SISTEMA DE COMUNICACIÓN Y MONITOREO PARA GRUPOS DE MOTOVIAJEROS, BASADOS EN GPS Y MENSAJERÍA MÓVIL

Oscar Fernando Espinosa Rocha
Yeison Camilo Rincón Ordoñez

Universidad Distrital Francisco José de Caldas
Facultad Tecnológica
Ingeniería en Telemática
Bogotá, Colombia
2022

SISTEMA DE COMUNICACIÓN Y MONITOREO PARA GRUPOS DE MOTOVIAJEROS, BASADOS EN GPS Y MENSAJERÍA MÓVIL

Oscar Fernando Espinosa Rocha Cód.: 2018267015
Yeison Camilo Rincón Ordoñez Cód.: 20182678006

Proyecto de grado presentado como requisito parcial para optar al título de:

Ingeniero en Telemática

Director (a):
Mgs. Ismael Antonio Ardila Sánchez

Universidad Distrital Francisco José de Caldas
Facultad Tecnológica
Ingeniería en Telemática
Bogotá, Colombia
2022

Nota de Aceptación

Tutor del proyecto
Mgs. Ismael Antonio Ardila Sánchez

Firma del jurado

Bogotá D.C, septiembre 2022

AGRADECIMIENTOS

Queremos dar un agradecimiento especial a:

- Todos los docentes de la facultad tecnológica de la Universidad Distrital Francisco José de Caldas, pues fueron ellos quienes nos prepararon durante todo este tiempo para poder llegar a este momento.

- Nuestras familias, pues día tras día fueron un inmenso apoyo y motivación para seguir adelante con las metas que nos propusimos.

- Al ingeniero Ismael Ardila, por su gran ayuda y guía durante todo este proceso, dedicándonos tiempo y esfuerzo; infinitas gracias a su asesoría, interés y compromiso demostrados.

- Todos aquellos compañeros y amigos quienes nos compartieron su tiempo, consejos y sabiduría durante la realización de este proyecto y también durante todo el proceso educativo.

TABLA DE CONTENIDO

TABLA DE CONTENIDO **12**
TABLA DE ILUSTRACIONES **14**
ILUSTRACIÓN DE TABLAS **15**
1. RESUMEN **1**
2. ABSTRACT **3**
3. PROYECTO **5**
3.1 PLANTEAMIENTO DEL PROBLEMA 5
4. OBJETIVOS **6**
4.1 OBJETIVO GENERAL 6
4.2 OBJETIVOS ESPECÍFICOS 6
5. ALCANCES **7**
6. DELIMITACIONES **8**
7. JUSTIFICACIÓN **9**
8. MARCO DE REFERENCIA **10**
8.1 ESTADO DEL ARTE 10
8.2 MARCO CONCEPTUAL 13
9. METODOLOGÍA **22**
9.1 METODOLOGÍA DE INVESTIGACIÓN 22
9.2 METODOLOGÍA DE PROYECTO 27
10. FACTIBILIDAD **28**
10.1 FACTIBILIDAD ECONÓMICA 28
11. CRONOGRAMA **31**
12. DESARROLLO PROYECTO **32**
12.1 ARQUITECTURA DE LA INFRAESTRUCTURA 32
12.1.1 BASE DE DATOS 33
12.1.2 SERVIDOR DE APLICACIONES 34
12.1.3 ARQUITECTURA CLOUD AWS 35
12.2 DESARROLLO DEL SOFTWARE 38
12.2.1 ARQUITECTURA DE SOFTWARE 38
12.2.2 API MOTO VIAJE 38
12.2.3 MANUAL DE INSTALACIÓN 38
12.3 PROBLEMAS ENCONTRADOS 38

12.3.1 DESARROLLO DEL PROYECTO 38
12.3.2 USO DE LA APLICACIÓN 39
13. RESULTADOS OBTENIDOS 40
13.1 PRUEBAS APLICACIÓN 40
13.1.1 REGISTRO NUEVO USUARIO E INICIO DE SESIÓN: 40
13.1.2 CREACIÓN DE GRUPOS 41
13.1.3 CREAR VIAJE Y ASOCIAR MIEMBROS AL VIAJE 43
13.1.4 VIAJE 44
13.2 ANÁLISIS DE LOS RESULTADOS 46
14. IMPACTOS 47
15. TRABAJO FUTURO 48
16. CONCLUSIONES 49
17. BIBLIOGRAFIA 50

TABLA DE ILUSTRACIONES

Ilustración 1: Encuesta de viabilidad.....23
Ilustración 2: Resultados pregunta 1.....24
Ilustración 3: Resultados pregunta 2.....24
Ilustración 4: Resultados pregunta 3.....24
Ilustración 5: Resultados pregunta 4.....25
Ilustración 6: Resultados pregunta 5.....25
Ilustración 7: Resultados pregunta 6.....25
Ilustración 8: Resultados pregunta 7.....26
Ilustración 9: Resultados pregunta 8.....26
Ilustración 10: Cronograma.....31
Ilustración 11: Actividades 31
Ilustración 12: Diagrama arquitectura solución 33
Ilustración 13: Reglas de firewall IPV4.....35
Ilustración 14: Reglas de firewall IPV6.....35
Ilustración 15: Diagrama arquitectura CLOUD AWS.....36
Ilustración 16: Registro de usuario.....40
Ilustración 17: Inicio de sesión.....41
Ilustración 18: Creación de grupo 42
Ilustración 19: Asociar miembro al grupo.....42
Ilustración 20: Creación nuevo viaje 43
Ilustración 21: Asociar viajero 43
Ilustración 22: Iniciar viaje.....44
Ilustración 23: Envío de notificaciones.....45
Ilustración 24: Envío de alerta y terminación del viaje 45

ILUSTRACIÓN DE TABLAS

Tabla 1: Características metodología investigativa.....22
Tabla 2: Factibilidad de Planeación 28
Tabla 3: Factibilidad Económica Desarrollo.....30
Tabla 4: Otros Gastos.....30
Tabla 5: Total Gastos.....30

1. RESUMEN

Basados en el comportamiento actual de las personas, que día a día usan más su vehículo (motocicleta) para recorrer largos trayectos solo con el fin de descubrir nuevos lugares y enfocados en los individuos que ejecutan estas actividades de forma grupal, se pensó en una solución a un problema de gran impacto a la hora de recorrer kilómetros con uno o varios compañeros.

Se encontró que, en muchas ocasiones, por motivos varios, se pierde el contacto con uno o más integrantes del grupo, dificultando la notificación y/o comunicación entre estos en caso de que se presente algún accidente o imprevisto que requiera de asistencia inmediata y oportuna, tanto entre los mismos miembros del grupo o con los servicios de emergencia y asistencia en carretera.

Con miras a ofrecer una propuesta que pueda ser de utilidad a dichos usuarios, se propone una solución que permita monitorear la ubicación en tiempo real de los compañeros del grupo que realizan la actividad usando el sistema de posicionamiento global de sus equipos móviles, facilitar la comunicación entre los integrantes del grupo y los servicios de emergencia y asistencia en carretera, generar alertas a los demás integrantes del grupo en caso de que necesite ayuda o asistencia ya sea porque un usuario solicita dicha asistencia o porque se detecten eventos que puedan indicar algún accidente, hurto o pérdida de señal, entre otros, e informar a los participantes de la actividad sobre alguna novedad, evento o parada que requiera la atención del grupo.

En una revisión inicial del mercado de aplicaciones disponibles, se descubrió que ninguna de las soluciones actuales posee todas estas características; localizando así un posible nicho de mercado que permita ofrecer este producto a un grupo específico de individuos que previamente han sido censados con el fin de obtener los datos necesarios para la construcción de nuestra aplicación.

Al introducir la idea en el mercado, nuestra proyección es muy positiva, siendo posibles usuarios de la misma aplicación, concebimos de forma muy entusiasta la distribución y el alcance de esta, cumpliendo con objetivos específicos como comunicar y permitir que los usuarios estén seguros en la ejecución de las actividades propias de transportarse en su vehículo privado.

Teniendo como pilar la protección de la vida humana, nuestra propuesta busca reducir al mínimo los posibles accidentes o errores que se puedan producir por falta de información o comunicación en una salida o un viaje largo; permitiendo,

si es necesario, monitorear el avance y el estado del viajero si éste así lo desea. O, si en caso de que se presente alguna eventualidad, poder reducir los tiempos de notificación y atención pudiendo así mitigar las posibles consecuencias negativas.

2. ABSTRACT

Based on the current behavior of people, who everyday use their vehicle (motorcycle) more and more to travel long distances just to discover new places and focusing on individuals who perform these activities in groups, we thought of a solution to a problem of great impact when traveling kilometers with one or more companions.

It was found that on many occasions, for various reasons, contact with one or more members of the group is lost, making it difficult to notify and/or communicate with each other in case of an accident or unforeseen event that requires immediate and timely assistance, either among the group members themselves or with emergency services and roadside assistance.

In order to offer a proposal that could be useful to these users, we propose a solution that allows monitoring the location in real time of the group members who are carrying out the activity using the global positioning system of their mobile devices, facilitating the communication between the group members and the emergency and roadside assistance services, generating alerts to the other members of the group in case help or assistance is needed, either because a user requests such assistance or because events are detected that may indicate an accident, theft or loss of signal, among others, and informing the participants of the activity about any news, event or stop that requires the attention of the group.

In an initial review of the market of available applications, it was discovered that none of the current solutions have all these features; thus, locating a possible market niche that allows us to offer this product to a specific group of individuals who have been previously surveyed in order to obtain the necessary data for the construction of our application.

At the time of introducing the idea in the market, our projection is very positive, being ourselves possible users of the same application. We enthusiastically conceived the distribution and scope of the application, fulfilling specific objectives such as communicating and allowing users to be safe in the execution of the activities of transporting themselves in their private vehicle.

Having as a pillar the protection of human life, our proposal seeks to minimize possible accidents or errors that may occur due to lack of information or communication in a departure or a long trip; allowing, if necessary, to monitor the progress and status of the traveler if he so wishes. Or, if any eventuality

should arise, to be able to reduce notification and attention times, thus being able to mitigate possible negative consequences.

3. PROYECTO

3.1 PLANTEAMIENTO DEL PROBLEMA

Los viajes en moto, también conocidos como moto turismo, han ganado mucha popularidad en los últimos años puesto que es más fácil y económico realizar viajes por carretera que con otros medios de transporte.

Recorrer largas distancias, es una de las cosas que más desmotiva a los viajeros para realizar los recorridos por carretera en carros o buses intermunicipales debido a los largos tiempos de viaje, también los embotellamientos que se puedan presentar, los costos de combustible y peajes son factores por los que se prefieren los viajes en moto.

Pero junto con estas ventajas frente a los otros medios de transporte también vienen algunas desventajas, tales como:

- Mayor tiempo de atención de posibles lesiones de gravedad en caso de accidentes de tránsito.
- Cuando se viaja en grupo es difícil mantener un canal de comunicación directo y efectivo entre los viajeros, porque al realizarse una comunicación directa se puede convertir en un factor de riesgo adicional.

En la actualidad no existe una solución tecnológica para poder mitigar estos problemas de una manera centralizada. Aunque existen herramientas de comunicación, tales como llamadas telefónicas o mensajería instantánea, estas, dependen del usuario para poder ser iniciadas, pero en el caso de un accidente de tránsito, en donde las condiciones no permitan que el moto viajero accidentado pueda avisarle a sus contactos u otros miembros del grupo de viaje, son un factor muy importante en cuanto a las secuelas que pueda dejar el accidente e incluso puede tener un desenlace fatal.

Junto a esta carencia en la notificación a otros viajeros cuando se viaja en grupo, el poder tener información sobre la ubicación de cada integrante es muy importante para verificar que todos siguen la ruta planeada y, si se presenta alguna eventualidad, poder tomar decisiones que tengan un menor impacto en el recorrido por el camino.

4. OBJETIVOS

4.1 OBJETIVO GENERAL

Desarrollar una aplicación para teléfonos inteligentes que permita la comunicación, monitoreo y notificación de eventualidades para grupos de motociclistas en tiempo real.

4.2 OBJETIVOS ESPECÍFICOS

1. Desarrollar una solución tecnológica que alerte de problemas en los viajes de motos, como robos, accidentes o fallos mecánicos.
2. Desarrollar un módulo que permita el acceso a la aplicación y las suscripciones a los grupos de viaje.
3. Implementar un sistema de notificación a los miembros del grupo, en caso de una eventualidad o emergencia.
4. Crear un módulo que permita hacer seguimiento a los miembros del grupo usando sistemas de mapas, posicionamiento global y el GPS de los dispositivos.
5. Utilizar un modelo de arquitectura distribuida en la nube, con el fin de garantizar la accesibilidad, la escalabilidad, la seguridad de los datos, la disponibilidad de los servicios y el almacenamiento de los datos.

5. ALCANCES

El proyecto desarrollará el prototipo de aplicación móvil para teléfonos inteligentes, que cuente con sensor de GPS y con conexión a internet de forma constante mientras se hace uso de la aplicación.

La aplicación permitirá el registro de nuevos usuarios y la creación de grupos para poder realizar el seguimiento de los viajes.

Durante el viaje será posible visualizar la posición de los integrantes del grupo, además permitirá notificar a todo el grupo en caso de que se presente alguna eventualidad, ya sea porque un integrante desencadena la notificación o porque se detecta un posible accidente.

La aplicación tendrá una forma de identificar posibles accidentes al detectar que la posición del usuario no cambia en un tiempo determinado, que podrá ser configurable, y mostrará un mensaje que le informe al usuario que se enviará la notificación de emergencia al grupo y que puede cancelar el envío de dicha notificación en caso de que no sea necesario enviarla.

En caso de no activar el botón en un periodo de tiempo corto, que también es configurable, se procede con la notificación a los demás miembros del grupo
También se pueden mostrar notificaciones desencadenadas por los usuarios que pueden ser cuando se presenten fallas mecánicas, paradas no planeadas para descansar, entre otras.

Para usuarios que requieran una cobertura en lugares más remotos de poca o nula señal móvil de operadores comunes, se recomienda adquirir servicios como él de Sigfox, una empresa que provee conexión satelital para conexión global, permitiendo así estar conectados y generar alertas de ser necesario, brindando seguimiento continuo al integrante a cada kilómetro de su recorrido.

6. DELIMITACIONES

Esta aplicación está pensada únicamente para dispositivos móviles inteligentes que cuenten con sensor GPS y conexión a internet.

El grupo objetivo de la aplicación son los moto viajeros, los cuales tendrán disponible el sistema para registrarse como usuarios y podrán hacer seguimiento solo de los compañeros que también estén registrados en la aplicación y accedan a ser incluidos en los grupos de seguimiento.

7. JUSTIFICACIÓN

Los viajes en motocicleta son una actividad que cada vez va ganando más participantes, quienes los usan como una alternativa de recreación y para escapar de la rutina de la cotidianeidad de las ciudades.

Estos viajes suelen hacerse entre grupos de amigos y con destinos alejados de sus lugares de partida; esto conlleva que, en muchos casos, la comunicación entre los viajeros se dificulte debido a la necesidad de hacerlo por medio de llamadas telefónicas, mensajes de texto o mensajes por chat, generando así inconvenientes de comunicación cuando se trata de eventos de emergencia (como lo puede ser un accidente o una falla mecánica).

Actualmente no se disponen de soluciones de comunicación inmediatos que puedan ser confiables para poder hacer el seguimiento del estado de los participantes de los viajes y, mucho menos, de notificarles de cuando se presentan eventos inesperados.

Actualmente podemos encontrar aplicaciones y proyectos que buscan dar solución a esta problemática pero presentan carencias en cuanto a sus prestaciones, ya sea por la cobertura geográfica (que se encuentran limitados a alguna ciudad o región en específico), por su medio de comunicación (por medio de llamadas telefónicas) o por que requieren de que el usuario deba interactuar con la aplicación para poder comunicarse (mensajes de texto o chats), las cuales no son óptimas para la comunicación entre viajeros en motocicletas.

De acuerdo con lo anteriormente mencionado, se propone desarrollar un prototipo de aplicación móvil que reúna los servicios de monitoreo de posición por GPS, notificación de eventualidades y/o accidentes y visualización de la ruta de viaje; de esta forma se pretende ayudar a mejorar la experiencia de los moto viajeros.

8. MARCO DE REFERENCIA

8.1 ESTADO DEL ARTE

Para el desarrollo de este proyecto se realizó una investigación y se encontró una base en los siguientes proyectos académicos de diferentes universidades a nivel nacional e internacional:

- Sistema para la localización de motos mediante GSM Y GPS en Bogotá de Camilo Frontado Escobar, tesis de grado que plantea el prototipo de un sistema de ubicación de motocicletas para la ciudad de Bogotá.

  En este proyecto se nos muestra una manera de lograr la comunicación en tiempo real entre una motocicleta, haciendo uso de comunicación por mensajes de texto, con una plataforma web que muestra la localización del vehículo que puede ser consultada en cualquier momento.
  Este proyecto se enfoca en la localización de motocicletas robadas, ubicándose por medio de comunicación GSM entre la motocicleta y la plataforma web por medio de mensajes de texto hacia el dispositivo ubicado en la motocicleta.[1]

  Aportes al proyecto:
  Este proyecto permite dar a conocer una forma de comunicación a través de redes GSM, lo cual demuestra que, a pesar de ser un medio de comunicación efectivo, no es suficiente para poder satisfacer las necesidades del proyecto que se pretende desarrollar.

- Aplicación Móvil de Alerta de Motocicletas con Dispositivo GPS, de Sebastián Tracal Antilef, tesis de grado que desarrolla una aplicación móvil para detectar el movimiento no autorizado de una motocicleta.

  La aplicación tiene como fin alertar en caso de que la moto sea sustraída del lugar estacionado, mediante un GPS instalado en un lugar estratégico del vehículo la que enviará sus coordenadas cada 1 minuto a la base de datos y esta podrá ser visualizada en el teléfono móvil o Tablet.
  Este proyecto tiene como objetivo el desarrollo de un sistema que permita determinar cuándo una motocicleta haya sido posiblemente hurtada por

[1] FRONTADO ESCOBAR, Camilo Andres. Sistema para la localización de motos mediante Gsm Y Gps en Bogotá - hdl:11349/22420. Repositorio Institucional Universidad Distrital - RIUD [página web]. (18, septiembre, 2019). [Consultado el 2, septiembre, 2022]. Disponible en Internet: <https://repository.udistrital.edu.co/handle/11349/22420>.

medio de la detección del cambio en la ubicación del vehículo, haciendo la comparación en las coordenadas actuales con las de hace un minuto y alertando al propietario por medio de un mensaje de texto.[2]

Aportes al proyecto:
Nos da una idea de la forma en la que se puede registrar el movimiento del vehículo, lo cual es de suma importancia para poder cumplir con el objetivo del proyecto.

- Diseño de dispositivos de seguimiento GPS aplicado al transporte público de pasajeros de Miguel Vargas Romero, tesis de grado que desarrolla un prototipo de dispositivo para la captura y monitoreo de datos obtenidos a través de múltiples sensores ubicados dentro de un vehículo de transporte público.

  Con las exigencias realizadas a los vehículos y las funciones genéricas que poseen los aparatos de GPS se hace necesario el diseño de un aparato que pueda recolectar la mayor cantidad de información del vehículo de manera muy similar a las cajas negras que poseen los aviones.

  Este proyecto desarrolla un prototipo de dispositivo para el monitoreo GPS que almacena la información del día a día de un vehículo de transporte público.
  Los datos que se almacenan son los capturados por los sensores a bordo del bus (GPS, humedad, choque, apertura de puertas, entre otros), cámara de vídeo y que además cuentan con conectividad bluetooth para conectarse al módulo central.[3]

  Aportes al proyecto:
  Plantea una estrategia de captura y monitoreo de datos que nos sirve como base para poder realizar un monitoreo más acertado de las condiciones de los vehículos y sus pasajeros.

[2] TRACAL ANTILEF, Sebastián. Aplicación móvil de alerta de motocicletas con dispositivo GPS. Biblioteca Digital UDLA [página web]. (2017). [Consultado el 1, septiembre, 2021]. Disponible en Internet: <http://repositorio.udla.cl/xmlui/handle/udla/258>.

[3] VARGAS ROMERO, MIGUEL ÁNGEL. Diseño de dispositivo de seguimiento gps aplicado al transporte público de pasajeros. Universidad Santo Tomás [página web]. (2020). [Consultado el 1, septiembre, 2021]. Disponible en Internet: <https://repository.usta.edu.co/handle/11634/28840>.

- Sistema de geolocalización y análisis vehicular para motociclistas, es un artículo escrito por David Garzón Ramírez y Michael Sanabria Guio para la revista VISIÓN ELECTRÓNICA.

  Este artículo propone un sistema de análisis vehicular en donde los motociclistas tienen acceso a una aplicación desarrollada para dispositivos con sistema operativo Android que les mostrará una metodología de orientación sin depender exclusivamente de una conexión a internet.[4]

  Aportes al proyecto:

  Aunque el artículo se basa en el funcionamiento de un microcontrolador que captura los datos de los medidores de la motocicleta, nos es de ayuda para la forma de capturar dichos datos, solo que en el proyecto a desarrollar serán capturados (los que apliquen) haciendo uso de un teléfono inteligente con sistema operativo Android.

- Desarrollo de un Sistema Web y Aplicativo Android Para el Control y Monitoreo por Geolocalización de Unidades de Transportación de Carga Pesada para la Empresa Transporte y Logística Honores Méndez, tesis de grado de Richard Paolo Asencio Vargas en el cual se implementa un sistema para obtener la ubicación geo referencial de los vehículos de carga pesada para la empresa de transporte y logística Honores Méndez (Guayaquil, Ecuador).

  El mencionado sistema está compuesto por dos plataformas, una web y otra móvil, la plataforma web nos permite registrar y consultar todos los procesos logísticos para la transportación de carga pesada mientras que el módulo de sistema móvil nos muestra la ubicación de la unidad. [5]

  Aportes al proyecto:

---

[4] DAVID, Garzón Ramírez y MICHAEL, Sanabria Guio. Sistema de geolocalización y análisis vehicular para motociclistas - hdl:11349/23570. Repositorio Institucional Universidad Distrital - RIUD [página web]. (2020). [Consultado el 1, septiembre, 2021]. Disponible en Internet: <https://repository.udistrital.edu.co/handle/11349/23570>.

[5] ASENCIO VARGAS, Richard Paolo. Repositorio Universidad de Guayaquil: Desarrollo de un Sistema Web y Aplicativo Android Para el Control y Monitoreo por Geolocalización de Unidades de Transportación de Carga Pesada para la Empresa Transporte y Logística Honores Méndez. Repositorio Universidad de Guayaquil: Página de inicio [página web]. (2017). [Consultado el 1, septiembre, 2021]. Disponible en Internet: <http://repositorio.ug.edu.ec/handle/redug/19496>.

Este proyecto nos aporta una forma de poder llevar el control y seguimiento de varios vehículos en tiempo real y la manera de visualizar su ubicación en un servicio de mapas (Google Maps).

8.2 MARCO CONCEPTUAL

8.2.1 GSM

"GSM "Global System For Mobile Comunication" es un Sistema Global para las Comunicaciones Móviles que incorporan tecnología digital. Por ser digital cualquier cliente de GSM puede conectarse a través de su teléfono con su ordenador y puede hacer, enviar y recibir mensajes por e-mail, faxes, navegar por Internet, acceso seguro a la red informática de una compañía (LAN/Intranet), así como utilizar otras funciones digitales de transmisión de datos, incluyendo el Servicio de Mensajes Cortos (SMS)."[6]

- Comunicación móvil, uso de redes de transmisión celular

"El Reglamento de Radiocomunicaciones de la UIT (Unión Internacional de Tele- comunicaciones) define el servicio móvil como un servicio de radiocomunica- ciones entre estaciones móviles y estaciones fijas, o entre estaciones móviles únicamente."[7]

- Recolección, gestión y almacenamiento de datos.
- Desarrollo de aplicaciones móviles, sistemas de comunicación y transmisión de datos.

Es un conjunto de procesos y procedimientos que involucran la escritura de software para dispositivos inteligentes inalámbricos, como celulares o tabletas. Es muy similar al desarrollo de aplicaciones web, con la diferencia de que el desarrollo móvil se centra en el aprovechamiento de los recursos y características únicas ofrecidas por los dispositivos móviles, GPS, acelerómetros, sensores y otros.

---

[6] SAMSUNG LATINOAMÉRICA. ¿Qué es GSM? | Samsung Latinoamérica. Samsung Latinoamérica [página web]. (2, septiembre, 2022). [Consultado el 2, octubre, 2022]. Disponible en Internet: <https://www.samsung.com/latin/support/mobile-devices/what-is-gsm>.

[7] JOSE MARIA, HERNANDO RÁBANOS; LUIS, MENDO TOMÁS y JOSE MANUEL, RIERA SALÏS. COMUNICACIONES MÓVILES. cerasa.es [página web]. (27, julio, 2015). [Consultado el 1, octubre, 2022]. Disponible en Internet: <https://www.cerasa.es/media/areces/files/book-attachment-2725.pdf>.

Este desarrollo de aplicaciones requiere de intercomunicación entre sistemas para enviar y recibir información capturada o procesada, necesitando infraestructura física, y de software que permita la transmisión de datos en tiempo real por parte de los dispositivos. [8]

8.2.2 Servicios cloud (AWS)

Amazon Web Services, es un proveedor de servicios y productos de TI tales como: almacenamiento, comunicación, administración en la nube entre muchos otros.

Es usado por un gran número de plataformas tecnológicas de vanguardia tales como Netflix, Reddit, Dropbox entre muchos otros.

Este proveedor nos proporciona la posibilidad de tener, mantener y manejar nuestros servicios informáticos y de infraestructura de forma óptima y remota. Con costos bajos, ya que solo factura el consumo o transacciones generadas a su sistema. [9]

- Seguridad de datos.

a. Cifrado punto a punto de datos.
b. Autenticación por token.
c. Seguridad en bases de datos.
d. Respaldo de los datos.
e. Seguridad en la infraestructura.
f. Computación distribuida.

8.2.3 Ubicación

La ubicación de los usuarios está determinada por el sensor GPS presente en los equipos móviles suscritos a la APP.

[8] KZGUNEA. Geolocalización, qué es y cómo funciona - KZblog. KZblog [página web]. (26, septiembre, 2018). [Consultado el 10, septiembre, 2021]. Disponible en Internet: <https://kzgunea.blog.euskadi.eus/blog/2017/03/31/geolocalizacion-que-es/>.

[9] AWS. ¿Qué es AWS? Amazon Web Services, Inc. [página web]. (2021). [Consultado el 16, septiembre, 2021]. Disponible en Internet: <https://aws.amazon.com/es/what-is-aws/>.

Esta utiliza la conexión provista por los operadores móviles y los satélites de posicionamiento global, este sistema de comunicación conjunta es conocido como AGPS (sistema de posicionamiento global asistido).

8.2.4 Herramientas de desarrollo

8.2.4.1 Lenguajes de programación

8.2.4.1.1 Java: “Java es un lenguaje de programación y una plataforma informática comercializada por primera vez en 1995 por Sun Microsystems. Hay muchas aplicaciones y sitios web que no funcionarán a menos que tenga Java instalado y cada día se crean más. Java es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas desde juegos hasta súper computadoras, desde teléfonos móviles hasta Internet”.[10]

Este lenguaje se encargará de contener y manejar la lógica de la aplicación, conexión a base de datos y exposición de los servicios REST para la aplicación móvil.

8.2.4.1.2 Spring Framework: “Es un framework open source que facilita la creación de aplicaciones de todo tipo en Java, Kotlin y Groovy. Se utilizará en la implementación del microservicio de la aplicación. Permitiéndonos exponer los API de consumo por parte de la aplicación móvil”.[11]

8.2.4.1.3 JWT: “JSON Web token (JWT) es un estándar abierto (RFC 7519) que define una forma compacta y autónoma de transmitir información de forma segura entre las partes un objeto JSON. Esta información se puede verificar y confiar porque está firmada digitalmente. Los JWT se pueden firmar usando un secreto (con el algoritmo HMAC) o un par de claves pública / privada usando RSA o ECDSA”[12].

---

[10] ORACLE. ¿Qué es la tecnología Java y por qué la necesito? Java [página web]. (2021). [Consultado el 17, septiembre, 2021]. Disponible en Internet: <https://www.java.com/es/download/help/whatis_java.html>.

[11] PAHINO, Ruben. ¿Qué son Spring framework y Spring Boot? Tu primer programa Java con este framework - campusMVP.es. campusMVP.es [página web]. (31, marzo, 2020). [Consultado el 17, septiembre, 2021]. Disponible en Internet: <https://www.campusmvp.es/recursos/post/que-son-spring-framework-y-spring-boot-tu-primer-programa-java-con-este-framework.aspx>.

[12] AUTH0. JWT.IO - JSON Web Tokens Introduction. JSON Web Tokens - jwt.io [página web]. (2021). [Consultado el 22, septiembre, 2021]. Disponible en Internet: <https://jwt.io/introduction/>.

JWT se implementará dentro de la aplicación con el fin de cifrar la comunicación entre la aplicación móvil y los servicios REST, protegiendo así los datos transportados. Se envía como cabecera de petición que será validada por el microservicio permitiendo así el acceso o no a la información suministrada por este.

8.2.4.1.4 Swagger: “Es un framework para documentar APIs Rest desde muy diferentes fuentes: Archivos de configuración, XML, C#, Javascript, Ruby, PHP, Java, Scala… además existen multitud de módulos que te pueden ayudar a integrarlo en tu proyecto. Con swagger puedes describir, producir, consumir y visualizar APIs, por lo que te será interesante como desarrollador de tu proyecto, pero también desde el punto de vista de un tercero que pudiese consumir tu API, o un usuario que busca información sobre un error o como tu API está construida, o un tester de tu equipo que quiere probar la funcionalidad de backend como primera medida para excluir un fallo en este lado para centrarse en Front end o no”[13].

Swagger nos permitirá mostrar la colección de API´S publicadas del microservicio de moto viaje, de esta manera los consumidores del API podrán identificar los métodos y parámetros necesarios para el consumo exitoso de los datos.

8.2.4.1.5 Java crypto: “Proporciona las clases e interfaces para realizar operaciones criptográficas. Las operaciones criptográficas definidas en este paquete incluyen encriptación, generación de claves y acuerdo de claves y generación de códigos de autentificación de mensajes (MAC).
El soporte para encriptación incluye algoritmos de cifrado simétricos, asimétricos, por bloques y de flujo”[14].

Usando la librería de java crypto, se cifrarán todos los datos tanto enviados en los cuerpos de la petición, como los datos generados como respuesta a los métodos invocados en el API, haciendo así

[13] ABAD, Ramón. ¿Qué es swagger? – Ramón Abad y punto… net. Ramón Abad y punto… net – Resuelve los problemas a los que otros ya nos hemos enfrentado en desarrollo .net [página web]. (4, abril, 2017). [Consultado el 21, septiembre, 2021]. Disponible en Internet: <http://ramonabadypuntonet.org/2017/04/04/que-es-swagger/>.

[14] ORACLE. javax.crypto (Java Platform SE 7). Moved [página web]. (2020). [Consultado el 9, junio, 2022]. Disponible en Internet: <https://docs.oracle.com/javase/7/docs/api/javax/crypto/package-summary.html>.

casi imposible acceder a la información comunicada a través de las peticiones HTTP.

8.2.4.1.6 Apache tomcat: “Es un contenedor de servlets que se puede usar para compilar y ejecutar aplicaciones web realizadas en Java. Implementa y da soporte tanto a servlets como a páginas JSP (Java Server Pages) o Java Sockets”[15]. Este servidor contendrá el archivo .war generado del API programado. Al ejecutarse este archivo se expondrá el proyecto a la red permitiendo así su consumo desde cualquier dispositivo móvil con la aplicación y las condiciones de seguridad configuradas previamente.

8.2.4.1.7 TypeScript: Es un lenguaje de código abierto que se basa en JavaScript, una de las herramientas más utilizadas del mundo, al agregar definiciones de tipos estáticos. Los tipos proporcionan una forma de describir la forma de un objeto, proporcionando una mejor documentación y permitiendo que TypeScript valida que su código está funcionando correctamente.[16]

8.2.4.1.8 Angular: “Es un framework open source desarrollado por Google para facilitar la creación y programación de aplicaciones web de una sola página, las webs SPA (Single Page Application). Angular separa completamente el frontend y el backend en la aplicación, evita escribir código repetitivo y mantiene todo más ordenado gracias a su patrón MVC (Modelo-Vista-Controlador) asegurando los desarrollos con rapidez, a la vez que posibilita modificaciones y actualizaciones”[17]. Angular es el framework base de la aplicación móvil, al usarlo en conjunto con ionic, se obtendrá una aplicación útil y funcional, de acuerdo con nuestras necesidades.

8.2.4.1.9 Ionic: “Ionic Framework es un SDK de frontend de código abierto para desarrollar aplicaciones híbridas basadas en tecnologías web (HTML, CSS y JS). Es decir, un framework que nos permite desarrollar aplicaciones para iOS nativo, Android y la web, desde

[15] ARSYS. Tomcat: Ventajas de uso en Servidores Cloud | Arsys. Blog de arsys.es [página web]. (26, enero, 2017). [Consultado el 14, julio, 2022]. Disponible en Internet: <https://www.arsys.es/blog/tomcat-servidores-cloud>.
[16] MICROSOFT. JavaScript With Syntax For Types. TypeScript: JavaScript With Syntax For Types. [página web]. (2022). [Consultado el 14, septiembre, 2021]. Disponible en Internet: <https://www.typescriptlang.org/>.
[17] QUALITY DEVS. ¿Qué es Angular y para qué sirve? Quality Devs [página web]. (16, septiembre, 2019). [Consultado el 23, septiembre, 2021]. Disponible en Internet: <https://www.qualitydevs.com/2019/09/16/que-es-angular-y-para-que-sirve/>.

una única base de código. Su compatibilidad y, gracias a la implementación de Cordova e Ionic Native, hacen posible trabajar con componentes híbridos. Se integra con los principales frameworks de frontend, cómo Angular, React y Vue, aunque también se puede usar Vanilla JavaScript. Este framework fue creado en 2013 por Drifty Co. y hasta la llegada de React Native ha sido una de las tecnologías líderes para el desarrollo de aplicaciones móviles híbridas"[18]. Al usar este framework podemos generar aplicaciones para los diferentes sistemas operativos ya bien sean Android o Apple, facilitando así el desarrollo de la aplicación incrementado la velocidad de producción de esta.

8.2.4.2 Versionamiento de código

8.2.4.2.1 Git: "Es un software de control de versiones diseñado por Linus Torvalds. La pregunta es ¿qué es control de versiones? Pues bien, se define como control de versiones a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración de este, es decir, a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración, y para los que aún no les queda claro del todo, control de versiones es lo que se hace al momento de estar desarrollando un software o una página web.
Exactamente es eso que haces cuando subes y actualizas tu código en la nube, o le añades alguna parte o simplemente le editas cosas que no funcionan como deberían o al menos no como tú esperarías"[19].
Git tendrá la función de alojar todo el código, permitiendo mantener el trabajo alineado entre los miembros del grupo, y al mismo tiempo mantendrá los proyectos seguros y disponibles en cloud.

8.2.5 Arquitectura de software

8.2.5.1 Orientada a microservicios: Normalmente cuando trabajamos con servicios siempre hay varios servicios que comparten un mismo

[18] AGÜERO ATMITIM, José María. Qué es Ionic: ventajas y desventajas de usarlo en apps móviles híbridas. Profile Software Services [página web]. (22, febrero, 2021). [Consultado el 10, septiembre, 2021]. Disponible en Internet: <https://profile.es/blog/que-es-ionic/>.
[19] ANDRÉS. Qué es Git. CódigoFacilito [página web]. (16, agosto, 2015). [Consultado el 22, septiembre, 2021]. Disponible en Internet: <https://codigofacilito.com/articulos/que-es-git>.

contexto. Por ejemplo, puede haber varios servicios para gestionar las nóminas y otro conjunto de servicios para gestionar por ejemplo los proveedores. Una vez definidos los contextos y aislados un grupo de servicios de otros podemos publicar la funcionalidad de estos a través de REST. Una vez que tengamos nuestros servicios agrupados de esta forma los convertiremos en aplicaciones independientes. Estas aplicaciones se pueden desplegar en un contenedor de forma totalmente independiente. Al ser cada microservicio independiente tenemos varias ventajas. Primero es más fácil para los desarrolladores hacerse con ellos ya que son relativamente "pequeños" y fáciles de abordar. [20]

8.2.6 Servicios cloud

8.2.6.1 AWS: "Amazon Web Services (AWS) es la plataforma en la nube más adoptada y completa en el mundo, que ofrece más de 200 servicios integrales de centros de datos a nivel global. Millones de clientes, incluso las empresas emergentes que crecen más rápido, las compañías más grandes y los organismos gubernamentales líderes, están usando AWS para reducir los costos, aumentar su agilidad e innovar de forma más rápida"[21]. Esta plataforma de nube nos permitirá tener nuestro API desplegado en el servidor tomcat y expuesto, facilitando el acceso a la información necesaria para el óptimo funcionamiento de la aplicación móvil.

8.2.6.2 EC2: "Amazon Elastic Compute Cloud (Amazon EC2) proporciona capacidad de Computación escalable en la nube de Amazon Web Services (AWS). El uso de Amazon EC2 elimina la necesidad de invertir inicialmente en hardware, de manera que puede desarrollar e implementar aplicaciones en menos tiempo. Puede usar Amazon EC2 para lanzar tantos servidores virtuales como necesite, configurar la seguridad y las redes y administrar el almacenamiento. Amazon EC2 le permite escalar hacia arriba o hacia abajo para controlar cambios en los requisitos o picos de popularidad, con lo que se reduce la necesidad de prever el

[20] DE ZÚÑIGA, Fernán García. Arquitectura de software: descubre en qué consiste | Arsys. Blog de arsys.es [página web]. (2, marzo, 2020). [Consultado el 24, septiembre, 2021]. Disponible en Internet: <https://www.arsys.es/blog/arquitectura-software>.
[21] AWS. ¿Qué es AWS? Amazon Web Services, Inc. [página web]. (2021). [Consultado el 16, septiembre, 2021]. Disponible en Internet: <https://aws.amazon.com/es/what-is-aws/>.

tráfico”[22]. EC2 nos permite tener el servidor Tomcat y el motor PostgreSQL funcionando sin necesidad de servidores físicos, además de estar disponibles y en línea para cumplir con nuestras necesidades, y en caso de aumentar el tráfico de la aplicación podrá soportarlo aumentando los recursos del servidor virtual.

8.2.7 Base de datos

8.2.7.1 PostgreSQL: “es un sistema de código abierto de administración de bases de datos del tipo relacional, aunque también es posible ejecutar consultas que sean no relaciones. En este sistema, las consultas relacionales se basan en SQL, mientras que las no relacionales hacen uso de JSON”[23]. Este motor de base de datos soportara la data de la aplicación móvil, al ser de código abierto y gratuito nos permite tener un buen soporte en línea, además de gran cantidad de información de cómo administrar y usar PostgreSQL.

8.2.8 Ubicación

8.2.8.1 Google Maps: “Es una herramienta de búsqueda de ubicaciones que permite localizar un punto concreto, calcular rutas, encontrar los lugares de interés más cercanos o ver la apariencia de un lugar a pie de calle a través de Google Street View. Fue desarrollada en 2005, inicialmente para Internet Explorer y Mozilla Firefox en PC. Sin embargo, su uso se ha generalizado de la mano de los Smartphone y ha sido entonces cuando ha pasado a convertirse en un elemento fundamental para hacer SEO local”[24]. Google maps se encargará de mostrar los mapas y las ubicaciones de los compañeros motociclistas, proporcionando funciones de posicionamiento para los usuarios de la aplicación.

---

[22] AWS. ¿Que es Amazon EC2? - Amazon Elastic Compute Cloud. Aws [página web]. (2022). [Consultado el 17, septiembre, 2021]. Disponible en Internet: <https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/concepts.html>.

[23] BORGES, Santiago. ¿Qué es PostgreSQL? - Para qué sirve, Características e Instalación. Infranetworking [página web]. (19, noviembre, 2019). [Consultado el 28, septiembre, 2021]. Disponible en Internet: <https://blog.infranetworking.com/servidor-postgresql/>.

[24] ARIMETRICS. Qué es Google Maps - Definición, significado y ejemplos. Arimetrics [página web]. (2022). [Consultado el 10, septiembre, 2021]. Disponible en Internet: <https://www.arimetrics.com/glosario-digital/google-maps>.

8.2.8.2 Sigfox: “Es una red de conectividad celular a nivel mundial enfocada para el Internet de las cosas (IoT: Internet of things). Esta red está diseñada para comunicaciones de baja velocidad permitiendo reducir los costes y el consumo de energía de los dispositivos conectados. Además, aparte de ser de baja velocidad, la comunicación también se basa en una banda muy estrecha, lo que permite a los dispositivos tener un alto poder de penetración a obstáculos, facilitando la comunicación a grandes distancias, incluso en suelo urbano”[25]. También existen otras tecnologías para desarrollar soluciones IOT como es Lorawan.

8.2.9 Notificación

8.2.9.1 Firebase: “Se trata de una plataforma móvil creada por Google, cuya principal función es desarrollar y facilitar la creación de apps de elevada calidad de una forma rápida, con el fin de que se pueda aumentar la base de usuarios y ganar más dinero. La plataforma está subida en la nube y está disponible para diferentes plataformas como iOS, Android y web. Contiene diversas funciones para que cualquier desarrollador pueda combinar y adaptar la plataforma a medida de sus necesidades. Firebase se inició cuando Google la compró en 2014, y seguidamente la fue mejorando mediante la compra del equipo de Divshot”[26]. Esta plataforma tendrá la responsabilidad de notificar a los demás miembros del viaje en la aplicación de los diferentes mensajes enviados, ya bien sean simples alertar o notificaciones de emergencia generadas por los usuarios.

8.2.9.2 Háblame: Es un servicio de mensajería de texto, expone un API de consumo que permite enviar mensajes de texto masivos a equipos móviles de diferentes operadores a cualquier parte de Colombia.[27]

---

[25] DSET. Sigfox. Productos IoT [página web]. (2022). [Consultado el 24, noviembre, 2021]. Disponible en Internet: <http://productos-iot.com/sigfox-3/>.

[26] PÉREZ CARDONA, MANUEL. Firebase, qué es y para qué sirve la plataforma de Google. Thinking for Innovation [página web]. (14, octubre, 2016). [Consultado el 17, noviembre, 2021]. Disponible en Internet: <https://www.iebschool.com/blog/firebase-que-es-para-que-sirve-la-plataforma-desarroladores-google-seo-sem/>.

[27] HABLAME. Hablame Colombia - Voz, Datos y Video. Hablame Colombia [página web]. (2022). [Consultado el 9, junio, 2022]. Disponible en Internet: <https://www.hablame.co/>.

9. METODOLOGÍA

9.1 METODOLOGÍA DE INVESTIGACIÓN

La metodología investigativa que usaremos es la cuantitativa. En donde se presentarán los antecedentes, experiencias y testimonios que nos sirvieron como base para llevar a cabo el desarrollo del proyecto.

La metodología de investigación cuantitativa hace uso de métodos y técnicas estadísticas para poder conocer los aspectos más relevantes del objeto de estudio.

Los métodos y técnicas de los cuales haremos uso para la investigación son:

Métodos	**Técnicas**
Encuestas	Cuestionarios
Experiencias	Recopilación de datos
Testimonios	Análisis de contenidos y herramientas existentes.

Tabla 1: Características metodología investigativa

Las características mayor utilidad para este proyecto son:

- En la metodología investigativa se tiende a adoptar un enfoque estructurado para estudiar el fenómeno, siendo esta estructura consecuente con gran parte de las técnicas utilizadas para recoger la información.

- Posibilita inferir los hallazgos de la investigación en poblaciones más grandes, esto es, permite la generalización estadística de los resultados.[28]

[28] AZUERO AZUERO, Ángel Enrique. Significatividad del marco metodológico en el desarrollo de proyectos de investigación | Azuero Azuero | Revista Arbitrada Interdisciplinaria Koinonía. Fundación Koinonía [página web]. (12, noviembre, 2018).

9.1.1 Estadísticas

Para el estudio de viabilidad del proyecto, se realizó la siguiente encuesta haciendo uso de la herramienta “Formularios” de Google:

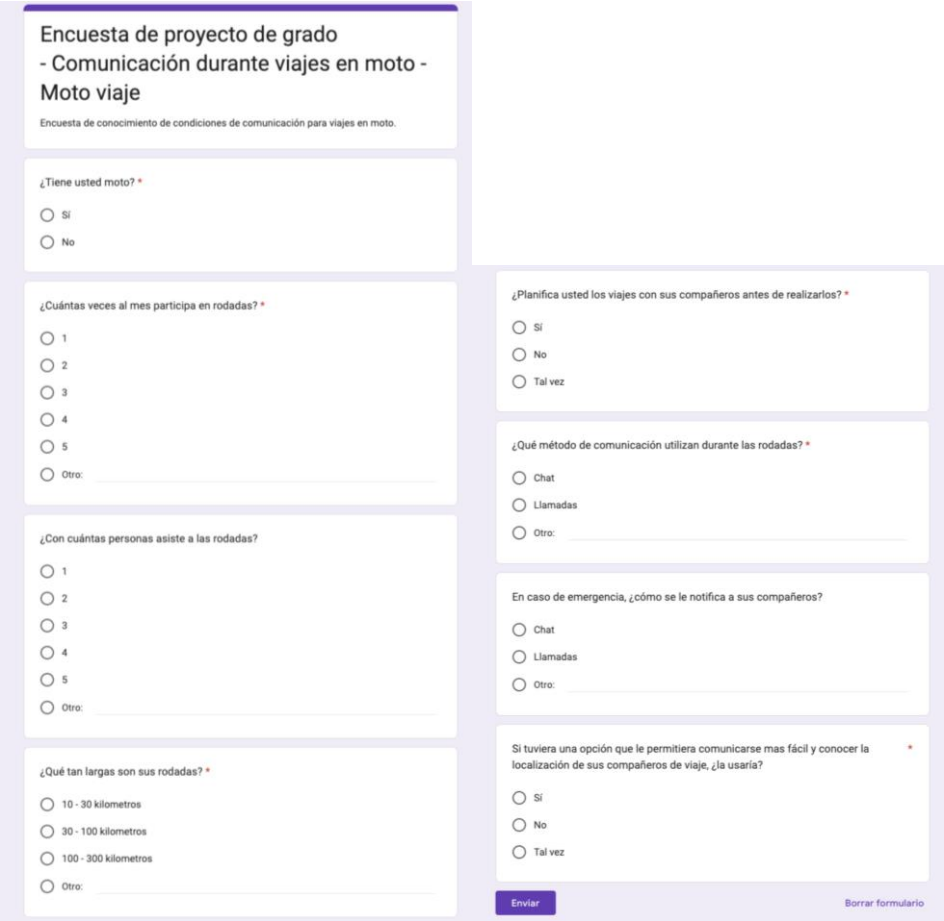


Ilustración 1: Encuesta de viabilidad

9.1.2 Resultados encuesta:

La encuesta estuvo disponible para responder durante 2 semanas y se registraron 105 respuestas.
A continuación se muestran los resultados de cada pregunta.

[Consultado el 1, octubre, 2021]. Disponible en Internet: <https://fundacionkoinonia.com.ve/ojs/index.php/revistakoinonia/article/view/274/html>.

- Esta pregunta tiene como fin identificar si los encuestados hacen parte del grupo objetivo al cuál se quiere enfocar la funcionalidad de la aplicación:

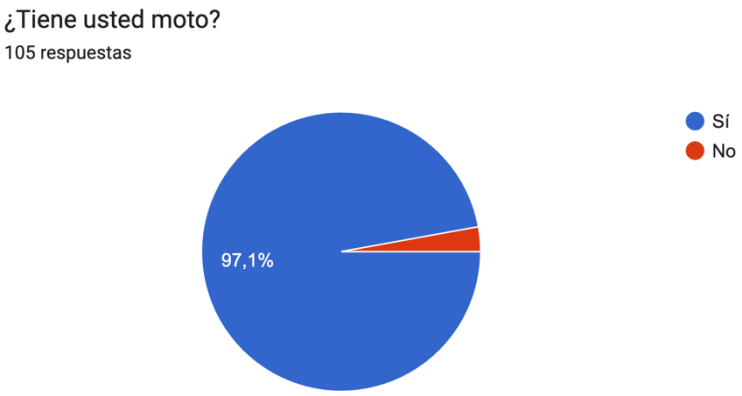


Ilustración 2: Resultados pregunta 1

- Esta pregunta tiene como fin obtener un promedio de viajes por mes, para poder estimar la frecuencia de uso que tendría la aplicación:

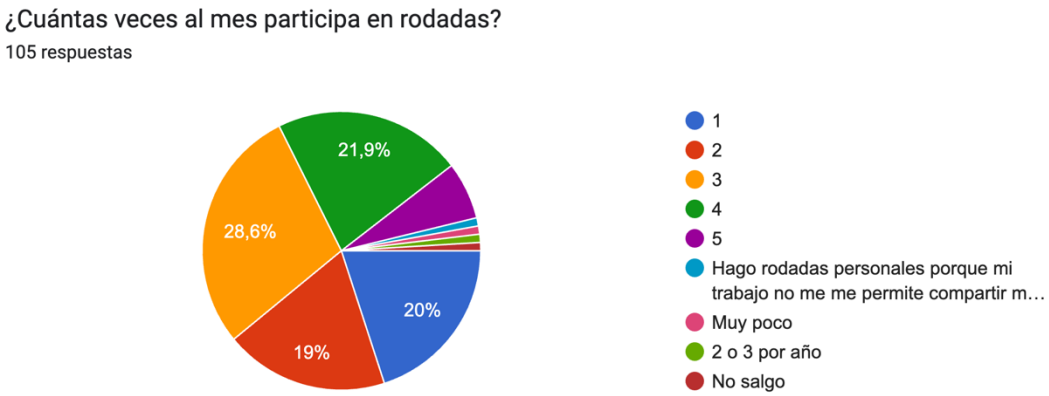


Ilustración 3: Resultados pregunta 2

- Esta pregunta tiene como fin conocer el promedio de personas que conforman grupos de viajes:

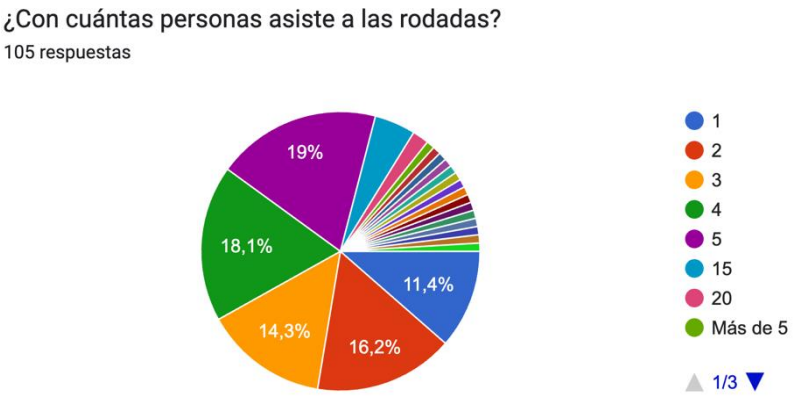


Ilustración 4: Resultados pregunta 3

- Esta pregunta tiene como fin el poder determinar las distancias que pueden recorrerse en un viaje:

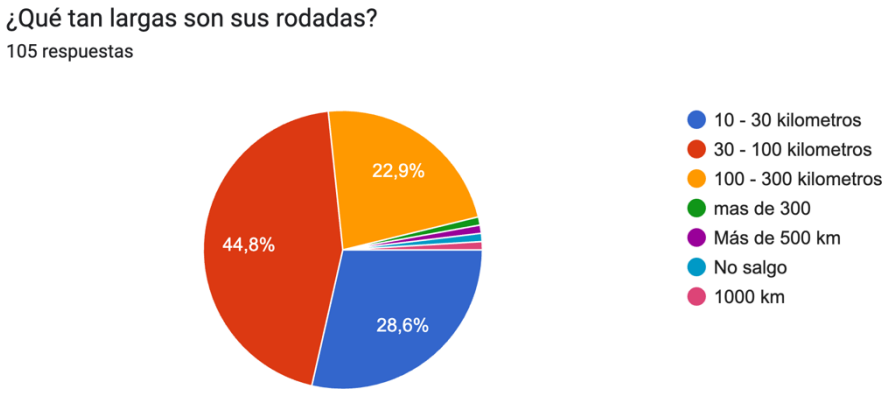


Ilustración 5: Resultados pregunta 4

- Esta pregunta tiene como fin identificar si los viajes en grupo cuentan con una planeación previa a su ejecución:



Ilustración 6: Resultados pregunta 5

- Esta pregunta tiene como fin identificar los medios de comunicación usados durante los viajes:

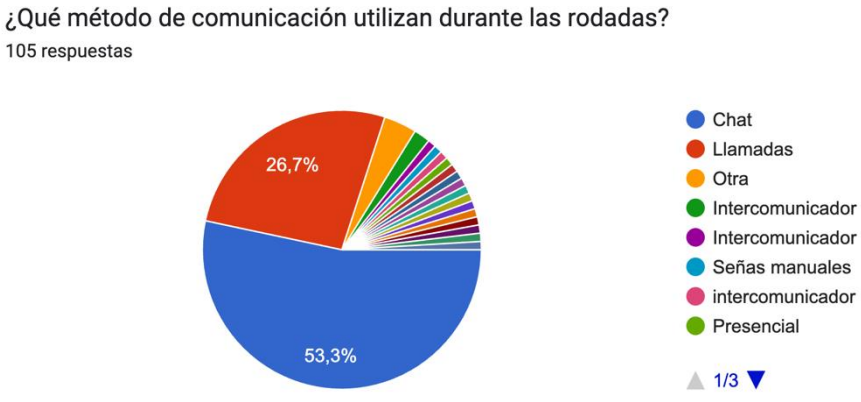


Ilustración 7: Resultados pregunta 6

- Esta pregunta tiene como fin conocer la forma en la que se comunican las emergencias e imprevistos que ocurren durante los viajes:

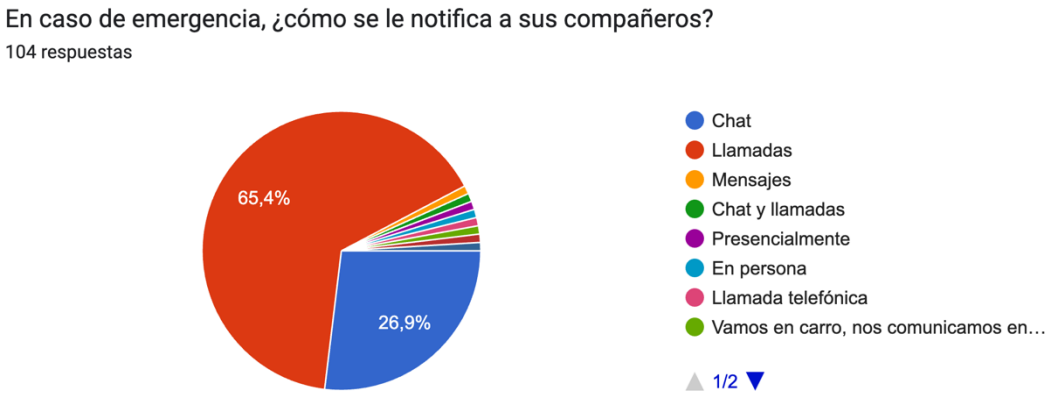


Ilustración 8: Resultados pregunta 7

- Esta pregunta tiene como fin identificar si los motoviajeros estarían dispuestos a usar la aplicación desarrollada durante sus recorridos:



Ilustración 9: Resultados pregunta 8

9.1.3 Análisis de los resultados de la encuesta:

- Un gran porcentaje de los encuestados estarían dispuestos a usar la propuesta de aplicación que se pretende desarrollar.
- El 86.6% de los encuestados prefieren hacer viajes en grupo.
- En promedio, los encuestados realizan 2 viajes por mes.
- El medio más usado para informar de algún incidente durante el viaje son las llamadas telefónicas.

9.2 METODOLOGÍA DE PROYECTO

Para el desarrollo del proyecto se va a hacer uso de la metodología de trabajo Scrum.

Scrum es un marco dentro del cual las personas pueden abordar problemas adaptativos complejos, mientras entregan productos de manera productiva y creativa del más alto valor posible.[29]

Scrum se compone de unos eventos, los cuales tienen la funcionalidad de fomentar un orden, regularidad y minimizar la necesidad de reuniones. Estos son los eventos definidos:

- Sprint: Son eventos de longitud fija de un mes o menos para crear consistencia. Un nuevo Sprint comienza inmediatamente después de la conclusión del Sprint anterior.
- Planificación Sprint: Es el diseño del trabajo que se realizará para el Sprint.
- Scrum diario: Su propósito es revisar el estado del progreso hacia el Objetivo Sprint.
- Revisión de Sprint: En este evento se inspecciona el resultado del Sprint y se determinan futuras adaptaciones.
- Retrospectiva Sprint: Se deben planificar formas de aumentar la calidad y la eficacia para los siguientes ciclos sprint.

[29] PROYECTOSAGILES. Qué es SCRUM. Proyectos Ágiles [página web]. (2022). [Consultado el 16, septiembre, 2021]. Disponible en Internet: <https://proyectosagiles.org/que-es-scrum/>.

10. FACTIBILIDAD

10.1 FACTIBILIDAD ECONÓMICA

Este proyecto es económicamente factible, teniendo en cuenta los pocos recursos que se requieren para poder ejecutarlo, los cuales son dos equipos de trabajo, asesorías de los tutores del proyecto, acceso a internet y pago de los proveedores de los servicios en la nube.

En las tablas relacionadas a continuación se describe dicha factibilidad, identificando los costos de hardware, software, servicios tecnológicos y recursos humanos necesarios para la realización del proyecto de investigación que se propone.

Se dividió en tres aspectos: planeación, desarrollo y otros recursos.

La distinción de los recursos de planeación se presenta en la Tabla 1 Factibilidad de Planeación:

Planeación				
Actividad	Descripción	Valor hora	Número unidades	Total
Asesorías	Asesoría por parte del tutor	$100.000	10	$1.000.000
Planeación	Planeación del proyecto	$22.000	60	$1.320.000
			Total	$2.320.000

Tabla 2: Factibilidad de Planeación

A continuación, en la Tabla 2 se presentarán los gastos correspondientes al desarrollo del proyecto:

Desarrollo									
Actividad	Descripción	Valor hora	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	Mes 6	Total
Base de datos	Estructuración y desarrollo de base de datos.	$35.000	15	10	2				$945.000

Desarrollo microservicios	Desarrollo de los microservicios necesarios para la lógica de la aplicación.	$32.000	120	100	100	100	100	100	$19.840.000
Desarrollo aplicación móvil	Desarrollo de la aplicación móvil para dispositivos móviles Android.	$32.000	120	100	100	100	100	100	$19.840.000
Desarrollo aplicación web	Desarrollo de la aplicación web para hacer el seguimiento.	$25.000	0	0	20	20	40	40	$3.000.000
Ambiente en AWS de base de datos	Implementar el espacio en AWS para la base de datos que usará la aplicación.	$168	100	100	100	100	100	100	$100.608
Ambiente en AWS para microservicios	Implementar el espacio en AWS para los microservicios que usará la aplicación.	$305	0	0	100	100	100	100	$121.948
Espacio en AWS para base de datos	Valor del espacio en AWS para la base de datos.	$305	0	0	100	100	100	100	$121.948
Espacio AWS para los microservicios	Valor del espacio en AWS para los microservicios.	$3.087	0	0	100	100	100	100	$1.234.744

								Total	$45.204.248

Tabla 3: Factibilidad Económica Desarrollo

Adicionalmente se muestran los gastos adicionales en la Tabla 3 que serán cubiertos por los desarrolladores del proyecto:

Otros				
Actividad	Descripción	Valor	Cantidad	Total
Equipos computo	Equipos de cómputo necesarios para el desarrollo del proyecto	$5.000.000	2	$10.000.000
Escritorio	Escritorios necesarios para el desarrollo del proyecto	$150.000	2	$300.000
Pantalla	Pantalla auxiliar para el desarrollador	$450.000	2	$900.000
Teclado y mouse	Teclado y mouse auxiliar para el desarrollador	$50.000	2	$100.000
Internet	Internet necesario para el desarrollo del proyecto	$80.000	12	$960.000
Sillas	Sillas para los desarrolladores	$220.000	2	$440.000
Licencias Windows	Licencias para los equipos de computo	$85.000	2	$170.000
			Total	$12.870.000

Tabla 4: Otros Gastos

En la tabla 4 se muestra el total de los gastos necesarios para el desarrollo del proyecto:

Ítem	Costo
Planeación	$2.320.000
Desarrollo	$45.204.248
Otros	$12.870.000
Total	$60.394.248

Tabla 5: Total Gastos

11. CRONOGRAMA

Aclaración:
Se ajustan las fechas del desarrollo del proyecto de acuerdo con la fecha de aprobación de este, que fue en el mes de marzo de 2022 y antes tenía como fecha de inicio noviembre de 2021.

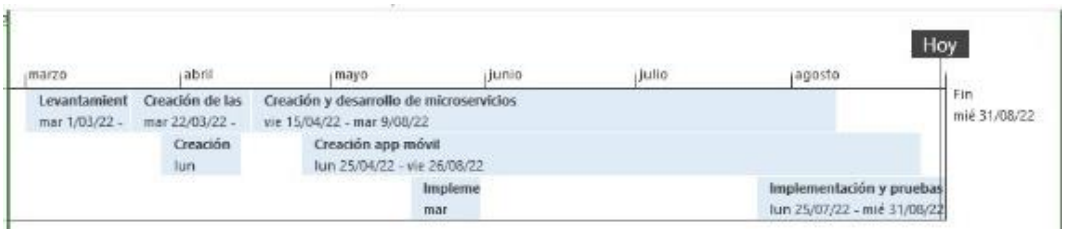
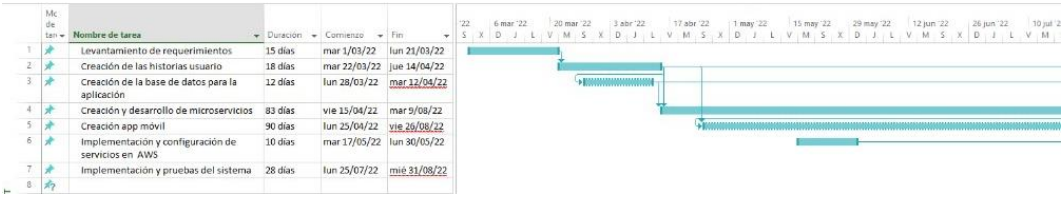


Ilustración 10: Cronograma



	Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin
1		Levantamiento de requerimientos	15 días	mar 1/03/22	lun 21/03/22
2		Creación de las historias usuario	18 días	mar 22/03/22	jue 14/04/22
3		Creación de la base de datos para la aplicación	12 días	lun 28/03/22	mar 12/04/22
4		Creación y desarrollo de microservicios	83 días	vie 15/04/22	mar 9/08/22
5		Creación app móvil	90 días	lun 25/04/22	vie 26/08/22
6		Implementación y configuración de servicios en AWS	10 días	mar 17/05/22	lun 30/05/22
7		Implementación y pruebas del sistema	28 días	lun 25/07/22	mié 31/08/22
8					

Ilustración 11: Actividades

12. DESARROLLO PROYECTO

12.1 ARQUITECTURA DE LA INFRAESTRUCTURA

Para el desarrollo del problema propuesto se requiere de una infraestructura capaz de soportar las comunicaciones y el almacenamiento de los datos que la aplicación necesita para su funcionamiento.

En pro de generar una arquitectura e infraestructura seguras, confiables y escalables se determinó que el mejor aliado es AWS; el cual nos permite crear los espacios necesarios para instalar el software requerido y decidido para el correcto funcionamiento.

AWS brinda servicios que nos permiten:

- Almacenar los datos generados por y para la aplicación, al mismo tiempo que nos permite respaldarlos en diferentes unidades de almacenamiento.

- Comunicar los diferentes componentes de forma segura; garantizando una alta disponibilidad de los servicios.

- Configurar y monitorear de forma fácil y eficaz el sistema, desde cualquier lugar en cualquier momento.

- Escalar de forma vertical el sistema cuando se incremente la concurrencia, aumentado los recursos físicos disponibles para cada servidor.

La arquitectura de la solución se ilustra en el siguiente diagrama:

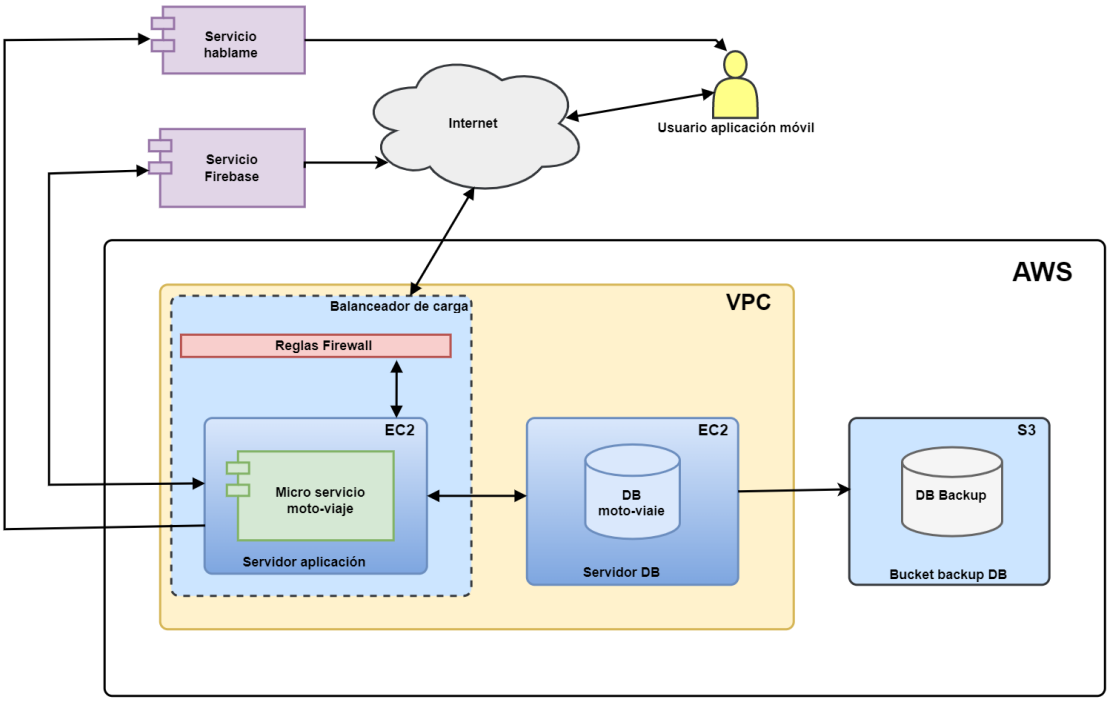


Ilustración 12: Diagrama arquitectura solución

12.1.1 Base de datos

La base de datos esta implementada en un motor PostgreSQL, y a su vez alojada en un servidor Linux. Este servidor está instalado en un servicio EC2 de AWS y cuenta con una conexión a la red que permite la comunicación con el servidor de aplicaciones por medio de credenciales.

12.1.1.1 Seguridad en la base de datos

A la base de datos solo se puede acceder por medio de credenciales y el único componente que podrá crear o modificar los datos de la misma es el micro servicio.

Como medida de precaución para garantizar la integridad de los datos, la contraseña del usuario está cifrada, usando la librería crypto de spring boot; solo el usuario conocerá su contraseña, ya que la aplicación móvil envía los datos cifrados hacia el servidor de aplicaciones, que a su vez vuelve a cifrar la contraseña para

que así este acorde con la que se encuentra almacenada en la base de datos.

12.1.1.2 Seguridad de los datos

Para asegurar que los datos estén respaldados y en caso de un problema no exista perdida de información se creó un bucket para almacenar los respaldos de la base de datos, este se ejecutará inicialmente una vez al día, y a medida que se incremente la cantidad de data, se reducirá el tiempo entre los intervalos del respaldo.

12.1.2 Servidor de aplicaciones

El micro servicio está corriendo en un servidor de aplicaciones Tomcat 9, este está contenido en un servicio EC2 de AWS. Permitirá desplegar el código JAVA, compilado en un WAR.

El micro servicio sirve de puente lógico entre la base de datos, los diferentes componentes externos como el generador de notificaciones push (Firebase) o el servicio de mensajes de texto (Háblame) y la aplicación móvil; Además de cifrar los datos usados para la comunicación entre la aplicación móvil y la base de datos enviados por medio de los métodos de petición HTTP.

12.1.2.1 Seguridad en el servidor de aplicaciones

El servidor de aplicaciones cuenta con las siguientes reglas de firewall:

- Reglas IPV4:

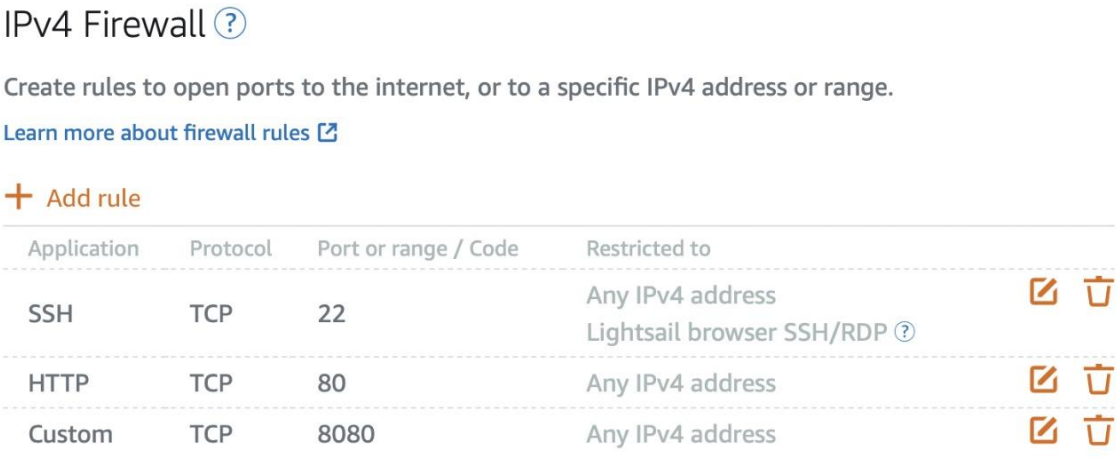


Ilustración 13: Reglas de firewall IPV4

- Reglas IPV6:

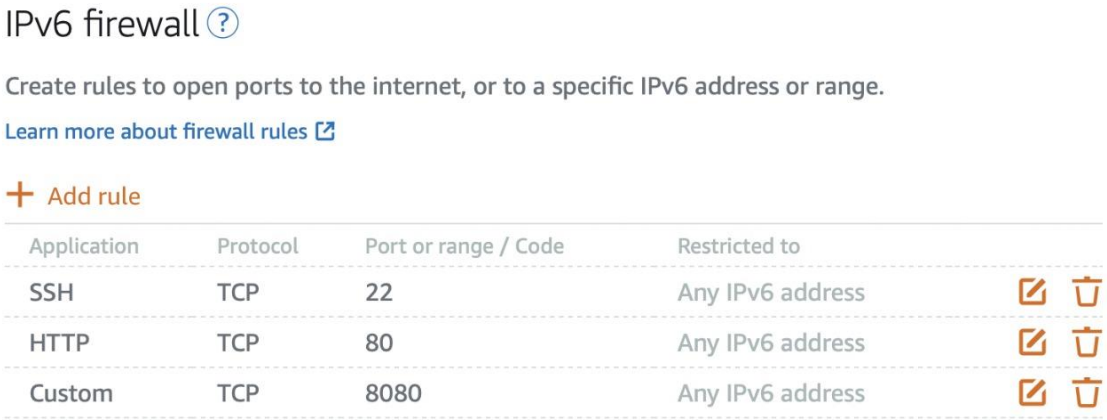


Ilustración 14: Reglas de firewall IPV6

Estas reglas se encargan de controlar los accesos al servidor, como por ejemplo el acceso por SSH para la administración, el acceso por HTPP para las peticiones del micro servicio, y el acceso TCP para la administración y despliegue en el servidor Tomcat.

Para administrarlo se requiere de credenciales de AWS así como de una llave privada de acceso que solo tienen el propietario de la cuenta de AWS.

12.1.3 Arquitectura CLOUD AWS

Para la solución se plantea un esquema de arquitectura implementado en AWS de la siguiente forma:

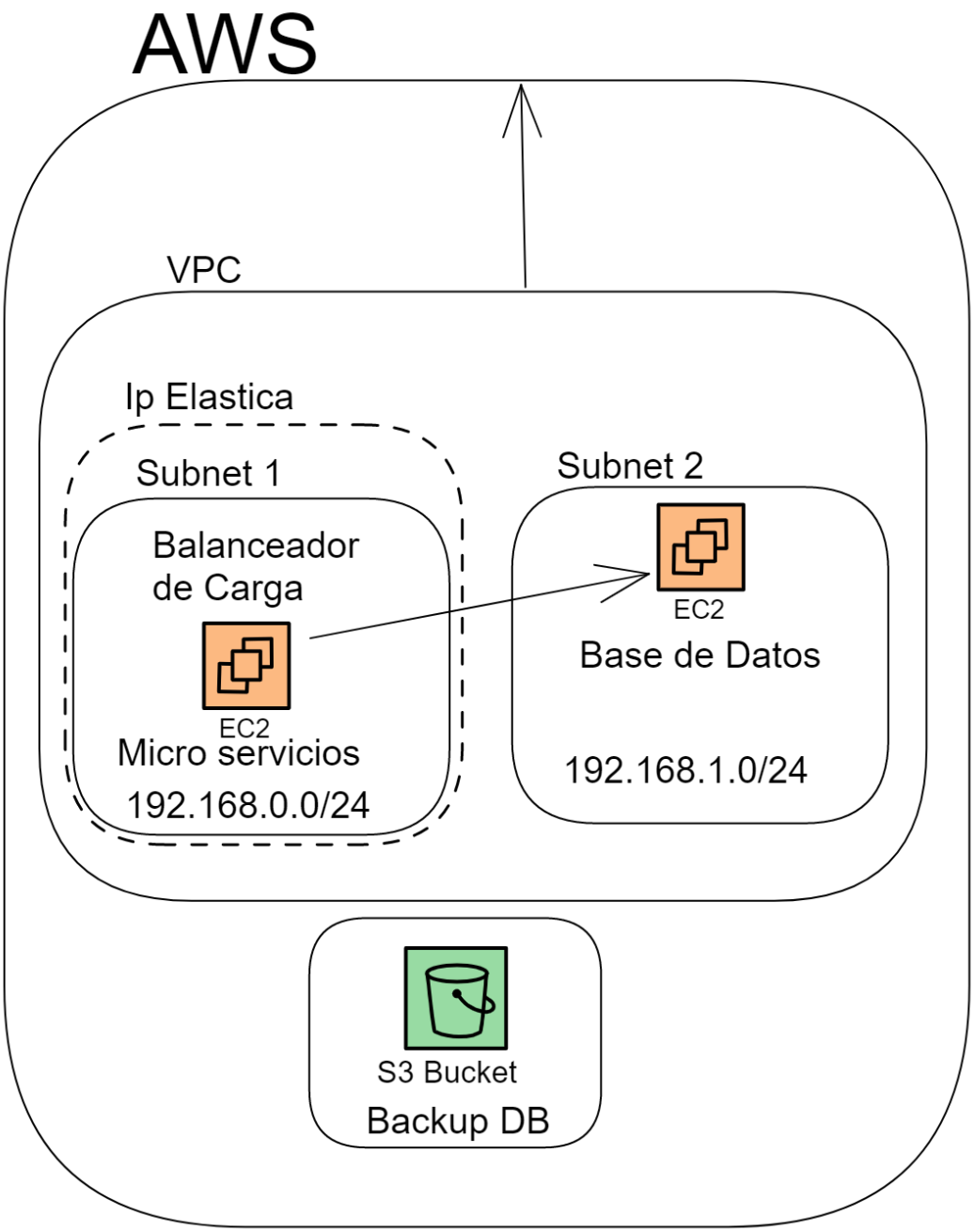


Ilustración 15: Diagrama arquitectura CLOUD AWS

En AWS está alojado el micro servicio, la base de datos y los respaldos de los datos.

Para que el sistema tenga una alta disponibilidad y sea estable se estructuró de la siguiente manera:

- **VPC**: Comunica las subredes creadas para contener los diferentes servidores; a su vez que provee y controla la conexión a la red externa.

- **IP elástica**: Nos provee el rango de IP disponibles para la asignación de las instancias del micro servicio.

- **Subnet 1**: Contiene el grupo de máquinas del micro servicio, al estar dentro de la VPC tiene conexión con la subnet 2.

- **Balanceador de carga**: Se encarga de distribuir las peticiones a las diferentes instancias creadas del micro servicio. Cuando se detecta un uso de CPU superior o igual al 60%, se crea una nueva instancia de micro servicio (hasta 5), para atender las solicitudes, logrando un escalamiento horizontal automático.

- **Subnet 2**: Contiene el servidor de base de datos, al estar dentro de la VPC tiene conexión con la subnet 1.

- **Bucket DB**: Almacena cada 12 horas una copia exacta del servidor de base de datos (AMI), permitiendo en caso de ser necesario una rápida recuperación ante errores o desastres.

12.1.3.1 Seguridad en el servicio de AWS

Para configurar y administrar todo el sistema antes descrito necesitamos conectarnos a la consola de administración de AWS, con las credenciales asignadas a nuestra cuenta privada.

Por medio de un acceso SSH, para el cual se necesitan una llave y credenciales de usuario, se puede acceder a la consola de administración de cada servidor, para administrarlo a nivel de sistema operativo.

12.2 DESARROLLO DEL SOFTWARE

12.2.1 ARQUITECTURA DE SOFTWARE

Diríjase al anexo 1 - Arquitectura de software moto viaje

12.2.2 API MOTO VIAJE

Diríjase al Anexo 2 - API moto viaje

12.2.3 MANUAL DE INSTALACIÓN

Diríjase al Anexo 3 - Manual de instalación

12.3 PROBLEMAS ENCONTRADOS

A continuación se detallaran los problemas encontrados durante las fases de planeación y desarrollo del proyecto, y del uso de la aplicación y cómo se solucionaron:

12.3.1 Desarrollo del proyecto

- La seguridad de las comunicaciones:

  Al momento de comunicar la aplicación con el micro servicio nos encontramos con que los datos podían ser interceptados fácilmente debido a que no viajan por un medio seguro. Para resolver esto, implementamos un sistema de cifrado AES con modo de operación ECB y esquema de relleno PKCS5 de 128 bits, que hace uso de llaves privadas para cifrar todas las entradas y salidas de datos tanto de la aplicación como del micro servicio.

- Escalabilidad de la infraestructura:

  Se debe garantizar que el sistema tenga una alta disponibilidad y soporte el alto tráfico que pudiera llegar a tener. Para dar solución a esto se implementó un balanceador de carga con reglas de replicación automática de hasta 5 instancias de micro servicio, una vez se alcance un 60% de uso de CPU.

12.3.2 Uso de la aplicación

- Disponibilidad de la red móvil:

  Para poder usar la aplicación se debe contar con una conexión a internet constante, la cual debe ser provista por el usuario de la aplicación. Si la aplicación llega a perder la conexión a internet, a los demás miembros del viaje se les mostrará la última ubicación que se envió.

- Recursos de hardware:

  Para el correcto funcionamiento de la aplicación se necesita acceder el sensor de GPS del dispositivo. Si no se le dan permisos de uso o si no se tiene el hardware de GPS la aplicación no permitirá ser usada.

- Compatibilidad de dispositivos:

  La aplicación fue desarrollada usando el framework Ionic, el cual permite el desarrollo de aplicaciones hibridas para los sistemas operativos Android y iOS, pero por limitaciones económicas no pudo ser probada en dispositivos con iOS.

13. RESULTADOS OBTENIDOS

A continuación se relacionan los resultados obtenidos del desarrollo del proyecto:

13.1 PRUEBAS APLICACIÓN

13.1.1 Registro nuevo usuario e inicio de sesión:

Se registra un nuevo usuario en la aplicación.

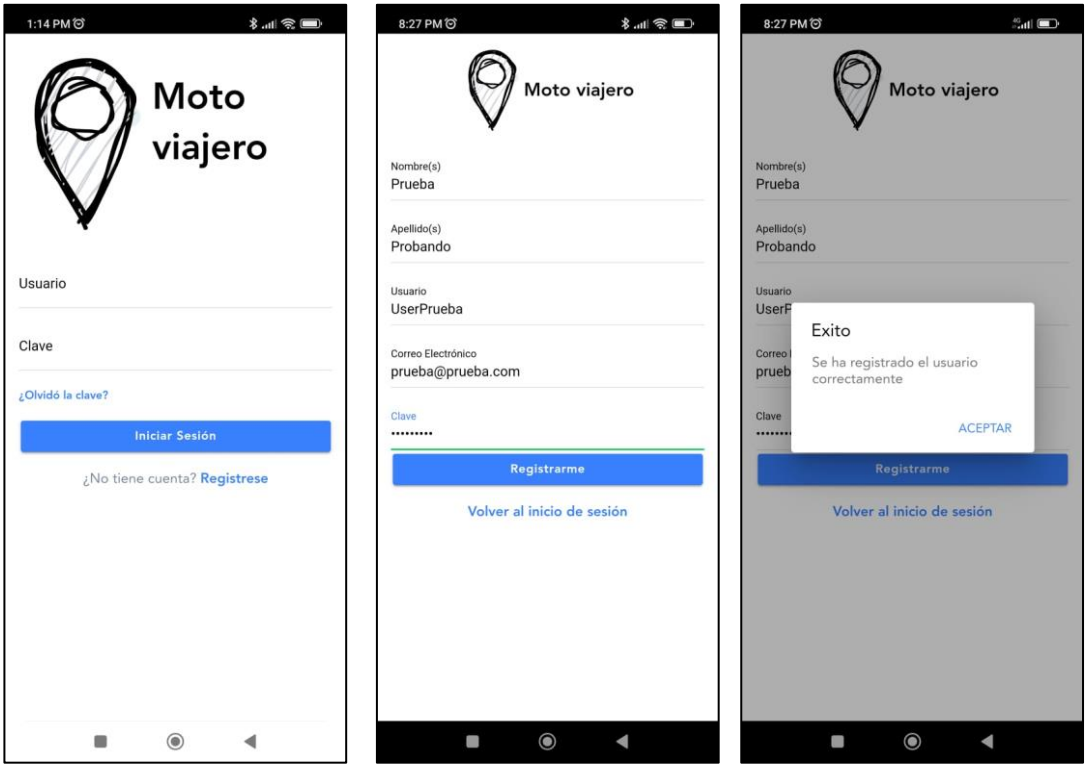


Ilustración 16: Registro de usuario

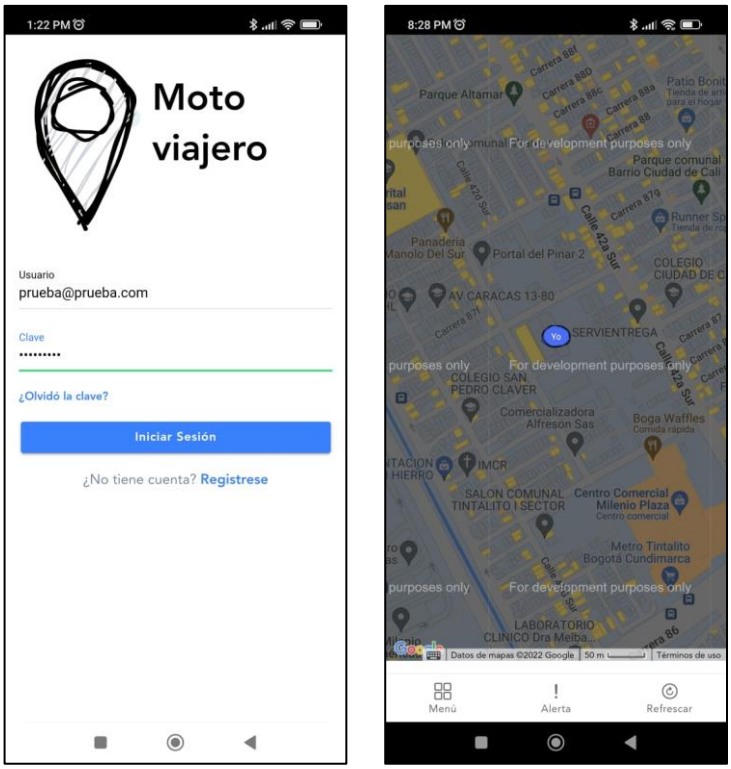


Ilustración 17: Inicio de sesión

Se evidencia que se crea el usuario “prueba@prueba.com” y que puede iniciar sesión correctamente.

13.1.2 Creación de grupos

Se crea un nuevo grupo y se asocian miembros al mismo.

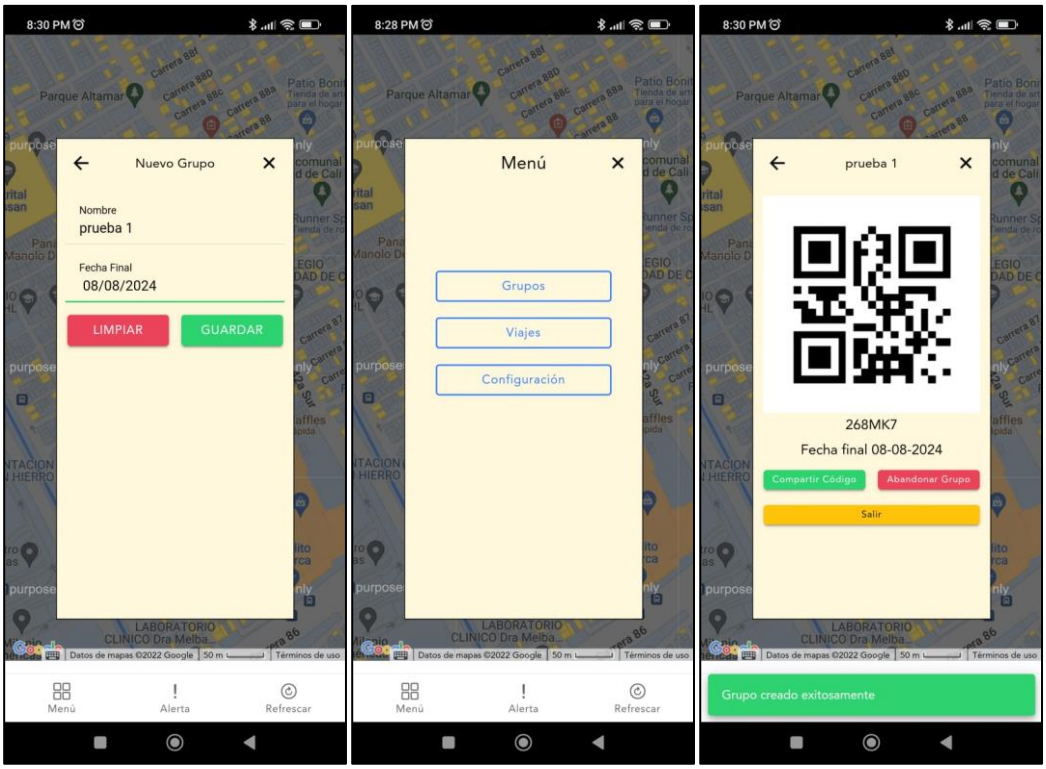


Ilustración 18: Creación de grupo

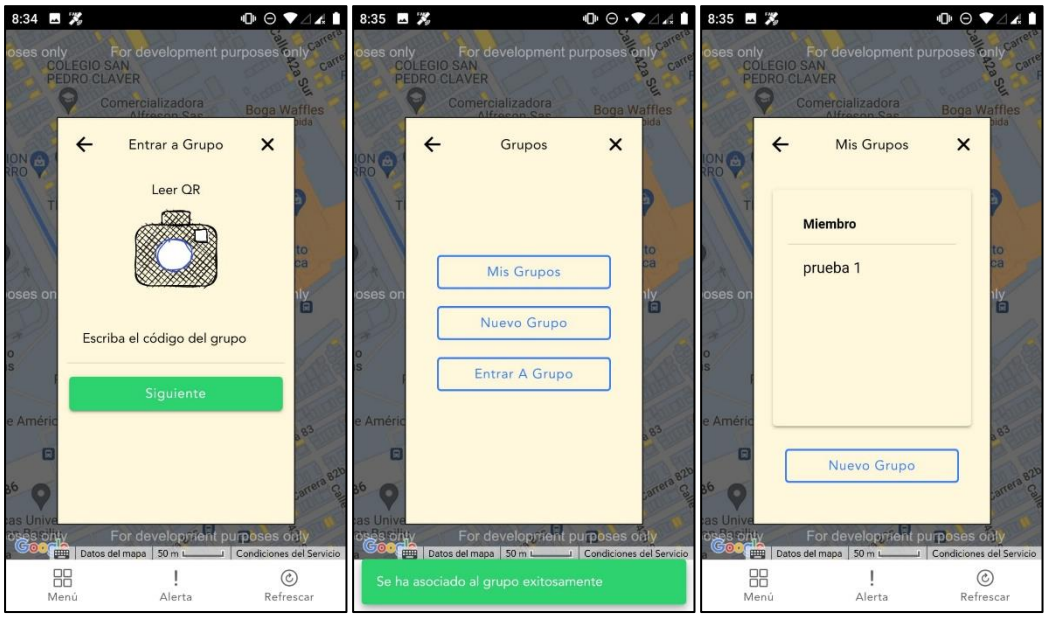


Ilustración 19: Asociar miembro al grupo

Se evidencia la creación del nuevo grupo “prueba 1” y se muestra el ingreso de moto viajeros al grupo.

13.1.3 Crear viaje y asociar miembros al viaje

Se crea un nuevo viaje y se le asocian viajeros.

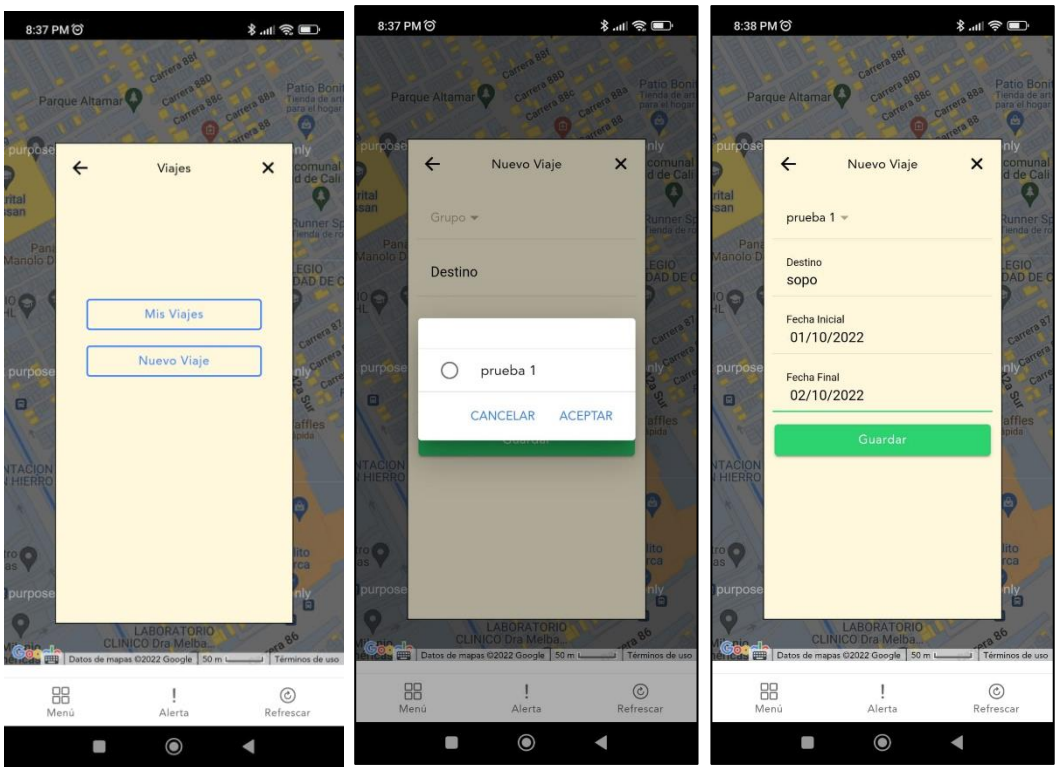


Ilustración 20: Creación nuevo viaje

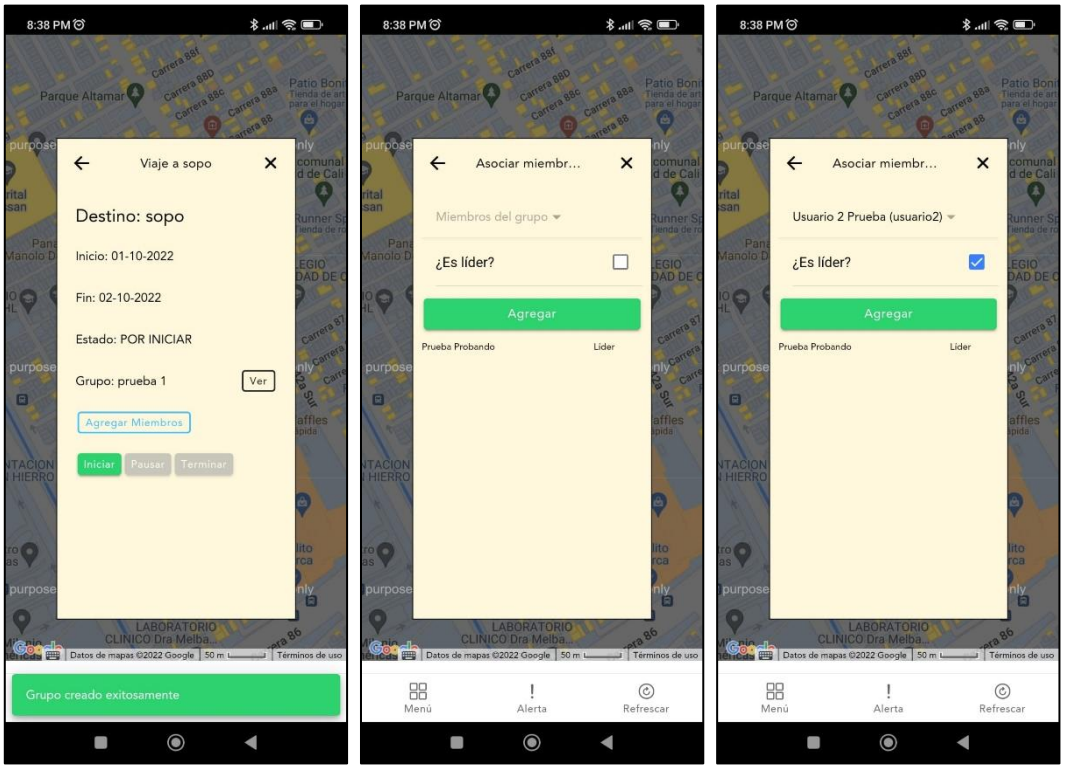


Ilustración 21: Asociar viajero

Se evidencia la creación del nuevo viaje a Sopo y la asociación del moto viajero “Usuario prueba 2”.

13.1.4 Viaje

Se inicia el viaje, se envían notificaciones durante el viaje y se termina el mismo.

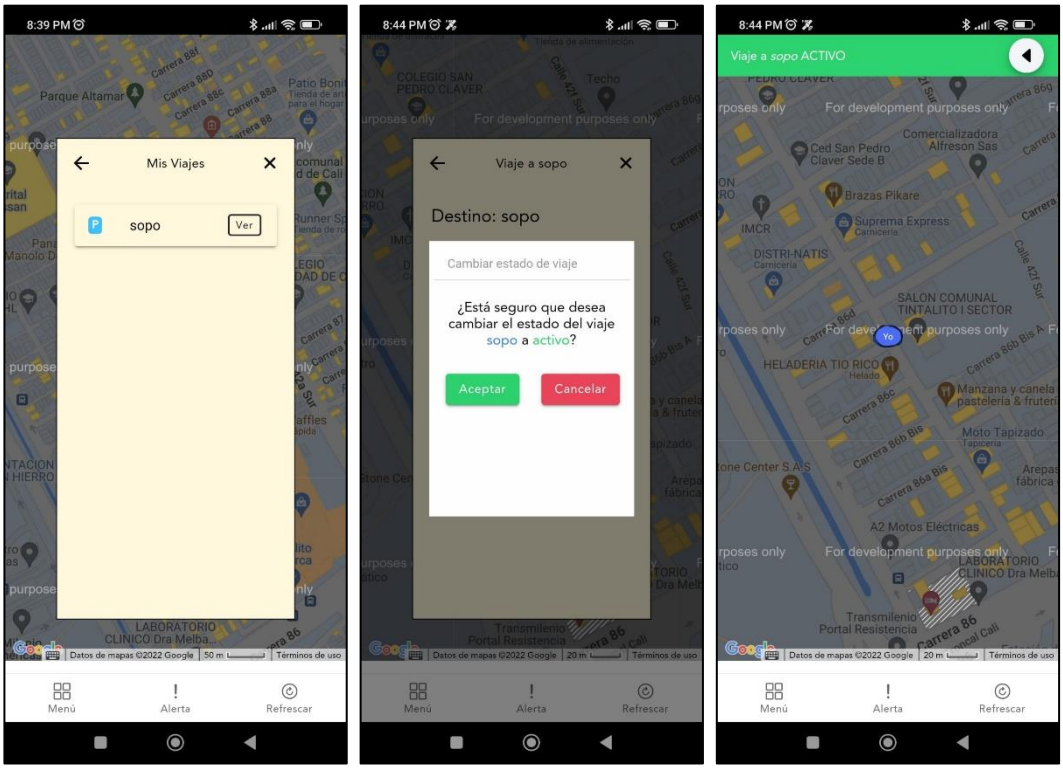


Ilustración 22: Iniciar viaje

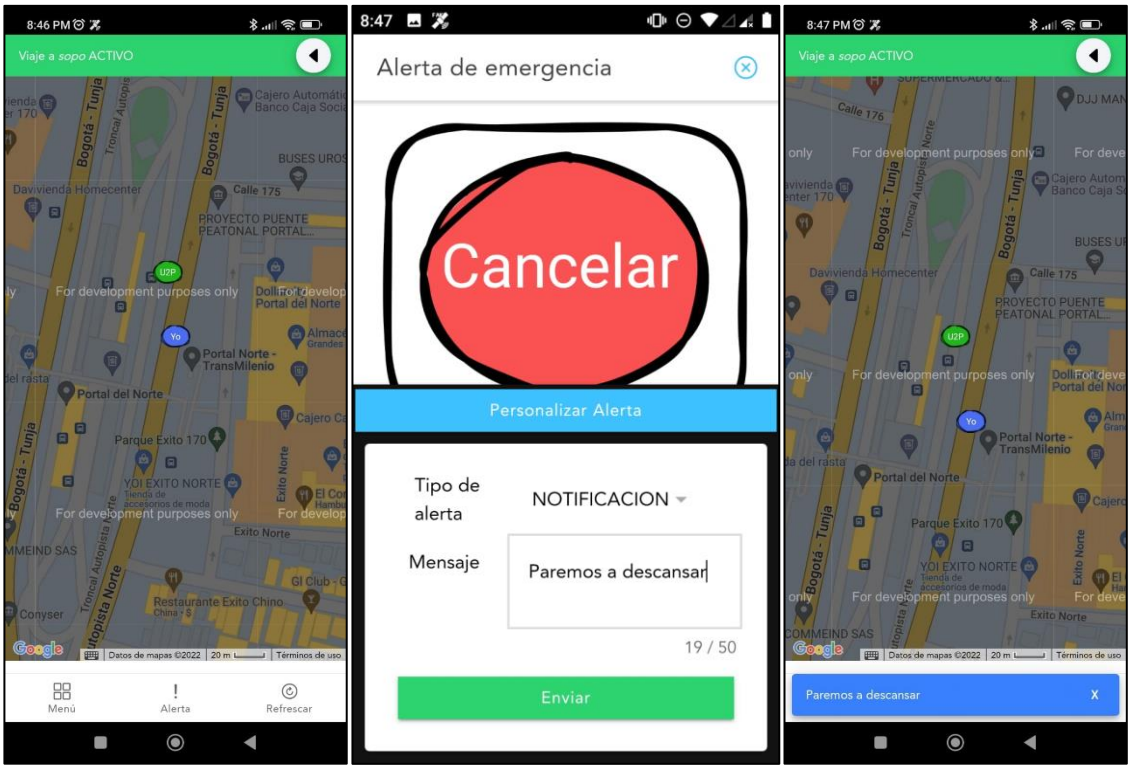


Ilustración 23: Envío de notificaciones

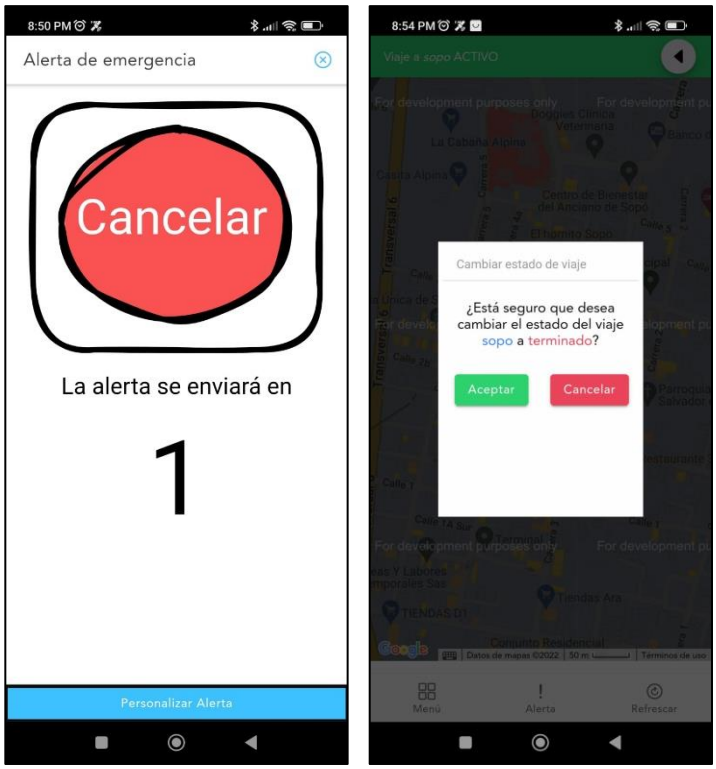


Ilustración 24: Envío de alerta y terminación del viaje

Se evidencia el inicio del viaje, el envío de notificaciones, envío de alerta, ubicación de miembros y terminación del viaje.

13.2 ANÁLISIS DE LOS RESULTADOS

Del desarrollo del proyecto podemos decir que:

- Se logró implementar una arquitectura distribuida, de alta disponibilidad, robusta, segura y fácilmente escalable que puede ser accedida desde cualquier lugar con una conexión a internet.

- Se configuró un sistema de respaldo de servidores periódico y un escalamiento automático cuando la plataforma detecta un alto tráfico de usuarios para garantizar el servicio.

- Se pudo establecer una comunicación segura entre el micro servicio y la aplicación, garantizando así una transferencia de datos confiable y protegida ante cualquier intento de modificación no autorizada.

- Se realizó un diseño de la aplicación intuitivo y amigable con el usuario para poder hacer uso de las funcionalidades sin dificultades.

- Se consiguió desarrollar el sistema cumpliendo con los alcances propuestos.

14. IMPACTOS

- Permitir la comunicación oportuna entre motoviajeros, haciendo que sea seguro y confiable realizar esta actividad.

- Facilitar el acceso a tecnologías de ubicación en tiempo real a personas que no poseen los recursos o conocimientos para instalar equipos de rastreo en sus motos.

- Disminuir los problemas de seguridad y comunicación de los motoviajeros, permitiéndoles en caso de una emergencia, accidente o eventualidad notificar su estado o informar de alguna anomalía como caídas, robos, inconvenientes mecánicos o simples paradas necesarias.

15. TRABAJO FUTURO

A continuación se plantean algunas características de la solución que se podrían implementar en un futuro:

- Solución de bugs.

  Como en cualquier proyecto de software, la aplicación puede tener bugs no identificados que se detectarán a medida que se masifique su uso.

- Buscador de sitios de destino:

  Cuando se crea un viaje, permitir elegir el destino ubicándolo en un mapa.

- Generar la versión para iOS:

  Aunque la aplicación se desarrolló para Android y iOS, no se desplegó la aplicación en iOS debido a que se requiere una licencia de desarrollador que tiene un costo de $99 USD por año.

- Seguimiento web:

  Para que una persona pueda ver la ubicación de algún moto viajero desde un navegador web.

- Implementación de estadísticas de uso y de viaje:

  Para poder mostrarle información de interés a los usuarios, por ejemplo: Distancia recorrida, destinos más visitados, entre otros.

- Infraestructura:

  A medida que se incrementen los usuarios de la aplicación será necesario aumentar las características de hardware y configuraciones del sistema.

16. CONCLUSIONES

- El prototipo de aplicación se desarrolló de forma tal que permite notificar de manera eficiente, a los miembros de un grupo, en caso de presentarse algún evento imprevisto o que requiera de atención inmediata.

- Dentro del prototipo se cuenta con módulos que permiten registrar usuarios, iniciar sesión y poder monitorear la ubicación de los miembros de un viaje, entre otro.

- Se hace uso de diferentes tecnologías que permiten la comunicación en tiempo real entre los usuarios de la aplicación, tales como Firebase y el API desarrollado para el prototipo de aplicación.

- La ubicación de los usuarios del prototipo de aplicación se puede visualizar en un mapa provisto por la tecnología de Google maps, así mismo, cuando se presenta un evento imprevisto, también se muestra la posición geográfica del usuario que sufre del imprevisto.

- El API del prototipo de aplicación se logró desplegar de manera satisfactoria haciendo uso de tecnologías cloud, específicamente Amazon Web Services, que permiten una alta disponibilidad y recuperación ante desastres para garantizar el uso correcto del software desarrollado en este documento.

17. BIBLIOGRAFIA

FRONTADO ESCOBAR, Camilo Andres. Sistema para la localización de motos mediante Gsm Y Gps en Bogotá - hdl:11349/22420. Repositorio Institucional Universidad Distrital - RIUD [página web]. (18, septiembre, 2019). [Consultado el 2, septiembre, 2022]. Disponible en Internet: <https://repository.udistrital.edu.co/handle/11349/22420>.

TRACAL ANTILEF, Sebastián. Aplicación móvil de alerta de motocicletas con dispositivo GPS. Biblioteca Digital UDLA [página web]. (2017). [Consultado el 1, septiembre, 2021]. Disponible en Internet: <http://repositorio.udla.cl/xmlui/handle/udla/258>.

VARGAS ROMERO, MIGUEL ÁNGEL. Diseño de dispositivo de seguimiento gps aplicado al transporte público de pasajeros. Universidad Santo Tomás [página web]. (2020). [Consultado el 1, septiembre, 2021]. Disponible en Internet: <https://repository.usta.edu.co/handle/11634/28840>.

DAVID, Garzón Ramírez y MICHAEL, Sanabria Guio. Sistema de geolocalización y análisis vehicular para motociclistas - hdl:11349/23570. Repositorio Institucional Universidad Distrital - RIUD [página web]. (2020). [Consultado el 1, septiembre, 2021]. Disponible en Internet: <https://repository.udistrital.edu.co/handle/11349/23570>.

ASENCIO VARGAS, Richard Paolo. Repositorio Universidad de Guayaquil: Desarrollo de un Sistema Web y Aplicativo Android Para el Control y Monitoreo por Geolocalización de Unidades de Transportación de Carga Pesada para la Empresa Transporte y Logística Honores Méndez. Repositorio Universidad de Guayaquil: Página de inicio [página web]. (2017). [Consultado el 1, septiembre, 2021]. Disponible en Internet: <http://repositorio.ug.edu.ec/handle/redug/19496>.

KZGUNEA. Geolocalización, qué es y cómo funciona - KZblog. KZblog [página web]. (26, septiembre, 2018). [Consultado el 10, septiembre, 2021]. Disponible en Internet: <https://kzgunea.blog.euskadi.eus/blog/2017/03/31/geolocalizacion-que-es/>.

AWS. ¿Qué es AWS? Amazon Web Services, Inc. [página web]. (2021). [Consultado el 16, septiembre, 2021]. Disponible en Internet: <https://aws.amazon.com/es/what-is-aws/>.

ORACLE. ¿Qué es la tecnología Java y por qué la necesito? Java [página web]. (2021). [Consultado el 17, septiembre, 2021]. Disponible en Internet: <https://www.java.com/es/download/help/whatis_java.html>.

PAHINO, Ruben. ¿Qué son Spring framework y Spring Boot? Tu primer programa Java con este framework - campusMVP.es. campusMVP.es [página web]. (31, marzo, 2020). [Consultado el 17, septiembre, 2021]. Disponible en Internet: <https://www.campusmvp.es/recursos/post/que-son-spring-framework-y-spring-boot-tu-primer-programa-java-con-este-framework.aspx>.

AUTH0. JWT.IO - JSON Web Tokens Introduction. JSON Web Tokens - jwt.io [página web]. (2021). [Consultado el 22, septiembre, 2021]. Disponible en Internet: <https://jwt.io/introduction/>.

ABAD, Ramón. ¿Qué es swagger? – Ramón Abad y punto… net. Ramón Abad y punto… net – Resuelve los problemas a los que otros ya nos hemos enfrentado en desarrollo .net [página web]. (4, abril, 2017). [Consultado el 21, septiembre, 2021]. Disponible en Internet: <http://ramonabadypuntonet.org/2017/04/04/que-es-swagger/>.

ORACLE. javax.crypto (Java Platform SE 7). Moved [página web]. (2020). [Consultado el 9, junio, 2022]. Disponible en Internet: <https://docs.oracle.com/javase/7/docs/api/javax/crypto/package-summary.html>.

ARSYS. Tomcat: Ventajas de uso en Servidores Cloud | Arsys. Blog de arsys.es [página web]. (26, enero, 2017). [Consultado el 14, julio, 2022]. Disponible en Internet: <https://www.arsys.es/blog/tomcat-servidores-cloud>.

MICROSOFT. JavaScript With Syntax For Types. TypeScript: JavaScript With Syntax For Types. [página web]. (2022). [Consultado el 14, septiembre, 2021]. Disponible en Internet: <https://www.typescriptlang.org/>.

QUALITY DEVS. ¿Qué es Angular y para qué sirve? Quality Devs [página web]. (16, septiembre, 2019). [Consultado el 23, septiembre, 2021]. Disponible en Internet: <https://www.qualitydevs.com/2019/09/16/que-es-angular-y-para-que-sirve/>.

AGÜERO ATMITIM, José María. Qué es Ionic: ventajas y desventajas de usarlo en apps móviles híbridas. Profile Software Services [página web]. (22, febrero, 2021). [Consultado el 10, septiembre, 2021]. Disponible en Internet: <https://profile.es/blog/que-es-ionic/>.

ANDRÉS. Qué es Git. CódigoFacilito [página web]. (16, agosto, 2015). [Consultado el 22, septiembre, 2021]. Disponible en Internet: <https://codigofacilito.com/articulos/que-es-git>.

DE ZÚÑIGA, Fernán García. Arquitectura de software: descubre en qué consiste | Arsys. Blog de arsys.es [página web]. (2, marzo, 2020). [Consultado el 24, septiembre, 2021]. Disponible en Internet: <https://www.arsys.es/blog/arquitectura-software>.

AWS. ¿Que es Amazon EC2? - Amazon Elastic Compute Cloud. Aws [página web]. (2022). [Consultado el 17, septiembre, 2021]. Disponible en Internet: <https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/concepts.html>.

BORGES, Santiago. ¿Qué es PostgreSQL? - Para qué sirve, Características e Instalación. Infranetworking [página web]. (19, noviembre, 2019). [Consultado el 28, septiembre, 2021]. Disponible en Internet: <https://blog.infranetworking.com/servidor-postgresql/>.

ARIMETRICS. Qué es Google Maps - Definición, significado y ejemplos. Arimetrics [página web]. (2022). [Consultado el 10, septiembre, 2021]. Disponible en Internet: <https://www.arimetrics.com/glosario-digital/google-maps>.

DSET. Sigfox. Productos IoT [página web]. (2022). [Consultado el 24, noviembre, 2021]. Disponible en Internet: <http://productos-iot.com/sigfox-3/>.

PÉREZ CARDONA, MANUEL. Firebase, qué es y para qué sirve la plataforma de Google. Thinking for Innovation [página web]. (14, octubre, 2016). [Consultado el 17, noviembre, 2021]. Disponible en Internet: <https://www.iebschool.com/blog/firebase-que-es-para-que-sirve-la-plataforma-desarroladores-google-seo-sem/>.

HABLAME. Hablame Colombia - Voz, Datos y Video. Hablame Colombia [página web]. (2022). [Consultado el 9, junio, 2022]. Disponible en Internet: <https://www.hablame.co/>.

PROYECTOSAGILES. Qué es SCRUM. Proyectos Ágiles [página web]. (2022). [Consultado el 16, septiembre, 2021]. Disponible en Internet: <https://proyectosagiles.org/que-es-scrum/>.

AZUERO AZUERO, Ángel Enrique. Significatividad del marco metodológico en el desarrollo de proyectos de investigación | Azuero Azuero | Revista Arbitrada Interdisciplinaria Koinonía. Fundación Koinonía [página web]. (12, noviembre, 2018). [Consultado el 22, septiembre, 2021]. Disponible en Internet: <https://fundacionkoinonia.com.ve/ojs/index.php/revistakoinonia/article/view/274/html>.

SAMSUNG. ¿Qué es GSM? | Samsung Latinoamérica. ¿Qué es GSM? [página web]. (14, diciembre, 2021). [Consultado el 2, septiembre, 2022]. Disponible en Internet: <https://www.samsung.com/latin/support/mobile-devices/what-is-gsm/>.

HERNANDO RÁBANOS, José María; MENDO TOMÁS, Luis y RIERA SALÍS, José Manuel. COMUNICACIONES MÓVILES. cerasa.es [página web]. (27, julio, 2015). [Consultado el 22, septiembre, 2021]. Disponible en Internet: <https://www.cerasa.es/media/areces/files/book-attachment-2725.pdf>.

ANEXOS

SISTEMA DE COMUNICACIÓN Y MONITOREO PARA GRUPOS DE MOTOVIAJEROS, BASADOS EN GPS Y MENSAJERÍA MÓVIL

ANEXO 1 ARQUITECTURA DE SOFTWARE MOTO VIAJE

Oscar Fernando Espinosa Rocha
Yeison Camilo Rincón Ordoñez

Universidad Distrital Francisco José de Caldas
Facultad Tecnológica
Ingeniería en Telemática
Bogotá, Colombia
2022

TABLA DE CONTENIDO

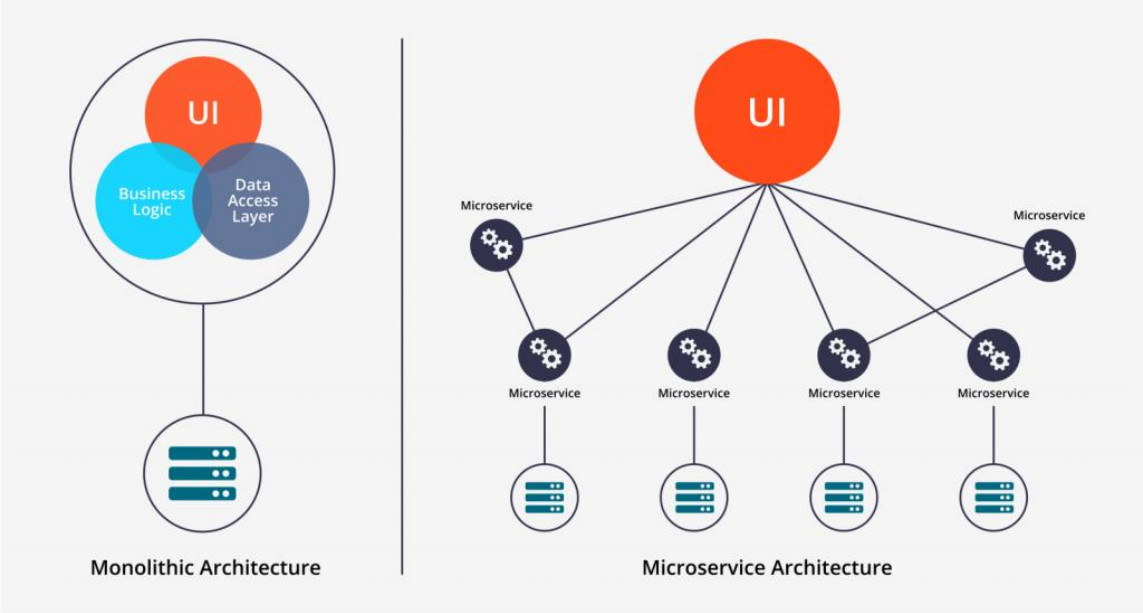
TABLA DE CONTENIDO 2
TABLA DE ILUSTRACIONES 3
1. ARQUITECTURA APLICACIÓN MOTO VIAJE 4
1.1. DIAGRAMA DE COMPONENTES 4
1.2. DIAGRAMA DE SECUENCIA 6
1.3. DIAGRAMA DE PAQUETES 6
1.4. DIAGRAMA DE CLASES 7

TABLA DE ILUSTRACIONES

Ilustración 1: Arquitectura orientada a micro servicios 4
Ilustración 2: Diagrama de componentes 5
Ilustración 3: 1.2 Diagrama de secuencia 6
Ilustración 4: Ejemplo de patrón singleton 7
Ilustración 5: Diagrama de paquetes 7
Ilustración 8: Diagrama de clases 8

1. ARQUITECTURA APLICACIÓN MOTO VIAJE

Para la aplicación moto-viaje se define una arquitectura orientada a micro servicios. Esta arquitectura al ser abierta y escalable nos permite un desarrollo más rápido y una manutención más eficiente de las aplicaciones al separar los modelos de negocio de nuestro sistema.



[1]

Ilustración 1: Arquitectura orientada a micro servicios

1.1. DIAGRAMA DE COMPONENTES

Para el proyecto moto-viaje se definen los siguientes componentes:

- Aplicación móvil
- API
- Base de datos

[1] Hierro, O., Gil , S., & Anguiano, I. (04 de Marzo de 2021). *Blog de Hiberus Tecnología*. Obtenido de Blog de Hiberus Tecnología: https://www.hiberus.com/crecemos-contigo/de-una-arquitectura-tradicional-a-microservicios/

- API´S externos

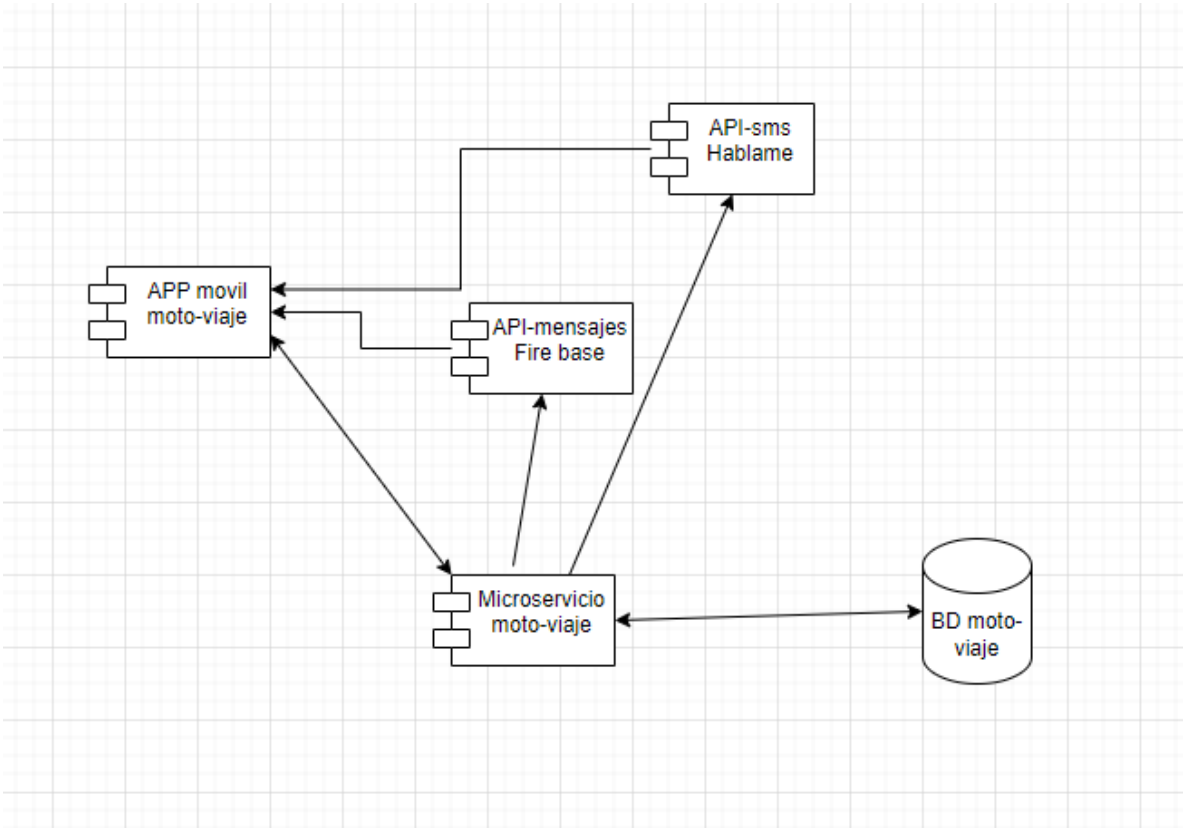


Ilustración 2: Diagrama de componentes

El orquestador de la comunicación entre los diferentes componentes del proyecto es el micro servicio moto-viaje, este se encarga de recibir y enviar los datos a los diferentes actores que intervienen en los procesos.

Consume el api de firebase para enviar los mensajes push de notificación a las aplicaciones móviles instaladas.

Solicita el envía de SMS a háblame en caso de ocurrir un problema para el usuario, consume el api de mensajería y notifica al contacto de emergencia.

Almacena y consulta la base de datos moto-viaje, donde se alojan los datos de la aplicación, tales como, usuarios, grupos, viajes, notificaciones, accesos, e.tc.

1.2. DIAGRAMA DE SECUENCIA

Tenemos una vista técnica a muy alto nivel de la interactividad de la aplicación moto viaje, los componentes que intervienen y la forma en que interactúan entre ellos.

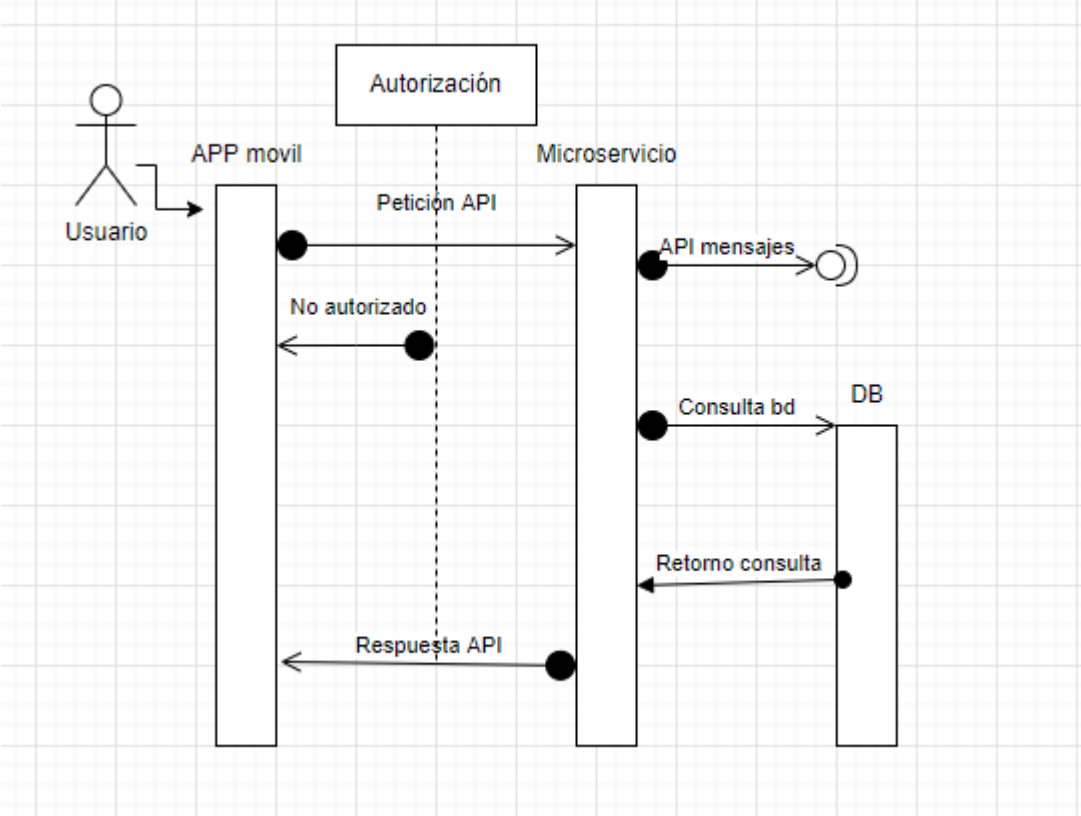
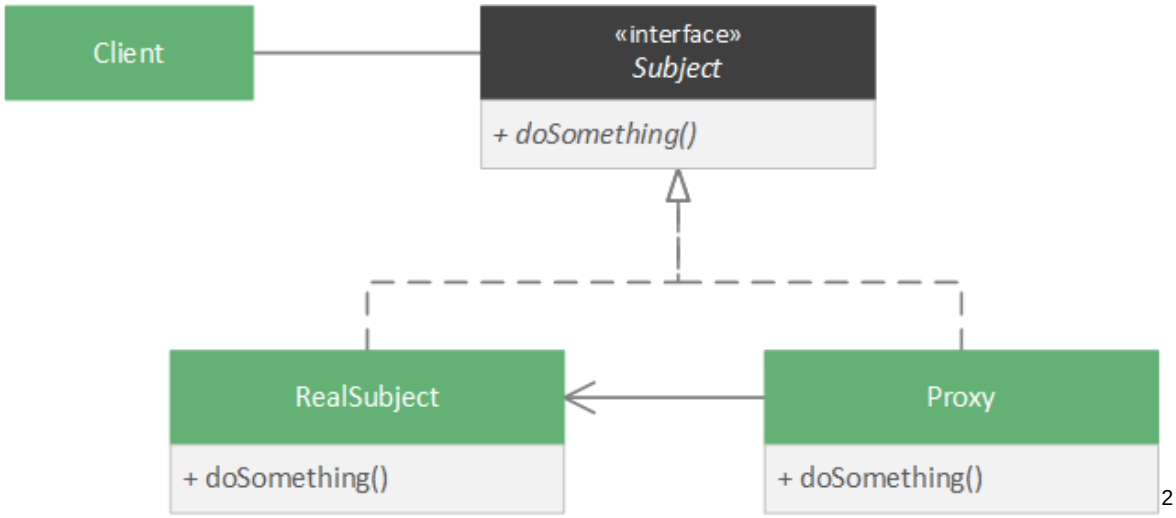


Ilustración 3: 1.2 Diagrama de secuencia

1.3. DIAGRAMA DE PAQUETES

El micro servicio usa un patrón de diseño único (Singleton Design Pattern), que es el recomendado cuando se trabaja con el framework spring boot.

Este patrón indica que las clases solo se instancian una vez y siguen un esquema de jerarquía predeterminado, teniendo como punto de acceso a la clase la instancia global creada.



[2]

Ilustración 4: Ejemplo de patrón singleton

Moto viaje usa este patrón en sus clases, representada en la estructura de sus paquetes así:

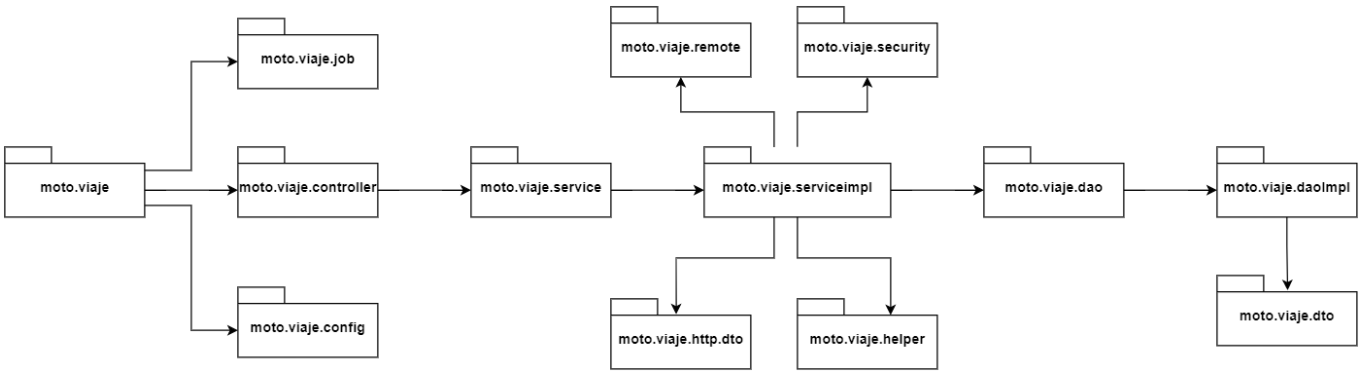


Ilustración 5: Diagrama de paquetes

1.4. DIAGRAMA DE CLASES

Dentro de esta paquetería están contenidas las clases más importantes del componente distribuidas así:

[2] Albano, J. (21 de Abril de 2020). *Baeldung*. Obtenido de Baeldung: https://baeldung-cn.com/spring-framework-design-patterns

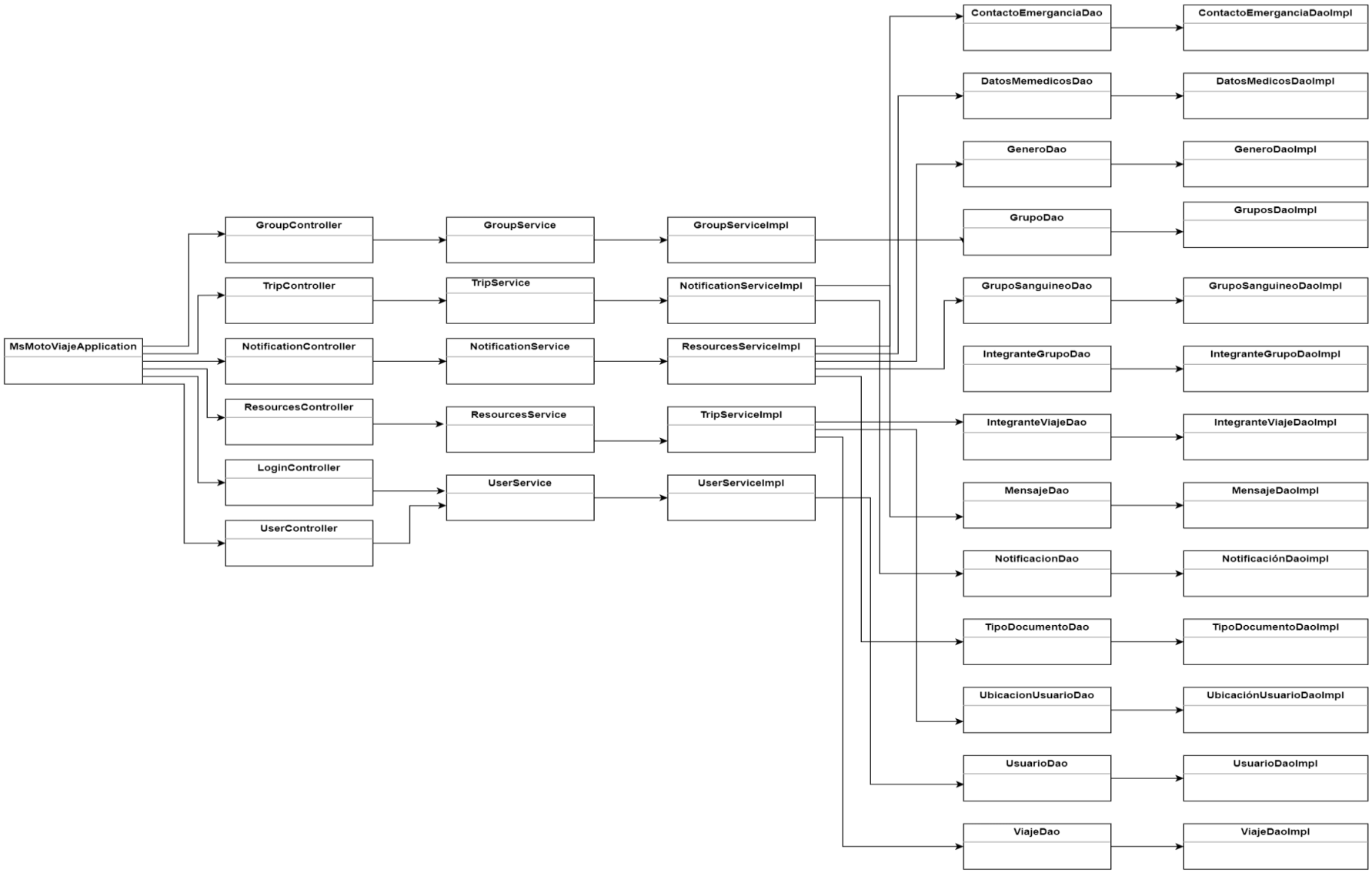


Ilustración 6: Diagrama de clases

SISTEMA DE COMUNICACIÓN Y MONITOREO PARA GRUPOS DE MOTOVIAJEROS, BASADOS EN GPS Y MENSAJERÍA MÓVIL

ANEXO 2 API MOTO VIAJE

Oscar Fernando Espinosa Rocha
Yeison Camilo Rincón Ordoñez

Universidad Distrital Francisco José de Caldas
Facultad Tecnológica
Ingeniería en Telemática
Bogotá, Colombia
2022

TABLA DE CONTENIDO

TABLA DE CONTENIDO 2
TABLA DE ILUSTRACIONES 4
ILUSTRACIÓN DE TABLAS 5
1. DISEÑO BASE DE DATOS 6
1.1 MODELADO DE LA BASE DE DATOS 6
1.1.1 Usuario 6
1.1.2 Contacto de emergencia 6
1.1.3 Datos médicos 6
1.1.4 Genero 7
1.1.5 Grupo sanguíneo 7
1.1.6 Tipo de documento 7
1.1.7 Ubicación 7
1.1.8 Grupo de usuarios 8
1.1.9 Integrantes grupo 8
1.1.10 Viajes grupos 8
1.1.11 Estados de viaje 8
1.1.12 Viajes usuarios 9
1.1.13 Notificaciones 9
1.1.14 Mensajes 9
1.1.15 Tipo mensaje 10
1.2 DIAGRAMA DE BASE DE DATOS 10
2. APLICACIÓN BACK–END 11
2.1 SEGURIDAD DEL API 11
2.1.1 Login en el API 11
2.1.2 Cabecera de token en las peticiones 11
2.1.3 Cifrado de la data de petición y respuesta. 12
2.2 API 12
2.2.1 Login 12
2.2.2 Recursos 14

2.2.3 Grupo 28
2.2.4 Viaje 33
2.2.5 Notificación 43
3. SWAGGER 45

TABLA DE ILUSTRACIONES

Ilustración 1: Diagrama base de datos moto viaje 10
Ilustración 2: Cifrado de datos API 12
Ilustración 3: Cifrado respuesta datos API 12

ILUSTRACIÓN DE TABLAS

Tabla 1: Parámetros login 13
Tabla 2: Parámetros crear usuario 13
Tabla 3: Parámetros editar usuario 18
Tabla 4: Parámetros consultar usuario por tipo y número de identificación 20
Tabla 5: Parámetros consultar usuario por email o usuario 20
Tabla 6: Parámetros consultar usuario por identificador 21
Tabla 7: Parámetros crear contacto de emergencia..... 22
Tabla 8: Parámetros editar contacto de emergencia..... 23
Tabla 9: Parámetros consultar contacto de emergencia por usuario 24
Tabla 10: Parámetros crear datos médicos..... 25
Tabla 11: Parámetros editar datos médicos..... 25
Tabla 12: Parámetros consultar datos médicos por usuario 26
Tabla 13: Parámetros actualizar ubicación usuario..... 27
Tabla 14: Parámetros actualizar contraseña usuario 27
Tabla 15: Parámetros editar identificador de firebase 28
Tabla 16: Parámetros crear grupo..... 29
Tabla 17: Parámetros consultar grupo por identificador..... 30
Tabla 18: Parámetros asociar miembros al grupo..... 30
Tabla 19: Parámetros desasociar miembros..... 31
Tabla 20: Parámetros consultar los grupos asociados y creados por el usuario... 31
Tabla 21: Parámetros consultar los usuarios asociados a un grupo 32
Tabla 22: Parámetros crear viaje 34
Tabla 23: Parámetros editar viaje..... 35
Tabla 24: Parámetros asociar miembros al viaje..... 36
Tabla 25: Parámetros confirmar viaje por usuario..... 37
Tabla 26: Parámetros buscar viajes por destino y usuario..... 37
Tabla 27: Parámetros finalizar el viaje 38
Tabla 28: Parámetros buscar viajes por grupo..... 39
Tabla 29: Parámetros localización de miembros por viaje 39
Tabla 30: Parámetros pausar viaje..... 40
Tabla 31: Parámetros buscar viajes por fechas y usuario..... 40
Tabla 32: Parámetros iniciar viaje 41
Tabla 33: Parámetros estado de miembro del viaje 41
Tabla 34: Parámetros miembros del viaje 42
Tabla 35: Parámetros viajes por usuario..... 43
Tabla 36: Parámetros envió notificación 44

1. DISEÑO BASE DE DATOS

1.1 MODELADO DE LA BASE DE DATOS

Se debe modelar un esquema de base de datos capaz de soportar los datos necesarios para la aplicación moto-viaje.

1.1.1 Usuario

Se debe crear una tabla que soporte los datos del usuario con los campos:

- Identificador
- Nombre
- Apellidos
- Contraseña
- Fecha nacimiento
- Identificador del tipo de documento
- Número de documento
- Identificador del género
- Número de celular
- Verificado
- Activo
- Fecha de creación
- Usuario
- Identificador de Firebase

1.1.2 Contacto de emergencia

Se debe crear una tabla que soporte los datos del contacto de emergencia relacionado al usuario con los campos:

- Identificador
- Nombre
- Apellido
- Teléfono
- Identificador del usuario

1.1.3 Datos médicos

Se debe crear una tabla que soporte los datos médicos relacionados al usuario con los campos:

- Identificador

- Identificador del grupo sanguíneo
- Identificador del usuario
- Prescripciones médicas

1.1.4 Genero

Se debe crear una tabla que soporte los géneros relacionados al usuario con los campos:

- Identificador
- Género
- Prefijo (M (Masculino), F(Femenino), O(Otro))

1.1.5 Grupo sanguíneo

Se debe crear una tabla que soporte los grupos sanguíneos relacionados al usuario con los campos:

- Identificador
- Factor Rh

1.1.6 Tipo de documento

Se debe crear una tabla que soporte los tipos de documentos relacionados al usuario con los campos:

- Identificador
- Nombre
- Código
- Prefijo

1.1.7 Ubicación

Se debe crear una tabla que soporte la ubicación relacionado al usuario con los campos:

- Identificador
- Identificador usuario
- Latitud
- Longitud
- Fecha ubicación

1.1.8 Grupo de usuarios

Se debe crear una tabla que soporte los grupos de usuarios con los campos:

- Identificador
- Nombre
- Fecha creación
- Fecha expiración
- Descripción
- Código grupo
- Identificador usuario creador

1.1.9 Integrantes grupo

Se debe crear una tabla que relacione los usuarios con los grupos, con los campos:

- Identificador
- Identificador usuario
- Identificador grupo
- Fecha ingreso
- Activo
- Fecha salida

1.1.10 Viajes grupos

Se debe crear una tabla que soporte los viajes de los grupos con los campos:

- Identificador
- Identificador grupo
- Destino
- Fecha inicio
- Fecha fin
- Latitud destino
- Longitud destino
- Identificador estado viaje

1.1.11 Estados de viaje

Se debe crear una tabla que soporte los estados del viaje con los campos:

- Identificador
- Estado
- Descripción

1.1.12 Viajes usuarios

Se debe crear una tabla que relacione los usuarios con los viajes, con los campos:

- Identificador
- Identificador viaje
- Identificador usuario
- Activo
- Líder

1.1.13 Notificaciones

Se debe crear una tabla que contenga los datos de las notificaciones enviadas por los usuarios de la aplicación, con los campos:

- Identificador
- Título
- Mensaje
- Latitud
- Longitud
- Identificador usuario
- Identificador viaje
- Data
- Identificador grupo
- Fecha envió
- Enviado
- Identificador mensaje

1.1.14 Mensajes

Se debe crear una tabla que contenga mensajes genéricos que puedan ser enviados por el usuario, con los campos:

- Identificador
- Identificador tipo mensaje
- Mensaje

1.1.15 Tipo mensaje

Se debe crear una tabla que contenga los tipos de mensajes que puedan ser enviados por el usuario, con los campos:

- Identificador
- Tipo

1.2 DIAGRAMA DE BASE DE DATOS

El diagrama resultante de la creación de las tablas es:

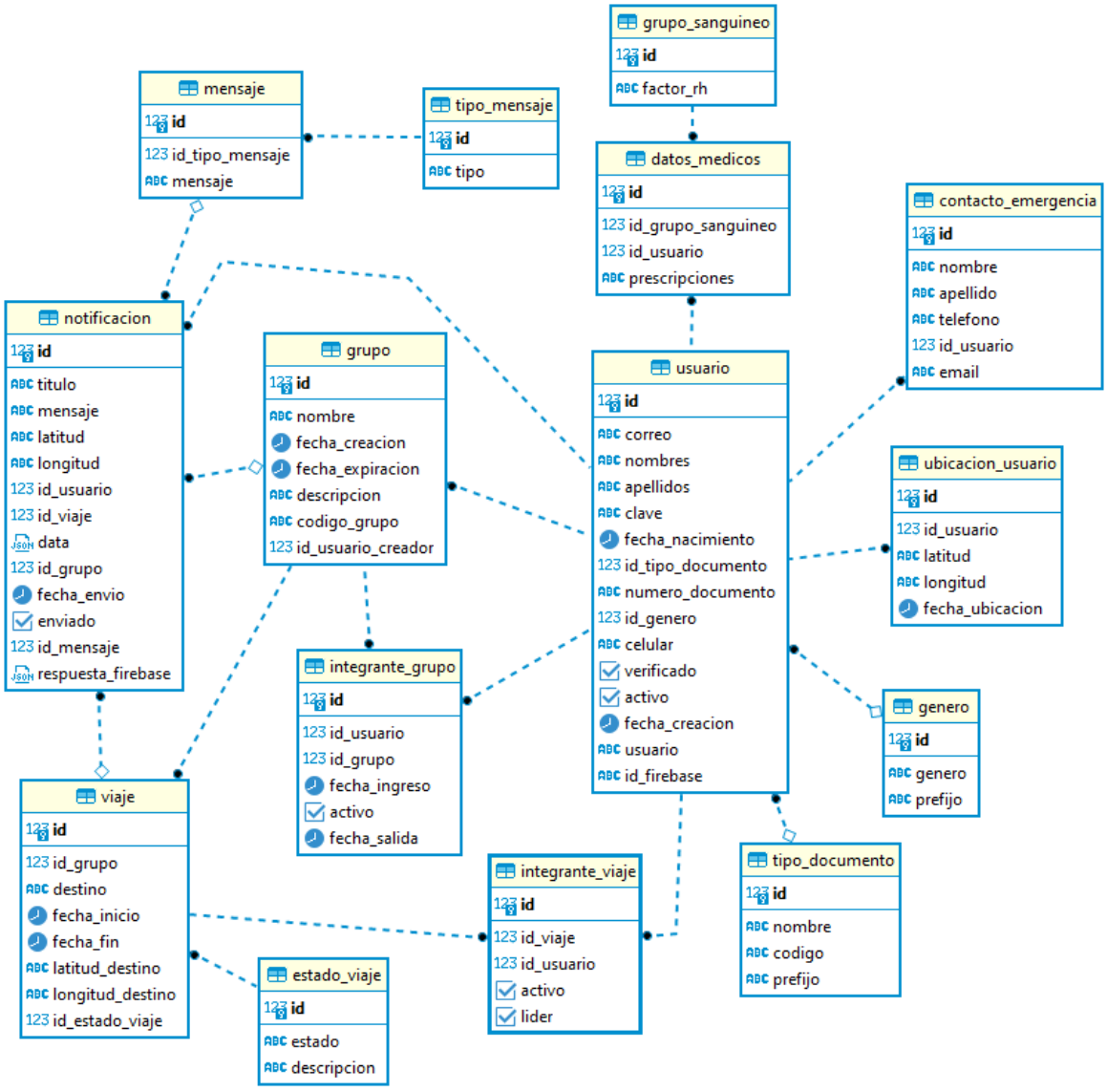


Ilustración 1: Diagrama base de datos moto viaje

2. APLICACIÓN BACK–END

Para la aplicación back-end se utilizaron diferentes tecnologías, que permiten la exposición de un API de consumo, este será invocado por la aplicación móvil y se encargará de la conexión a la base de datos, así como de la lógica de los crud necesarios y demás procesos y funciones.

Las tecnologías usadas se describen a continuación:

- Spring boot
- Java
- Swagger
- JWT
- Java crypto
- Apache tomcat

2.1 SEGURIDAD DEL API

Para la seguridad del API, se implementan tres factores que impiden el acceso no autorizado a la información, protegiendo así los datos de los usuarios adscritos a la aplicación móvil y asegurando la integridad y la confidencialidad de los mismos.

2.1.1 Login en el API

Como primer factor de seguridad se implementa por medio de la librería de JWT un login en el API; Solo los usuarios registrados en la aplicación móvil podrán acceder con sus credenciales, usando su usuario y su contraseña, la cual también se almacena en la base de datos cifrada para que no sea visible ni entendible.

2.1.2 Cabecera de token en las peticiones

En el segundo factor se solicita una cabecera por cada petición excepto en el de login y registro de usuarios, esta contendrá un token generado como respuesta del método de logueo del usuario, al enviarlo correctamente el API entregara los datos solicitados.

2.1.3 Cifrado de la data de petición y respuesta.

El tercer factor cifra los datos tanto de consumo como de respuesta en una cadena de tipo Base64. Para poder ver el contenido de esta se debe tener una llave privada que solo comparten el API y la aplicación móvil, haciendo imposible interpretar el contenido de los mensajes compartidos entre los dos puntos de comunicación.

Las peticiones enviadas serán del tipo:

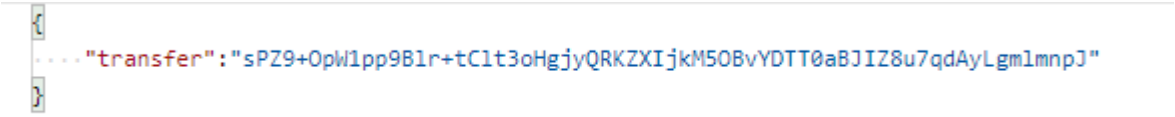


Ilustración 2: Cifrado de datos API

Y las respuestas:


Body
Cookies
Headers (14)
Test Results
Status: 200 OK
Time: 1489 ms
Size: 826 B
Save Response
Pretty
Raw
Preview
Visualize
JSON
1
IR2DmIjb1D0Qw1XOudpxE3iK1G/+NwMi2M%/m94OhqY+0t8PgcS0Ea8ea2e/7dEaud6GhOdVQpiBrL6EihCFEPrWoTbGlniFXCi6TxwGPO2fHySwvACIOmW9/oY+qjgSJXWF/ERisG2XeKJpbD12erGVI3u+fLqjk1s7xLQxKUZIpxqJ9WP5Iu9LHH27163u7KAYk/U00C2ncMAR
+Qd11f9YpVeF3z3x0hnpKY3FTOgHKqmPDvGSmVF/E8+ZPatxqvBFk8ovzvqyME+GJLeb2K8gCPeGz2FwtF6QEyyTad+Hpq+At1bIC0g1LUn1xFkWycyPe9UeC97g7R8OxE0A2xLp91m0d9iIqhEiBnjfJEW27/f5+rxM5Qs7z9tmCcZ8

Ilustración 3: Cifrado respuesta datos API

2.2 API

Se exponen los siguientes métodos contenidos en la API de moto-viaje

2.2.1 Login

2.2.1.1 Login

URL: http://{ip}:{puerto}/moto-viaje/login
Método: POST

Permite el login de los usuarios de la aplicación, si es exitoso generara el token para el consumo de los otros métodos expuestos.

Parámetro	Tipo	Descripción
password	Alfanumérico	Contraseña del usuario
username	Alfanumérico	Nombre del usuario

Tabla 1: Parámetros login

Petición:

```
{
  "password": "string",
  "username": "string"
}
```

Respuesta:

```
{
  "token": "string",
  "bearer": "Bearer",
  "username": "string",
  "autoAuthorities": [

  ]
}
```

2.2.1.2 Crear usuario

URL: http://{ip}:{puerto}/moto-viaje/login/create-user
Método: PUT

Permite crear un usuario nuevo en la aplicación, responde con los datos del usuario creado.

Parámetro	Tipo	Descripción
apellidos	Alfabético	Apellidos del usuario del usuario
clave	Alfanumérico	Contraseña del usuario
correo	Alfanumérico	Correo electrónico del usuario
nombres	Alfabético	Nombres del usuario
usuario	Alfanumérico	Usuario logueado

Tabla 2: Parámetros crear usuario

Petición:


```
{
  "apellidos": "string",
  "clave": "string",
  "correo": "string",
  "nombres": "string",
  "usuario": "string"
}
```


Respuesta:


```
{
  "activo": true,
  "apellidos": "string",
  "celular": "string",
  "clave": "string",
  "correo": "string",
  "fechaCreacion": "string",
  "fechaNacimiento": "string",
  "id": 0,
  "idFirebase": "string",
  "idGenero": 0,
  "idTipoDocumento": 0,
  "nombres": "string",
  "numeroDocumento": "string",
  "usuario": "string",
  "verificado": true
}
```


2.2.2 Recursos

Conjunto de métodos que exponen listas y elementos necesarios para la aplicación.

2.2.2.1 Tipo de sangre

URL: http://{ip}:{puerto}/moto-viaje/resources/blood-type
Método: GET

Regresa el listado de tipos de sangre.

Respuesta:

```
[
        {
          "factorRh": "string",
          "id": 0,
        }
]
```

2.2.2.2 Tipo de sangre por id

URL: http://{ip}:{puerto}/moto-viaje/resources/blood-type/{id}
Método: GET

Regresa el tipo de sangre por su identificador.

Respuesta:

```
        {
            "factorRh": "string",
             "id": 0,
        }
```

2.2.2.3 Tipos de documento

URL: http://{ip}:{puerto}/moto-viaje/resources/document-type
Método: GET

Regresan los tipos de documento configurados en el sistema

Respuesta:

```
[
        {
          "codigo": "string",
          "id": 0,
          "nombre": "string",
          "prefijo": "string"
        }
]
```

2.2.2.4 Tipo de documento por id

URL: http://{ip}:{puerto}/moto-viaje/resources/document-type/{id}
Método: GET

Regresa un tipo de documento por su identificador

Respuesta:

```
{
  "codigo": "string",
  "id": 0,
  "nombre": "string",
  "prefijo": "string"
}
```

2.2.2.5 Género

URL: http://{ip}:{puerto}/moto-viaje/resources/gender
Método: GET

Regresan los géneros configurados en el sistema.

Respuesta:

```
[
  {
    "genero": "string",
    "id": 0,
    "prefijo": "string"
  }
]
```

2.2.2.6 Género por id

URL: http://{ip}:{puerto}/moto-viaje/resources/gender/{id}
Método: GET

Regresa un objeto de género por su identificador

Respuesta:

```
{
    "genero": "string",
    "id": 0,
    "prefijo": "string"
}
```

2.2.2.7 Mensajes

URL: http://{ip}:{puerto}/moto-viaje/resources/message
Método: GET

Regresa un listado de mensajes genéricos para las notificaciones

Respuesta:

```
[
    {
      "codigo": "string",
      "id": 0,
      "nombre": "string",
      "prefijo": "string"
    }
]
```

2.2.2.8 Estado viaje

URL: http://{ip}:{puerto}/moto-viaje/resources/travel-status
Método: GET

Regresa el listado de estados de viaje configurados en el sistema

Respuesta:

```
[
    {
      "codigo": "string",
      "id": 0,
      "nombre": "string",
      "prefijo": "string"
    }
```

]

2.2.2.9 Usuario

Conjunto de métodos que contiene la lógica de la administración de los usuarios de la aplicación.

2.2.2.10 Editar usuario

URL: http://{ip}:{puerto}/moto-viaje/user
Método: PATCH

Permite editar los datos del usuario.

Parámetro	**Tipo**	**Descripción**
activo	booleano	Indica si el usuario se encuentra activo
apellidos	Alfabético	Apellidos del usuario del usuario
celular	Numérico	Contraseña del usuario
correo	Alfanumérico	Correo electrónico del usuario
fechaCreacion	Alfanumérico	Fecha de creación del usuario
fechaNacimiento	Alfanumérico	Fecha de nacimiento del usuario. Formato(dd-MM-yyyy)
id	Numérico	Id del usuario
idFirebase	Alfanumérico	Id de firebase del usuario
idGenero	Numérico	Id del genero del usuario
idTipoDocumento	numérico	Id del tipo de identificación del usuario
nombres	Alfabético	Nombres del usuario
numeroDocumento	Alfanumérico	Número de identificación del usuario
usuario	Alfanumérico	Usuario logueado
verificado	booleano	Indica si el usuario ya fue verificado

Tabla 3: Parámetros editar usuario

Petición:

```
{
  "activo": true,
  "apellidos": "string",
  "celular": "string",
```

```
  "clave": "string",
  "correo": "string",
  "fechaCreacion": "string",
  "fechaNacimiento": "string",
  "id": 0,
  "idFirebase": "string",
  "idGenero": 0,
  "idTipoDocumento": 0,
  "nombres": "string",
  "numeroDocumento": "string",
  "usuario": "string",
  "verificado": true
}
```


Respuesta:


```
{
  "activo": true,
  "apellidos": "string",
  "celular": "string",
  "clave": "string",
  "correo": "string",
  "fechaCreacion": "string",
  "fechaNacimiento": "string",
  "id": 0,
  "idFirebase": "string",
  "idGenero": 0,
  "idTipoDocumento": 0,
  "nombres": "string",
  "numeroDocumento": "string",
  "usuario": "string",
  "verificado": true
}
```


2.2.2.11 Consultar usuario por tipo y número de identificación

URL: http://{ip}:{puerto}/moto-viaje/user/{identificationType}/{identification}
Método: GET

Consulta un usuario por tipo y número de identificación

Parámetro	Tipo	Descripción
identificationType	Numérico	Identificador del tipo de documento
identification	Alfanumérico	Número de identificación

Tabla 4: Parámetros consultar usuario por tipo y número de identificación

Respuesta:


```
{
  "activo": true,
  "apellidos": "string",
  "celular": "string",
  "clave": "string",
  "correo": "string",
  "fechaCreacion": "string",
  "fechaNacimiento": "string",
  "id": 0,
  "idFirebase": "string",
  "idGenero": 0,
  "idTipoDocumento": 0,
  "nombres": "string",
  "numeroDocumento": "string",
  "usuario": "string",
  "verificado": true
}
```


2.2.2.12 Consultar usuario por email o usuario

URL: http://{ip}:{puerto}/moto-viaje/user/{user}
Método: GET

Consulta un usuario por correo electrónico o por usuario de aplicación

Parámetro	Tipo	Descripción
user	Alfanumérico	Email o usuario de la aplicación

Tabla 5: Parámetros consultar usuario por email o usuario

Respuesta:


```
{
  "activo": true,
  "apellidos": "string",
  "celular": "string",
  "clave": "string",
  "correo": "string",
  "fechaCreacion": "string",
  "fechaNacimiento": "string",
  "id": 0,
  "idFirebase": "string",
  "idGenero": 0,
  "idTipoDocumento": 0,
  "nombres": "string",
  "numeroDocumento": "string",
  "usuario": "string",
  "verificado": true
}
```


2.2.2.13 Consultar usuario por identificador

URL: http://{ip}:{puerto}/moto-viaje/user/id/{id}
Método: GET

Consulta el usuario por su identificador

Parámetro	Tipo	Descripción
id	Numérico	Identificador del usuario

Tabla 6: Parámetros consultar usuario por identificador

Respuesta:


```
{
        "activo": true,
        "apellidos": "string",
        "celular": "string",
        "clave": "string",
        "correo": "string",
        "fechaCreacion": "string",
        "fechaNacimiento": "string",
        "id": 0,
        "idFirebase": "string",
```

```
    "idGenero": 0,
    "idTipoDocumento": 0,
    "nombres": "string",
    "numeroDocumento": "string",
    "usuario": "string",
    "verificado": true
}
```


2.2.2.14 Crear Contacto de emergencia

URL: http://{ip}:{puerto}/moto-viaje/user/emergency-contact
Método: PUT

Crea un contacto de emergencia asociado al usuario de la aplicación.

Parámetro	Tipo	Descripción
apellido	Alfabético	Apellido del contacto de emergencia
idUsuario	Numérico	Identificador del usuario al cual está relacionado el contacto
nombre	Alfabético	Nombre del contacto de emergencia
teléfono	Numérico	Número de teléfono del contacto de emergencia
email	Alfanumérico	Correo electrónico del contacto de emergencia

Tabla 7: Parámetros crear contacto de emergencia

Petición:


```
{
  "apellido": "string",
  "idUsuario": 0,
  "nombre": "string",
  "telefono": "string",
  "email": "string"
}
```

Respuesta:


```
{
	"apellido": "string",
	"id": 0,
	"idUsuario": 0,
	"nombre": "string",
	"telefono": "string",
	"email": "string"
}
```


2.2.2.15 Editar contacto de emergencia

URL: http://{ip}:{puerto}/moto-viaje/user/emergency-contact
Método: PATCH

Permite editar un contacto de emergencia.

Parámetro	**Tipo**	**Descripción**
apellido	Alfabético	Apellido del contacto de emergencia
idUsuario	Numérico	Identificador del usuario al cual está relacionado el contacto
nombre	Alfabético	Nombre del contacto de emergencia
teléfono	Numérico	Número de teléfono del contacto de emergencia
id	Numérico	Identificador del contacto de emergencia

Tabla 8: Parámetros editar contacto de emergencia

Petición:


```
{
	"apellido": "string",
	"id": 0,
	"idUsuario": 0,
	"nombre": "string",
	"telefono": "string"
}
```

Respuesta:

```
{
        "id": 0,
        "idGrupoSanguineo": 0,
        "idUsuario": 0,
        "prescripciones": "string"
}
```

2.2.2.16 Consultar contacto de emergencia por usuario

URL: http://{ip}:{puerto}/moto-viaje/user/emergency-contact/{IdUser}
Método: GET

Consulta el contacto de emergencia asociado al usuario.

Parámetro	Tipo	Descripción
idUser	Numérico	Identificador del usuario

Tabla 9: Parámetros consultar contacto de emergencia por usuario

Respuesta:

```
{
  "apellido": "string",
  "id": 0,
  "idUsuario": 0,
  "nombre": "string",
  "telefono": "string"
}
```

2.2.2.17 Crear datos médicos

URL: http://{ip}:{puerto}/moto-viaje/user/medical-data
Método: PUT

Crea datos médicos asociados al usuario de la aplicación.

Parámetro	Tipo	Descripción
idGrupoSanguineo	Numérico	Identificador del grupo sanguíneo
idUsuario	Numérico	Identificador del usuario
prescripciones	Alfanumérico	Descripción de las prescripciones médicas si las tuviese

Tabla 10: Parámetros crear datos médicos

Petición:

```
{
  "idGrupoSanguineo": 0,
  "idUsuario": 0,
  "prescripciones": "string",
}
```

Respuesta:

```
{
  "id": 0,
  "idGrupoSanguineo": 0,
  "idUsuario": 0,
  "prescripciones": "string"
}
```

2.2.2.18 Editar datos médicos

URL: http://{ip}:{puerto}/moto-viaje/user/medical-data
Método: PATCH

Edita el registro de datos médicos asociados al usuario de la aplicación

Parámetro	Tipo	Descripción
idGrupoSanguineo	Numérico	Identificador del grupo sanguíneo
idUsuario	Numérico	Identificador del usuario
prescripciones	Alfanumérico	Descripción de las prescripciones médicas si las tuviese
id	Numérico	Identificador del dato medico

Tabla 11: Parámetros editar datos médicos

Petición:


```
{
  "id": 0,
  "idGrupoSanguineo": 0,
  "idUsuario": 0,
  "prescripciones": "string"
}
```


Respuesta:


```
{
  "id": 0,
  "idGrupoSanguineo": 0,
  "idUsuario": 0,
  "prescripciones": "string"
}
```


2.2.2.19 Consultar datos médicos por usuario

URL: http://{api}:{puerto}/moto-viaje/user/medical-data/{id}
Método: GET

Extrae los datos médicos por el usuario de la aplicación

Parámetro	Tipo	Descripción
idUser	Numérico	Identificador del usuario

Tabla 12: Parámetros consultar datos médicos por usuario

Respuesta:


```
{
  "id": 0,
  "idGrupoSanguineo": 0,
  "idUsuario": 0,
  "prescripciones": "string"
}
```


2.2.2.20 Actualizar ubicación usuario

URL: http://{ip}:{puerto}/moto-viaje/user/update-location
Método: PUT

Actualiza la ubicación geográfica del usuario de la aplicación

Parámetro	Tipo	Descripción
idUsusario	Numérico	Identificador del usuario
latitud		Latitud de la ubicación geográfica del usuario
longitud		Longitud de la ubicación geográfica del usuario

Tabla 13: Parámetros actualizar ubicación usuario

Petición:

```
{
  "idUsusario": 0,
  "latitud": "string",
  "longitud": "string"
}
```

Respuesta:

Booleano

2.2.2.21 Actualizar contraseña usuario

URL: http://{ip}:{puerto}/moto-viaje/user/update-password
Método: PATCH

Actualiza la contraseña del usuario de la aplicación

Parámetro	Tipo	Descripción
id	Numérico	Identificador del usuario de la aplicación
pass	Alfanumérico	Nueva contraseña

Tabla 14: Parámetros actualizar contraseña usuario

Petición:

```
{
  "id": 0,
  "pass": "string"
}
```

Respuesta:

```
{
  "mensaje": "string",
  "realizado": true
}
```

2.2.2.22 Editar identificador de firebase

URL: http://{ip}:{puerto}/moto-viaje/user/user-id/{idUser}
Método: PATCH

Método que edita el identificador de firebase de la aplicación

Parámetro	Tipo	Descripción
idUser	Numérico	Identificador del usuario de la aplicación
idFirebase	Alfanumérico	Identificador asignado por firebase a la aplicación

Tabla 15: Parámetros editar identificador de firebase

Petición:

```
{
  "idFirebase": "string"
}
```

Respuesta:

Booleano

2.2.3 Grupo

Crea y administra las propiedades y elementos del grupo de viaje

2.2.3.1 Crear grupo

URL: http://{ip}:{puerto}/moto-viaje/group
Método: PUT

Crea un grupo en la aplicación

Parámetro	Tipo	Descripción
descripcion	Alfabético	Descripción general del grupo
fechaExpiracion	Alfabético	Fecha en la que expira el grupo, esta debe ir en Formato(dd-MM-yyyy)
idUsuarioCreador	Numérico	Identificador del usuario creador del grupo
nombre	Alfabético	Nombre del grupo

Tabla 16: Parámetros crear grupo

Petición:

```
{
  "codigoGrupo": "string",
  "descripcion": "string",
  "fechaCreacion": "string",
  "fechaExpiracion": "string",
  "id": 0,
  "idUsuarioCreador": 0,
  "nombre": "string"
}
```

Respuesta:

```
{
  "codigoGrupo": "string",
  "descripcion": "string",
  "fechaCreacion": "string",
  "fechaExpiracion": "string",
  "id": 0,
  "idUsuarioCreador": 0,
  "nombre": "string"
}
```

2.2.3.2 Consultar grupo por identificador

URL: http://{ip}:{puerto}/moto-viaje/group/{idGroup}
Método: GET

Consulta los datos de un viaje por su identificador

Parámetro	Tipo	Descripción
idGroup	Numérico	Identificador del grupo

Tabla 17: Parámetros consultar grupo por identificador

Respuesta:

```
{
  "codigoGrupo": "string",
  "descripcion": "string",
  "fechaCreacion": "string",
  "fechaExpiracion": "string",
  "id": 0,
  "idUsuarioCreador": 0,
  "nombre": "string",
}
```

2.2.3.3 Asociar miembros al grupo

URL: http://{ip}:{puerto}/moto-viaje/group/associate-members
Método: POST

Asocia miembros al grupo creado

Parámetro	Tipo	Descripción
codigoGrupo	Alfanumérico	Código único asignado a cada grupo
idsUsuario	Numérico	Lista de identificaciones de los usuarios a asociar al grupo

Tabla 18: Parámetros asociar miembros al grupo

Petición:

```
{
  "codigoGrupo": "string",
  "idsUsuario": [
     0
    ]
}
```

Respuesta:

Booleano

2.2.3.4 Desasociar miembros

URL: http://{ip}:{puerto}/moto-viaje/group/disassociate-member
Método: POST

Desasocia los miembros del grupo

Parámetro	Tipo	Descripción
codigoGrupo	Alfanumérico	Código único asignado a cada grupo
idsUsuario	Numérico	Lista de identificaciones de los usuarios a asociar al grupo

Tabla 19: Parámetros desasociar miembros

Petición:

```
{
  "codigoGrupo": "string",
  "idsUsuario": [
    0
  ]
}
```

Respuesta:

Booleano

2.2.3.5 Consultar los grupos asociados y creados por el usuario

URL: http://{ip}:{puerto}/moto-viaje/group/user-groups/{IdUser}
Método: GET

Consulta los grupos asociados y/o creados por el usuario

Parámetro	Tipo	Descripción
idUser	Numérico	Identificador del usuario

Tabla 20: Parámetros consultar los grupos asociados y creados por el usuario

Respuesta:


```
{
  "asociados": [
    {
      "codigoGrupo": "string",
      "descripcion": "string",
      "fechaCreacion": "string",
      "fechaExpiracion": "string",
      "id": 0,
      "idUsuarioCreador": 0,
      "nombre": "string"
    }
  ],
  "creados": [
    {
      "codigoGrupo": "string",
      "descripcion": "string",
      "fechaCreacion": "string",
      "fechaExpiracion": "string",
      "id": 0,
      "idUsuarioCreador": 0,
      "nombre": "string"
    }
  ]
}
```


2.2.3.6 Consultar los usuarios asociados a un grupo

URL: http://{ip}:{puerto}/moto-viaje/group/users-group/{IdUser}
Método: GET

Consulta los usuarios asociados a un grupo

Parámetro	**Tipo**	**Descripción**
idGroup	Numérico	Identificador del grupo

Tabla 21: Parámetros consultar los usuarios asociados a un grupo

Respuesta:


```
[
        {
```

```
    "activo": true,
    "apellidos": "string",
    "celular": "string",
    "clave": "string",
    "correo": "string",
    "fechaCreacion": "string",
    "fechaNacimiento": "string",
    "id": 0,
    "idFirebase": "string",
    "idGenero": 0,
    "idTipoDocumento": 0,
    "nombres": "string",
    "numeroDocumento": "string",
    "usuario": "string",
    "verificado": true
  }
]
```


2.2.4 Viaje

Crea y administra las propiedades y elementos del viaje

2.2.4.1 Crear viaje

URL: http://{ip}:{puerto}/moto-viaje/trip
Método: PUT

Crea un viaje asociado a un grupo

Parámetro	**Tipo**	**Descripción**
destino	Alfabético	Destino del viaje
fechaFin	Alfabético	Fecha fin del viaje. Formato(dd-MM-yyyy)
fechaInicio	Alfabético	Fecha de inicio del viaje. Formato(dd-MM-yyyy)
idGrupo	Numérico	Identificador del grupo asociado al viaje
idUsuarioCreador	Numérico	Identificador del usuario creador del viaje

latitudDestino	Alfanumérico	Latitud geográfica destino del viaje
lider	Booleano	Indica si el usuario es líder
longitudDestino	Alfanumérico	Longitud geográfica destino del viaje

Tabla 22: Parámetros crear viaje

Petición:


```
{
  "destino": "string",
  "fechaFin": "string",
  "fechaInicio": "string",
  "idGrupo": 0,
  "idUsuarioCreador": 0,
  "latitudDestino": "string",
  "lider": true,
  "longitudDestino": "string"
}
```


Respuesta:


```
{
  "destino": "string",
  "fechaFin": "string",
  "fechaInicio": "string",
  "id": 0,
  "idEstadoViaje": 0,
  "idGrupo": 0,
  "idUsuarioCreador": 0,
  "latitudDestino": "string",
  "lider": true,
  "longitudDestino": "string"
}
```


2.2.4.2 Editar viaje

URL: http://{ip}:{puerto}/moto-viaje/trip
Método: PATCH

Edita un viaje creado

Parámetro	Tipo	Descripción
destino	Alfabético	Destino del viaje
fechaFin	Alfabético	Fecha fin del viaje. Formato(dd-MM-yyyy)
fechaInicio	Alfabético	Fecha de inicio del viaje. Formato(dd-MM-yyyy)
idGrupo	Numérico	Identificador del grupo asociado al viaje
idUsuarioCreador	Numérico	Identificador del usuario creador del viaje
latitudDestino	Alfanumérico	Latitud geográfica destino del viaje
lider	Booleano	Indica si el usuario es lider
longitudDestino	Alfanumérico	Longitud geográfica destino del viaje
id	Numérico	Identificador del viaje
idEstadoViaje	Numérico	Identificador del estado del viaje

Tabla 23: Parámetros editar viaje

Petición:


```
{
  "destino": "string",
  "fechaFin": "string",
  "fechaInicio": "string",
  "id": 0,
  "idEstadoViaje": 0,
  "idGrupo": 0,
  "idUsuarioCreador": 0,
  "latitudDestino": "string",
  "lider": true,
  "longitudDestino": "string"
}
```


Respuesta:


```
{
  "destino": "string",
  "fechaFin": "string",
  "fechaInicio": "string",
  "id": 0,
  "idEstadoViaje": 0,
  "idGrupo": 0,
```

```
  "idUsuarioCreador": 0,
  "latitudDestino": "string",
  "lider": true,
  "longitudDestino": "string"
}
```


2.2.4.3 Asociar miembros al viaje

URL: http://{ip}:{puerto}/moto-viaje/trip/associate-trip-member
Método: POST
Asocia miembros al viaje

Parámetro	Tipo	Descripción
idViaje	Numérico	Identificador del viaje
miembroViaje	Lista de objetos	Lista de objetos de datos del usuario asociado
idUsuario	Numérico	Identificador del usuario
lider	Booleano	Indica si el usuario es líder

Tabla 24: Parámetros asociar miembros al viaje

Petición:


```
{
  "idViaje": 0,
  "miembroViaje": [
    {
      "idUsuario": 0,
      "lider": true
    }
  ]
}
```


Respuesta:
Booleano

2.2.4.4 Confirmar viaje por usuario

URL: http://{ip}:{puerto}/moto-viaje/trip/confirm-trip
Método: PUT

El usuario asociado al viaje confirma que viaje desea iniciar o continuar según sus necesidades

Parámetro	Tipo	Descripción
confirmedTrip	Objeto	Objeto de viajes a confirmar
confirmed	Booleano	Indica si el viaje es confirmado
idViaje	Numérico	Identificador del viaje
idUser	Numérico	identificador del usuario

Tabla 25: Parámetros confirmar viaje por usuario

Petición:

```
{
  "confirmedTrip": [
    {
      "confirmed": true,
      "idViaje": 0
    }
  ],
  "idUser": 0
}
```

Respuesta:

Boolean

2.2.4.5 Buscar viajes por destino y usuario

URL: http://{ip}:{puerto}/moto-viaje/trip/destination/{destino}/user-id/{idUser}
Método: GET

Busca los viajes a un destino específico asociados a un usuario

Parámetro	Tipo	Descripción
destino	Alfanumérico	Destino que se desea buscar
idUser	Numérico	Identificador del usuario

Tabla 26: Parámetros buscar viajes por destino y usuario

Respuesta:

```
[
{
  "destino": "string",
  "fechaFin": "string",
  "fechaInicio": "string",
  "id": 0,
  "idEstadoViaje": 0,
  "idGrupo": 0,
  "idUsuarioCreador": 0,
  "latitudDestino": "string",
  "lider": true,
  "longitudDestino": "string"
}
]
```


2.2.4.6 Finalizar el viaje

URL: http://{ip}:{puerto}/moto-viaje/trip/finish-trip/{idViaje}/{idUsuario}
Método: PATCH

Termina un viaje iniciado

Parámetro	**Tipo**	**Descripción**
idViaje	Numérico	Identificador del viaje
idUsuario	Numérico	Identificador del usuario

Tabla 27: Parámetros finalizar el viaje

Respuesta:
Booleano

2.2.4.7 Buscar viajes por grupo

URL: http://{ip}:{puerto}/moto-viaje/trip/group-id/{idGrupo}
Método: GET

Busca los viajes por grupo

Parámetro	Tipo	Descripción
idGrupo	Numérico	Identificador del grupo

Tabla 28: Parámetros buscar viajes por grupo

Respuesta:

Booleano

2.2.4.8 Localización de miembros por viaje

URL: http://{ip}:{puerto}/moto-viaje/trip/members-location/{idViaje}
Método: GET

Consulta la localización de los miembros del viaje

Parámetro	Tipo	Descripción
IdViaje	Numérico	Identificador del viaje

Tabla 29: Parámetros localización de miembros por viaje

Respuesta:

```
[
{
  "apellidos": "string",
  "fechaUbicacion": "string",
  "idUsuario": 0,
  "latitud": "string",
  "longitud": "string",
  "nombres": "string"
}
]
```

2.2.4.9 Pausar viaje

URL: http://{ip}:{puerto}/moto-viaje /trip/pause-trip/{idViaje}/{idUsuario}
Método: PATCH

Parámetro	Tipo	Descripción
idViaje	Numérico	Identificador del viaje
idUsuario	Numérico	Identificador del usuario

Tabla 30: Parámetros pausar viaje

Respuesta:


```
[
{
  "destino": "string",
  "fechaFin": "string",
  "fechaInicio": "string",
  "id": 0,
  "idEstadoViaje": 0,
  "idGrupo": 0,
  "idUsuarioCreador": 0,
  "latitudDestino": "string",
  "lider": true,
  "longitudDestino": "string"
}
]
```


2.2.4.10 Buscar viajes por fechas y usuario

URL: http://{ip}:{puerto}/moto-viaje/trip/start-date/{fechaIn}/enddate/{fechaF}/user-id/{idUser}
Método: GET

Busca viajes pro su fecha inicio, fecha fin y usuario

Parámetro	Tipo	Descripción
fechaIn	Alfanumérico	Fecha inicio viaje, Formato (dd-MM-yyyy)
fechaF	Alfanumérico	Fecha fin viaje, Formato (dd-MM-yyyy)
idUser	Numérico	Identificador del usuario

Tabla 31: Parámetros buscar viajes por fechas y usuario

Respuesta:


```
[
{
  "destino": "string",
  "fechaFin": "string",
```

```
  "fechaInicio": "string",
  "id": 0,
  "idEstadoViaje": 0,
  "idGrupo": 0,
  "idUsuarioCreador": 0,
  "latitudDestino": "string",
  "lider": true,
  "longitudDestino": "string",
  "transfer": "string"
}
]
```


2.2.4.11 Iniciar viaje

URL: http://{ip}:{puerto}/moto-viaje/trip/start-trip/{idViaje}/{idUsuario}
Método: PATCH

Inicia un viaje

Parámetro	Tipo	Descripción
idViaje	Numérico	Identificador del viaje
idUsuario	Numérico	Identificador del usuario

Tabla 32: Parámetros iniciar viaje

Respuesta:
Booleano

2.2.4.12 Estado de miembro del viaje

URL: http://{ip}:{puerto}/moto-viaje/trip/state-member-trip
Método: PATCH

Cambia el estado y el indicador de líder del miembro del viaje

Parámetro	Tipo	Descripción
activo	Booleano	Indica si el usuario está activo o inactivo
idUsuario	Numérico	Identificador del usuario
idViaje	Numérico	Identificador del viaje
lider	Booleano	Indicia si el usuario es líder

Tabla 33: Parámetros estado de miembro del viaje

Petición:


```
{
  "activo": true,
  "idUsuario": 0,
  "idViaje": 0,
  "lider": true
}
```


Respuesta:

Booleano

2.2.4.13 Miembros del viaje

URL: http://{ip}:{puerto}/moto-viaje/trip/trip-member/{idViaje}/{activo}
Método: GET

Consulta miembros del viaje por estado

Parámetro	Tipo	Descripción
idViaje	Numérico	Identificador del viaje
activo	Booleano	Indica si el usuario está activo o inactivo

Tabla 34: Parámetros miembros del viaje

Respuesta:


```
[
{
  "apellidos": "string",
  "fechaUbicacion": "string",
  "idUsuario": 0,
  "latitud": "string",
  "longitud": "string",
  "nombres": "string",
  "transfer": "string"
}
]
```

2.2.4.14 Viajes por usuario

URL: http://{ip}:{puerto}/moto-viaje/trip/user?status={status}&user-id={user-id}
Método: GET

Consulta los viajes por usuario y estado del viaje

Parámetro	Tipo	Descripción
user-id	Numérico	Identificador del usuario
status	Alfabético	Indica el estado del viaje

Tabla 35: Parámetros viajes por usuario

Respuesta:


```
[
{
  "destino": "string",
  "fechaFin": "string",
  "fechaInicio": "string",
  "id": 0,
  "idEstadoViaje": 0,
  "idGrupo": 0,
  "idUsuarioCreador": 0,
  "latitudDestino": "string",
  "lider": true,
  "longitudDestino": "string"
}
]
```


2.2.5 Notificación

Genera las notificaciones push del sistema y la aplicación

2.2.5.1 Envió notificación

URL: http://{ip}:{puerto}/moto-viaje/notification/send
Método: POST

Envía los mensajes y notificaciones entre usuarios

Parámetro	Tipo	Descripción
data	Objeto	Parámetros extra a enviar
additionalProp	Alfanumérico	Parámetro extra enviado
enViaje	Booleano	Indica si el usuario está en viaje
enviado	Booleano	Indica si el mensaje fue enviado
fechaEnvio	Alfanumérico	Fecha den envio, formato (dd-MM-yyyy)
idGrupo	Numérico	Identificador del grupo
idMensaje	Numérico	Identificador del mensaje genérico
idUsuario	Numérico	Identificador del usuario
idViaje	Numérico	Identificador del viaje
latitud	Alfanumérico	Latitud desde donde se envía el mensaje
longitud	Alfanumérico	Longitud desde donde se envía el mensaje
mensaje	Alfanumérico	Texto del mensaje
titulo	Alfanumérico	Título del mensaje

Tabla 36: Parámetros envió notificación

Petición:


```
{
  "data": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "enViaje": true,
  "enviado": true,
  "fechaEnvio": "string",
  "idGrupo": 0,
  "idMensaje": 0,
  "idUsuario": 0,
  "idViaje": 0,
  "latitud": "string",
  "longitud": "string",
  "mensaje": "string",
  "titulo": "string",
}
```

Respuesta:


```
{
  "data": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "enViaje": true,
  "enviado": true,
  "fechaEnvio": "string",
  "id": 0,
  "idGrupo": 0,
  "idMensaje": 0,
  "idUsuario": 0,
  "idViaje": 0,
  "latitud": "string",
  "longitud": "string",
  "mensaje": "string",
  "titulo": "string",
  "transfer": "string"
}
```


3. SWAGGER

Toda las API´S contenidas en el documento, están expuestas de forma visual y pueden ser consultadas desde un navegador web usando la ruta especificada para swagger que nos provee el micro servicio.

Al ser desplegado se puede acceder desde una ruta similar a:

http://{ip}:{puerto}/moto-viaje/swagger-ui.html

SISTEMA DE COMUNICACIÓN Y MONITOREO PARA GRUPOS DE MOTOVIAJEROS, BASADOS EN GPS Y MENSAJERÍA MÓVIL

ANEXO 3 MANUAL DE INSTALACIÓN MOTOVIAJE

Oscar Fernando Espinosa Rocha
Yeison Camilo Rincón Ordoñez

Universidad Distrital Francisco José de Caldas
Facultad Tecnológica
Ingeniería en Telemática
Bogotá, Colombia
2022

TABLA DE CONTENIDO

TABLA DE CONTENIDO 2
TABLA DE ILUSTRACIONES 3
ILUSTRACIÓN DE TABLAS 4
1. MANUAL DE INSTALACIÓN DE LA APP MOTO VIAJE 5
1.1. BASE DE DATOS 5
1.1.1. Requerimientos técnicos 5
1.1.2. Instalación 5
1.2. MICROSERVICIO 7
1.2.1. Requerimientos técnicos 7
1.2.2. Compilación 7
1.2.3. Instalación 10
1.3. APLICACIÓN MOVIL 12
1.3.1. Requerimientos técnicos 12
1.3.2 Instalación de la aplicación 12

TABLA DE ILUSTRACIONES

Ilustración 1: Creación base de datos 6
Ilustración 2: Paso 1 restauración base de datos 6
Ilustración 3: Paso 2 restauración base de datos 7
Ilustración 4: Configuración micro servicio 8
Ilustración 5: Compilación micro servicio 9
Ilustración 6: Resultado compilación micro servicio 10
Ilustración 7: Vista apache tomcat 10
Ilustración 8: Instalación apache tomcat 11
Ilustración 9: Swagger 11
Ilustración 10: Descarga APK 12
Ilustración 11: Instalación APK 13
Ilustración 12: Permisos APK 14
Ilustración 13: Fin instalación APK 14

ILUSTRACIÓN DE TABLAS

Tabla 1: Versión 5

1. MANUAL DE INSTALACIÓN DE LA APP MOTO VIAJE

Versión	Autor-fecha	Notas
V:1.0	Yeison Camilo Rincon-Oscar Fernando Espinosa 29/07/2022	Versión inicial del manual de instalación

Tabla 1: Versión

El presente manual detalla los pasos necesarios para poder realizar la instalación de la aplicación moto viaje.

A continuación se especificarán los mínimos requerimientos para el correcto funcionamiento de la aplicación.

1.1. BASE DE DATOS

1.1.1. Requerimientos técnicos

1.1.1.1. Requerimientos de hardware

- Servidor de aplicaciones

1.1.1.2. Requerimientos de software

- Sistema operativo: Linux o Windows server
- Motor de base de datos PostgreSQL

1.1.2. Instalación

Se debe instalar la copia de seguridad “motoviajedb.backup”, en el motor de base de datos así:

- Se debe crear la base de datos en el motor:

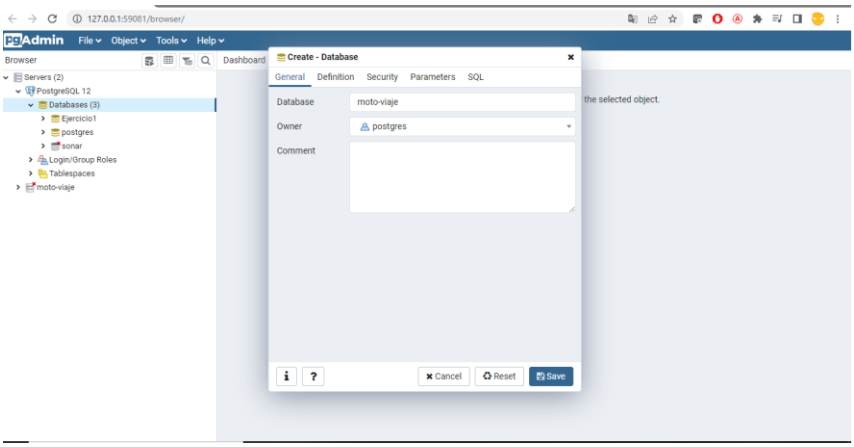


Ilustración 1: Creación base de datos

- Restaurar la base de datos

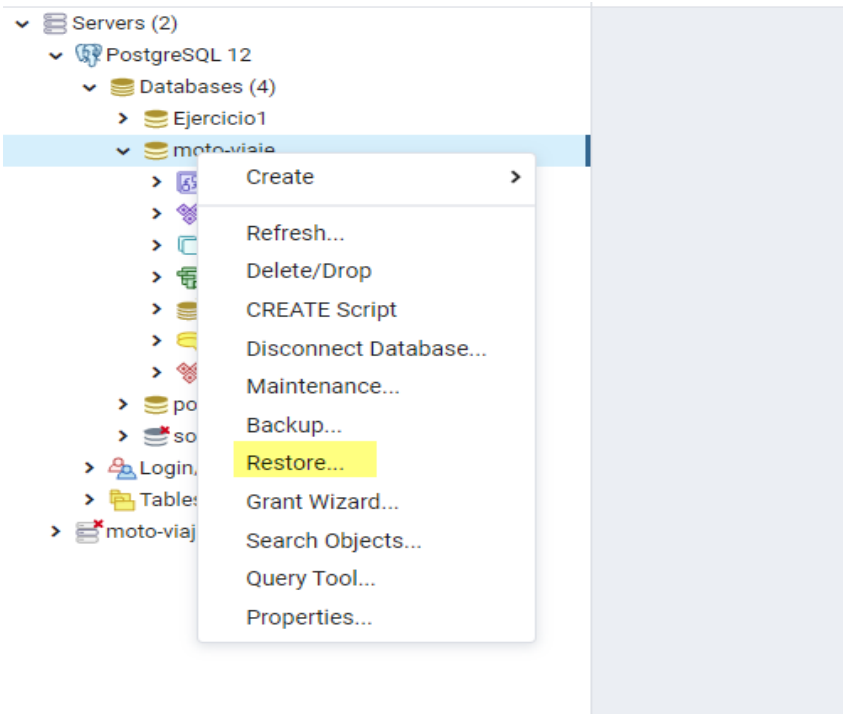


Ilustración 2: Paso 1 restauración base de datos

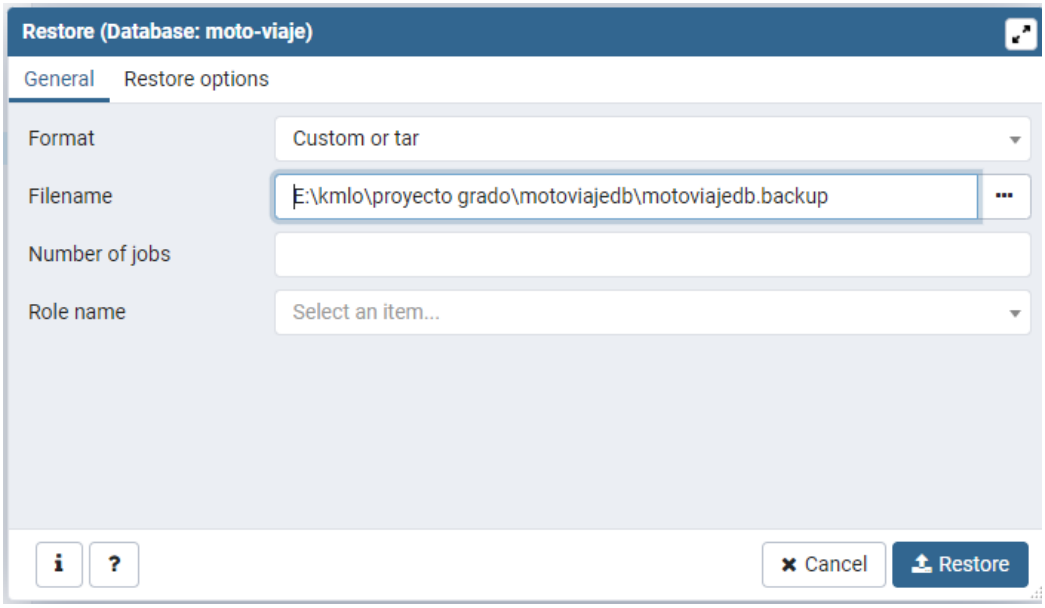


Ilustración 3: Paso 2 restauración base de datos

1.2. MICROSERVICIO

1.2.1. Requerimientos técnicos

1.2.1.1. Requerimientos de hardware

- Servidor de aplicaciones
- Equipo de cómputo para compilación

1.2.1.2. Requerimientos de software

- Sistema operativo: Linux o Windows server
- Spring tools
- Servidor Apache tomcat 9
- Apache Maven
- JDK 1.8 o posterior

1.2.2. Compilación

Para compilar la aplicación es necesario tener Maven instalado en el equipo,
Abrir el código en la herramienta spring tools importada como un proyecto Maven.

- Configurar los recursos:

Se debe configurar el acceso a la base de datos, el correo electrónico creado como emisor de las notificaciones, la contraseña de las trasferencias de datos de los cuerpos de las respuestas, y en el cliente de mensajería también se deben configurar sus credenciales de accesos.



Ilustración 4: Configuración micro servicio

- Compilar:

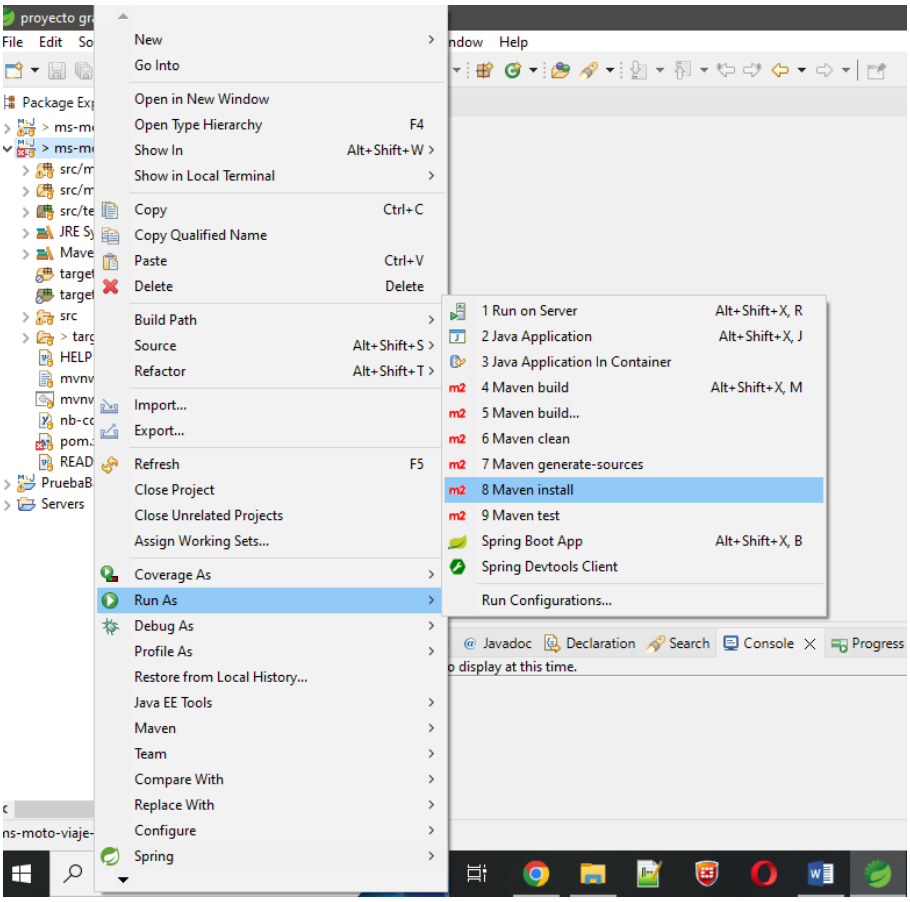


Ilustración 5: Compilación micro servicio

- Resultado de la compilación:

Se genera un archivo .war que es el que se va a desplegar en el servidor tomcat y expone los servicios.

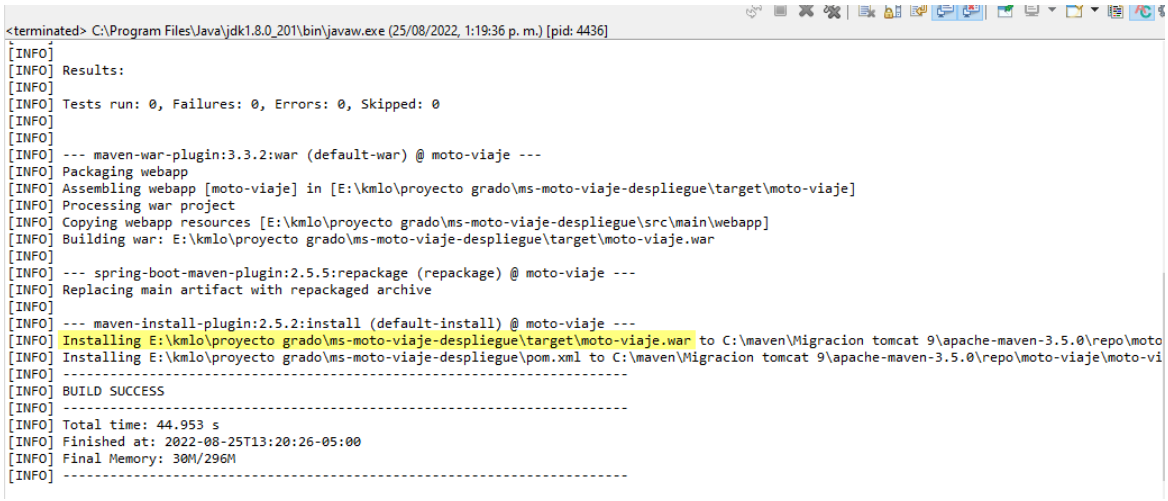


Ilustración 6: Resultado compilación micro servicio

1.2.3. Instalación

- Se debe abrir la consola del servidor tomcat y elegir la opción Archivo WAR a desplegar.

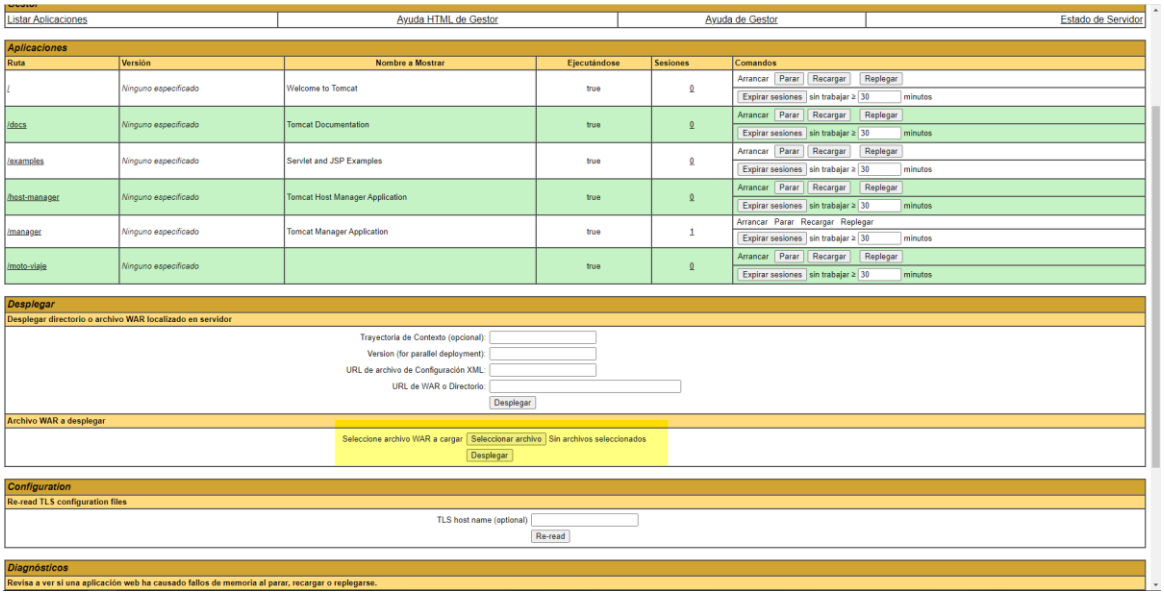


Ilustración 7: Vista apache tomcat

- Al desplegar el WAR quedará una instancia así:

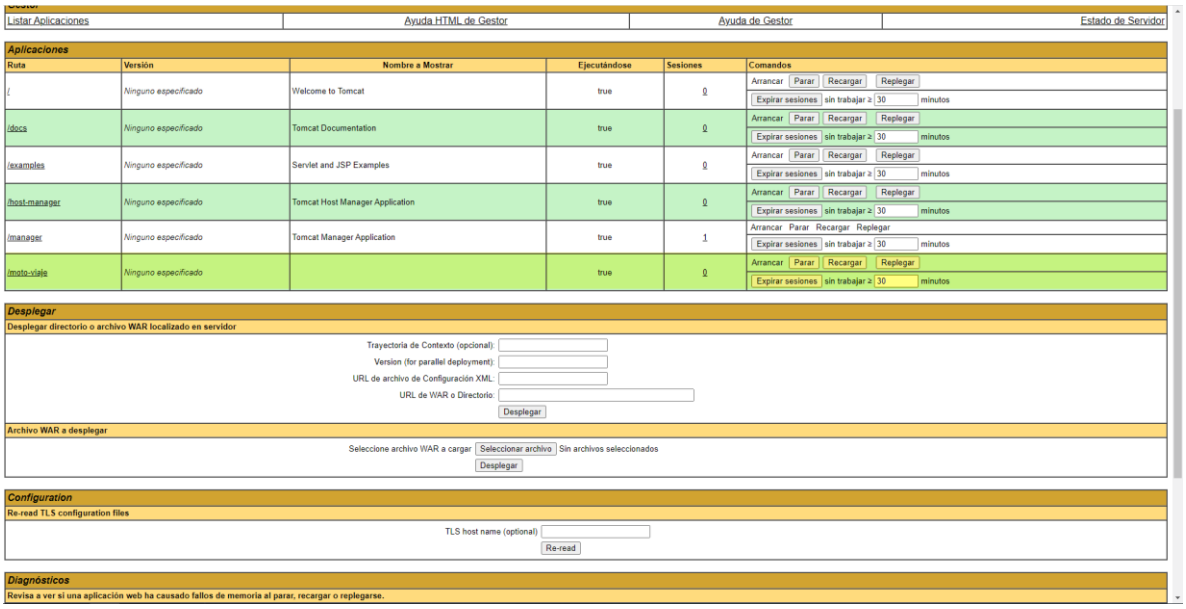


Ilustración 8: Instalación apache tomcat

- Después de realizada la instalación se podrá acceder al swagger de la aplicación, mostrando los servicios desplegados y su respectiva documentación:

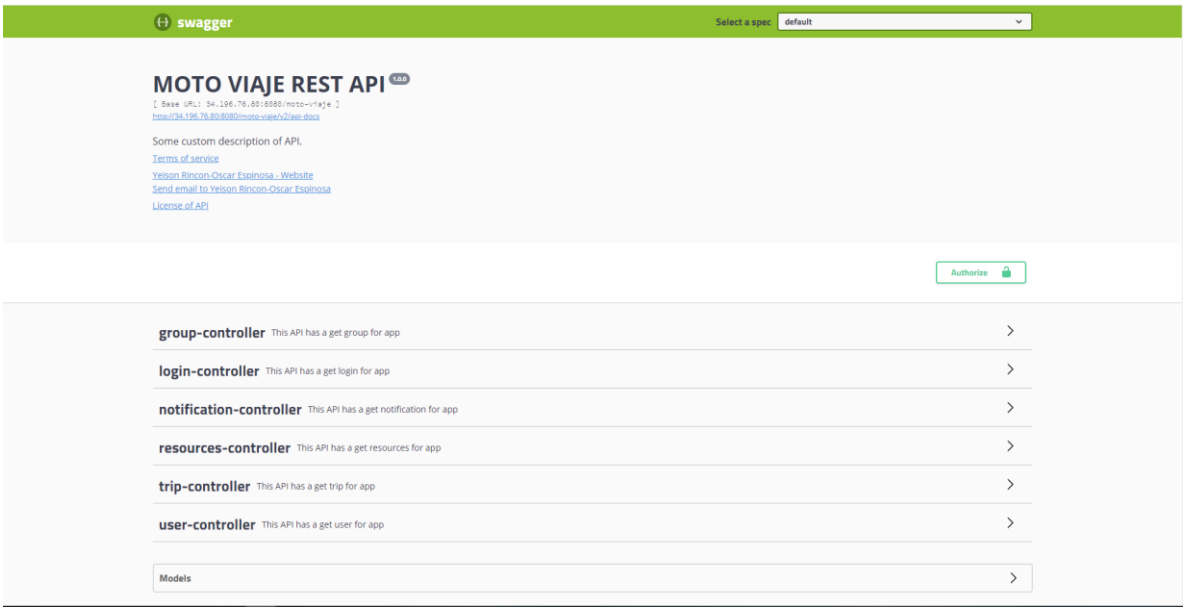


Ilustración 9: Swagger

1.3. APLICACIÓN MOVIL

1.3.1. Requerimientos técnicos

1.3.1.1. Requerimientos de hardware

- Memoria RAM: 1GB o superior.
- Procesador: CPU Quad Core 1.2GHz RAM (GB) 2 GB

1.3.1.2. Requerimientos de software

- Sistema operativo: Android 5.0 o superior

1.3.2 Instalación de la aplicación

1.3.2.1 Descarga de la aplicación

- Descargar el archivo moto-viaje.apk en el dispositivo móvil.
  Se puede descargar de la siguiente URL:
  https://github.com/moto-viaje/app-moto-viaje/blob/main/apk/moto-viaje.apk



Ilustración 10: Descarga APK

1.3.2.2 Instalar la aplicación:

- Navegar hasta el directorio donde se descargó el archivo y abrirlo.

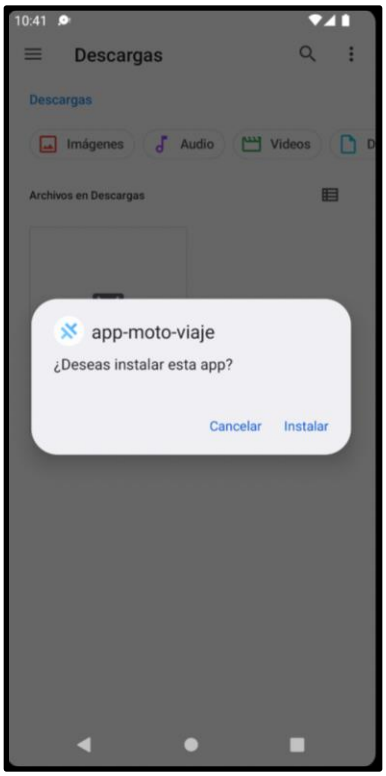


Ilustración 11: Instalación APK

1.3.2.3 Permitir fuentes desconocidas:

- En algunos dispositivos, por cuestiones de seguridad, se debe autorizar la instalación de aplicaciones de fuentes desconocidas.

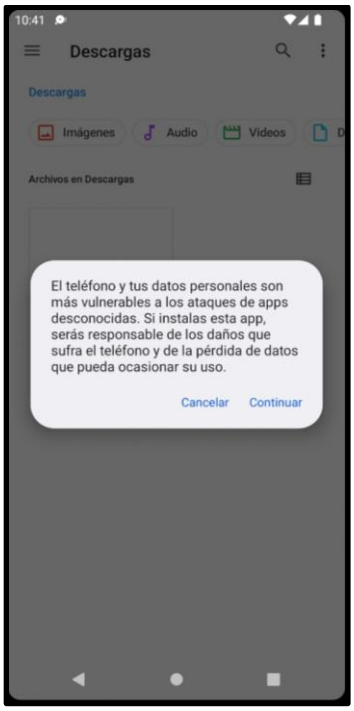


Ilustración 12: Permisos APK

1.3.2.4 Fin de la instalación:

- La aplicación ya está lista para ser usada.

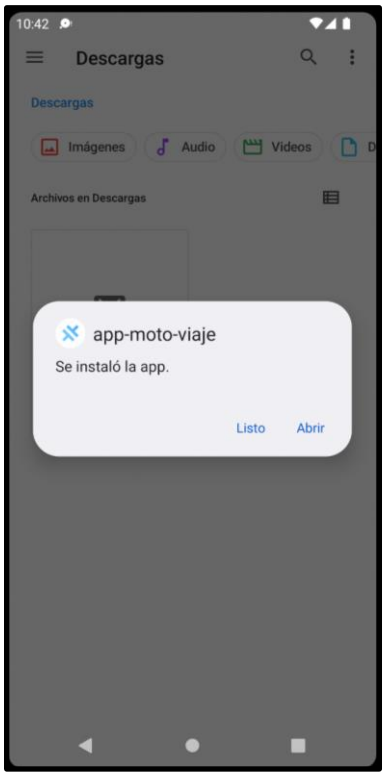


Ilustración 13: Fin instalación APK