



**UNIVERSIDAD CÉSAR VALLEJO**

**FACULTAD DE INGENIERÍA Y ARQUITECTURA  
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

Aplicación móvil de asignación de ambulancias para el traslado de  
pacientes de Emergencia

**TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE:**  
Ingeniero de Sistemas

**AUTOR:**

Huaroto Quispe, Julio Cesar (ORCID: 0000-0002-4412-8183)

**ASESOR:**

Dr. Alfaro Paredes, Emigdio Antonio (ORCID: 0000-0002-0309-9195)

**LÍNEA DE INVESTIGACIÓN:**

Sistema de información y comunicaciones

LIMA – PERÚ

2020

### **Dedicatoria**

Dedico este proyecto a mi hija Valentina, mi mayor motivación en mi vida encaminada al éxito. Ella fue el ingrediente perfecto para poder culminar esta etapa importante de mi vida.

## **Agradecimiento**

Expreso mi agradecimiento a la Universidad César Vallejo, por educar a los peruanos, fortaleciendo y formando profesionales con calidad.

A mi asesor Dr. Emigdio Antonio Alfaro Paredes, ya que con su guía y apoyo se logró superar los problemas que hubo en este largo tiempo.

## Índice de contenidos

<b>I. INTRODUCCIÓN</b> .....	1
<b>II. MARCO TEÓRICO</b> .....	8
<b>III. MÉTODO</b> .....	16
3.1 Tipo y diseño de la investigación.....	17
3.2 Operacionalización de variables.....	18
3.3 Población, muestra y muestreo .....	20
3.4 Técnicas e instrumentos de recolección de datos, validez y confiabilidad.....	21
3.5 Procedimientos .....	22
3.6 Método de análisis de datos .....	22
3.7 Aspectos éticos.....	22
<b>IV. RESULTADOS</b> .....	24
<b>4.1 Prueba de la hipótesis específica 1</b> .....	25
4.1.1 Con 75 solicitudes .....	25
4.1.2 Con 100 solicitudes.....	29
<b>4.2 Prueba de la hipótesis específica 2</b> .....	34
4.2.1 Con 75 solicitudes.....	34
4.2.2 Con 100 solicitudes.....	34
<b>4.3 Prueba de la hipótesis general</b> .....	34
<b>4.4 Resumen</b> .....	36
<b>V. DISCUSIÓN</b> .....	38
<b>VI. CONCLUSIONES</b> .....	41
<b>VII. RECOMENDACIONES</b> .....	43
<b>REFERENCIAS</b> .....	45

## Índice de tablas

Tabla 1.	Matriz de Operacionalización de las variables .....	19
Tabla 2.	Resultados de la simulación de 75 solicitudes de emergencia. ....	26
Tabla 3.	Datos estadísticos de la prueba de normalidad de Kolmogorov–Smirnov. ....	28
Tabla 4.	Datos estadísticos de la prueba de comprobación de Wilcoxon. ....	28
Tabla 5.	Datos estadísticos de la prueba Z.....	29
Tabla 6.	Resultados de la simulación de 100 solicitudes de emergencia. ....	30
Tabla 7.	Datos estadísticos de la prueba de normalidad Kolmogórov-Smirnov.....	32
Tabla 8.	Datos estadísticos de la prueba comprobación Wilcoxon .....	33
Tabla 9.	Datos estadísticos de la prueba Z.....	33
Tabla 10.	Datos estadísticos de la prueba de normalidad de Kolmogorov – Smirnov. ....	35
Tabla 11.	Datos estadísticos de la prueba de comprobación de Wilcoxon .....	35
Tabla 12.	Datos estadísticos de la prueba Z.....	35
Tabla 13.	Síntesis de los resultados de las comprobaciones de las hipótesis: .....	36
Tabla 14.	Matriz de consistencia.....	54
Tabla 15.	Requerimientos funcionales.....	65
Tabla 16.	Requerimientos no funcionales .....	66
Tabla 17.	Modelo de procesos.....	66
Tabla 18.	Datos técnicos de laptop Acer.....	67
Tabla 19.	Datos técnicos de laptop Toshiba .....	67
Tabla 20.	Datos técnicos de móvil Samsung.....	68
Tabla 21.	Datos técnicos de móvil Lenovo.....	68
Tabla 22.	Storycard identificación de usuario.....	69
Tabla 23.	Storycard modulo menú principal.....	69
Tabla 24.	Storycard Solicitud de ambulancia .....	70
Tabla 25.	Storycard simular .....	70
Tabla 26.	Storycard generar solicitudes.....	71
Tabla 27.	Descripción de la tabla Ambulancia .....	72
Tabla 28.	Descripción de la tabla Estado .....	72
Tabla 29.	Descripción de la tabla Alergia .....	72
Tabla 30.	Descripción de la tabla Perfil.....	72
Tabla 31.	Descripción de la tabla Solicitud. ....	73
Tabla 32.	Descripción de la tabla Tipo_emergencia. ....	73
Tabla 33.	Descripción de la tabla Ubicación. ....	73
Tabla 34.	Descripción de la tabla Usuario.....	73
Tabla 35.	Descripción de la tabla Usuario_alergia.....	74
Tabla 36.	Recomendaciones para el equipo móvil. ....	90
Tabla 37.	Recomendaciones para la conexión al servidor. ....	90
Tabla 38.	Prueba unitaria 1- Módulo de identificación de usuario .....	91
Tabla 39.	Prueba unitaria 2 - Módulo menú principal.....	91
Tabla 40.	Prueba unitaria 3 - Módulo solicitud de ambulancia .....	92
Tabla 41.	Prueba unitaria 4 - Módulo simulación.....	92
Tabla 42.	Prueba unitaria 5 - Módulo generación de solicitudes .....	93

## Índice de figuras

Figura 1. Atenciones realizadas por el SAMU entre los años 2017 y 2018 (Arenas, 2019).....	20
Figura 2. Fases y etapas de la metodología Mobile -D.....	63
Figura 3. Entorno de funcionamiento del App.....	68
Figura 4: Modelo relacional de la base de datos. ....	71
Figura 5. Pantalla de identificación de usuario. ....	74
Figura 6. Pantalla del menú principal.....	76
Figura 7. Pantalla del módulo solicitud. ....	78
Figura 8. Pantalla del módulo simulación. ....	81
Figura 9: Pantalla del módulo generación de solicitudes. ....	86

## Índice de anexos

Anexo 1: Matriz de consistencia .....	54
Anexo 2. Algunas teorías relacionadas sobre emergencias y urgencias.....	55
Anexo 3. Algunas teorías relacionadas sobre algoritmos .....	56
Anexo 4. Algunas teorías relacionadas a las tecnologías y aplicaciones móviles .....	57
Anexo 5. Algunas teorías relacionadas sobre IDE de desarrollo Android Studio.....	58
Anexo 6. Algunas teorías relacionadas de la Api de Google Maps .....	59
Anexo 7. Metodologías de desarrollo .....	60
Anexo 8. Etapas de la metodología Mobile-D para el desarrollo de la aplicación de asignación de ambulancias para el traslado de pacientes de emergencia. ....	64
Anexo 9: Diccionario de la base de datos. ....	72

## Índice de abreviaturas

<b>Sigla</b>	<b>Significado</b>	<b>Pág.</b>
XP	Extreme Programming	59
API	Application Program Interface (Interfaz de programación de aplicaciones)	17,40,41,56,62,89
SPSS	Statistical Package for the Social Sciences	27
RUP	Rational Unified Process	10,59
UML	Unified Modeling Language (Lenguaje Unificado Modelado)	10



## Resumen

El problema de la investigación fue: ¿Cuál sería el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia? El objetivo general de la investigación fue determinar el impacto que tendría la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia. La investigación realizada fue de tipo aplicada, de enfoque cuantitativo y diseño no experimental, transversal y descriptivo.

El desarrollo de la aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia fue posible gracias a la revisión de fuentes confiables con los indicadores planteados dentro de la investigación, con la finalidad de determinar qué impacto generaría la implementación de una aplicación móvil de solicitud de ambulancias para el traslado de pacientes de emergencia. Se aplicó la metodología ágil Mobile D, ya que está muy centrada en las plataformas móviles, tanto para la gestión del proyecto como para el desarrollo de la aplicación.

Los resultados de la investigación mostraron que con el algoritmo planteado se tuvo un porcentaje de emergencias atendidas del 93%, lo que fue superior a los estudios anteriores encontrados. Además, los resultados evidenciaron un tiempo de respuesta promedio de 18.42 minutos, lo que redujo el 45.35% del tiempo con respecto al mejor estudio previo encontrado. Por lo tanto, se demostró que el uso de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia aumentaría el porcentaje de emergencias atendidas y reduciría los tiempos de respuesta. Se recomendó para futuras investigaciones que se amplíe el rango de latitud y longitud de la muestra para la simulación de solicitudes masivas y que se implemente la aplicación móvil en una entidad de salud del Minsa o de Essalud para poder realizar pruebas con usuarios y ambulancias en tiempo real.

**Palabras claves:** Asignación de ambulancias, traslado de pacientes, ambulancia de emergencia, sistema de emergencia, tiempo de asignación de ambulancias

## **Abstract**

The research problem was: What would be the impact of implementing a mobile ambulance assignment application for emergency patient transport? The general objective of the research was to determine the impact of the implementation of a mobile ambulance assignment application for the transfer of emergency patients. The research was applied, quantitative, non-experimental, cross-sectional and descriptive.

The development of the mobile application for the assignment of ambulances for the transfer of emergency patients was possible thanks to the review of reliable sources with the indicators proposed in the research, in order to determine the impact of the implementation of a mobile application for the request of ambulances for the transfer of emergency patients. The Mobile D agile methodology was applied, since it is very focused on mobile platforms, both for project management and application development.

The results of the research showed that with the algorithm proposed there was a percentage of emergencies attended of 93%, which was higher than previous studies found. In addition, the results showed an average response time of 18.42 minutes, which reduced 45.35% of the time with respect to the best previous study found. Therefore, it was demonstrated that the use of a mobile ambulance assignment application for the transfer of emergency patients would increase the percentage of emergencies attended and reduce response times. It was recommended for future research that the range of latitude and longitude of the sample be extended for the simulation of massive requests and that the mobile application be implemented in a Minsa or Essalud health entity to be able to test with users and ambulances in real time.

**Keywords:** Ambulance assignment, patient transfer, emergency ambulance, emergency system, ambulance assignment time.

## **I. INTRODUCCIÓN**

En este capítulo se desarrolló la realidad problemática, en la que se manifestó la falta de aplicaciones móviles para la asignación de ambulancias por nivel de emergencia y diferencia espacio - tiempo en el traslado de pacientes de emergencia. También se explicó las justificaciones de la investigación, así como los problemas, los objetivos y las hipótesis de la investigación.

Los especialistas del Ministerio de Salud (2017) indicaron que actualmente en el Perú se ha identificado la problemática siguiente: “la subsistencia de alto promedio de letalidad por urgencias y emergencias médicas individuales o masivas” (p. 2). Al respecto, Ortega (2018) indicó que años anteriores era enrevesado demandar ayuda ante una emergencia y que la ayuda solo arribaba por parte de representantes sociales o entidades comprometidas en proteger la vida en riesgo de las personas afectadas, como la policía, bombero o serenazgo (p. 2). Además, Ortega (2018) explicó que este tipo de unidades en el deseo de apoyar realizaban traslados totalmente insuficientes, debido a la poca experiencia, la falta de infraestructura de las unidades y la ausencia de gestión entre las organizaciones (p. 2).

Ríos (2019) explicó que en los estándares internacionales el tiempo de respuesta van a partir de los 5 hasta 20 minutos (p. 1). Además, Ríos (2019) agregó que en la ciudad de Lima se están operando tiempos de respuesta que van a partir de los 5 hasta los 40 minutos y que actualmente hay factores que interfieren para determinar este tiempo y por consiguiente el impacto final (p. 1). Al respecto, Isong et al. (2016) indicaron que una prestación eficaz y eficiente de servicios públicos como los servicios de salud son muy importantes en la actualidad (p. 14). La finalidad general de este estudio fue establecer el impacto que tendría poner en efecto una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia, teniendo como resultado la comprobación de dos hipótesis específicas: disminuir el tiempo de respuesta y aumentar la cantidad de emergencias atendidas.

En esta sección se detalló artículos y tesis previos relacionados al desarrollo de aplicaciones móviles para el traslado de pacientes de emergencia. También, se logró reconocer que ninguno de los estudios previos encontrados se centró en llevar a cabo una aplicación móvil con la API de Google Maps para

la optimización de rutas y la creación de un algoritmo diferencial entre el nivel de emergencia, el espacio y el tiempo.

Castañeda y Bareño (2015) explicaron que una vez que se logra solicitar el servicio de ambulancia, esta presenta demoras debido a que al conductor no le es fácil encontrar la dirección que se le indicó (p. 3). Igualmente, Castañeda y Bareño (2015) agregaron que otro punto importante es que se asigna a varias ambulancias el mismo caso, por lo que al sitio del accidente llegan al mismo tiempo más de una ambulancia, lo que genera gastos innecesarios y tiempo perdido, haciendo que otras solicitudes estén en espera (p. 4). Asimismo, Castañeda y Bareño (2015) determinaron que la implementación de esta aplicación optimizaría significativamente los tiempos de respuesta en la asignación de ambulancias (p. 71).

Los especialistas del Ministerio de Salud (2017) indicaron que entre los años 2007 y 2012 el traslado por una ambulancia de los organismos de salud o bomberos coinciden una mínima segmentación del total de traslados (p. 14). Al respecto, el Ministerio de Salud (2017) detalló que las lesiones graves corresponden el 11.47% y potencialmente graves el 16.49%, lo que significa que el 88.53% de los heridos graves y el 83.51% de los heridos potencialmente graves son transportados en circunstancias que no certifican el transporte socorrido apropiado por organismos distintos como lo son: el transporte particular, las unidades del serenazgo, las patrullas de la policía y otros que acrecienten la gravedad de las lesiones y la mortalidad (p. 14).

Alvarez y Andrómaco (2016) explicaron que el tiempo de respuesta es un señalizador de evolución de servicio que se usa en las decisiones generales, concede conocer la desenvolvura y la productividad de cada equipo de trabajo y posiciona a planeamientos logísticos para la ubicación de centrales operativas y número de móviles en el rango del día, mes y año; asimismo, se considera un indicador de calidad asistencial por la captación de satisfacción del paciente (p. 50). Bedoya y Laurente (2014) concluyeron que la solución proporcionaba un trayecto óptimo; a pesar de ello, el asunto no se resuelve con ello por la afluencia de vehículos y obstáculos en las calles (p. 7).

En esta sección se expuso las justificaciones de la investigación con la intención de producir un debate académico sobre el entendimiento existente dentro de este estudio, así como la incorporación de instrumentos tecnológicos que faciliten el traslado de pacientes con emergencia. Al respecto, esta investigación contiene el estudio, desarrollo y la creación de una aplicación móvil de solicitud de ambulancias para el traslado de pacientes de emergencia.

La justificación teórica se sustenta a través de estudios como los de Alvarez y Andrómaco (2016), quienes señalaron que las grandes demoras localizadas suceden en las etapas de despacho y partida de la unidad móvil que emplean aproximadamente la mitad del tiempo de respuesta en general, siendo estos tiempos desmedidos para las normas establecidas (p. 55). Asimismo, Isong et al. (2016) indicaron que los desafíos van desde una comunicación deficiente, una red de carreteras deficiente y una dirección no estructurada causando que no lleguen las ambulancias, lo que conlleva pérdidas de vidas que se pueden prevenir (p. 14).

Adicionalmente, Isong et al. (2016) consideraron a los pacientes les resultaba difícil o imposible acceder al transporte médico en ambulancia durante las situaciones de emergencia, lo que provocaba las pérdidas de vidas que podían evitarse (p. 21). Del mismo modo, Ortega (2018) indicó que el periodo de respuesta de estos sucesos de emergencia y el aspecto en cómo serán tomados estos eventos, donde la vida del accidentado está en peligro no son los más esperados y que están diversos incidentes externos e internos que retrasan el apoyo (p. 2).

Con respecto a la justificación metodológica, este estudio refuerza la importancia y la oportunidad de las aplicaciones móviles. Al respecto, Ramírez (2011) precisó que la finalidad es conseguir periodos de avances vertiginosos con una cantidad de personal muy reducida, máximo 10 desarrolladores, laborando en una misma área física y con ese procedimiento se deben obtener productos plenamente útiles en un plazo no mayor a 10 semanas (p. 42).

La justificación tecnológica se fundamenta con investigaciones como las de Santamaría y Hernández (2015) en la que indicaron que en los años anteriores se ha evidenciado la evolución permanente en las tecnologías, con

una utilización progresiva de dispositivos móviles en la comunidad, accediendo a informes en todo momento, incorporados en asuntos de salud (p. 599). Al respecto, Castañeda y Bareño (2015) agregaron que la utilización de las tecnologías de la información y comunicación admite apresurar todo tipo de procesos reduciendo los tiempos de respuesta, otorgando una mayor concisión a los usuarios y posibilitando el paso al servicio de ambulancia a través de dispositivos móviles (p. 71).

Asimismo, Santamaría y Hernández (2015) concluyeron que los trabajadores de la salud gracias a los avances tecnológicos tienen acceso a varias herramientas de diagnóstico, las cuales le conceden un mayor acceso a la información y por ende brindar un mejor control de las enfermedades de los pacientes (p. 605). También, Martínez (2017) indicó que mediante el uso de soluciones tecnológicas fundamentadas en la informática y en las telecomunicaciones se pueden mejorar significativamente los sistemas de transporte terrestre (p. 2).

Considerando la realidad problemática mostrada se proyectó el problema general y los problemas específicos del estudio. Asimismo, el problema general de este estudio fue: ¿Cuál sería el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia? Los temas específicos de la investigación son los siguientes:

- **PE1:** ¿Cuál sería el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia en el tiempo de respuesta?
- **PE2:** ¿Cuál sería el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia en el número de emergencias atendidas?

El objetivo general fue determinar el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia. Los objetivos específicos son los siguientes:

- **OE1:** Determinar el impacto que tendría la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia en el tiempo de respuesta.

- **OE2:** Determinar el impacto que tendría la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes en el número de emergencias atendidas.

La hipótesis general de estudio fue la siguiente: “La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia reduciría el tiempo de respuesta y aumentaría el número de emergencias atendidas”. Castañeda y Bareño (2015) indicaron que utilización de novedades tecnológicas para el desarrollo de software permite crear instrumentos de optimización que pueden ser muy útiles para la solicitud de servicios médicos como es la ambulancia (p. 71). Al respecto, Santamaría y Hernández (2015) precisaron que con la manifestación de estas novedades tecnológicas, su congestión, continuo perfeccionamiento y creación modifican nuestras vidas, sin que se pueda ser extraño a estas modificaciones y proporcionan potenciales ventajas para mejoras en la salud (p. 605). Las hipótesis específicas fueron las siguientes:

- La primera hipótesis específica fue: “La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia reduciría el tiempo de respuesta”. Castañeda y Bareño (2015) precisaron que el uso de las tecnologías de la información y comunicación permite agilizar todo tipo de procesos minimizando los tiempos de respuesta de los mismos, dando una mayor brevedad a los usuarios (p. 71) y en este caso se facilita el acceso al servicio de ambulancia por medio de dispositivos móviles.
- La segunda hipótesis específica fue: “La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia aumentaría el número de emergencias atendidas”. Bedoya y Laurente (2014) precisaron que la base principal de emergencia verifica el estatus de las unidades ambulatorias para detectar la disposición de estas y la base principal destina automáticamente la alerta a la unidad disponible (p. 28).  
Bedoya y Laurente (2014) indicaron que la asignación se ejecuta mediante un algoritmo por un programa interno que considera la



distancia que tiene la unidad ambulatoria hasta el lugar de la emergencia, el tiempo que se retrasara en llegar y la hora de llegada de la alerta (p. 28). Asimismo, Castañeda y Bareño (2015) crearon una aplicación móvil que permitiría solicitar una unidad ambulatoria considerando la disponibilidad y la distancia al punto del usuario en situación de emergencia. Cabe resaltar que el traslado de personas heridas solicita condiciones especiales de alto riesgo y es por ello que con el desarrollo de esta aplicación se aspiró a brindar un instrumento para que el servicio sea competente y efectivo (Castañeda y Bareño, 2015, p. 71).

## **II. MARCO TEÓRICO**

En este capítulo se puntualiza los estudios de aplicaciones móviles para la solicitud de ambulancias en tres etapas: trabajos previos, teorías relacionadas y marco conceptual, basados en una investigación minuciosa de fuentes confiables. En los estudios previos se especificó cada uno de ellos dentro de los diferentes campos con la finalidad de sustentar la originalidad de la aplicación planteada. Con respecto a las teorías relacionadas, se describe las diferentes teorías que apoyan al estudio a reforzar los conceptos sobre los algoritmos y las aplicaciones móviles aplicadas en el sector salud. Finalmente, se especifica los diferentes orígenes que sirvieron como contexto conceptual a este estudio alusivo a las variables y medidas planteadas.

Este apartado del capítulo comprendió a los trabajos previos, donde se incorporó estudios relevantes acerca de la implementación de aplicaciones móviles de solicitud de ambulancias para el traslado de pacientes de emergencia. Estos estudios posibilitaron contrastar los resultados obtenidos en este proyecto bajo los indicadores planteados con las conclusiones de estudios anteriores en los que se investigó acerca del impacto de la implementación de aplicaciones móviles.

Bedoya y Laurente (2014) proyectaron y ejecutaron un sistema central de gestión de unidades ambulatorias para una empresa de seguros en Lima, utilizando un algoritmo desarrollado para asignar automáticamente las ambulancias a los diferentes puntos de emergencia tomando en cuenta prioridades como el turno de llegada de la alerta, el tiempo y la distancia (p. 18). Además, Bedoya y Laurente (2014) utilizaron la metodología Scrum para la gestión del proyecto, debido a que solo se involucraron dos personas para realizar las pruebas y obtener los resultados por completo en un corto plazo; asimismo, para el desarrollo del producto se utilizó la metodología Extreme Programming posibilitando reconocer los desaciertos y mejorarlos de antemano a incorporar una nueva funcionalidad (p. 23).

Al respecto, Bedoya y Laurente (2014) concluyeron que la solución brindaba una ruta óptima; sin embargo, el problema no se solucionaba con ello por diversos factores como el tráfico y obstáculos en las calles (p. 61). Por lo tanto, se optó por optimizar cada actividad en la que estaba involucrado en el proceso como es el envío de alertas, asignación de alertas con ambulancias,

visualización de centros médicos y registro de resumen de historia clínica (Bedoya y Laurente, 2014, p. 61).

Castañeda y Bareño (2015) diseñaron un prototipo de aplicación móvil que permitió el acceso a una ambulancia de manera rápida y eficiente, para las personas que se encuentren en estado de emergencia a causa de un accidente (p. 2). Además, Castañeda y Bareño (2015) indicaron que utilizaron la metodología RUP (Rational Unified Process), la metodología estándar más empleada para el análisis, la implementación y la documentación de sistemas orientados a objetos, soportando ordenar todos los procesos y calibrar la eficacia de la agrupación (p. 30).

Castañeda y Bareño (2015) determinaron que con el desarrollo de este proyecto se puede mejorar significativamente los tiempos de respuesta en la asignación de una ambulancia y mejorar el servicio de la atención, ya que se cuenta con un chat donde expone la situación en la que se encuentra el usuario, y se visualiza la posición exacta donde está el afectado (p. 71). Al respecto, Castañeda y Bareño (2015) desarrollaron un sistema para la solicitud de ambulancias, capaz de sostener un crecimiento individual de cada uno de los elementos implementados: servicio web, servidor de aplicaciones, base de datos y aplicaciones móviles (p. 71).

Isong et al. (2016) implementaron un prototipo de sistema utilizando una aplicación móvil para ofrecer servicios rentables a los pacientes durante las emergencias, cuyo objetivo fue reducir las largas colas en hospitales y largos períodos de espera para una ambulancia a través de servicios basados en la localización (p. 14). Al respecto, Isong et al. (2016) utilizaron el algoritmo de Dijkstra, permitiendo equipar el sistema con la capacidad de manejar varias reservas y muchas situaciones de emergencia en las que hay varias ambulancias solicitadas, pero solo pocas están disponibles (p. 19).

Isong et al. (2016) determinaron que el algoritmo de Dijkstra ofreció la función de calcular los trayectos más cortos a la ubicación de los pacientes, con el fin de garantizar que todos los pacientes sean atendidos en el menor tiempo posible independientemente desde que aldea lo hayan solicitado (p. 19).

Chen et al. (2014) desarrollaron la aplicación “Agente de operación y de carreteras” para completar el proceso de guía de ruta dinámica para un vehículo de emergencia (p. 848). Al respecto, Chen et al. (2014) investigaron la ruta de orientación mediante el uso de tecnología multiagente combinado con el algoritmo mejorado de Dijkstra brindando cierta superioridad a los problemas de tráfico (p. 848).

Chen et al. (2014) describieron que este documento estudió cómo encarrilar la guía para un vehículo de emergencia basado en el dominio de precedencia de la señal en las intersecciones bajo la propiedad variable tiempo, el lenguaje de la teoría de grafos y el algoritmo mejorado de Dijkstra (p. 849). Al respecto, Chen et al. (2014) determinaron que al utilizar el software de simulación Starlogo con la ayuda de Matlab para simular el proceso de guía de ruta de un vehículo de emergencia basado en la demora de las intersecciones brinda la viabilidad y eficacia del método dado (p. 851). Asimismo, Chen et al. (2014) agregaron que los datos artificiales en este experimento se obtuvieron de velocidades promedio de vehículos en carreteras que se generaron aleatoriamente, donde se tomó el tiempo total de viaje como índice de evaluación (p. 851).

Wen-mei et al. (2015) propusieron un modelo matemático multiobjetivo basado en las teorías de la racionalidad delimitada, la seguridad de la ruta se modela como producto a través de arcos incluidos en el trayecto (p. 1). Además, Wen-mei et al. (2015) agregaron que el objetivo del modelo de planificación de rutas existente era reducir al mínimo la transmisión logística; sin embargo, la seguridad de la ruta también debe tenerse en cuenta como objetivo del modelo de planificación de rutas en caso de emergencia gestión de la logística teniendo en cuenta la vulnerabilidad de los humanos en tiempos de desastre (p. 1).

Además, Wen-mei et al. (2015) describieron que los dos objetivos del modelo son minimizar el tiempo total de viaje a lo largo de la ruta y maximizar la seguridad de la misma, respectivamente (p. 2). Asimismo, Wen-mei et al. (2015) propusieron un método heurístico fácil de implementar para obtener rápidamente una solución eficiente y factible, basado en el modelo de camino más corto con un solo objetivo (p. 2).

Al respecto, Wen-mei et al. (2015) determinaron finalmente que desarrollaron una heurística fácil de implementar para obtener una solución eficiente y factible, demostrando la eficacia y viabilidad de los patrones y algoritmos presentados en el desempeño del simulacro (p. 2). Asimismo, Wen-mei et al. (2015) agregaron que la seguridad del camino se modela como el producto de la seguridad a través de arcos incluidos en el camino (p. 2).

Henchey et al. (2015) realizaron pruebas en rutas determinadas a partir de un problema del trayecto más corto basado en el algoritmo de Dijkstra y un problema del trayecto más corto restringido por varianza bajo condiciones variables de tráfico y clima en un simulacro "inteligente medio ambiente" (p. 569). Al respecto, Henchey et al. (2015) indicaron que el desarrollo de sistemas de transporte inteligentes hará que la adquisición de datos en tiempo real sea factible; por lo tanto, el problema de interés es la determinación de un método para utilizar los datos disponibles en tiempo real sobre la congestión de la red de tráfico (p. 570). Asimismo, Henchey et al. (2015) compararon el rendimiento de dos posibles métodos de enrutamiento que utilizan estas técnicas de fusión de datos: un modelo de ruta más corta de Dijkstra (1959) y un modelo VCSP de Sivakumar y Batta [1994] (p. 570).

Henchey et al. (2015) determinaron que el VCSP es un modelo más sofisticado que utiliza más datos; pero, no entrega soluciones de alta calidad y esto se debe al comportamiento inestable del modelo que puede dar soluciones que son demasiado propensas o reacias a los riesgos debido al valor del parámetro (p. 577). Además, Henchey et al. (2015) agregaron que el estudio muestra la necesidad de un posible método para predecir los tiempos de viaje futuros y utilizarlos para encaminar a los vehículos y si lo hace puede proporcionar una mejor ruta, ya que podría reflejar niveles crecientes de congestión como resultado de una caída en el tiempo (p. 577).

Kikelomo et al. (2017) utilizaron la teoría de grafos para desarrollar un mapa móvil del campus de la Universidad de tecnología de Yaba aplicando el algoritmo de ruta más corto para determinar la ruta más corta y rápida desde el origen hasta el destino (p. 3300). Asimismo, Kikelomo et al. (2017) describieron que el diseño se centró en el algoritmo de Dijkstra asumiendo el nodo fuente

como primer nodo permanente y se le asigna un valor de 0, revisa todos los nodos vecinos desde el nodo permanente anterior y calcula el costo acumulado de cada nodo vecino y los hace temporales, luego elige el nodo con el menor costo acumulativo y lo establece como un nodo permanente (p. 3300). De la misma manera, Kikelomo et al. (2017) identificaron los diferentes nodos que conducen a un destino particular, la distancia y el tiempo desde una fuente a un destino se calculan utilizando el mapa de Google (p. 3300).

Al respecto, Kikelomo et al. (2017) determinaron que el problema de la ruta más corta se solucionó con la teoría de grafos que encuentra la ruta más corta y rápida (p. 3304). Kikelomo et al. (2017) diseñaron e implementaron una aplicación de mapas para dispositivos móviles utilizando Android Studio el cual fue aprobado ya que cumple con el estándar y la solución requerida para resolver el problema identificado (p. 3304).

Komatsu et al. (2016) propusieron un esquema de guía de evacuación automática basada en la interacción implícita entre los evacuados y sus nodos móviles [teléfonos inteligentes] (p. 127). Asimismo, Komatsu et al. (2016) indicaron que el esquema propuesto estima automáticamente los tramos de carretera bloqueados a partir de la diferencia entre la ruta de evacuación presentada y la ruta de evacuación real y, a continuación, vuelve a calcular la ruta de evacuación alternativa (p. 127).

Komatsu et al. (2016) recomendaron que la ruta más corta se pueda calcular mediante los algoritmos de búsqueda de gráficos existentes, por ejemplo, el algoritmo de ruta más corto de Dijkstra (p. 131). Al respecto, Komatsu et al. (2016) determinaron que las aplicaciones calculan las rutas de evacuación con esta información local y navegan a través de las rutas a los evacuados cuando se producen desastres (p. 129).

Al respecto, Komatsu et al. (2016) demostraron a través de experimentos de simulación que el esquema propuesto funciona bien cuando el grado de daño es alto o los segmentos de la carretera están continuamente bloqueados y el segundo punto que demostraron es el tiempo promedio de evacuación, el cual puede mejorarse incluso en una pequeña proporción de penetración del

propuesto sistema y como último punto la comunicación inalámbrica directa puede admitir muchas evacuaciones casi al mismo nivel que la infraestructura de comunicación cuando el número de evacuados aumenta (Komatsu et al., 2016, p. 137).

Pérez (2018) presentó una metodología para la selección óptima de la flota de vehículos necesaria para distribuir productos médicos desde el centro de almacenamiento de la compañía a muchos nodos de transferencia intermedios, y desde estos a un grupo de hospitales públicos (p. 1). Asimismo, Pérez (2018) estimó la cantidad y el tipo de transporte necesario para mover los productos dentro de la red de la manera más eficiente posible (p. 2).

Pérez (2018) utilizó tres modelos para lograr la optimización global del problema: (a) en el primer modelo se usó una versión modificada del algoritmo de Dijkstra para el problema de agrupamiento, (b) el segundo modelo fue el conocido algoritmo de agente de viaje para calcular las rutas de transporte en cada sección de la red y (c) el tercer modelo es una instancia de programación lineal de enteros mixtos para la selección del transporte que se va a comprar (Pérez, 2018, p. 6). Al respecto, Pérez (2018) demostró que el modelo propuesto selecciona los transportes de carga óptimos para viajar en las rutas permitidas y para esto el transporte programado en cada ejecución computacional del modelo especificó claramente qué vehículo debe viajar, en qué áreas, a plena carga y con distancias mínimas (p. 16).

Como segunda parte de este apartado del contexto teórico concernió a las teorías relacionadas. Las fuentes seguras que dieron base a la investigación sobre abstracciones de algoritmos, aplicaciones móviles, base de datos y metodologías ágiles que ayudaron a ampliar los conocimientos y al desarrollo de este proyecto.

Erickson (2019) definió que el algoritmo es una secuencia de instrucciones elementales explícitas, precisas, inequívocas y mecánicamente ejecutables, generalmente destinadas a lograr un propósito específico (p. 1). Al respecto, Rodríguez et al. (2013) afirmaron que avances tecnológicos como la telemedicina y la salud móvil conceden optimizar el paso a los servicios, mejorando su eficiencia y calidad de atención y que esta propuesta de la



telemedicina permite llegar a las comunidades más aisladas y vulnerables (Rodríguez et al., 2019, p. 417). Asimismo, Santamaría y Hernández (2015) agregaron que las aplicaciones muestran grandes réditos, como optimizar la calidad de servicio y evitar errores médicos mediante algoritmos clínicos, reducir costos, evitan consultas médicas innecesarias y mejoran el acceso a los servicios de salud (p. 602).

Oppel y Sheldon (2006) indicaron que el término “base de datos” se refiere a una recopilación de nombres y referencias de ubicaciones hasta un dificultoso sistema de restablecimiento y acopio de datos basado en servidores, computadoras y módulos de usuario. Asimismo, Oppel y Sheldon (2006) explicaron que el lenguaje SQL permite realizar la creación, la actualización y el sostenimiento de un base de datos relacional (p. 4).

Campo (2019) consideró que el API de Google Maps es una aplicación de mapeo que brinda módulos avanzados en línea, permitiendo la manipulación de datos por parte del usuario, aplicándose en el proyecto para obtener las ubicaciones en tiempo real de las ambulancias más cercanas al punto del acontecimiento, reduciendo el tiempo de llegada al lugar y siendo de gran conveniencia para mejorar el servicio de traslado asistencial básico (Campo, 2019, p. 134)

Ramírez (2011) precisó que la metodología Mobile-D le permitió conseguir ciclos de desarrollos muy cortos y laborando de esa manera se deben conseguir productos totalmente funcionales en menos de 70 días (p. 42). Al respecto, se estableció una contraposición entre estas metodologías para decidir la metodología de desenvolvimiento que se utilizaría para el estudio, luego de la cual se estableció que la metodología Mobile-D era la más oportuna y fue aplicada para el desarrollo.

### **III. MÉTODO**

El estudio fue de tipo aplicado, de enfoque cuantitativo, diseño no experimental, transversal, descriptivo y la variable fue el impacto que tendría la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia. La muestra tuvo 100 solicitudes generadas en simultáneo para experimentar la suficiencia de sostener una gran concurrencia, de las cuales fueron procesadas 93 solicitudes, acercando el tiempo de llegada y traslado al tiempo estimado por la OMS. Se usó la recopilación de datos simulados realizando la creación de la base de datos, la cual contó con tres tablas maestras: perfil, estado y tipo de emergencia y las tablas relacionadas: usuario, solicitud, ambulancia, alergia y ubicación con sus diferentes campos cada uno. Asimismo, se detalló los mecanismos realizados en el estudio y los elementos éticos practicados.

### **3.1 Tipo y diseño de la investigación**

El tipo de estudio fue aplicado, de enfoque cuantitativo y diseño no experimental, transversal y descriptivo. Al respecto, Rodríguez et al. (2013) afirmaron que avances tecnológicos como la telemedicina y la salud móvil conceden optimizar el paso a los servicios, mejorando su eficiencia y calidad de atención, esta propuesta de la telemedicina permite llegar a las comunidades más aisladas y vulnerables, posibilitando ganar a los obstáculos geográficos que entorpecen el acceso al cuidado de la salud y a la vez permiten extender la cobertura y optimizar la constancia de la atención (Rodríguez et al., 2019, p. 417).

Cortés e Iglesias (2004) agregaron que la investigación cuantitativa ocupa como núcleo de su desarrollo a las calibraciones numéricas, usa la contemplación del proceso en forma de recopilación de datos y los estudia para lograr responder sus dudas sobre el estudio realizado (p. 10). Asimismo, Hernández et al. (2014) indicaron que el enfoque cuantitativo utiliza la recolección de datos para probar hipótesis con base en la medición numérica y el análisis estadístico, con el fin establecer pautas de comportamiento y probar teorías (p. 4). En esta investigación se desarrolló una aplicación de asignación de ambulancias para el traslado de pacientes de emergencia con la intención de utilizar nuevos enfoques para la resolución de problemas de optimización de

tiempos de respuesta y disponibilidad de ambulancias para la atención de emergencias.

Cortés et al. (2004) indicaron que la investigación no experimental no utiliza deliberadamente las variables a estudiar y que la función principal de este tipo de estudio es observar los fenómenos como se suscitan en su contexto real, para luego ser analizados (p. 27). También, Hernández et al. (2014) agregaron que dentro de la investigación cuantitativa se encuentran diferentes tipos de diseños: experimental y no experimental (p. 129). Hernández et al. (2014) agregaron que la investigación no experimental se subdivide en dos tipos de diseños transversales y longitudinales (p. 129).

Al respecto, Cortés et al. (2004) indicaron que el diseño transversal recolecta los datos en una sola interacción dada con el fin de interrelacionar las variables en un tiempo único y que este tipo de investigaciones se puede categorizar como descriptiva o de correlación, según el problema de la investigación (p. 27). Asimismo, Hernández et al. (2014) agregaron que gracias a los estudios descriptivos se puede detallar los atributos recopilados, aplicando mediciones en las diferentes situaciones o fenómenos importantes que se analicen (p. 92).

### **3.2 Operacionalización de variables**

El efecto de la aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia fue planteado con el propósito reducir el tiempo de respuesta y mejorar la disponibilidad para aumentar la cantidad de emergencias atendidas. Para complementar los conceptos con la variable, Hernández et al. (2014) indicaron que el ascenso de una variable teórica a señalizadores empíricos comprobables y cuantificables se le califica como operacionalización (p. 211). Asimismo, Hernández et al. (2014) definieron a la operacionalización como un conjunto de procedimientos y actividades que se desarrollan para medir una variable (p. 120).

En la tabla 1 se señala como la variable fue analizada y medida. Para ello, se desmontó la variable en dimensiones y esta a su vez en señalizadores a ser medidos en la ficha de registro en la asignación de ambulancias para el traslado de pacientes de emergencia.

Tabla 1. Matriz de Operacionalización de las variables

Matriz de operacionalización de las variables de la investigación

Variable	Definición Conceptual	Definición Operacional	Dimensión	Indicador	Instrumento	Escala de Medición
Impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia.	Consecuencias que se obtienen después de implementar una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia.	Aplicación móvil que facilitara la solicitud de ambulancias y recibir en un menor tiempo de respuesta la asistencia médica.	<b>Tiempo de respuesta</b> (Tejedor, 2000, p. 24; Ortega, 2018, p. 20; Bravo, 2012, p. 42)	<b>Disminución porcentual del tiempo de respuesta a solicitudes de ambulancias para el traslado de pacientes de emergencias.</b>  $TDR = \frac{\sum(Hllegada - hsolicitud)}{N^{\circ} servicios}$  $PTR = \frac{TRA - TDRD}{TRA}$  (Tejedor, 2000, p.24; Ortega, 2018, p. 20; Bravo, 2012, p. 42)	Ficha de Registro	Ratio
(Tejedor, 2000, p. 24; Ortega, 2018, p. 20; Bravo, 2012, p. 42; Bedoya y Laurente, 2014, p. 8; Rodríguez, Osorno y Maya, 2015, p. 169)	(Tejedor, 2000, p. 24; Ortega, 2018, p. 20; Bravo, 2012, p. 42; Bedoya y Laurente, 2014, p. 8; Rodríguez, Osorno y Maya, 2015, p. 169)	(Tejedor, 2000, p. 24; Ortega, 2018, p. 20; Bravo, 2012, p. 42; Bedoya y Laurente, 2014, p. 8; Rodríguez, Osorno y Maya, 2015, p. 169)	<b>Emergencias atendidas</b> (Rodríguez, Osorno y Maya, 2015, p. 169; Bedoya y Laurente, 2014, p. 8)	<b>Incremento porcentual de la cantidad de emergencias atendidas</b>  $PEA = \frac{NEAD - NEAA}{NEAA}$  (Rodríguez, Osorno y Maya, 2015, p. 169; Bedoya y Laurente, 2014, p. 8)	Ficha de registro	Ratio

### 3.3 Población, muestra y muestreo

El poblado se simuló con la iteración de 100 solicitudes de ambulancias para pacientes de emergencia, categorizados por los niveles de manera masiva. Al respecto, Bedoya y Laurente (2014) representaron un estímulo específico para poder estudiar el rendimiento del sistema, presentaron 20 solicitudes sincronizadas previo a exponer algún tipo de corrupción en el aplicativo, mostrando un lapso de 10 segundos, esto se basó en el número de solicitudes ya que la aseguradora no recibía más de 20 alertas en menos del lapso mencionado anteriormente (p.32). Asimismo, Castañeda y Bareño (2015) ensayaron la facultad de soportar una superior concurrencia simulando 100 peticiones simultáneas por medio de la herramienta SOAP UI, obteniendo como resultado de las pruebas: 100 solicitudes realizadas, 90 solicitudes atendidas y 10 solicitudes no atendidas (p. 61).

Arenas (2019) mostró el número de atenciones ejecutadas a través del SAMU en la figura 1, las cuales fueron categorizadas con prioridad II, sucesivamente prioridad III y con menos asiduidad la prioridad IV con 3487, 1140 y 362 casos respectivamente (p. 57). Además, Arenas (2019) agregó que el número de atenciones de Prioridad I aumenta a pesar de que el número de atenciones para el 2018 es menor (p. 57).

		Año 2017		Año 2018		TOTAL	
		Nominal	%anual	Nominal	% anual	Nominal	% total
PRIORIDAD	Prioridad I	376	12.09%	444	16.44%	820	14.12%
	Prioridad II	2009	64.62%	1478	54.74%	3487	60.03%
	Prioridad III	581	18.69%	559	20.70%	1140	19.62%
	Prioridad IV	143	4.60%	219	8.11%	362	6.23%
	Total	3109	100%	2700	100%	5809	100%

Fuente :Oficina de Prevención y control de emergencia y desastres (DIRESA)

Figura 1. Atenciones realizadas por el SAMU entre los años 2017 y 2018 (Arenas, 2019)

### **3.4 Técnicas e instrumentos de recolección de datos, validez y confiabilidad**

En esta sección se representó los instrumentos de acumulación de datos y las técnicas que fueron aplicadas a este estudio, como la recopilación de datos simulados, la ficha de registro, métodos y análisis de datos que estuvieron basados en la simulación masiva de solicitudes de ambulancias, generando varias iteraciones e información posteriormente del empleo de la aplicación móvil. Para mayor información acerca de los datos generados revisar el anexo 9.

Al respecto, Candelaria et al. (2011) indicaron que un sistema de simulación está compuesto de expresiones algebraicas teóricas que permiten procesar variables de un contexto real simulando el resultado de distintas alternativas (p. 1000). Asimismo, Batanero (2009) agregó que el beneficio es obvio cuando se trabaja con simulaciones ya que admite compactar el ensayo en un espacio y tiempo determinado, permitiendo simplificar la propia realidad e inferir un trabajo de contemplación sobre ella (p. 45).

#### **Instrumento de recolección de datos**

Niño (2011) indicó que las fichas de recolección son herramientas practicas tanto para documentar como para otras técnicas de recolección de datos (p. 98). Además, Niño (2011) agregó que actualmente el indagador cuenta con innumerables herramientas tecnológicas que permiten procesar y almacenar información de manera práctica (p. 98).

#### **Validez**

Niño (2011) indicó que la validez permite cuantificar la variable de manera precisa, aceptando describir o medir la variable a través de una pregunta o ítem (p. 87). Asimismo, Ñaupás et al. (2014) indicaron que les la adecuación de un mecanismo de medición que permite mostrar, detallar o predecir el atributo que es importante para el evaluador (p. 215).

Al respecto, Hernández et al. (2014) comentaron que la validez de contenido se refiere al nivel en que un instrumento refleja un sometimiento específico de contenido de lo que se mide (p. 201). Los artículos citados en esta

investigación le dan la validez de contenido a las variables, dimensiones e indicadores de la ficha de recolección de datos.

### **3.5 Procedimientos**

Para el procedimiento de recolección de datos se utilizaron los siguientes pasos:

- Paso 1: Generar la base de datos en el motor de base de datos SQL Server 2008 R2. Se tuvo tres tablas maestras: perfil, estado, tipo de emergencia y las tablas relacionales: usuario, solicitud, ambulancia, alergia y ubicación con sus diferentes campos cada uno.
- Paso 2: Para poblar la base de datos se utilizó Scripts, sentencias SQL y a través de inserts se realizó la carga de datos simulada.

Se tuvo dos iteraciones de 100 y 75 solicitudes de emergencia con una disponibilidad de 100 ambulancias validando el número de solicitudes procesadas para la primera y segunda simulación.

### **3.6 Método de análisis de datos**

En esta simulación se utilizó dos iteraciones: una de 100 y otra de 75 solicitudes con la disponibilidad de 100 ambulancias. El nivel de confianza fue 95%, el error máximo aceptable fue 5% y los resultados fueron significativos al nivel de 0.05 ( $p < 0.05$ ). Se usó la prueba de normalidad de Kolmogorov-Smirnov y la prueba de promedios no paramétrica de Wilcoxon.

Al respecto, Saldaña (2016) indicó que la prueba de Kolmogorov-Smirnov es conocida como prueba K-S y es una prueba de significación estadística para verificar si los datos de la muestra proceden de una distribución normal (p. 36). La prueba K-S es utilizada para estimar la normalidad de variables cuantitativas continuas cuando el tamaño muestral es mayor que 50 (Saldaña, 2016, p. 36).

### **3.7 Aspectos éticos**

A lo largo del progreso de este estudio se reconoció la autoría de los orígenes mencionando y referenciando a través de la ISO 690:2010, a la vez se consideró los valores infundidos como competente de ingeniería y dentro de estos valores se encuentran la puntualidad, responsabilidad, honestidad, tolerancia y lealtad. El desarrollo de este estudio no menoscabo a la organización educativa, ni a la comunidad.



Ñaupas et al. (2014) precisaron si el estudio científico es un emprendimiento social, es coherente pensar que, si la comunidad tiene defectos morales, infectará los investigadores y científicos (p. 462). Asimismo, Ñaupas et al. (2014) agregaron que no es necesario solo responsabilizarse por el puntonor de los sujetos que intervienen en los avances de la investigación, ni de los organismos dedicados a la investigación, se trata de ocuparse de las diplomacias de investigación estatales y sobre todo de los mismos investigadores que debieran acoplarse a un código de ética. (p. 462). Al respecto, Cantú (2015) agregó que la inserción de la bioética en el plano de la educación superior examina primero las formas para abandonar la formación profesional sustentada solo en el utilitarismo, como razón instrumental del quehacer profesional y en segundo término favorece los aspectos de carácter humanístico de la preparación profesional en asistencia a la población, porque tiene que ver con el beneficio común (p. 47).

Por otro lado, este estudio se alineó a los principios fundamentales, establecidos por los especialistas del Colegio de Ingenieros del Perú (2018) promoviendo la lealtad profesional, la honestidad, el honor profesional, la responsabilidad, la solidaridad, el respeto, la justicia y la inclusión social expuestos por el Código de Ética del Colegio de Ingenieros del Perú en el artículo 15 (p. 3). Del mismo modo, esta investigación se acogió a los principios de ética en investigación propuestos en el artículo 3 del Código de Ética de Investigación de la Universidad César Vallejo, siendo el respeto por las personas en su integridad y autonomía, la justicia, la honestidad y el rigor científico y la no maleficencia (Universidad César Vallejo, 2020, p. 8).

Los artículos del Código de Ética de la Investigación de la UCV que se está cumpliendo en esta investigación son: (a) artículo 1, ya que se cumpliendo con los máximos estándares de rigor científico, responsabilidad y honestidad; (b) artículo 2 sobre la obligatoriedad del cumplimiento del citado código de ética para el desarrollo de la investigación, debido a que es obligatorio para quienes realizan investigación en la Universidad César Vallejo; (c) artículo 3, puesto que se cumplió con la autonomía, beneficencia, justicia y libertad; (d) artículo 9, ya que esta investigación pasó por un sistema de anti-plagio (Turnitin), para así tener la certeza que es un trabajo con un desarrollo diferente a los demás (Universidad César Vallejo, 2020, p. 5).

## **IV. RESULTADOS**

En este capítulo se detalla el efecto que tendría la aplicación del estudio, con los indicadores “tiempo de respuesta” y “número de emergencias atendidas”, junto al procesamiento de los datos obtenidos de cada indicador con el programa SPSS Statistics 25 y el impacto de la implementación de una aplicación móvil para la asignación de ambulancias para el traslado de pacientes de emergencia.

#### **4.1 Prueba de la hipótesis específica 1**

La hipótesis específica 1 fue la siguiente:

- **HE1<sub>0</sub>**: La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia no reduciría el tiempo de respuesta.
- **HE1<sub>1</sub>**: La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia reduciría el tiempo de respuesta.

##### **4.1.1 Con 75 solicitudes**

En la tabla 2 se muestra los registros obtenidos con una iteración de 75 solicitudes de emergencia con una disponibilidad de 100 ambulancias. En la primera columna se muestra el código que identifica cada una de las solicitudes procesadas que se enlazaron con las ambulancias asignadas. En la segunda columna se observa el código de la ambulancia que se le asignó a la solicitud anteriormente procesada. En la tercera columna se muestra la fecha y hora en la que se registró la solicitud. En la cuarta columna se observa la fecha y hora en la que llegó la ambulancia al punto de solicitud. En la quinta columna se muestra el tiempo estimado que demoró la ambulancia en llegar al punto de solicitud. Finalmente, en la sexta columna se muestra el nivel de emergencia que se solicitó y el nivel de la ambulancia. Los registros con ítems 74 y 75 no se reflejan en los resultados de la simulación debido a que fueron las solicitudes que no se procesaron o fueron asistidas por una ambulancia.

Tabla 2. Resultados de la simulación de 75 solicitudes de emergencia.

N°	codigo_solicitud	codigo_ambulancia	fecha_solicitud	fecha_llegada	tiempo	tipo_emergencia
1	576	11	15:07:29	15:17:30	00:10:01	3
2	577	89	15:07:30	15:19:04	00:11:34	4
3	578	90	15:07:30	15:27:26	00:19:56	4
4	579	83	15:07:30	15:30:26	00:22:56	4
5	580	82	15:07:31	15:23:28	00:15:57	4
6	581	88	15:07:31	15:19:03	00:11:32	4
7	582	85	15:07:32	15:32:04	00:24:32	4
8	583	64	15:07:32	15:19:36	00:12:04	3
9	584	84	15:07:33	15:20:37	00:13:04	4
10	585	59	15:07:33	15:32:35	00:25:02	3
11	586	78	15:07:33	15:27:36	00:20:03	2
12	587	81	15:07:34	15:26:37	00:19:03	2
13	588	80	15:07:34	15:20:36	00:13:02	2
14	589	97	15:07:35	15:21:37	00:14:02	2
15	590	79	15:07:35	15:26:33	00:18:58	2
16	591	58	15:07:35	15:32:34	00:24:59	3
17	592	86	15:07:36	15:27:37	00:20:01	4
18	593	10	15:07:36	15:24:44	00:17:08	3
19	594	13	15:07:37	15:22:46	00:15:09	4
20	595	95	15:07:37	15:24:44	00:17:07	2
21	596	69	15:07:41	15:18:17	00:10:36	3
22	597	21	15:07:41	15:31:18	00:23:37	4
23	598	87	15:07:41	15:27:18	00:19:37	4
24	599	20	15:07:42	15:18:18	00:10:36	4
25	600	77	15:07:42	15:21:16	00:13:34	2
26	601	60	15:07:43	15:18:17	00:10:34	3
27	602	61	15:07:43	15:23:18	00:15:35	3
28	603	68	15:07:44	15:23:19	00:15:33	3
29	604	96	15:07:44	15:26:25	00:18:41	2
30	605	12	15:07:45	15:22:25	00:14:40	4
31	606	9	15:07:45	15:23:25	00:15:40	3
32	607	91	15:07:45	15:22:26	00:14:41	4
33	608	19	15:07:46	15:25:27	00:17:41	4
34	609	18	15:07:46	15:26:25	00:18:39	4
35	610	76	15:07:47	15:22:25	00:14:38	2
36	611	17	15:07:47	15:22:25	00:14:38	4
37	612	94	15:07:48	15:19:02	00:11:14	2
38	613	98	15:07:48	15:20:27	00:12:39	2
39	614	14	15:07:48	15:18:58	00:11:10	4
40	615	93	15:07:49	15:21:58	00:14:09	2
41	616	16	15:07:49	15:27:00	00:19:11	4

N°	codigo_solicitud	codigo_ambulancia	fecha_solicitud	fecha_llegada	tiempo	tipo_emergencia
42	617	101	15:07:50	15:21:01	00:13:11	2
43	618	5	15:07:50	15:31:00	00:23:10	3
44	619	67	15:07:50	15:22:58	00:15:28	3
45	620	15	15:07:51	15:23:58	00:16:07	4
46	621	66	15:07:51	15:23:00	00:15:09	3
47	622	65	15:07:52	15:30:00	00:22:08	3
48	623	63	15:07:52	15:23:00	00:15:08	3
49	624	6	15:07:52	15:32:06	00:24:14	4
50	625	100	15:07:53	15:29:06	00:21:13	2
51	626	22	15:07:53	15:31:08	00:23:15	4
52	627	23	15:07:54	15:19:09	00:11:15	4
53	628	62	15:07:54	15:29:08	00:21:14	3
54	629	8	15:07:55	15:27:07	00:19:12	3
55	630	36	15:07:55	15:31:06	00:23:11	3
56	631	28	15:07:55	15:26:42	00:18:47	4
57	632	24	15:07:56	15:21:41	00:13:45	4
58	633	26	15:07:56	15:22:42	00:14:46	4
59	634	7	15:07:56	15:23:49	00:15:53	3
60	635	25	15:07:57	15:30:50	00:22:53	4
61	636	99	15:07:57	15:32:51	00:24:54	2
62	637	57	15:07:58	15:31:50	00:23:52	2
63	638	56	15:07:58	15:21:50	00:13:52	2
64	639	35	15:07:58	15:26:49	00:18:51	3
65	640	43	15:07:59	15:22:50	00:14:51	3
66	641	4	15:07:59	15:31:50	00:23:51	2
67	642	27	15:08:00	15:28:50	00:20:50	4
68	643	42	15:08:00	15:25:50	00:17:50	3
69	644	55	15:08:01	15:31:24	00:23:23	2
70	645	70	15:08:01	15:25:59	00:17:58	3
71	646	54	15:08:01	15:25:59	00:17:58	2
72	647	34	15:08:02	15:26:58	00:18:56	3
73	648	41	15:08:02	15:21:59	00:13:57	3

De los datos anteriormente mencionados se ejecuta la prueba de normalidad con los datos de la columna tiempo obteniendo los siguientes resultados:

Tabla 3. Datos estadísticos de la prueba de normalidad de Kolmogorov–Smirnov.

Pruebas de normalidad						
	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
Tiempo	.105	73	.047	.953	73	.009

Se verifica que la prueba de normalidad al ser mayor de 50 unidades se estaría utilizando los resultados de Kolmogorov-Smirnov y se identifica que el p valor es menor a 0.05 entonces no es normal la repartición de la muestra, para pruebas no normales se utiliza la comprobación no paramétrica de Wilcoxon.

### Prueba de Wilcoxon

Prueba no paramétrica para comparar la media de dos muestras relacionadas y determinar si existen diferencias entre ellos. Al realiza la prueba de Wilcoxon comparando con las muestras de Essalud se obtienen los siguientes resultados:

Tabla 4. Datos estadísticos de la prueba de comprobación de Wilcoxon.

Rangos				
		N	Rango promedio	Suma de rangos
BRAVO_ESSALUD - tiempo	Rangos negativos	0 <sup>a</sup>	.00	.00
	Rangos positivos	73 <sup>b</sup>	37.00	2701.00
	Empates	0 <sup>c</sup>		
	Total	73		

a. BRAVO\_ESSALUD < tiempo

b. BRAVO\_ESSALUD > tiempo

c. BRAVO\_ESSALUD = tiempo

Tabla 5. Datos estadísticos de la prueba Z.

Estadísticos de prueba <sup>a</sup>	
	BRAVO_ESSAL UD - tiempo
Z	-7.424 <sup>b</sup>
Sig. asintótica(bilateral)	.000

a. Prueba de rangos con signo de Wilcoxon

b. Se basa en rangos negativos.

Gracias a los datos alcanzados que se reflejan en la tabla 5 se computó la utilidad de Z, que fue -7.424, siendo inferior a 1.96 y el nivel de significancia, el cual fue 0.000, el cual es menor que 0.05, se desestima la hipótesis nula aprobando la hipótesis alterna con un 95% de nivel de confianza, la cual indica que “El uso de una aplicación móvil de asignación de ambulancias disminuirá el tiempo de respuesta a solicitudes de ambulancia para el traslado de pacientes de emergencia”.

#### **4.1.2 Con 100 solicitudes**

En la tabla 6 se muestra los registros obtenidos con una iteración de 100 solicitudes de emergencia con una disponibilidad de 100 ambulancias, en la primera columna se muestra el código que identifica cada una de las solicitudes procesadas que hicieron match con las ambulancias asignadas, en la segunda columna se muestra el código de la ambulancia que se le asignó a la solicitud anteriormente procesada, en la tercera columna la fecha y hora en la que se registró la solicitud, en la cuarta columna la fecha y hora en la que llegó la ambulancia al punto de solicitud, en la quinta columna el tiempo estimado que demora la ambulancia en llegar al punto de solicitud y en la sexta columna el nivel de emergencia que se solicitó y el nivel de la ambulancia. Los registros con ítem del 94 al 100 no se reflejan en los resultados de la simulación debido a que fueron las solicitudes que no se procesaron o fueron asistidas por una ambulancia.

Tabla 6. Resultados de la simulación de 100 solicitudes de emergencia.

N°	codigo_solicitud	codigo_ambulancia	fecha_solicitud	fecha_llegada	tiempo	tipo_emergencia
1	649	78	15:18:11	15:39:03	00:20:52	2
2	650	89	15:18:12	15:36:22	00:18:10	4
3	651	81	15:18:12	15:28:22	00:10:10	2
4	652	90	15:18:12	15:31:21	00:13:09	4
5	653	80	15:18:13	15:33:21	00:15:08	2
6	654	97	15:18:13	15:29:22	00:11:09	2
7	655	83	15:18:14	15:36:22	00:18:08	4
8	656	11	15:18:14	15:35:21	00:17:07	3
9	657	64	15:18:14	15:42:21	00:24:07	3
10	658	82	15:18:15	15:42:22	00:24:08	4
11	659	59	15:18:15	15:43:30	00:25:15	3
12	660	88	15:18:16	15:36:31	00:18:15	4
13	661	85	15:18:16	15:34:30	00:16:14	4
14	662	79	15:18:16	15:29:29	00:11:13	2
15	663	84	15:18:17	15:38:31	00:20:14	4
16	664	86	15:18:17	15:33:30	00:15:13	4
17	665	95	15:18:18	15:35:30	00:17:12	2
18	666	77	15:18:18	15:31:30	00:13:12	2
19	667	96	15:18:18	15:37:30	00:19:12	2
20	668	13	15:18:19	15:40:30	00:22:11	4
21	669	76	15:18:19	15:37:21	00:19:02	2
22	670	94	15:18:20	15:38:21	00:20:01	2
23	671	21	15:18:20	15:42:18	00:23:58	4
24	672	58	15:18:20	15:35:19	00:16:59	3
25	673	10	15:18:21	15:43:19	00:24:58	3
26	674	69	15:18:21	15:34:18	00:15:57	3
27	675	98	15:18:22	15:41:06	00:22:44	2
28	676	60	15:18:22	15:43:06	00:24:44	3
29	677	61	15:18:22	15:42:08	00:23:46	3
30	678	87	15:18:23	15:41:08	00:22:45	4
31	679	68	15:18:23	15:44:10	00:25:47	3
32	680	93	15:18:24	15:44:10	00:25:46	2
33	681	9	15:18:24	15:37:11	00:18:47	3
34	682	101	15:18:24	15:33:12	00:14:48	2
35	683	5	15:18:25	15:33:12	00:14:47	3
36	684	100	15:18:25	15:42:14	00:23:49	2
37	685	20	15:18:26	15:29:14	00:10:48	4
38	686	67	15:18:26	15:42:15	00:23:49	3
39	687	99	15:18:26	15:33:16	00:14:50	2
40	688	57	15:18:27	15:41:17	00:22:50	2
41	689	12	15:18:27	15:38:23	00:19:56	4



N°	codigo_solicitud	codigo_ambulancia	fecha_solicitud	fecha_llegada	tiempo	tipo_emergencia
42	690	91	15:18:28	15:43:24	00:24:56	4
43	691	56	15:18:28	15:35:24	00:16:56	2
44	692	4	15:18:28	15:38:26	00:19:58	2
45	693	66	15:18:29	15:43:26	00:24:57	3
46	694	55	15:18:29	15:35:27	00:16:58	2
47	695	65	15:18:30	15:37:29	00:18:59	3
48	696	54	15:18:30	15:36:28	00:17:58	2
49	697	53	15:18:30	15:36:32	00:18:02	2
50	698	63	15:18:31	15:29:30	00:10:59	3
51	699	62	15:18:31	15:36:32	00:18:01	3
52	700	19	15:18:32	15:43:35	00:25:03	4
53	701	8	15:18:32	15:33:34	00:15:02	3
54	702	18	15:18:32	15:38:36	00:20:04	4
55	703	52	15:18:33	15:35:36	00:17:03	2
56	704	51	15:18:33	15:36:36	00:18:03	2
57	705	17	15:18:34	15:36:39	00:18:05	4
58	706	36	15:18:34	15:28:38	00:10:04	3
59	707	14	15:18:34	15:42:40	00:24:06	4
60	708	16	15:18:35	15:39:40	00:21:05	4
61	709	50	15:18:35	15:34:41	00:16:06	2
62	710	75	15:18:36	15:35:24	00:16:48	2
63	711	7	15:18:36	15:33:23	00:14:47	3
64	712	15	15:18:36	15:37:41	00:19:05	4
65	713	6	15:18:37	15:29:43	00:11:06	4
66	714	22	15:18:37	15:41:42	00:23:05	4
67	715	35	15:18:38	15:36:22	00:17:44	3
68	716	48	15:18:38	15:32:27	00:13:49	2
69	717	23	15:18:38	15:41:29	00:22:51	4
70	718	43	15:18:39	15:33:30	00:14:51	3
71	719	42	15:18:39	15:42:21	00:23:42	3
72	720	28	15:18:40	15:36:31	00:17:51	4
73	721	49	15:18:40	15:31:30	00:12:50	2
74	722	71	15:18:40	15:43:30	00:24:50	2
75	723	24	15:18:41	15:33:30	00:14:49	4
76	724	74	15:18:41	15:30:29	00:11:48	2
77	725	72	15:18:42	15:29:30	00:10:48	2
78	726	70	15:18:42	15:36:30	00:17:47	3
79	727	26	15:18:42	15:29:30	00:10:48	4
80	728	73	15:18:43	15:43:30	00:24:47	2
81	729	34	15:18:43	15:33:29	00:14:46	3
82	730	47	15:18:44	15:29:59	00:11:15	2
83	731	25	15:18:44	15:37:58	00:19:14	4
84	732	41	15:18:45	15:36:58	00:18:13	3

N°	codigo_solicitud	codigo_ambulancia	fecha_solicitud	fecha_llegada	tiempo	tipo_emergencia
85	733	46	15:18:45	15:42:58	00:24:13	2
86	734	45	15:18:45	15:40:58	00:22:13	2
87	735	3	15:18:46	15:31:58	00:13:12	2
88	736	27	15:18:46	15:37:58	00:19:12	4
89	737	40	15:18:47	15:39:58	00:21:11	3
90	738	29	15:18:47	15:33:58	00:15:11	3
91	739	2	15:18:47	15:28:57	00:10:10	2
92	740	92	15:18:48	15:40:07	00:21:19	2
93	741	37	15:18:48	15:44:07	00:25:19	3

De los datos anteriormente mencionados se ejecuta la comprobación de la normalidad con los datos de la columna tiempo obteniendo los siguientes resultados:

Tabla 7. Datos estadísticos de la prueba de normalidad Kolmogórov-Smirnov

Pruebas de normalidad						
	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
tiempo	,095	93	.037	.952	93	.002

Se verifica que la prueba de normalidad al ser mayor de 50 unidades se estaría utilizando los resultados de Kolmogórov-Smirnov y se identifica que el p valor es menor a 0.05 entonces no es normal la repartición de la muestra, para pruebas no normales se utiliza la comprobación no paramétrica de Wilcoxon.

### Prueba de Wilcoxon

Comprobación no paramétrica para paralelar la media de dos muestras relacionadas y decidir si existen diferencias entre ellos. Al realiza la comprobación de Wilcoxon comparando con las muestras de Essalud se obtienen los siguientes resultados:

Tabla 8. Datos estadísticos de la prueba comprobación Wilcoxon

Rangos				
		N	Rango promedio	Suma de rangos
Bravo_Essalud - tiempo	Rangos negativos	0 <sup>a</sup>	.00	.00
	Rangos positivos	93 <sup>b</sup>	47.00	4371.00
	Empates	0 <sup>c</sup>		
	Total	93		

a. Bravo\_Essalud < tiempo

b. Bravo\_Essalud > tiempo

c. Bravo\_Essalud = tiempo

Como es <1.96 se aprueba la hipótesis 1

Tabla 9. Datos estadísticos de la prueba Z

Estadísticos de prueba <sup>a</sup>	
	Bravo_Essalud - tiempo
Z	-8.374 <sup>b</sup>
Sig. asintótica(bilateral)	.000

De los números conseguidos que se muestran en la tabla 9 se computó el valor de Z, que fue -8.374<sup>b</sup>, que es inferior a 1.96 y el nivel de significancia, el cual fue 0.000, al ser menor que 0.05, se desestima la hipótesis nula aprobando la hipótesis alterna con un 95% de nivel de confianza. Señalando que “El uso de una aplicación móvil de asignación de ambulancias disminuirá el tiempo de respuesta a solicitudes de ambulancia para el traslado de pacientes de emergencia”.

## 4.2 Prueba de la hipótesis específica 2

La hipótesis específica 2 fue la siguiente:

- **HE2<sub>0</sub>**: La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia no aumentaría el número de emergencias atendidas.
- **HE2<sub>1</sub>**: La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia aumentaría el número de emergencias atendidas.

### 4.2.1 Con 75 solicitudes

Se obtuvo un crecimiento positivo en los que respecta a la disponibilidad y número de emergencias atendidas donde se demuestra a una demanda masiva de solicitudes generadas en diferentes niveles de emergencia responde en un 97.33% al número total de solicitudes optimizando la disponibilidad de las unidades ambulatorias.

### 4.2.2 Con 100 solicitudes

Se obtuvo un crecimiento positivo en los que respecta a la disponibilidad y número de emergencias atendidas donde se demuestra a una demanda masiva de solicitudes generadas en diferentes niveles de emergencia responde en un 93.00% al número total de solicitudes optimizando la disponibilidad de las unidades ambulatorias.

## 4.3 Prueba de la hipótesis general

- **HG<sub>0</sub>**: La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia no aumentaría el número de emergencias atendidas ni reduciría el tiempo de respuesta.
- **HG<sub>1</sub>**: La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia aumentaría el número de emergencias atendidas y reduciría el tiempo de respuesta.

Se verifica que la prueba de normalidad al ser mayor de 50 unidades se estaría utilizando los resultados de Kolmogorov-Smirnov y se identifica que el p valor es menor a 0.05 entonces no es normal la distribución de la muestra, para pruebas no normales se utiliza la prueba no paramétrica de Wilcoxon.

Tabla 10. Datos estadísticos de la prueba de normalidad de Kolmogorov – Smirnov.

<b>Pruebas de normalidad</b>						
	Kolmogorov-Smirnov <sup>a</sup>			Shapiro-Wilk		
	Estadístico	gl	Sig.	Estadístico	gl	Sig.
HUAROTO	.538	100	.000	.278	100	.000

### Prueba de Wilcoxon

Tabla 11. Datos estadísticos de la prueba de comprobación de Wilcoxon

<b>Rangos</b>				
		N	Rango promedio	Suma de rangos
CASTAÑEDA - HUAROTO	Rangos negativos	3 <sup>a</sup>	2.00	6.00
	Rangos positivos	0 <sup>b</sup>	.00	.00
	Empates	97 <sup>c</sup>		
	Total	100		

Tabla 12. Datos estadísticos de la prueba Z.

<b>Estadísticos de prueba<sup>a</sup></b>	
	CASTAÑEDA - HUAROTO
Z	-1.732 <sup>b</sup>
-Sig. asintótica(bilateral)	.083

Gracias a los datos conseguidos que se exponen en la tabla 12 se computó el valor de Z, siendo -1.732, que es inferior a 1.96 y el nivel de significancia, siendo 0.000, que es menor que 0.05, se desestima la hipótesis nula aprobando la hipótesis alterna con un 95% de nivel de confianza, la cual indica que “el uso de una aplicación móvil de asignación de ambulancias aumentara la cantidad de emergencias atendidas”, con un incremento del promedio de solicitudes aceptadas del 3.00% equivalente a 3 solicitudes más a comparación de la aplicación de Castañeda y Bareño.

#### 4.4 Resumen

En la tabla 13 se presenta la síntesis de los resultados de las comprobaciones de las hipótesis.

Tabla 13. Síntesis de los resultados de las comprobaciones de las hipótesis:

<b>Cód.</b>	<b>Hipótesis</b>	<b>Resultado</b>
<b>HE1</b>	La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia reduciría el tiempo de respuesta	<b>Aceptada</b>
<b>HE2</b>	La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia aumentaría el número de emergencias atendidas	<b>Aceptada</b>
<b>HG</b>	La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia reduciría el tiempo de respuesta y aumentaría el número de emergencias atendidas	<b>Aceptada</b>

Conforme se muestra en la tabla 13, con las derivaciones logradas se declaró que las hipótesis alternas expuestas inicialmente fueron aprobadas, adjudicando cumplir el propósito general y los propósitos específicos. Se produjo

un considerable tiempo de respuesta promedio de 00:18:25 minutos equivalentes al 54.65% y reduciendo en un 45.35% el tiempo de respuesta para solicitudes de ambulancias. También se obtuvo un crecimiento positivo en los que respecta a la disponibilidad y número de emergencias atendidas de 3.00% donde demuestra a una demanda masiva de solicitudes generadas en diferentes niveles de emergencia responde en un 93.00% al número total de solicitudes optimizando la disponibilidad de las unidades ambulatorias.

## **V. DISCUSSION**



En este capítulo se discute acerca de los resultados obtenidos en la presente investigación en comparación con los antecedentes de otras investigaciones de aplicaciones móviles de solicitud de ambulancias para paciente de emergencia, exclusivamente para demostrar el mejor rendimiento en los aspectos de tiempo de respuesta y el número de solicitudes procesadas. Además, en los indicadores asociados a las comprobaciones de las hipótesis donde se comparó semejanzas y diferencias al momento de implementar la aplicación.

La aplicación móvil tendría un efecto positivo al ser utilizada para la generación de solicitudes de emergencias nivel I, II y III, logrando disminuir el tiempo de respuesta y e incrementar el número de solicitudes procesadas. De acuerdo a los resultados estimados, el tiempo de respuesta antes de implementar la aplicación móvil para los niveles de emergencia I, II y III según Essalud eran de 35:03, 32:22 y 33:31 minutos y segundos respectivamente, obteniendo como promedio de tiempo de respuesta 00:33:39 minutos, tiempos que no ingresan dentro de los parámetros de la Organización Mundial de la Salud (Bravo, 2012, p. 26). El tiempo de respuesta después del uso de la aplicación móvil para los niveles I, II y III obtuvieron como promedio de tiempo de respuesta 00:18:25 minutos equivalentes al 54.65% y reduciendo en un 45.35% el tiempo de respuesta para solicitudes de ambulancias. Los resultados demostraron que la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia reduciría el tiempo de respuesta y aumentaría el número de emergencias atendidas.

Los resultados conseguidos fueron contrapuestos con algunos trabajos llevados a cabo anteriormente; sin embargo, se pudo contemplar que son equiparables a los de Castañeda y Bareño (2015) donde presentaron un prototipo de un sistema circunstante para la solicitud de servicio de ambulancia utilizando la herramienta SOAP UI para lanzar 100 peticiones simultaneas logrando solo 90 solicitudes aceptadas y 10 rechazadas (p. 60). Además, Castañeda y Bareño (2015) tomaron en cuenta la latitud y longitud del usuario como se tomó en cuenta para este proyecto, por motivos económicos en su caso para la adquisición de diferentes dispositivos móviles únicamente se realizaron pruebas al servicio Web que se encarga de recibir la solicitud y asignar la ambulancia disponible más cercana distancia y tiempo (p. 60). Asimismo, los

resultados de la presente investigación se basaron en un algoritmo diferencial espacio-tiempo tomando en cuenta el nivel de emergencia, la latitud y longitud del usuario, donde también se realizaron pruebas de alta concurrencia lanzando 100 solicitudes de ambulancia en diferentes niveles de emergencia obteniendo una mayor aceptación de 93 solicitudes procesadas y 7 rechazadas.

También se encontraron semejanzas y diferencias con respecto a la tesis Bedoya y Laurente (2014) donde consiguieron implementar una aplicación móvil para la asignación automática de unidades de ambulancias a los puntos de emergencia con prioridades de tiempo, distancia y turno de llegada de la alerta, en la cual solo se manejó en su concurrencia 20 registros de solicitudes e implemento la API de Google Maps (p. 62). Asimismo, los resultados de la presente investigación demuestran que se trabaja con una mayor concurrencia de número de solicitudes y una mejora en el tiempo de proceso de la solicitud de 0.48 minutos. En la aplicación de este proyecto también se implementó la API de Google Maps con la diferencia que no se trabajó con real time sino con data simulada.

Del mismo modo, se encontraron semejanzas con el proyecto de Isong et al. (2016) donde implementaron un prototipo de solicitud de ambulancia de emergencia en zonas rurales usando el API de Google Maps para la optimización de la ruta con el fin de garantizar que todos los pacientes sean atendidos en el menor tiempo posible utilizando adicionalmente el algoritmo First Come First Serve y Dijkstra (p. 19). Asimismo, la presente investigación utiliza el API de Google Maps para optimización de rutas y visualización del trayecto de la ambulancia, adicionalmente en la presente aplicación se implementa un algoritmo diferencial espacio tiempo que permite mejorar y optimizar los tiempos al momento de seleccionar la ambulancia mas cercana de acuerdo al nivel de emergencia que el paciente solicita.

## **VI. CONCLUSIONES**

Las conclusiones de la investigación fueron las siguientes:

1. De acuerdo con los datos obtenidos, el uso de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencias poseyó un impacto propicio en el transcurso de generación de solicitudes de emergencia a demanda, realizando iteraciones de 75 y 100 solicitudes obteniendo un tiempo de respuesta 00:18:25 minutos y reduciendo el tiempo de respuesta a comparación de Essalud en un 45.35%, acercando el tiempo de respuesta al tiempo estándar de la OMS para estos niveles de emergencia entre 10 a 15 minutos.
2. El porcentaje de emergencias atendidas o solicitudes procesadas se incrementó en un 3.00%, con esto se demuestra que la aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia ante una demanda masiva de solicitudes generadas en diferentes niveles de emergencia responde en un 93.00% al número total de solicitudes optimizando la disponibilidad de las unidades ambulatorias.
3. El porcentaje de solicitudes que no fueron asignadas con las ambulancias disponibles o que no fueron procesadas fue disminuido del 10.00% al 7.00% demostrando que la aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia disminuye el número de solicitudes que no fueron procesadas, aumentando su disponibilidad gracias la lógica de nivel de emergencia, tiempo y distancia.

## **VII. RECOMENDACIONES**

Las sugerencias para futuros estudios son las siguientes:

1. Realizar estudios comparativos sobre aplicaciones con algoritmos de optimización de rutas y diferenciales de espacio-tiempo para integrar los procesos de solicitud de ambulancias en diferentes organismos públicos y privados del departamento de salud, con la finalidad de optimizar los tiempos de respuesta y disponibilidad de unidades ambulatorias.
2. Acrecentar la investigación por medio de la amplificación de la población y la muestra para acopiar más información agregando al rango de latitud y longitud de los distritos de El Agustino y Cercado de Lima Metropolitana para la generación de solicitudes masivas.
3. Desarrollar el módulo administrador que pueda funcionar en múltiples plataformas como la Web, Android Studio o iOS, para monitorear de manera centralizada las ubicaciones de las unidades ambulatorias y las solicitudes procesadas.
4. Implementar un algoritmo de enrutamiento y compararlo con otros algoritmos de optimización de ruta que reciba en cuenta la afluencia de autos en tiempo real como el algoritmo de Dijkstra, Bellman-Ford y Floyd-Warshall.
5. Implementar la aplicación móvil en una entidad como el Minsa o Essalud para poder realizar pruebas con usuarios y ambulancias en tiempo real y agregar nuevos módulos como biografía clínica electrónica resumida, en donde se localizará la información básica digitalizada del paciente, obteniendo datos importantes para la atención pre-ambulatoria.
6. Llevar a efecto un estudio correlacional que establezca cuál sería la relación beneficio y costo que pueda soportar la puesta en funcionamiento de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia, estimando la viabilidad de la aplicación en grandes organizaciones.

## **REFERENCIAS**

- ALVAREZ, R., ANDROMACO, M. Demora en tiempos de respuestas a códigos amarillos en un servicio de emergencias médicas de la ciudad de Córdoba. *Revista de Salud Pública* [en línea]. 2016, **vol. 20, no 3**, p. 48-58 [consultado: 24 de febrero de 2021]. Disponible en: <https://revistas.unc.edu.ar/index.php/RSD/article/view/14541/15583>
- ARENAS, R. *Frecuencia, tipo y número de atenciones realizadas por el Samu, en la ciudad de Tacna, en los años 2017 y 2018* [en línea]. Lima, tesis de fin de grado. Universidad Privada de Tacna, 2018 [consultado: 24 de febrero de 2021]. Disponible en: <https://repositorio.upt.edu.pe/bitstream/handle/20.500.12969/688/Arenas-Mendes-Romula.pdf?sequence=1&isAllowed=y>
- BHASIN, H. *Algorithms: Design and Analysis* [en línea]. Oxford University, Singapore, 2015, **vol. 1**, p. 1-727. [consultado: 24 de febrero de 2021]. ISBN: 978-0-19-945666-6 Disponible en: <https://lib.hpu.edu.vn/handle/123456789/28981>
- BRAVO, J. *Tiempo de respuesta del centro regulador y coordinador nacional de emergencias y urgencias de Essalud según niveles de prioridad* [en línea]. Lima, tesis de fin de grado. Universidad San Martín de Porres, 2012 [consultado: 24 de febrero de 2021]. Disponible en: [https://repositorio.usmp.edu.pe/bitstream/handle/20.500.12727/1385/Bra-vo\\_j.pdf?sequence=1&isAllowed=y](https://repositorio.usmp.edu.pe/bitstream/handle/20.500.12727/1385/Bra-vo_j.pdf?sequence=1&isAllowed=y)
- BEDOYA, A., LAURENTE, C. *Análisis, diseño e implementación de un sistema de gestión del servicio de ambulancias para la atención de emergencias de una empresa aseguradora en Lima 2015* [en línea]. Lima, tesis de fin de grado. Pontificia Universidad Católica del Perú, 2014 [consultado: 24 de febrero de 2021]. Disponible en: <https://tesis.pucp.edu.pe/repositorio/handle/20.500.12404/5952>
- CAMPO, J. *Las tecnologías de información y comunicación y la gestión empresarial* [en línea]. Colombia, 2019, **vol. 1**, p. 1-844. Universidad



Simón Bolívar [consultado: 24 de febrero de 2021]. Disponible en:  
<http://www1.serbi.luz.edu.ve/pdf/9789804022395.pdf>

CARVAJAL, J. *Metodologías ágiles: Herramientas y modelo de desarrollo para aplicaciones Java EE como metodología empresarial* [en línea]. Barcelona, tesis de Maestría. Universidad Politécnica de Catalunya, 2008 [consultado: 24 de febrero de 2021]. Disponible en:  
<https://upcommons.upc.edu/bitstream/handle/2099.1/5608/50015.pdf?sequence=1&isAllowed=y>

CASTAÑEDA, J., BAREÑO, L. *Prototipo de un Sistema Concurrente para la Solicitud de Servicio de Ambulancia Utilizando Geo-Referencia* [en línea]. Bogotá, tesis de fin de grado. Universidad Distrital Francisco José de Caldas, 2015 [consultado: 24 de febrero de 2021]. Disponible en:  
[https://repository.udistrital.edu.co/bitstream/handle/11349/3220/ANDRES%20CASTA%c3%91DA\\_LIZ\\_BARE%c3%91O\\_2015.pdf?sequence=1&isAllowed=y](https://repository.udistrital.edu.co/bitstream/handle/11349/3220/ANDRES%20CASTA%c3%91DA_LIZ_BARE%c3%91O_2015.pdf?sequence=1&isAllowed=y)

CHEN, S., ZHAO, X., SHENG, Y. Research on dynamic route guidance for an emergency vehicle considering the intersection delay. *Applied Mechanics and Materials. Trans Tech Publications* [en línea]. China, 2014 **Vols. 641-642**, p. 848-852 [consultado: 24 de febrero de 2021]. Disponible en:  
<https://doi.org/10.4028/www.scientific.net/AMM.641-642.848>

COLEGIO DE INGENIEROS DEL PERÚ. Código de ética del colegio de ingenieros del Perú. Código de Ética del CIP, 26. 1999. Disponible en:  
[http://www.cip.org.pe/publicaciones/reglamentosCNCD2018/codigo\\_de\\_etica\\_del\\_cip.pdf](http://www.cip.org.pe/publicaciones/reglamentosCNCD2018/codigo_de_etica_del_cip.pdf)

CORMEN, T. et al. *Introduction to algorithms*: Third Edition. MIT Press [en línea]. Cambridge, England, 2009. [consultado: 24 de febrero de 2021]. ISBN: 978-0-262-533305-8. Disponible en:  
<https://books.google.es/books?hl=es&lr=&id=aefUBQAAQBAJ&oi=fnd&pg=PR5&dq=CORMEN,+Thomas+H.,+et+al.+Introduction+to+algorithms.>

[+MIT+press,+2009.&ots=dO6nWz-PgX&sig=ZcSsuctaDq8VqQzRU51zX1YfYPU#v=onepage&q&f=false](#)

CORTÉS, M., IGLESIAS, M. *Generalidades sobre Metodología de la Investigación*. Primera Edición. Universidad Autónoma del Carmen [en línea]. Campeche, México, 2004. [consultado: 24 de febrero de 2021]. ISBN: 968-6624-87-2. Disponible en:

<http://up-rid2.up.ac.pa:8080/xmlui/handle/123456789/1750>

CUELLO, J., VITTONI, J. *Diseñando apps para móviles*: Primera edición. Catalina Duque Giraldo [en línea]. Barcelona, España, 2013. [consultado: 24 de febrero de 2021]. ISBN: 978-84-616-5070-5. Disponible en:

[https://books.google.com.pe/books?id=ATiqsjH1rvwC&printsec=frontcover&dq=CUELLO,+Javier%3B+VITTONI,+Jos%C3%A9.+Dise%C3%B1ando+apps+para+m%C3%B3viles.+Jos%C3%A9+Vittone%E2%80%94Javier+Cuello,+2013.&hl=es&sa=X&redir\\_esc=y#v=onepage&q=CUELLO%2C%20Javier%3B%20VITTONI%2C%20Jos%C3%A9.%20Dise%C3%B1ando%20apps%20para%20m%C3%B3viles.%20Jos%C3%A9%20Vittone%E2%80%94Javier%20Cuello%2C%202013.&f=false](https://books.google.com.pe/books?id=ATiqsjH1rvwC&printsec=frontcover&dq=CUELLO,+Javier%3B+VITTONI,+Jos%C3%A9.+Dise%C3%B1ando+apps+para+m%C3%B3viles.+Jos%C3%A9+Vittone%E2%80%94Javier+Cuello,+2013.&hl=es&sa=X&redir_esc=y#v=onepage&q=CUELLO%2C%20Javier%3B%20VITTONI%2C%20Jos%C3%A9.%20Dise%C3%B1ando%20apps%20para%20m%C3%B3viles.%20Jos%C3%A9%20Vittone%E2%80%94Javier%20Cuello%2C%202013.&f=false)

DEITEL, P., DEITEL, H. *Java: cómo programar*. Séptima edición. Pearson Educación [en línea]. México, 2008. [consultado: 24 de febrero de 2021]. ISBN: 978-970-26-1190-5. Disponible en:

[https://books.google.com.pe/books?id=PoKdPwAACAAJ&dq=como+programar+en+java+deitel+2008&hl=es&sa=X&redir\\_esc=y](https://books.google.com.pe/books?id=PoKdPwAACAAJ&dq=como+programar+en+java+deitel+2008&hl=es&sa=X&redir_esc=y)

DIJKSTRA, E. W. *A note on two problems in connexion with graphs*: (Numerische Mathematik, 1 (1959), p. 269-271). [consultado: 24 de febrero de 2021]. Disponible en:

<https://www.narcis.nl/publication/recordid/oai%3Acwi.nl%3A9256>

HENCHEY, M., et al. A study of situationally aware routing for emergency responders. *Journal of the Operational Research Society* [en línea]. Buffalo, USA, 26 March 2014, Volumen 1, p. 570-578 [consultado: 24 de febrero de 2021]. DOI: [doi:10.1057/jors.2014.21](https://doi.org/10.1057/jors.2014.21)

GASCA, M., CAMARGO, L., MEDINA, B. Metodología para el desarrollo de aplicaciones móviles. *Tecnura* [en línea]. Universidad Distrital Francisco José de Caldas: abril 2014, **vol. 18, no 40**, p. 20-35 [consultado: 24 de febrero de 2021]. ISSN: 0123-921X. Disponible en: <https://www.redalyc.org/articulo.oa?id=257030546003>

ISONG, B., DLADLU, N., MAGOGODI, T. Mobile-Based Medical Emergency Ambulance Scheduling System. *International Journal of Computer Network & Information Security* [en línea]. North-West University. South Africa, 2016, **vol. 8, no 11**, p. 14-22 [consultado: 24 de febrero de 2021]. DOI: [10.5815/ijcnis.2016.11.02](https://doi.org/10.5815/ijcnis.2016.11.02). Disponible en: [https://d1wqtxts1xzle7.cloudfront.net/67080806/bdc05d0538296c8c10cc033bf06456a3c93d-with-cover-page-v2.pdf?Expires=1640944213&Signature=Gge5GNWM09qCt2sJ1lgr2pz7rKdB3j9p2NPsfXEruEgxQ~Z4-vn14xSC-u3ZI81NgdUk8iTtelkQq8uJxR1Ffd4Z3bEe1~RL0SjyAKuCTU6x4hOGI0CREcE2fY0~o3DNJPzn8W6vFSwPgQl~GVwcz3u86YBTmetd9DxpgQ6afvT3N-nsEtGgNFr2k5cRnlc-2P~2V7CDKiTIFPkfoE6wRrhCXWBB5me95bOPKN2TlyqdY-9HXQ-rcFmkW3mb3keJ-GY~T2DRz6C-4bnZEIqEkN-iGdwCy7q8NcCYU-udTKUn29OQNegpYiHEpZEe2YWDnD9gTWByzzRUyJn4l1zcQ\\_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA](https://d1wqtxts1xzle7.cloudfront.net/67080806/bdc05d0538296c8c10cc033bf06456a3c93d-with-cover-page-v2.pdf?Expires=1640944213&Signature=Gge5GNWM09qCt2sJ1lgr2pz7rKdB3j9p2NPsfXEruEgxQ~Z4-vn14xSC-u3ZI81NgdUk8iTtelkQq8uJxR1Ffd4Z3bEe1~RL0SjyAKuCTU6x4hOGI0CREcE2fY0~o3DNJPzn8W6vFSwPgQl~GVwcz3u86YBTmetd9DxpgQ6afvT3N-nsEtGgNFr2k5cRnlc-2P~2V7CDKiTIFPkfoE6wRrhCXWBB5me95bOPKN2TlyqdY-9HXQ-rcFmkW3mb3keJ-GY~T2DRz6C-4bnZEIqEkN-iGdwCy7q8NcCYU-udTKUn29OQNegpYiHEpZEe2YWDnD9gTWByzzRUyJn4l1zcQ_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA)

KIKELOMO, A. et al. Design and implementation of mobile map application for finding shortest direction between two pair locations using shortest path algorithm: a case study. *International Journal of Advanced Networking and Applications* [en línea]. Yaba, 2017, vol. 9, p. 3300-3305 [consultado: 24 de febrero de 2021]. Disponible en: <https://www.proquest.com/openview/f2c8bfc5f764555a0453855551b7761b/1?pq-origsite=gscholar&cbl=886380>

KOMATSU, N., et al. Automatic evacuation guiding scheme based on implicit interactions between evacuees and their mobile nodes. *Geoinformatica*

[en línea]. New York, 2018, **vol. 22, no 1**, p. 127-141 [consultado: 24 de febrero de 2021]. DOI: [0.1007/s10707-016-0270-1](https://doi.org/10.1007/s10707-016-0270-1). Disponible en: <https://link.springer.com/article/10.1007/s10707-016-0270-1>

LADRÓN, J. *Implementación de aplicación Android para control de dispositivo asistencial Pressmatic* [en línea]. Tesis de Licenciatura. Universidad Carlos III de Madrid, 2017 [consultado: 24 de febrero de 2021]. Disponible en: [https://e-archivo.uc3m.es/bitstream/handle/10016/26296/TFG\\_Javier\\_Ladron\\_de\\_Guevara\\_Ojalvo.pdf?sequence=1&isAllowed=y](https://e-archivo.uc3m.es/bitstream/handle/10016/26296/TFG_Javier_Ladron_de_Guevara_Ojalvo.pdf?sequence=1&isAllowed=y)

LIRA, M. Tiempo de respuesta en el transporte primario de prioridades I y II en el servicio de sistema de transporte asistido de emergencia STAE-ESSALUD. *Emergencias (St. Vicenç dels Horts)* [en línea]. San Juan de Miraflores, Lima. Perú, 2008, **vol. 20**, p. 316-321 [consultado: 24 de febrero de 2021]. Disponible en: <https://pesquisa.bvsalud.org/porta1/resource/pt/ibc-67476>

MELÉNDEZ, S., et al. *Metodología ágil de desarrollo de software programación extrema* [en línea]. MSC. Calero Castro. Tesis de postgrado. Universidad Nacional Autónoma de Nicaragua, 2016 [consultado: 24 de febrero de 2021]. Disponible en: <https://repositorio.unan.edu.ni/id/eprint/1365>

MINISTERIO DE SALUD. Reducción de la mortalidad por emergencias y urgencias médicas. *Programa Presupuestal 104* [en línea]. Lima, Perú, 2017, p. 1–166 [consultado: 24 de febrero de 2021]. Disponible en: [https://www.minsa.gob.pe/presupuestales2017/archivos\\_apelacion/anexo2/anexo2reduccion%20de%20la%20mortalidad%20por%20emergencias%20y%20urgencias%20medicas.pdf](https://www.minsa.gob.pe/presupuestales2017/archivos_apelacion/anexo2/anexo2reduccion%20de%20la%20mortalidad%20por%20emergencias%20y%20urgencias%20medicas.pdf)

NAVARRO, A., et al. Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva* [en línea]. Colombia, 2013, **vol. 11, no 2**, p. 30-39

[consultado: 24 de febrero de 2021]. ISSN: 1692-8261. Disponible en:  
<https://www.redalyc.org/pdf/4962/496250736004.pdf>

OPPEL, A., SHELDON, R. *Fundamentos de SQL*. Tercera edición. McGraw-Hill [en línea]. México, 2010 [consultado: 24 de febrero de 2021]. ISBN: 978-607-15-0251-3. Disponible en:  
<https://insebas.files.wordpress.com/2011/02/fundamentos-de-sql-3ra-edicion3b3n-andy-oppel.pdf>

ORTEGA, S., *Asociación de factores que influyen en el tiempo de llegada del sistema de atención móvil de urgencia y emergencia en el distrito de Lince* [en línea]. Tesis de pregrado. Universidad Privada San Juan Bautista, 2018 [consultado: 24 de febrero de 2021]. Disponible en:  
<http://repositorio.upsjb.edu.pe/handle/upsjb/1805>

PÉREZ, G. Optimal logistics strategy to distribute medicines in clinics and hospitals. *Journal of Mathematics in Industry* [en línea]. México, 2018, vol. 8, no 2, p. 1-18 [consultado: 24 de febrero de 2021]. DOI:  
<https://doi.org/10.1186/s13362-018-0044-5>. Disponible en:  
<https://link.springer.com/article/10.1186/s13362-018-0044-5>

RAMÍREZ, R. *Métodos para el desarrollo de aplicaciones móviles* [en línea]. Tesis de pregrado. Catalunya, España [consultado: 24 de febrero de 2021]. Disponible en:  
[https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia\\_y\\_desarrollo\\_en\\_dispositivos\\_moviles/Tecnologia\\_y\\_desarrollo\\_en\\_dispositivos\\_moviles\\_\(Modulo\\_4\).pdf](https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_(Modulo_4).pdf)

RÍOS, R. *Factores asociados al tiempo de respuesta de la prioridad I atendido por el sistema de transporte asistido de emergencia de Essalud durante el primer semestre del año 2018* [en línea]. Tesis de pregrado. Lima, Perú. Universidad Privada San Juan Bautista, 2019 [consultado: 24 de febrero de 2021]. Disponible en:  
<https://renati.sunedu.gob.pe/handle/sunedu/2843881>

ROBLEDO, D. *Desarrollo de aplicaciones para Android I*. Aula Mentor [en línea]. España, Ministerio de Educación, Cultura y Deporte, 2016. [consultado: 24 de febrero de 2021]. ISBN: 978-84-369-5687-0. Disponible en: [https://books.google.es/books?hl=es&lr=&id=PHmbDQAAQBAJ&oi=fnd&pg=PT134&dq=Robledo,+D.+\(2017\).+Desarrollo+de+aplicaciones+para+Android+I&ots=sMswfHfcPn&sig=lvZon27mr1nO1XB43DvR7BXsW-#v=onepage&q&f=false](https://books.google.es/books?hl=es&lr=&id=PHmbDQAAQBAJ&oi=fnd&pg=PT134&dq=Robledo,+D.+(2017).+Desarrollo+de+aplicaciones+para+Android+I&ots=sMswfHfcPn&sig=lvZon27mr1nO1XB43DvR7BXsW-#v=onepage&q&f=false)

RODRÍGUEZ, S. et al. Relación médico paciente y la eSalud. *Revista Cubana de Investigaciones Biomédicas* [en línea]. Cuba. **Vol. 32 (4)**, p. 411-420 [consultado: 24 de febrero de 2021]. Disponible en: [https://web.archive.org/web/20201126141206id\\_/http://scielo.sld.cu/pdf/ibi/v32n4/ibi04413.pdf](https://web.archive.org/web/20201126141206id_/http://scielo.sld.cu/pdf/ibi/v32n4/ibi04413.pdf)

SANTAMARÍA, G., HERNÁNDEZ, E. Aplicaciones Médicas Móviles: definiciones, beneficios y riesgos. *Salud Uninorte* [en línea]. Barranquilla, Colombia. **Vol. 31 (3)**, p. 599-607 [consultado: 24 de febrero de 2021]. Disponible en: <https://doi.org/10.14482/sun.31.3.7662>.

SIVAKUMAR, R., BATTA, R. *The variance-constrained shortest path problem*. *Transportation Science*, 1994, 28(4), 309-316 [consultado: 24 de febrero de 2021]. Disponible en: <https://pubsonline.informs.org/doi/abs/10.1287/trsc.28.4.309>

SKIENA, S. *The Algorithm Design Manual: Second Edition*. Springer [en línea]. Department of Computer Science, New York, 2008. [consultado: 24 de febrero de 2021]. ISBN: 978-1-84800-069-8. Disponible en: <https://link.springer.com/book/10.1007/978-1-84800-070-4>

UNIVERSIDAD CÉSAR VALLEJO. Código de ética en investigación de la Universidad César Vallejo. 2020, p. 1-12 Disponible en: <https://www.ucv.edu.pe/datafiles/C%C3%93DIGO%20DE%20C3%89TICA.pdf>

WEN-MEI, G. et al. Multiobjective route planning model and algorithm for emergency management. *Hindawi Publishing Corporation* [en línea]. Singapore, 2015. p. 1-17 [consultado: 24 de febrero de 2021]. Disponible en: <http://dx.doi.org/10.1155/2015/565403>

## Anexo 1: Matriz de consistencia

Tabla 14. Matriz de consistencia

PROBLEMAS	OBJETIVOS	HIPOTESIS	VARIABLE	DIMENSIONES	INDICADORES
General	General	General			
¿Cuál sería el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia?	Determinar el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia.	La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia reduciría el tiempo de respuesta y aumentaría el número de emergencias atendidas.	-	-	<p>PTR = Porcentaje del tiempo de respuesta.                      TPA = Tiempo de respuesta anterior.                      TDRD= Tiempo de respuesta después.</p> $PTR = \frac{TRA - TDRD}{TRA}$
Específicos	Específicos	Específicos			Indicadores
¿Cuál sería el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia en el tiempo de respuesta?	Determinar el impacto que tendría la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia en el tiempo de respuesta.	<p>La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia reduciría el tiempo de respuesta.</p> <p>(Tejedor, 2000, p. 24; Ortega, 2018, p. 20; Bravo, 2012, p. 42)</p>	<p><b>Impacto del uso de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia.</b></p> <p>(Tejedor, 2000, p.24; Ortega, 2018, p. 20; Bravo, 2012, p. 42; Bedoya y Laurente, 2014, p.8; Rodríguez, Osorno y Maya, 2015, p. 169)</p>	<p><b>Tiempo de respuesta de la solicitud.</b></p> <p>(Tejedor, 2000, p.24; Ortega, 2018, p. 20; Bravo, 2012, p. 42)</p>	<p><b>Disminución porcentual del tiempo de respuesta a solicitudes de ambulancias para el traslado de pacientes de emergencias.</b></p> $TDR = \frac{\sum(Hllegada - hsolicitud)}{N^{\circ} servicios}$ <p>TDR: Tiempo de respuesta                      Hllegada: Hora de llegada                      HSolicitud: Hora de solicitud                      N° Servicios: Número de Servicio.                      (Tejedor , 2000 ,p.24; (Ortega , 2018, p. 20; Bravo , 2012, p. 42)</p>
¿Cuál sería el impacto de la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia en el número de emergencias atendidas?	Determinar el impacto que tendría la implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes en el número de emergencias atendidas.	<p>La implementación de una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia aumentaría el número de emergencias atendidas.</p> <p>(Bedoya y Laurente, 2014, p. 8; Rodríguez, Osorno y Maya, 2015, p. 169)</p>		<p><b>Emergencias atendidas</b></p> <p>(Rodríguez, Osorno y Maya, 2015 p.169)                      (Bedoya y Laurente, 2014, p. 8)</p>	<p><b>Incremento porcentual de la cantidad de emergencias atendidas.</b></p> $PEA = \frac{NEAD - NEAA}{NEAA}$ <p>PEA=Porcentaje de emergencias atendidas.                      NEAD= Numero de emergencias atendidas después.                      NEAA= Número de emergencias atendidas antes.                      (Bedoya y Laurente, 2014, p. 8; Rodríguez, Osorno y Maya, 2015 p.169)</p>

En la tabla 14 se muestra la matriz de consistencia, la que permitió evaluar el grado de coherencia y conexión lógica entre el título, los problemas, objetivos, hipótesis, variables, dimensiones, e indicadores, además de mostrar los sustentos teóricos.



## **Anexo 2. Algunas teorías relacionadas sobre emergencias y urgencias**

Agip (2018) indicó que según la Asociación Médica Americana, la emergencia “es aquella situación urgente que pone en peligro inmediato la vida del paciente o la función de un órgano”, concluyendo que es una situación crítica de riesgo vital inminente, en la que la vida puede estar en peligro por la importancia o gravedad de la condición si no se toman medidas inmediatas (p. 13). También, Agip (2018) agregó que según la Organización de la Salud la definición de urgencia es “la aparición fortuita en cualquier lugar o actividad de un problema de causa diversa y gravedad variable que genera la conciencia de una necesidad inminente de atención por parte del sujeto que lo sufre o de su familia” (p. 13). Al respecto, Agip (2018) agregó que una urgencia es una situación en la cual no existe riesgo inminente de muerte, pero se requiere asistencia médica en un lapso reducido de tiempo según la condición para evitar complicaciones mayores (p. 13).

Al respecto, Álvarez y Andrómaco (2016) indicaron que una urgencia es una situación clínica con capacidad para generar deterioro, peligro para la salud o la vida de un paciente en función del tiempo transcurrido entre su aparición y la instauración de un tratamiento efectivo (p. 50).

### **A. Prioridad I (emergencia o gravedad súbita extrema):**

Agip (2018) indicó que la prioridad I se identifica por atender pacientes con alteración súbita y crítica del estado de salud, en riesgo inminente de muerte, que requieren atención inmediata en la sala de resucitación o trauma shock (p. 14).

### **B. Prioridad II (urgencia mayor):**

Agip (2018) clasificó esta prioridad como pacientes portadores de cuadro súbito, agudo con riesgo de muertes o complicaciones serias, cuya atención debe realizarse en un tiempo de espera no mayor o igual de 10 minutos (p. 17).

### **C. Prioridad III (urgencia menor):**

Ortega (2018) indicó que en esta prioridad los pacientes no presentan riesgo de muerte ni secuelas invalidantes. Ameritan su atención en el tópico de emergencia (p. 21).

### **Anexo 3. Algunas teorías relacionadas sobre algoritmos**

Bhasin (2015) definió el algoritmo como los pasos a seguir para llevar a cabo una tarea en particular (p. 3). Al respecto, Bhasin (2015) agregó que un algoritmo es un proceso paso a paso para analizar y resolver el problema de una manera lógica. Se requiere un algoritmo bien diseñado para desarrollar códigos de programa eficientes, así como para minimizar el uso de recursos informáticos (p. 3).

Del mismo modo Erickson (2019) describió que el algoritmo es una secuencia de instrucciones elementales explícitas, precisas, inequívocas y mecánicamente ejecutables, generalmente destinadas a lograr un propósito específico (p. 1). Además, Erickson (2019) consideró que los programas de computadora son representaciones concretas de algoritmos, pero los algoritmos no son programas. Más bien, los algoritmos son procedimientos mecánicos abstractos que pueden implementarse en cualquier lenguaje de programación que admita las operaciones primitivas subyacentes (p. 13).

Cormen et al. (2012) indicaron que informalmente, un algoritmo es cualquier procedimiento computacional bien definido que toma algún valor, o conjunto de valores, como entrada y produce algún valor, o conjunto de valores, como salida. Un algoritmo es, por lo tanto, una secuencia de pasos computacionales que transforman la entrada en la salida (p. 5).

Del mismo modo, Skiena (2008) definió el algoritmo como un procedimiento para llevar a cabo una tarea específica, un algoritmo es la idea detrás de cualquier programa de computadora razonable (p. 3). Además, Skiena (2008) agregó que un algoritmo debe resolver un problema general y bien especificado. Un problema algorítmico se especifica describiendo el conjunto completo de instancias en las que debe trabajar y de su salida después de ejecutarse en una de estas instancias (p. 3).

Ladrón (2013) comentó que Java es un lenguaje de programación desarrollado por Sun Microsystems. Java fue presentado en la segunda mitad de 1995 y desde entonces se ha convertido en un lenguaje de programación muy popular. Java es un lenguaje muy valorado porque los programas Java se pueden ejecutar en diversas plataformas con sistemas operativos como Windows, Mac OS, Linux o Solaris (p. 2).

#### **Anexo 4. Algunas teorías relacionadas a las tecnologías y aplicaciones móviles**

Santamaría y Hernández (2015) mencionaron que en los últimos años se ha visto el desarrollo constante en las tecnologías de las telecomunicaciones, con un uso creciente de dispositivos móviles inalámbricos en la población, con la capacidad de acceso a información en todo momento, incluyendo los temas en salud (p. 599). Además, Santamaría y Hernández (2015) agregaron que el desarrollo de estas tecnologías ha llevado a la aparición de los denominados “teléfonos inteligentes”, que son teléfonos inalámbricos con mayor capacidad de cómputo y conectividad, con capacidad de uso de aplicaciones (p. 599).

Rodríguez et al. (2013) comentaron que la directora de la OPS/OMS, Carissa F. Etienne, en un boletín informativo de eSalud, afirmó que las tecnologías de la información y la comunicación (TIC) aplicadas a la salud, como la telemedicina, la salud móvil y los registros de salud electrónicos, "mejoran el acceso a los servicios, su eficiencia y calidad, las iniciativas de telemedicina están ayudando a llegar a las poblaciones vulnerables y aisladas, para prestar servicios de salud. Esto nos permite superar las barreras geográficas que dificultan el acceso a la atención de salud (p. 417).

Al respecto, Santamaría y Hernández (2015) explicaron que se entiende por apps aquellas aplicaciones de software que funcionan en teléfonos móviles, tabletas o computadoras y que son distribuidos a través de servicios o tiendas (p. 600). Además, Santamaría y Hernández (2015) agregaron que las apps presentan potenciales beneficios, como mejorar la calidad de prestación y evitar errores médicos mediante algoritmos clínicos, disminuir costos, evitan consultas médicas innecesarias y mejoran el acceso a los servicios de salud (p. 602).

Asimismo, Cuello y Vittone (2013) indicaron que últimamente, no solo se han vuelto más populares y atractivas para los usuarios, sino también para los diseñadores y desarrolladores que están sacando provecho de las posibilidades que ofrecen las nuevas pantallas de teléfono de mayor calidad (p.7). Además, Cuello y Vittone (2013) agregaron que, en esencia, una aplicación no deja de ser un software. Para entender un poco mejor el concepto, podemos decir que las aplicaciones son para los móviles lo que los programas son para los ordenadores de escritorio (p. 14).

## **Anexo 5. Algunas teorías relacionadas sobre IDE de desarrollo Android Studio**

Kikelomo et al. (2017) indicaron que Android Studio es un software utilizado en la codificación y el diseño de aplicaciones Android nativas. La razón de la elección del lenguaje de programación se debe a que el sistema operativo Android es uno de los principales sistemas operativos que dominan el mundo de los dispositivos móviles hoy en día y como los dispositivos móviles también están de moda, reduciendo el uso habitual de ordenadores portátiles y de sobremesa, se tomó la decisión de diseñar una aplicación móvil (p. 3302).

Al respecto, Kikelomo et al. (2017) diseñaron e implementaron una aplicación de mapas usando el estudio de Android, y Android SDK. El sistema ha sido probado y cumple con las normas estándar y solución requerida para resolver el problema identificado (p. 3305).

Asimismo, Cuello y Vittone (2013) comentaron que para programar aplicaciones para este sistema operativo es indistinto tener un Mac o un PC con Windows o con Linux. Se puede descargar el software Android Studio y todo el material necesario para desarrollar una app desde la web de desarrolladores de Android. En el momento del desarrollo, Android Studio permite usar los simuladores de diferentes dispositivos, o conseguir una prueba de funcionamiento más real conectando el terminal al ordenador (p. 43).

También, Robledo (2017) indicó que Android Studio es un programa que requiere un PC con bastantes recursos de memoria y CPU, sobre todo al renderizar componentes gráficos o al compilar aplicaciones. Un equipo actualizado, con la última versión del sistema operativo, un buen procesador y memoria suficiente (2GB mínimo, 4GB recomendando) ayudan mucho a mejorar el rendimiento general y los tiempos de respuesta. Recomendamos a los alumnos que lean con atención en la página de descarga los requerimientos mínimos de hardware antes de instalarlo en su PC. (p. 9). Para la elaboración de este proyecto se utilizará Android Studio 3.0.0.

## **Anexo 6. Algunas teorías relacionadas de la Api de Google Maps**

Deitel y Deitel (2008) indicaron que la mayoría de los programadores en Java aprovechan las ricas colecciones de clases existentes en las bibliotecas de clases de Java, que también se conocen como APIs (Interfaces de programación de aplicaciones) de Java. Por lo tanto, en realidad existen dos fundamentos para conocer el “mundo” de Java. El primero es el lenguaje Java en sí, de manera que usted pueda programar sus propias clases; el segundo son las clases incluidas en las extensas bibliotecas de clases de Java (p. 8).

Además, Deitel y Deitel (2008) agregaron que utilizar las clases y métodos de las APIs de Java en vez de escribir sus propias versiones puede mejorar el rendimiento de sus programas, ya que estas clases y métodos están escritos cuidadosamente para funcionar de manera eficiente, reduciendo el tiempo de desarrollo de los programas (p. 9).

Al respecto Campo (2019) indicó que la API de Google Maps es una aplicación de mapeo en línea, que requiere el acceso a las fuentes de datos en el formato de una base de datos comercial almacenado en un seguro de datos servidor y que ofrece funcionalidades sofisticadas para el usuario para manipular los datos; esta Api se usó en el proyecto a la hora en que haya una emergencia y se quiera saber cuál ambulancia se encuentra más cerca al sitio del acontecimiento, y así reducir el tiempo de llegada al lugar; es de gran utilidad, ya que se podrá evitar que el usuario permanezca mucho tiempo en el lugar de la emergencia sin asistencia, y así mejorar el servicio de traslado asistencial básico (Campo, 2019, p. 134) .

## **Anexo 7. Metodologías de desarrollo**

### **I. Introducción**

#### **a. Metodología de desarrollo**

Avison y Fitzgerald (1995) presentaron una descripción de las metodologías muy clara citado por Carvajal (2008) mencionó lo siguiente:

“Una metodología es una colección de procedimientos, técnicas, herramientas y documentos auxiliares que ayudan a los desarrolladores de software en sus esfuerzos por implementar nuevos sistemas de información. Una metodología está formada por fases, cada una de las cuales se puede dividir en sub-fases, que guiarán a los desarrolladores de sistemas a elegir las técnicas más apropiadas en cada momento del proyecto y también a planificarlo, gestionarlo, controlarlo y evaluarlo” (p. 59).

Al respecto, Chandi (2017) comentó que el crecimiento de las apps móviles ha dado lugar a que sus desarrolladores adapten metodologías de desarrollo o surjan nuevas propuestas que permitan cumplir con las particularidades que implica el desarrollo de apps móviles. Dichas particularidades pueden ser: ciclos de desarrollos cortos y frecuentes, cambios tecnológicos (plataformas, sistemas operativos, sensores, etc.), documentación limitada, requisitos y recursos específicos del equipo de desarrollo y del cliente o usuario final, entre otros; siendo todos estos sujetos de innovación constante (p. 1).

Gasca et al. (2013) mencionaron que el escenario tecnológico actual posiciona a los desarrolladores de servicios como la nueva fuerza de negocios en las redes de próxima generación Application Stores, allí radica la importancia de unificar metodologías y entornos para el desarrollo de servicios, que faciliten la creación de estos con alto nivel de calidad, llevando dicho desarrollo al éxito, de forma atractiva y eficiente (p. 22). Del mismo modo, Chandi (2017) agregó que el proceso de desarrollo de aplicaciones móviles es fundamental, así el equipo de trabajo llega a identificar sus tareas, tiempos, recursos, objetivos, etc., logrando la elaboración eficaz y eficiente del producto, y cumpliendo con los requisitos del sistema de la mejor manera posible (p. 1).

## **b. Metodologías no ágiles**

Al respecto, Navarro et al. (2013) indicaron que las metodologías tradicionales de desarrollo de software son orientadas por planeación. Inician el desarrollo de un proyecto con un riguroso proceso de licitación de requerimientos, previo a etapas de análisis y diseño. Con esto tratan de asegurar resultados con alta calidad circunscritos a un calendario (p. 31).

Asimismo, Navarro et al. (2013) agregaron que en las metodologías tradicionales se concibe un solo proyecto, de grandes dimensiones y estructura definida; se sigue un proceso secuencial en una sola dirección y sin marcha atrás; el proceso es rígido y no cambia; los requerimientos son acordados de una vez y para todo el proyecto, demandando grandes plazos de planeación previa y poca comunicación con el cliente (p. 31).

## **c. Metodologías ágiles**

Al respecto, Meléndez et al. (2016) indicaron que las metodologías ágiles resuelven los problemas surgidos, posteriormente, a la masificación del uso del computador personal, dado que las expectativas y necesidades por parte de los usuarios se hicieron más urgentes y frecuentes (p. 24).

Asimismo, Chandi (2017) precisó que los métodos ágiles nacen como respuesta a potenciar el desarrollo de software, en lugar de su planificación. Estos métodos dan mayor importancia a la capacidad de respuesta, en lugar del seguimiento estricto de un plan. Estas propuestas nacieron puesto que se requiere en muchos proyectos de mayor flexibilidad, de permitir cambios sin que implique un costo excesivo (p. 21). Además, Chandi (2017) agregó que las metodologías ágiles ofrecen un desarrollo más ligero y rápido. Han logrado un alto nivel de popularidad y el éxito en la industria de software reconociendo que una buena relación cliente / desarrollador es crucial para el éxito en el desarrollo de software (p. 21).

Del mismo modo, Carvajal (2008) indicó que las metodologías ágiles les dan una mayor importancia a las personas en vez de a los procesos y se caracterizan principalmente por el uso de técnicas para agilizar el desarrollo del software, así como de una mayor flexibilidad para adaptarse a los cambios en

los requisitos del proyecto (p. 12). Además, Navarro et al. (2013) agregaron que las metodologías ágiles tienen dos diferencias fundamentales con las metodologías tradicionales; la primera es que los métodos ágiles son adaptativos no predictivos y la segunda diferencia es que las metodologías ágiles son orientadas a las personas no orientadas a los procesos (p. 30).

#### **d. Metodologías Mobile –D**

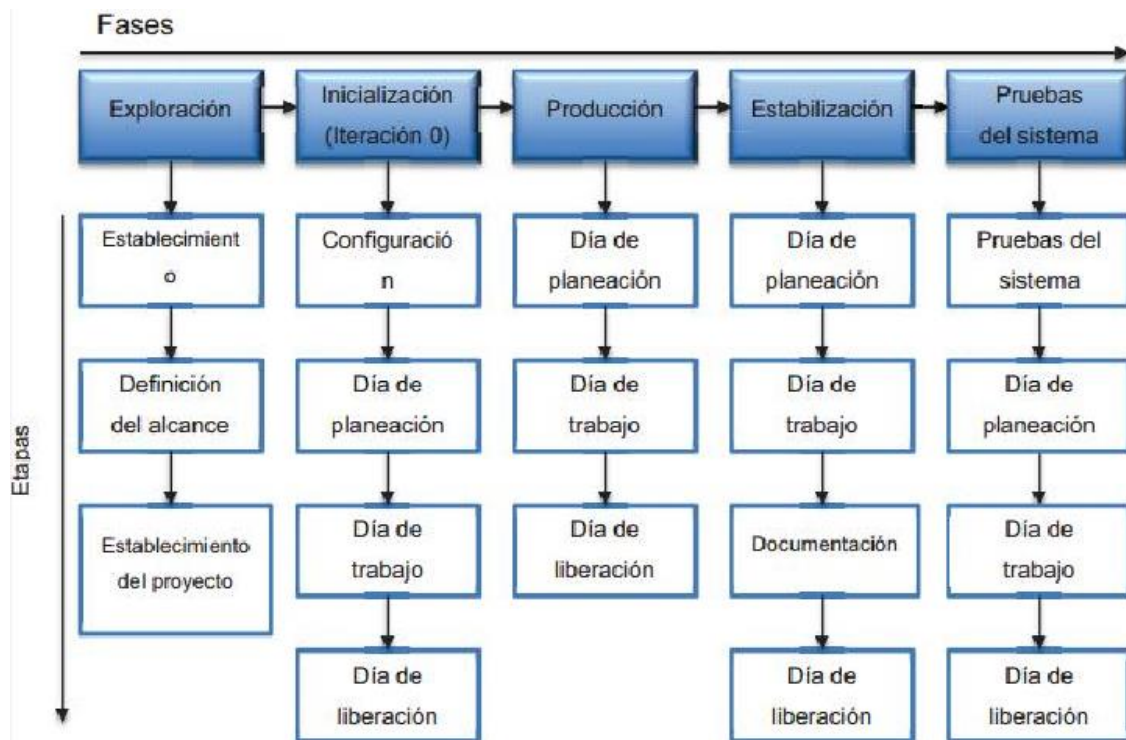
Amaya (2013) indicó que la Metodología ágil Mobile-D es propuesta por Pekka Abrahamsson y su equipo del VTT (Valtion Teknillinen Tutkimuskeskus, en inglés Technical Research Centre of Finland) en Finlandia que lidera una corriente muy importante de desarrollo ágil muy centrada en las plataformas móviles (p. 118). También, Ramírez (2011) precisó que el objetivo es conseguir ciclos de desarrollos muy rápidos en equipos muy pequeños (de no más de diez desarrolladores) trabajando en un mismo espacio físico. Según este método, trabajando de esa manera se deben conseguir productos totalmente funcionales en menos de diez semanas (p. 42). Además, Ramírez (2011) agregó que se trata de un método basado en soluciones conocidas y consolidadas: Extreme Programming (XP), Crystal Methodologies y Rational Unified Process (RUP), XP para las prácticas de desarrollo, Crystal para escalar los métodos y RUP como base en el diseño del ciclo de vida (p. 42). Sus etapas son las siguientes:

- Exploración: se dedica a la planificación y a los conceptos básicos del proyecto. Es diferente del resto de fases ya que se realiza en tres etapas: establecimiento de actores, definición del alcance y el establecimiento de proyectos.
- Inicialización: se preparan e identifican todos los recursos necesarios. Se establece el entorno técnico. Esta etapa se divide en cuatro etapas: la puesta en marcha del proyecto, la planificación inicial, el día de prueba y el día de salida.
- Productización o fase de producto: se repiten iterativamente las subfases, con un día de planificación, uno de trabajo y uno de entrega.
- Fase de estabilización: se llevan a cabo las acciones de integración para asegurar que el sistema completo funciona correctamente.



- Fase de pruebas y reparación: tiene como meta la disponibilidad de una versión estable y plenamente funcional del sistema según los requisitos del cliente.

Figura 2. Fases y etapas de la metodología Mobile -D



## **Anexo 8. Etapas de la metodología Mobile-D para el desarrollo de la aplicación de asignación de ambulancias para el traslado de pacientes de emergencia.**

### **I. FASE I: EXPLORACIÓN**

#### **a. Establecimiento de los grupos de interés o stakeholders**

En este estudio se determinaron los roles de las personas involucradas en el proyecto y se especificaron sus respectivas tareas y responsabilidades:

##### Área de desarrollo:

- Huaroto Quispe, Julio Cesar

##### Asesor de Proyecto:

- Dr. Alfaro Paredes, Emigdio Antonio

##### Grupo de Interés:

- Empresas del sector salud, hospitales nacionales o aseguradoras de Lima Metropolitana los cuales por el número de solicitudes de emergencia que reciben diariamente, se observa una deficiencia en el uso de sus aplicaciones móviles para optimizar su proceso de solicitud de emergencias.
- Usuario de la aplicación, es el actor encargado de hacer uso de la aplicación, solicitud de servicio y aceptación del mismo, en este caso estudiantes universitarios.

#### **b. Objetivos:**

- La aplicación móvil tiene como propósito agilizar el método de solicitud de ambulancias para pacientes de emergencia.
- Geolocalizar al paciente al momento de procesar la solicitud mediante su latitud y longitud.
- Mejorar el tiempo de respuesta en la solicitud, asignando la ambulancia que cumpla con la condición del tipo de emergencia y la que se encuentre más cercana.

- Aumentar la disponibilidad de las unidades ambulatorias y por ende el número de emergencias atendidas.

**c. Alcance:**

- Desarrollar una aplicación móvil de asignación de ambulancias para el traslado de pacientes de emergencia.

**d. Limitaciones**

- Geográfica: Esta aplicación móvil será utilizada dentro de las latitudes y longitudes del distrito de San Juan de Lurigancho.

**e. Definición del proyecto**

- Para el desarrollo de la aplicación móvil, se utilizará la plataforma gratuita de Android Studio 3.0, SQL Server 2008 para almacenamiento de la base de datos y el servicio del Hosting SmarterASP.net de manera gratuita.

Tabla 15. Requerimientos funcionales

Código	Descripción
RF001	Identificar al usuario por su Id y clave.
RF002	Botón de pánico para la solicitud de ambulancias.
RF003	ComboBox para la selección del nivel de emergencia.
RF004	Botón Simular para iniciar la simulación del trayecto de las ambulancias al punto de solicitud.
RF005	API de Google Maps para visualizar el trayecto de la ambulancia hacia el punto de solicitud.
RF006	Textbox para el ingreso de número de solicitudes.
RF006	Botón generar para realizar la simulación de solicitudes de manera masiva.

Tabla 16. Requerimientos no funcionales

Código	Descripción
RNF001	La aplicación móvil será desarrollada en la plataforma Android Studio 3.0.
RFN002	La base de datos se creará con SQL Server Management Studio 2008.
RFN003	Se desarrollará el proyecto bajo el lenguaje de programación Java.
RFN004	La aplicación podrá ser accedido por el administrador y usuarios previamente registrados.

Tabla 17. Modelo de procesos

Modulo	Código	Proceso	Requerimiento
Módulo de identificación del usuario.	M001	Identificación del usuario y el administrador mediante id y clave.	RF001
Módulo menú principal.	M002	Modulo que permite elegir el módulo que desea utilizar el usuario.	RNF001, RNF002, RNF003, RFN004.
Módulo Solicitud	M003	Este módulo permite solicitar ambulancias para el tipo de emergencia requerido.	RF002, RF003, RNF001, RNF002, RNF003, RFN004.
Módulo Simulación	M004	Este módulo permite simular la llegada de las ambulancias al punto de los solicitantes.	RF004, RF005, RNF001, RNF002, RNF003, RFN004.
Módulo Generación de Solicitudes	M005	Este módulo permite generar solicitudes aleatorias a demanda.	RF006, RF007, RNF001, RNF002, RNF003, RFN004.

## II. FASE II: INICIALIZACION

### Establecimiento del entorno técnico

- En esta etapa se determinan los recursos físicos y técnicos necesarios para el inicio del desarrollo de la aplicación móvil. Las herramientas que utilizara son las siguientes:

Tabla 18. Datos técnicos de laptop Acer

Laptop Acer Aspire 3	
Modelo	A315-53G-56ZT
Sistema operativo	Windows 10 Home Single Language 64 bits.
Procesador	Intel Core I5-8250u 1.6 GHz
Generación	Octava generación.
RAM	8000 MB
Disco Duro	1 TB

Tabla 19. Datos técnicos de laptop Toshiba

Laptop Toshiba Satellite	
Modelo	L745-SP4251LL
Sistema operativo	Windows 10 Home Single Language 64 bits.
Procesador	Intel Core I5-2450 2.50 GHz
Generación	Quinta generación.
RAM	4096 MB
Disco Duro	500 GB

Tabla 20. Datos técnicos de móvil Samsung

Celular Samsung	
Modelo	SM-A505G
Sistema operativo	Android Pie 9
Procesador	Exynos 9610.
RAM	4 GB
Almacenamiento	64 GB

Tabla 21. Datos técnicos de móvil Lenovo

Celular Lenovo	
Modelo	A6020I37
Sistema operativo	Android 5.1.1 Lollipop
Procesador	Qualcomm MSM8929 Snapdragon 415
RAM	2 GB
Almacenamiento	16 GB

**a. Planeamiento inicial**

Durante esta fase, se planifican exposiciones de planos de proyectos y arquitectura de la solución. El esquema de la investigación se dividió en tres partes:

- Aplicación Móvil
- Servicio Web
- Base de datos

Figura 3. Entorno de funcionamiento del App

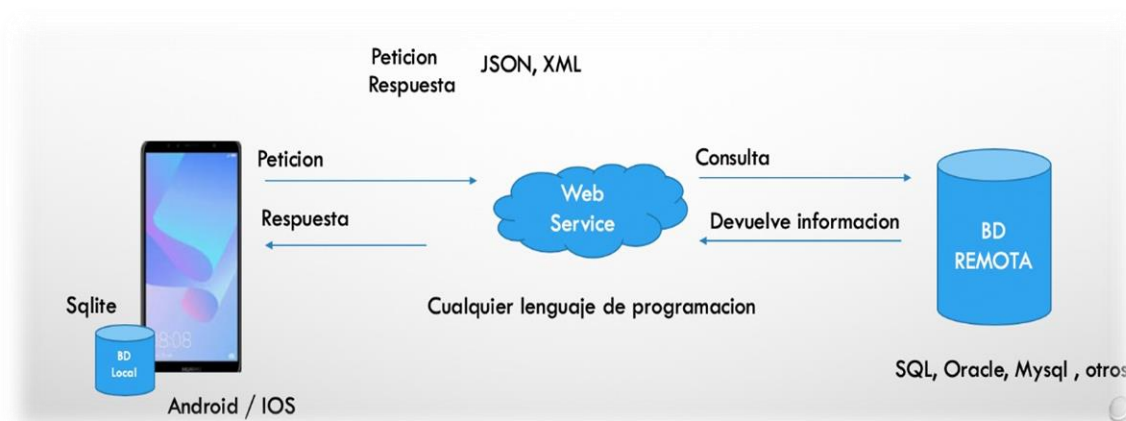


Tabla 22. Storycard identificación de usuario

NUMERO	01	DIFICULTAD	Medio	PRIORIDAD	Alta	
DESCRIPCION						
<p>El usuario (paciente) o administrador deberá ingresar al programa por medio de un id (letra del primer nombre seguido del apellido paterno) y clave (número de DNI) para poder identificarse.</p>						
EXEPCIONES						
<p>Solo podrá ingresar el administrador y usuarios anteriormente registrados en la base de datos. Si el celular no está conectado a la red, el proceso de identificación no podrá ejecutarse satisfactoriamente.</p>						
FECHA	09/08/19	ESTADO	Definido			
	22/08/19		Hecho			
	05/09/19		Implementado			
	11/09/19		Prueba			
	18/09/19		Verificado			

Tabla 23. Storycard modulo menú principal

NUMERO	02	DIFICULTAD	Alta	PRIORIDAD	Alta	
DESCRIPCION						
<p>El usuario (paciente) o administrador al terminar de validar correctamente su usuario y contraseña se cargará el módulo menú principal donde se mostrarán los módulos solicitud, simulación y generación de solicitudes.</p>						
EXEPCIONES						
<p>Si el celular no está conectado a la red, el proceso de los tres módulos no podrá ejecutarse satisfactoriamente.</p>						
FECHA	09/08/19	ESTADO	Definido			
	22/08/19		Hecho			
	05/09/19		Implementado			
	11/09/19		Prueba			
	18/09/19		Verificado			

Tabla 24. Storycard Solicitud de ambulancia.

NUMERO	03	DIFICULTAD	Alta	PRIORIDAD	Alta	
DESCRIPCION						
<p>El usuario (paciente) o administrador al seleccionar el módulo solicitud se mostrará una ventana con un botón de pánico de solicitud de emergencia, además un combobox que le permitirá seleccionar el nivel de emergencia que más se adecue a su solicitud.</p>						
EXEPCIONES						
<p>Si el celular no está conectado a la red, el proceso del módulo solicitud no podrá ejecutarse satisfactoriamente. Si no selecciona el nivel de emergencia no se procesará su solicitud.</p>						
FECHA	09/08/19	ESTADO	Definido			
	22/08/19		Hecho			
	05/09/19		Implementado			
	11/09/19		Prueba			
	18/09/19		Verificado			

Tabla 25. Storycard similar

NUMERO	04	DIFICULTAD	Alta	PRIORIDAD	Alta	
DESCRIPCION						
<p>El usuario (paciente) o administrador al seleccionar el módulo solicitud se mostrará una ventana con un botón de pánico de solicitud de emergencia, además un combobox que le permitirá seleccionar el nivel de emergencia que más se adecue a su solicitud.</p>						
EXEPCIONES						
<p>Si el celular no está conectado a la red, el proceso del módulo solicitud no podrá ejecutarse satisfactoriamente. Si no selecciona el nivel de emergencia no se procesará su solicitud.</p>						
FECHA	09/08/19	ESTADO	Definido			
	22/08/19		Hecho			
	05/09/19		Implementado			
	11/09/19		Prueba			
	18/09/19		Verificado			



Tabla 26. Storycard generar solicitudes

NUMERO	05	DIFICULTAD	Alta	PRIORIDAD	Alta
DESCRIPCION					
<p>El usuario (paciente) o administrador al seleccionar el módulo generación de solicitudes se mostrará una ventana con un textbox donde se ingresará el número de solicitudes, dos botones uno para generar el número de solicitudes de manera aleatoria y un botón que le permitirá calcular el tiempo promedio de respuesta por cada iteración realizada.</p>					
EXEPCIONES					
<p>Si el celular no está conectado a la red, el proceso del módulo generación de solicitudes no podrá ejecutarse satisfactoriamente. Si no ingresa el usuario el número de solicitudes no se procesará su solicitud.</p>					
FECHA	09/08/19	ESTADO	Definido		
	22/08/19		Hecho		
	05/09/19		Implementado		
	11/09/19		Prueba		
	18/09/19		Verificado		

2:52

← AppAmbulancias

24

Generar

$$TDR = \frac{\sum(Hllegada - hsolicitud)}{N^{\circ} servicios}$$

Calcular

TDR: 16.52

Anexo 3.2.5 Modelo relacional de la base de datos.

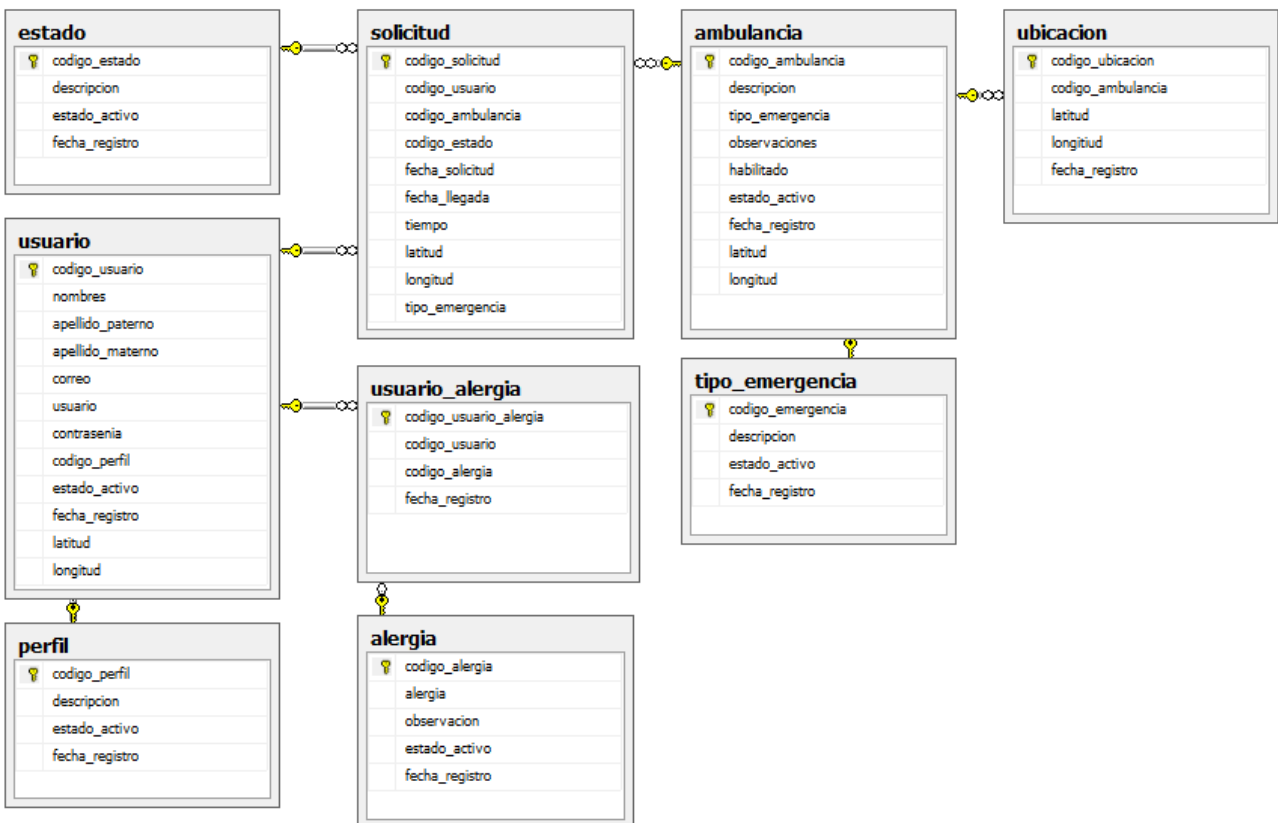


Figura 4: Modelo relacional de la base de datos.

## Anexo 9: Diccionario de la base de datos.

Tabla 27. Descripción de la tabla Ambulancia

Columna	Descripción	Tipo	Nulo
Codigo_ambulancia	Código que identifica la ambulancia.	int	No
Descripción	Descripción estado de la ambulancia.	varchar(250)	No
Tipo_emergencia	Nivel de emergencia de la solicitud	int	No
Observaciones	Observaciones técnicas de la ambulancia.	varchar(500)	No
Habilitado	Habilita la ambulancia online u offline.	bit	No
Estado_activo	Identifica si la ambulancia está en servicio.	bit	No
Fecha_registro	Fecha que se registró la ambulancia.	datetime	No
latitud	Detalla la latitud actual de la ambulancia.	varchar (150)	No
longitud	Detalla la longitud actual de la ambulancia.	varchar (150)	No

Tabla 28. Descripción de la tabla Estado

Columna	Descripción	Tipo	Nulo
Codigo_estado	Código que identifica el estado de la solicitud,	int	No
Descripción	Descripción del estado de la solicitud.	varchar(250)	No
Estado_activo	Estado actual en el registro de la solicitud,	bit	No
Fecha_registro	Fecha que se registró la modificación del estado de la solicitud.	datetime	No

Tabla 29. Descripción de la tabla Alergia

Columna	Descripción	Tipo	Nulo
Codigo_alergia	Código que identifica la alergia.	int	No
Alergia	Descripción de la alergia.	varchar(200)	No
Observación	Observación de la alergia.	varchar(500)	No
Estado_activo	Habilita código alergia.	bit	No
Fecha_registro	Fecha que se registró la alergia.	datetime	No

Tabla 30. Descripción de la tabla Perfil

Columna	Descripción	Tipo	Nulo
Codigo_perfil	Código que identifica tipo de usuario.	int	No
Descripcion	Descripción del tipo de usuario.	varchar(250)	No
Estado_activo	Estado del usuario en la base de datos.	bit	No
Fecha_registro	Fecha que se registró al usuario.	datetime	No

Tabla 31. Descripción de la tabla Solicitud.

Columna	Descripción	Tipo	Nulo
Codigo_solicitud	Código que identifica la solicitud.	int	No
codigo_usuario	Código que identifica al usuario solicitante.	int	No
codigo_ambulancia	Código que identifica la ambulancia.	int	No
codigo_estado	Código que identifica el estado de la solicitud	int	No
fecha_solicitud	Fecha y hora que ingresa la solicitud.	datetime	No
fecha_llegada	Fecha y hora de llegada de la ambulancia al punto de solicitud.	datetime	No
tiempo	Tiempo estimado de llegada al punto de solicitud.	varchar(20)	No
latitud	Latitud del punto de la solicitud.	varchar(250)	No
longitud	Longitud del punto de la solicitud.	varchar(250)	No
tipo_emergencia	Nivel de emergencia de la solicitud.	int	No

Tabla 32. Descripción de la tabla Tipo\_emergencia.

Columna	Descripción	Tipo	Nulo
codigo_emergencia	Código que identifica el tipo de emergencia.	int	No
Descripcion	Descripción del tipo de emergencia.	varchar(250)	No
Estado_activo	Estado del tipo de emergencia en la base de datos.	bit	No
Fecha_registro	Fecha que se registró el tipo de emergencia.	datetime	No

Tabla 33. Descripción de la tabla Ubicación.

Columna	Descripción	Tipo	Nulo
codigo_ubicacion	Código que identifica la ubicación.	int	No
codigo_ambulancia	Código que identifica la ambulancia.	int	No
latitud	Latitud actualizada de la ubicación de la ambulancia.	float	No
longitud	Longitud actualizada de la ubicación de la ambulancia.	float	No
fecha_registro	Fecha y hora actualizada de la ubicación.	datetime	No

Tabla 34. Descripción de la tabla Usuario.

Columna	Descripción	Tipo	Nulo
Codigo_usuario	Código que identifica al usuario solicitante.	int	No
Nombres	Nombres del usuario solicitante.	varchar(250)	No
Apellido_paterno	Apellido paterno del usuario solicitante.	varchar(250)	No
Apellido_materno	Apellido materno del usuario solicitante.	varchar(250)	No
correo	Correo del usuario solicitante.	varchar(250)	No
usuario	Id del usuario solicitante.	varchar(100)	
contrasenia	Contraseña del usuario.	varchar(50)	No
codigo_perfil	Código de perfil que le corresponde al usuario.	int	No
estado_activo	Estado del usuario en la base de datos.	bit	No
fecha_registro	Fecha y hora en la que se registró al usuario.	datetime	No
latitud	Latitud actualizada de la ubicación del usuario.	varchar(150)	No
longitud	Longitud actualizada de la ubicación del usuario.	varchar(150)	No

Tabla 35. Descripción de la tabla Usuario\_alergia.

Columna	Descripción	Tipo	Nulo
codigo_usuario_alergia	Código que identifica usuario - alergia.	int	No
cosigo_usuario	Código que identifica al usuario solicitante.	int	No
codigo_alergia	Código que identifica la alergia.	int	No
fecha_registro	Fecha y hora de registro.	int	No

### III. FASE III: PRODUCCION

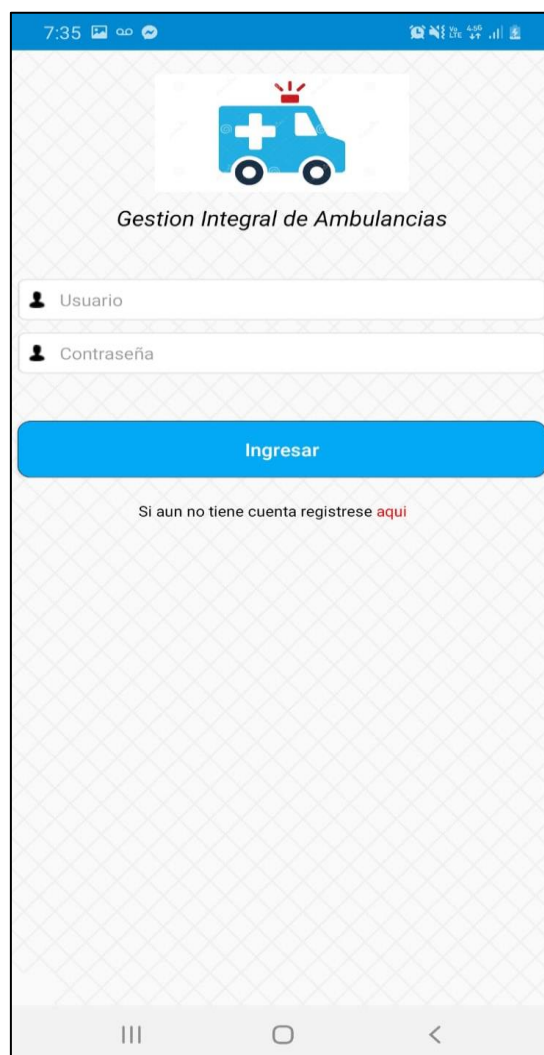


Figura 5. Pantalla de identificación de usuario.

## a. Codificación de la Storycard identificación de usuario.

```
public class LoguinActivity extends AppCompatActivity implements
Validator.ValidationListener {

    @Required(order = 1, message = "Debe ingresar el usuario")
    @BindView(R.id.edt_usuario)
    EditText edt_usuario;

    @Required(order = 2, message = "Debe ingresar la contraseña")
    @BindView(R.id.edt_contraseña)
    EditText edt_contraseña;

    SweetAlertDialog pd;

    Validator validator;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_loguin);
        ButterKnife.bind(this);

        init();
    }

    private void init() {

        validator = new Validator(this);
        validator.setValidationListener(this);
    }

    @OnClick(R.id.btn_ingresar)
    public void ingresar() {

        validator.validate();
    }

    @OnClick(R.id.tv_nuevo_usuario)
    public void nuevoUsuario() {

        pd = new SweetAlertDialog(LoguinActivity.this,
SweetAlertDialog.WARNING_TYPE);

pd.getProgressHelper().setBarColor(Color.parseColor("#03A9F4"));
pd.setContentText("Disponible proxicamente!!");
pd.setCancelable(false);
pd.show();
return;
    }
}
```

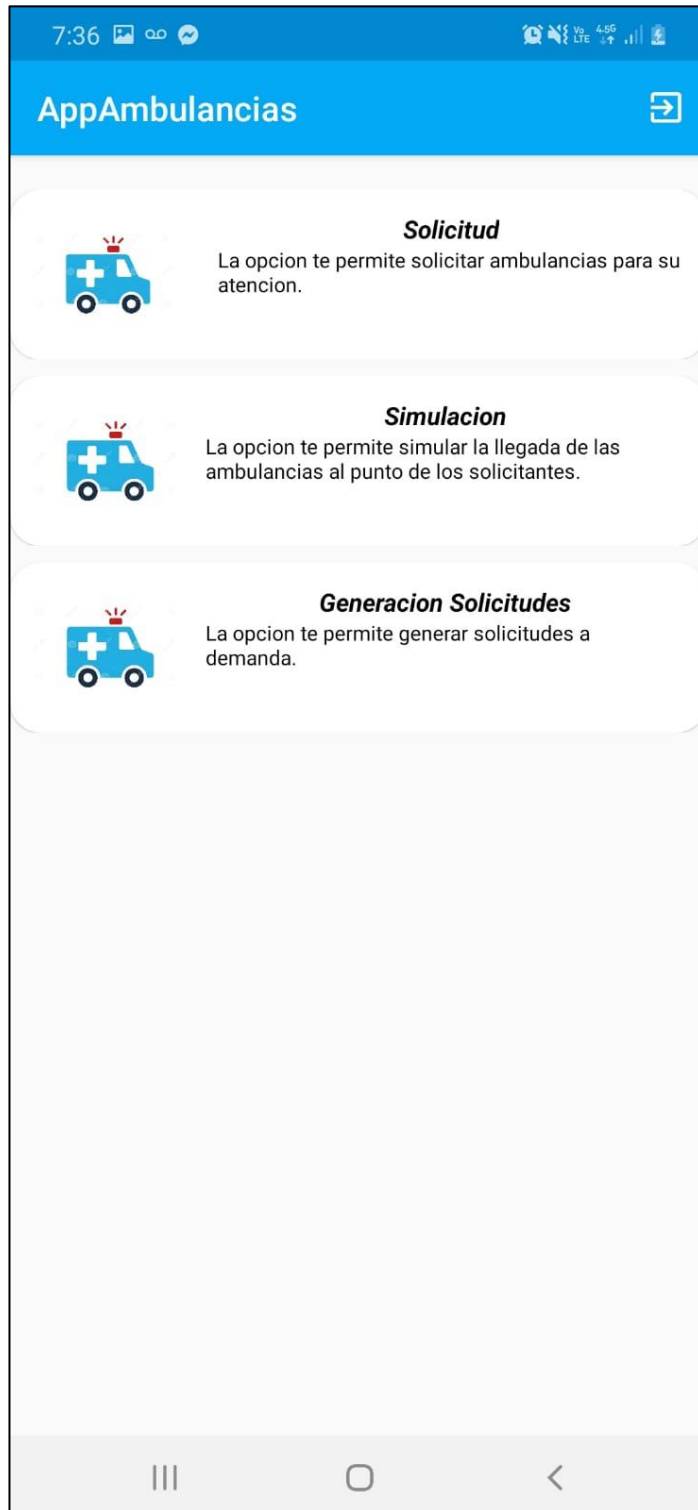


Figura 6. Pantalla del menú principal.

## b. Codificación de la Storycard menú principal.

```
public class MenuActivity extends AppCompatActivity {
    String usuario;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_menu);
        ButterKnife.bind(this);
        Bundle bundle = getIntent().getExtras();
        if (bundle != null) {
            usuario = bundle.getString("usuario");
        } else {
            usuario = "Visitante";
        }
    }
    @OnClick(R.id.btn_solicitud)
    public void solicitud() {
        Bundle bundle = new Bundle();
        bundle.putString("usuario", usuario);
        Intent intent = new
Intent(MenuActivity.this, SolicitarAmbulanciaActivity.class);
        intent.putExtras(bundle);
        startActivity(intent);
    }
    @OnClick(R.id.btn_simulacion)
    public void simulacion() {
        Intent intent = new
Intent(MenuActivity.this, SeguimientoActivity.class);
        startActivity(intent);
    }
    @OnClick(R.id.btn_generacion_solicitudes)
    public void generacionSolicitudes() {
        Intent intent = new
Intent(MenuActivity.this, GenerarSolicitudesActivity.class);
        startActivity(intent);
    }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu, menu);
        return true;
    }
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.salir) {
            irLoguin();
            return true;
        }
        return super.onOptionsItemSelected(item);
    }
    private void irLoguin() {
        Intent i = new Intent(MenuActivity.this, LoguinActivity.class);
        startActivity(i);
    }
}
```



Figura 7. Pantalla del módulo solicitud.



### c. Codificación de la storycard solicitud.

```
public class SolicitarAmbulanciaActivity extends AppCompatActivity {

    Location locationA = new Location("UbicacionUsuario");
    locationA.setLatitude(Double.parseDouble(latitud_usuario));
    locationA.setLongitude(Double.parseDouble(longitud_usuario));
    int tipo = Integer.parseInt(codigo_tipo_emergencia);

    //Definimos variables
    float distancia_menor = 0;
    Ambulancia ambulancia = new Ambulancia();

    int primera_ambulancia = 0;

    //Recorremos la lista de ambulancias
    for (int i = 0; i < list_ambulancia.size(); i++) {

        //Verificamos el tipo de emergencia de la ambulancia
        if (list_ambulancia.get(i).getTipo_emergencia() == tipo) {

            //Obtenemos la ubicación de cada ambulancia
            Location locationB = new Location(list_ambulancia.get(i).getDescripcion());

            locationB.setLatitude(Double.parseDouble(list_ambulancia.get(i).getLatitud()));
            locationB.setLongitude(Double.parseDouble(list_ambulancia.get(i).getLongitud()));

            //Obtenemos la distancia entre el usuario y la ambulancia
            float distancia_actual = locationA.distanceTo(locationB);

            //Si es la primera ambulancia, asumimos que es la más cercana y guardamos sus
datos
            if (primera_ambulancia == 0) {

                distancia_menor = distancia_actual;

            }

            ambulancia.setCodigo_ambulancia(list_ambulancia.get(i).getCodigo_ambulancia());
            ambulancia.setDescripcion(list_ambulancia.get(i).getDescripcion());
            ambulancia.setTipo_emergencia(list_ambulancia.get(i).getTipo_emergencia());
            ambulancia.setLatitud(list_ambulancia.get(i).getLatitud());
            ambulancia.setLongitud(list_ambulancia.get(i).getLongitud());
            ambulancia.setEstado_activo(list_ambulancia.get(i).isEstado_activo());
            primera_ambulancia++;
        }
    }
}
```

```

//Si es la segunda ambulancia, verificamos si la distancia es menor a la anterior
else {
    //Verificamos si la distancia es menor a la ambulancia anterior
    if (distancia_actual < distancia_menor) {
        //Si la distancia de la ambulancia actual es menor a la anterior, guardamos sus datos
        distancia_menor = distancia_actual;

ambulancia.setCodigo_ambulancia(list_ambulancia.get(i).getCodigo_ambulancia());
ambulancia.setDescripcion(list_ambulancia.get(i).getDescripcion());
ambulancia.setTipo_emergencia(list_ambulancia.get(i).getTipo_emergencia());
ambulancia.setLatitud(list_ambulancia.get(i).getLatitud());
ambulancia.setLongitud(list_ambulancia.get(i).getLongitud());
ambulancia.setEstado_activo(list_ambulancia.get(i).isEstado_activo());
    }
}
}
}

if (ambulancia.getCodigo_ambulancia() == 0) {

    pd = new SweetAlertDialog(SolicitarAmbulanciaActivity.this,
SweetAlertDialog.WARNING_TYPE);
    pd.getProgressHelper().setBarColor(Color.parseColor("#03A9F4"));
    pd.setContentText("No hay ninguna ambulancia disponible!!");
    pd.setCancelable(false);
    pd.show();
    return;

} else {

    pd = new SweetAlertDialog(SolicitarAmbulanciaActivity.this,
SweetAlertDialog.PROGRESS_TYPE);
    pd.getProgressHelper().setBarColor(Color.parseColor("#102670"));
    pd.setContentText("Por favor, espere...");
    pd.setCancelable(false);
    pd.show();

//Mostramos la ambulancia asignada
Log.d("jhuaroto", "Codigo: " + ambulancia.getCodigo_ambulancia());
Log.d("jhuaroto", "Descripcion: " + ambulancia.getDescripcion());
Log.d("jhuaroto", "Tipo: " + ambulancia.getTipo_emergencia());
Log.d("jhuaroto", "Lat: " + ambulancia.getLatitud());
Log.d("jhuaroto", "Long: " + ambulancia.getLongitud());
Log.d("jhuaroto", "Estado: " + ambulancia.isEstado_activo());

```

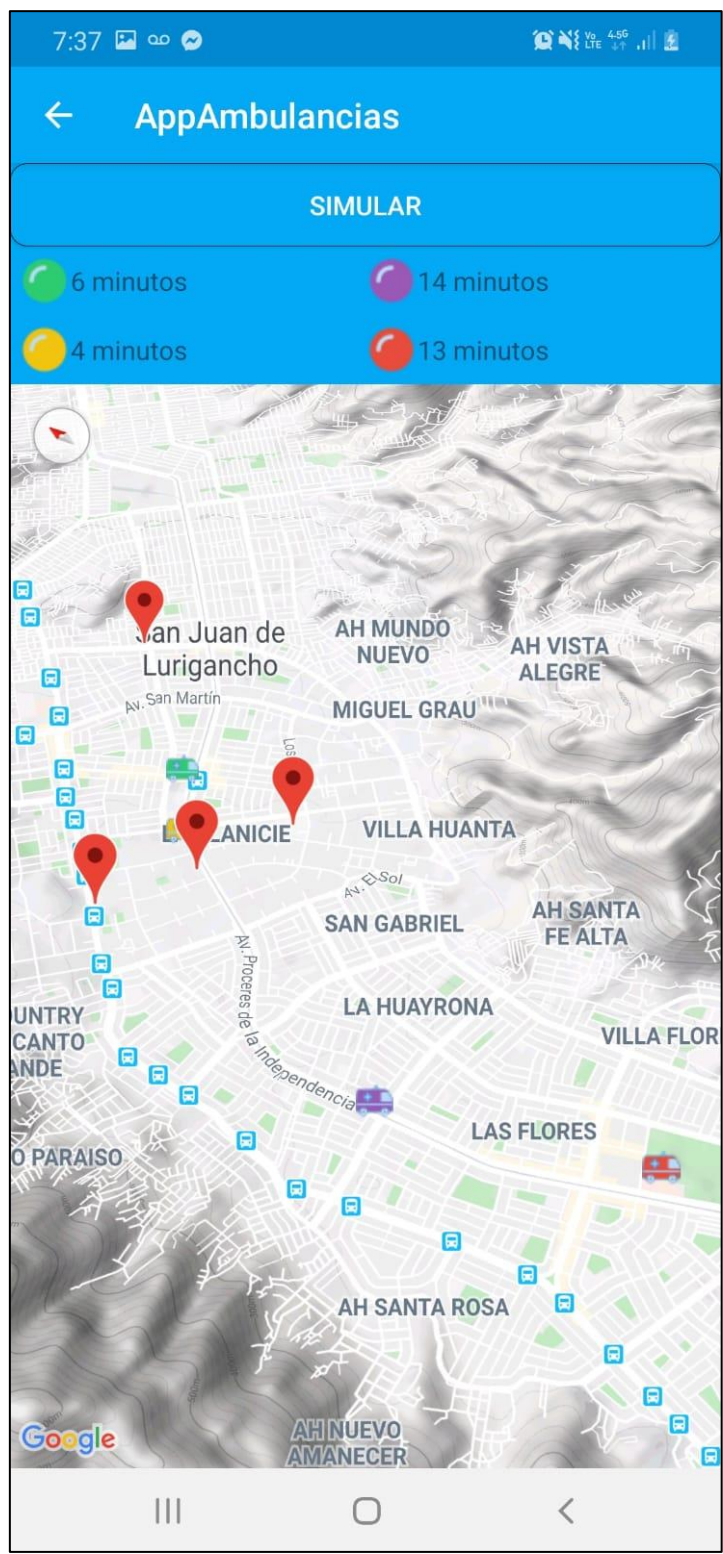


Figura 8. Pantalla del módulo simulación.

#### d. Codificación de la storycard simulación.

```
public class SeguimientoActivity extends AppCompatActivity implements
OnMapReadyCallback {

    private GoogleMap mMap;

    ArrayList<Ubicacion> list_ambulancia_uno = new ArrayList<>();
    ArrayList<Ubicacion> list_ambulancia_dos = new ArrayList<>();
    ArrayList<Ubicacion> list_ambulancia_tres = new ArrayList<>();
    ArrayList<Ubicacion> list_ambulancia_cuatro = new ArrayList<>();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_seguimiento);
        ButterKnife.bind(this);

        init();

        SupportMapFragment mapFragment = (SupportMapFragment)
        getSupportFragmentManager()
            .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);
    }

    //Ubicación de solicitud y ambulancia.

    LatLng solicitud1 = new LatLng(-11.9817158, -77.0001547);
    MarkerOptions marker1 = new MarkerOptions()
        .position(solicitud1)
        .draggable(true)
        .title("Solicitud1");
    mMap.addMarker(marker1);

    //Ubicacion de la ambulancia 01

    LatLng ambulancia1 = new LatLng(-11.975694, -77.002243);
    MarkerOptions markerambulancia1 = new MarkerOptions()
        .position(ambulancia1)
        .icon(iconVerde)
        .draggable(true)
        .title("Ambulancia1");
    mMap.addMarker(markerambulancia1);

    //Ubicacion de la solicitud 02

    LatLng solicitud2 = new LatLng(-11.966273, -76.99649);
    MarkerOptions marker2 = new MarkerOptions()
        .position(solicitud2)
        .draggable(true)
        .title("Solicitud2");
    mMap.addMarker(marker2);

    //Ubicacion de la ambulancia 02

    LatLng ambulancia2 = new LatLng(-11.997121, -77.010108);
    MarkerOptions markerambulancia2 = new MarkerOptions()
        .position(ambulancia2)
        .icon(iconMorado)
```

```

        .draggable(true)
        .title("Ambulancia2");
mMap.addMarker(markerambulancia2);

//Ubicacion de la solicitud 03

LatLng solicitud3 = new LatLng(-11.978724, -77.010768);
MarkerOptions marker3 = new MarkerOptions()
    .position(solicitud3)
    .draggable(true)
    .title("Solicitud3");
mMap.addMarker(marker3);

//Ubicacion de la ambulancia 03

LatLng ambulancia3 = new LatLng(-11.97885, -77.005066);
MarkerOptions markerambulancia3 = new MarkerOptions()
    .position(ambulancia3)
    .icon(iconAmarillo)
    .draggable(true)
    .title("Ambulancia3");
mMap.addMarker(markerambulancia3);

//Ubicacion de la solicitud 04

LatLng solicitud4 = new LatLng(-11.98053, -77.005675);
MarkerOptions marker4 = new MarkerOptions()
    .position(solicitud4)
    .draggable(true)
    .title("Solicitud4");
mMap.addMarker(marker4);

//Ubicacion de la ambulancia 04

LatLng ambulancia4 = new LatLng(-12.008397, -77.003677);
MarkerOptions markerambulancia4 = new MarkerOptions()
    .position(ambulancia4)
    .icon(iconRojo)
    .draggable(true)
    .title("Ambulancia4");
mMap.addMarker(markerambulancia4);

//Ubicacion de la ruta de la ambulancia01

Ubicacion ubicacion = new Ubicacion();
ubicacion.setLat(-11.976228);
ubicacion.setLog(-77.002563);
list_ambulancia_uno.add(ubicacion);

Ubicacion ubicacion2 = new Ubicacion();
ubicacion2.setLat(-11.977097);
ubicacion2.setLog(-77.003446);
list_ambulancia_uno.add(ubicacion2);

Ubicacion ubicacion3 = new Ubicacion();
ubicacion3.setLat(-11.978023);
ubicacion3.setLog(-77.004003);
list_ambulancia_uno.add(ubicacion3);

Ubicacion ubicacion4 = new Ubicacion();
ubicacion4.setLat(-11.979418);

```

```
ubicacion4.setLog(-77.002251);
list_ambulancia_uno.add(ubicacion4);

Ubicacion ubicacion5 = new Ubicacion();
ubicacion5.setLat(-11.98171);
ubicacion5.setLog(-77.0001547);
list_ambulancia_uno.add(ubicacion5);

Ubicacion ubicacion6 = new Ubicacion();
ubicacion6.setLat(-11.9817158);
ubicacion6.setLog(-77.0001547);
list_ambulancia_uno.add(ubicacion6);
```

*//Ubicacion de la ruta de la ambulancia02*

```
Ubicacion ubicacion7 = new Ubicacion();
ubicacion7.setLat(-11.996919);
ubicacion7.setLog(-77.009494);
list_ambulancia_dos.add(ubicacion7);
```

```
Ubicacion ubicacion8 = new Ubicacion();
ubicacion8.setLat(-11.99344);
ubicacion8.setLog(-77.010534);
list_ambulancia_dos.add(ubicacion8);
```

```
Ubicacion ubicacion9 = new Ubicacion();
ubicacion9.setLat(-11.989991);
ubicacion9.setLog(-77.010538);
list_ambulancia_dos.add(ubicacion9);
```

```
Ubicacion ubicacion10 = new Ubicacion();
ubicacion10.setLat(-11.984952);
ubicacion10.setLog(-77.006627);
list_ambulancia_dos.add(ubicacion10);
```

```
Ubicacion ubicacion11 = new Ubicacion();
ubicacion11.setLat(-11.978147);
ubicacion11.setLog(-77.003846);
list_ambulancia_dos.add(ubicacion11);
```

```
Ubicacion ubicacion12 = new Ubicacion();
ubicacion12.setLat(-11.966273);
ubicacion12.setLog(-76.99649);
list_ambulancia_dos.add(ubicacion12);
```

*//Ubicacion de la ruta de la ambulancia03*

```
Ubicacion ubicacion13 = new Ubicacion();
ubicacion13.setLat(-11.979072);
ubicacion13.setLog(-77.005899);
list_ambulancia_tres.add(ubicacion13);
```

```
Ubicacion ubicacion14 = new Ubicacion();
ubicacion14.setLat(-11.97682);
ubicacion14.setLog(-77.009907);
list_ambulancia_tres.add(ubicacion14);
```

```
Ubicacion ubicacion15 = new Ubicacion();
ubicacion15.setLat(-11.978281);
```

```

ubicacion15.setLog(-77.010739);
list_ambulancia_tres.add(ubicacion15);

Ubicacion ubicacion16 = new Ubicacion();
ubicacion16.setLat(-11.977759);
ubicacion16.setLog(-77.01051);
list_ambulancia_tres.add(ubicacion16);

Ubicacion ubicacion17 = new Ubicacion();
ubicacion17.setLat(-11.978724);
ubicacion17.setLog(-77.010768);
list_ambulancia_tres.add(ubicacion17);

Ubicacion ubicacion18 = new Ubicacion();
ubicacion18.setLat(-11.978724);
ubicacion18.setLog(-77.010768);
list_ambulancia_tres.add(ubicacion18);

//Ubicacion de la ruta de la ambulancia04

Ubicacion ubicacion19 = new Ubicacion();
ubicacion19.setLat(-12.008499);
ubicacion19.setLog(-77.004162);
list_ambulancia_cuatro.add(ubicacion19);

Ubicacion ubicacion20 = new Ubicacion();
ubicacion20.setLat(-11.997258);
ubicacion20.setLog(-77.009411);
list_ambulancia_cuatro.add(ubicacion20);

Ubicacion ubicacion21 = new Ubicacion();
ubicacion21.setLat(-11.993357);
ubicacion21.setLog(-77.010572);
list_ambulancia_cuatro.add(ubicacion21);

Ubicacion ubicacion22 = new Ubicacion();
ubicacion22.setLat(-11.985004);
ubicacion22.setLog(-77.006556);
list_ambulancia_cuatro.add(ubicacion22);

Ubicacion ubicacion23 = new Ubicacion();
ubicacion23.setLat(-11.98064);
ubicacion23.setLog(-77.005351);
list_ambulancia_cuatro.add(ubicacion23);

Ubicacion ubicacion24 = new Ubicacion();
ubicacion24.setLat(-11.98053);
ubicacion24.setLog(-77.005675);
list_ambulancia_cuatro.add(ubicacion24);

}

```



Figura 9: Pantalla del módulo generación de solicitudes.



## e. Codificación de la storycard generar solicitudes.

```
public class GenerarSolicitudesActivity extends AppCompatActivity {

    @BindView(R.id.edt_numero_solicitudes)
    EditText edt_numero_solicitudes;

    @BindView(R.id.contenedor)
    LinearLayout contenedor;

    @BindView(R.id.tv_trd)
    TextView tv_trd;

    List<Ubicacion> list_ubicacion_solicitante = new ArrayList<>();
    List<Ambulancia> list_ambulancia = new ArrayList<>();
    List<TipoEmergencia> list_tipo_emergencia = new ArrayList<>();

    SweetAlertDialog pd;
    int numero_inicial;
    int numero;
    int position;
    int position2;

    int tiempo_total = 0;
    int numero_solicitudes = 0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_generar_solicitudes);
        ButterKnife.bind(this);
        callApiRest();
    }

    private void callApiRest() {

        pd = new SweetAlertDialog(GenerarSolicitudesActivity.this,
            SweetAlertDialog.PROGRESS_TYPE);

        pd.getProgressHelper().setBarColor(Color.parseColor("#102670"));
        pd.setContentText("Por favor, espere...");
        pd.setCancelable(false);
        pd.show();

        MethodWs methodWs2 =
            HelperWs.getConfiguration().create(MethodWs.class);
        Call<List<Ambulancia>> responseBodyCall =
            methodWs2.obtenerAmbulancias();
        responseBodyCall.enqueue(new Callback<List<Ambulancia>>() {
            @Override
            public void onResponse(Call<List<Ambulancia>> call,
                Response<List<Ambulancia>> response) {

                if (response.isSuccessful()) {

                    list_ambulancia = response.body();
                    pd.dismiss();
                }
            }
        })
    }
}
```

```

        @Override
        public void onFailure(Call<List<Ambulancia>> call, Throwable t)
        {

                pd.dismiss();
                pd = new
SweetAlertDialog(GenerarSolicitudesActivity.this,
SweetAlertDialog.WARNING_TYPE);

pd.getProgressHelper().setBarColor(Color.parseColor("#03A9F4"));
                pd.setContentText(t.getMessage().toString());
                pd.setCancelable(false);
                pd.show();
                return;
        }
    });

    MethodWs methodWs =
HelperWs.getConfiguration().create(MethodWs.class);
    Call<List<TipoEmergencia>> responseBodyCall2 =
methodWs.obtenerTipoEmergencias();
    responseBodyCall2.enqueue(new Callback<List<TipoEmergencia>>()
    {

        @Override
        public void onResponse(Call<List<TipoEmergencia>> call,
Response<List<TipoEmergencia>> response) {

                if (response.isSuccessful()) {

                        list_tipo_emergencia = response.body();

                                numero_inicial =
Integer.parseInt(list_tipo_emergencia.get(0).getCodigo_emergencia().to
String());

                                        pd.dismiss();
                                }
                }

        @Override
        public void onFailure(Call<List<TipoEmergencia>> call,
Throwable t) {

                pd.dismiss();
                pd = new
SweetAlertDialog(GenerarSolicitudesActivity.this,
SweetAlertDialog.WARNING_TYPE);

pd.getProgressHelper().setBarColor(Color.parseColor("#03A9F4"));
                pd.setContentText(t.getMessage().toString());
                pd.setCancelable(false);
                pd.show();
                return;
        }
    });
}
@OnClick(R.id.btn_generar_tdr)
public void generarTdr() {

    DecimalFormat df = new DecimalFormat("#.00");

```

```

        double total = (double) tiempo_total / numero_solicitudes;

        tv_trd.setText("" + df.format(total));
    }

    @OnClick(R.id.btn_generar_solicitudes)
    public void generarSolicitudes() {

        int numero_solicitudes =
Integer.parseInt(edt_numero_solicitudes.getText().toString());

        if (numero_solicitudes <= 0) {

            pd = new SweetAlertDialog(GenerarSolicitudesActivity.this,
SweetAlertDialog.WARNING_TYPE);

pd.getProgressHelper().setBarColor(Color.parseColor("#03A9F4"));
            pd.setContentText("Debe ingresar un numero mayor a 0");
            pd.setCancelable(false);
            pd.show();

            return;
        }

        for (int i = 0; i < numero_solicitudes; i++) {

            getLocation(-11.966324, -77.003628, 300);
        }

Random aleatorio1 = new Random();
int tiempo_solicitud = tiempo_minimo +
aleatorio1.nextInt((tiempo_maximo + 1) - tiempo_minimo);

for (int i = 0; i < 200; i++) {

    Random aleatorio2 = new Random();
    int tiempo_solicitud2 = tiempo_minimo +
aleatorio2.nextInt((tiempo_maximo + 1) - tiempo_minimo);
    Log.i("jhuaroto", "" + tiempo_solicitud2);
}

tiempo_total = tiempo_total + tiempo_solicitud;
numero_solicitudes = numero_solicitudes + 1;

RegistrarSolicitudRequest registrarSolicitudRequest = new
RegistrarSolicitudRequest();
registrarSolicitudRequest.setCodigo_usuario(3);
registrarSolicitudRequest.setCodigo_ambulancia(ambulancia.getCodigo_amb
ambulancia());
registrarSolicitudRequest.setLatitud(String.valueOf(ambulancia.getLati
titud()));
registrarSolicitudRequest.setLongitud(String.valueOf(ambulancia.getLon
gitud()));
registrarSolicitudRequest.setTipo_emergencia(numero);
registrarSolicitudRequest.setTiempo(tiempo_solicitud);

```

#### IV. FASE IV: Estabilización.

En esta fase se logra implementar el correcto funcionamiento de la aplicación, además se asegura la calidad y se comprueba que tenga las características necesarias para cumplir con los objetivos esperados, como la interfaz de usuario, la arquitectura de software y diseño apropiado.

Tabla 36. Recomendaciones para el equipo móvil.

Equipo movil	
Hardware	<ul style="list-style-type: none"><li>* Memoria 4 GB.</li><li>* Exynos 9610 de ocho núcleos.</li><li>* Pantalla: 6.4 pulgadas (Super AMOLED).</li><li>* GPS, A-GPS, Glonass, Galileo.</li><li>* Conexión Wifi 802.11</li><li>* Memoria interna: 64 GB.</li></ul>
Software	<ul style="list-style-type: none"><li>*Sistema operativo: Andorid Pie 9</li><li>* Versión de Kernel: 4.14.62</li><li>* Red : 4.5 g</li></ul>

Tabla 37. Recomendaciones para la conexión al servidor.

Servidor	
Sql Server	Versión 2008
Equipo	<ul style="list-style-type: none"><li>*Laptop Acer Aspire 3 L745-SP4251LL</li><li>* Intel Core I5-2450 2.50 GHz</li><li>* Disco duro: 500 GB</li></ul>
SERVICIO	Hosting SmarterASP.Net

## V. FASE V: Pruebas.

Tabla 38. Prueba unitaria 1- Módulo de identificación de usuario

CODIGO	NOMBRE
M001	Módulo de identificación del usuario.
OBJETIVO	Validar el acceso solo a pacientes y el administrador de la aplicación móvil, validando id de usuario y contraseña.
PASOS	<ul style="list-style-type: none"> <li>• Seleccionar aplicativo para su inicialización.</li> <li>• Cuando aparezca la pantalla de usuario ingresar el primer campo el id del usuario.</li> <li>• Ingresar en el segundo campo la contraseña asignada.</li> </ul>
RESULTADO OBTENIDO	Al ingresar correctamente los datos solicitados, se muestra un mensaje "por favor espere" en el cual se estará validando a través del servicio y la base datos los usuarios anteriormente registrados, apareciendo el menú principal. Caso contrario el mensaje: "Las credenciales son incorrectas o usuario no está habilitado".
EVALUACIÓN	Aprobado

Tabla 39. Prueba unitaria 2 - Módulo menú principal

CODIGO	NOMBRE
M002	Módulo menú principal.
OBJETIVO	Permite elegir la opción o el módulo que desea utilizar el usuario o administrador de la aplicación móvil.
PASOS	<p>*Después de colocar correctamente los datos de usuario aparece la pantalla o menú principal.</p> <p>*Te brinda la opción de poder elegir los siguientes módulos: solicitud, simulación y generación de solicitudes.</p>
RESULTADO OBTENIDO	El módulo muestra lo solicitado.
EVALUACIÓN	Aprobado

Tabla 40. Prueba unitaria 3 - Módulo solicitud de ambulancia

CODIGO	NOMBRE
M003	Módulo de Solicitud.
OBJETIVO	Solicitar ambulancias seleccionando el tipo de emergencia requerido y a través de un botón de pánico para poder enviar la latitud y longitud del usuario solicitante.
PASOS	<ul style="list-style-type: none"> <li>• Selecciona el módulo solicitud.</li> <li>• Le aparecerá un mensaje mientras carga el módulo: "Por favor, espere".</li> <li>• Se abrirá una ventana en la cual se solicitará la ambulancia seleccionando el tipo o nivel de emergencia y un botón de pánico para mandar sus latitudes y longitudes actualizadas.</li> </ul>
RESULTADO OBTENIDO	*Se le asigna la ambulancia cumpliendo la lógica interna del nivel de emergencia y seleccionando la ambulancia más cercana al punto del solicitante.
EVALUACIÓN	Aprobado

Tabla 41. Prueba unitaria 4 - Módulo simulación

CODIGO	NOMBRE
M004	Módulo Simulación
OBJETIVO	Implementar la Api de Google Maps en este módulo para simular las solicitudes procesadas y el trayecto de las ambulancias hacia el punto de solicitud.
PASOS	<ul style="list-style-type: none"> <li>• Seleccionar modulo simulación.</li> <li>• Seleccionamos el botón Simular, para que inicie la simulación con las solicitudes anteriormente registrados.</li> </ul>
RESULTADO OBTENIDO	<p>Al ingresar al módulo simulación se visualizan los puntos de solicitudes de emergencia y ambulancias activas en el mapa.</p> <p>Al seleccionar el botón simular se valida que las ambulancias recorren el trayecto para llegar al punto de solicitud. También se visualiza que para cada viaje o recorrido de las ambulancias les asigna un tiempo estimado de llegada al punto de solicitud.</p>
EVALUACIÓN	Aprobado

Tabla 42. Prueba unitaria 5 - Módulo generación de solicitudes

CODIGO	NOMBRE
M004	Módulo generación de solicitudes.
OBJETIVO	<p>Generar solicitudes masivas aleatoriamente entre niveles de emergencia y la ubicación del paciente. Determinar el Tiempo de respuesta y cantidad de solicitudes procesadas ante diferentes iteraciones realizadas de 50, 75 y 100 solicitudes.</p>
PASOS	<ul style="list-style-type: none"> <li>• Seleccionar modulo generación de solicitudes.</li> <li>• Ingresar el número de solicitudes.</li> <li>• Seleccionar el botón generar para que inicie la actividad.</li> <li>• Seleccionar el botón calcular para que nos brinde el tiempo de respuesta promedio de las solicitudes ingresadas.</li> </ul>
RESULTADO OBTENIDO	<p>Se valida que al ingresar el número de solicitudes y seleccionar el botón generar se carga correctamente el número de solicitudes que se generan de manera aleatoria entre niveles de emergencia, latitudes y longitudes. Teniendo como resultado de esta tabulación el tiempo de respuesta promedio para la llegada de la ambulancia al punto de solicitud.</p>
EVALUACIÓN	Aprobado