TSE'DE
Revista de Investigación Científica
ISSN 2600 - 5557

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE BOTONES DE PÁNICO PARA AMBIENTES INSTITUCIONALES

DESIGN AND IMPLEMENTATION OF A PANIC BUTTON SYSTEM FOR INSTITUTIONAL ENVIRONMENTS

PROJECTO E IMPLEMENTAÇÃO DE UM SISTEMA DE BOTÃO DE PÂNICO PARA AMBIENTES INSTITUCIONAIS

M.Sc. Dorian Reyes Torres
dorian.reyes@ute.edu.ec
Universidad UTE
Orcid: 0009-0002-5064-1861

Elian Aldaz Suarez
elian.aldaz@ute.edu.ec
Universidad UTE
Orcid:0009-0006-0631-6796

Petter Vargas Naranjo
petter.vargas@ute.edu.ec
Universidad UTE
Orcid:0009-0004-2062-8433

Jeremy Aveiga Loor
jeremy.aveiga@ute.edu.ec
Universidad UTE
Orcid:0009-0008-2741-6838

REVISTA TSE´DE
Instituto Superior Tecnológico Tsa´chila
ISSN: 2600-5557

Resumen

Ante el incremento de incidentes violentos en espacios universitarios ecuatorianos, esta investigación propone el diseño e implementación de un sistema de botones de pánico para oficinas administrativas críticas. Utilizando microcontroladores ESP32 y comunicación Wi-Fi, el sistema permite emitir alertas automáticas mediante Telegram, activando señales visuales en zonas estratégicas. La solución fue instalada en los departamentos de Tesorería y Admisiones, definidos como áreas de alto riesgo según análisis institucional. La arquitectura contempla activadores analógicos, protocolos de seguridad validados por el jefe de seguridad y una integración eficiente con la red inalámbrica existente. El objetivo es reducir los tiempos de respuesta ante emergencias, logrando una latencia promedio de 0.5 segundos. Además, se evalúa el impacto en la percepción de seguridad de los usuarios. Este estudio se enmarca en la necesidad de herramientas tecnológicas accesibles, confiables y de bajo costo que fortalezcan la gestión de riesgos en entornos educativos de alta afluencia.

Palabras clave: Arduino IDE, Botones de pánico, Alto Riesgo, Seguridad, Microcontroladores.

TSE'DE
Revista de Investigación Científica
ISSN 2600 - 5557

Abstract

In response to the rising incidence of violence in Ecuadorian university settings, this study presents the design and implementation of a panic button system for critical administrative offices. The system employs ESP32 microcontrollers and Wi-Fi communication to send automated alerts via Telegram, activating visual signals in strategic locations. Installed in the Treasury and Admissions departments—identified as high-risk zones through institutional risk analysis—the system integrates analog triggers, validated security protocols, and seamless connectivity with the existing wireless network. The primary goal is to reduce emergency response times, achieving an average latency of 0.5 seconds. Additionally, the study evaluates the system's impact on users' perceived safety. This initiative addresses the urgent need for accessible, reliable, and cost-effective technological tools to enhance risk management in high-traffic educational environments.

Keywords: Arduino IDE, Panic Buttons, High Risk, Security, Microcontrollers.

Periocidad Semestral
Vol. 8, núm. 2
revistatsede@tsachila.edu.ec

Recepción: 06-08-2025
Aprobación: 28-08-2025
Publicación: 25-12-2025

URL:
http://tsachila.edu.ec/ojs/index.php/TSEDE/issue/archive

Resumo

Face ao aumento de incidentes violentos em ambientes universitários no Equador, este estudo apresenta o desenho e a implementação de um sistema de botões de pânico para escritórios administrativos críticos. O sistema utiliza microcontroladores ESP32 e comunicação Wi-Fi para enviar alerta automáticos através da plataforma Telegram, ativando sinais visuais em locais estratégicos. A solução foi instalada nos departamentos de Tesouraria e Admissões, identificados como zonas de alto risco com base em análises institucionais. A arquitetura inclui acionadores analógicos, protocolos de segurança validados pelo chefe de segurança e integração eficiente com a rede sem fios existente. O objetivo principal é reduzir os tempos de resposta em situações de emergência, alcançando uma latência média de 0,5 segundos. O estudo também avalia o impacto na perceção de segurança dos utilizadores. Esta proposta responde à necessidade urgente de ferramentas tecnológicas acessíveis, fiáveis e de baixo custo para reforçar a gestão de riscos em ambientes educativos com elevada afluência.

Palavras-chave: Arduino IDE, Botões de pânico, Alto risco, Segurança, Microcontroladores

Introducción

La seguridad en campus universitarios es relevante como respuesta a la creciente vulnerabilidad de espacios con alta afluencia de usuarios. Estudios recientes muestran que la implementación de sistemas de alerta temprana y botones de pánico en entornos académicos mejora significativamente la gestión de incidentes y la percepción de seguridad entre estudiantes, docentes y personal administrativos. En especial considerando el contexto del país de ejecución de la investigación como lo es Ecuador, El cuál posee estadísticas complejas en materia de seguridad expuestos por Olmedo (2023), como:

El periodo desde el 1 de enero hasta el 20 de marzo de 2022, las entidades pertinentes registraron un total de 815 fallecimientos violentos o como en el lapso equivalente de los primeros 78 días del 2023, donde, existe el reporte de 1356 incidentes, lo que representó un aumento del 66,4% (p.1).

Considerando estadísticas de incremento de fatalidades e incidentes dentro del país se considera un método preventivo el incremento de seguridad con el uso de la tecnología, por tal motivo el estudio se centra en el análisis y la propuesta de implementación de un sistema de botón de pánico en el entorno universitario. Funcionando como respuesta a la creciente preocupación por la seguridad en espacios académicos impulsado por la necesidad de desarrollo de herramientas tecnológicas que permitan una respuesta rápida y efectiva ante situaciones de emergencia. En este contexto, el objetivo de la investigación es diseñar e implementar un sistema de botones de pánico basado en microcontroladores y comunicación Wi-Fi que, a través de la plataforma Telegram y activadores analógicos, optimice la

seguridad y los tiempos de respuesta ante emergencias en oficinas administrativas críticas. A raíz del objetivo planteado nacen objetivos específicos como: Analizar los requisitos de seguridad y normativas aplicables a la instalación de dispositivos de alerta manual en entornos administrativos universitarios. Definir la arquitectura de hardware para garantizar la fiabilidad y durabilidad en las oficinas seleccionadas. Desarrollar un firmware en Arduino IDE que capture el evento de pulsación y envíe, en menos de 0.5 s, la notificación cifrada a un Bot de Telegram configurado para el jefe de seguridad e Integrar la solución con la red inalámbrica existente, validando la cobertura y estabilidad de la conexión en todos los puntos de la instalación.

Todo esto nos lleva a realizarnos la siguiente pregunta de investigación. ¿Cómo impacta la implementación de un sistema de botones de pánico basada en microcontroladores y comunicación Wi-Fi con notificaciones automáticas a través de Telegram en los tiempos de respuesta ante emergencias y en la percepción de seguridad de los usuarios en oficinas administrativas críticas?

Los aspectos que se deben enmarcar y se conoce sobre la institución es: se define como una institución privada sin fines de lucro, que funciona bajo el rubro de educación superior y la solicitud del estudio fue definida por la misma institución haciendo hincapié en las oficinas que consideran de alta afluencia como lo son el departamento de tesorería y admisión.

En los últimos años, la seguridad en entornos educativos ha cobrado una gran relevancia, especialmente ante situaciones de riesgo como intrusiones, accidentes o amenazas externas (Castro 2024). Diversos estudios han abordado soluciones

tecnológicas que permitan emitir alertas rápidas y efectivas en situaciones de emergencia.

En el trabajo de Jiménez y Chávez (2023), se implementó un sistema IoT de vigilancia de bajo costo y fácil instalación para residencias, este sistema cuenta con dos módulos Esp32Cam, los cuales están configurados Maestro-Esclavo y utilizan el protocolo de comunicación ESP-NOW para poder recibir y enviar datos de un módulo a otro, como es la información de los sensores de movimiento y magnético, además se integra una alarma sonora al sistema de vigilancia, lo interesante es cuando se detecte movimiento en la residencia se envía un mensaje de alerta junto a imágenes de la misma a través de una API de Telegram.

Por otro lado, la investigación de Romero y Torres (2022) se enfocaron en la seguridad de los vehículos que transportan ganado el sistema busca reducir e incluso frustrar los planes por parte de los perpetradores a través de un botón de pánico en los vehículos de manera que envíen señales de alerta a los puntos de control y monitoreo, además cuenta con sensores en las puertas que al activarse el personal a cargo de la seguridad debe realizar el respectivo procedimiento de seguridad por medio de la cámara de vigilancia.

A nivel internacional, Lozano-Castillo, (2025) desarrolló un prototipo de alarma integral diseñado exclusivamente para mejorar la seguridad en los conjuntos residenciales, éste permite prevenir la intrusión en los apartamentos y activar una alarma silenciosa, también alerta de eventos sísmicos en tiempo real esto se realiza a través de una interconexión mediante microcontroladores (ESP32), y comunicación inalámbrica a través del protocolo LoRa (Módulo LoRa Reyax 998), permitiendo así, un alcance

superior a grandes distancias sin requerir una conexión constante a internet. No obstante, el sistema ofrece un control a través de la aplicación Telegram (plataforma para envió de mensajes), brindando a los residentes del apartamento la posibilidad de auto gestionar su sistema de alarma, recibir notificaciones en tiempo real a través de internet desde cualquier lugar.

Estos antecedentes nos muestran cómo se ha ido integrando los microcontroladores para la seguridad de distintos lugares ya que la situación actual de crisis en materia de seguridad que atraviesa el país es cada vez más crítica (Granda, 2023). Entonces contar con medidas de seguridad ante diversas crisis que se puedan suscitar dentro de espacios públicos o privados se vuelve ya una necesidad (Peral-Belmont, 2013) y el botón de alarma silenciosa es una de esas tantas soluciones eficientes y de bajo costo que se pueden implementar ya que protege y no pone en riesgo la integridad física de las personas involucradas (Tapia, 2023).

La estructuración del sistema está basada mediante la metodología guía para la implementación de Sistemas de Alerta Temprana (Unidad Nacional para la Gestión del Riesgo de Desastres, 2023), adaptada en el entorno educativo superior. Definido en cinco etapas.

1. Identificación de amenazas y usuario clave.
2. Definición de requisitos técnicos y normativos.
3. Selección y montaje de hardware.
4. Desarrollo e integración de firmware en Arduino IDE.
5. Pruebas de campo y ajuste de parámetros de comunicación.

Cada etapa validada con gestión documental y talleres con el jefe de seguridad del campus, cumpliendo con un ciclo iterativo de mejora continua.

Metodología

El pilar fundamental de la investigación y desarrollo de este sistema de botones de pánicos es el diseño de la arquitectura del sistema (Arrescurrenaga, 2023), donde se define los materiales tanto para los múltiples emisores como para el receptor, considerando según la arquitectura del sistema los componentes del emisor como: Un pulsador de cuatro pines NA, microcontrolador ESP32 s2 mini, baquelita, un cargador de 5V y una caja plástica a medida. Mientras que, los componentes del receptor son: Un capacitor electrolítico 1uF, Resistencia de 10k Ω, un transistor 2N3904, borneras de conexión, un diodo Rectificador 1N4001, un microcontrolador ESP32 s2 mini, un cargador de 5V, baquelita y una caja plástica a la medida. Todos los elementos seleccionados por su fiabilidad y bajo costo.

El funcionamiento de un sistema de botón de pánico comienza desde que se presiona el botón, el microcontrolador captura el evento, el cuál envía una señal a los receptores que procesa la información y los distribuye hacia los accionadores previamente configurados (Villamarín, 2023), en este caso, una señal analógica y una señal digital de aprovechamiento de seguridad y flexibilidad de Telegram para ofrecer notificaciones instantáneas y discretas (Bennett, 2024).

Dentro de las consideraciones de integración de los protocolos de seguridad y el canal de notificaciones se debe tomar en cuenta casos hipotéticos como la caída del servidor (Casanova et al., 2021) o la propia seguridad de la red de la institución que realiza una rotación de IP en ciclos de 24 horas, además, enfocar que el bot de Telegram utiliza la

API oficial que envía la información a un canal privado y secreto para asegurar la confidencialidad de las alertas (CCN-CERT, 2023).

Simultáneamente, se adoptó un enfoque mixto para garantizar la seguridad de la señal, considerando que los datos críticos viajan por un canal seguro y único para el sistema, sin interferencia ni tráfico de ningún otro dispositivo (Escuela politécnica Nacional, 2021), ya que se definió una IP única y sin capacidad de rotación solo para el canal de información de la alarma y la información de logs de eventos se almacenan en un servidor local, manejado por la misma institución.

En cuanto al firmware, al estar escrito en Arduino IDE se necesita que esté incorporando en su arquitectura: lectura de interrupción del Hardware, capacidad de selectividad de hardware, enviando marca de tiempo, ubicación y confirmación de recepción (Muñoz-Zuta, 2023). Mientras que, para la fase de pruebas la aplicación metodológica de trabajos previos sobre tótems de alerta, valorando la capacidad de modularidad del código y la facilidad de mantenimiento (Rosales, 2023), es lo propicio.

El sistema de botones de pánico con alerta silenciosa debe complementarse con una distribución clave de los pulsadores de alerta (Vinueza, 2019), siendo clave la aplicación de un análisis de riesgos enfocado en peligros sociales comunes y específicos con cierto grado de posibilidad en el rubro de la institución (Quintero, 2022), información que sirve de ayuda a la arquitectura del hardware, a la implementación y el dimensionamiento de la red que afecta directamente a la latencia (Castro, 2024).

A fin de adecuar el diseño y la implementación del sistema de botones de pánico a las particularidades de cada núcleo administrativo, se desarrolla la Tabla 1, que responde

a un análisis de riesgos y peligros en las principales áreas de la institución. Evaluando secciones de manera general, considerando las amenazas más probables en instituciones educativas y específicas de cada área, vinculadas a atentados, robos de dinero, así como el nivel de severidad en el caso no consentido de materializarse.

Tabla 1

Análisis de riesgos y peligros por áreas de la institución

Área	Riesgos y peligros	Escenario de amenaza	Probabilidad	Severidad	Función del botón de pánico
Admisiones	Robo con intimidación de personal o usuarios (Córdoba-Castañeda, 2021).	Asalto a ventanilla para exigencia de dinero en efectivo	Alta	Alta	Envía alerta silenciosa al puesto de seguridad y activa el protocolo de entrada rápida
Tesorería	Robo organizado de grandes sumas de efectivo; colusión interna (Peñafiel & Ponce, 2023).	Ingreso de varios agresores para despojo de fondos (Suárez-Sánchez, 2021).	Media-Alta	Alta	Envía alerta silenciosa al puesto de seguridad y activa el protocolo de entrada rápida
Laboratorios	Intrusión para robo de equipos caros o insumos peligrosos	Robo de reactivos o equipos durante horarios inactivos	Baja	Media	
Aulas de clases	Hurtos de objetos personales (dinero, dispositivos)	Hurtos oportunistas en descansos o cambios de turno	Media	Media	
Direcciones de facultades	Robo o extorsión para obtener fondos de proyectos o caja chica	Amenaza a administrativos	Baja-Media	Media	
Bodega	Robo de inventario valioso (herramientas, repuestos)	Asalto por personal externo o interno para sustraer mercancía	Media	Media	

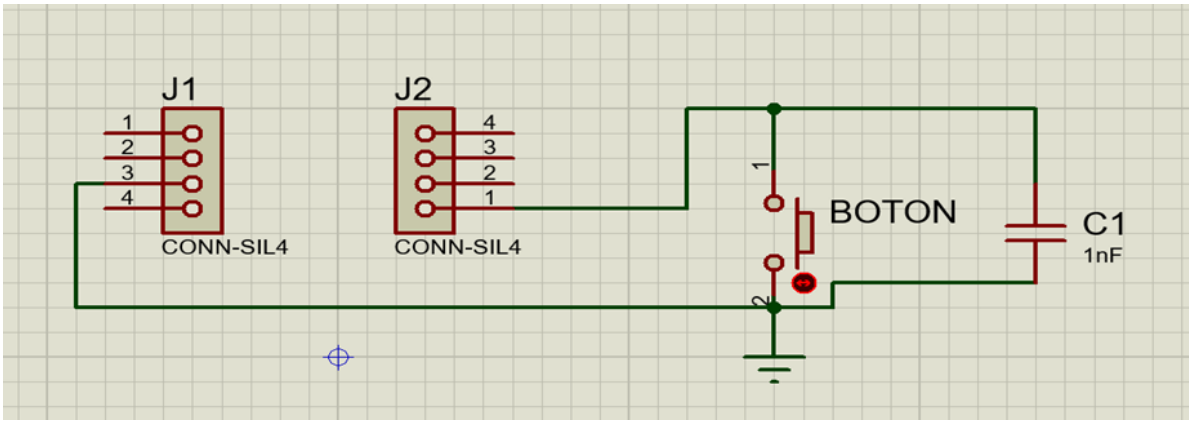
El resultado se presenta en forma de matriz, donde se cruzan la probabilidad de que ocurra el evento y el impacto de cada escenario con la función específica que cumple

el botón de pánico (por ejemplo, alerta silenciosa, activación de sirena interior o bloqueo de accesos). La Tabla 1 permite priorizar los recursos de seguridad, ajustar el protocolo de respuesta y orientar futuras mejoras en el sistema según el grado de criticidad de cada espacio. Sin embargo, para consideraciones iniciales se selecciona los dos departamentos más críticos para la ejecución del sistema (Rodríguez et al., 2021).

En cuanto a la arquitectura del circuito, primero se diseña el circuito en un software de simulación, el seleccionado por su simplicidad de la interfaz y funcionalidad intuitiva es Proteus Professional 8 que es un software eléctrico donde se pueda diseñar circuitos utilizando una variedad de componentes eléctricos que proporciona la librería del software, El diseño del circuito emisor se ve reflejado en la Figura 1, donde, se conecta el pin GPI01 a la fuente de 5 voltios externa y pin GND del esp32 a la tierra común, luego se conecta un pin del pulsador al GPI01 del ESP32 y el otro terminal del pulsador al GND.

Figura 1

Esquema de conexión del emisor



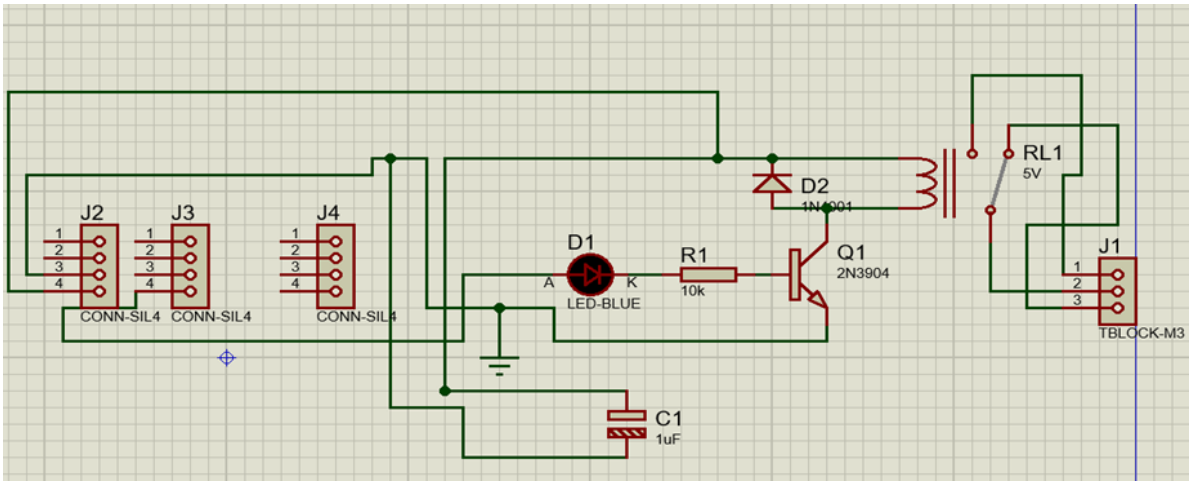
Posteriormente se realiza el diseño del receptor, el cuál posee mayor complejidad en funcionamiento ya que se deben utilizar elementos que filtren las señales de ruido para

que la señal principal llegue sin interrupción, el diseño se ve reflejado en la Figura 2, donde, se ve conectado el cátodo del diodo al pin GPI03 del ESP32 y el ánodo al terminal positivo de la fuente y el capacitor se lo conecto al GPI04 y GND del ESP32, luego se conecta la base del transistor al GPIO3 del ESP32 a través de la resistencia de 10 kilo ohmios, el emisor se conecta al GND y el colector lo conectamos al terminal negativo de la fuente.

Finalmente, al plasmar los diseños del emisor y receptor en Proteus se realiza la simulación, al observar que el programa corre correctamente se puede realizar el ensamble real con validez y confianza en una baquelita.

Figura 2

Esquema de conexión del receptor



El ensamblaje final de cada circuito en la baquelita se puede divisar en la Figura 3, previo a la instalación de los circuitos en los lugares definidos, se realizan pruebas de funcionamiento que cumplan con el lineamiento que al presionar un botón se debe enviar una coordenada “a” y cuando vuelve a ser presionada debe enviar una coordenada “b”, de la misma manera el otro botón debe enviar una coordenada “c” y al presionar nuevamente se envía una coordenada “d”, de manera intercalada. Esto

para que cuando el receptor reciba esas señales se pueda enviar un mensaje diferente cada vez que un botón es presionado. Como parte del diseño del receptor se considera al sistema de almacenamiento de información, conociendo que se realiza mediante Telegram, se dispone de un chat único donde se verifiquen las alertas, creando un Bot de Telegram, a través de Bot Father, el logo definido en la Figura 3, con el nombre de Emergencia Bot.

Figura 3

Bot de Telegram

Los pasos para crear el Bot son sencillos, iniciando con /start continuo de /newbot y se coloca el nombre para el Bot.

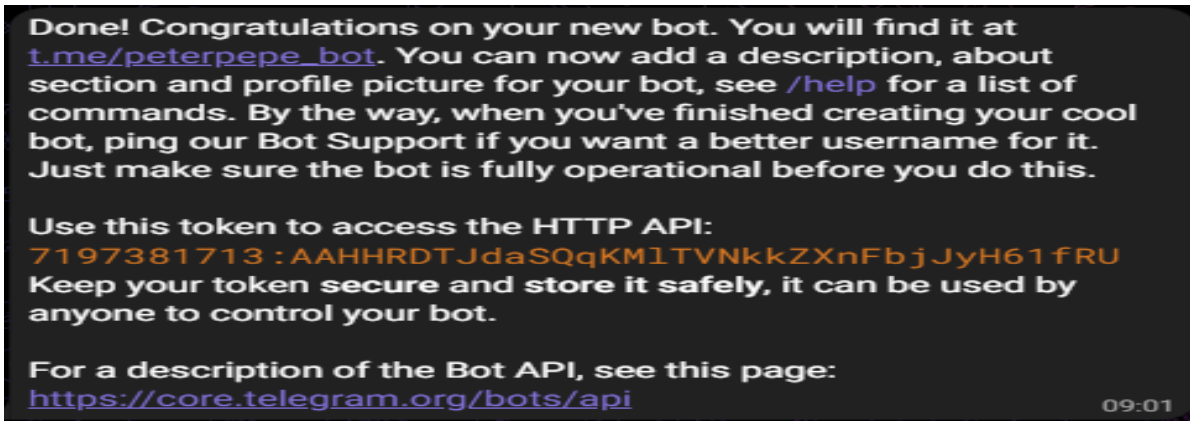
Figura 4

Creación del Bot de Telegram

Con el Bot listo, el servidor BotFather entrega el Token y el link de invitación como se visualiza en la Figura 5, es importante aclarar que el Token no debemos compartirlo con nadie ya que podría ser manipulado por otra persona e incluso eliminarlo de Telegram.

Figura 5

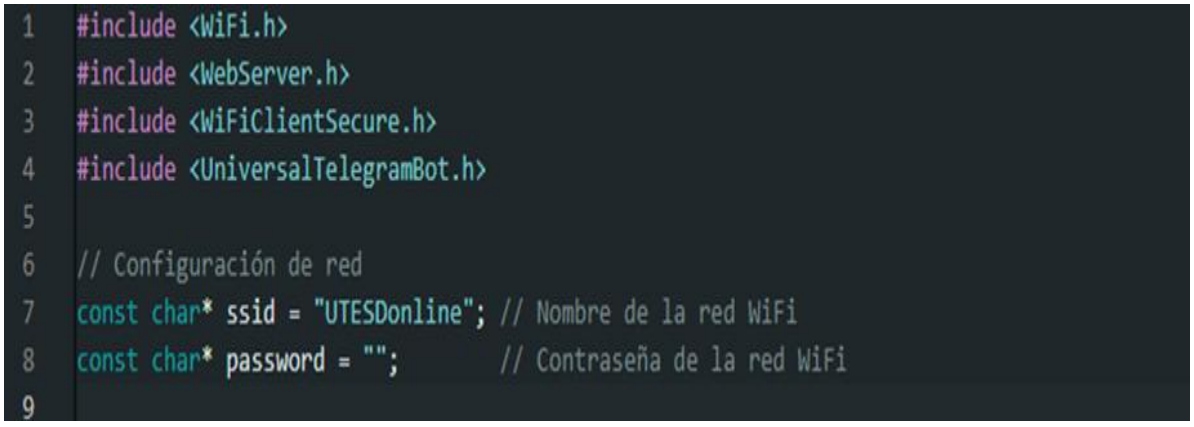
Enlace de invitación de Bot de Telegram



Con el Bot listo se procede a la programación en el receptor a través de la aplicación Arduino IDE, de la línea 1 a la 4 se incluyen las bibliotecas que se usaran, en la línea 7 y 8 se escribe el nombre de la red Wifi a la que queremos conectarnos y también escribimos la clave, en este caso al ser una red abierta se deja el espacio en blanco.

Figura 6

Declaración de las librerías del programa de Arduino



El código se ve reflejado en la Figura 6, sin embargo, se detectó una complicación, ya que la institución cuenta con varios repetidores de WiFi y existía la posibilidad que al momento de apagarse y prenderse el receptor, este podía conectarse a un repetidor diferente al que teníamos pensado cambiando así su IP, en aras de solventar esta complicación situacional, se crea una máscara de seguridad que se conecta al repetidor requerido siempre y cuando se tenga conocimiento de la dirección IP fija.

De la línea 11 a la 15 se crea la máscara de seguridad con la dirección IP planteada en la Figura 7, desea que se conecte continuamente, en la línea 17 se crea un objeto llamado servidor de la clase WebServer. Este objeto se encargará de manejar las peticiones HTTP en un servidor web que se ejecutará en el dispositivo. El número 80 indica el puerto que se utiliza para el servidor web, que por convención es el puerto estándar para HTTP. En la línea 18 se define la constante pinRelay que está conectada al pin 15 del módulo esp32 S2 mini y es de carácter const int ya que no se necesita que cambie de valor mientras se ejecuta el programa.

Figura 7

Código de configuración IP estática

```
10  // Configuración de la IP estática
11  IPAddress local_IP(172, 24, 13, 213);    // Dirección IP deseada
12  IPAddress gateway(172, 24, 13, 1);       // Puerta de enlace (generalmente el router)
13  IPAddress subnet(255, 255, 255, 0);      // Máscara de subred
14  IPAddress primaryDNS(8, 8, 8, 8);        // Servidor DNS primario
15  IPAddress secondaryDNS(8, 8, 4, 4);      // Servidor DNS secundario
16
17  WebServer server(80); // Servidor web en el puerto 80
18  const int pinRelay = 15; // Pin al que está conectado el relé
```

La Figura 8 muestra la línea 21 y 22, donde, en la línea 21 se escribe el token del Bot y en la línea 22 la ID de la cuenta de Telegram que deseamos que reciba las

alertas y almacene la información de las señales, la información dicha se encuentra censurada en la Figura 8, por motivos de seguridad.

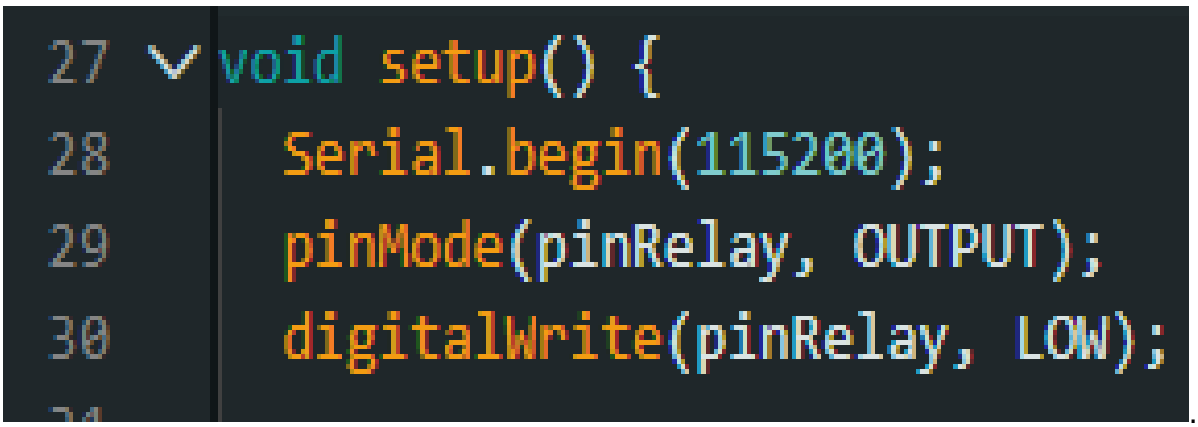
Figura 8

Configuración de seguridad del Bot de Telegram

Se obtiene la ID de la cuenta mediante el Bot “IDBOT” en Telegram, iniciando con /star y se refleja la ID. La Figura 9 muestra los códigos de la configuración de cada uno de los pines de los componentes electrónicos utilizados.

Figura 9

Códigos de la configuración de cada uno de los pines de los componentes electrónicos utilizados

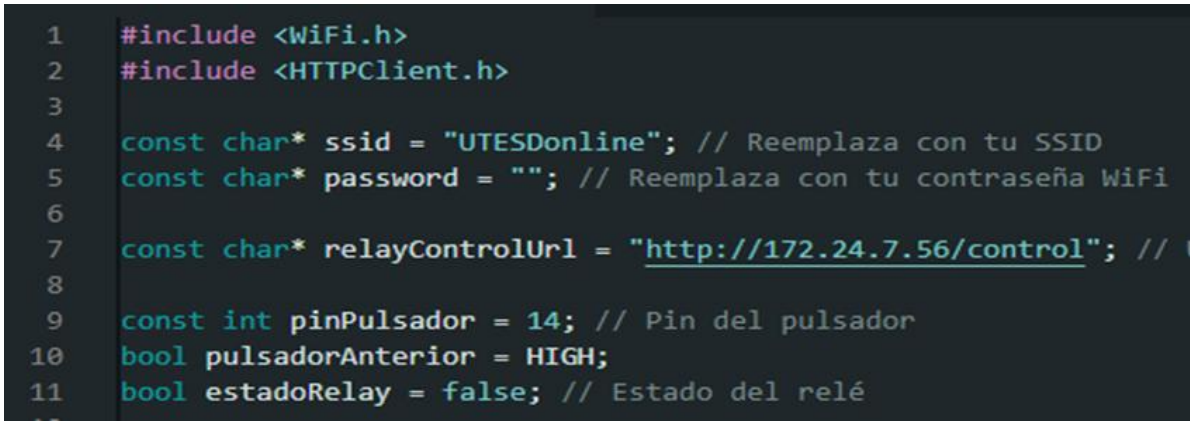


En la línea 27 se crea la función, donde, se configura los pines para iniciar las comunicaciones, la línea 28 “Serial.begin” sirve para iniciar la comunicación entre el microcontrolador y el ordenador, se configura en 115200 baudios ya que es una

velocidad común para una comunicación rápida, en la línea 30, define a la configuración inicial para asegurarnos que el relé este apagado al momento de encender. Los emisores colocados en los departamentos de Admisiones y Tesorería, poseen un código similar al del receptor, iniciando con las bibliotecas, selección de red WiFi y en la línea 7 se dfine la IP que envía el mensaje al emisor (recordando que la IP la brinda el receptor a través del monitor serial) tal como se observa en la Figura 10.

Figura 10

Declaración de las bibliotecas del código del receptor



La Figura 10, visualiza el código y detecta cuando se presiona un botón conectado a un pin digital (pinPulsador). Si detecta un cambio de estado (de no presionado a presionado), invierte el estado de un relé (estadoRelay), enviando el comando “a” para encender o “b” para apagar mediante la función sendCommand. Además, imprime el comando enviado en el monitor serial y espera 500 ms para evitar rebotes. Finalmente, actualiza el estado del botón para futuras detecciones.

La función sendCommand (char comando) como se visualiza en la Figura 11, envía un comando a un servidor a través de una solicitud HTTP GET. Primero, crea la URL combinando relayControlUrl con el parámetro comando.

Figura 11

Códigos de los pines de los pulsadores

```
void loop() {
  bool pulsadorActual = digitalRead(pinPulsador) == LOW; // Detectar pulsación

  // Detecta cambios en el pulsador
  if (pulsadorActual != pulsadorAnterior && pulsadorActual == LOW) {
    estadoRelay = !estadoRelay; // Cambiar el estado
    char comando = estadoRelay ? 'a' : 'b'; // Envía "a" para encender, "b" para apagar
    sendCommand(comando); // Enviar el comando al receptor
    Serial.print("Comando enviado: ");
    Serial.println(comando);
    delay(500); // Evitar rebotes
  }

  pulsadorAnterior = pulsadorActual; // Actualizar el estado anterior
}
```

Luego, inicia la conexión HTTP y realiza la solicitud GET. Si la respuesta es positiva (httpCode > 0), imprime el código HTTP en el monitor serial. Si falla, muestra un mensaje de error. Finalmente, cierra la conexión HTTP, todo el proceso es repetido para el emisor de tesorería con el cambio en las variables que serán “c” y “d” para poder diferenciar coordenadas. información complementada en la Figura 12.

Figura 12

Código para realizar la solicitud del GET

```
void sendCommand(char comando) {
  HTTPClient http;
  String url = String(relayControlUrl) + "?comando=" + comando; // Formatear URL con el comando

  http.begin(url);
  int httpCode = http.GET(); // Realizar la solicitud GET

  if (httpCode > 0) {
    Serial.printf("Respuesta HTTP: %d\n", httpCode); // Mostrar código HTTP en el monitor serie
  } else {
    Serial.println("Error en la solicitud HTTP");
  }
  http.end();
}
```

En la Figura 11 también se ve la aplicación de la función handleRelayControl () que gestiona comandos recibidos por URL para controlar un relé. Si el comando es "a", activa el relé y envía un mensaje indicando que "EMERGENCIA ADMISIONES" está

activado. Si es "b", lo desactiva y notifica la desactivación. Los comandos "c" y "d" realizan la misma acción, pero para "EMERGENCIA TESORERIA". Si el comando es inválido, envía un error HTTP 400. Al finalizar, responde con un mensaje HTTP 200 confirmando que el comando fue procesado. El complemento del circuito con las variables de los dos botones se visualiza en la Figura 13.

Figura 13

Códigos con las variables de los emisores

```
void handleRelayControl() {
  String comando = server.arg("comando"); // Obtener el comando enviado en la URL

  if (comando == "a") {
    digitalWrite(pinRelay, HIGH); // Activar el relé
    sendTelegramMessage("EMERGENCIA ADMISIONES ACTIVADO.");
    Serial.println("Relé activado por Emisor 1.");
  } else if (comando == "b") {
    digitalWrite(pinRelay, LOW); // Desactivar el relé
    sendTelegramMessage("EMERGENCIA ADMISIONES DESACTIVADO.");
    Serial.println("Relé desactivado por Emisor 1.");
  } else if (comando == "c") {
    digitalWrite(pinRelay, HIGH); // Activar el relé
    sendTelegramMessage("EMERGENCIA TESORERIA ACTIVADO.");
    Serial.println("Relé activado por Emisor 2.");
  } else if (comando == "d") {
    digitalWrite(pinRelay, LOW); // Desactivar el relé
    sendTelegramMessage("EMERGENCIA TESORERIA DESACTIVADO.");
    Serial.println("Relé desactivado por Emisor 2.");
  } else {
    Serial.println("Comando inválido recibido.");
    server.send(400, "text/plain", "Comando inválido");
    return;
  }
```

Resultados y Discusión

La implementación del sistema de botones de pánico redujo el tiempo de alerta a emergencias graves en los departamentos de Admisiones y Tesorería a 0.48 segundos, en la Tabla 2, se detallan las métricas de desempeño del sistema, destacando el tiempo de latencia o tiempo de alerta del sistema siendo un valor promedio bastante reducido.

Efectuando 200 pulsaciones distribuidas equitativamente entre las oficinas de Admisión y Tesorería. El 98.7% de las alertas se recibió sin necesidad de reintentos, y únicamente en 2 ocasiones fue necesario reintentar la transmisión.

Tabla 2

Tiempos de respuesta (N=200)

Métrica	Valor promedio	Desviación estándar	N° eventos
Tiempo de alerta (s)	0.48	0.05	200
Tasa de entrega en primer intento (%)	98.7	1.2	200
Eventos de reintento de transmisión	2	-	200

La Tabla 3 indica los datos de la estabilidad de la conexión de Wi-Fi, datos obtenidos en el intervalo de tiempo de octubre 2024 – febrero 2025. Se registraron 7 breves interrupciones de enlace (3.5%), todas recuperadas automáticamente por el protocolo de reintentos integrado en el firmware.

Tabla 3

Porcentaje de estabilidad de la conexión Wi-Fi (N=200)

Indicador	Porcentaje	N° eventos
Conexión estable en el primer intento	96.5	200
Caídas de enlace durante el periodo	7	200
Reconexiones automáticas exitosas	100	7

La encuesta incluyó a 50 usuarios (estudiantes, docentes y personal administrativo). El 92% manifestó sentirse “muy seguro” o “seguro” con el sistema instalado. Y se obtiene un puntaje medio de aceptación alto en todas las preguntas de la encuesta como se puede ver en la Tabla 4.

Tabla 4

Distribución de respuestas por ítem (escala Likert 1-5, N=50)

Item de encuesta	Puntaje medio	Desviación Estandar	N° encuestados
Sensación de seguridad post instalación	4.6	0.4	50
Facilidad de uso del pulsador	4.7	0.3	50
Confianza en la notificación vía Telegram	4.5	0.5	50

Realizando pruebas de falsas alarmas dentro de los registros del Bot de Telegram se analizó que solo se obtuvo un reporte de falsa alarma (0.5%) atribuida a un rebote de pulsador, sin fallos de hardware o software durante el periodo de prueba, información destacada en la Tabla 5.

Tabla 5

Incidencia de fallos y falsas alarmas (N=200)

Tipo de incidencia	N° de ocurrencias	Porcentaje (%)	N° eventos
Falsas alarmas	1	0.5	200
Fallos de hardware o software	0	0.0	200

La eficiencia del sistema desarrollado se evidencia no solo en sus métricas internas, sino también al compararlo con otras soluciones de seguridad implementadas en Ecuador, Latinoamérica y a nivel internacional. En particular, se destacan tres aspectos clave:

1. **Latencia optimizada.** El sistema alcanza una latencia promedio de 0.48 segundos, inferior a la reportada en estudios similares como el de Jiménez & Chavez (2023), cuyo sistema IoT residencial presenta latencias superiores a 1

segundo. Esta mejora se atribuye al uso de interrupciones por hardware y la comunicación directa vía Telegram.

2. **Fiabilidad y recuperación automática:** Con una tasa de entrega en el primer intento del 98.7% y recuperación automática del 100% ante caídas de enlace, el sistema supera estándares de plataformas comerciales como HALO iPanic, que dependen de servidores externos y requieren doble activación para emitir alertas (IPVideo Corporation, 2025).
3. **Costo y escalabilidad:** La arquitectura basada en ESP32-S2 mini, pulsadores analógicos y redes Wi-Fi existentes permite una implementación de bajo costo y alta escalabilidad. A diferencia de sistemas que requieren sensores especializados o suscripciones, esta solución puede replicarse fácilmente en otras áreas institucionales.

La comparación con otros sistemas de botón de pánico en la Tabla 6 da validez tanto frente a sistemas que se encuentran como viables en el mercado como frente a sistemas de desarrollo de investigación con componentes similares.

Otro aspecto a resaltar es el uso de Telegram como canal de notificación ofrece ventajas como cifrado de extremo a extremo, canales privados, y compatibilidad multiplataforma, reforzando la seguridad y accesibilidad del sistema.

Los resultados obtenidos no solo demuestran la efectividad técnica del sistema, sino que también lo posicionan favorablemente frente a otras soluciones de seguridad. La latencia inferior a 0.5 segundos representa una mejora significativa respecto a sistemas IoT similares, mientras que la alta tasa de entrega y recuperación automática refuerzan su fiabilidad operativa., Comparado con alternativas comerciales como

HALO iPanic, el sistema propuesto destaca por su independencia de servidores externos, menor complejidad de activación y facilidad de integración. Estas características, junto con su bajo costo y escalabilidad, lo convierten en una herramienta tecnológica viable y replicable en entornos institucionales de alto riesgo, especialmente en contextos latinoamericanos, donde, la infraestructura y presupuesto suelen ser limitados.

Tabla 6

Comparación con otros sistemas de botón de pánico

Sistema	Latencia	Tecnología	Dependencia de nube	Costo	Aplicación
Sistema en estudio	0.48 s	ESP32 + Telegram	Baja	Bajo	Universidades
HALO iPanic (EE.UU)	Variable (requiere doble pulsación)	IoT+ HALO Cloud	Alta	Alto	Escuelas, Oficinas
Botón tipo tótem UPS (Ecuador)	No especificada	ESP32 + GPS	Media	Media	Universidades
Botón IoT BUAP (México)	Alrededor de un segundo	ESP32 + Web configurable	Media	Bajo	Residencias

El sistema desarrollado cumple con los objetivos planteados, optimizando los tiempos de respuesta ante emergencias y mejorando la percepción de seguridad de los usuarios. La eficiencia del sistema se atribuyó al uso de interrupciones por hardware y la optimización del firmware en Arduino IDE, lo cual permitió superar los estándares reportados en estudios similares. La alta fiabilidad en la transmisión de alertas, junto

con la recuperación automática ante caídas de enlace, coincide con lo planteado por (Casanova et al., 2021) sobre la importancia de protocolos de reintento en sistemas de alerta temprana.

La mejora en la percepción de seguridad se alinea con lo reportado por (Lozano-Castillo, 2025), quien evidenció que la integración de sistemas de alerta silenciosa en residencias elevó la confianza de los usuarios. En el contexto universitario, este impacto emocional positivo refuerza la aceptación institucional del sistema. La arquitectura del sistema, su bajo costo y facilidad de implementación lo posicionan como una herramienta tecnológica viable para entornos educativos de alta afluencia. Aunque se identificó una falsa alarma por rebote de pulsador, se recomienda incorporar filtros de software para mitigar este tipo de eventos. Finalmente, la robustez operativa y la escalabilidad del sistema permiten proyectar su aplicación en otras áreas del campus o en instituciones con características similares, consolidando su valor como solución efectiva para la gestión de riesgos.

Conclusiones

La implementación del sistema de botones de pánico basado en microcontroladores ESP32 y comunicación Wi-Fi, con notificaciones automáticas vía Telegram, ha demostrado ser una solución eficaz para entornos administrativos críticos. Su desempeño técnico, con una latencia promedio de 0.48 segundos y una tasa de entrega del 98.7%, supera los estándares reportados en sistemas similares, lo que evidencia su eficiencia operativa.

Desde una perspectiva social y criminológica, el sistema responde a la creciente necesidad de seguridad en instituciones educativas ecuatorianas, donde los

dispositivos tradicionales suelen ser reactivos y poco integrados. La incorporación de alertas silenciosas y canales cifrados de comunicación fortalece la capacidad de respuesta institucional ante amenazas como robos, extorsiones o intrusiones.

Comparado con otros sistemas de seguridad como HALO iPanic o soluciones tipo tótem, el sistema propuesto destaca por su bajo costo, facilidad de implementación y escalabilidad. Su arquitectura modular permite adaptarlo a diferentes áreas del campus sin requerir infraestructura adicional, lo que lo convierte en una herramienta replicable en contextos similares de Latinoamérica.

La percepción de seguridad entre los usuarios, medida mediante encuesta tipo Likert, alcanzó un puntaje promedio de 4.6/5, reflejando no solo la funcionalidad técnica del sistema, sino también su impacto emocional positivo. Esta aceptación institucional refuerza su viabilidad como modelo de gestión de riesgos.

El sistema desarrollado no solo cumple con los objetivos técnicos planteados, sino que aporta valor científico al integrar tecnología accesible con análisis contextual, ofreciendo una solución preventiva, eficiente y adaptable para mejorar la seguridad en espacios educativos de alta afluencia.

Referencias Bibliográficas

Arrescurrenaga Yanavilca, E. (2023). Botón de pánico y la gestión administrativa en la sección de familia de una unidad policial [Tesis de maestría, Universidad César Vallejo]. Repositorio Institucional UCV. https://repositorio.ucv.edu.pe/bitstream/handle/20.500.12692/151377/Arrescurrenaga_YEM-SD.pdf?sequence=1&isAllowed=y

Bennett, N. (2024). ¿Qué tan seguro es Telegram? Características, limitaciones y controversias. *FreeVPNPlanet Blog*. https://freevpnplanet.com/es/blog/how-safe-is-telegram/

Castro, L. (2024). Trabajo de suficiencia profesional para título profesional [Trabajo de suficiencia profesional, Universidad Tecnológica del Perú]. Repositorio Institucional UTP. https://repositorio.utp.edu.pe/bitstream/handle/20.500.12867/9126/L.Castro_Trabajo_de_Suficiencia_Profesional_Titulo_Profesional_2024.pdf

Castro Hernández, N. C. (2024). Integración de la seguridad privada con la seguridad ciudadana [Artículo científico]. Policía del Estado de Aragua. https://www.researchgate.net/profile/Nohelia-Yaneth-Alfonzo-Villegas/publication/383396668_Interseccion_entre_personas_en_situacion_de_calle_y_seguridad_ciudadana_como_podemos_ayudar/links/66ca82fc97265406eaaa73b8/Interseccion-entre-personas-en-situacion-de-calle-y-seguridad-ciudadana-como-podemos-ayudar.pdf#page=175

Casanova Cruz, D., Miranda Díaz, C. & Yáñez Corvalán, A. M. (2021). Sistema de alerta temprana: Centinela, una experiencia para la retención estudiantil en la Universidad Católica de la Santísima Concepción. *Calidad en la Educación, 55*, 105–119. https://doi.org/10.31619/caledu.n55.1056

CCN-CERT. (2023). Principales riesgos en el uso de Telegram (IA-23/17). Centro Criptológico Nacional. https://www.ccn-cert.cni.es/es/comunicacion-

eventos/comunicados-ccn-cert/5047-principales-riesgos-en-el-uso-de-telegram?format=html

Celis Tapia, T. (2023). Diseño y desarrollo de un sistema de apoyo en situaciones de emergencias sociales [Memoria para optar al título de Ingeniera Civil en Computación, Universidad de Chile]. Universidad de Chile. https://repositorio.uchile.cl/handle/2250/153408

Córdoba Castañeda, M. J. (2021). Diagnóstico del estrés laboral en la Dirección de Admisiones, Registro y Gestión [Trabajo de grado, Universidad Cooperativa de Colombia]. Repositorio Institucional Universidad Cooperativa de Colombia. https://repository.ucc.edu.co/server/api/core/bitstreams/32ce4e95-bea0-4b13-a3e2-693276cf7c2d/content

Escuela Politécnica Nacional. (2021). Política de Uso de la Información, Activos de Información Institucional y Seguridad Informática (Resolución RCP-152-2021). https://www.csirt-epn.edu.ec/normativa

Granda Dávila, P. E. (2023). Importancia del rol preventivo en materia de seguridad ciudadana, como competencia de los gobiernos autónomos descentralizados del Ecuador. *Ciencia Latina: Revista Multidisciplinar*. https://ciencialatina.org/index.php/cienciala/article/view/9046/13491

IPVideo Corporation (2025). Botón de pánico HALO iPanic: seguridad Inalámbrica para oficinas y empresas. HALO Detect. https://halodetect.com/es/capability/panic-button/

Jiménez Villegas, P. B. & Chávez Lalán, L. A. (2023). Sistema IoT de vigilancia a través de la plataforma Telegram [Proyecto integrador, Facultad de Ingeniería en

Electricidad y Computación, ESPOL]. https://www.dspace.espol.edu.ec/handle/123456789/60566

Lozano Castillo, J. (2025). Prototipo de sistema de alarma de seguridad integral para conjuntos residenciales con IoT [Proyecto aplicado, Universidad Nacional Abierta y a Distancia]. https://repository.unad.edu.co/handle/10596/69762

Muñoz Zuta, J. L. (2023). Diseño de un sistema inteligente de monitoreo intradomiciliario y remoto para el cuidado de la población adulta mayor [Tesis, Pontificia Universidad Católica del Perú]. https://tesis.pucp.edu.pe/server/api/core/bitstreams/5dca48a9-b534-4cf7-860c-cd80841c5559/content

Olguín, I. (2018). Desarrollo de un botón de pánico inteligente con tecnología IoT [Trabajo de titulación, Benemérita Universidad Autónoma de Puebla]. Repositorio Institucional BUAP. https://repositorioinstitucional.buap.mx/items/1023f322-92cf-4a62-96a2-5b1a06496426

Olmedo Cueva, L. (2023). Diseño e implementación de un sistema de seguridad estilo botón de pánico tipo tótem y su sistema de gestión, para el campus de la Universidad Politécnica Salesiana sede Cuenca [Tesis de grado, Universidad Politécnica Salesiana]. https://dspace.ups.edu.ec/handle/123456789/26510

Peral Belmont, F. (2013). Utilización de los recursos de la seguridad pública con fines privados: sus efectos en la seguridad de los ciudadanos y el bien común [Tesis de maestría, Universidad Nacional de La Plata]. SEDICI. https://sedici.unlp.edu.ar/handle/10915/34193

Quintero Ávila, O. (2022). El análisis y mapeo delictivo como apoyo para el diseño de una política pública de seguridad, y de la prevención social de violencia y la delincuencia en la colonia Colinas de San Bernabé (Fomerrey 25) en Monterrey, Nuevo León [Tesis doctoral, Universidad Autónoma de Nuevo León]. Repositorio Académico Digital UANL. http://eprints.uanl.mx/23380/1/1080080851b.pdf

Rodríguez, J., Parra, C., Solís, D., López, M., López, M. & Parra, J. (2021). Técnica de jerarquización de activos MCCR: Matriz de criticidad cualitativa de riesgo [Artículo científico]. Tecnológico de Costa Rica. https://www.researchgate.net/profile/Carlos-Parra-19/publication/351334679_Tecnica_de_Jerarquizacion_de_Activos_MCCR_Matriz_de_Criticidad_Cualitativa_de_Riesgo_Caso_de_estudio_Equipos_de_produccion_de_lentes_Oftalmicas_del_laboratorio_de_la_empresa_PRATS_Costa_Rica_Industria/links/6091aed5a6fdccaebd091fdc/Tecnica-de-Jerarquizacion-de-Activos-MCCR-Matriz-de-Criticidad-Cualitativa-de-Riesgo-Caso-de-estudio-Equipos-de-produccion-de-lentes-Oftalmicas-del-laboratorio-de-la-empresa-PRATS-Costa-Rica-Industria.pdf

Romero Mosquera, R. A., & Torres Ramos, J. F. (2022). Desarrollo e implementación de un sistema de seguridad para camiones transportistas basado en un botón de pánico y alarma usando un módulo GPS [Tesis de grado, Escuela Superior Politécnica del Litoral]. https://www.dspace.espol.edu.ec/bitstream/123456789/57108/1/T-113042%20ROMERO%20-%20TORRES.pdf

Rosales, D. (2023). Tecnologías emergentes en botones de pánico [Trabajo de titulación, Instituto Tecnológico Superior Sudamericano]. https://editorial.risei.org/index.php/risei/catalog/download/botonesdepanico/63/1243?inline=1

Suárez Sánchez, J. (2023). El control interno y su influencia en la gestión del área de tesorería de la Unidad de Gestión Educativa Local Corongo, 2018 [Tesis de pregrado, Universidad Católica Los Ángeles de Chimbote]. Repositorio ULadech. https://repositorio.uladech.edu.pe/bitstream/handle/20.500.13032/25132/CONTROL_INTERNO_TESORERIA_Y_UGEL_SANCHEZ_PAUCAR_ELMER_EMERSON.pdf?sequence=1&isAllowed=y

Unidad Nacional para la Gestión del Riesgo de Desastres. (2023). Protocolo Nacional de Respuesta ante Incendios Forestales. Gobierno de Colombia. https://portal.gestiondelriesgo.gov.co/Documents/Protocolo-Nacional-de-Respuesta-ante-Incendios-Forestales.pdf

Villamarín, G. (2023). Botones de pánico: innovación en la protección ciudadana [Trabajo de titulación, Risei]. https://editorial.risei.org/index.php/risei/catalog/view/botonesdepanico/62/1241

Vinueza, D. (2019). Implementación de un prototipo de sistema domótico basado en la plataforma Arduino gestionado a través del Internet [Trabajo de titulación, Escuela Politécnica Nacional]. https://bibdigital.epn.edu.ec/bitstream/15000/19952/1/CD-9412.pdf