como la base de datos jerárquica (que se basaba en un modelo de árbol y permitía una relación de uno a muchos) y la base de datos de red (un modelo más flexible que permitía relaciones múltiples), eran los sistemas originales que se utilizaban para almacenar y manipular datos. Aunque eran sencillos, estos primeros sistemas eran inflexibles. En la década de 1980, se hicieron populares las bases de datos relacionales, seguidas de las bases de datos orientadas a objetos en la década de 1990. Más recientemente, las bases de datos NoSQL surgieron como respuesta al crecimiento de Internet y la necesidad de acelerar la velocidad y el procesamiento de los datos no estructurados. Hoy en día, las bases de datos en la nube y las bases de datos de autogestión están abriendo nuevos horizontes en lo que respecta a la forma en la que se recopilan, se almacenan, se gestionan y se utilizan los datos.

Según Oracle que existen muchos tipos diferentes de bases de datos y señala que la mejor base de datos para una organización específica depende de cómo pretenda la organización utilizar los datos.

Bases de datos relacionales

- Las bases de datos relacionales se hicieron predominantes en la década de 1980. Los elementos de una base de datos relacional se organizan como un conjunto de tablas con columnas y filas. La tecnología de bases de datos relacionales proporciona la forma más eficiente y flexible de acceder a información estructurada.

Bases de datos orientadas a objetos

- La información de una base de datos orientada a objetos se representa en forma de objetos, como en la programación orientada a objetos.

Bases de datos distribuidas

- Una base de datos distribuida consta de dos o más archivos que se encuentran en sitios diferentes. La base de datos puede almacenarse en varios ordenadores, ubicarse en la misma ubicación física o repartirse en diferentes redes.

Almacenes de datos

- Un repositorio central de datos, un data warehouse es un tipo de base de datos diseñado específicamente para consultas y análisis rápidos.

Bases de datos NoSQL

- Una base de datos NoSQL, o base de datos no relacional, permite almacenar y manipular datos no estructurados y semiestructurados (a diferencia de una base de datos relacional, que define cómo se deben componer todos los datos insertados en la base de datos). Las bases de datos NoSQL se hicieron populares a medida que las aplicaciones WEB se volvían más comunes y complejas.

Bases de datos orientadas a grafos

- Una base de datos orientada a grafos almacena datos relacionados con entidades y las relaciones entre entidades.
- Bases de datos OLTP. Una base de datos OLTP es una base de datos rápida y analítica diseñada para que muchos usuarios realicen un gran número de transacciones.

Estos son solo algunos de las varias docenas de tipos de bases de datos que se utilizan hoy en día. Otras bases de datos menos comunes se adaptan a funciones científicas, financieras o de otro tipo

muy específicas. Además de los diferentes tipos de bases de datos, los cambios en los enfoques de desarrollo tecnológico y los avances considerables, como la nube y la automatización, están impulsando a las bases de datos en direcciones completamente nuevas.

Algunas de las bases de datos más recientes incluyen:

Bases de datos de código abierto

- Un sistema de base de datos de código abierto es aquel cuyo código fuente es de código abierto; tales bases de datos pueden ser bases de datos SQL o NoSQL.

Bases de datos en la nube

- Una base de datos en la nube es una recopilación de datos, estructurados o no estructurados, que reside en una plataforma de cloud computing privada, pública o híbrida. Existen dos tipos de modelos de bases de datos en la nube: el modelo tradicional y el de base de datos como servicios (database as a service, DBaaS). Con DBaaS, un proveedor de servicios realiza las tareas administrativas y el mantenimiento.

Base de datos multimodelo

- Las bases de datos multimodelo combinan distintos tipos de modelos de bases de datos en un único servidor integrado. Esto significa que pueden incorporar diferentes tipos de datos.

Bases de datos de documentos/JSON

- Diseñadas para almacenar, recuperar y gestionar información orientada a los documentos, las bases de datos de documentos son una forma moderna de almacenar los datos en formato JSON en lugar de hacerlo en filas y columnas.

Bases de datos de autogestión

- El tipo de base de datos más nuevo e innovador, las bases de datos de autogestión (también conocidas como bases de datos autónomas) están basadas en la nube y utilizan el machine learning para automatizar el ajuste de la base de datos, la seguridad, las copias de seguridad, las actualizaciones y otras tareas de gestión rutinarias que tradicionalmente realizan los administradores de bases de datos.

HeidiSQL

HeidiSQL es un software libre y tiene como objetivo que sea fácil de aprender. "Heidi" le permite ver y editar datos y estructuras desde computadoras que ejecutan uno de los sistemas de base de datos MariaDB, MySQL, Microsoft SQL, PostgreSQL y SQLite. Inventado en 2002 por Ansgar, HeidiSQL pertenece a las herramientas más populares para MariaDB y MySQL en todo el mundo. (HeidiSQL, s.f.)

Según la página [uptpdown](#), HeidiSQL es una excelente herramienta para gestionar una base de datos MySQL y constituye una seria alternativa a aplicaciones como phpMyAdmin. Proporciona una poderosa interfaz gráfica para administrar las tablas, los registros, y los usuarios de una base de datos en dicho formato.

Las principales actividades que puedes realizar con este programa son los siguientes:

- Generar informes SQL.
- Sincronizar la información contenida en dos tablas diferentes.
- Importar archivos de texto y exportar las tablas a otros formatos.

Pero aún hay más, el programa dispone de un pequeño gestor de usuarios que permite administrar sus privilegios de forma fácil y rápida. Además, con este programa podrás editar a tu antojo una base de datos: crear o eliminar tablas; insertar, editar y eliminar registros; incluir campos MEMO con soporte para Bitmap, GIF y JPEG; y mucho más.

MariaDB

El documento MariaDB Server Documentation generado de la base de conocimientos de MariaDB con fecha 2023-05-14 declara lo siguiente:

MariaDB Server es un sistema de gestión de base de datos relacional, de código abierto y de uso general, optimizado para el rendimiento y la facilidad de uso; tiene sus raíces en MySQL Server y es una alternativa a Postgres, Oracle Database y otras bases de datos relacionales y NoSQL.

MariaDB Server tiene sus raíces en MySQL Server. Comenzó como una bifurcación de MySQL Server, usando la misma licencia GPLv2.

Sin embargo, aunque la documentación de MySQL Server siempre estuvo disponible públicamente, nunca se publicó con una licencia de documentación gratuita. Esto significa que la documentación de MariaDB Server se creó desde cero, o mejor dicho, de los textos de ayuda en línea, que tenían una licencia abierta compatible que los hacía utilizables como punto de partida.

El lugar en el que se escribió la documentación fue etiquetado como "Base de conocimiento", por el creador de MySQL y MariaDB, Michael "Monty" Widenius. La base de conocimientos fue, y sigue siendo, un esfuerzo comunitario. Al igual que con muchos esfuerzos de la comunidad, hay colaboradores principales en torno a los cuales se centra el trabajo. Aquí es donde Daniel Bartholomew cargó el texto de ayuda en línea, como primera semilla. Durante aproximadamente los últimos diez años, el editor principal de la base de conocimientos de MariaDB ha sido Ian Gilfillan, que trabaja para la Fundación MariaDB y reside en Sudáfrica. Por lo tanto, su nombre está en la portada del libro. Sin embargo, hay una gran cantidad de otros colaboradores, muchos de los cuales provienen de MariaDB Corporation, tanto como desarrolladores de código como escritores de documentación.

Thonny

Thonny es un Python IDE (entorno de desarrollo integrado) apto para principiantes que permite a los usuarios con poco o ningún conocimiento de programación comenzar su primer paso en el camino de dominar el lenguaje de programación Python. (FileHorse, 2023)

Según un artículo de investigación publicado en la página web ACM DL Digital Library Thonny es un nuevo IDE de Python para aprender y enseñar programación que puede hacer que la visualización de programas sea una parte natural del flujo de trabajo de los principiantes. Entre sus características destacadas se encuentran diferentes formas de recorrer el código paso a paso, evaluación de expresiones paso a paso, visualización intuitiva de la pila de llamadas y modo para explicar los conceptos de referencias y montón. Apoya la investigación educativa al registrar las acciones del usuario para reproducir o analizar el proceso de programación. Es de uso gratuito y abierto para la extensión.

De 2014 a 2018, el desarrollo principal de Thonny tuvo lugar en el Instituto de Ciencias de la Computación de la Universidad de Tartu, Estonia. (Thony, 2018)

Python

Python es un lenguaje de programación interpretado, interactivo y orientado a objetos. Incorpora módulos, excepciones, tipos dinámicos, tipos de datos dinámicos de muy alto nivel y clases. Admite múltiples paradigmas de programación más allá de la programación orientada a objetos, como la programación procedimental y funcional. (Python Software Foundation, 2024)

1.4 JUSTIFICACIÓN

Actualmente, los espacios de la Ingeniería en Redes, como las aulas de cómputo, son utilizados por diversas carreras de la División de Ciencias, Ingeniería y Tecnología (DCIT), e incluso carreras de otras divisiones para diversas actividades como clases, talleres, conferencias, capacitación académica, entre otras. Hoy en día, no se cuenta con estadísticas de uso de las aulas de redes por parte de los alumnos y profesores de la Universidad (por ejemplo: por carrera, por asignatura de clase, por horario, por día, práctica de laboratorio, etc). Tales estadísticas, son de interés para mantenimiento de las aulas, programación de horarios, e incluso, en etapas de acreditación de las diversas carreras de la DCIT. El presente trabajo, busca implementar un sistema de hardware y software para administrar el ingreso a las aulas de la Ingeniería en redes con la finalidad de generar estadísticas de uso.

Asimismo, es fundamental destacar que la implementación del sistema debe permitirnos optimizar los recursos disponibles. En este sentido, la adopción de un sistema de monitoreo de acceso basado en códigos QR ofrece la ventaja de ser altamente rentable, ya que nos ofrece la opción de portar los códigos QR en nuestros dispositivos, es decir no se requiere la impresión física y por lo tanto es bastante flexible y económico.

Otra ventaja del sistema reside en el uso de MariaDB como gestor de base de datos, el cual es compatible con MySQL. MariaDB cuenta con una licencia de código abierto y cuenta con una amplia comunidad de usuarios, lo que facilita el acceso a información y recursos de manera sencilla.

Entonces a continuación se realizan las siguientes interrogantes:

¿Por qué es importante el implementar el sistema control de acceso en el laboratorio de DCTI de la carrera de ingeniería en redes?

Debido a cuestiones seguimiento y monitoreo.

¿Para que se requiere contar con un sistema de administración de acceso en el laboratorio de DCTI de la carrera de ingeniería en redes?

La implementación de un sistema de administración de acceso a espacios de la ingeniería en redes en la DCIT permitiría conocer las necesidades y requerimientos del área, lo cual coadyuvaría a enfocar los recursos para subsanar, complementar u optimizar la funcionalidad de equipos, inmuebles y espacios.

1.5 OBJETIVOS

1.5.1 Objetivo general

Implementar un sistema de administración de acceso a espacios de la Ingeniería en Redes en la DCIT.

1.5.2 Objetivos específicos

1. Realizar el análisis de requerimientos del sistema a desarrollar (hardware y software).

2. Proponer una arquitectura del hardware del sistema a desarrollar.
3. Proponer un sistema de software para control del acceso a laboratorios de la Ingeniería en Redes.
4. Realizar pruebas del sistema integrado bajo funcionamiento.
5. Redacción de la tesis.

2 MARCO TEÓRICO

Un sistema de control de acceso surgió debido a la necesidad de llevar el control de entradas y salidas en lugares específicos.

Sobre la importancia de contar con un sistema de control de acceso es relevante comprender lo siguiente:

“El control de acceso es un elemento esencial de la seguridad que determina quién tiene permiso para tener acceso a determinados datos, aplicaciones y recursos y en qué circunstancias. De la misma forma que las claves y listas de invitados con aprobación previa protegen los espacios físicos, las directivas de control de acceso protegen los espacios digitales. En resumen: permiten que la gente adecuada entre y que la que no lo es se quede fuera.” (Microsoft, 2024)

Para introducirnos en los avances de las tecnologías de control de acceso la página web WBSgo nos brinda una breve historia sobre la evolución de los sistemas de control de acceso:

La tecnología de control de acceso empresarial sigue evolucionando desde hace décadas. Su historia se remonta a los años 70, cuando se introdujeron los primeros sistemas de control de acceso basados en tarjetas. Se trataba de tarjetas perforadas para controlar el acceso a las instalaciones de las empresas.

Fue a mediados de la década de los 80 cuando se comenzaron a introducir los primeros sistemas de control de acceso basados en huellas dactilares. Se utilizaban para verificar la identidad de los usuarios en el contexto corporativo. A finales de la misma década, se introdujeron los primeros sistemas de reconocimiento basados en el iris.

A mediados de los 90, llegó el turno de los sistemas de reconocimiento por voz. Y a finales de esta década, el reconocimiento facial.

No obstante, a pesar de la popularidad de estos sistemas de control de acceso, la historia comenzó mucho antes con el control por medio de contraseñas. Es un método de seguridad que requiere que los usuarios ingresen una contraseña para acceder a un sistema o recurso. Esta técnica es utilizada comúnmente para proteger computadoras, sistemas, aplicaciones, redes u otros sistemas informáticos.

La historia del control de acceso con contraseña se remonta a los años 60. En un principio, estos sistemas se utilizaron principalmente para proteger los datos militares y de inteligencia. Con el tiempo, el control de acceso con contraseña se convirtió en una herramienta de seguridad comúnmente utilizada en entornos empresariales y gubernamentales. A mediados de los años noventa, comenzó a utilizarse más ampliamente en entornos domésticos. Esto se debió en parte a la proliferación de Internet y las computadoras personales.

Al permitir que los usuarios accedieran fácilmente a recursos y datos en línea, el control de acceso con contraseña se convirtió en una forma efectiva de proteger la privacidad y la seguridad de los usuarios. A medida que Internet se ha vuelto cada vez más integral en nuestras vidas, este tipo de control en el acceso se ha convertido en una de las principales formas de proteger nuestras cuentas y datos en línea.

Desde cuentas de correo electrónico y redes sociales, hasta bancos y tiendas virtuales. En definitiva, el control de acceso con contraseña es una de las maneras más populares de mantener seguros nuestros datos personales y financieros.

A pesar de que el uso de contraseñas data de los años 60, hemos adoptado pocas medidas para garantizar su seguridad. Ha sido en los últimos años cuando se ha puesto mayor énfasis en que estas contraseñas sean seguras, con combinaciones de letras mayúsculas y minúsculas, y números que complican la vida del usuario. Sin embargo, las vulnerabilidades siguen existiendo, ya que los usuarios acaban utilizando la misma contraseña en varios sitios, o en todos. Además, en la mayoría de los casos, utilizan contraseñas poco robustas para recordarlas (123456, password, qwerty, etc.).

2.1 SISTEMAS DE ADMINISTRACIÓN DE CONTROL DE ACCESO.

Actualmente existen gran variedad de sistemas, dispositivos e incluso aplicaciones móviles creados principalmente enfocados al controlar el acceso de personas hacia áreas o edificios restringidos, sin embargo en algunos caso se requiere obtener más información por medio de estos sistemas, por lo tanto es importante contar una base de datos que nos permita guardar registros sobre las actividades y concurrencia que se realiza en ciertas áreas, y debido a la naturaleza de algunos espacios se requieren decenas de identificadores, por lo tanto un sistema que permita tener como identificador un código QR, proporcionaría informes.

3 DESARROLLO

3.1 REQUERIMIENTOS DEL SISTEMA

Requerimientos:

Componentes del sistema:

Raspberry Pi: Es el dispositivo principal que captura y registra las entradas a través de una cámara.

Cámara: Se utiliza para capturar imágenes o videos que se registrarán en el sistema.

Computadora (laptop): Actúa como servidor de base de datos para almacenar y gestionar los registros.

Conexión de red:

El sistema permite la conexión a la red, ya sea a través de una conexión alámbrica o inalámbrica, para garantizar la transferencia de datos y el acceso al servidor de base de datos.

Captura de datos:

La Raspberry Pi está equipada con una cámara que captura el código QR.

El dato obtenido a través de la imagen se procesa, realiza una solicitud de respuesta y después se almacena en la base de datos.

Servidor de base de datos:

El servidor de base de datos, alojado en la computadora (laptop), almacena todos los registros de entrada.

Se utiliza un sistema de gestión de bases de datos (DBMS) como MySQL, PostgreSQL, o SQLite para gestionar y consultar los datos.

Almacenamiento y gestión de datos:

Los registros se almacenan en tablas de la base de datos.

Acceso y consulta de datos:

Se puede acceder a la información de la base de datos a través de la página web “sistema de registros” o en la base de datos a través de una consulta realizada por el usuario.

Escalabilidad:

El sistema debe ser capaz de gestionar un volumen creciente de registros a medida que se capturan más entradas.

Compatibilidad y requisitos del sistema:

Se detallan los requisitos mínimos para la computadora que actúa como servidor de base de datos, como el sistema operativo y la capacidad de almacenamiento.

3.2 FASE DE PLANIFICACIÓN

3.2.1 Descripción de interés de elaboración del sistema

Aquellos que buscan la implementación de un sistema de control de acceso pueden ser instituciones de cualquier naturaleza que requieran obtener información sobre el acceso a áreas específicas dentro de sus instalaciones. Además, es fundamental que el sistema pueda recopilar datos y que nos brinde información sobre las actividades más relevantes en las áreas designadas.

Entre los requisitos clave para este sistema se encuentran:

- Registro de actividades: debe ser capaz de registrar y almacenar información precisa sobre las actividades realizadas por los usuarios o visitantes en las áreas.
- Flexibilidad y personalización: debe ser configurable y adaptable a los requisitos específicos de cada institución, permitiendo la definición de políticas y perfiles personalizados.

Es fundamental que el sistema cumpla con los estándares de calidad, disponibilidad y usabilidad, asegurando una implementación eficiente y una interfaz intuitiva para los usuarios. Con estos requisitos formalizados, se espera que la implementación del sistema de control de accesos a áreas específicas cumpla con las expectativas y necesidades de las instituciones esperadas

3.2.2 Historias de usuario

Se elaboró el sistema a partir de las necesidades externadas por el usuario para obtener las funcionalidades del sistema de registro de acceso, lo cual se describe en la siguiente tabla:

Requerimiento	Descripción
Agregar personas	El usuario podrá agregar usuarios en la base de datos
Agregar actividades	El usuario podrá agregar actividades en la base de datos
Asignar actividades	El usuario podrá agregar actividades a las personas en la base de datos
Registro de datos	Se podrán almacenar los registros de accesos en la base de datos

3.3 FASE DE DISEÑO

Para la elaboración del sistema se implementaron conocimientos previos.

3.3.1 Elementos utilizados para implementar aplicaciones de desarrollo.

A continuación, se describen los elementos utilizados para la realización del dispositivo que se encargará de leer, enviar información y almacenar la información.

Tarjeta de desarrollo Raspberry.

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

La Raspberry Pi fue creada en febrero del 2012 por la Raspberry Pi Foundation, originalmente pensado para promover y enseñar las ciencias básicas de la computación en las escuelas y universidades de Reino Unido. Originalmente lanzaron dos modelos, el Modelo A y el Modelo B. Al poco tiempo de su lanzamiento ya había una comunidad formada por miles de “locos por la tecnología” que compraron una Raspberry para empezar a experimentar con nuevos proyectos.” (Raspberrypi.cl, s.f.)

Raspberry Pi es una computadora de 8.5 cm por 5.5 cm en la cual se puede realizar diferente tipo de proyectos orientados a la programación, el dispositivo corre el sistema operativo Linux.

Maria DB.

“Cuando el predecesor de MariaDB Server, MySQL, fue comprado por Oracle en 2009, el fundador de MySQL, Michael “Monty” Widenius, bifurcó (fork) el proyecto debido a preocupaciones sobre la administración de Oracle. Monty llamo al nuevo proyecto MariaDB. MySQL lleva el nombre de su primera hija, My, mientras que MariaDB lleva el nombre de su segunda hija, Maria.

La mayoría de los desarrolladores originales se unieron al nuevo proyecto y MariaDB Server ha seguido desarrollándose rápidamente desde entonces.

La Fundación MariaDB, una organización sin ánimo de lucro encargada de garantizar la apertura, la adopción y la continuidad de MariaDB Server, también se formó en 2012.

MariaDB Server es un sistema de gestión de bases de datos relacionales de código abierto. Es uno de los servidores de bases de datos más populares del mundo, con usuarios notables como Wikipedia, WordPress.com y Google. MariaDB Server se publica bajo la licencia de código abierto GPLv2 y se garantiza entonces que seguirá siendo abierto.

Se puede utilizar para datos de transacciones de alta disponibilidad, análisis de datos, como servidor integrado, y una amplia gama de herramientas y aplicaciones soportan MariaDB Server.” (MariaDB Foundation, 2024)

El equipo utilizado para la instalación de la base de datos (MariaDB) necesita que el equipo cuente con un mínimo de 512 MB de memoria RAM, 1 GB de espacio en disco duro y arquitectura del sistema de 32 o 64 bits, el sistema operativo utilizado en el proyecto es Windows 10 pro, sin embargo, no es necesario que sea este el sistema operativo, ya que MariaDB también puede ser instalada y configurada en algunas de las versiones de Linux.

Elemento de interfaz push button.

El push button es un componente que funciona dentro de un mecanismo como interruptor, este cuenta con las opciones encendido o apagado.

Como se menciona en la página "Control Electromecánico" (UNAM, 2018), sección "push buttons":

"Estos pueden ser Normalmente Abierto (NO, Normally Open) o Normalmente Cerrado (NC, Normally Closed). Estos dispositivos en su configuración básica necesitan tener una presión mecánica en ellos para mantenerse conectados o desconectados.

También se pueden tener push button de doble acción, en estos, tienen sus contactos NC y NO, y se puede ocupar cualquiera de las dos conexiones. Al ser de doble acción, se tienen ambos contactos y mientras uno está cerrado, el otro par de conexiones está abierto, teniendo así 'dos' en uno."

A continuación, se muestra el símbolo eléctrico del push button, normalmente abierto y normalmente cerrado:

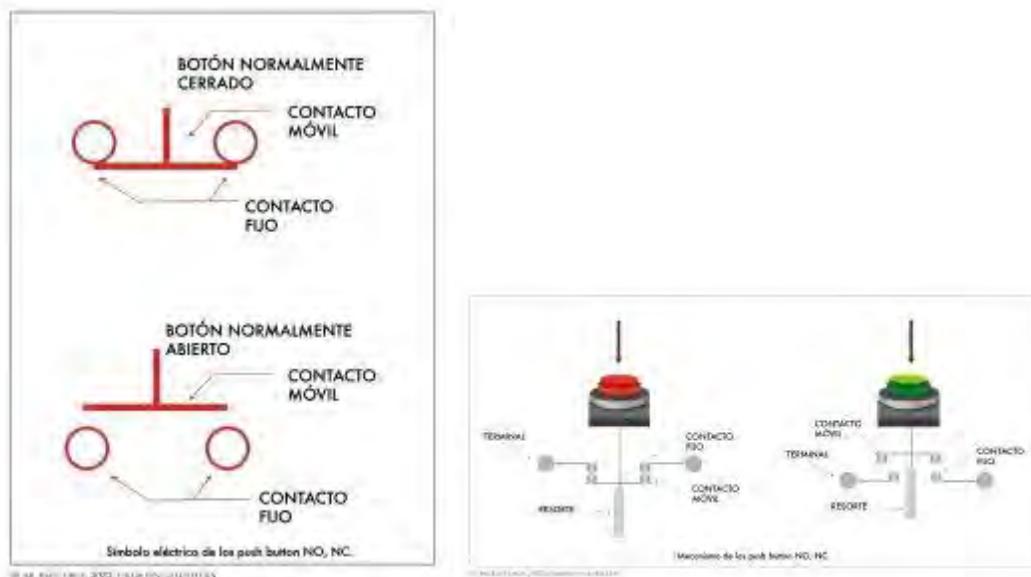


Figura 3, Push button, imagenes obtenidas de google.

Elemento de interfaz display.

Un elemento o módulo display es un componente electrónico utilizado para mostrar información visual, ya sea imágenes, texto o números.

Existen diversos tipos de tecnologías de pantallas que se pueden implementar en una raspberry, por ejemplo, la pantalla LCD táctil XPT2046 Touch Controller de 5 pulgadas, también la pantalla de tinta electrónica como ejemplo la pantalla Pico e-Paper 5.83.

Elemento cámara.

Una cámara es un dispositivo con el cual se puede capturar imágenes estáticas o en movimiento. Adicionar información para qué va a ser utilizada la cámara, tipos de cámaras, tecnología, etc.

3.3.2 Arquitectura del sistema

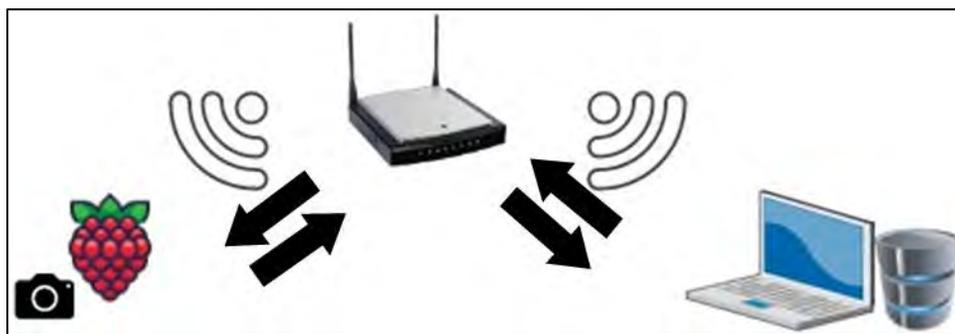


Figura 4. Diagrama de arquitectura del sistema.

La arquitectura del sistema se compone de dos dispositivos conectados a través de la red, los cuales pueden establecer la conexión utilizando el puerto Ethernet o mediante una conexión inalámbrica (WiFi). En este caso, los dispositivos involucrados son una Raspberry Pi y una Laptop, los cuales están conectados por medio de WiFi.

La Laptop actúa como servidor de base de datos, con la instalación y configuración de MariaDB. Por su parte, la Raspberry Pi cuenta con una pantalla integrada, un botón de acción y una cámara incorporada. Al ejecutar el código en la Raspberry Pi, el sistema entra en funcionamiento.

El proceso comienza cuando se presenta un código QR frente a la cámara de la Raspberry Pi. Este código QR previamente ha sido asignado a un número de matrícula. Al detectar y escanear el código QR, se envía la información correspondiente a la base de datos y se solicita la lista de actividades registradas para dicha matrícula.

Las actividades se muestran en la pantalla de la Raspberry Pi. El usuario puede seleccionar una actividad manteniendo presionado el botón de acción. Una vez que se haya realizado la selección, la información se envía y almacenará en la base de datos.

Hardware

A continuación, se describe el hardware que se integró para conformar el dispositivo:

Raspberry Pi 3 model B V1.2 (2015)

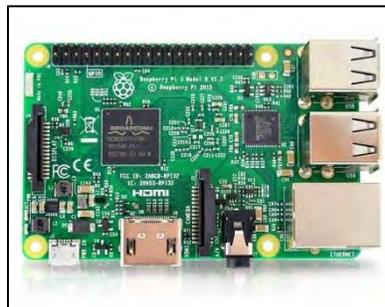


Figura 5. Raspberry Pi Model B V1.2, 2015, imagen obtenida de Google.

El Raspberry Pi Modelo B versión 1.2 es una computadora compacta de bajo costo que cuenta con diferentes puertos, sin embargo para la construcción del sistema, se mencionarán los principales para el funcionamiento de dicho sistema, por lo tanto, se cuenta con 4 puertos USB, en los cuales se puede conectar mouse, teclado, entre otros, también cuenta con un puerto HDMI en el cual se puede conectar un monitor, la placa cuenta con 40 pines GPIO por los cuales se pueden conectar diversos componentes, en este caso se utilizó una pantalla OLEDQT 0.960D y el push button, también cuenta con un chip que controla el módulo de la red WiFi.

Las especificaciones para la Raspberry Pi 3 modelo B son las siguientes (Raspberrypi.com, 2024):

“Procesador de 64 bits Broadcom BCM2837 de cuatro núcleos a 1,2 GHz

- 1 GB de memoria RAM
- LAN inalámbrica BCM43438 y Bluetooth Low Energy (BLE) a bordo
- Ethernet base 100

- GPIO extendido de 40 pines
- 4 puertos USB 2
- Salida estéreo de 4 polos y puerto de video compuesto
- HDMI de tamaño completo
- Puerto de cámara CSI para conectar una cámara Raspberry Pi
- Puerto de pantalla DSI para conectar una pantalla táctil Raspberry Pi
- Puerto Micro SD para cargar su sistema operativo y almacenar datos
- Fuente de alimentación Micro USB conmutada actualizada hasta 2.5A”

Módulo display OLED 0.960D

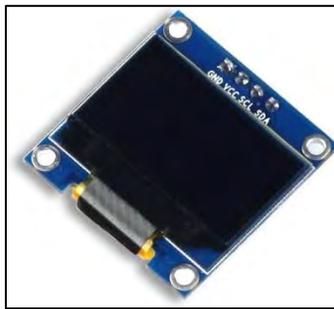


Figura 6. Pantalla OLEDQT 0.960D, imagen obtenida de google.

El módulo display se encarga de mostrar la información que se programa.

Según una publicación en Amazon este dispositivo tiene las siguientes características:

VCC: 3.3~5V

SCL: I2C Serial Clock (Uno: A5; Mega: 21)

SCL: I2C Serial Data (Uno: A4; Mega: 20)

Push Button



Figura 7. Push button, imagen obtenida de google.

Como se menciona con anterioridad el push button es un componente que funciona como interruptor (encendido o apagado), sin embargo, dependiendo de la programación se puede establecer como un “seleccionador” es decir en este sistema específico el push button al mantenerlo presionado por un tiempo realiza una acción determinada.

Cámara (componente reciclado y adaptado a un cable de conexión USB tipo A)



Figura 8. Webcam reciclada

La función de una cámara en generales captar imágenes, en el caso del sistema, la cámara tiene la orden de captar un QR (Quick Response), el cual contiene información del usuario/alumno por ingresar a los espacios de la UQROO.

3.3.3 DIAGRAMA DE CONEXIÓN

En la Raspberry Pi se conectó la cámara, la cual fue reciclada y adaptada se conecta a uno de los puertos USB, también se conectó una pantalla al puerto HDMI y en seguida se conecta la Raspberry Pi a la red inalámbrica.

Para las conexiones de los componentes de display y el push button se ocuparon los pines de la raspberry que corresponden en el caso del display al pin 3 (SDA), 5 (SCL) 4 (DC Power 5V) Y 6 (Ground) de la Raspberry y para el push button se ocuparon los pines 9 (Ground) y 11 (Gen 0), por otro lado, se ocuparon los puertos USB y HDMI para conectar receptor de mouse, teclado inalámbrico y una pantalla respectivamente.

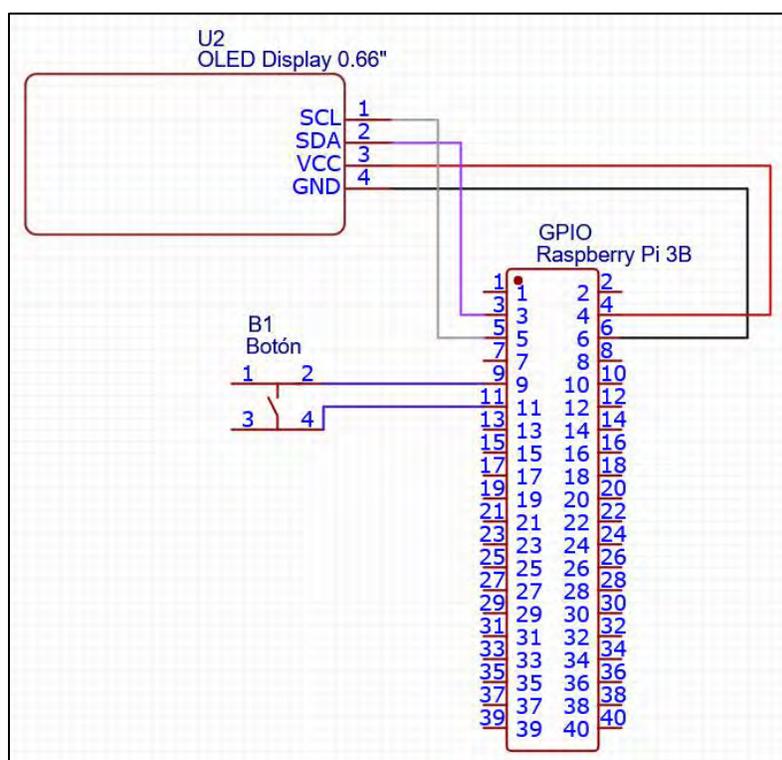


Figura 9. Conexión en pines de la Raspberry.

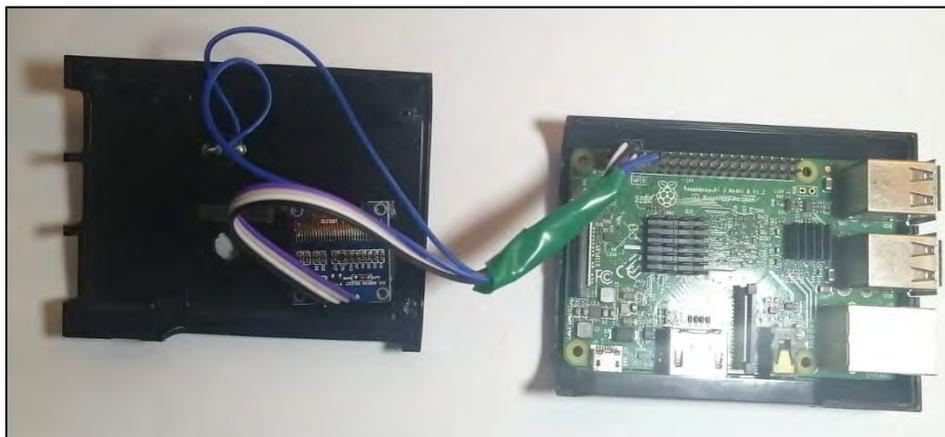


Figura 10. Conexiones en pines GPIO de Raspberry Pi

Como prueba al dispositivo se le asignó el ARE 1 y de este provienen los registros.

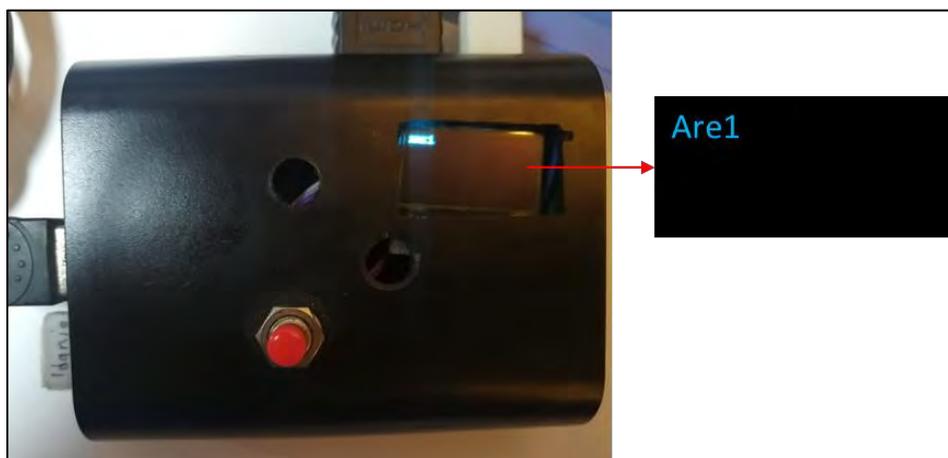


Figura 11. Sistema de administración de acceso de ARE 1.

A continuación, al presentar la QR frente a la cámara el dispositivo muestra la lista de actividades a las que tiene permiso el usuario al que pertenece dicha matricula.

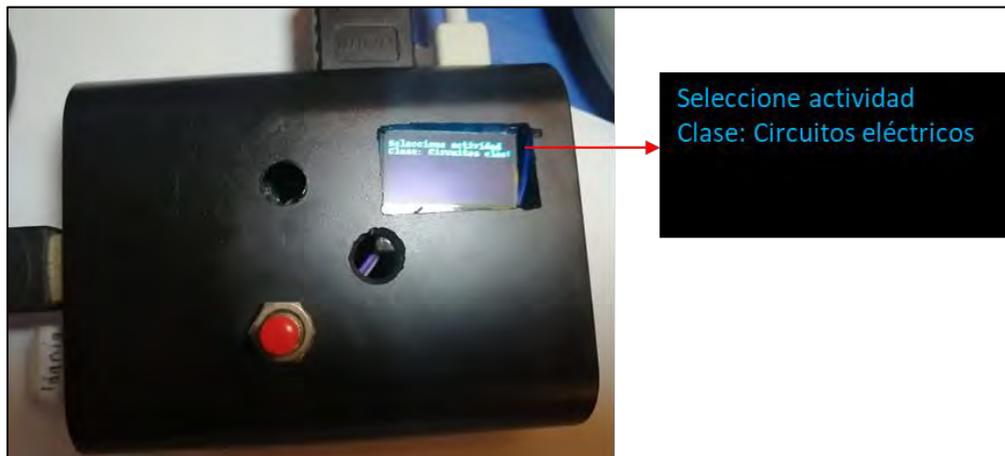


Figura 12. Actividades que tiene permitido el usuario dentro del área.

Al presionar el botón durante 3 segundos se selecciona la actividad y se envía a la base de datos, en cambio si durante 10 segundos no se realiza la selección se determina que el tiempo se encuentra agotado, es decir se declina el ingreso del usuario al área, en ese caso se requiere que el usuario vuelva a presentar el QR frente a la cámara del sistema y seleccionar la actividad para que la información sea registrada en la base de datos.



Figura 13. Tiempo agotado para el registro de acceso.

3.3.4 PRUEBAS

Al realizar el procedimiento de registro, se puede observar información agregada a la base de datos:

Mostrando filas 0 - 6 (total de 7, La consulta tardó 0,0016 segundos.)

SELECT * FROM `registros`

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según

+ Opciones

	ID	PersonalID	LugarID	ActividadID	Fecha
<input type="checkbox"/> Editar Copiar Borrar	1	1	1	4	2023-10-18 15:43:18
<input type="checkbox"/> Editar Copiar Borrar	2	2	5	2	2023-10-18 15:43:18
<input type="checkbox"/> Editar Copiar Borrar	3	3	4	4	2023-10-18 15:43:18
<input type="checkbox"/> Editar Copiar Borrar	4	4	3	1	2023-10-18 15:43:18
<input type="checkbox"/> Editar Copiar Borrar	5	5	5	3	2023-10-18 15:43:18
<input type="checkbox"/> Editar Copiar Borrar	6	4	2	2	2023-10-18 15:43:18
<input type="checkbox"/> Editar Copiar Borrar	7	3	1	2	2023-10-18 15:43:18

Seleccionar todo Para los elementos que están marcados: Editar Copiar Borrar

Figura 14. Tabla de registros en la base de datos.

Registro de entradas

Show: 10 entries Search:

Lugar	Fecha	Actividad	Nombre	Matricula
ARE1	2023-10-18 15:43:18	Asistencia a clase	Persona 1	22-00001
ARE1	2023-10-18 15:43:18	Uso de equipo de laboratorio	Persona 3	22-00003
ARE2	2023-10-18 15:43:18	Uso de equipo de laboratorio	Persona 4	22-00004
ARE3	2023-10-18 15:43:18	Uso de computadoras	Persona 4	22-00004
ARE4	2023-10-18 15:43:18	Asistencia a clase	Persona 3	22-00003
Telemática	2023-10-18 15:43:18	Uso de equipo de laboratorio	Persona 2	22-00002
Telemática	2023-10-18 15:43:18	Asesoría	Persona 5	22-00005
Lugar	Fecha	Actividad	Nombre	Matricula

Showing 1 to 7 of 7 entries Previous 1 Next

Figura 15. Tabla de registros en la página web.

Software

Thonny

Se utilizó el entorno Thonny para programar la aplicación en el lenguaje de Python la parte de la captura de imagen y envió de información a la base de datos (MaríaDB), esta aplicación está incluida de forma predeterminada en Raspberry Pi, el entorno Thonny fue descargado en el siguiente enlace: <https://thonny.org/>.

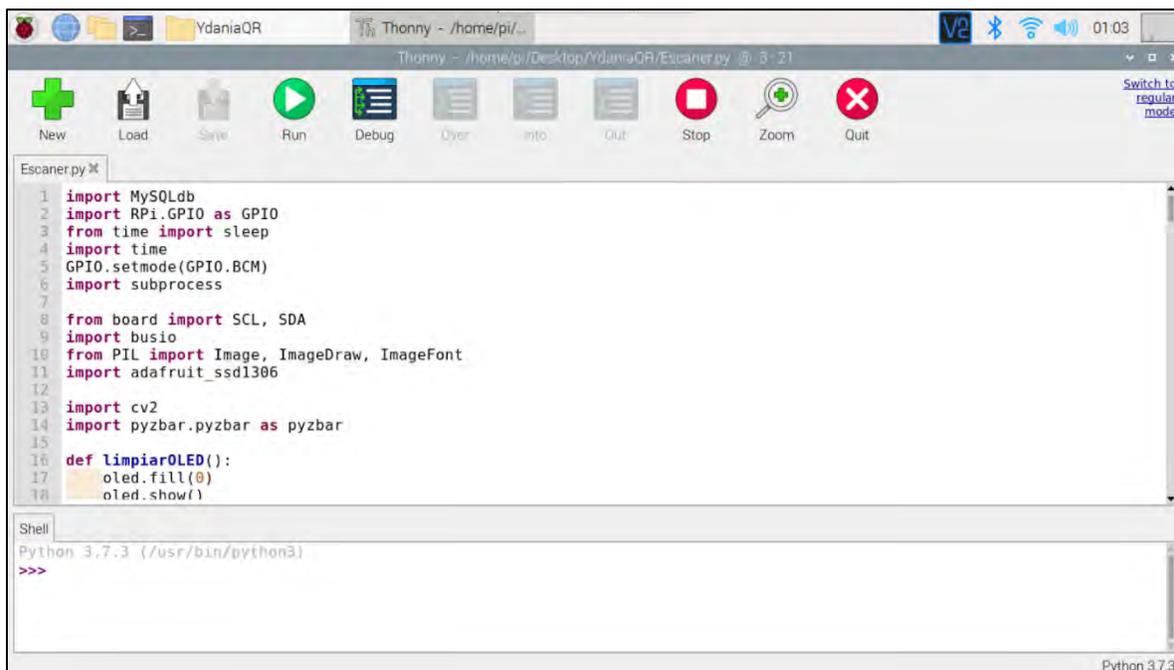


Figura 14. Entorno Thonny

b) La instalación de MariaDB incluye de forma predeterminada HeidiSQL, el cual se utilizó para conectarse al servidor local de MariaDB y en ella se crearon las tablas siguientes:

- Roles: Esta tabla contiene el tipo de función que tiene una persona al ingresar en un área de la institución por lo tanto la tabla contiene las columnas
- Carreras: Esta tabla contiene una lista de las carreras o cursos que se imparten en la institución.
- Estados: Esta tabla contiene la lista de estados en los que se encuentra un alumno, invitado, etcétera.
- Personas: Esta tabla contiene una lista de las matrículas y nombres.

- Rolespersonas: Esta tabla contiene los ID's de las tablas "roles", "personas", "carreras" y "estados" respectivamente.
- Lugares: Esta tabla contiene una lista de las áreas en las que se coloque el dispositivo.
- Actividades: Esta tabla contiene una lista de las matrículas y nombres.
- Permisos: Esta tabla se encuentra relaciona con los ID's de las tablas "personas" y "actividades" que estas pueden realizar.
- Agenda: Esta tabla relaciona los ID's de las tablas "lugares" y "actividades" respectivamente, también se incluye el horario de inicio y fin en el que se realizan las actividades
- Registros: Esta tabla relaciona los ID's de las tablas "personas", "lugares" y "actividades" respectivamente, también se incluye el momento en el que se ingresa a las áreas.

A continuación, se muestra el diagrama de Entidad Relación de las tablas de la base de datos llamada "proyecto":

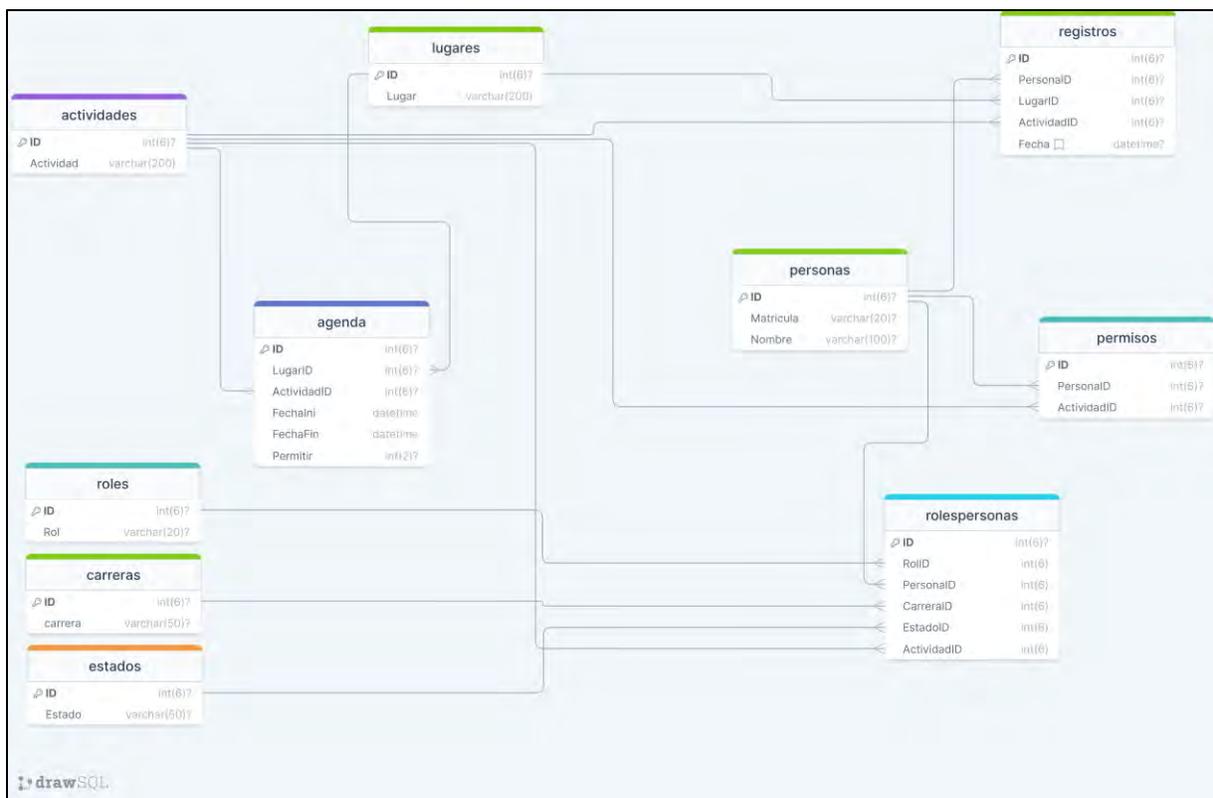


Figura 15. Diagrama E-R de la base de datos.

2.4.2 Casos de uso de la solución

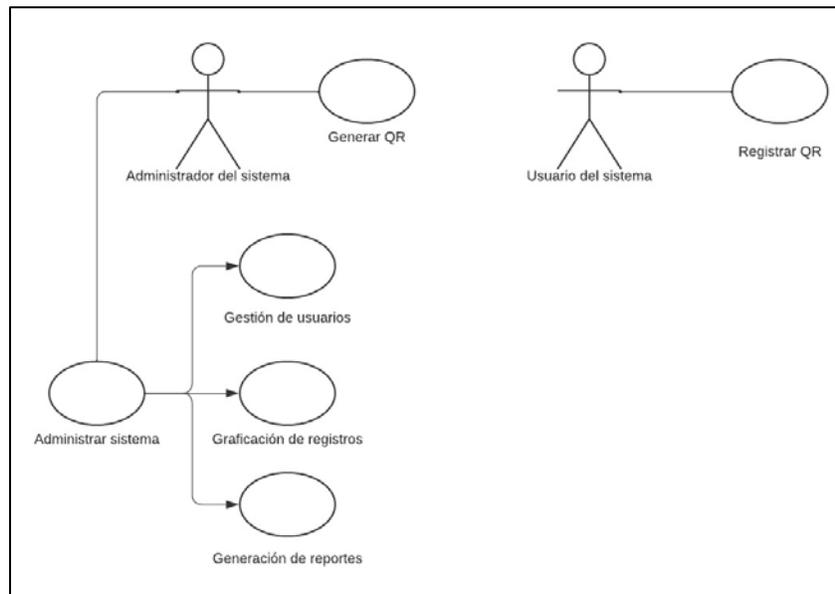


Figura 16. Diagrama UML de funcionamiento de sistema

Funcionamiento:

- Se crea el código QR con la aplicación “Escáner QR” descargada de la aplicación “Play Store” de Android. El formato del número de matrícula debe ser de la siguiente forma XX-YYYYYY, donde X representa los dos últimos dígitos de año de inscripción y la Y representa el número que se asigna al momento de la inscripción.
- El actor presenta el código QR frente a la cámara y éste transmite la información vía inalámbrica a la Raspberry Pi para ser interpretado.
- El dispositivo envía una solicitud a través de la red WIFI al servidor de base de datos para obtener la lista de actividades relacionadas con el número del código QR.
- El servidor de base de datos recibe la solicitud y selecciona en la base de datos las actividades relacionadas con el número del código QR.
- El servidor de base de datos envía la lista de actividades al dispositivo.
- El dispositivo muestra la lista de actividades en pantalla.
- El actor selecciona una de las actividades en el dispositivo.
- El dispositivo envía la información de la actividad seleccionada a la base de datos en el servidor de base de datos.

- i) La base de datos registra la información de la actividad.

RESULTADOS DEL DESARROLLO

HARDWARE

El sistema de administración de acceso a espacios de la Ingeniería en Redes en la DCIT elaborado a partir de los dispositivos y componentes antes mencionados, tiene la capacidad de captar el QR el cual representa la matrícula del usuario, al estar conectado el dispositivo a la misma red, se comunica con el servidor local de la base de datos mientras que la aplicación se ejecuta en la Raspberry Pi.

SOFTWARE

Se descargó e instaló la versión 10.6 de la base de datos MariaDB de 64 bits, dentro de la instalación se incluyó HeidiSQL, éste permite la conexión a la base de datos.

Se utilizó el entorno Thonny para desarrollar en Python la aplicación para la lectura de QR en la Raspberry Pi.

Para la ejecución del sistema de administración de acceso se requiere que se creen las tablas y añadan datos a la base de datos, de igual forma la creación de la aplicación en Raspberry Pi, es importante mencionar que cada vez que se ejecuta esta aplicación es necesario corroborar la información de la base de datos a la que se conectará, por ejemplo el equipo donde se encuentra configurado el servidor local de la base de datos (MariaDB) no cuenta con IP fija por lo cual, si la IP que se encuentra capturada en el sistema no corresponde, en consecuencia no se podrá comunicar el dispositivo con la base de datos y no se podrán realizar los registros de información.

The screenshot shows the HeidiSQL interface with the following components:

- Title Bar:** Laragon.MySQL\proyecto\ - HeidiSQL Portable 12.1.0.6537
- Menu Bar:** Archivo, Editar, Buscar, Consulta, Herramientas, Ir a, Ayuda
- Toolbar:** Standard SQL client navigation and tool icons.
- Database Tree (Left):**
 - Laragon.MySQL
 - information_schema
 - mysql
 - performance_schema
 - proyecto** (336.0 KiB)
 - actividades (16.0 KiB)
 - agenda (48.0 KiB)
 - carreras (16.0 KiB)
 - estados (16.0 KiB)
 - lugares (16.0 KiB)
 - permisos (48.0 KiB)
 - personas (16.0 KiB)
 - registros (64.0 KiB)
 - roles (16.0 KiB)
 - rolespersonas (80.0 KiB)
 - sys

- Host/Database Info:** Host: 127.0.0.1, Base de datos: proyecto
- Query Editor (Bottom):**

```

130 SELECT * FROM `proyecto`.`rolespersonas` LIMIT 1000;
131 SHOW CREATE TABLE `proyecto`.`actividades`;
132 SHOW CREATE TABLE `proyecto`.`lugares`;

```
- Status Bar (Bottom):** r1 : c1 (0 B) | Conectado: MySQL 8.0.30 | Activo durante: 1 días, 0 Hora del ser | Preparado.

Figura 17. Pantalla de información relacionada a la base de datos implementada.

`SELECT * FROM `registros``

Perfilando [Edita]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla | Ordenar según la clave: Ninguna

+ Opciones

	ID	PersonalID	LugarID	ActividadID	Fecha
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	1	1	4	2023-10-18 15:43:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	2	5	2	2023-10-18 15:43:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	3	3	4	4	2023-10-18 15:43:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	4	4	3	1	2023-10-18 15:43:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	5	5	5	3	2023-10-18 15:43:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	6	4	2	2	2023-10-18 15:43:18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	7	3	1	2	2023-10-18 15:43:18

Seleccionar todo | Para los elementos que están marcados: Editar Copiar Borrar Exportar

Figura 18. Datos en la tabla "registros".

Laragon.MySQL\proyecto\registros - HeidiSQL Portable 12.1.0.6537

Archivo Editar Buscar Consulta Herramientas Ir a Ayuda

Filtro de bases de datos: Filtro de tablas

Host: 127.0.0.1 | Base de datos: proyecto | Tabla: registros | Datos | Consulta*

```

1 SELECT * FROM registros, personas, lugares, actividades
2 WHERE registros.PersonalID = personas.ID
3 AND registros.LugarID = lugares.ID
4 AND registros.ActividadID = actividades.ID;

```

Resultado #1 (7r x 12c)

ID	PersonalID	LugarID	ActividadID	Fecha	ID	Matricula	Nombre	ID	Lugar	ID	Actividad
1	1	1	4	2023-09-12 20:03:44	1	22-00001	Persona 1	1	ARE1	4	Asistencia a clase
2	2	5	2	2023-09-12 20:03:44	2	22-00002	Persona 2	5	Telemática	2	Uso de equipo de laboratorio
3	3	4	4	2023-09-12 20:03:44	3	22-00003	Persona 3	4	ARE4	4	Asistencia a clase
7	3	1	8	2023-09-12 20:03:44	3	22-00003	Persona 3	1	ARE1	8	Clase: Python
4	4	3	1	2023-09-12 20:03:44	4	22-00004	Persona 4	3	ARE3	1	Uso de computadoras
6	4	2	8	2023-09-12 20:03:44	4	22-00004	Persona 4	2	ARE2	8	Clase: Python
5	5	5	3	2023-09-12 20:03:44	5	22-00005	Persona 5	5	Telemática	3	Asesoría

Figura 19. Base de datos creada para la aplicación bajo desarrollo.

Se comenta que, actualmente el sistema no cuenta con una interfaz amigable para cualquier usuario, por lo cual el usuario a cargo tendría que conocer, comprender y elaborar consultas específicas para elaborar filtros de los datos y poder graficarlos, sin embargo, se establecen las bases para su desarrollo como trabajo a futuro.

CONCLUSIONES

La implementación del sistema de administración de acceso a espacios de la Ingeniería en Redes en la DCIT, permitirá conocer cuáles son las áreas que son más concurridas por los usuarios, de la Universidad Autónoma del Estado de Quintana Roo. Así mismo, permitirá tener conocimiento de las actividades que se frecuentan en los laboratorios de la DCIT.

Como se menciona anteriormente uno de los principales puntos de la aplicación del sistema es tener conocimiento sobre las visitas, o accesos, realizados a los espacios que conforman los laboratorios de la DCIT y con ello realizar tener estadísticas de uso de las áreas bajo análisis y su operación.

En cuanto a los componentes de un sistema de este tipo, a pesar de ser un equipo compacto y de bajo costo, puede contar con la capacidad de configurarse e implementar o añadir nuevas funcionalidades en un futuro.

Como se menciona anteriormente, la implementación de dicho sistema es de muy fácil configuración, ejecución y manejo, no se necesita de equipos especializados ni requerimientos extra, ya que la Raspberry Pi nos brinda muchas posibilidades para implementar diversas aplicaciones y sistemas. Como en este caso, también se pudo complementar el sistema con la creación del código QR, en este caso se ocupó una aplicación gratuita en el dispositivo móvil para la creación de dicho código, pero existen diversas plataformas y páginas web que nos permiten crear múltiples códigos de forma gratuita. También, es importante mencionar que se requiere tener conocimiento en programación en python y base de datos, además de conocer la estructura de la Raspberry Pi para la conexión de los dispositivos externos.

Como estudiante/profesional involucrado en este proyecto, he encontrado gratificante observar cómo la implementación de este sistema proporciona información valiosa. En resumen, esta experiencia no solo ha contribuido a mi desarrollo académico y profesional, sino que también ha destacado la viabilidad y utilidad práctica de la implementación de sistemas de control de acceso en entornos académicos, como el de la DCIT.

REFERENCIAS

- Becker, A. (10 de mayo de 2023). *uptodown*. Obtenido de <https://heidisql.uptodown.com/windows>
- FileHorse*. (20 de noviembre de 2023). Obtenido de <https://www.filehorse.com/es/descargar-thonny/>
- HeidiSQL. (s.f.). Obtenido de <https://www.heidisql.com/>
- MariaDB Foundation. (2024). *mariadb.org*. Obtenido de <https://mariadb.org>
- Microsoft. (2024). *¿Qué es el control de acceso?* Obtenido de <https://www.microsoft.com/es-mx/security/business/security-101/what-is-access-control>
- Obtenido de nuoplanet: <https://blog.nuoplanet.com/tipos-control-acceso>
- ORACLE. (2024). *OCI*. Obtenido de <https://www.oracle.com/mx/database/what-is-database/>
- Python Software Foundation. (2024). Obtenido de <https://docs.python.org/3/faq/general.html>
- Raspberrypi.cl*. (s.f.). Obtenido de <https://raspberrypi.cl/que-es-raspberry/>
- Raspberrypi.com*. (2024). Obtenido de <https://www.raspberrypi.com/products/raspberry-pi-3-model-b/>
- Tecnipesa*. (24 de marzo de 2021). Obtenido de <https://www.tecnipesa.com/blog/69-tecnologia-rfid-que-ventajas-tiene>
- Termired. (2020). *Empresa Telecomunicaciones Madrid*. Obtenido de <https://termired.com/control-de-acceso-con-tarjeta-como-funciona/>
- Thony. (2018). Obtenido de <https://thonny.org/>
- UNAM. (2018). Obtenido de <https://masam.cuautitlan.unam.mx/dycme/ce/push-buttons/>
- Universidad Nacional Autónoma de México. (2018). Obtenido de <https://masam.cuautitlan.unam.mx/dycme/ce/push-buttons/>
- WBSgo. (12 de abril de 2022). *Avances de la tecnología en el control de acceso*. Obtenido de <https://whitebearsolutions.grupocibernos.com/blog/avances-de-la-tecnologia-en-el-control-de-acceso#:~:text=La%20tecnolog%C3%ADa%20de%20control%20de,las%20instalaciones%20de%20las%20empresas.>

ANEXO A

A continuación, se incluyen las capturas de los pasos para la instalación y configuración de MariaDB 10.6 en Windows 10:



Figura 18. Paso 1

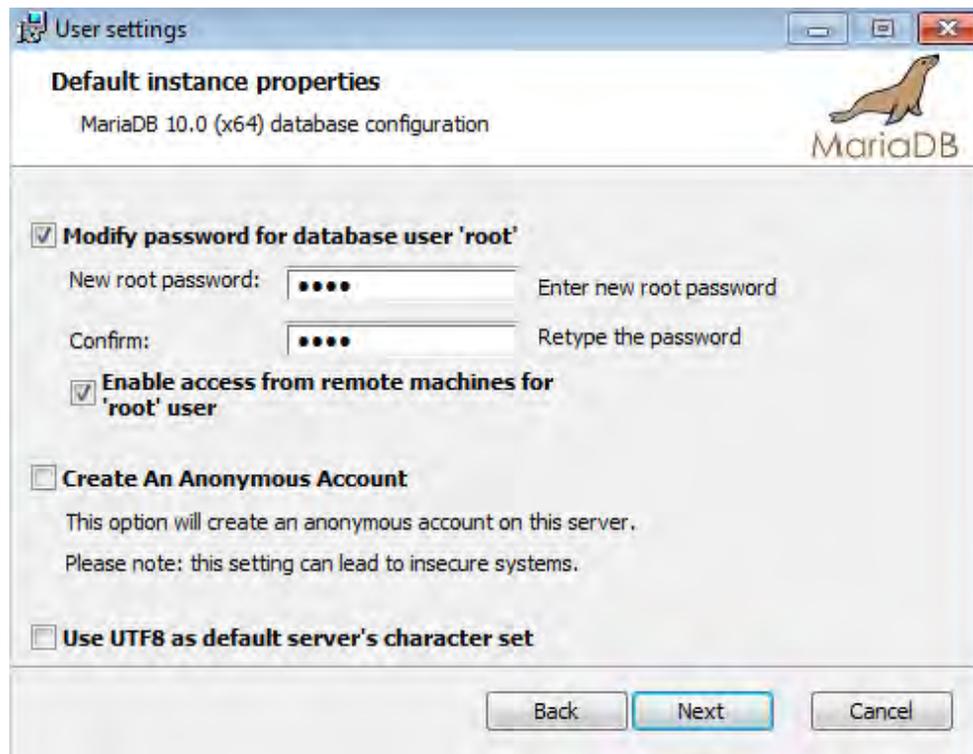


Figura 19. Paso 2

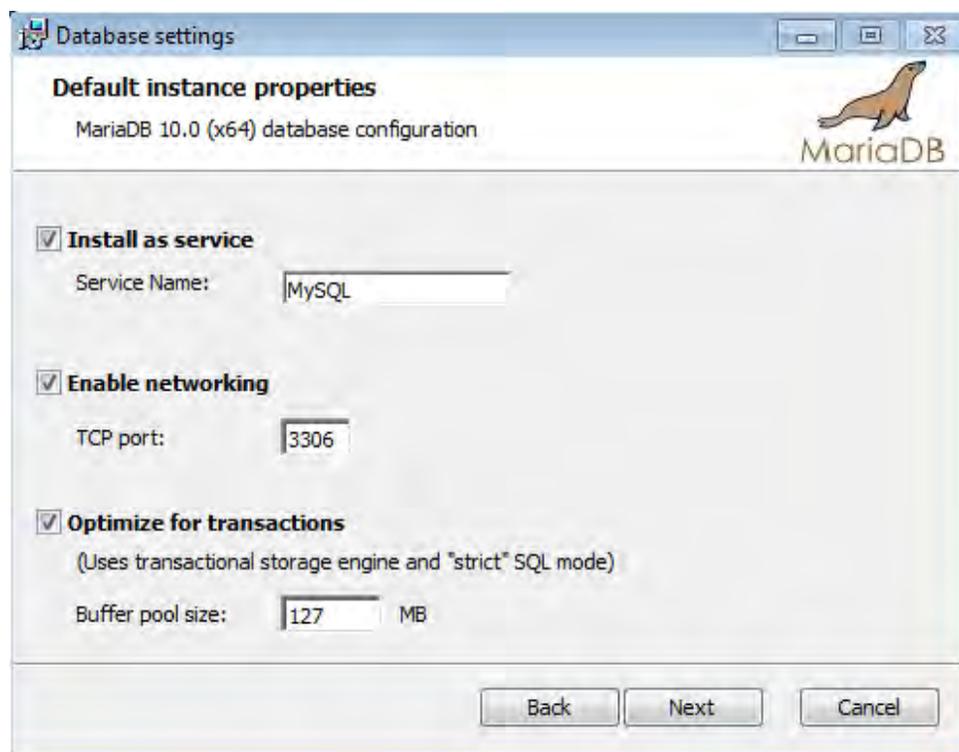


Figura 20. Paso 3

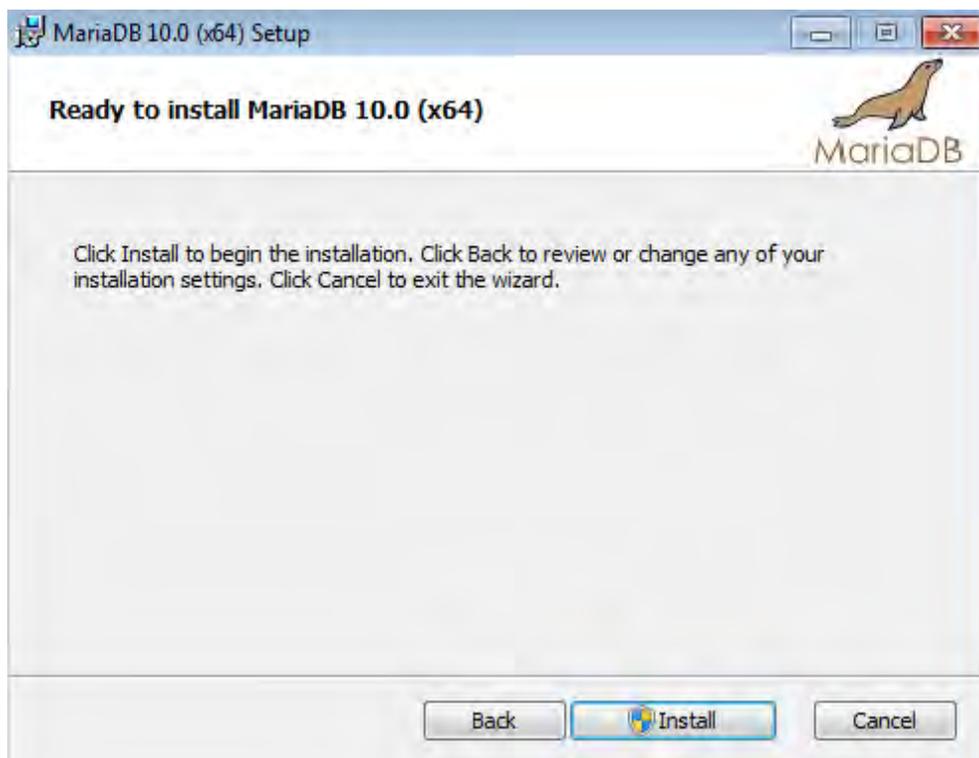


Figura 21. Paso 4

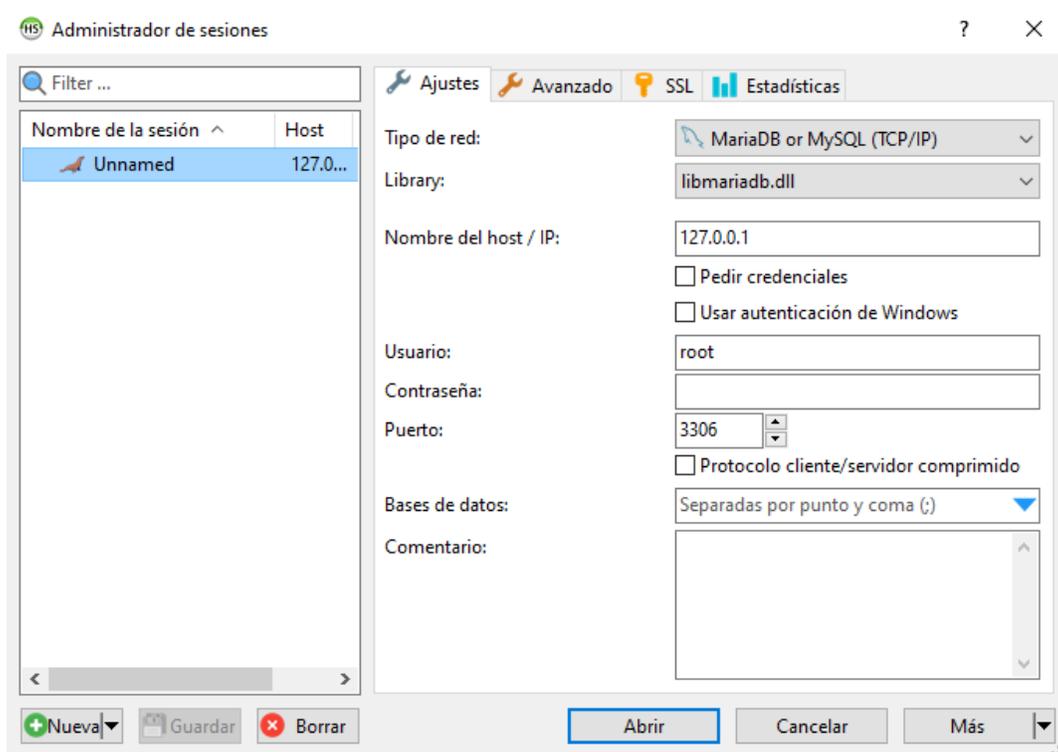


Figura 22. . Paso 5

c) Para la generación de código QR se descargó la aplicación “Escaner QR” de la empresa TeaCapps desde la aplicación Play Store en android.

A continuación, se incluyen las capturas para la creación de QR en el dispositivo móvil con sistema operativo android:

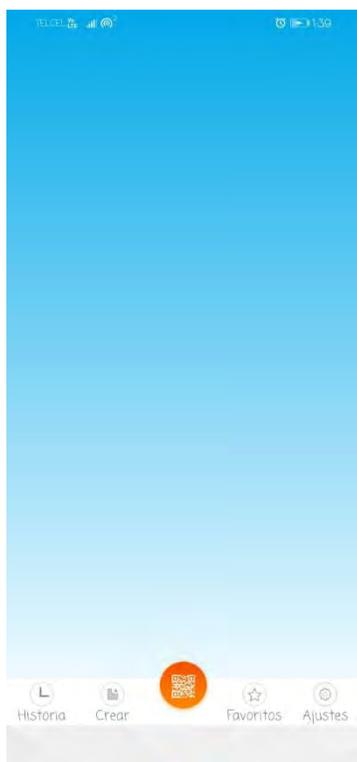


Figura 23. Paso 1, inicio de la aplicación



Figura 24. Paso 2, seleccionar el botón crear, escribir matricula



Figura 25. Paso 3, descargar el código QR

ANEXO B

CREACIÓN DE TABLAS

```

CREATE TABLE roles(
    ID INT(6) PRIMARY KEY AUTO_INCREMENT,
    Rol VARCHAR(20));

CREATE TABLE carreras(
    ID INT(6) PRIMARY KEY AUTO_INCREMENT,
    Carrera VARCHAR(50));

CREATE TABLE estados(
    ID INT(6) PRIMARY KEY AUTO_INCREMENT,
    Estado VARCHAR(50));

CREATE TABLE personas(
    ID INT(6) PRIMARY KEY AUTO_INCREMENT,
    Matricula VARCHAR(20),
    Nombre VARCHAR(100));

CREATE TABLE rolespersonas(
    ID INT(6) PRIMARY KEY AUTO_INCREMENT,
    RolID INT(6) NOT NULL,
    PersonaID INT(6) NOT NULL,
    CarreraID INT(6) NOT NULL,
    EstadoID INT(6) NOT NULL,
    FOREIGN KEY (RolID)
        REFERENCES roles(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT,
    FOREIGN KEY (PersonaID)
        REFERENCES personas(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT),
    FOREIGN KEY (CarreraID)
        REFERENCES carrera(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT),
    FOREIGN KEY (EstadoID)
        REFERENCES estado(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT);

CREATE TABLE actividades(
    ID INT(6) PRIMARY KEY AUTO_INCREMENT,
    Actividad VARCHAR(200) NOT NULL);

```

```

CREATE TABLE lugares(
    ID INT(6) PRIMARY KEY AUTO_INCREMENT,
    Lugar VARCHAR(200) NOT NULL);

CREATE TABLE agenda(
    ID INT(12) PRIMARY KEY AUTO_INCREMENT,
    LugarID INT(6),
    ActividadID INT(6),
    FechaIni DATETIME NOT null,
    FechaFin DATETIME NOT null,
    Permitir INT(2),
    FOREIGN KEY (ActividadID)
        REFERENCES actividades(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT,
    FOREIGN KEY (LugarID)
        REFERENCES lugares(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT);

CREATE TABLE registros(
    ID INT(12) PRIMARY KEY AUTO_INCREMENT,
    PersonaID INT(6),
    LugarID INT(6),
    ActividadID INT(6),
    Fecha DATETIME DEFAULT (CURRENT_TIMESTAMP),
    Id (secuencia) crear una secuencia
    FOREIGN KEY (PersonaID)
        REFERENCES personas(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT,
    FOREIGN KEY (LugarID)
        REFERENCES lugares(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT,
    FOREIGN KEY (ActividadID)
        REFERENCES actividades(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT);

CREATE TABLE permisos(
    ID INT(6) PRIMARY KEY AUTO_INCREMENT,
    PersonaID INT(6),
    ActividadID INT(6),
    FOREIGN KEY (PersonaID)
        REFERENCES personas(ID)
        ON UPDATE RESTRICT ON DELETE RESTRICT,
    FOREIGN KEY (ActividadID)

```

```
REFERENCES actividades(ID)
ON UPDATE RESTRICT ON DELETE RESTRICT);
```

ANEXO C

Creación de aplicación en el lenguaje Python para RaspBerry Pi:

```

import MySQLdb

import RPi.GPIO as GPIO

from time import sleep

import time

GPIO.setmode(GPIO.BCM)

import subprocess

from board import SCL, SDA

import busio

from PIL import Image, ImageDraw, ImageFont

import adafruit_ssd1306

import cv2

import pyzbar.pyzbar as pyzbar

def limpiarOLEED():

    oled.fill(0)

    oled.show()

def mostrarTextoOLEED(texto1 = "", texto2 = "", texto3 = "", texto4 = "", texto5 = ""):

    limpiarOLEED()

    image = Image.new("1", (128, 64))

    draw = ImageDraw.Draw(image)

    font = ImageFont.load_default()

    draw.text((0,0), texto1,font = font, fill = 255)

    draw.text((0,8), texto2,font = font, fill = 255)

    draw.text((0,16), texto3,font = font, fill = 255)

    draw.text((0,24), texto4,font = font, fill = 255)

    draw.text((0,32), texto5,font = font, fill = 255)

    oled.image(image)

    oled.show()

def conectaDB():

    # Settings for database connection

    hostname = '192.168.0.12'

```

```

username = 'root'

password =

database = 'proyecto'

conn = MySQLdb.connect(host = hostname, user = username, passwd = password, db = database)

return conn, conn.cursor()

def desconectaDB(cursor):

    cursor[0].close()

opcion = 0

    seleccion = False

    presionado = True

    tUltimo = time.time()

    while(not seleccion):

        #Solo actualiza la pantalla cuando se ha oprimido el boton

        if presionado:

            mostrarTextoOLED("Seleccione actividad", actividadesAct[opcion])

            #Evalua el tiempo de espera para considerar alumno ausente

            if((time.time() - tUltimo) > tAbortar):

                seleccion = True    #Solo para poder salir

                mostrarTextoOLED("Tiempo agotado")

                print("Tiempo agotado...")

                sleep(5)

#Determina la operacion del boton

    if (not GPIO.input(btn)) and (not presionado):

        tUltimo = now = time.time()

        while(not GPIO.input(btn)):

            #con esto nos bucleamos mientras no se suelta el boton

            nadadenada = 0

            if((time.time() - tUltimo) > tSeleccion):

                seleccion = True

                print("Actividad seleccionada: " + actividadesAct[opcion])

                mostrarTextoOLED("Seleccionado:", actividadesAct[opcion])

                query = "INSERT INTO registros (PersonaID, LugarID, ActividadID) VALUES (" +
str(personID) + "," + str(lugarDispID) + "," + str(actividadesNum[opcion]) + ")"

```

```

        cursor[1].execute(query)

        cursor[0].commit()

presionado = True

        opcion = opcion + 1

        if opcion >= cont:
            opcion = 0

            sleep(.1)

        else:

            if presionado and GPIO.input(btn):

                presionado = False

                sleep(.1)

            else:

                print("No hay actividades asignadas")

                mostrarTextoOLED("No autorizado", "pongase en contacto", "con el profeso X")

except Exception as error:

    print(error)

finally:

    desconectaDB(cursor)

#Obtener los datos del QR (Actualmente sólo interesa la matrícula)

def leerQR(image):

    gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)

    barcodes = pyzbar.decode(gray)

    print('reading...', end='\r')

    for barcode in barcodes:

        barcodeData = str(barcode.data.decode())

        #print(barcodeData)

    return barcodeData

#Configuracion de tiempos en segundos

tSeleccion = 3      #Mantener presionado el boton para considerarlo "Enter"

tAbortar = 10      #Tiempo para considerar que el alumno se ha ido

#Configuracion de boton

btn = 4

GPIO.setup(btn, GPIO.IN, pull_up_down=GPIO.PUD_UP)

```

```

#Configuracion de OLED
i2c = busio.I2C(SCL, SDA)
oled = adafruit_ssd1306.SSD1306_I2C(128,64,i2c)
limpiarOLED()
lugarDispID = 1 #ID que identifica al dispositivo en cuestion
#Configuracion de cámara
camera = cv2.VideoCapture(0)
while(True):
    mostrarTextoOLED("ARE1")

    print("-----")
    esperando = True
    while (esperando):
        ret, frame = camera.read() #Monitorear la cámara
        matAlumno = str(LeerQR(frame))
        if (matAlumno.find("-") > 0): #Si efectivamente se encontró un guion, entonces lo considera una
matricula
            esperando = False
            print(matAlumno)
            listarActividadesPersona(matAlumno) #Muestra las actividades disponibles para la matricula
encontrada
            ret = ""
            frame = ""
    sleep(.5)

```

ANEXO D

Para hacer funcionar a MariaDB con App Server se realizaron las siguientes instrucciones brindadas por ChatGPT:

Instalación de AppServ:

Descarga la última versión de AppServ desde su sitio web oficial (<https://www.appserv.org/>).

Ejecuta el instalador de AppServ y sigue las instrucciones. Durante la instalación, podrás seleccionar los componentes que deseas instalar, como Apache, PHP, phpMyAdmin y MySQL. Asegúrate de seleccionar MySQL.

Finaliza la instalación y asegúrate de que AppServ se inicie automáticamente con Windows.

Instalación de MariaDB:

Descarga la última versión de MariaDB para Windows desde el sitio web oficial de MariaDB (<https://mariadb.org/download/>).

Ejecuta el instalador de MariaDB y sigue las instrucciones. Durante la instalación, puedes configurar la contraseña del usuario root de MariaDB.

Finaliza la instalación de MariaDB.

Configuración para hacerlos funcionar juntos:

Abre la consola de comandos o PowerShell en Windows 11.

Accede a la carpeta donde se encuentra el cliente MySQL de MariaDB. Por defecto, esta carpeta suele estar en la ruta C:\Program Files\MariaDB\MariaDB 10.6\bin. Asegúrate de ajustar la ruta según tu instalación.

Inicia sesión en MariaDB utilizando el cliente MySQL. Puedes hacerlo con el siguiente comando:

```
mysql -u root -p
```

Se te pedirá que ingreses la contraseña del usuario root de MariaDB que configuraste durante la instalación.

Una vez que hayas iniciado sesión en MariaDB, puedes crear bases de datos, tablas y gestionar tus datos.

Para acceder a tus bases de datos a través de un navegador web, puedes utilizar phpMyAdmin, que generalmente viene incluido en el paquete de AppServ. Abre un navegador y visita <http://localhost/phpmyadmin>.