FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE PÁNICO PARA ALERTA DE ROBOS EN LOS ALREDEDORES DE LAS SEDES DE LA UNIVERSIDAD DE LAS AMÉRICAS

Autores

Miguel Ángel Baquero Tello
Marcos Kevin Gavela Moreno

Año
2019

FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE PÁNICO PARA ALERTA DE ROBOS EN LOS ALREDEDORES DE LAS SEDES DE LA UNIVERSIDAD DE LAS AMÉRICAS

Trabajo de Titulación presentado en conformidad con los requisitos establecidos para optar por el título de Ingenieros en Redes y Telecomunicaciones

Profesor guía

Mg. Iván Ricardo Sánchez Salazar

Autores

Miguel Ángel Baquero Tello

Marcos Kevin Gavela Moreno

Año

2019

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo, Diseño e implementación de un sistema de pánico para alerta de robos en los alrededores de las sedes de la Universidad De Las Américas, a través de reuniones periódicas con los estudiantes Miguel Ángel Baquero Tello y Marcos Kevin Gavela Moreno, en el semestre 201910, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.”

_____

Iván Ricardo Sánchez Salazar

Magíster en Calidad, Seguridad y Ambiente

C.C.1803456142

DECLARACIÓN DEL PROFESOR CORRECTOR

“Declaro haber revisado este trabajo, Diseño e implementación de un sistema de pánico para alerta de robos en los alrededores de las sedes de la Universidad De Las Américas, de los estudiantes Miguel Ángel Baquero Tello y Marcos Kevin Gavela Moreno, en el semestre 201910, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación”.

_____

Nathaly Verónica Orozco Garzón.

Doctora en Ingeniería Eléctrica en el área de Telecomunicaciones y Telemática

C.C. 1720938586

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaramos que este trabajo es original, de nuestra autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

Miguel Ángel Baquero Tello	Marcos Kevin Gavela Moreno
C.C. 1718164765	C.C. 1721992806

AGRADECIMIENTOS

Agradecemos a nuestras familias y amigos, quienes fueron un pilar primordial para alcanzar nuestros objetivos académicos y profesionales, al personal del Servicio Integrado de Seguridad ECU 911 por habernos facilitado las herramientas para la culminación de nuestra meta y a todos los docentes que forman parte de la Universidad De Las Américas por transmitir todos sus conocimientos a las nuevas generaciones de personas innovadoras.

DEDICATORIA

Dedico este proyecto a mis familiares, quienes me han ofrecido su soporte absoluto en los momentos más difíciles y para superar las dificultades a lo largo de todo el trayecto. Sin ellos no hubiese sido posible alcanzar este logro.

Miguel Ángel Baquero

DEDICATORIA

Para mis padres: Marcos y Martha, por acompañarme durante todas las etapas de mi vida, con amor y paciencia, enseñándome el camino hacia la superación. A mi hermano Esteban por su compresión y apoyo. A mis amigos por permitirme aprender más de la vida juntos. Esto sólo fue posible gracias a ustedes.

Marcos Kevin Gavela

RESUMEN

En el presente documento se detalla el diseño e implementación de un prototipo de alarma de pánico a través de una aplicación móvil para teléfonos inteligentes, capaz de notificar de incidentes de robos al Servicio Integrado de Seguridad ECU 911 y a un servidor de la Universidad De Las Américas. Dicha alarma de pánico la activa el usuario a través de una combinación de botones físicos del teléfono inteligente, en ese instante se envía la información del usuario al ECU911 y al servidor implementado para la UDLA, el cual activará un clúster de cámaras para videovigilancia y enviará una notificación a la persona encargada de la UDLA quien podrá observar a través de un enlace lo que está sucediendo en el lugar de los hechos y poder de alguna manera socorrer a la víctima.

Palabras clave: Android Studio, Aplicaciones móviles, Java, NetBeans, *Sockets*, *Threads*, Videovigilancia.

ABSTRACT

The present document will detail the design and implementation of a prototype of a panic alarm through a mobile app for smartphones, capable of reporting incidents of theft to *Servicio Integrado de Seguridad ECU 911* and a server of UDLA. This panic alarm is activated by the user through a combination of physical buttons on the smartphone, in that moment the user information is sent to ECU911 and to the server implemented for UDLA, which will activate a cluster of video surveillance cameras and sends a notification to the person in charge of UDLA, who will watch the activity of the incidence place through a URL and somehow help the victim.

Keywords: Android Studio, Java, Mobile apps, NetBeans, Sockets, Threads, Video surveillance.

ÍNDICE

ÍNDICE DE FIGURAS

1. Capítulo I. Introducción.

1.1 Antecedentes

En un censo estadístico publicado en el año 2013 se pudo evidenciar que, entre los años 2011 y 2013 los asaltos denunciados en la ciudad tuvieron un incremento entre el 49% y el 53% en el sector Norte con respecto al resto del cantón Quito. También podemos verificar que el 58% de estos son en la calle o vía pública (Observatorio Municipal de Seguridad Ciudadana, 2013, pág. 34).

Según estadísticas consultadas, la tasa de variación de robos a personas se incrementó 1.18% en el primer semestre del año 2018 (2648 casos) con respecto al 2017 (2617 casos). El mayor número de incidencias actuales de robos se da entre las 18:00 y 23:59 (955 casos en lo que va del año 2018) (Ministerio del Interior, 2018).

El día 14 de diciembre de 2017, Teleamazonas difundió una noticia a través de su página web, informando acerca de un robo en el sector de la Universidad de las Américas sede Granados (Teleamazonas, 2017). Este no ha sido un hecho aislado, también existen varias denuncias que han sido reportadas por los estudiantes siendo víctimas de atracos, violencia u otros hechos en los alrededores de los campus Granados, Queri y UDLAPARK.

Existen aplicaciones móviles enfocadas a la seguridad que rastrean la ubicación del usuario o funcionan con un botón que puede ser accionado después de haber encendido la pantalla del dispositivo, pero ninguna de ellas cuenta con una alarma de pánico que se pueda activar de una manera fácil y desapercibida, integrando un sistema de videovigilancia.

Por estos motivos, promover el bienestar personal de los estudiantes de la UDLA se propone la opción de desarrollar una aplicación móvil que permita alertar a las autoridades cuando la seguridad de un individuo se encuentre amenazada.

1.2 Alcance

Se desarrollará una aplicación en iOS y Android, la cual permitirá a los usuarios por medio de una combinación de botones físicos de un teléfono móvil inteligente generar una alerta que será enviada a los servicios de seguridad de la universidad y al Servicio Integrado de Seguridad ECU911. A su vez, se efectuará un prototipo de un clúster de videovigilancia que será activado en respuesta a la alerta creada de la aplicación móvil, dicho clúster facilitará el *streaming* de video en tiempo real a través de una red de cámaras.

El clúster antes mencionado será controlado mediante un servidor que contenga una aplicación y/o un microcontrolador, el cual se encargará de administrar las cámaras de video para *streaming* en tiempo real y esta pueda ser visualizada por los servicios de emergencia y ECU911.

1.3 Justificación

En nuestra realidad nacional, la delincuencia se encuentra en aumento y varias personas, diariamente, son víctimas de robos. Ocasionalmente, la gente que se enfrenta a la delincuencia resulta gravemente herida o pierde su vida.

Dada la necesidad de velar por el bienestar de los estudiantes y funcionarios de la Universidad, como medio de alerta y notificaciones, se propone la elaboración de un sistema tecnológico por el cual, cualquier usuario de teléfono móvil inteligente que tenga instalada la aplicación en su teléfono pueda alertar a las respectivas autoridades (ECU911) y/o encargados de la UDLA en caso de una ocurrencia que amenace a la integridad de las víctimas.

El proyecto garantiza que se pueda recopilar información (video) como evidencia de las ocurrencias y así identificar de una manera óptima a los atacantes. También, existe una mayor probabilidad de detener a los maleantes *in fraganti* al notificar a las autoridades competentes.

1.4 Objetivo General

Diseñar e implementar un prototipo de sistema de pánico para alerta de robos en los alrededores de las sedes de la Universidad de las Américas.

1.4.1 Objetivos Específicos

- Analizar el entorno de la universidad para determinar estratégicamente los sitios en los cuales se ubicará el sistema de pánico.
- Investigar las tecnologías existentes en ECU911 y servicio de seguridad de la Universidad para determinar los elementos útiles para la integración del sistema.
- Diseñar una aplicación móvil en iOS y Android que tenga la capacidad de enviar una notificación de alerta a un servidor. El servidor controla el sistema de pánico.
- Implementar un prototipo del sistema de pánico para alertar de robos considerando el esquema presentado en el ítem anterior.
- Realizar pruebas de integración de la aplicación, servidor, sistema de y cámaras.
- Analizar los costos de implementación del sistema de pánico en las sedes de la Universidad.

2. Capítulo II. Análisis de ubicación de sistema de pánico e investigación de tecnologías

2.1 Encuesta realizada a los estudiantes de la UDLA

Como una fuente de información adicional, fueron encuestados cuarenta y seis estudiantes de los diferentes Campus de la Universidad De Las Américas, el cual se detalla en el anexo 5.

Según la información recopilada dentro de la encuesta, se pueden resumir las siguientes afirmaciones:

- En la encuesta realizada, a los 4 campus existentes, el 58,7% de los estudiantes pertenecen al campus Queri, seguido del Campus Colón con el 23,9% de los estudiantes y finalmente, los Campus Granados y UDLAPARK; ambos, con el 8,7% de estudiantes.

- 9 estudiantes encuestados (19,6%) han sufrido eventos relacionados con robos, asaltos o agresiones en los alrededores de los Campus de la UDLA, siendo este un porcentaje significativo.
- La dirección que registra más eventos de incidentes es la Av. 6 de Diciembre y Colón (5 estudiantes), seguida de la calle Joel Polanco y calle De Los Colimes (2 estudiantes).
- La aplicación móvil de la alarma de pánico tiene una aceptación del 87%, a 40 estudiantes de los 46 encuestados les gustaría que la aplicación móvil sea desarrollada y desplegada.

2.2 Ubicación estratégica del sistema

Para instalar el sistema de pánico se puede hacer referencia a estadísticas de las incidencias de sitios geográficos de robos a personas o identificar los lugares a los redores de la universidad por los cuales los estudiantes transitan rutinariamente para retornar a sus hogares.

Se solicitó información estadística al ECU911 para reconocer los lugares más críticos donde se han registrado eventos de robos. Dicha entidad pública facilitó un informe semanal correspondiente al rango de fechas del 8 de octubre de 2018 al 14 de octubre de 2018 donde se pueden observar las incidencias en los sectores “Jipijapa” e “Iñaquito”, sectores correspondientes a los Campus Granados y Campus Queri de la Universidad De Las Américas.

En el anexo 1 se especifica que el robo a personas es la tercera incidencia con mayor número de eventos, las alertas son generadas todos los días de la semana, siendo los más altos registros los jueves, viernes y sábado en horario vespertino y nocturno.

Adicionalmente, la Policía Nacional del Ecuador ha facilitado un informe de robos a personas en los alrededores de los Campus de la Universidad De Las Américas en los años 2017 y 2018. El Campus Colón es el que mayor número de incidencias registra, seguido de los Campus Queri y Granados, finalmente el campus UDLAPARK con 11 registros.

En el anexo 3 se pueden visualizar detalles del informe policial.

También existen parámetros para fijar la ubicación del sistema de cámaras, optar por lugares que no tengan obstáculos que obstruyan el campo de visión de las cámaras y preferir la mayor elevación posible para la instalación, esto mejora la cobertura visual (GVS Colombia, s.f.).

Las cámaras *PTZ* son la mejor opción para monitorear dinámicamente un espacio en concreto debido a su capacidad para realizar barridos del panorama, cambiar de ángulo verticalmente y acercamientos con *zoom* óptico y digital. Las cámaras, dependiendo del fabricante, pueden ser controladas mediante su interfaz web o su respectivo hardware controlador.

Para la implementación del sistema de pánico se han elegido cámaras *PTZ (pant – tilt – zoom)* de la marca Hikvision, modelo DS-2DE4425IW-DE. Las cámaras, de acuerdo con su hoja de datos (anexo 10), cuentan con un sensor infrarrojo para detectar el calor con un alcance máximo de 100 metros cuando las condiciones de luz no son adecuadas. Dichas cámaras se encuentran actualmente en el mercado nacional, pero esto no asegura su existencia dentro del país en un futuro. Se deberán volver a tomar en cuenta las características técnicas de una cámara similar para un nuevo análisis.

El sistema de videovigilancia, al implementar visión de 360º, cuenta con un alcance radial de 100 metros si no se consideran los obstáculos, como es posible observar en las figuras 1, 4, 7 y 8.

Para definir la ubicación de las cámaras se han tomado en cuenta las estadísticas del Informe de la Policía Nacional del Ecuador (anexo 3) y el alcance máximo con el que cuentan las cámaras seleccionadas.

Al momento de ejecutar el plan para una implementación del sistema, la localización de las cámaras puede variar debido a factores como; permisos municipales, comunitarios y públicos para la instalación, obras civiles o estructuras físicas que bloqueen el campo visual de la cámara, especificaciones técnicas de las cámaras elegidas o cualquier otro factor que

impida la instalación de la cámara en el sitio geográfico especificado.

Es importante considerar que varios de los espacios definidos no son propiedad de la Universidad De Las Américas, por lo tanto, se deben realizar las respectivas negociaciones con la entidad pública a cargo de la administración y de otorgar los permisos de instalación o a los dueños de la propiedad privada. Esto puede hacer que la ubicación de las cámaras varíe en el sistema de videovigilancia.

De ser necesario, en un futuro, se pueden incluir más cámaras a las contempladas en este análisis de ubicación estratégica del sistema.

2.2.1 Campus Colón

El campus se encuentra en la Av. Cristóbal Colón y José Tamayo. Las vías más transitadas por los estudiantes son la Av. Cristóbal Colón y la Av. 6 de Diciembre. Con frecuencia, también recorren las calles José Tamayo, Camilo Destruge, Plácido Camaño, Av. La Coruña y Av. 12 de Octubre.

La figura 1 indica la configuración de cámaras a usarse para el campus Colón.

La ubicación de las cámaras fue contemplada para cubrir todo el tramo de la Av. Colón desde la Av. 6 de Diciembre hasta la Av. La Coruña.

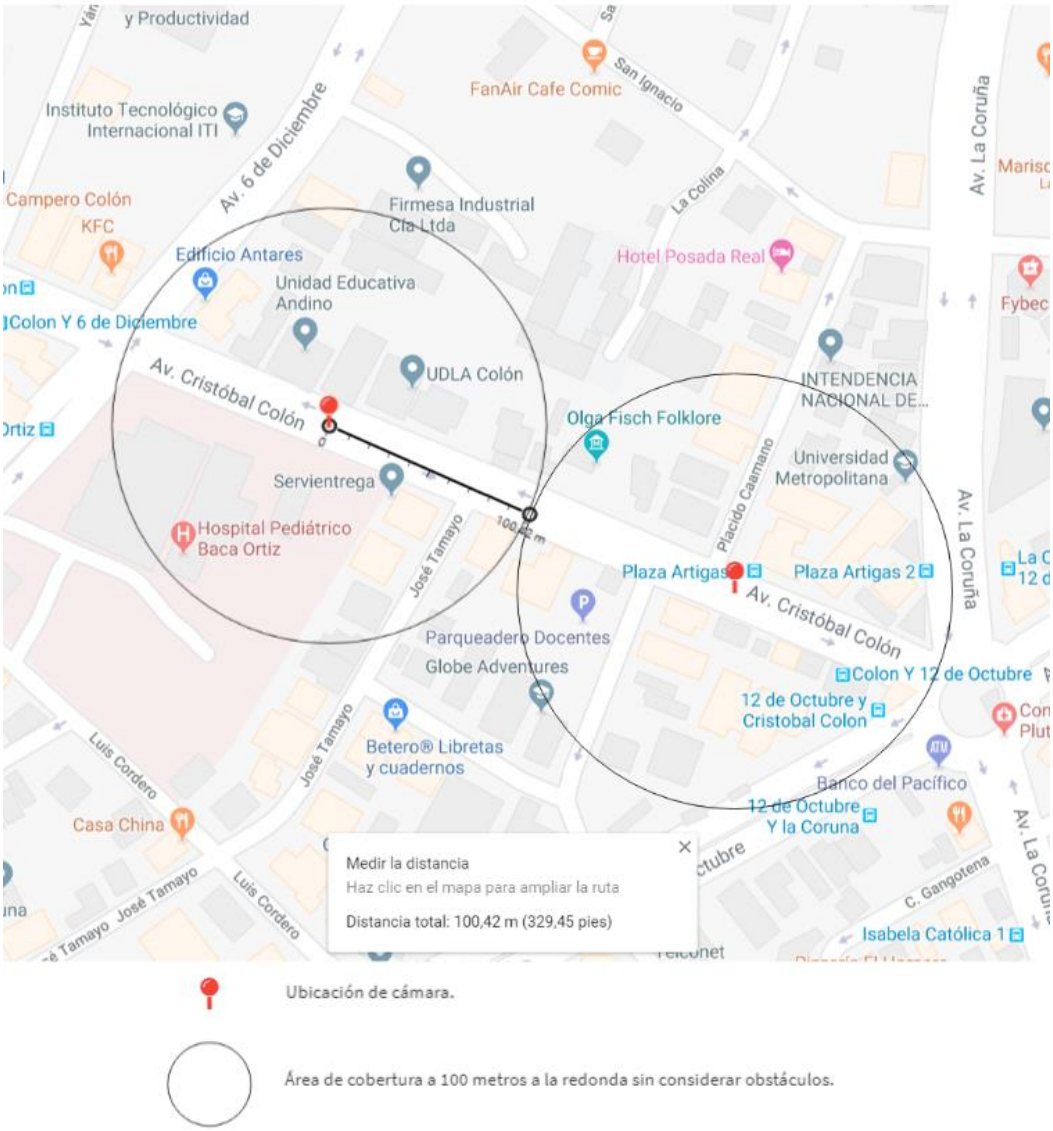


Figura 1. Mapa UDLA Sede Colón con puntos geográficos de cámaras. Adaptado de (Google Maps, s.f.).

Un aspecto importante para ser tomado en cuenta es que, algunas de las calles que se desean monitorear son muy extensas para cubrirlas con una sola cámara, como es el caso de la figura 1. Aún con un *zoom* óptico, lo suficientemente potente para hacer acercamientos extremos, el ángulo de visión disminuye y esto no es algo que favorece a la videovigilancia.

Para comprender por qué el campo de visión disminuye es necesario entender

el concepto de la distancia focal. La distancia focal es la relación entre la longitud óptica desde un punto donde la luz converge hasta plasmar la imagen clara de un cuerpo. A mayor distancia focal, menor es el ángulo de visión (Berkenfeld, Corrado, & Silverman, s.f.).

La distancia focal tiene una relación directa con el *zoom* aplicado en la cámara, a mayor *zoom* mayor será la distancia focal y, por lo tanto, el campo de visión será menor. En la figura 2 se muestra un ejemplo de la afirmación.

Con el concepto citado anteriormente, se puede comprender de una mejor manera porqué no es correcto el uso de una sola cámara para una calle o avenida que tiene una gran longitud, para el caso específico de este proyecto, dicha longitud es de 100 metros. Lo ideal es usar más de una cámara para captar los eventos que suceden a lo largo de la calle o avenida.

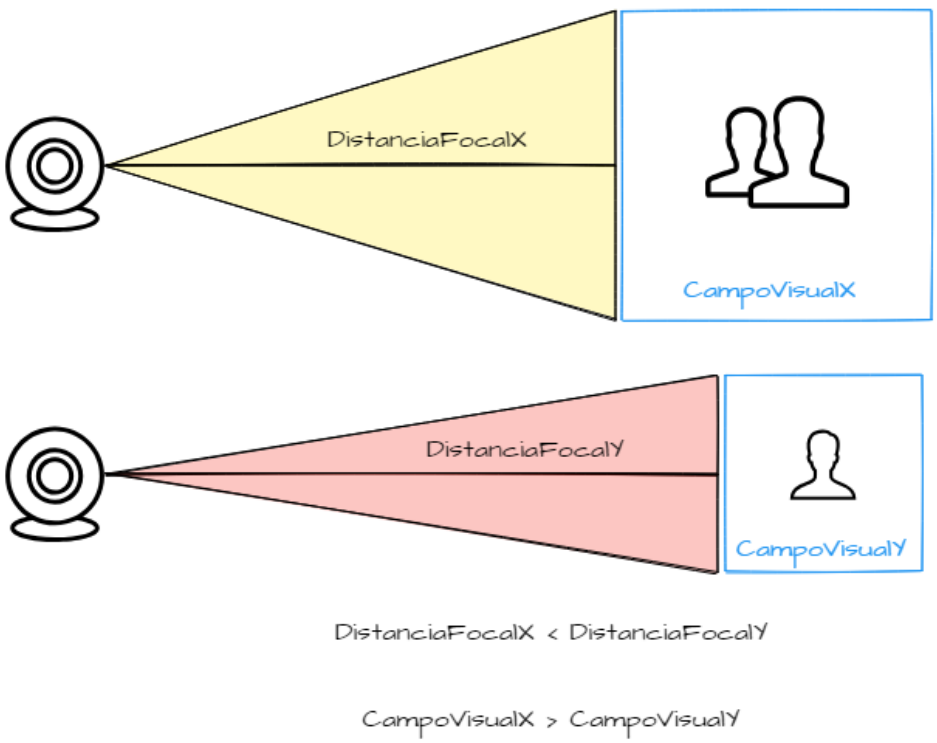


Figura 2. Relación entre el campo de visión y la distancia focal.

También se deben revisar las especificaciones técnicas de la cámara para

conocer cuál es la distancia máxima en la que ofrece un campo visual óptimo y con tecnologías que permitan una mejor visualización de imagen en condiciones de luz escasa, como infrarrojo, visión nocturna, entre otras. La cámara seleccionada ofrece una distancia máxima de 100 metros, considerando los factores expuestos.

La figura 3 muestra un esquema suponiendo un ambiente en donde varias cámaras deben cubrir un área extensa en una calle o avenida, para este ejemplo se muestra una cámara en un punto de la calle y una cámara en otro punto de la misma, exponiendo los tramos de cobertura de una de las cámaras.

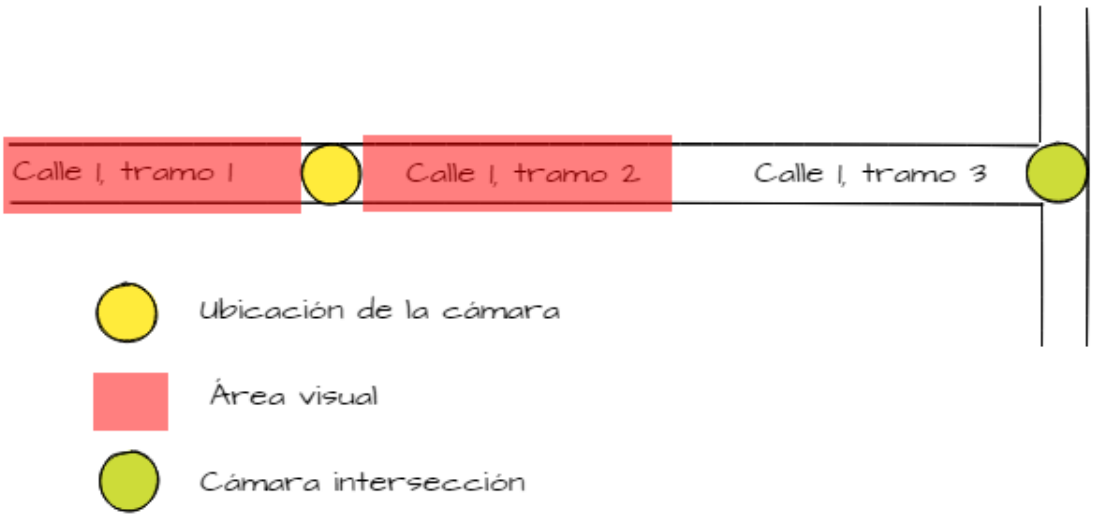


Figura 3. Cámaras en calles de gran extensión.

2.2.2 Campus Queri

En la calle José Queri, entre las Avenidas Eloy Alfaro y De Los Granados, se encuentra el Campus Queri. De manera peculiar, este campus cuenta con un pasaje de ingreso, tanto vehicular como peatonal. Por tal motivo, es importante otorgar campo de cobertura de videovigilancia a dicho pasaje de ingreso.

El tránsito peatonal de los estudiantes tiene un mayor flujo en la Calle José Queri y en la Av. De Los Granados. Los estudiantes, también frecuentan la Av. Eloy Alfaro.

En la figura 4, se aprecia en el diseño del sistema de cámaras en el campus

Queri. Las cámaras son colocadas, en su mayoría, en las intersecciones. Existe una intersección en “cruz” y una intersección en “T” en el diseño propuesto.

La posición de las cámaras cubre toda la calle José Queri y sus respectivas intersecciones con las avenidas De Los Granados y Eloy Alfaro. También se cubre el área de la entrada principal al campus Queri.

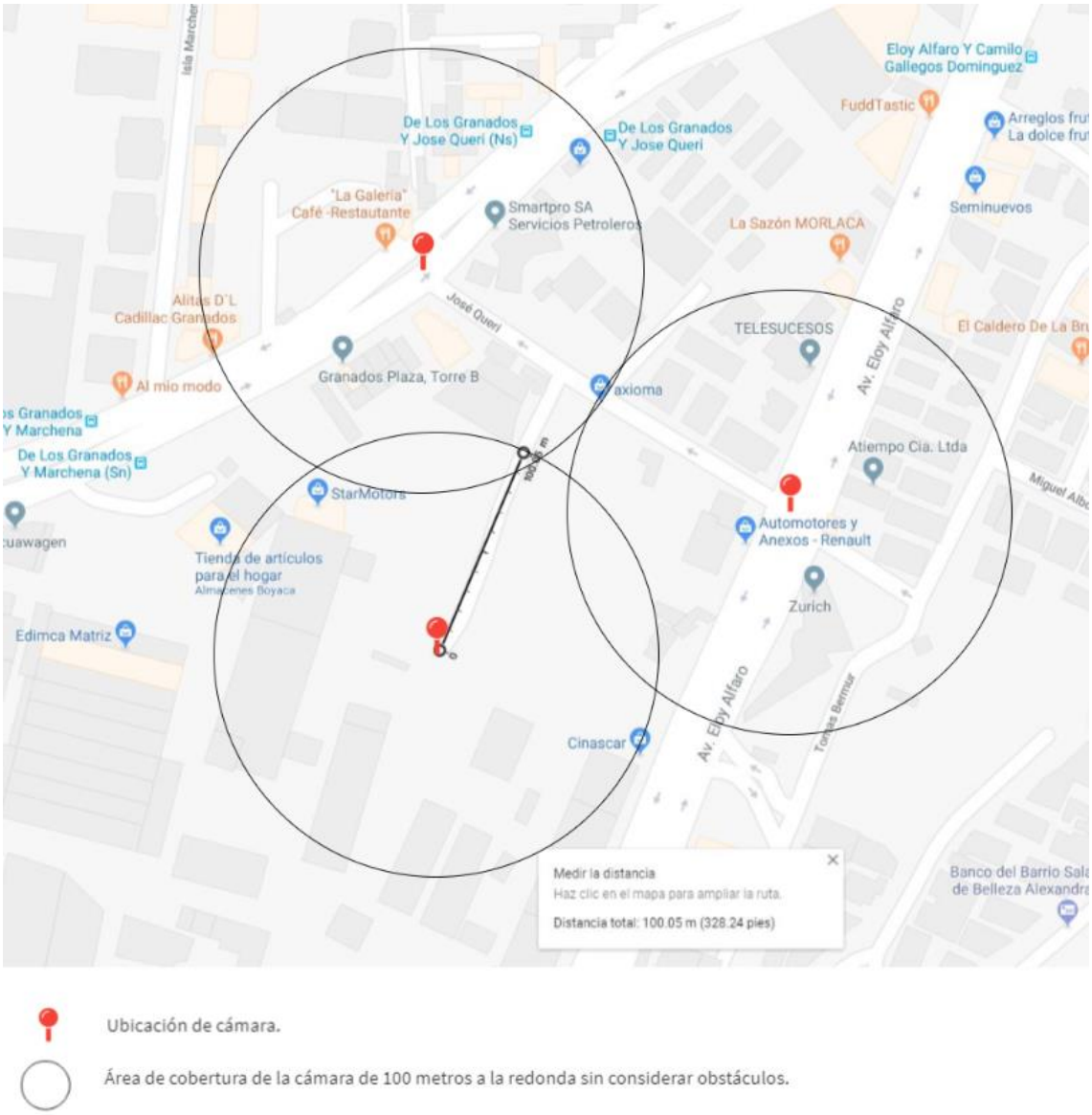


Figura 4. Mapa UDLA Sede Queri con puntos geográficos de cámaras. Adaptado de (Google Maps, s.f.).

Teniendo en cuenta que las cámaras se instalarán en los exteriores de los Campus de la UDLA, los mejores lugares para colocar el sistema de cámaras son en las intersecciones de las calles. La decisión es acertada debido a que con una cámara *PTZ* se puede supervisar más de una calle de acuerdo con la posición a la que apunte el lente de la cámara, tal y como los modelos de las figuras 5 y 6.

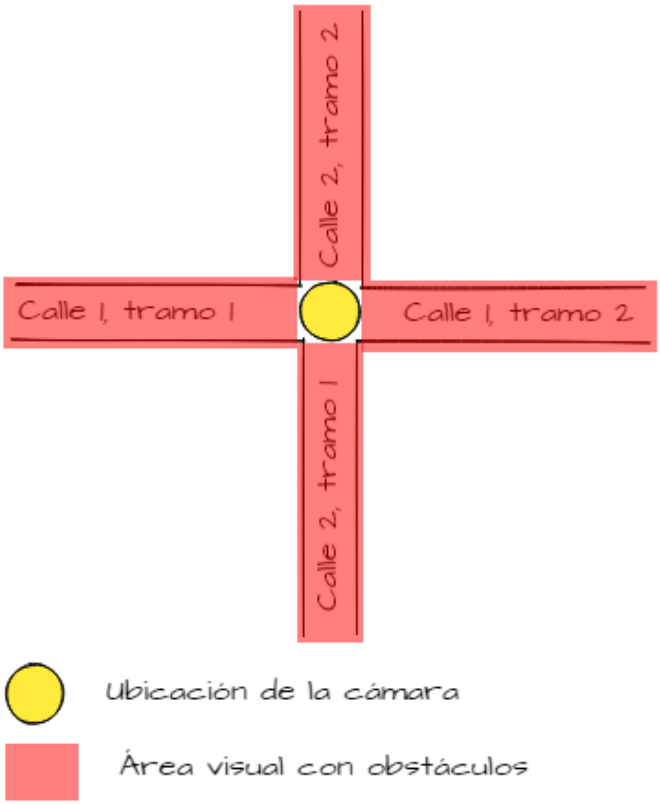


Figura 5. Área visual de la cámara con obstáculos en intersección en cruz.

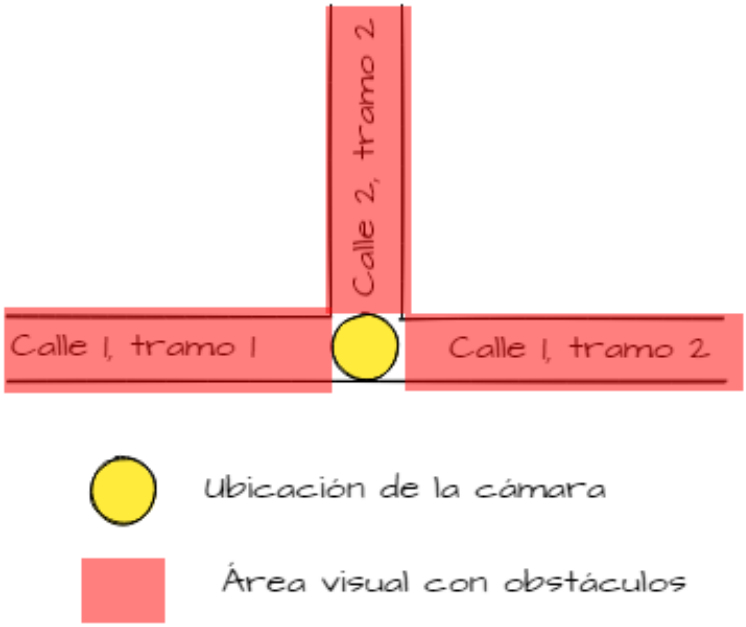


Figura 6. Área visual de la cámara con obstáculos en intersección en “T”.

2.2.3 Campus UDLAPARK

El campus se localiza entre la Vía a Nayón y la Av. Simón Bolívar, siendo esta última la que carece de circulación peatonal de estudiantes por tener características de una autopista perimetral.

La Vía a Nayón, es la única calzada para ingresar a este Campus, por lo cual, cubrir la extensión de esta calle es de vital importancia. Se deben instalar cámaras en varios puntos de la vía, de acuerdo con la configuración de cámaras de la figura 3. A continuación, en la figura 7, se detallan las ubicaciones de las cámaras.

Figura 7. Mapa UDLA Sede UDLAPARK con puntos geográficos de cámaras.

Adaptado de (Google Maps, s.f.).

2.2.4 Campus Granados

Según las estadísticas expuestas en el anexo 3 este campus ubicado en la Av. De Los Granados y De Los Colimes, es la sede que abarcará la mayor cantidad de cámaras, debido a la diversidad de ubicaciones de robos a personas registrados de manera estadística. Los estudiantes transitan principalmente en la Av. De Los Granados, Isla Marchena y De Los Colimes.

Otro punto que genera alertas, especialmente entre los estudiantes que circulan por el sector en horario nocturno, es la calle Joel Polanco, intersecada por las calles De Los Colimes e Isla Marchena. Se puede observar que las cámaras propuestas en estas calles tienen la configuración de la figura 3 (cubrir una calle de gran extensión). También, se colocan las cámaras en 2

intersecciones estratégicas, De Los Colimes – Joel Polanco (intersección en “cruz” que consta en la figura 5) y De Los Granados – De Los Colimes (intersección en “T” que se explicó en la figura 6).

En la figura 8, se detallan las ubicaciones estratégicas de las cámaras para cubrir la mayor superficie posible de las calles alrededor del campus.

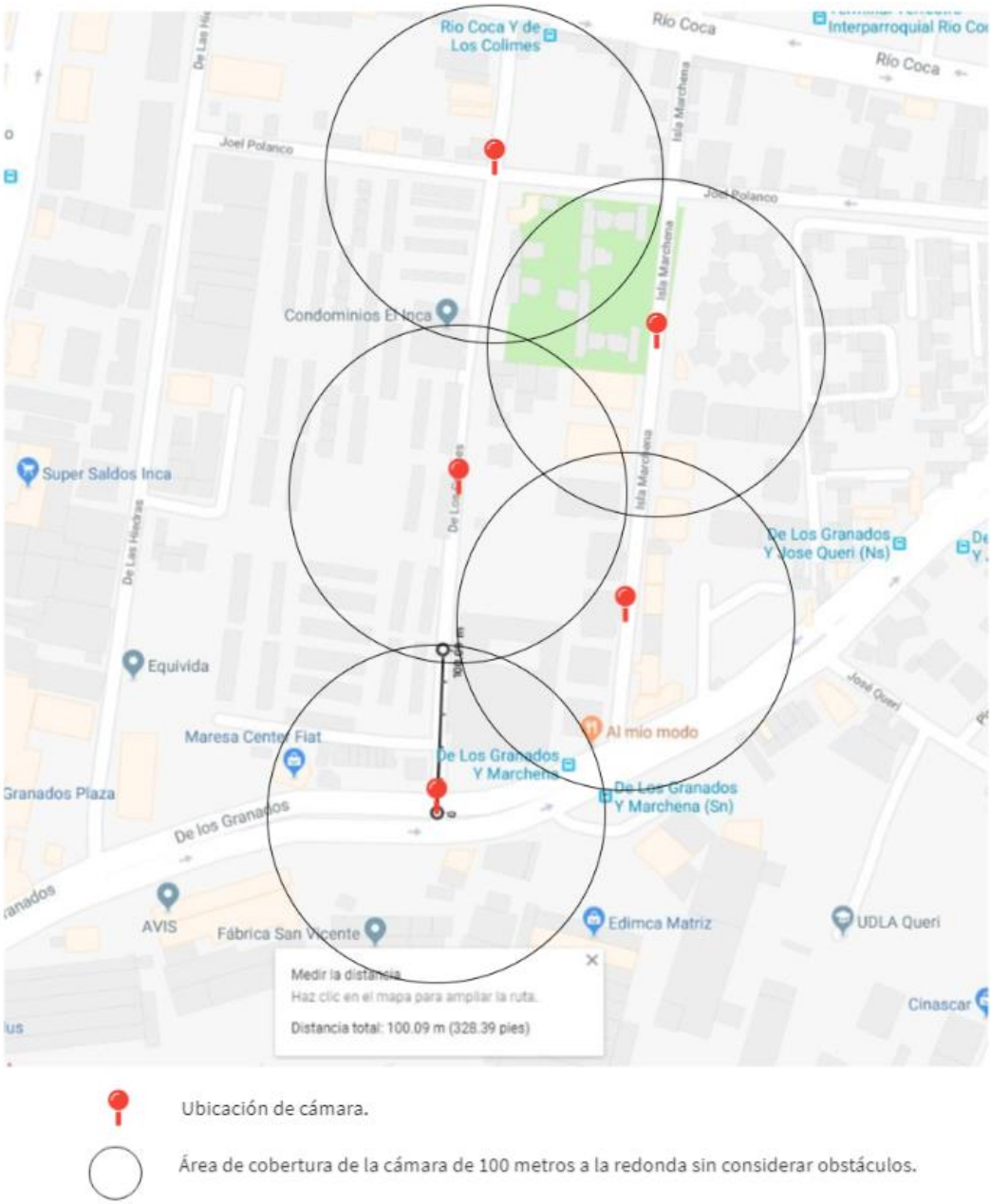


Figura 8. Mapa UDLA Sede Granados con puntos geográficos de cámaras. Adaptado de (Google Maps, s.f.).

En la tabla 1 se encuentra la información con las coordenadas de todas las cámaras de todos los Campus detalladas anteriormente. También se considera la distancia que existe entre cada cámara y el campus universitario, así como un poste de 9 metros de altura para posteriormente realizar el cálculo de materiales necesarios en el caso de una implementación real. En el anexo 7 se puede encontrar los detalles de los cálculos de distancia efectuados.

Tabla 1.

Coordenadas geográficas de las cámaras.

Campus	Etiqueta	Latitud	Longitud	Distancia desde campus	Altura poste	Distancia total
Colón	Colon1	-0.202463	-78.485593	72m	9m	81m
Colón	Camano1	-0.203082	-78.483971	167m	9m	176m
Queri	EntradaQueri1	-0.1691	-78.470919	36	9m	45m
Queri	GranadosQueri1	-0.16754	-78.470922	282m	9m	291m
Queri	EloyAlfaro1	-0.168444	-78.469439	328m	9m	337m
Granados	GranadosColimes1	-0.168398	-78.473132	46m	9m	55m
Granados	Colimes1	-0.166628	-78.472933	209m	9m	218m
Granados	Colimes2	-0.164809	-78.472713	415m	9m	424m
Granados	Marchena1	-0.167282	-78.472075	214m	9m	223m
Granados	Marchena2	-0.165769	-78.471919	378m	9m	387m
Udlapark	NayonBolivar1	-0.162748	-78.462611	444m	9m	453m
Udlapark	Nayon1	-0.162367	-78.460868	256m	9m	265m
Udlapark	EntradaPark1	-0.162844	-78.459398	64m	9m	73m

2.3 Control del clúster de videovigilancia del servidor

Implementar un clúster de videovigilancia en el prototipo tiene el objetivo de mostrar al encargado del sistema de videovigilancia el video en tiempo real de una o de varias cámaras cercanas que se encuentren dentro del rango de cobertura de un evento reportado por el usuario desde su aplicación móvil. El rango de cobertura, de acuerdo con las especificaciones de la cámara seleccionada es de 100 metros. En las figuras 1, 4, 7 y 8 se presentan los

diseños con el rango especificado.

Para hacer efectivo el funcionamiento del clúster, se consideró que la mejor opción es definir un rango de cobertura en un radio a la redonda de las coordenadas geográficas donde se encuentra físicamente la cámara.

Definir el funcionamiento del clúster de la manera propuesta anteriormente se debe a que, al añadir una nueva cámara o al cambiar la ubicación de una cámara instalada previamente, solamente toma copiar líneas de código e ingresar como único parámetro las nuevas coordenadas de donde se va a situar la cámara. Por el contrario, en el caso de no implementar el rango de cobertura radial, un algoritmo de definición de cobertura rectangular requiere 3 parámetros: coordenadas de la cámara, coordenadas del límite inferior y coordenadas del límite superior. Esto puede suponer un problema al momento de incluir una nueva cámara o modificar la posición de una cámara existente en el sistema. Se puede observar un diagrama de lo que se acaba de lo explicado en la figura 9.

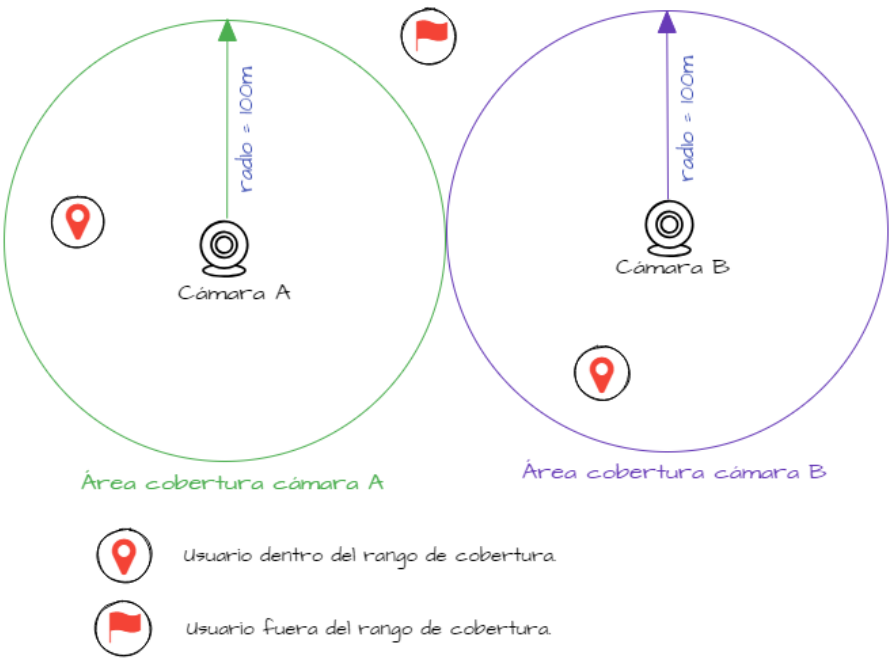


Figura 9. Diagrama del área de cobertura del clúster de videovigilancia.

Si un usuario de la aplicación móvil reporta un evento dentro del área de cobertura de las cámaras, a las personas encargadas les llegará un correo electrónico con la notificación de dicho evento y un segundo correo electrónico con el enlace para ingresar a monitorear el contenido multimedia de las cámaras en tiempo real. Por el contrario, si el usuario de la aplicación se encuentra fuera del área de cobertura, solamente les llegará a los encargados un correo electrónico con la notificación del evento, mas no el enlace para acceder a las cámaras. En la figura 9, se fija un radio de cobertura de 100 metros a la redonda para realizar las pruebas del prototipo del sistema de pánico, este parámetro puede ser cambiado en un futuro dependiendo de las necesidades del área a la redonda de cada cámara.

2.4 Proceso operativo de respuesta del sistema ECU911

Para entender la forma en la cual se procesan los mensajes recibidos por medio de la aplicación desarrollada en Android por el Sistema Integrado de Emergencias ECU911, se debe conocer cómo es el proceso operativo por el cual se reciben y responden las notificaciones que llegan hacia los operadores de ECU911.

El proceso operativo comienza cuando un usuario descarga la aplicación en su teléfono, en esta se tienen que consentir las cláusulas para ejecutar la aplicación; existen políticas de protección para el mal uso de la aplicación. Una vez que se aceptan los términos de uso, se procederá a llenar un formulario donde aparte de datos personales se solicitará al usuario su tipo de sangre, si tiene alguna discapacidad (visual, auditiva, motriz) y se pide también ingresar un contacto de emergencia el cual será usado en caso de que el usuario sea víctima de un evento; Estos datos serán almacenados dentro de la aplicación en preferencias compartidas y serán encapsulados junto con la ubicación (latitud y longitud) y el código de la emergencia que el usuario selecciona.

Esta notificación es recibida por un evaluador del servicio integrado de seguridad ECU911, el cual tiene la obligación de devolver la llamada al usuario que hizo la petición; esto no se permite en caso de que la emergencia sea robo personal, robo a domicilio o secuestro (ECU911).

Durante la llamada el responsable del ECU911 evalúa la notificación para que esta sea derivada hasta el departamento de seguridad competente, el cual se encarga de movilizar los recursos para responder la emergencia que se recibió usando la aplicación. En la figura 10 se explicará cómo funciona operativamente cuando se reporta un evento dentro de la aplicación de ECU911.

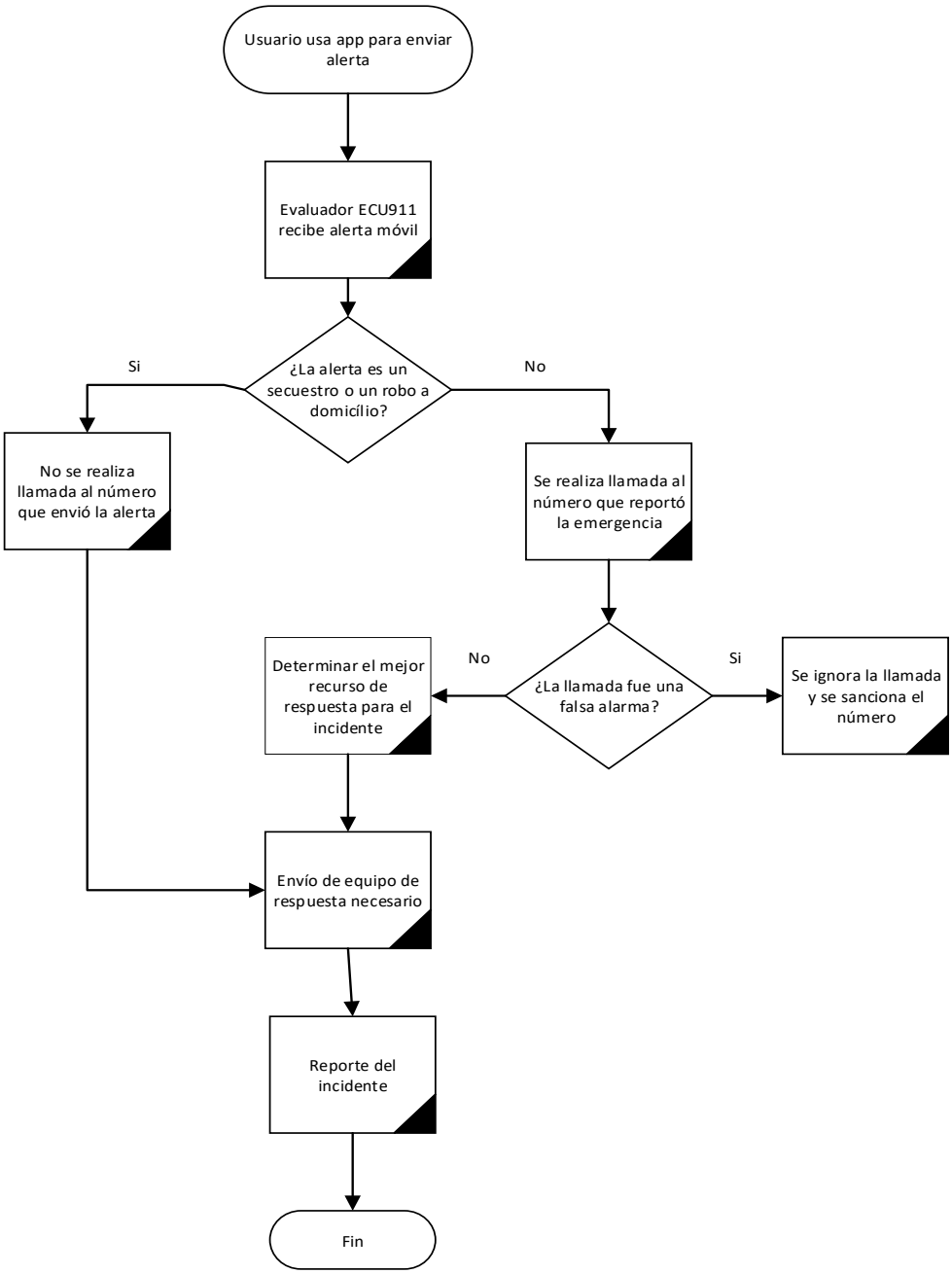


Figura 10. Diagrama de flujo de funcionamiento de la aplicación desde ECU911 hacia el cliente.

2.5 Definición de tecnologías

Es importante conocer las tecnologías con las cuales es posible establecer una comunicación con el sistema integrado ECU911. Para esto se tomará en cuenta que los mensajes recibidos por medio de dispositivos móviles son manejados dentro de un servidor de servicios web. Estos mensajes recibidos son del tipo SOAP (*Simple Object Access Protocol*) y son enviados usando como formato de cadenas JSON (*JavaScript Object Notation*). A continuación, se explicará cada uno de ellos.

2.5.1 Servicios WEB

Servicios Web son componentes que son consumidos por los protocolos Web usando XML (*eXtended Markup Language*) para realizar y responder solicitudes entre cliente-servidor.

Se hace referencia a estos como una serie de métodos a los que se puede interactuar desde un explorador web cualquiera. Cuando se realiza una petición a un servidor web lo que se recibe como respuesta es por lo general una página web en el caso de ser *http*. Estos son independientes del lenguaje usado para su concepción debido a que todos pueden ser interpretados por los navegadores de internet (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014). Entre sus principales características se encuentran:

- Debe ser accesible por medio de internet. Por lo cual debe usar protocolo estandarizado *http*.
- Debe contar con una descripción de sí mismo, de esta forma el navegador podrá interpretar la funcionalidad y la interfaz de la aplicación sin la necesidad de la intervención de un usuario.
- Tener un mecanismo el cuál pueda permitir que se localice al servicio web dependiendo de su funcionalidad. Así se permite que el navegador ubique el servicio de forma automática.

2.5.1.1 Tipos de servicios Web

En un nivel conceptual, un servicio web es un software que se proporciona por medio de un servidor. Estos usan mensajes para el intercambio de la

información. En un nivel más técnico, un servicio web puede ser implementado de varias maneras. Se diferencian dos tipos de servicios: RESTFul (*Representational State Transfer*) y SOAP las cuales serán detalladas a continuación. (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014)

2.5.1.2 Servicios *Web SOAP*

SOAP tiene características de mensajería independiente del transporte. Se encarga de transmitir datos XML como mensajes *SOAP*; Cada uno de estos mensajes tiene funciones conocidas dentro de documentos *XML*. La estructura de este documento sigue un patrón específico, pero el contenido va a variar dependiendo de la funcionalidad que se le dé al mensaje. Los mensajes *SOAP* van a ser enviados por medio de *HTTP*, contando como protocolo estándar (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014). La figura 11 muestra detalles de la estructura *SOAP*.

Figura 11. Estructura de un mensaje SOAP.

Tomado de (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014).

Un mensaje SOAP consiste en:

- En el documento SOAP existe un elemento principal, la etiqueta <*Envelope*>. Este es la etiqueta que inicia el documento XML. La misma divide en dos partes: la primera es el encabezado y la segunda es el cuerpo del mensaje.

- El encabezado contiene los datos de enrutamiento que permitirá al documento XML, el cliente al cual deberá llegar.
- El cuerpo contendrá el contenido del mensaje que se desea enviar hacia el servidor.

2.5.1.3 WSDL (Web Services Description Language)

Un documento *WSDL* describe un servicio Web; especifica la localización de servicio y los métodos que usará para establecer la conexión de transferencia de datos. El cliente que invoque un servicio web debe conocer la ruta en la cual este reside para poder ejecutarlo. A su vez necesita saber cuál es la funcionalidad para que pueda invocar el servicio correcto. (IBM Technologies, s.f.)

2.5.1.4 Arquitectura de los servicios Web

Todo *framework* necesita cierta arquitectura para asegurarse que todo en ambiente de trabajo funcione correctamente. En las aplicaciones Web existe una arquitectura la cual consiste en tres roles distintivos:

- Proveedor: El proveedor crea el servicio web y hace que este esté disponible a cualquier solicitud que se le presente.
- Solicitante: Un solicitante es aquel cliente que busca o pide consumir un servicio Web. Esta puede ser .Net, Java u otro lenguaje de programación soportado por un servicio web.
- *Broker*: Es la aplicación que da acceso al UDDI (*Universal Description, Discovery and Integration*). Esto permite a la aplicación cliente que localice el servicio web.

A continuación, la figura 12 ejemplifica un diagrama de la comunicación cliente – servidor al momento que este recibe una petición de transferencia de un paquete de datos JDBC (*The Java Database Connectivity*) que es un estándar de comunicación entre el lenguaje de programación Java y una base de datos SQL (*Structured Query Language*) (1), el servidor acepta la transferencia (2) y se realiza una conexión usando SOAP para el envío del paquete de datos(3).

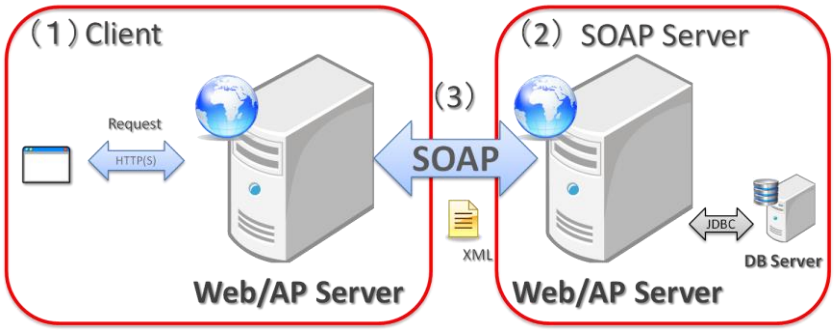


Figura 12. Comunicación cliente/servidor SOAP.

Tomado de (NTT DATA, s.f.).

2.5.1.5 Servicios Web visto desde el cliente

Un cliente de servicio Web puede o no estar dentro del mismo equipo para ser consumido. Se trata de proporcionar la mayor transparencia en caso de que el servicio sea contactado remotamente debido a que el cliente no es capaz de diferenciar si el servicio es local o remoto. (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014). En la figura 13, se evidencia el servicio visto desde un cliente, a través de un proveedor de servicios con el componente *Port* el contenedor del servicio. A su vez, se evidencia la interfaz y la clase *Service* (*SI: Service Interface*), y un terminal de servicio (*SEI: Service Endpoint Interface*).

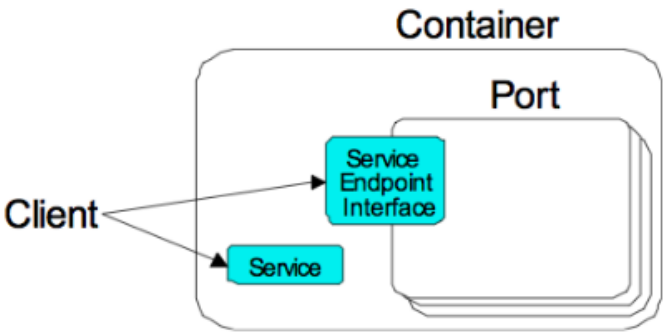


Figura 13. Servicio Web visto desde el cliente.

Tomado de (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014).

Se definen los métodos por los cuales un cliente puede hacer uso de un componente *Port* en un servicio web si este se encuentra declarado en la clase *Service*. El cliente usa esta interfaz para acceder al componente *Port*. Este no cuenta con una identidad hacia el cliente, debido a que lo considera como un objeto sin estado.

Cuando un cliente accede a un servicio web, lo hace usando un SEI, este puede ser especificado por el servidor que provee el servicio web. Por el lado del servidor, el *runtime* de este contenedor es el que se encarga de asignar las clases que se encargarán de responder las solicitudes desde el cliente. (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014)

2.5.1.6 Servicios Web vistos desde el Servidor

El documento *WSDL* que se encuentra en formato *XML* se encarga de interpretar diferentes servicios web de ser necesario, e incluye especificaciones de requerimientos sobre el transporte de los datos y su formato en la red. Este documento no impone requerimientos sobre los modelos de desarrollo establecidos tanto para servidor como para cliente como se puede observar en la figura 14. Dentro de la especificación de servicios Web para Java EE (JSR-109) Existen tres formas de implementar un servicio web siguiendo una lógica de negocio. (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014).

- *Bean* de Sesión: El desarrollo del servicio Web se establece definiendo un *Bean* de sesión sin estado, integrando los métodos que se encuentran dentro del *SEI (Service Endpoint Interface)*.
- Clase Java: Para este caso particular el *Port* es implementado como un *Servlet JAX-WS*.
- *Singleton Session Bean*: De ser el caso se procederá a implementar un *singleton session bean* en el cual se establecen los métodos de un SEI.

Como se muestra en la figura 14 el componente antes presentado *Port* define cómo el servidor va a visualizar el servicio Web. Cada uno de estos proporciona el servicio que se encuentra en una dirección MAC la cual está dentro del atributo *address* en la etiqueta <*port*> en el archivo *WDSL*. El despliegue del servicio depende del contenedor del componente antes mencionado, esta define una clase Java que implementa los métodos disponibles en el *SEI* (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014).

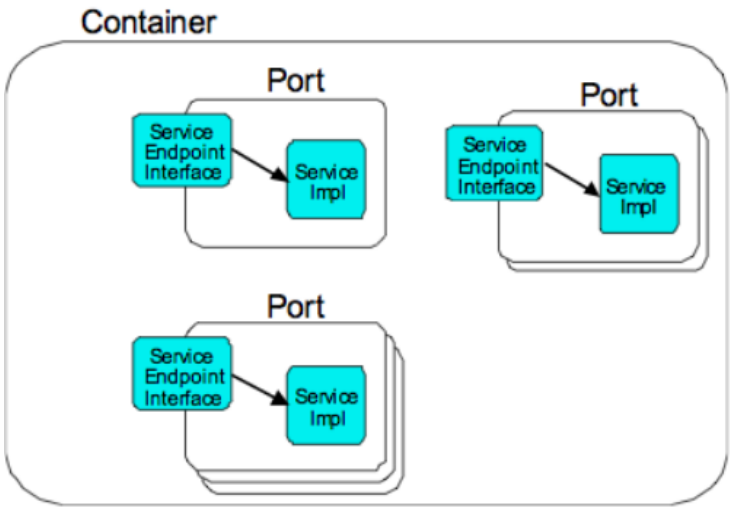


Figura 14. Servicio Web visto desde el servidor.

Tomado de (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014).

2.5.1.7 Despliegue de un servicio web

La herramienta de despliegue se encarga de iniciar el proceso verificando la petición de un cliente, para de esta forma se pueda evaluar cuales son los módulos que se encuentran dentro del servicio Web, una vez obtenida la información de enlace (*binding*), se entrega los componentes y servicios web definidos en el módulo. Posteriormente, hace pública la información sobre el documento *WSDL* que contiene información sobre los servicios web que han sido desplegados e inicializa la aplicación a la espera de peticiones (Departamento de Ciencia de la Computación e Inteligencia Artificial, 2014).

2.5.2 JSON

JSON representa un formato de accesible para ser compilado y ejecutado por las estaciones de trabajo que está basado en *JavaScript*. *JSON* usa objetos que no dependen de un lenguaje de programación, pero se basa formatos de texto similares a los presentes en lenguajes como *C, C++, C# Java*, *Javascript*, *Perl*, *Python* y muchos otros. Esto hace de *JSON* una herramienta ideal para la transferencia de datos. (ECMA International, 2017)

JSON está dividido en 2 estructuras:

- Colección de pares nombre/valor (objetos, diccionarios, estructuras).
- Lista de los valores. Para la mayoría de los lenguajes de programación se realiza con arreglos (*arrays*, vectores, listas o secuencias).

Estructuras de datos universales que pueden ser interpretados por cualquier lenguaje de programación que soporte arreglos (*arrays*) de una forma u otra.

En *JSON* se cuentan estas formas:

Objeto: Conjunto no ordenado de pares (nombre/valor). Este comienza con una llave de apertura y finaliza con una de cierre. Cada sentencia es seguida por dos puntos y los pares se separan con el símbolo de coma. En la figura 15 se muestra un diagrama de la estructura de un objeto JSON.

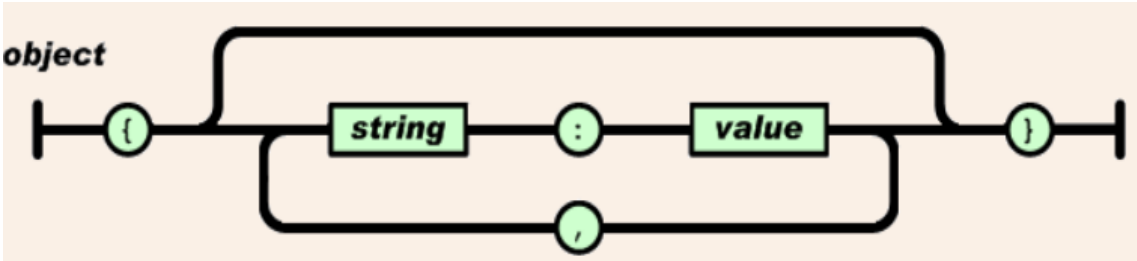


Figura 15. Formato de un objeto.

Tomado de (ECMA International, 2017).

Arreglo (array): Un arreglo es una cadena de texto que se encuentra entre corchetes, como se muestra en la figura 16. De igual forma los valores son

separados por una coma.

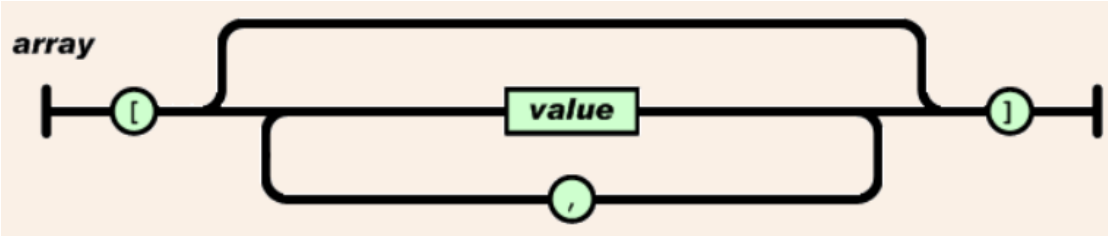


Figura 16. Formato de un arreglo (array).

Tomado de (ECMA International, 2017).

Cadena de datos (*String*): Es una cadena de caracteres que se encuentran en un conjunto de comillas llamado *string*; estos pueden ser un número, una variable booleana la cual es posible observar en la figura 17. Estos valores pueden ser anidados.

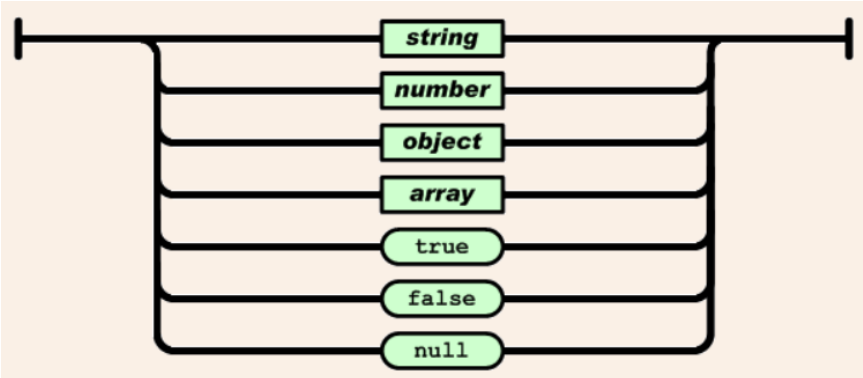


Figura 17. Formato de *String.*

Tomado de (ECMA International, 2017).

Cadena de caracteres (*String*): Es un conjunto de 0 o más caracteres en formato Unicode, que se encuentran entre comillas; un carácter es representado como una cadena simple de caracteres detallada en la figura 18.

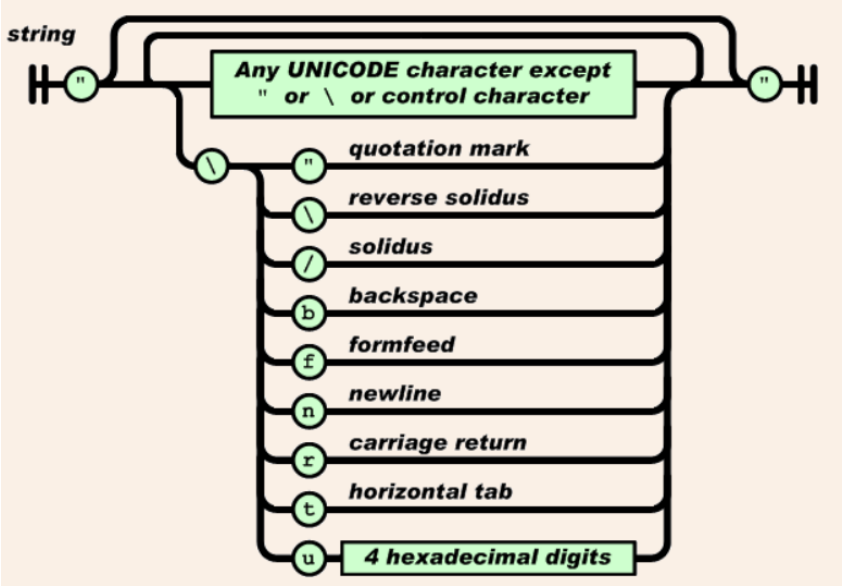


Figura 18. Tipos de cadenas de caracteres.

Tomado de (ECMA International, 2017).

2.5. 6 Envío de alerta hacia ECU911 usando servicios Web

La aplicación envía un mensaje SOAP usando como formato JSON hacia el servidor de recepción de notificaciones *smartphone* del ECU911 con los datos del usuario, contacto de emergencia y el código del incidente el cual se quiere para asumir el tipo de asistencia a darse por parte del evaluador a cargo. El servidor al recibir esta petición responde hacia el usuario solicitante con el siguiente código, que por motivos de explicación será llamado R:

- R=-3: El usuario no se encuentra en el área de cobertura de ECU911.
- R=-2: El número se encuentra bloqueado por mal uso.
- R<0: Responde con el número de incidencia a la espera de que un evaluador tome el incidente

Posteriormente la aplicación hará una petición para saber el estado de la emergencia. En la figura 19 se presenta una explicación sobre los estados de una petición hacia ECU911.

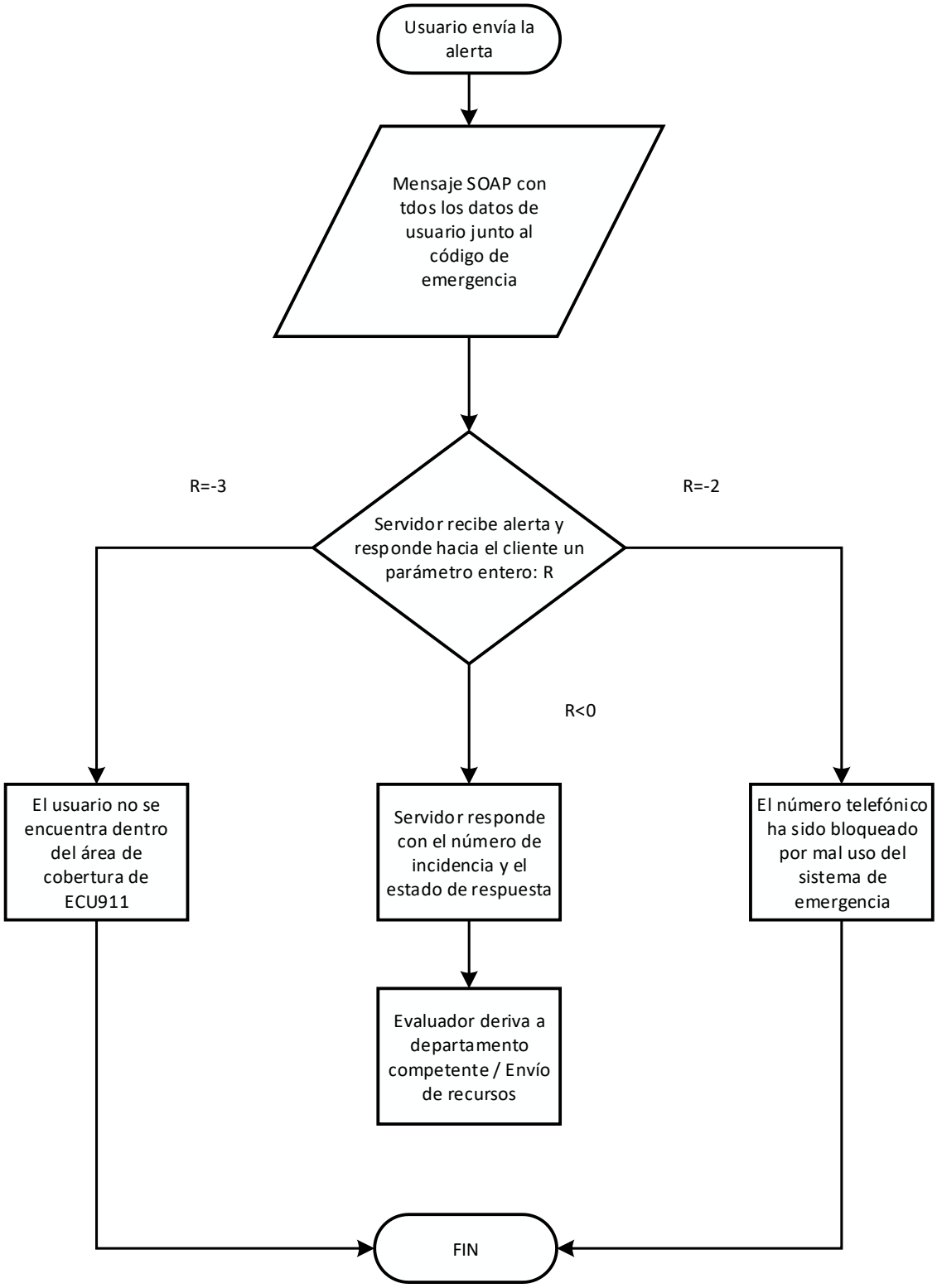


Figura 19. Diagrama de flujo de interacción cliente, servidor dentro de ECU911.

2.6 Características en Android

Para el desarrollo de la aplicación de servicio de emergencia dentro del sistema Android se tomará en consideración dos características principales dentro del sistema operativo:

- *Broadcast Receiver*
- *Services*

2.6.1 Mensajes *Broadcast* en Android

Las aplicaciones en Android pueden recibir mensajes de difusión del sistema o de otra aplicación las cuales son similares al patrón de diseño publicar-suscribir. Estos mensajes son enviados cuando un evento de interés ocurre.

Las aplicaciones pueden registrar difusiones creadas específicamente para un evento en el que el usuario intervenga. Algunos ejemplos pueden ser: el dispositivo ha sido reiniciado o se puso a cargar. (Google LLC, 2018)

Las aplicaciones pueden registrarse para recibir mensajes de difusión específicos; cuando un mensaje es enviado, el sistema lo enruta automáticamente hacia las aplicaciones que han sido suscritas para recibir específicamente este tipo de broadcast.

Los mensajes de broadcast son usados para que la aplicación esté a la escucha de las acciones que el usuario realice con el botón de encendido

2.6.1.1 Broadcast Receiver

Se denomina *intent* a una acción que es usada para solicitar otra acción de otro componente dentro de la aplicación que se esté desarrollando (Google LLC, 2018). Para este caso práctico esta acción será el esperar y sumar la cuenta de cuántas veces es presionado el botón de poder (*POWER*).

El *Broadcast Recevier* permite a la aplicación recibir *intents* que se están emitiendo aún si la aplicación no se encuentra en ejecución en segundo plano. Existen dos formas de realizar esto: la primera es declararla dentro del archivo AndroidManifest.xml, el cual es un archivo de configuración que se encuentra en la raíz y donde se puede aplicar cambios a la aplicación en general; a su vez se puede crear el receiver dinámicamente en el código y registro (Google

LLC, 2018).

2.6.2 Servicios en Android

Un servicio dentro de Android es un componente que se encarga de realizar tareas que se ejecutan en *background* y que no están asociadas a un modelo gráfico, lo que permite al usuario cambiar de aplicación o dejar de usar su dispositivo sin interrumpir la tarea que se encuentra en nuestro servicio. Dicho servicio puede variar entre dos características:

- Servicio Iniciado: se da inicio en el momento que es llamado usando el componente de sistema *startService().* Este puede ejecutarse en segundo plano indefinidamente inclusive cuando se destruye el componente el cual lo inició.
- Servicio de enlace: se inicia en el momento que un módulo de un aplicativo se enlaza a él. Este brinda arquitectura Cliente - Servidor permitiendo a dichos módulos de la aplicación o componentes externos interactuar entre sí, entre ellos intercambiar resultados, iniciar procesos de comunicación entre diferentes aplicaciones como otras características.

El servicio se creará en una subclase dentro de la implementación, donde existirán métodos de *callback,* los cuales toman control de funciones fundamentales en el ciclo de vida de un servicio y de la aplicación. Los más importantes son:

- ***onStartCommand()***

Se invoca a este método una vez que el servicio es iniciado por una *activity* o un componente usando *startService().* Una vez es ejecutado este método, el servicio pasa a segundo plano y se ejecuta indefinidamente.

- ***onBind()***

Se llama a este método una vez que otro componente necesita usar el servicio. A este método se le proporcionará una interfaz, que será vital para los usuarios en la comunicación con el servicio, arrojando una variable *iBinder*. En cualquier

tipo de aplicación se debe implementar este método; pero si no se quiere responder a una petición se devolverá *NULL*.

- **onCreate()**

Este método es llamado, solamente, cuando el servicio es creado. Se usa para administrar las instrucciones de las configuraciones una sola vez. Si el servicio se encuentra en ejecución, no se necesita llamar a este método.

- **onDestroy()**

Se usa este método cuando el servicio no se encuentra en uso y está siendo finalizado. El sistema lo usa para liberar recursos del sistema que estén siendo usados por el servicio para dárselos a la aplicación que el usuario esté usando. De esta forma se detiene un servicio.

Cuando un módulo invoca a *startService*(), también se llama a *onStartCommand().* El servicio se ejecutará hasta que se detenga con *stopSelf()* o por la acción de un componente que invoca a *stopService().*

Si *bindService()* es llamado por un componente para la creación del servicio y no es llamado por *onStartCommand(),* el servicio es ejecutado si se encuentra enlazado con el módulo. El sistema elimina los servicios en caso de que dicho módulo se desconecta de los clientes. (Google LLC, 2018)

Android sólo fuerza un servicio que se ha iniciado sea detenido en caso de que la memoria sea insuficiente en el sistema y necesite recobrar recursos para la actividad en la que se encuentra el usuario. Cuando el servicio depende de la actividad en la que el usuario se encuentra, es menos probable que este sea destruido; si el mismo está declarado para su ejecución en primer plano, casi nunca se detendrá. Por otro lado, si se ha iniciado el servicio y es de larga duración, se lo colocará dentro de una baja jerarquía dentro del administrador de tareas.

En este caso de estudio invocaremos el servicio declarándolo dentro del AndroidManifest.xml.

2.7 Características de iOS

2.7.1 Ejecución en *background*

A diferencia de Android, iOS cuenta con una forma diferente con las que se ejecutan las aplicaciones en *background*, pues existe un listado limitado de las tareas de larga ejecución que pueden permanecer en *background*:

- Aplicaciones que reproducen contenido audible para el usuario mientras se encuentra en segundo plano, como una aplicación de reproductor de música.
- Aplicaciones que graban contenido de audio mientras se encuentra en segundo plano.
- Aplicaciones que mantienen a los usuarios informados de su ubicación en todo momento, como una aplicación de navegación.
- Aplicaciones compatibles con el protocolo de voz sobre IP (*VoIP*).
- Aplicaciones que necesitan descargar y procesar contenido nuevo regularmente.
- Aplicaciones que reciben actualizaciones periódicas de accesorios externos.

Las aplicaciones que cuentan con esta clase de tareas deben declarar los servicios que soportan y usar *frameworks* del sistema para implementar aspectos relevantes (Apple Inc., 2017).

2.7.2 Capturar eventos de *hardware*

Desafortunadamente, no existe información oficial acerca de si se pueden capturar los eventos de botones físicos en iOS de la misma manera que Android, pero varios colaboradores y participantes de la comunidad de desarrolladores en páginas web como stackoverflow.com o community.clickteam.com manifiestan que no es posible capturar dichos eventos debido a que el *software* de iOS limita a los desarrolladores a usar componentes del *hardware* por completo. En el anexo 6 se puede encontrar

información acerca de esta pronunciación por parte de la comunidad de desarrolladores. Al tratarse de páginas web de *Q&A (Questions and Answers)* no pueden ser incluidas como una fuente de información legible dentro de este documento.

2.7.3 Disposición de iOS

Con los dos puntos expuestos anteriormente, “Ejecución en *Background*” y “Capturar eventos de hardware”, el desarrollo de la aplicación en esta plataforma no es posible.

3. Capítulo III. Desarrollo de la aplicación móvil

3.1 Planteamiento de sistema de pánico UDLA

Dentro de este proyecto se busca desarrollar una aplicación que permita a un estudiante que se encuentre afectado por una emergencia dentro de los alrededores de la Universidad de las Américas enviar una notificación hacia el encargado de seguridad en la universidad por medio de un correo electrónico a su vez, esta alerta será enviada al Sistema Integrado de Emergencias ECU911. Con esto se espera acortar el tiempo de réplica en una emergencia para la comunidad UDLA. En la figura 20 se especifica como el usuario envía una alerta por la red móvil hasta el evaluador del ECU911 como al encargado de seguridad de la universidad el cual podrá ver en tiempo real lo sucedido en la ubicación.

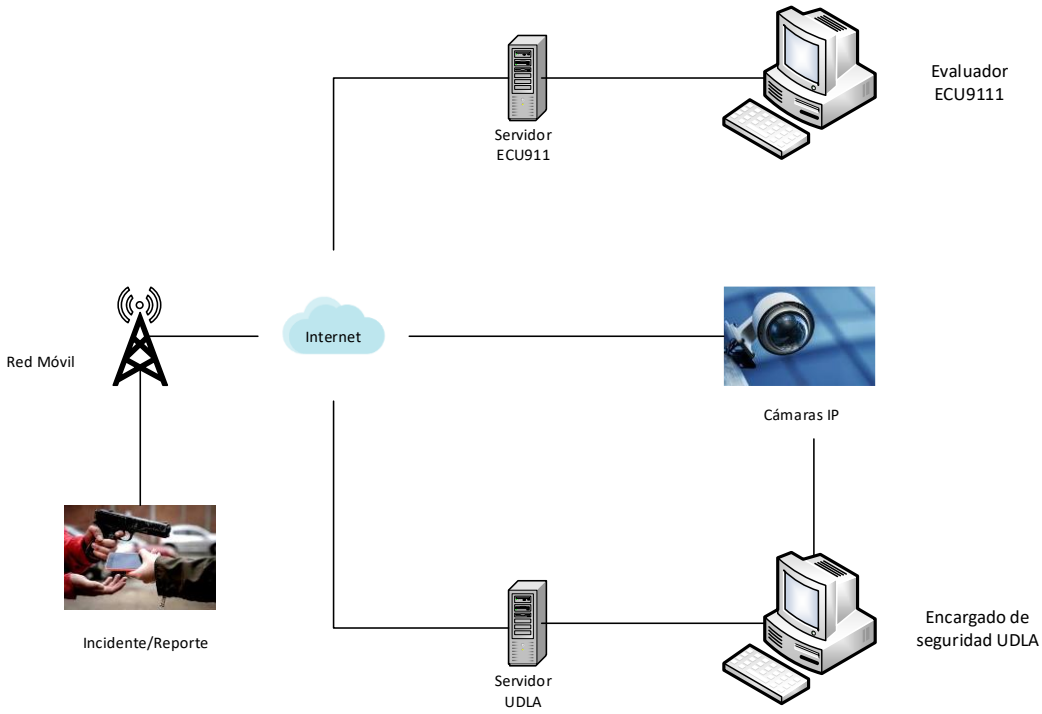


Figura 20. Planteamiento de la solución del sistema de pánico.

La aplicación debe ser capaz de ejecutarse en segundo plano, con una notificación persistente que permita al usuario saber que el servicio de acceso rápido se encuentra activo y al momento de presionar cuatro veces de forma consecutiva el botón de bloqueo envíe una notificación hacia el Sistema Integrado de Emergencias ECU911 con el fin de que este pueda responder de una forma más eficiente en caso de que la persona se encuentre ante una emergencia. De igual forma, se envíe un correo electrónico como notificación junto con un enlace que desplegará video en vivo si se encuentra dentro de la superficie de cobertura del clúster de cámaras antes explicado; hacia la persona encargada de la seguridad dentro de la Universidad de las Américas.

Dentro de los procesos de la aplicación presentada se tendrán como entradas: el evento generado al momento de presentarse una emergencia; como también el sistema de vigilancia que se encuentra constantemente haciendo *streaming* del video. Esta información será procesada dentro de los servidores

correspondientes y como producto final se obtendrá el reporte, tanto para el sistema integrado de emergencia ECU911 como para el encargado de recibir la notificación dentro de la Universidad de las Américas, como se muestra en la figura 21.

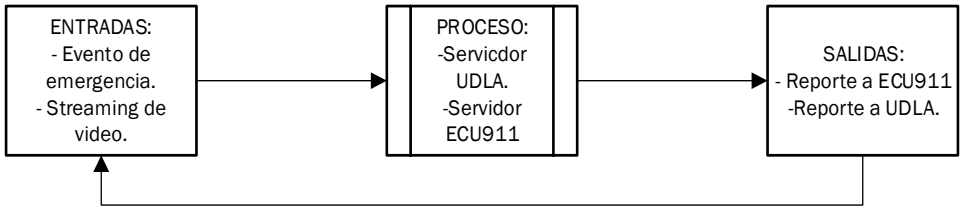


Figura 21. Diagrama de entradas, procesos y salidas.

3.2 Desarrollo de la aplicación

3.2.1 Especificaciones

La interfaz principal de la aplicación de servicio de emergencia constará de un interruptor que permitirá definir si se necesita o no mantener la aplicación corriendo en segundo plano a la espera de que se accione el evento de acceso rápido. Contará también con un botón el cual permitirá enviar la alerta directamente, sin necesidad de presionar cuatro veces el botón de pánico.

Dentro de *MyService,* se tendrá una notificación creada específicamente para mantener al servicio ejecutándose en segundo plano con la finalidad de que la clase *MyReceiver* esté a la escucha del evento de presionar el botón de *POWER*.

Una vez el *switch* cambia de estado (de *off* a *on*) se creará una notificación dentro del canal de notificaciones de Android, la misma permitirá a la aplicación mantenerse en segundo plano, a la escucha de que la acción sea completada y de esta forma enviar la notificación hacia los destinatarios.

Se establecerá un contador dentro del cual se validará si el botón de poder (*POWER)* fue presionado dentro de un intervalo de tiempo, una vez que este tiempo expire, el contador volverá a cero y esperará a que la acción sea

realizada para iniciar nuevamente la cuenta.

Para esta aplicación se creó un método dentro del *activity MyReceiver,* el cual es el encargado de interpretar las acciones de *SCREEN_ON* y *SCREEN_OFF* haciendo un ciclo de pulsación del botón de bloqueo, simulando la acción necesaria para que se realice el envío del mensaje.

Se propone dentro del proyecto que el usuario deba presionar cuatro veces el botón de poder (*POWER*) para enviar la alerta debido a que por lo general presionan este botón es usado de una a tres veces consecutivamente, ya sea para desbloquear o revisar notificaciones recibidas.

Una vez que el *receiver* obtiene la información que se ha capturado en el evento, se procederá al envío de la información hacia el servidor de la Universidad como al servidor para alertas por medio de *smartphone* que se encuentra en las instalaciones de ECU911.

Toda esta información, tanto de usuario como de especificaciones médicas como contacto de emergencia se empaquetará en un formato *JSON* para ser transportada hacia el destino.

Esta aplicación contará con un hilo de envío que permitirá la comunicación entre cliente servidor. De esta manera se permite a la aplicación hacer una petición de lectura de datos hacia el servidor establecido para la Universidad de las Américas.

El funcionamiento de la aplicación en un sentido general será explicado de forma detallada en la figura 22.

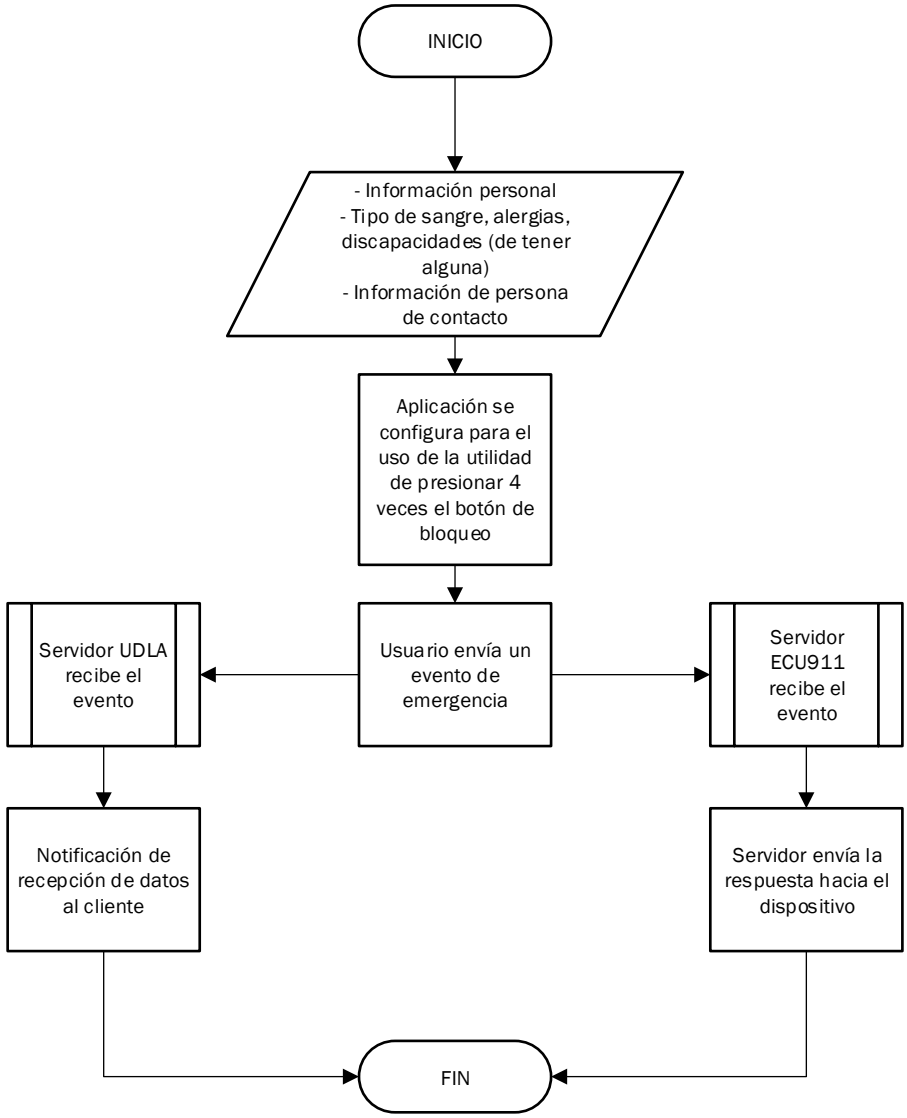


Figura 22. Diagrama de flujo explicativo sobre la aplicación.

3.2.2 Funcionamiento de la aplicación

Como se puede observar en la figura 23, al momento de instalar la aplicación se pedirá que se ingresen los siguientes datos de usuario: Nombre y apellido, se confirma si el ciudadano es ecuatoriano, un número de identificación, en nuestro caso cédula de identidad y número telefónico, con la finalidad de que estos datos sean enviados hacia el servidor de ECU911. En el caso de los datos de cédula y número telefónico se procederá a verificarlos usando

métodos de confirmación estándar.

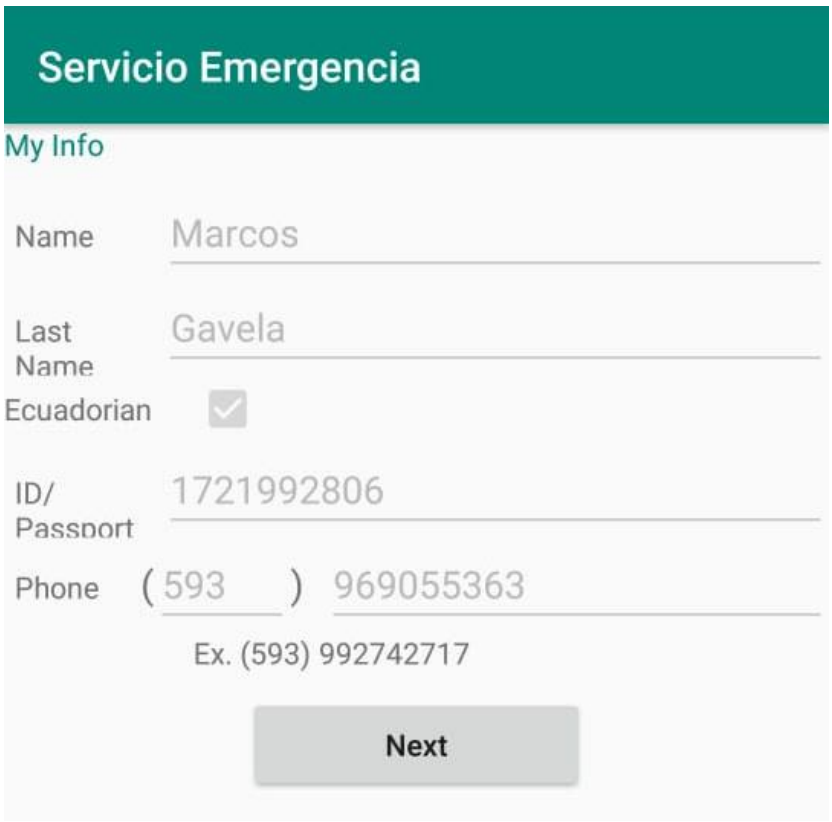


Figura 23. Ingreso de datos del usuario.

Como se puede observar en la figura 24, una vez ingresados estos datos se pedirá al usuario ingresar datos médicos de mayor relevancia, en este caso: tipo de sangre, si tiene alguna discapacidad (auditiva, visual, física) y se solicita una pequeña descripción sobre la misma.

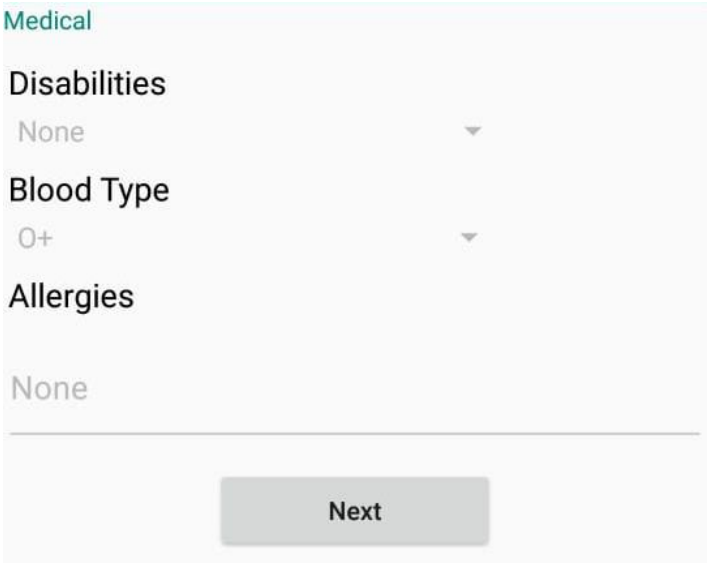


Figura 24. Ingreso de datos médicos generales.

Una vez ingresado los datos médicos, la aplicación requerirá información sobre un contacto de emergencia, esto con el fin de poder comunicarse con dicha persona si el usuario de la aplicación sufre algún percance. Como se muestra en la figura 25.

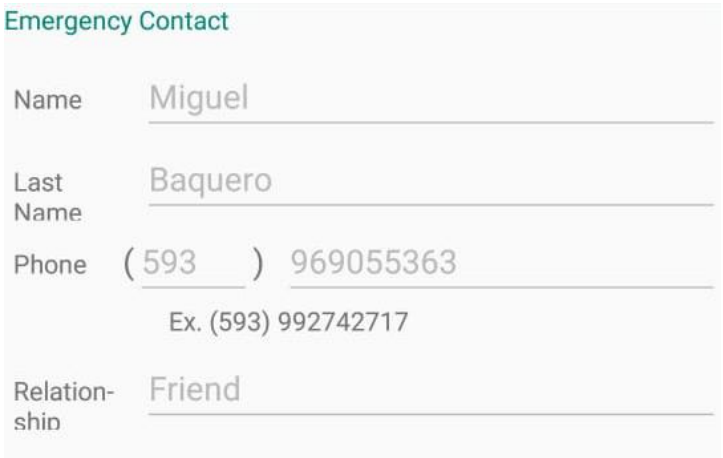


Figura 25. Ingreso de datos de contacto en caso de emergencia.

Esta información, no podrá ser editada, en caso de ser necesario un cambio en algún dato de la información se deberá desinstalar la aplicación y volverla a

descargar.

La figura 26 describe el ambiente que presenta la aplicación, donde el *activity* principal se encontrará una descripción sobre las funcionalidades que ofrece, junto con el *switch* que permite activar el servicio y un botón que se usará para enviar una notificación si se desea.

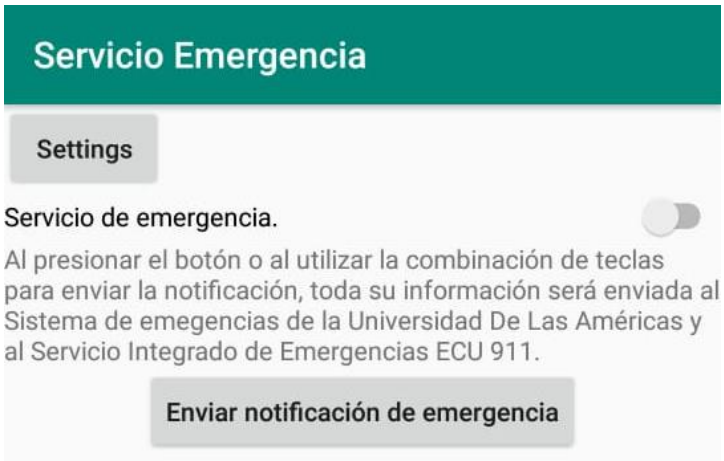


Figura 26. Presentación principal de la aplicación.

Como se puede observar en la figura 27, el usuario tendrá la opción de enviar la emergencia presionando el botón de envío o a su vez usando el *switch* activará la opción de acceso rápido el cual permitirá notificar hacia los servicios de emergencia presionando 4 veces el botón de bloqueo.

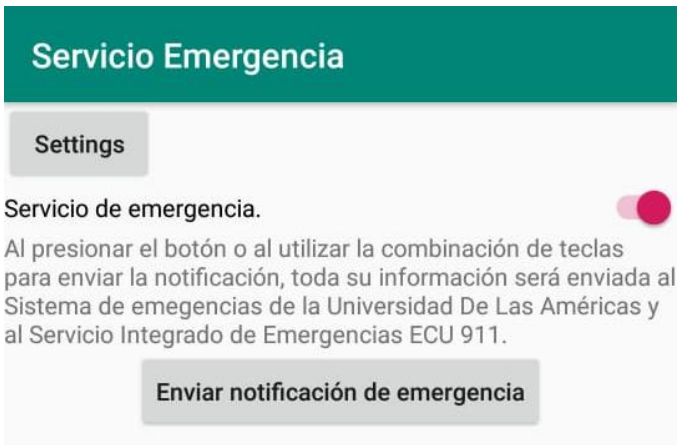


Figura 27. Presentación principal. *Switch* activo.

Cuando la opción de acceso rápido es seleccionada, se crea una notificación de la aplicación la cual permite que el servicio se ejecute en segundo plano (*backgound*) se puede verificar en la figura 28, que la aplicación se encuentra a la espera de que el evento sea disparado para realizar el envío de la notificación.

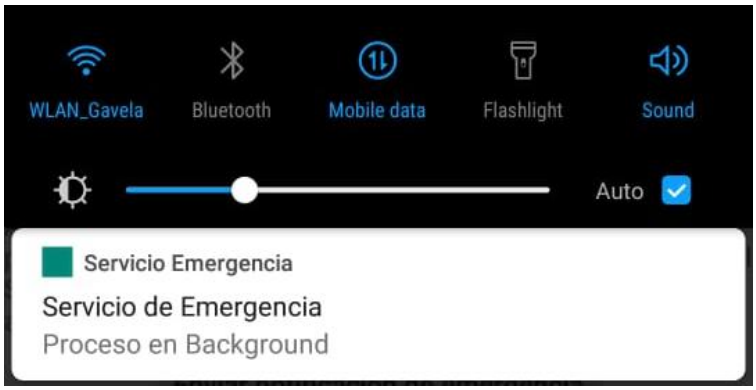


Figura 28. Notificación persistente. Canal de notificaciones.

En cuanto el servicio es usado, la aplicación se encargará de enviar las notificaciones a los siguientes servidores:

- Servidor ECU911
- Servidor UDLA

Se desplegará la ventana presentada en a figura 29, con la cual se puede confirmar que la emergencia fue enviada correctamente hacia el Servicio Integrado de Seguridad ECU911 y hacia el servidor UDLA.

Complementariamente una vez el servidor desarrollado para la recepción de notificaciones enviará un correo electrónico hacia el encargado de seguridad. Esta notificación será enviada a una persona designada para el monitoreo de la videovigilancia dentro de la Universidad de las Américas, como se puede apreciar en la figura 30.

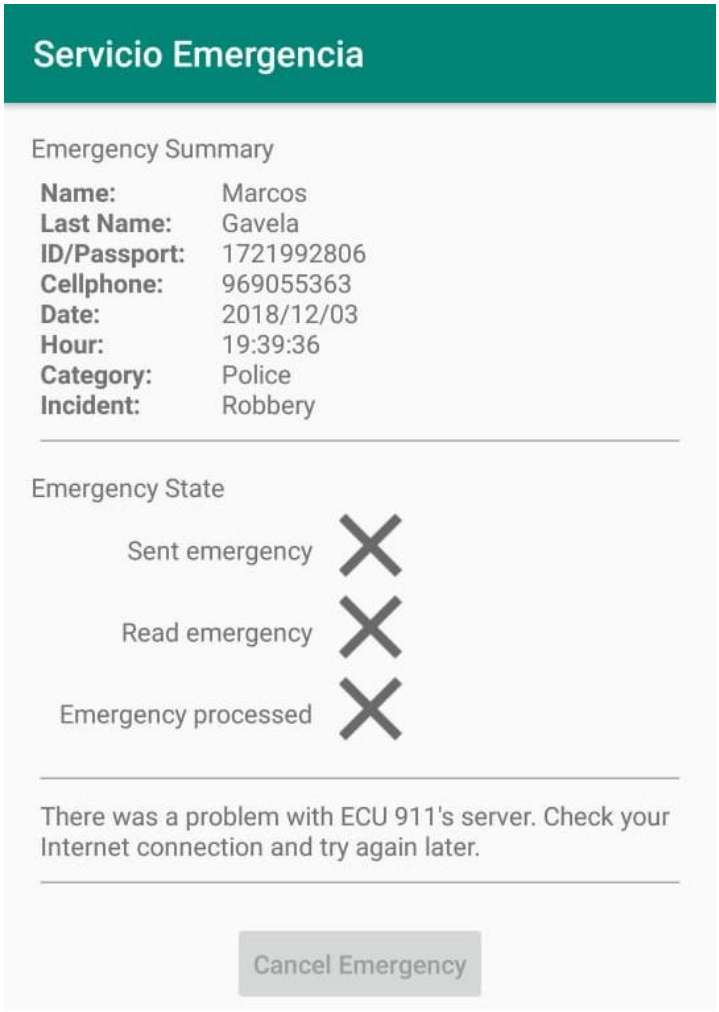


Figura 29. Presentación de resultado positivo de envío de notificación.

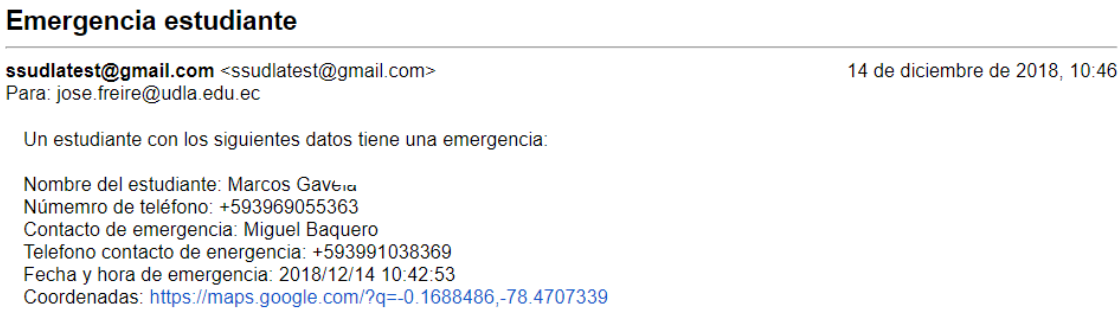


Figura 30. Contenido de correo electrónico enviado al encargado con información de la persona.

En la figura 31 se observa el correo contará con la información del estudiante, contacto de emergencia, la fecha/hora y la ubicación del evento; consecuentemente de encontrarse dentro del rango de cobertura se envía otro correo en el cual se encuentra un enlace para la visualización de la cámara.

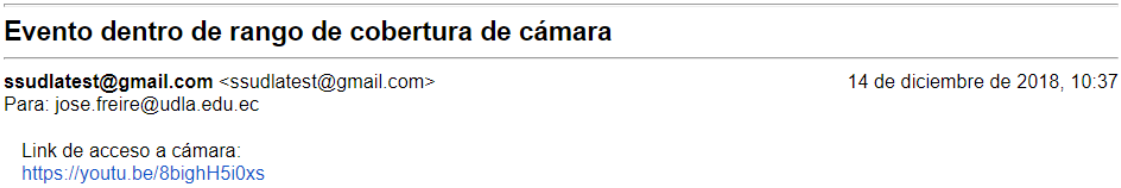


Figura 31. Contenido de correo electrónico enviado al encargado con el enlace de acceso a la cámara de videovigilancia.

Al momento en que se abre el enlace contenido dentro del segundo correo electrónico, se presenta la imagen en vivo de la o de las cámaras que se encuentren más cercanas al evento, como se puede observar en la figura 32.

Figura 32. Imagen multimedia obtenida del prototipo del clúster de videovigilancia.

3.4 Implementación de aplicación de servicios UDLA

3.4.1 Funcionamiento de la aplicación proveedora de servicios

Es importante definir que la aplicación proveedora de servicios es la que se encuentra alojada dentro un servidor de la UDLA.

La aplicación se comunica por medio de Hilos o *Threads* de Java. Un *Thread* o Hilo es un proceso ligero. A diferencia de otros lenguajes de programación, Java cuenta con soporte para una comunicación Multi-hilo o *Multi-threaded*, la cual contiene uno o más procesos que pueden ejecutarse simultáneamente. Adicionalmente, para establecer la comunicación se debe implementar una Interfaz en el ciclo de vida *Runnable,* según se explica en la figura 33, dicha interfaz debe contar con herencia desde la clase de Java *Thread* (Tuli, 2018).

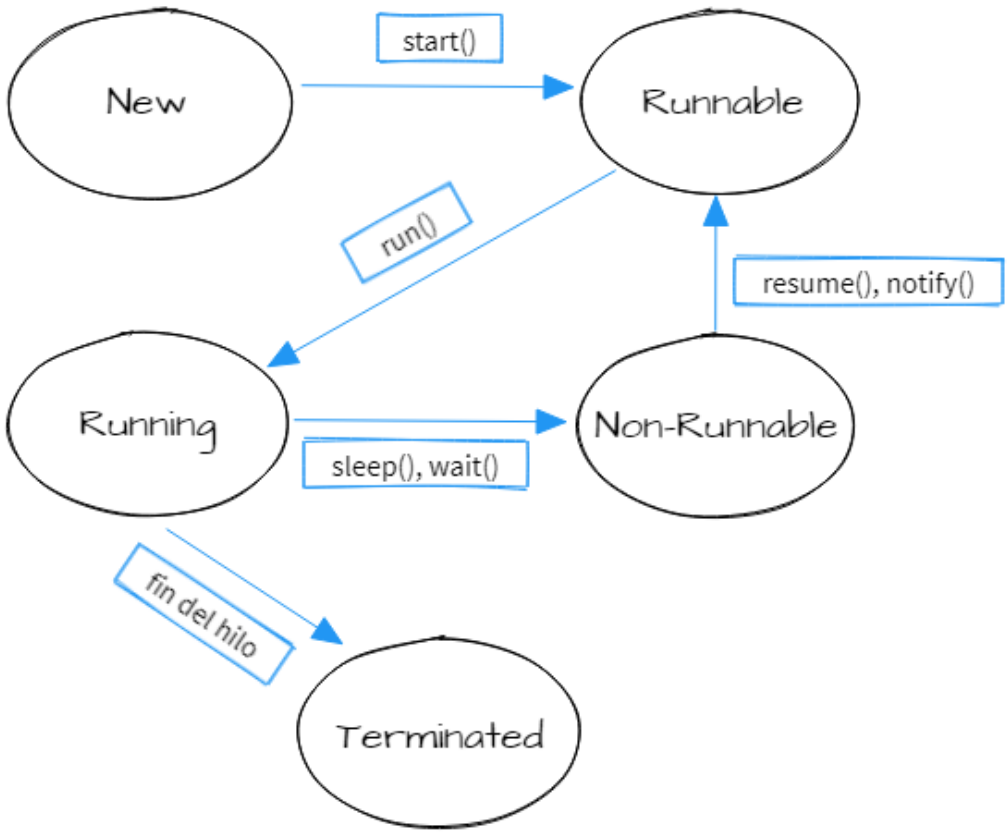


Figura 33. Descripción de los ciclos de vida de Hilos en Java.

Como se había destacado previamente, este tipo de comunicación tiene la característica de soportar varias conexiones como procesos simultáneos (multi-hilo), por lo cual, no existe problema alguno en el momento que se realicen varias conexiones al mismo tiempo de parte de las aplicaciones móviles hacia la aplicación alojada en el servidor, así como en el ejemplo de la figura 34.

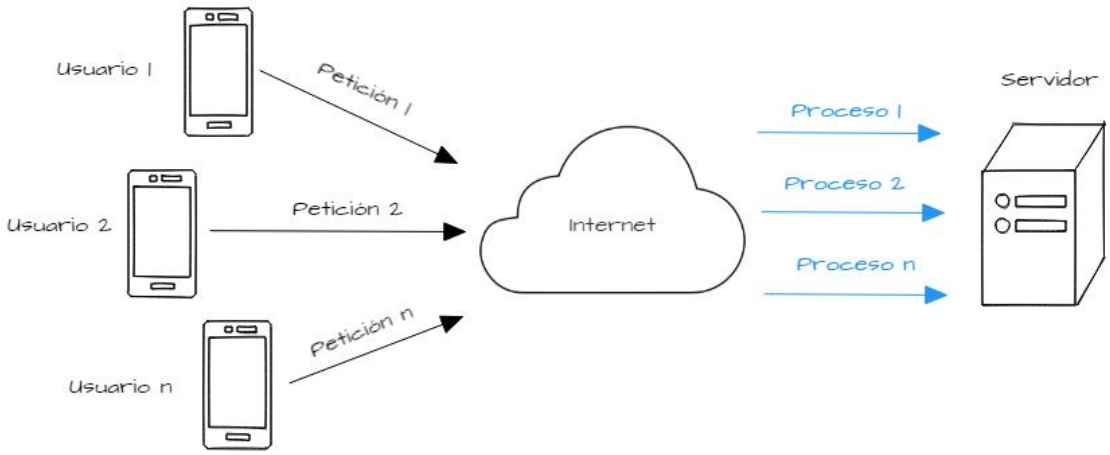


Figura 34. Peticiones de conexión simultáneas de los dispositivos móviles que crean un proceso independiente para cada uno de ellos.

Para hacer posible la comunicación es necesario establecer un canal de conexión entre el servidor y el cliente. Con el diseño de la aplicación realizada se consideró un socket. Los sockets fijan una comunicación orientada a la conexión, *TCP (Transmission Control Protocol)*. *UDP (User Datagram Protocol)*, al contrario de *TCP*, no necesita una conexión permanente para que la comunicación sea efectiva.

En la aplicación, se definió el puerto 24666 para establecer la comunicación entre Hilos de Java. En la figura 35, se muestra una gráfica del diagrama de conexión mediante el puerto.

Solamente es necesario definir un socket con un solo puerto, debido a que la aplicación móvil enviará información por el canal de comunicaciones y el servidor solamente recibirá los datos en formato JSON.

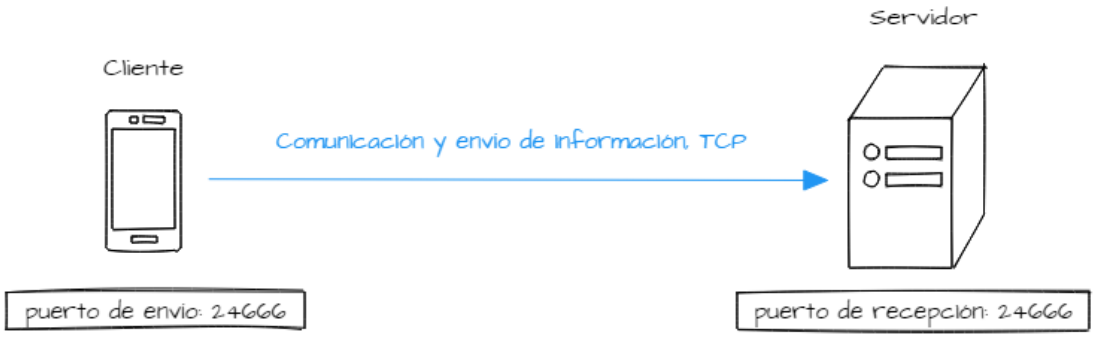


Figura 35. Socket y puerto definido para establecer la comunicación *TCP* y hacer posible el envío de la información en un modelo Cliente - Servidor.

El propósito de establecer el canal de comunicación radica en enviar la siguiente información del usuario desde la aplicación móvil en una cadena datos en formato JSON:

- Nombre
- Apellido
- Nacionalidad ecuatoriana (*true* – *false*)
- ID / Pasaporte
- Número de teléfono
- Discapacidades
- Tipo de sangre
- Alergias
- Nombre de contacto de emergencia
- Apellido de contacto de emergencia
- Número de teléfono del contacto de emergencia
- Relación del contacto de emergencia que tiene con el usuario
- Latitud (coordenadas)
- Longitud (coordenadas)
- IMEI del teléfono

Los datos enlistados anteriormente se encuentran en conformidad con la

información que solicita el Servicio Integrado de Seguridad ECU 911. Esto se debe a que existe una convergencia en la aplicación móvil del usuario para notificar a dicho servicio integrado y también a la aplicación proveedora de servicios de manera simultánea.

De una manera concreta, el proceso de comunicación entre la aplicación proveedora de servicios y la aplicación móvil del usuario cumple el siguiente esquema:

1. Una vez que el servidor se encuentra preparado, la aplicación se encuentra esperando una conexión.
2. En el momento que un usuario decide notificar un evento a hacia el servidor, la conexión con el *socket* es iniciada por medio del puerto especificado.
3. Una vez que la conexión es exitosa, la comunicación es aceptada por el servidor y se realiza el envío de la información del usuario en formato JSON por medio de un flujo de datos UTF desde la aplicación móvil del usuario hacia el servidor. Esto es posible gracias al método de Java *DataOutputStream.writeUTF().*
4. El servidor, al tener una comunicación entrante, inicia su método lectura de datos UTF a través de *DataInputStream.readUTF()*. Finalmente, todos los datos del usuario en formato JSON son recibidos por el servidor.
5. La conexión establecida por medio del *socket* finaliza una vez que el usuario desactiva el servicio en segundo plano de la aplicación y cierra por completo la misma.

Es importante considerar que la aplicación móvil del usuario también cuenta con el mismo método de comunicación de Hilos de Java, por lo cual, dentro de la aplicación se definió un servicio con *Thread* y una interfaz, de la misma manera en que se desarrolló dentro de la aplicación proveedora de servicios.

3.4.2 Clases de Java para comunicación entre usuario y servidor

Tanto en la aplicación del servidor y el usuario es necesario implementar las

siguientes clases:

- Interfaz de lectura de datos: clase abstracta donde se establecen y se aceptan las conexiones entre Hilos de Java. El ciclo de vida del *Thread* se encuentra en modo *Runnable* para poder aceptar una conexión cuando el servidor se encuentre en espera de esta. Debe ser implementado, tanto en el servidor como en el usuario.

- Conexión entre Hilos: clase donde se fija el puerto o los puertos a ser utilizados en la conexión de sockets, donde se llama a un proceso de envío de datos cuando el usuario desea reportar un evento, esto para el caso de la aplicación móvil o se da el inicio a un proceso de recepción y lectura de datos cuando el servidor detecta un flujo de información de entrada, en el caso del servidor. Se implementa en el servidor y en la aplicación del usuario, para este último como un servicio activo.

- Hilo de envío de datos: clase que es invocada cuando el usuario activa el envío de una emergencia. Dicha clase inicia un proceso en el cual se establece un envío de flujo de datos por medio del *socket* en modo de espera. La clase se implementa en la aplicación del usuario.

- Hilo de recepción de datos: se da inicio a esta clase una vez que el servidor detecta una entrada de flujo de datos por medio de la conexión activa. Este flujo de datos contiene toda la información que ha sido enviada desde el dispositivo móvil del usuario en formato JSON, por lo tanto, en esta clase, también se realiza la interpretación de datos JSON para su posterior procesamiento en caso de que las condiciones establecidas sean cumplidas, para posteriormente activar el clúster de cámaras y enviar una notificación por correo electrónico.

3.4.3 Notificación por correo electrónico

El servidor, al haber recibido todos los datos del usuario de la aplicación móvil, procede a enviar un correo electrónico a una dirección en específico para alertar a las personas encargadas.

Java tiene la ventaja de tener una librería externa que debe ser importada, se trata de *JavaxMail*, la cual tiene compatibilidad total con los servidores de correo electrónico de *Gmail*.

Al ser un servicio de notificaciones por correo electrónico, no existe la necesidad de implementar una bandeja de entrada con protocolos como *POP3 (Post Office Protocol 3)* o *IMAP (Internet Message Access Protocol)*. Ambos protocolos mencionados realizan la tarea de otorgar el acceso al usuario de mensajes que se encuentran en un servidor de correo electrónico.

Solamente se estableció el envío de correos electrónicos con *SMTP* (*Simple Mail Transfer Protocol*) a través de puerto 587. En la figura 36 se observa un diagrama de la comunicación que realiza el servidor UDLA con los servidores de correo electrónico de *Gmail* para entregar las notificaciones a la cuenta destinataria.

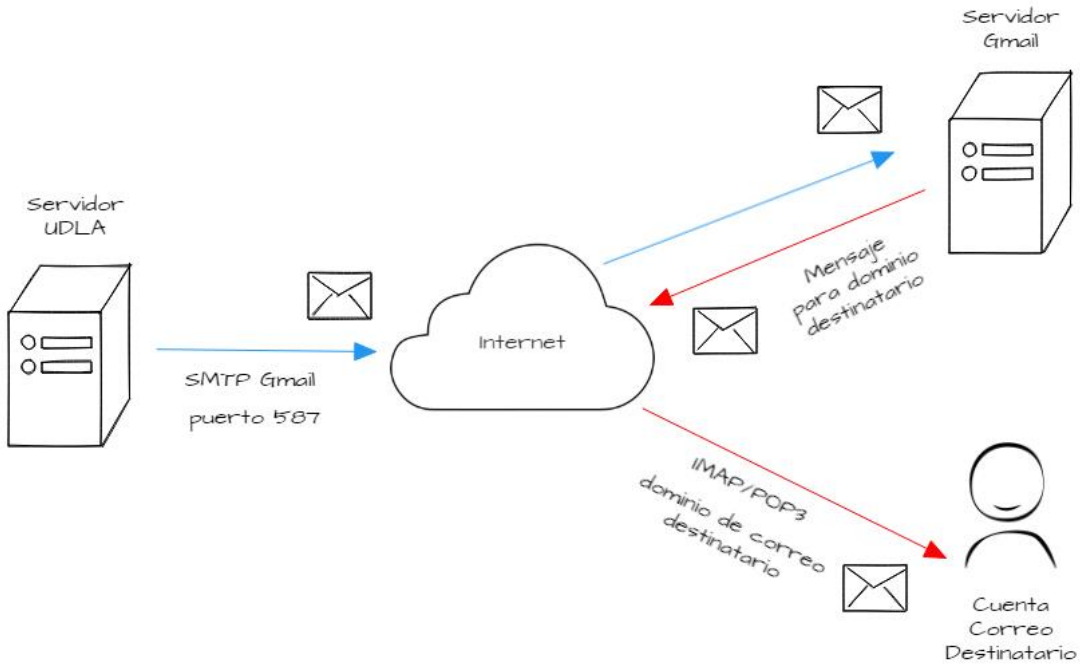


Figura 36. Comunicación del servidor de la UDLA con los servicios de *Gmail* para enviar correos electrónicos.

3.5 Consideraciones para implantación del sistema de videovigilancia

Según el requerimiento del proyecto, el sistema debe contar con protocolos de comunicación IP. Se puede utilizar *NVR (Network Video Recorders)*, debido a su facilidad para administrar, gestionar y almacenar datos de videovigilancia.

Un aspecto para tomar en cuenta es que, los *NVRs* deben ser compatibles con las cámaras IP que se elijan, puesto que cada uno de ellos tienen establecidos ciertos protocolos de comunicación implementados, como RS-485. Para el sistema de videovigilancia es necesario el protocolo de comunicación *Ethernet* con conectores RJ-45.

También se puede utilizar cámaras IP conectadas directamente a un *Switch* o *Router*, pero el almacenamiento de la información de video requiere de un servidor o un software de terceros. Las cámaras elegidas dentro de este proyecto cuentan con servicio de *FTP (File Transfer Protocol)* para guardar la información multimedia del video en una unidad de almacenamiento informático de preferencia sin necesidad de software adicional.

Los *NVRs* (*Network Video Recorder*) actuales cuentan con slots de inserción de *HDD (Hard Disk Drive)* y/o *SSD (Solid State Drive)*, esto hace versátil la forma de almacenar archivos multimedia de video.

Las cámaras deben contar con una interfaz web para que las personas encargadas tengan acceso desde su navegador favorito. Además, preferentemente, el sistema debe ser visualizable desde cualquier red, por lo que es recomendable que se encuentre con una IP estática de carácter público. Es posible visualizar en la figura 37 un esquema de la estructura del sistema de vigilancia.

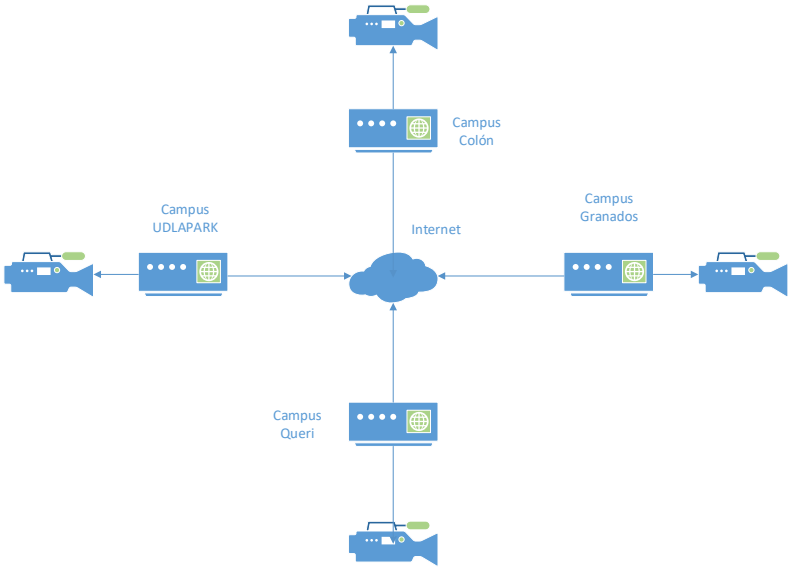


Figura 37. Diagrama de circuito de videovigilancia.

En caso de no contar con una IP pública, como una alternativa, se puede designar personal dentro de cada Campus de la Universidad para que el sistema sea visualizado a nivel LAN.

Las cámaras, al encontrarse en un ambiente de red, son vulnerables a ser atacadas. Como una medida de seguridad, se puede activar la autenticación por medio de usuario y contraseña en la interfaz web para la visualización de las cámaras.

3.5.1 Consideraciones técnicas para la instalación de las cámaras

Tomando en cuenta que el medio de transmisión a ser utilizado es un cable Ethernet *UTP (Unshielded Twisted Pair)*, *STP (Shielded Twisted Pair)* o *FTP (Foiled Twisted Pair)*, la distancia máxima para realizar un enlace a lo largo de todo el canal es de 100 metros (Gobierno del Estado de Tabasco, s.f.).

Si la instalación no tiene más de 100 metros en todo el enlace, no debería suponer problema alguno.

Pero también se debe tener en cuenta que las cámaras deben contar con una

fuente de energía para su funcionamiento. Para esto se tienen 3 soluciones:

- Utilizar cámaras con soporte *PoE (Power over Ethernet)*.
- Instalar cable de red y una toma eléctrica para cada cámara.
- Utilizar convertidores de medios (*Media Converters*) con fibra óptica y una toma eléctrica para cada cámara.

Power over Ethernet es una tecnología que permite transmitir energía a través del cable *Ethernet* y así, alimentar los equipos finales (televisores, cámaras de video, teléfonos VoIP, entre otros). Pero, nuevamente, la limitación radica en el medio de transmisión que tiene una distancia máxima de 100 metros. Esta solución es ideal para cámaras *PoE* que tienen una distancia al *NVR PoE o Switch PoE* menor a la especificada anteriormente.

Es necesario aclarar que todos los elementos de red deben ser compatibles con la tecnología *PoE* (cable de red, cámara IP y *NVR /Switch*) para el correcto funcionamiento. En caso de que la cámara IP y el *NVR/Switch* no cuenten con tecnología PoE, se pueden incluir inyectores PoE. En la figura 38 se coloca un diagrama de conexión de las cámaras con la tecnología PoE descrita anteriormente.

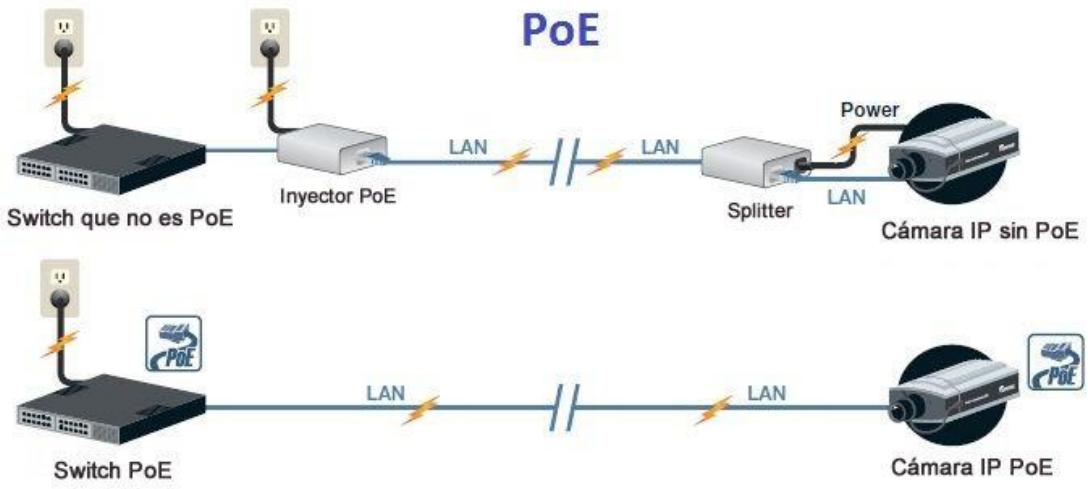


Figura 38. Diagrama de videovigilancia con *PoE*.

Tomado de (TECNOSeguro, s.f.)

Del mismo modo, el cable de red o *Ethernet* debe ser para exteriores (chaqueta con aislamiento y construcción para ambientes exteriores) a pesar de que sea instalado por medio de una tubería, se pueden presentar filtraciones de líquidos o minerales que pueden perjudicar al rendimiento del cable.

La segunda solución propuesta tiene un factor importante a tomar en cuenta, las interferencias que sufre un cable de red. Al colocar un cable de red junto a una instalación eléctrica, la interferencia electromagnética es inminente. Para reducir la interferencia producida por un cable eléctrico, es altamente recomendable el uso de un cable de red de tipo *STP*.

Al igual que en la primera solución propuesta, este método está limitado a una distancia máxima de 100 metros y el cable de red debe tener una construcción para exteriores.

La tercera y última solución es la que se destaca cuando se trata de alcanzar grandes distancias.

Los convertidores de medios son dispositivos simples usados para conectar dos dispositivos de red que no son totalmente compatibles, debido a las diferentes velocidades de transmisión, tipo de medio (fibra, ethernet o coaxial). Típicamente son utilizados para insertar segmentos de fibra óptica en redes de cobre (Black Box Corporation, s.f.).

Para los enlaces mayores a 100 metros, se puede utilizar un segmento de fibra óptica, 2 convertidores de medios, 2 cables de red y los dispositivos de red a ser enlazados (cámara IP y *NVR /Switch)*. El diagrama del funcionamiento de un convertidor de medios se muestra en la figura 39.

Al instalar la fibra óptica junto al cable eléctrico, no existe problema alguno, puesto que la fibra óptica es inmune a la interferencia electromagnética (Gobierno del Estado de Tabasco, s.f.).

La distancia máxima para realizar el enlace depende del convertidor de medios a ser utilizado.

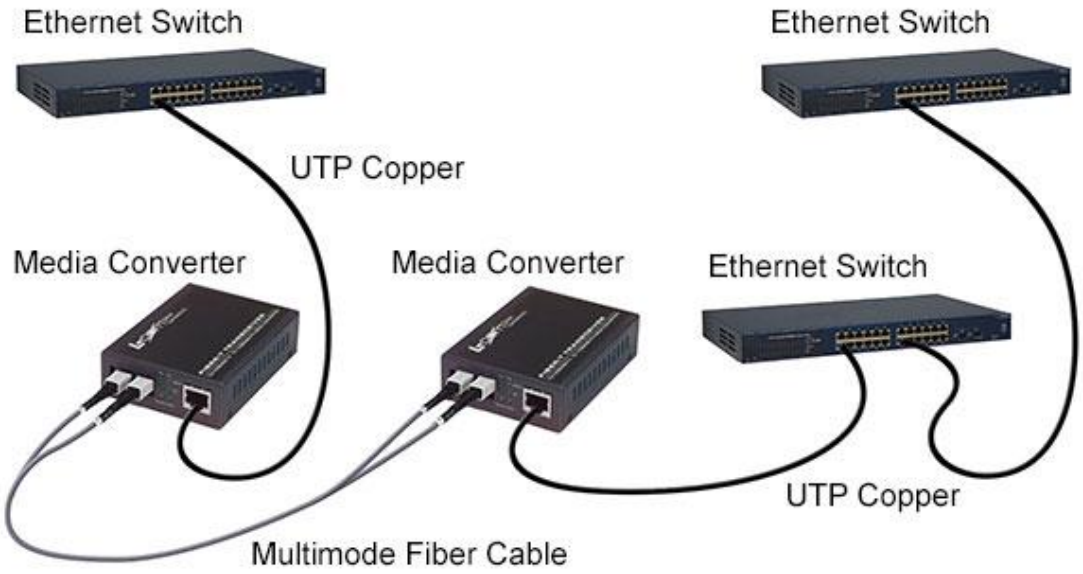


Figura 39. Diagrama de red con convertidores de medios.

Tomado de (Infinite Electronics International, Inc, s.f.).

Finalmente, se debe establecer el tipo de cámara que se ajuste más para la finalidad de la alarma de pánico.

Debido a que las cámaras se encontrarán en los exteriores de los Campus Universitarios, las cámaras deben contar con características para ambientes exteriores, esto significa que deben tener un alto nivel de protección contra el ingreso del polvo y del agua. No es necesaria la protección contra la inmersión, porque la cámara solamente quedará expuesta a la lluvia y al viento.

El nivel de protección de ingreso *(IP: Ingress Protection),* es un estándar internacional *EN 60529 (British BS EN 60529:1992, European IEC 60509:1989)* (Rainford Solutions Ltd, 2017). Este estándar da un número dentro de la escala *IP* para la protección de dispositivos electrónicos contra polvo y agua. El formato se expresa como *IPXY*, donde “X” es el nivel de protección contra polvo y “Y” es el índice de protección contra líquidos. En el anexo 2 se muestra información más específica acerca de este estándar. El nivel de protección que más se acomoda a las necesidades del sistema de videovigilancia requerido es IP66.

Además de enfrentar las condiciones ambientales, las cámaras deben tener la capacidad de adaptabilidad a la luz. Actualmente, la UDLA tiene un horario regular de acogida a los estudiantes de 7:00 a 22:00, de lunes a viernes. Para facilitar el trabajo de los operadores o espectadores de las cámaras, cuentan con funciones como Visión Nocturna o Día/Noche.

Clasificadas por su funcionalidad, existen 2 tipos de cámaras; estáticas o fijas y *PTZ (Pant – Tilt – Zoom)*.

Las cámaras fijas son ideales para ambientes interiores como corredores, entradas, salas o lugares cerrados, donde no se requiere un ajuste del campo de visión. Por el contrario, las cámaras *PTZ* son muy adaptables en entornos abiertos como lugares públicos, parqueaderos, o calles muy afluidas donde se desea captar detalles que distingan a la gente (GVS Colombia, s.f.).

Para el caso de uso de este proyecto, el tipo de cámara que se ajusta a las necesidades es una cámara *PTZ.*

Con todas estas consideraciones propuestas, la figura 40 muestra un diagrama de red de un solo segmento del circuito de videovigilancia con tecnología *PoE*.

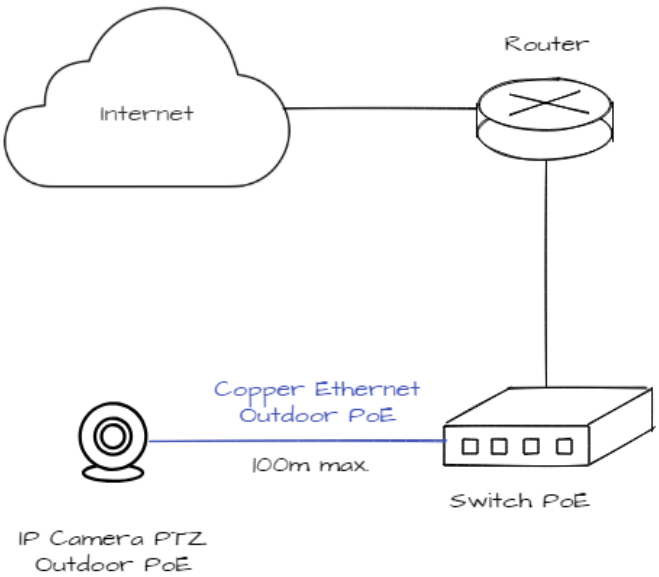


Figura 40. Diagrama de videovigilancia con tecnología *PoE*.

También, la figura 41 expone el diagrama de red cuando se utiliza convertidor de medios de medios *(media converter).*

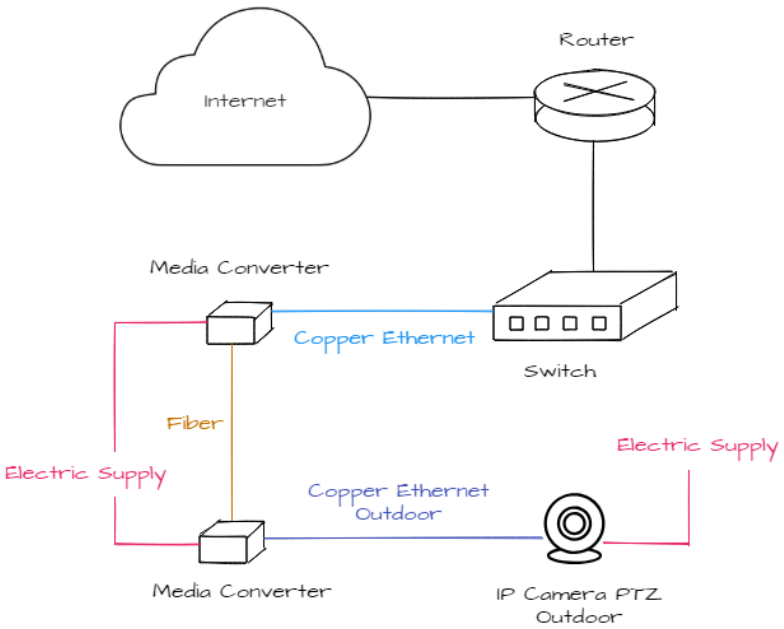


Figura 41. Diagrama videovigilancia con convertidor de medios.

4. Capítulo IV. Implementación prototipo de sistema de pánico

4.1 Consideraciones para la implementación del servidor para la UDLA

Para el adecuado funcionamiento de la aplicación se deben tener en cuenta los siguientes parámetros:

- IP pública.
- Contar con un sistema operativo capaz de ejecutar IDE's de Java, de preferencia, un equipo con características de servidor (seguridad de red, software, entre otros).
- J.R.E. (*Java Runtime Enviorment*).

Por un lado, la IP pública permitirá a la aplicación ser contactada desde cualquier parte de la ciudad para la recepción de mensajes de emergencia. A

su vez, el entorno de Java permite al servidor ejecutar la aplicación que recibirá las cadenas de información, con secuentemente se produce el envío de correos electrónicos y presentación de video en vivo, siempre y cuando la emergencia sea reportada dentro del área de cobertura de las cámaras.

Dentro del levantamiento del servidor de recepción de eventos de emergencia para la Universidad de las Américas, se creó una aplicación *Java,* esta se encargará de recibir y leer la información enviada desde la aplicación para posteriormente, usando un algoritmo de ubicación, desplegará la imagen en vivo dependiendo del sitio en el cual sucedió el evento y que, además, enviará una notificación sobre lo sucedido hacia el encargado mediante un correo electrónico.

Se utilizará el servidor “ServidorJSON”, el cual receptará varias cadenas de texto en las cuales constará la información que el usuario ingresó al momento de registrar la aplicación y enviadas usando formato JSON.

El hilo de lectura será el encargado de crear un canal de comunicación entre el dispositivo móvil (*smartphone*) y el servidor para enviar las cadenas de datos con la información del usuario.

Se ha definido dentro del servidor que la comunicación se efectuará a través de los puertos 24666 y 24667, para el Hilo de lectura y el Hilo de escritura respectivamente, esto se debe a que estos puertos no son reservados para ningún servicio. Dichos puertos deben estar definidos en una regla dentro del *firewall* del servidor para que la comunicación sea exitosa.

Como se muestra en la figura 42; en cuanto el servidor es iniciado, estará a la espera de que se realice la comunicación con el dispositivo; de esta forma se despliega una ventana con el mensaje “Esperando conexión”.

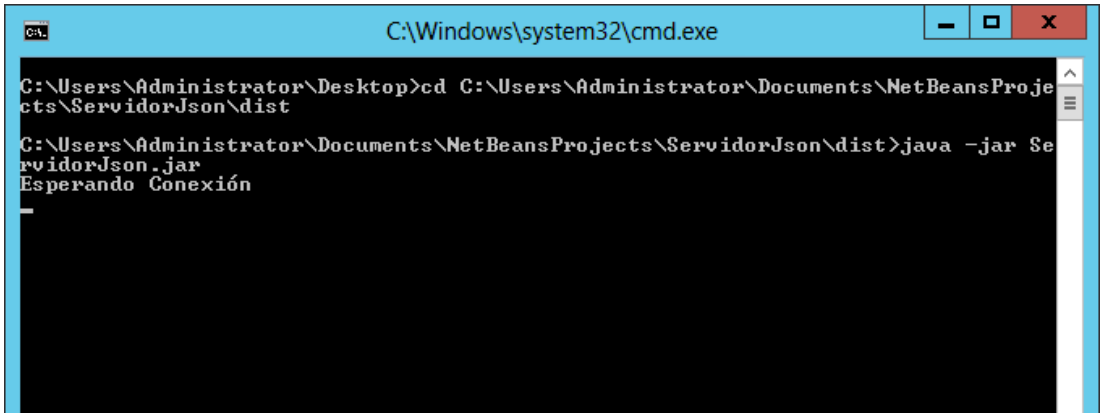


Figura 42. Ventana del servidor a espera de conexión.

Cuando el usuario se encuentre dentro de la página principal de la aplicación, el servidor detectará una conexión al dispositivo usando el hilo de lectura y brindará la información sobre la dirección IP pública solicitante desplegando el siguiente mensaje en pantalla, evidenciado en la figura 43.

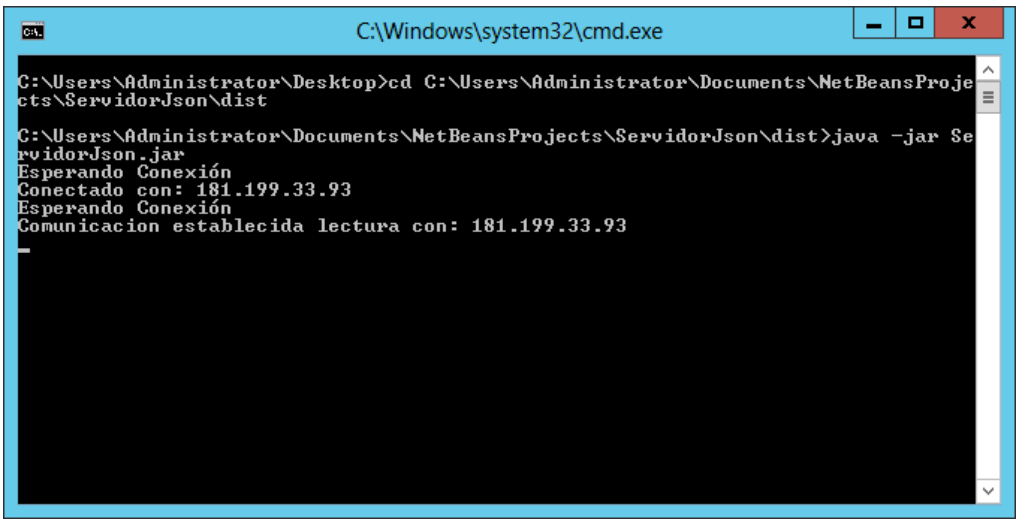


Figura 43. Conexión establecida con éxito.

Una vez el usuario presione el botón “Enviar notificación” la aplicación se encargará de enviar las solicitudes hacia los servidores, en primera instancia hacia el servidor de smartphone ubicado en las instalaciones de ECU911 y posteriormente hacia el servidor UDLA, este servidor desplegará la toda la

información del evento dentro de una ventana en *cmd (Command prompt)* en Windows. Como se presenta en la figura 44.

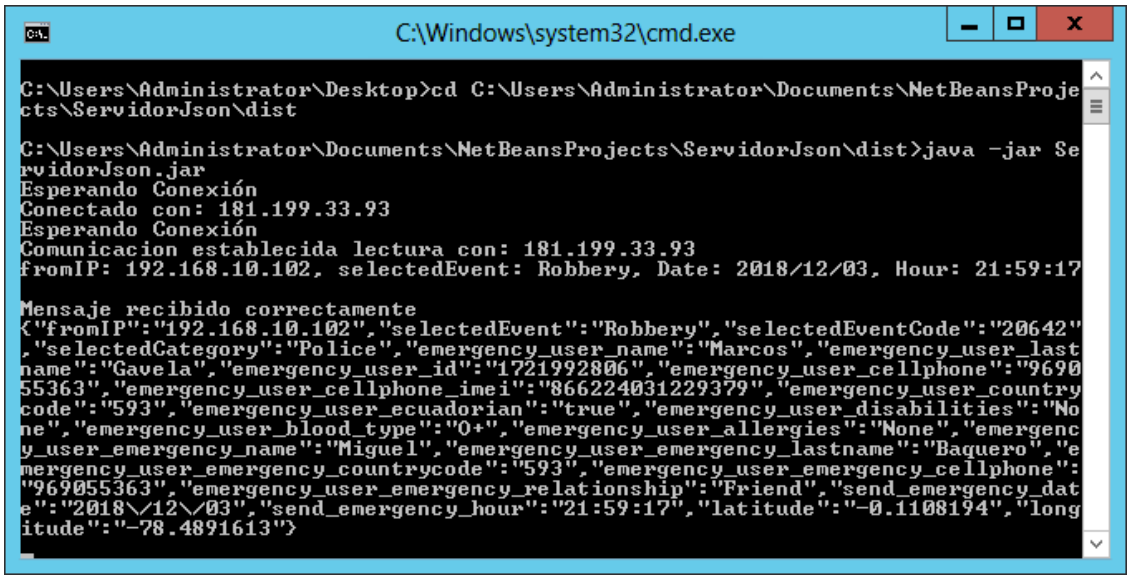


Figura 44. Recepción exitosa de datos.

Al mismo tiempo el servidor será el encargado de enviar un correo junto con el enlace de visualización con el *straming* en tiempo real de lo que está sucediendo. La imagen presentada dependerá de la ubicación del evento. Evidenciado en la figura 45.

Figura 45. Video en vivo del incidente.

Como consecuencia la imagen que proviene de la ubicación es desplegada en el servidor, este se encargará de dos mensajes mediante correo electrónico hacia el encargado de recibirlo. El primero con la información personal de la persona que realizó el envío de la información y el segundo con un enlace hacia la cámara en caso de encontrarse dentro del área de cobertura.

4.2 Sistema de videovigilancia

Al tratarse de un prototipo, no se llevó a cabo la implementación real del sistema de videovigilancia, debido que el acceso a una cámara del sistema de videovigilancia se efectúa con una dirección *IP* y un puerto en específico que se coloca en un navegador de red. Por ejemplo, suponiendo que una cámara tiene definida la dirección 192.168.1.1 y el puerto 8182, en el navegador de red se coloca la dirección con el formato “192.168.1.1:8182”. Del mismo modo, se pueden llevar a cabo las pruebas del prototipo con una cámara incorporada en un teléfono móvil o en una computadora, junto con una aplicación que facilite una dirección *IP* o un *URL* para el acceso al contenido multimedia del video mediante cualquier navegador.

El servicio que ofrece Google a sus usuarios mediante *YouTube Live* fue considerado como la mejor opción para realizar la simulación, como se muestra en la figura 46 filma video y lo transmite en tiempo real. Se toma el *URL* generado por la transmisión en vivo de *YouTube* y se la coloca en el algoritmo del Servidor de la UDLA que se encarga de enviar el correo electrónico a una persona encargada del monitoreo de la actividad. También, el dispositivo móvil o computadora que se encuentre realizando la transmisión del video debe encontrarse en la ubicación geográfica definida para una cámara perteneciente al sistema de videovigilancia.

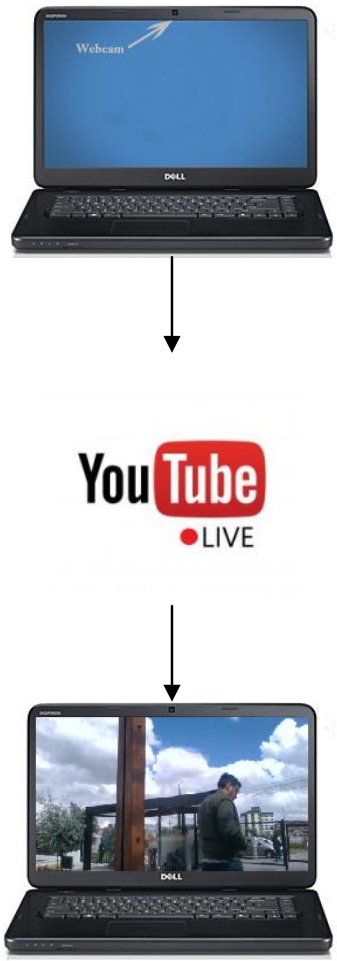


Figura 46. Explicación del uso de *webcam* del pc para el prototipo.

5. Capítulo V. Pruebas y análisis de costos

5.1 Pruebas en los Campus

Las pruebas se realizaron el día 19 de diciembre de 2018 en tres campus de la Universidad de las Américas:

- Queri
- Granados
- UDLAPARK

Se realizaron eventos de emergencia alrededor de los campus mencionados para comprobar que el servidor pueda recibir múltiples peticiones y sea capaz de responderlas.

El dispositivo usado fue un teléfono inteligente Samsung Galaxy J5 Pro, con la aplicación instalada y los siguientes datos:

- Nombre: Miguel
- Apellido: Baquero
- Nacionalidad: ecuatoriana
- Cédula de identidad: 1718164765
- Número de teléfono: + 593 983511126
- Discapacidades: ninguna
- Tipo de sangre: O+
- Alergias: Ninguna
- Nombre de contacto de emergencia: Mariano
- Apellido de contacto de emergencia: Tello
- Número de teléfono del contacto de emergencia: + 593 999681473
- Relación del contacto de emergencia que tiene con el usuario: abuelo

En total se realizaron veinte pruebas en los campus antes mencionados dando como resultados dieciocho pruebas correctas y dos pruebas con resultados negativos que serán explicados a continuación. En la tabla 2 se coloca la información obtenida de las pruebas junto con la fecha y un enlace para ver la localización del incidente. Los datos se obtuvieron del servidor de la UDLA, el cual registra las notificaciones generadas por el usuario de la aplicación móvil.

Tabla 2.

Pruebas realizadas en el prototipo del sistema de alarma de pánico.

#	Campus	Exitosa	Fecha/Hora	Ubicación	Observaciones
1	Queri	Si	12/19/2018 15:08	-0.1694631,-78.4710829	
2	Queri	No	12/19/2018 15:12	-0.1680193,-78.4702626	Ubicación errónea.
3	Queri	Si	12/19/2018 15:13	-0.1681101,-78.4702396	

4	Queri	Si	12/19/2018 15:14	-0.1680277,-78.470348	
5	Queri	Si	12/19/2018 15:16	-0.1684207,-78.4696808	
6	Queri	Si	12/19/2018 15:19	-0.1674689,-78.4708798	
7	Queri	Si	12/19/2018 15:20	-0.1675257,-78.4710067	
8	Granados	Si	12/19/2018 15:44	-0.1681628,-78.4728565	
9	Granados	Si	12/19/2018 15:48	-0.1667074,-78.4729565	
10	Granados	Si	12/19/2018 15:51	-0.1650741,-78.4726948	
11	Granados	Si	12/19/2018 15:52	-0.1651692,-78.4718446	
12	Granados	Si	12/19/2018 15:55	-0.1663824,-78.4719202	
13	Granados	Si	12/19/2018 15:57	-0.1679511,-78.4720165	
14	UDLAPARK	Si	12/19/2018 17:13	-0.1623699,-78.4616415	
15	UDLAPARK	Si	12/19/2018 17:15	-0.1623184,-78.4617591	
16	UDLAPARK	Si	12/19/2018 17:17	-0.1630898,-78.4626836	
17	UDLAPARK	Si	12/19/2018 17:18	-0.1632715,-78.4629283	
18	UDLAPARK	Si	12/19/2018 17:20	-0.1622993,-78.4616977	
19	UDLAPARK	Si	12/19/2018 17:35	-0.1628299,-78.4594461	
20	UDLAPARK	No	12/19/2018 17:40	N/A	El servidor no recibió la notificación.

En la figura 47 se puede observar un mapa con las ubicaciones de las pruebas para el campus Queri, correspondiente a la información presentada en la tabla 2.

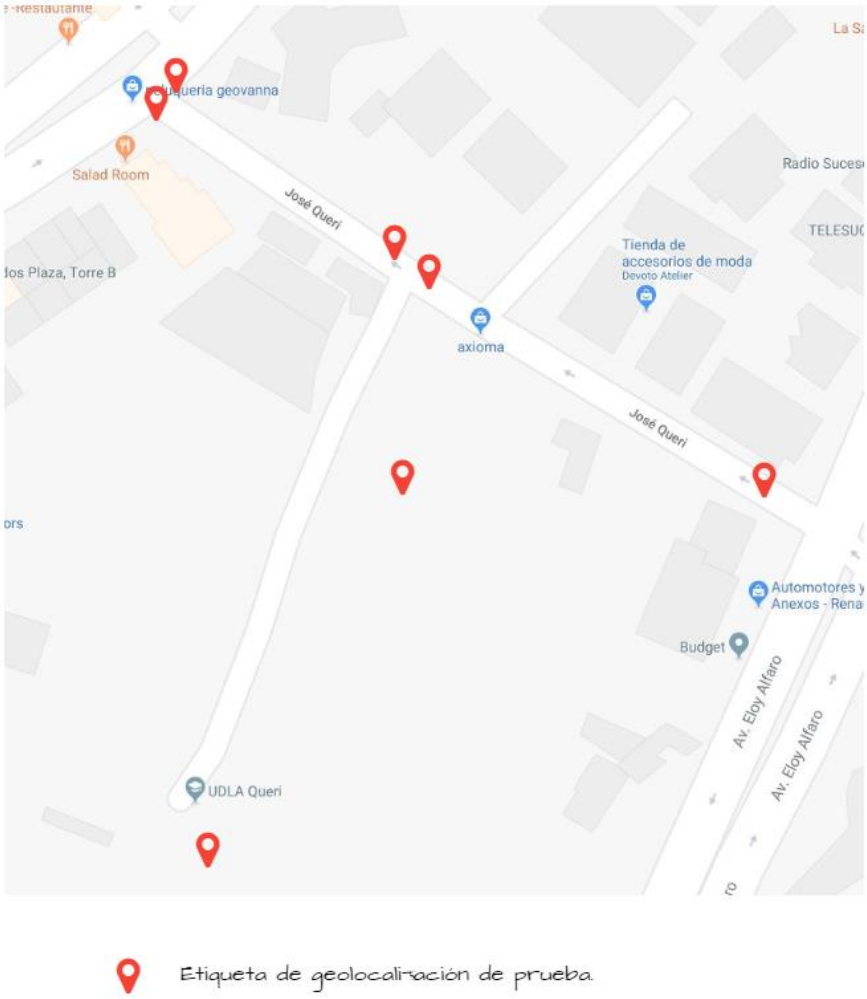


Figura 47. Pruebas en el Campus Queri.

Adaptado de (Google Maps, s.f.).

La primera prueba fallida se dio en el Campus Queri; en este caso la emergencia fue disparada desde el parqueadero del campus, pero la ubicación tuvo un error de localización aproximadamente 50 metros, dando como resultado la ubicación en la calle José Queri, como se muestra en la siguiente figura 48. El error se debe a que el método de ubicación que implementa la gama específica del teléfono utilizado para las pruebas, también se centra en la última ubicación conocida del dispositivo. Las pruebas fueron realizadas mientras el usuario de la aplicación se desplazaba caminando en las calles que rodean el campus.

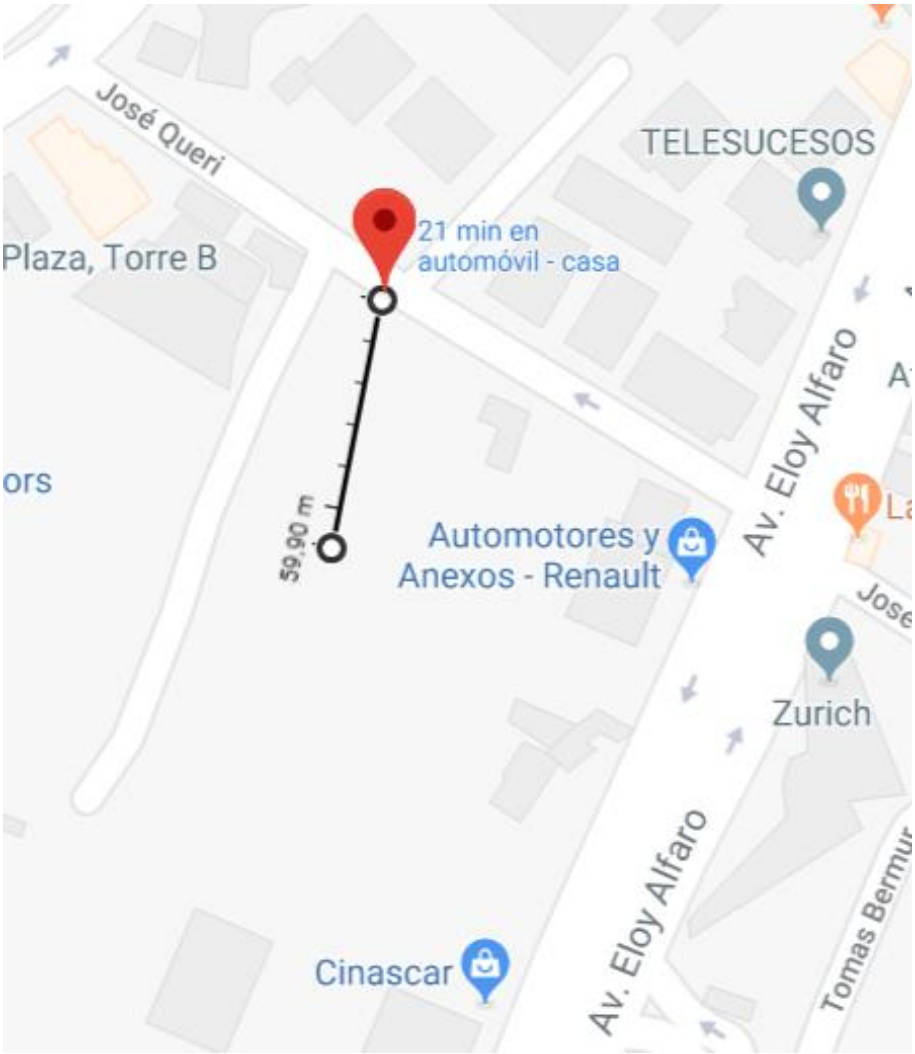


Figura 48. Distancia entre el evento y la ubicación recibida.

Adaptado de (Google Maps, s.f.).

Para el caso de las pruebas realizadas en el campus Granados, se evidencian las ubicaciones en la figura 49 de los eventos como se muestra en la tabla 1.

Figura 49. Pruebas en el Campus Granados.

Adaptado de (Google Maps, s.f.).

Continuando con las pruebas, se presenta un mapa con la geolocalización de los eventos que fueron realizados en el Campus UDLAPARK, especificado en la figura 50.

Figura 50. Pruebas en el Campus UDLAPARK.

Adaptado de (Google Maps, s.f.).

La segunda prueba errónea fue la realizada al interior del campus UDLAPARK. Se debió a que existieron intermitencias o un corte temporal en la conexión del teléfono inteligente con la radio base del proveedor de servicios de datos móviles por lo cual la información del usuario no fue enviada hacia el servidor de la UDLA, pero sí hacia el servidor de pruebas del Servicio Integrado de Seguridad ECU911.

El servidor UDLA, al tener una comunicación por medio de *TCP*, necesita una conexión permanente entre el teléfono que contiene la aplicación móvil y el servidor. Al presentarse intermitencias en la red de datos, es evidente que se puede generar un fallo en el envío de la notificación.

Al contrario del servidor de la UDLA, el servidor ECU911 cuenta con una comunicación *UDP*. Esto implica que, si existe una intermitencia en la red de datos, el paquete que contiene la información y viaja a través de la red es

colocado en cola hasta que la comunicación sea restablecida.

5.2 Análisis de costos

El análisis de costos determinará si el proyecto es viable para ser implementado desde un punto de vista económico y el impacto social que genere.

5.2.1 Identificación de costos y beneficios

En un análisis financiero, los beneficios son las ganancias o ingresos generados y los costos son los pagos que se realizan por la materia prima que se adquiere a los precios actuales del mercado (de Rus, 2010).

Para el caso de este proyecto, los costos se relacionan con el material tecnológico a ser adquirido, con exactitud, se necesitan los precios y cantidades de estos. En cuanto a los beneficios, al no ser un producto que se encontrará a la disposición del mercado, se determinará mediante el impacto social que genera la solución y las pérdidas económicas que van a ser evitadas gracias a la implementación de este.

5.2.2 Medición de costos

Después de haber realizado el análisis de ubicación estratégica de las cámaras y haber definido los requerimientos técnicos en el capítulo 4, se puede determinar el material necesario para realizar las respectivas instalaciones del sistema de pánico. En el anexo 7 se encuentran todas las mediciones realizadas para determinar el material necesario expresado en longitud.

- Fibra óptica, 3028 metros.
- Cable eléctrico, 6056 metros.
- Tubería, 3028 metros.

Los *NVR* a ser utilizados son de marca Epcom y las cámaras PTZ son de marca HIKVISION, ambos elementos son compatibles debido a que poseen comunicación Ethernet, en caso de que se implemente la solución con *NVR*. Si no se considera la implementación del sistema de pánico con *NVR's* la única condición que deben cumplir las cámaras son que cuenten con comunicación

IP *Ethernet*.

El conversor propuesto es de marca TP-Link. La característica principal de este modelo en particular (modelo se detalla en tabla 2) es que solamente utiliza un hilo de fibra para realizar la transmisión, debido a que funciona con tecnología *WDM (Wavelength Division Multiplexing)* en las ventanas de 1310nm y 1550nm para envío y recepción de información. Esto reduce los costos al utilizarse el 50% de *pigtails*.

A pesar de que solamente se necesita un solo hilo de fibra óptica para realizar la transmisión de información, se plantea el uso de fibra óptica de 2 hilos en la implementación del sistema de pánico, debido a que en un futuro puede surgir la necesidad de cambiar los conversores de medios a modelos que sí utilicen 2 hilos de fibra óptica para enviar y recibir información.

Se debe tomar en cuenta que, en el mercado existen bienes tecnológicos que tienen una cantidad mínima para ser vendidos, como es el tema de la fibra óptica y del cable eléctrico, rollos de 2km y rollos de 100m respectivamente.

En la tabla 3 se detallan las marcas, modelos, cantidades y precios para la implementación del sistema de pánico.

Tabla 3.

Medición de Costos

Descripción	Marca	Modelo	Unidad de medida	Cant.	Precio uni.	Precio total
NVR 8 puertos (opcional)	Epcom	XR28A/8P	unidad	1	$299,44	$299,44
NVR 4 puertos (opcional)	Epcom	XR24A/4P	unidad	3	$180,18	$540,54
Gabinete 6 UR o superior para albergar NVR's y conversor de medios	Beaucoup	I -1070 -N	unidad	4	$139,99	$559,96
Bandejas simples de 2 UR	Beaucoup	I-1107	unidad	24	$11,63	$279,12
Cámaras para exteriores IP 66 o superior	Hikvision	PTZ DS-2DE4425IW-DE	unidad	13	$605,00	$7.865,00
Conversor de medios WDM 10/100Mbps	TP-Link	MC111CS	unidad	26	$51,50	$1.339,00
Pigtails 1310 nm - 1550 nm,	N/A	N/A	unidad	26	$4,99	$129,74

conector SC, monomodo						
Patch cord RJ45 categoría 6 o superior, 1 metro	Nexxt	N/A	unidad	13	$3,50	$45,50
Patch cord RJ45 categoría 6 o superior, 2 metros	Nexxt	N/A	unidad	13	$4,75	$61,75
Fibra óptica para exteriores, monomodo, 1310 nm - 1550 nm, 2 hilos	Hentel	N/A	rollo 2 Km	3	$341,99	$1.025,97
Cajetín para exteriores IP 67 o superior para resguardar Conversor de medios y toma de corriente para cámaras	Beaucoup	I-0223	unidad	13	$13,92	$180,96
Tomacorriente simple con cajetín	Leviton	N/A	unidad	13	$3,21	$41,73
Cable eléctrico 14 AWG o mayor calibre	Conalesa	N/A	rollo de 100m.	61	$34,00	$2.074,00
Tubería para instalación de cables de fibra y eléctrico	Plastigama	N/A	metros	3028	$1,58	$4.784,24
TOTAL						**$19.226,95**

5.2.3 Medición del beneficio

La interpretación del beneficio en el sistema de alarma de pánico es un parámetro impredecible y no existe forma de calcularlo con exactitud. Se tiene un punto de partida con el hecho de que la aplicación tiene finalidades de resguardar la integridad física y evitar la sustracción de bienes materiales de las personas. No se puede saber cuántos robos a personas existirán en un futuro dentro de la zona. Por este motivo, la medición del beneficio se realizará con la generación del impacto social que tenga el proyecto en la comunidad.

Por otro lado, ECU911 puede hacer uso de la utilidad de la aplicación para enfocarlo en sectores vulnerables, tales como tercera edad, personas que padecen de discapacidades u otros para atenderlas de manera más eficaz en caso de que se encuentren sufriendo una emergencia.

A su vez, en comunidades donde sea muy difícil el acceso el servicio de ECU911 se puede enfocar el sistema de videovigilancia para que un encargado

en la comunidad pueda asistir donde ocurrió el evento con primeros auxilios hasta que los servicios de emergencia lleguen al lugar de los hechos.

6. CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

El análisis de ubicación de las cámaras en este proyecto se realizó de acuerdo con las especificaciones técnicas de la cámara elegida. En caso de efectuarse un cambio en el modelo de las cámaras, se necesita un nuevo análisis de ubicación.

El sistema de cámaras está diseñado de tal forma que los interesados en implementar este proyecto puedan escoger la cantidad de cámaras que se encuentren dentro del sistema. Esta flexibilidad permite al sistema con tan sólo una cámara tener la posibilidad de realizar videovigilancia en un lugar estratégico en concreto. Cabe destacar que mientras más cámaras se encuentren en el clúster, se tendrá una mejor visibilidad del lugar del evento.

Durante la investigación se obtuvieron datos que permiten visualizar la realidad en nuestro país sobre robos, asaltos y hurtos en los alrededores de los campus de la Universidad de las Américas, es por esta razón el proyecto tiene la expectativa de que disminuyan estos actos delictivos para una convivencia más segura de la comunidad en general.

En iOS, debido a restricciones de sistema no es posible obtener la captura de eventos de *hardware* sin necesidad de alterar el *software* del dispositivo (*Jailbreak*). A su vez, cumpliendo con el listado de tareas de ejecución larga definidas por iOS, una forma de mantener en segundo plano la aplicación es que se desarrolle en conjunto con un dispositivo externo (a manera de control remoto para activar el envío de la alerta desde el teléfono móvil hacia el servidor) con tecnología *Bluetooth*.

La aplicación permite que el botón de encendido se use como un disparador para el envío de la alerta, esto se puede hacer en la plataforma Android sin problema debido a que permite capturar las acciones de los botones físicos.

El usar el dispositivo móvil como una herramienta para el envío de notificaciones de emergencia se debe a las facilidades que los mismos

presentan, como la conectividad inalámbrica, haciendo de un smartphone un instrumento capaz de generar una alerta de forma sencilla y en cualquier parte de la ciudad.

El servidor y la aplicación, al usar comunicación a través de *TCP*, pueden ser propensos a fallos de conexión en caso de que exista un corte temporal o intermitencias en la red de datos del servidor o la red de datos inalámbrica del teléfono inteligente en el que se aloja la aplicación móvil.

Se debe tomar en cuenta que en el servidor donde vaya a ser alojada la aplicación, debe contar con una dirección IP pública, para permitir al usuario de la aplicación móvil conectarse hacia el servidor desde cualquier ubicación en la que necesite enviar una notificación de emergencia. Sin una dirección IP pública, el sistema de pánico solamente funcionaría en condiciones *LAN (Local Área Network)*.

En el método que se utilizó para comunicar la aplicación móvil del usuario con la aplicación proveedora de servicios es necesario aplicar reglas en el *Firewall* (habilitar un puerto) del servidor para que la conexión del *socket* sea exitosa.

Se constató mediante las pruebas realizadas que la precisión de la ubicación para el envío de la emergencia hacia el servidor depende netamente de la gama del teléfono inteligente que se use (baja, media, alta). Se entenderá que mientras mayor sea la gama del teléfono, la precisión de la ubicación también será mayor.

Los costos propuestos fueron realizados en base a los precios actuales del mercado nacional. En un futuro, los precios y la disponibilidad de los materiales tecnológicos pueden variar. Esto puede ocasionar que se tenga que optar por una importación de productos o elegir productos alternativos de diferente precio, teniendo un impacto directo en los costos del proyecto.

A pesar de que el proyecto no está dedicado a crear ganancias monetarias, genera un gran impacto social al prevenir que los objetos de valor de los estudiantes sean sustraídos y también, podría salvar vidas en caso de una

intervención oportuna de las autoridades cuando un incidente sea reportado.

Se optó por la opción de presionar 4 veces el botón de bloqueo debido a que los usuarios por lo general usan de una a tres veces este botón para desbloquear o para revisar notificaciones entrantes. No fueron utilizados los botones de control de volumen porque estos son empleados cuando el usuario está consumiendo contenido multimedia.

La solución del sistema de pánico es bastante efectiva respecto a los tiempos de respuesta, puesto que no existe retardo al momento de enviar la notificación desde el teléfono inteligente hacia el servidor de la UDLA. En cuanto el envío de la notificación desde el teléfono inteligente hacia el servidor del ECU911 sufre un retardo de, aproximadamente, 5 segundos, debido que la alerta enviada atraviesa procesos de validación con el servidor.

Para hacer posible que la notificación llegue hasta el Sistema Integrado ECU911, fue necesario realizar una visita para definir todos los parámetros e información con los que debe contar la aplicación que va a ser instalada en el teléfono inteligente. No se puede efectuar la comunicación de un aplicativo con el ECU911 sin un previo acuerdo o contrato. También, ECU911 provee de un usuario y contraseña para establecer una sesión entre la aplicación móvil y el servidor del ECU911.

6.2 Recomendaciones

Se aconseja realizar una mejora al servidor y a la aplicación móvil, cambiando el tipo de comunicación de *TCP* a *UDP* e implementando un servidor *SOAPful* o *RESTful* para la recepción de la información. *Esto se debe a que el protocolo TCP puede tener fallos cuando existen cortes temporales o intermitencias en la red.*

Es recomendable usar dispositivo con un sistema operativo mayor a Android KiKat v4.4 (API 19) debido a compatibilidad y mejoras en los métodos de desarrollo para la obtención de la ubicación dentro de la plataforma de programación Android Studio.

Si se desea lanzar una versión de la aplicación de forma independiente en la tienda *Google (Play Store)* se sugiere modificar el código fuente de la aplicación que integra la conexión con el servidor del ECU911. Además, se pueden realizar mejoras futuras del diseño en la aplicación desarrollada para la Universidad.

Para superar los obstáculos que se tuvieron al desarrollar la aplicación en *iOS,* es recomendable utilizar un dispositivo externo para accionar la alarma de pánico como una alternativa. Puede tratarse de un terminal que se enlace al *iPhone* por medio de una red *Wi-Fi* o *Bluetooth*, como por ejemplo un *Beacon.*

Para reforzar el impacto que el sistema de pánico genera en la comunidad como método de alerta en contra de actos delictivos, se puede añadir un conjunto de alarmas, tanto audibles como luminosas, dentro de la superficie de cobertura del sistema de videovigilancia y que sean activadas mediante la señal que envía la aplicación móvil al servidor de la UDLA.

REFERENCIAS

Apple Inc. (2017). *Background Execution*. Recuperado el 2 de Enero de 2019, de https://developer.apple.com/library/archive/documentation/iPhone/Conceptual/iPhoneOSProgrammingGuide/BackgroundExecution/BackgroundExecution.html

Berkenfeld, D., Corrado, M., & Silverman, L. (s.f.). Entendiendo la Distancia Focal. Recuperado el 01 de Enero de 2019, de https://www.nikon.com.mx/learn-and-explore/a/tips-and-techniques/entendiendo-la-distancia-focal.html#

Black Box Corporation. (s.f.). *Media Converters*. Recuperado el 30 de Diciembre de 2018, de https://www.blackbox.com/en-us/products/black-box-brand-products/networking/media-converters

de Rus, G. (2010). *Introduction to Cost-Benefit Analysis Looking for Reasonable Shortcuts.* (E. Elgar, Ed.) Northampton, Massachusetts, Estados Unidos: Edward Elgar Publishing, Inc.

Departamento de Ciencia de la Computación e Inteligencia Artificial. (2014). Creación de Servicios Web SOAP. Recuperado el 12 de Noviembre de 2018, de http://www.jtech.ua.es/j2ee/publico/servc-web-2012-13/sesion02-apuntes.html

Departamento de Ciencia de la Computación e Inteligencia Artificial. (2014). Introducción a los Servicios Web. Invocación de servicios web SOAP. Recuperado el 12 de Noviembre de 2018, de http://www.jtech.ua.es/j2ee/publico/servc-web-2012-13/sesion01-apuntes.html#SOAP

ECMA International. (2017). *The JSON Data Interchange Syntax.* Recuperado

el 15 de Octubre de 2018, de http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf

ECU911. (s.f.). ¿Como reportar al 9-1-1? Recuperado el 28 de Octubre de 2018, de http://www.ecu911.gob.ec/como-reportar-al-9-1-1/

Gobierno del Estado de Tabasco. (s.f.). GUIA PARA APLICAR LA NORMA TIA/EIA 568 PARA CABLEADO ESTRUCTURADO. Recuperado el 30 de Diciembre de 2018, de https://tabasco.gob.mx/sites/default/files/Manual-para-aplicar-la-norma-TIA-EIA-para-Cableado-Estructurado.pdf

Google LLC. (2018). *Broadcast*. Recuperado el 23 de Octubre de 2018, de https://developer.android.com/guide/components/broadcasts

Google LLC. (2018). *BroadcastReceiver*. Recuperado el 23 de Octubre de 2018, de https://developer.android.com/reference/android/content/BroadcastReceiver

Google LLC. (2018). *Intents* y filtros de *intents*. Recuperado el 23 de Octubre de 2018, de https://developer.android.com/guide/components/intents-filters?hl=es-419

Google LLC. (2018). Servicios. Recuperado el 16 de Octubre de 2018, de https://developer.android.com/guide/components/services?hl=es-419

Google Maps. (s.f.). Distancia entre el evento y la ubicación recibida. Recuperado de https://www.google.com/maps/place/UDLA+QUERI/@-0.167215,-78.4720672,17z/data=!4m5!3m4!1s0x91d591c4f5e63ba9:0x58a449e78a19f984!8m2!3d-0.1675047!4d-78.4709514

Google Maps. (s.f.). Mapa UDLA Sede Colón con puntos geográficos de cámaras. Recuperado de

https://www.google.com/maps/place/UDLA+Col%C3%B3n/@-0.2023234,-78.4863551,18z/data=!4m5!3m4!1s0x91d59a0d470e2bbb:0xda7d17be14f8cce6!8m2!3d-0.2023234!4d-78.4852608

Google Maps. (s.f.). Mapa UDLA Sede Granados con puntos geográficos de cámaras. Recuperado de https://www.google.com/maps/place/UDLA/@-0.1681562,-78.4738979,17z/data=!4m5!3m4!1s0x91d591d7f98963dd:0x52a0c8426a406a8c!8m2!3d-0.167888!4d-78.46804

Google Maps. (s.f.). Mapa UDLA Sede Queri con puntos geográficos de cámaras. Recuperado de https://www.google.com/maps/place/UDLA+QUERI/@-0.167215,-78.4720672,17z/data=!4m5!3m4!1s0x91d591c4f5e63ba9:0x58a449e78a19f984!8m2!3d-0.1675047!4d-78.4709514

Google Maps. (s.f.). Mapa UDLA Sede UDLAPARK con puntos geográficos de cámaras. . Recuperado de https://www.google.com/maps/place/Udlapark/@-0.163131,-78.4612229,17z/data=!4m5!3m4!1s0x91d590130ce77433:0x7f43dc0aeefb5973!8m2!3d-0.1630237!4d-78.4590879

Google Maps. (s.f.). Pruebas en el Campus Granados. Recuperado de https://www.google.com/maps/place/UDLA/@-0.1681562,-78.4738979,17z/data=!4m5!3m4!1s0x91d591d7f98963dd:0x52a0c8426a406a8c!8m2!3d-0.167888!4d-78.46804

Google Maps. (s.f.). Pruebas en el Campus Queri. Recuperado de https://www.google.com/maps/place/UDLA+QUERI/@-0.167215,-78.4720672,17z/data=!4m5!3m4!1s0x91d591c4f5e63ba9:0x58a449e78a19f984!8m2!3d-0.1675047!4d-78.4709514

Google Maps. (s.f.). Pruebas en el Campus UDLAPARK. Recuperado de https://www.google.com/maps/place/Udlapark/@-0.163131,-

78.4612229,17z/data=!4m5!3m4!1s0x91d590130ce77433:0x7f43dc0aeefb5973!8m2!3d-0.1630237!4d-78.4590879

GVS Colombia. (s.f.). ¿Porque es mejor usar cámaras PTZ? Recuperado el 31 de Diciembre de 2018, de https://www.gvscolombia.com/porque-es-mejor-usar-camaras-ptz/

IBM Technologies. (s.f.). ¿Qué es *WSDL*? Recuperado el 31 de Diciembre de 2018, de https://www.ibm.com/support/knowledgecenter/es/SSMKHH_10.0.0/com.ibm.etools.mft.doc/ac34640_.htm

Infinite Electronics International, Inc. (s.f.). *Ethernet Media Converter Overview*. Recuperado el 30 de Diciembre de 2018, de http://www.l-com.com/ethernet-converters

Ministerio del Interior. (2018). Indicadores de Seguridad Ciudadana. Recuperado el 22 de Julio de 2018, de http://cifras.ministeriodelinterior.gob.ec/comisioncifras/inicio.php

NTT DATA. (s.f.). *SOAP Web Service (Server/Client)*. Recuperado el 10 de Octubre de 2018, de https://terasolunaorg.github.io/guideline/5.2.0.RELEASE/en/ArchitectureInDetail/WebServiceDetail/SOAP.html

Observatorio Municipal de Seguridad Ciudadana. (2013). 20 OMSC Informe Estadistico y Georeferenciacion Octubre 2013. Recuperado el 22 de Julio de 2018, de http://omsc.quito.gob.ec/index.php/biblioteca-virtual/informes-mensuales.html

Rainford Solutions Ltd. (2017). *What is Ingress Protection?* Recuperado el 31 de Diciembre de 2018, de https://www.rainfordsolutions.com/what-is-ingress-protection

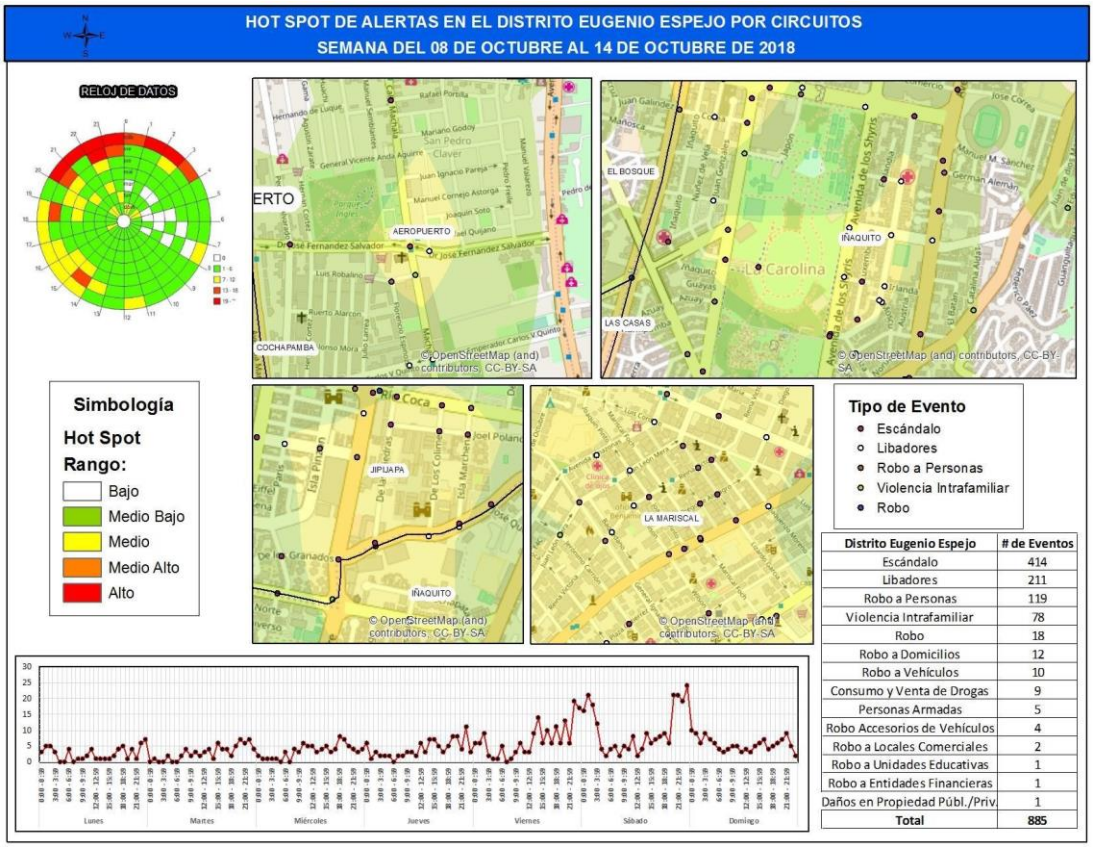
TECNOSeguro. (s.f.). *¿Qué es PoE ?* Recuperado el 30 de Diciembre de 2018, de https://www.tecnoseguro.com/faqs/electronica/que-es-poe

Tuli, S. (2018). *Java Thread Tutorial: Creating Threads and Multithreading in Java*. Recuperado el 28 de Diciembre de 2018, de https://dzone.com/articles/java-thread-tutorial-creating-threads-and-multithr

ANEXOS

ANEXO 1

Hot Spot de alertas en el distrito Eugenio Espejo, por circuitos, semana del 8 al de octubre al 14 de octubre de 2018. Cortesía de ECU 911.



Distrito Eugenio Espejo	# de Eventos
Escándalo	414
Libadores	211
Robo a Personas	119
Violencia Intrafamiliar	78
Robo	18
Robo a Domicilios	12
Robo a Vehículos	10
Consumo y Venta de Drogas	9
Personas Armadas	5
Robo Accesorios de Vehículos	4
Robo a Locales Comerciales	2
Robo a Unidades Educativas	1
Robo a Entidades Financieras	1
Daños en Propiedad Públ./Priv.	1
Total	**885**

ANEXO 2

Tabla de referencia del nivel de protección de ingreso *(Ingress Protection)*

IP Rating	First Digit - SOLIDS	Second Digit - LIQUIDS
IP54	Protected from limited dust ingress	Protected from water spray from any direction, limited ingress protection
IP55	Protected from limited dust ingress	Protected from low pressure water jets from any direction, limited ingress protection
IP56	Protected from limited dust ingress	Protected from high pressure water jets from any direction, limited ingress protection
IP57	Protected from limited dust ingress	Protected from immersion between 15 centimeters and 1 meter in depth, limited ingress protection
IP58	Protected from limited dust ingress	Protected from long term immersion up to a specified pressure, limited ingress protection
IP60	Protected from total dust ingress	Not protected from liquids, limited ingress protection
IP61	Protected from total dust ingress	Protected from condensation, limited ingress protection
IP62	Protected from total dust ingress	Protected from water spray less than 15 degrees from vertical, limited ingress protection
IP63	Protected from total dust ingress	Protected from water spray less than 60 degrees from vertical, limited ingress protection
IP64	Protected from total dust ingress	Protected from water spray from any direction, limited ingress protection
IP65	Protected from total dust ingress	Protected from low pressure water jets from any direction, limited ingress protection
IP66	Protected from total dust ingress	Protected from high pressure water jets from any direction, limited ingress protection
IP67	Protected from total dust ingress	Protected from immersion between 15 centimeters and 1 meter in depth, limited ingress protection
IP68	Protected from total dust ingress	Protected from long term immersion up to a specified pressure, limited ingress protection
IP69K	Protected from total dust ingress	Protected from steam-jet cleaning, limited ingress protection

ANEXO 3

Informe de Policía Nacional del Ecuador N.-2018-32-EJCF-DGO, requerimiento de datos acerca de sitios geográficos donde han ocurrido incidencias de robo a personas alrededor de los Campus de la Universidad De Las Américas en la ciudad de Quito.

POLICÍA NACIONAL DEL ECUADOR
DIRECCIÓN GENERAL DE OPERACIONES
R. DEL E.

Ministerio del Interior

INFORME N.- 2018-32-EJCF-DGO, REQUERIMIENTO DE DATOS ACERCA DE LOS SITIOS GEOGRÁFICOS DONDE HAN OCURRIDO INCIDENCIAS DE ROBO A PERSONAS ALREDEDOR DEL CAMPUS DE LA UNIVERSIDAD DE LAS AMÉRICAS EN LA CIUDAD DE QUITO.

DE : Edgar Carrera
Cabo Segundo de Policía
ESTADÍSTICO DE LA DIRECCIÓN GENERAL DE OPERACIONES.

PARA : Gabriel Añasco
Capitán de Policía
JEFE DE LA SECCIÓN DE ESTADÍSTICA DE LA DIRECCIÓN GENERAL DE OPERACIONES.

ASUNTO : Informe de **ROBO A PERSONAS** alrededor del campus de la Universidad de Las Américas en la ciudad de Quito, periodo 2017 -2018

FECHA : Distrito Metropolitano de Quito, 04 de noviembre de 2018

3. **ANTECEDENTES:**

- Oficio N.º 2018/292/AJ/DGO/PN, de fecha 25 de octubre de 2018, suscrito por la Sr. Ab. Cbop. Enrique Sanabria Aucancela, Auxiliar Jurídico del Departamento de Asesoría Jurídica-DGO.
- Memorando N.º 2018-15401-CG-QX-PN, de fecha 23 de octubre de 2018, suscrito por El Sr. Gral. MSc. Nelson Humberto Villegas Ubillús, Comandante General de la Policía Nacional del Ecuador.
- Oficio S/N, de fecha 18 de octubre de 2018, suscrito por el Sr. Miguel Ángel Tello, Estudiante de la Carrera de Ingeniería en Redes y Telecomunicaciones de la Universidad de las Américas.

4. **Trabajos RealizadoS**

La Universidad de las Américas, fue fundada en 1995 por un grupo de empresarios chilenos y ecuatorianos. La UDLA, mediante resolución decretada el 10 de mayo de 2016 por el CEAACES, se ubica actualmente en la categoría B. Tiene cuatro campus:

- Campus Granados, entre Isla Manchena y De los Colimes (Este y Oeste), entre Joel Polanco y Av. De los Granados (Norte y Sur)
- Campus Queri, entre Av. Eloy Alfaro y Av. de los Granados (Este y Oeste), entre José Queri y Gaspar de Villaroel (Norte y Sur).
- Campus UDLAPark, Av. de las Azucenas y Av. Granados (Oeste), entre Av. Nayon y Av. Simón Bolívar (Norte y Sur).
- Campus Colón, entre Plácido Camaño y Av. 6 de Diciembre (Este y Oeste), entre San Ignacio y Av. Colón (Norte y Sur).

En la Universidad de las Américas existe un aproximado de 1800 estudiante en todas las carreras académicas que ofrece por tal motivo y en petición del Sr. Miguel Ángel Baquero estudiante de la carrera de Ingeniería en Redes y Telecomunicación de la mencionada Universidad y según datos oficiales que se encuentran en la Dirección General de Operaciones se puede detallar que:

POLICÍA NACIONAL DEL ECUADOR
DIRECCIÓN GENERAL DE OPERACIONES
R. DEL E.

ROBO A PERSONAS EN EL CAMPUS DE LA UDLA "COLON"

CAMPUS COLON	
MODALIDAD	FRECUENCIA
ARRANCHADORES	36
ASALTO	80
ATURDIMIENTO POR SUSTANCIAS	5
CARTERISTAS	6
SACAPINTAS	2
Total general	**129**

ROBO A PERSONAS EN EL CAMPUS DE LA UDLA "UDLAPARK"

CAMPUS UDLAPARK	
MODALIDAD	FRECUENCIA
ARRANCHADORES	2
ASALTO	9
Total general	**11**

ROBO A PERSONAS EN EL CAMPUS DE LA UDLA "QUERI Y GRANADOS"

CAMPUS GRANADOS Y QUERI	
MODALIDAD	FRECUENCIA
ARRANCHADORES	18
ASALTO	44
ATURDIMIENTO POR SUSTANCIAS	1
CARTERISTAS	2
SACAPINTAS	3
Total general	**68**

Como se puede observar en todos los campus de la Universidad de las Américas existe un total de 208 robo a personas en diferentes modalidades así en resumen general:

ROBO A PERSONAS EXTERIORES DE LA UDLA	
MODALIDAD	**FRECUENCIA**
ASALTO	133
ARRANCHADORES	56
CARTERISTAS	8
ATURDIMIENTO POR SUSTANCIAS	6
SACAPINTAS	5
Total general	**208**

COM SE PUEDE OBSERVAR LA MODALIDAD DE ASALTO ES CON LA QUE MAYOR FRECUENCIA COMETEN ESTÉ TIPO DE DELITO.

III.- CONCLUSIONES:

Con los antecedentes expuestos se concluye el siguiente:

1. La sede COLON de la Universidad de las Américas es la que tiene más delitos de robo a personas con un valor absoluto de 129 eventos.
2. La modalidad que frecuenta el delito robo a personas es el de ASALTO seguido de arranchadores.
3. Los datos presentados en el presente informe son los que actualmente se encuentran en la DIRECCIÓN GENERAL DE OPERACIONES.

IV.- RECOMENDACIÓN

Con los antecedentes expuestos me permito realizar las siguientes recomendaciones:

1. Remitir el presente informe al peticionario para su trámite respectivo.

Particular que me permito poner en su conocimiento para los fines pertinente

ANEXO INFORME DEL SERVICIO DE TURISMO DEL DMG.

Atentamente,
DIOS, PATRIA Y LIBERTAD.

Edgar Cantera
Cabo Segundo de Policía
ESTADISTICO DE LA DIRECCIÓN GENERAL DE OPERACIONES.
District.
Orig. : SUB-DGO
C.I. : Arch.

ANEXO 4

Encuesta de seguridad en los alrededores de la Universidad De Las Américas. - Formularios de Google

Encuesta de seguridad en los alrededores de los campus de la Universidad De Las Américas.

Solicitamos que la encuesta sea llenada solamente si eres alumno de la Universidad De Las Américas. Esta encuesta es realizada con motivos académicos, la información será utilizada para un proyecto de titulación. Se asegura el anonimato de las respuestas proporcionadas.

* Required

1. **Email address** *

2. **¿En qué campus estudias?** *
   Mark only one oval.
   - ◯ UDLAPARK
   - ◯ Queri
   - ◯ Granados
   - ◯ Colón

3. **¿Has sufrido percances relacionados a robos, asaltos, agresiones o similares en los alrededores del campus en el que estudias?** *
   Mark only one oval.
   - ◯ Sí
   - ◯ No

4. **En caso de que tu respuesta anterior haya sido "sí", ¿en qué dirección ha sucedido el incidente?**

5. **¿Te gustaría contar con una alarma de pánico que se active desde tu teléfono inteligente?** *
   Mark only one oval.
   - ◯ Sí
   - ◯ No
   - ◯ Tal vez

ANEXO 5

Encuesta de seguridad en los alrededores de la Universidad De Las Américas.

a. Dirección de correo electrónico
b. ¿En qué campus estudias?
c. ¿Has sufrido percances relacionados a robos, asaltos, agresiones o similares en los alrededores del campus en el que estudias?
d. En caso de que tu respuesta anterior haya sido "sí", ¿en qué dirección ha sucedido el incidente?
e. ¿Te gustaría contar con una alarma de pánico que se active desde tu teléfono inteligente?

Se incluyó como parámetro de las preguntas la dirección de correo electrónico para comprobar que los estudiantes pertenezcan a la Universidad De Las Américas por medio de los dominios de los correos institucionales (udla.edu.ec o udlanet.ec). A pesar de que 5 estudiantes de los 46 encuestados (10,86%) no colocaron su correo institucional de la Universidad, se logró comprobar que se encuentran estudiando actualmente en la UDLA haciendo contacto directo con ellos.

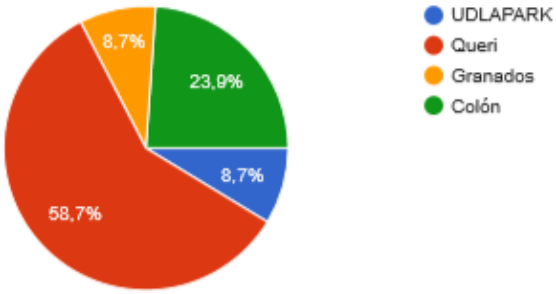
A continuación, en los siguientes gráficos, se exponen los resultados gráficos de las preguntas detalladas anteriormente, a excepción de los correos electrónicos de cada estudiante.

Encuesta de seguridad en los alrededores de los campus de la Universidad De Las Américas.

46 respuestas

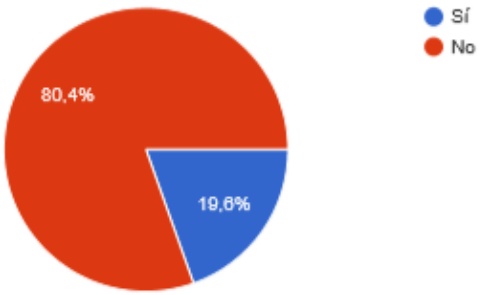
¿En qué campus estudias?

46 respuestas



¿Has sufrido percances relacionados a robos, asaltos, agresiones o similares en los alrededores del campus en el que estudias?

46 respuestas



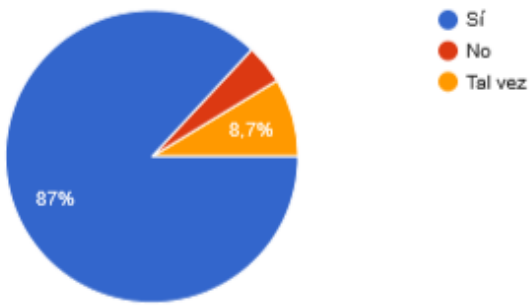
En caso de que tu respuesta anterior haya sido "sí", ¿en qué dirección ha sucedido el incidente?

9 respuestas

- En 6 de diciembre y granados
- José Polanco y Colimes
- Colones y José Polanco
- Colon
- Colón y 6 de diciembre
- Colón y 6 de diciembre
- Av Colón y Coruña
- Av. Colón y 6 de diciembre
- Colon o en la 6 de diciembre

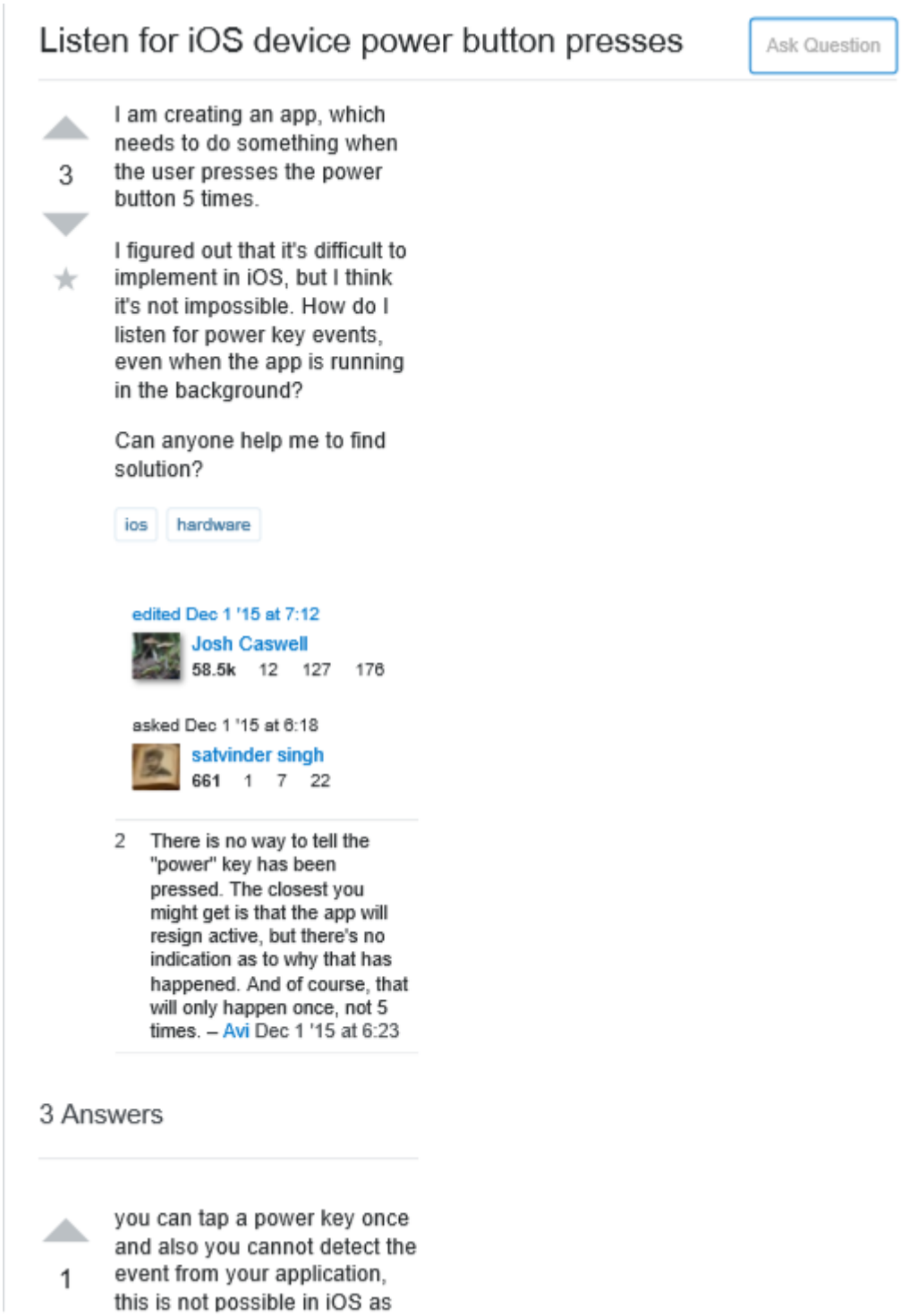
¿Te gustaría contar con una alarma de pánico que se active desde tu teléfono inteligente?

46 respuestas



ANEXO 6

Respuestas de la comunidad de desarrolladores respecto a capturar evento de hardware



Listen for iOS device power button presses

Ask Question

3

I am creating an app, which needs to do something when the user presses the power button 5 times.

I figured out that it's difficult to implement in iOS, but I think it's not impossible. How do I listen for power key events, even when the app is running in the background?

Can anyone help me to find solution?

ios hardware

edited Dec 1 '15 at 7:12

Josh Caswell

58.5k 12 127 176

asked Dec 1 '15 at 6:18

satvinder singh

661 1 7 22

2 There is no way to tell the "power" key has been pressed. The closest you might get is that the app will resign active, but there's no indication as to why that has happened. And of course, that will only happen once, not 5 times. – Avi Dec 1 '15 at 6:23

3 Answers

1

you can tap a power key once and also you cannot detect the event from your application, this is not possible in iOS as

in your app and then send a alert based on those conditions if satisfied, I think this would be better,

Somehow ,If you try to override the existing functionality of the power key, apple will reject your app I think so,

answered Dec 1 '15 at 7:06

satheesh
1,961 1 7 16

1

You can't directly get the power button events. But there are notifications which you can count like `UIApplicationProtectedDataWillBecomeUnavailable` or `UIApplicationWillResignActiveNotification` . Or just register for all low level notifications with `CFNotificationCenter` and see if you find something fitting like `com.apple.springboard.lockstate` .

edited Dec 1 '15 at 10:47

answered Dec 1 '15 at 10:34

orkoden
12.6k 2 48 44

thanks @orkoden, for answering, is using CFNotificationCenter for detecting power key presses according to apple's guidlines? – satvinder singh Dec 1 '15 at 12:34

You can use `CFNotificationCenter`

satvinder singh Dec 2 '15 at 6:56

0

I don't think you can override system level actions like holding the power button, pressing the home button, overriding the mute sound switch within your own app. iOS system doesn't exactly behave like an normal computer OS, it's to much more limited.

> Apple is not allowing you to use hardware components completely. They have added some restrictions. They provided the method in the app delegate i.e. applicationDidEnterBackground can catch the home button press .Also they has provided the the API's to access the camera,bluetooth etc .At least this much of API's I know which provided by apple publicly to access the hardware. You cannot access the other hardware elements in your application which not provided publicly by apple .If you are able to do this by any way then also your application will not approved by apple .

edited Dec 1 '15 at 7:25

answered Dec 1 '15 at 7:19

Vvk
3,045 19 40

Thread: How can I detect if the power button is pressed?

Like Be the first of your friends to like this.

Thread Tools Search Thread

3rd January 2016, 06:50 PM #1

mutantdude

Clicker

Join Date: Apr 2008
Location: Nova Scotia, Canada
Posts: 48
Mentioned: 0 Post(s)
Tagged: 0 Thread(s)

How can I detect if the power button is pressed?

Is there a way to detect if the power button has been pressed on a device? (The button used to turn off the screen.) The Android Object has conditions for 'On Back Button Pressed' and 'On Home Button Pressed,' but nothing for the power button.

I'm working on a game in which I don't want the user to have the ability to pause. By pressing the power button and turning off the screen, the user can come back to where they left off later on, which I don't want.

If there isn't a way to detect if the power button has been pressed, is there a way to detect when the screen has been turned off or if the application has been deactivated?

Reply With Quote

4th January 2016, 02:13 PM #2

Fernando

Clickteam

Join Date: Dec 2006
Posts: 5,397
Mentioned: 46 Post(s)
Tagged: 2 Thread(s)

unfortunately, this is not possible, except perhaps via custom firmware.

what you can do, is using android object

"Sleep prevention: start"

before close your game sor dev

"Sleep prevention: stop"

Regards,

Fernando Vivolo

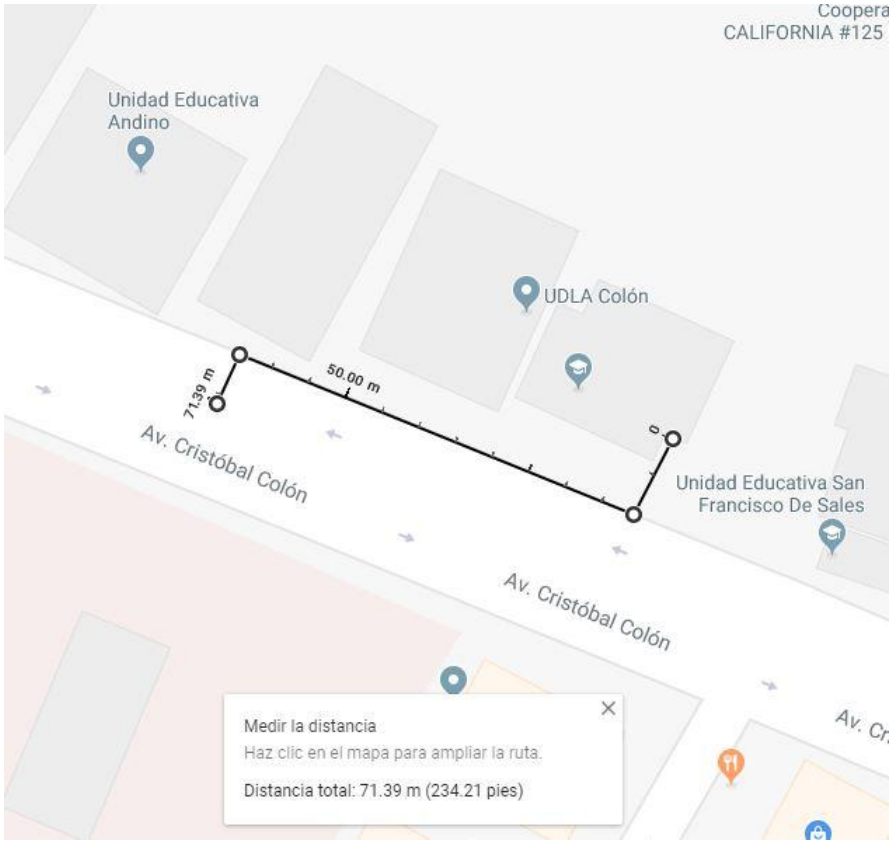
... new things are coming ...

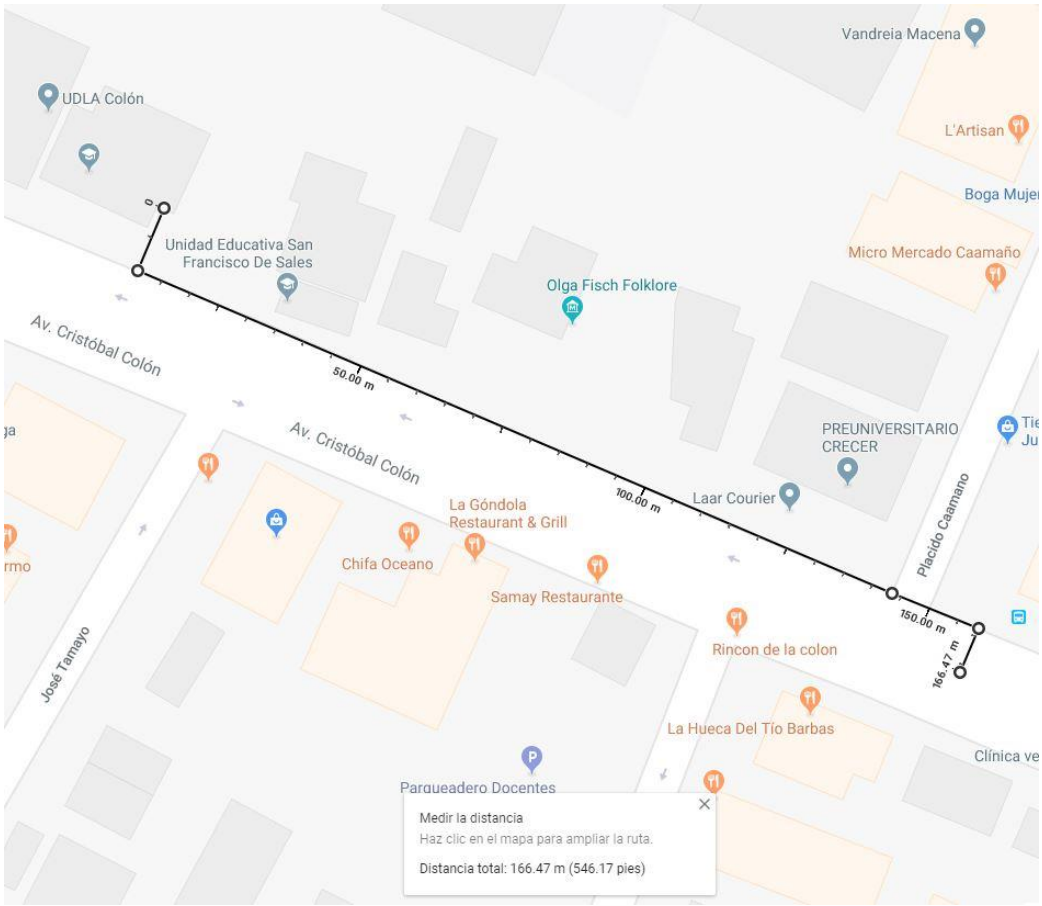
Reply With Quote

Quick Navigation **Android Export Module 2.5** **Top**

ANEXO 7

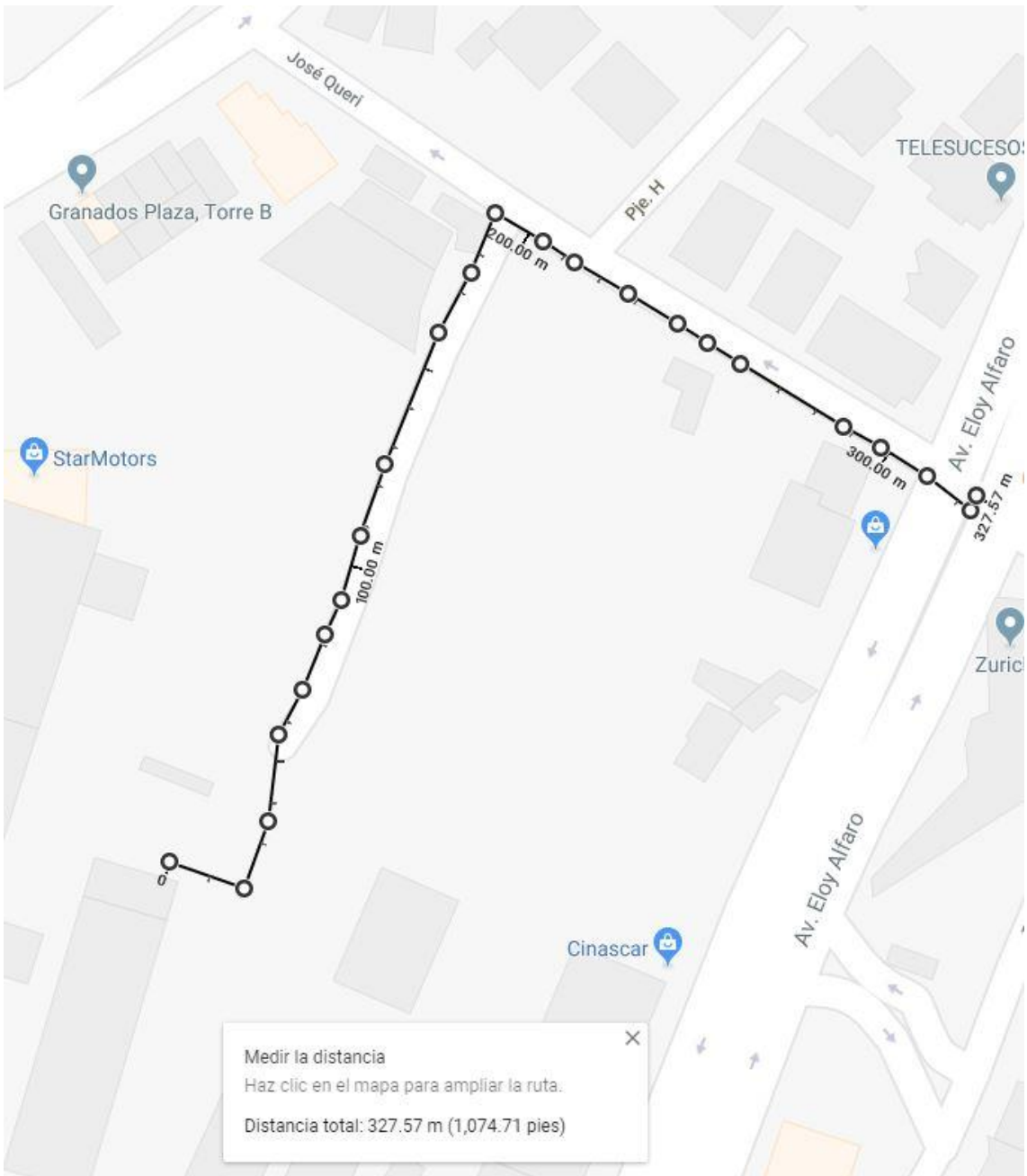
Medición de los materiales cuantificables en unidades de longitud desde el Campus hasta la respectiva cámara, a través de la herramienta de medición de *Google Maps*.




Vandreia Macena
UDLA Colón
L'Artisan
Boga Muje
Unidad Educativa San Francisco De Sales
Micro Mercado Caamaño
Olga Fisch Folklore
Av. Cristóbal Colón
50.00 m
Av. Cristóbal Colón
PREUNIVERSITARIO CRECER
100.00 m
Laar Courier
Placido Caamano
La Góndola Restaurant & Grill
Chifa Oceano
Samay Restaurante
150.00 m
165.47 m
Rincon de la colon
José Tamayo
La Hueca Del Tío Barbas
Clínica ve
Parqueadero Docentes
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 166.47 m (546.17 pies)

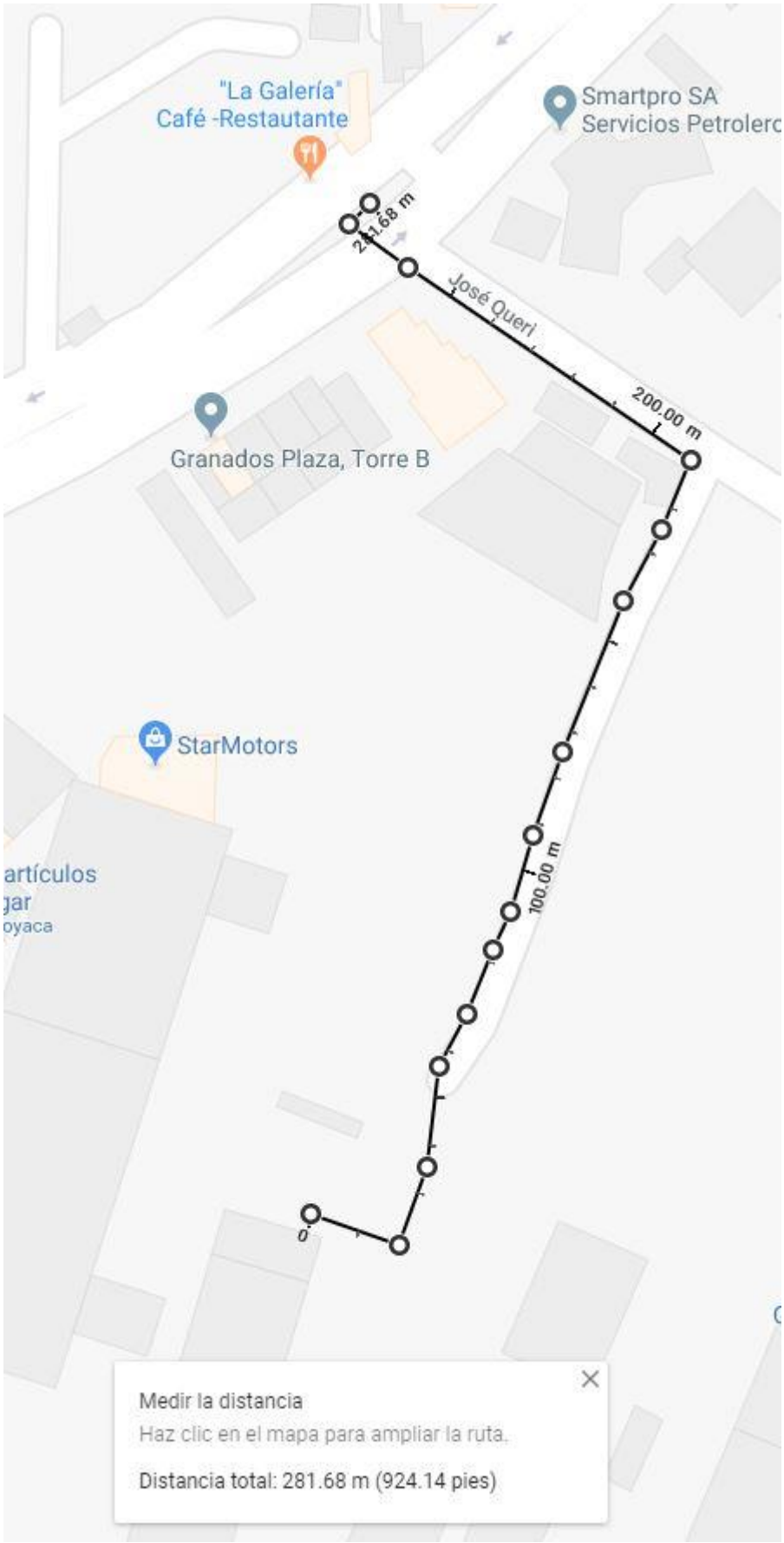

208.97 m
De Los Colimes
100.00 m
UDLA Granados
De Los Granados
Y Marchena
De los Granados
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 208.97 m (685.61 pies)


414.92 m
400.00 m
200.00 m
Condominios El Inca
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 414.92 m (1,361.29 pies)

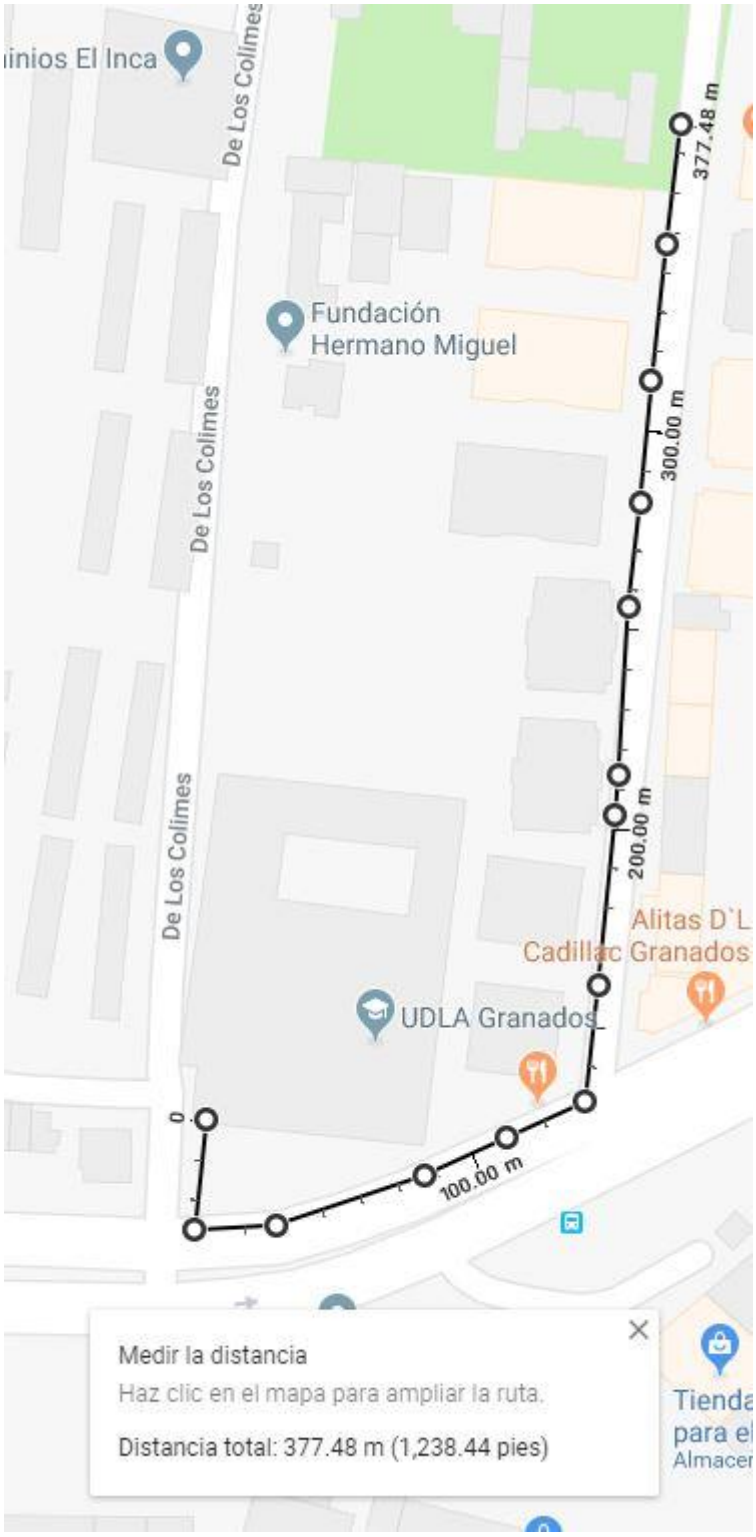

José Queri
Granados Plaza, Torre B
Pje. H
TELESUCESOS
200.00 m
Av. Eloy Alfaro
StarMotors
300.00 m
327.57 m
100.00 m
Zurich
Av. Eloy Alfaro
Cinascar
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 327.57 m (1,074.71 pies)

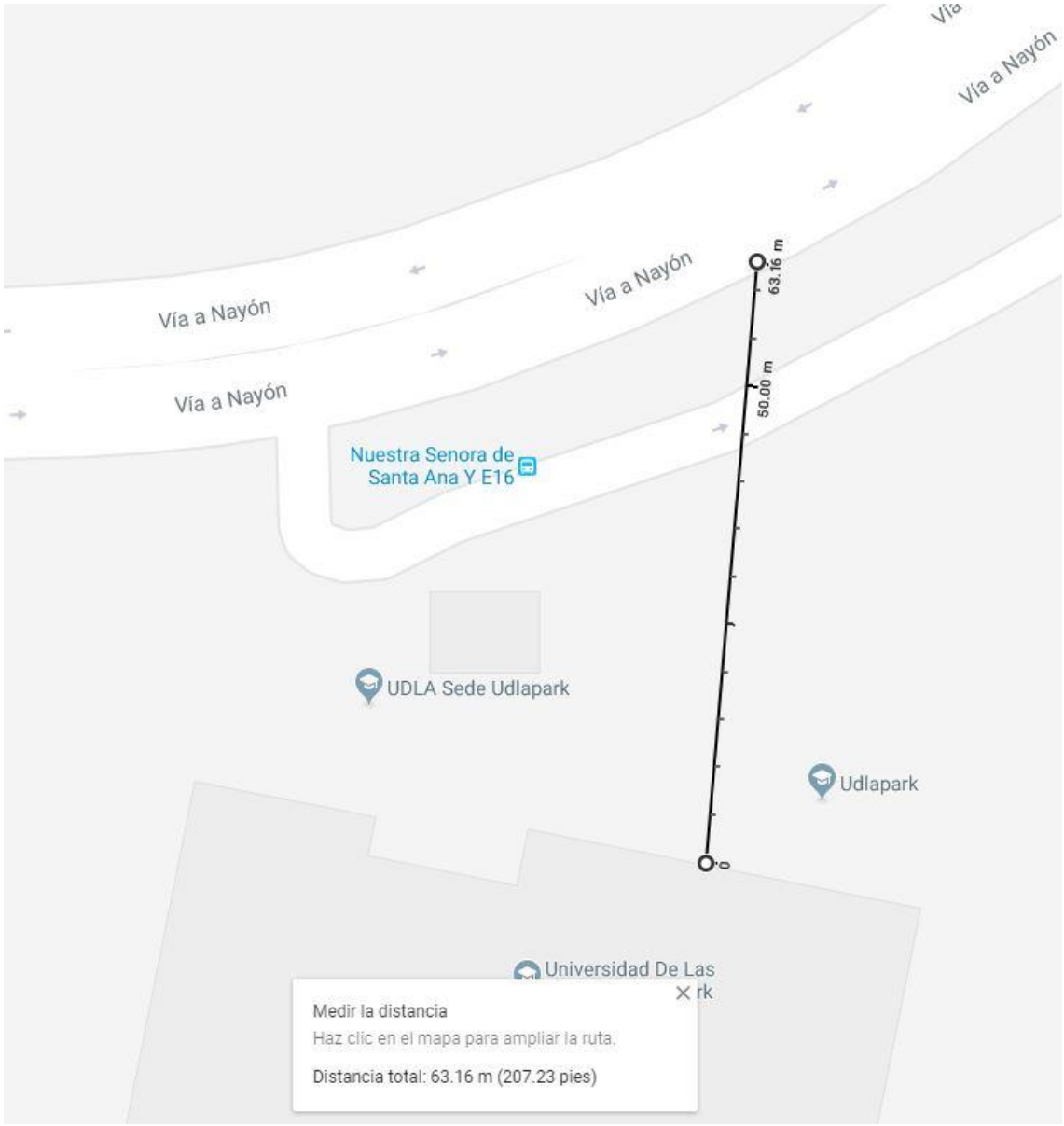

UDLA Queri
35.78 m
0
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 35.78 m (117.39 pies)

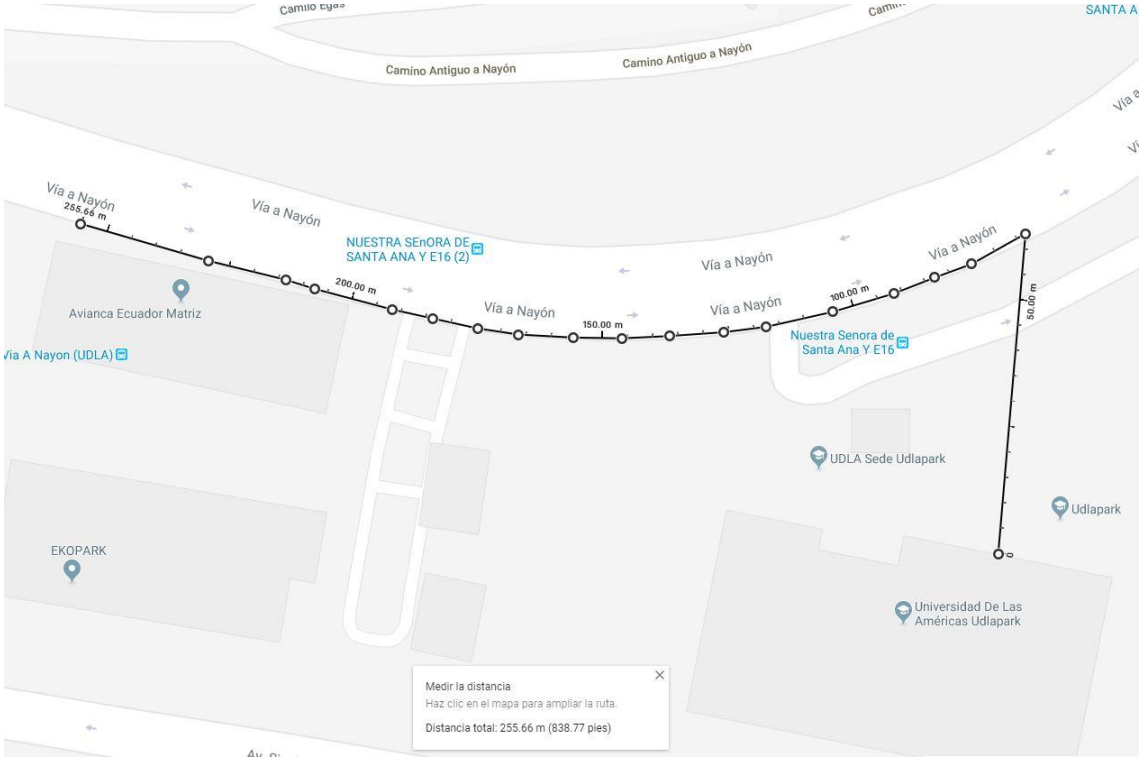

UDLA Granados
Al
0
De Los Granados
Y Marchena
De Los Granados
Y Marchena (Sn)
45.59 m
Ecuawagen
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 45.59 m (149.57 pies)

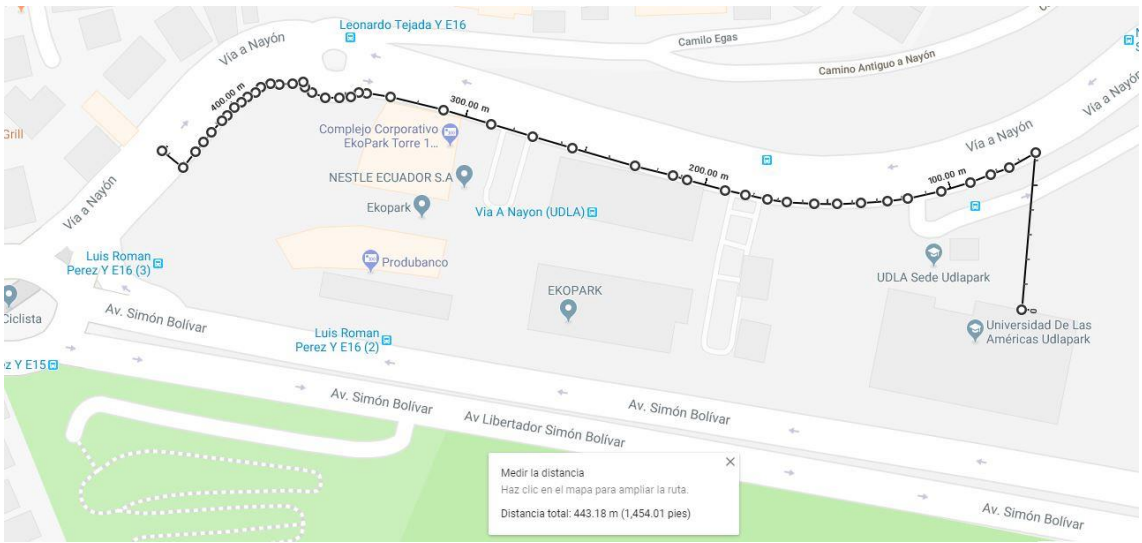

"La Galería"
Café -Restautante
Smartpro SA
Servicios Petroler
281.68 m
José Queri
200.00 m
Granados Plaza, Torre B
StarMotors
artículos
gar
oyaca
100.00 m
0
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 281.68 m (924.14 pies)


De Los Colimes
UDLA Granados
De Los Granados
Y Marchena
De Los Granados
Y Marchena (Sn)
De los Gra
0
100.00 m
213.69 m
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 213.69 m (701.07 pies)


inios El Inca
De Los Colimes
Fundación
Hermano Miguel
De Los Colimes
De Los Colimes
377.48 m
300.00 m
200.00 m
100.00 m
0
Alitas D`L
Cadillac Granados
UDLA Granados
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 377.48 m (1,238.44 pies)
Tienda
para el
Almacen


Vía
Vía a Nayón
Vía a Nayón
Vía a Nayón
Vía a Nayón
63.16 m
50.00 m
Nuestra Senora de Santa Ana Y E16
UDLA Sede Udlapark
Udlapark
Universidad De Las
rk
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 63.16 m (207.23 pies)


Camino Antiguo a Nayón
Camino Antiguo a Nayón
Vía a Nayón
255.66 m
Vía a Nayón
NUESTRA SEnORA DE SANTA ANA Y E16 (2)
Vía a Nayón
Vía a Nayón
Vía a Nayón
200.00 m
150.00 m
100.00 m
50.00 m
Avianca Ecuador Matriz
Nuestra Senora de Santa Ana Y E16
UDLA Sede Udlapark
Udlapark
EKOPARK
Universidad De Las Américas Udlapark
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 255.66 m (838.77 pies)


Leonardo Tejada Y E16
Vía a Nayón
Camilo Egas
Camino Antiguo a Nayón
400.00 m
300.00 m
200.00 m
100.00 m
Complejo Corporativo EkoPark Torre 1...
NESTLE ECUADOR S.A
Ekopark
Via A Nayon (UDLA)
Luis Roman Perez Y E16 (3)
Produbanco
EKOPARK
UDLA Sede Udlapark
Av. Simón Bolívar
Luis Roman Perez Y E16 (2)
Universidad De Las Américas Udlapark
Av. Simón Bolívar
Av Libertador Simón Bolívar
Av. Simón Bolívar
Medir la distancia
Haz clic en el mapa para ampliar la ruta.
Distancia total: 443.18 m (1,454.01 pies)

ANEXO 8

Reporte de reunión de presentación de la aplicación y pruebas en las instalaciones del Sistema Integrado de Seguridad ECU911.

ECU 911	PLANIFICACIÓN Y GESTIÓN ESTRATÉGICA	
	GESTIÓN DE SERVICIOS, PROCESOS Y CALIDAD	
	ACTA DE REUNIÓN	
	CÓDIGO	VERSIÓN
	SGC_FOR_18	00

Fecha: 30 Noviembre 2018	**Hora:** 10:00	**Lugar:** Instalaciones ECU 911
Tema: Prueba en ambiente de produccion Demo Aplicativo Smartphone Android Funcionalidad Acceso Rápido		
N° ACTA:		

ORDEN DEL DIA

* Instalacion de Aplicativo Demo en Dispositivo Android
* Envio de Alerta mediante la funcionalidad de Acceso Rapido
* Verificacion de la recepcion de la alerta en Sala Operativa

DESARROLLO

1. Se instalo el Aplicativo Demo en un dispositivo seleccionado (Solo Android)
2. Se procedio al envio de una alerta con la funcionalidad acceso rápido que son: presionar cuatro veces el boton de encendido
3. Se procedio a verificar la recepcion de la alerta emitida por el smartphone, donde constatamos que el operador de llamadas recibio la alerta en el sistema de sala operativa y procedio a devolver la llamada y registrar en el sistema de sala operativa esta alerta de forma exitosa.

Todo esto se realizo en presencia del Sr. director Tenis y alumnos VDLA en conjunto con personal del ECU911.

COMPROMISOS

RESPONSABLE	ACCIONES	FECHA LIMITE DE CUMPLIMIENTO
Miguel Barquera / Marcos Gavela	Buscar la forma de implementar una funcionalidad semejante para la plataforma IOS (Apple - iPhone)	———
		———
		———

	PLANIFICACION Y GESTION ESTRATEGICA GESTION DE SERVICIOS, PROCESOS Y CALIDAD LISTADO DE ASISTENCIA	
	CODIGO SGE_FOR_17	VERSION 00

Prueba en ambiente de produccion Demo Aplicativo Smartphone Android con funcionalidad Acceso Aspirante 30 Noviembre 2018, 10:00 horas

Ord.	NOMBRES	INSTITUCION	CARGO	TELEFONO	EMAIL	FIRMA
1	Miguel Ángel Baquero	UDLA	estudiante	0983511126	miguel.baquero@udla.edu.ec	Migu
2	Marcos Gaudo	UDLA	Estudiante	0969055363	marcos.gaudo@udla.edu.ec	
3	Marco Chuncho	ECU911	Analista	0988370643	marco.chuncho@ecu911.gob.ec	
4	Diego Meuña	ECU911	consultor DNO	0983249394	diego.meuna@ecu911.gob.ec	
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						

ANEXO 9

Fotografía con el personal del Servicio Integrado de Seguridad ECU911 con fecha 30 de noviembre de 2018

ANEXO 10

Hoja de datos de la videocámara Hikvision DS-2DE4425IW-DE

HIKVISION

DS-2DE4425IW-DE
4MP 25× Network IR Speed Dome

Hikvision DS-2DE4425IW-DE 4MP 25× Network IR Speed Dome adopts 1/2.5" progressive scan CMOS chip. With the 25× optical zoom lens, the camera offers more details over expansive areas.

This series of cameras can be widely used for wide ranges of high-definition, such as the rivers, roads, railways, airports, squares, parks, scenic spots, and venues, etc.

Key Features

- 1/2.5" progressive scan CMOS
- Up to 2560 × 1440@30fps resolution
- Min. illumination:
  Color: 0.005 Lux @(F1.6, AGC ON)
  B/W: 0.001 Lux @(F1.6, AGC ON)
  0 Lux with IR
- 25× optical zoom, 16× digital zoom
- WDR, HLC, BLC, 3D DNR, Defog, EIS, Regional Exposure, Regional Focus
- Up to 100 m IR distance
- 12 VDC & PoE+ (802.3 at, class 4)
- Support H.265+/H.265 video compression

www.hikvision.com

Specification

Camera Module	
Image Sensor	1/2.5" progressive scan CMOS
Min. Illumination	Color: 0.005 Lux @(F1.6, AGC ON) B/W: 0.001Lux @(F1.6, AGC ON) 0 Lux with IR
White Balance	Auto/Manual/ATW (Auto-tracking White Balance)/Indoor/Outdoor/Fluorescent Lamp/Sodium Lamp
Gain	Auto/Manual
Shutter Time	50Hz: 1/1 s to 1/30,000 s 60Hz: 1/1 s to 1/30,000 s
Day & Night	IR Cut Filter
Digital Zoom	16×
Privacy Mask	24 programmable privacy masks
Focus Mode	Auto/Semi-automatic/Manual
WDR	120 dB WDR (not supported by full frame rate)
Lens	
Focal Length	4.8 mm to 120 mm, 25× optical zoom
Zoom Speed	Approx. 3.6 s (optical lens, wide-tele)
Field of View	Horizontal field of view: 59.5° to 2.6° (Wide-Tele) Vertical field of view: 35.7° to 1.5° (Wide-Tele) Diagonal field of view: 66.6 to 3.0° (Wide-Tele)
Working Distance	10 mm to 1500 mm (wide-tele)
Aperture Range	F1.6 to F3.5
IR	
IR Distance	100 m
Smart IR	Support
PTZ	
Movement Range (Pan)	360° endless
Pan Speed	Configurable, from 0.1°/s to 80°/s, Preset speed: 80°/s
Movement Range (Tilt)	From -15° to 90° (auto-flip)
Tilt Speed	Configurable, from 0.1°/s to 80°/s Preset Speed: 80°/s
Proportional Zoom	Support
Presets	300
Patrol Scan	8 patrols, up to 32 presets for each patrol
Pattern Scan	4 pattern scans, record time over 10 minutes for each scan
Power-off Memory	Support
Park Action	Preset/Pattern Scan/Patrol Scan/Auto Scan/Tilt Scan/Random Scan/Frame Scan/Panorama Scan
3D Positioning	Support
PTZ Position Display	Support

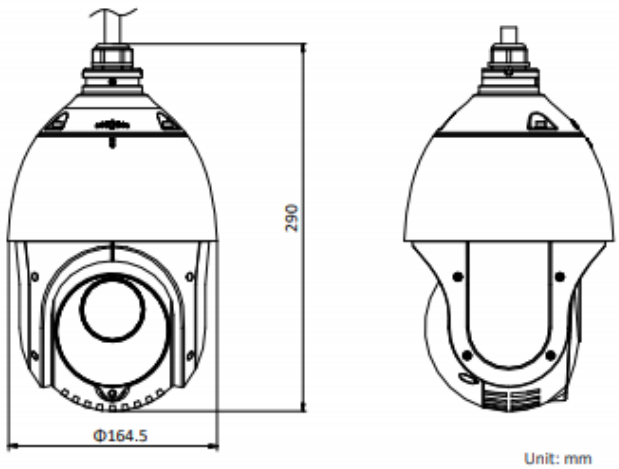
Preset Freezing	Support
Scheduled Task	Preset/Pattern Scan/Patrol Scan/Auto Scan/Tilt Scan/Random Scan/Frame Scan/Panorama Scan/Dome Reboot/Dome Adjust/Aux Output
Compression Standard	
Video Compression	Main Stream: H.265+/H.265/H.264+/H.264 Sub-stream: H.265/H.264/MJPEG Third Stream: H.265/H.264/MJPEG
H.264 Type	Baseline Profile/Main Profile/High Profile
H.264+	Support
H.265 Type	Baseline Profile/Main Profile/High Profile
H.265+	Support
Video Bitrate	32 Kbps to 16384 Kbps
Audio Compression	G.711alaw/G.711ulaw/G.722.1/G.726/MP2L2/PCM
Audio Bitrate	G.711alaw/G.711ulaw: 64 Kbps G.722.1/G.726: 16 Kbps MP212/PCM: 32 Kbps/64 Kbps/128 Kbps
SVC	Support
Smart Features	
Basic Event	Motion Detection, Video Tampering Detection, Exception
Smart Event	Intrusion Detection, Line Crossing Detection, Region Entrance Detection, Region Exiting Detection, Object Removal Detection, Unattended Baggage Detection, Audio Exception Detection
Smart Record	ANR (Automatic Network Replenishment), Dual-VCA
ROI	Main stream, sub-stream and third stream respectively support four fixed areas.
Image	
Max. Resolution	2560 × 1440
Main Stream	50Hz: 25fps (2560 × 1440, 2048 × 1536, 1920 × 1080, 1280 × 960, 1280 × 720) 50fps (1920 × 1080, 1280 × 960, 1280 × 720) 60Hz: 30fps (2560 × 1440, 2048 × 1536, 1920 × 1080, 1280 × 960, 1280 × 720) 60fps (1920 × 1080, 1280 × 960, 1280 × 720)
Sub-Stream	50Hz: 25fps (704 × 576, 640 × 480, 352 × 288) 60Hz: 30fps (704 × 480, 640 × 480, 352 × 240)
Third Stream	50Hz: 25fps (1920 × 1080, 1280 × 960, 1280 × 720, 704 × 576, 640 × 480, 352 × 288) 60Hz: 30fps (1920 × 1080, 1280 × 960, 1280 × 720, 704 × 480, 640 × 480, 352 × 240)
Image Enhancement	HLC/BLC/3D DNR/Defog/EIS/Regional Exposure/Regional Focus
Network	
Network Storage	Built-in memory card slot, support Micro SD/SDHC/SDXC, up to 256 GB; NAS (NPS, SMB/CIPS), ANR
Protocols	IPv4/IPv6, HTTP, HTTPS, 802.1x, Qos, FTP, SMTP, UPnP, SNMP, DNS, DDNS, NTP, RTSP, RTCP, RTP, TCP/IP, UDP, IGMP, ICMP, DHCP, PPPoE, Bonjour
API	Open-ended, support ONVIF, ISAPI, and CGI, support HIKVISION SDK and Third-Party Management Platform
Simultaneous Live View	Up to 20 channels

User/Host	Up to 32 users 3 levels: Administrator, Operator and User
Security Measures	User authentication (ID and PW), Host authentication (MAC address); HTTPS encryption; IEEE 802.1x port-based network access control; IP address filtering
Client	iVMS-4200, iVMS-4500, iVMS-5200, Hik-Connect
Web Browser	IE 8 to 11, Chrome 31.0 to 44, Firefox 30.0 to 51
Interface	
Audio Interface	1-ch audio input and 1-ch audio output
Network Interface	1 RJ45 10 M/100 M Ethernet, PoE+ (802.3 at, class4)
General	
Power	12 VDC and PoE+ (802.3 at, class4) Max.: 18 W (Max. 7W for IR)
Working Temperature	-30°C to 65°C (-22°F to 149°F)
Working Humidity	≤ 90%
Protection Level	IP66 Standard, TVS 4000V Lightning Protection, Surge Protection and Voltage Transient Protection
Material	ADC 12, PC+10% GF
Dimensions	Φ 164.5 mm × 290 mm (Φ 6.48" × 11.42")
Weight	Approx. 2 kg (4.41 lb)

Available Model

DS-2DE4425IW-DE 12 VDC & PoE+ (802.3 at, class 4)

Dimensions



Unit: mm

Accessories

	DS-1618ZJ	Wall Mounting Bracket
	DS-1602ZJ	Wall Mounting Bracket
	DS-1602ZJ-corner	Corner Mounting Bracket
	DS-1602ZJ-pole	Vertical Pole Mounting Bracket
	DS-1602ZJ-box	Wall Mounting Bracket with Junction Box
	DS-1671ZJ-SDM9	In-ceiling Mounting Bracket
	DS-1663ZJ	Ceiling Mounting Bracket
	DS-1661ZJ	Pendant Mounting Bracket
	DS-1662ZJ	Pendant Mounting Bracket
	DS-1619ZJ	Gooseneck Mounting Bracket
	DS-1100KI	Network Keyboard
	DS-1005KI	USB Joy-stick

Headquarters
No.555 Qianmo Road, Binjiang District,
Hangzhou 310051, China
T +86-571-8807-5998
overseasbusiness@hikvision.com