INSTITUTO POLITÉCNICO NACIONAL

Imagen que contiene señal, reloj, firmar

Descripción generada automáticamente

**INSTITUTO POLITÉCNICO NACIONAL**

**ESCUELA SUPERIOR DE CÓMPUTO**

**ESCOM**

*Trabajo Terminal*

**“Prototipo de aplicación móvil para la asistencia Informativa a migrantes en tránsito en México Con algoritmos de machine learning y base de datos”**

2025-B153

***Presentan***

Dehesa Nieves Angel,

Mendoza Bedolla Daniela,

Michaus Gutierrez Sabrina

*Directores*

***M. en C. Ortiz Castillo Marco Antonio , Doc. Rodrigez Sarabia Tania***Icono

Descripción generada automáticamente

**RESUMEN**

El flujo migratorio en la frontera sur de México hacia la frontera de Estados Unidos continúa en aumento esta problemática, es fundamental desarrollar soluciones tecnológicas que ofrezcan a los migrantes una guía de rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad, conectándonos con ONG y guiándonos a través de un algoritmo de rutas que los conecte con otro centro de ayuda humanitaria. Este prototipo de aplicación destaca los recursos específicos de las organizaciones, como albergues y comedores, los cuales podrán ser evaluados por los usuarios. La herramienta integrará una base de datos y un sistema que genere rutas y recomendaciones basado en *Machine Learning*, abordando preocupaciones como la inseguridad y el acceso a servicios esenciales.

**Palabras clave –** *Machine Learning*, Migración, Ruta, Base de datos

**Advertencia**

*“Este documento contiene información desarrollada por la Escuela Superior de Cómputo del Instituto Politécnico Nacional, a partir de datos y documentos con derecho de propiedad y por lo tanto, su uso quedará restringido a las aplicaciones que explícitamente se convengan.”*

*La aplicación no convenida exime a la escuela de su responsabilidad técnica y da lugar a las consecuencias legales que para tal efecto se determinen.*

*Información adicional sobre este reporte técnico podrá obtenerse en:*

*La Subdirección Académica de la Escuela Superior de Cómputo del Instituto Politécnico Nacional, situada en Av. Juan de Dios Bátiz s/n Teléfono: 57296000, extensión 52000.*

# Índice

[**Índice 4**](#)

[**Abreviaturas 9**](#)

[**1. Capítulo 1 Fundamentos de la Investigación 11**](#)

[1.1. Introducción 11](#_oi3yp4q55pkb)

[1.2. Planteamiento del problema 12](#_gthitp1hguj3)

[1.3. Propuesta de solución 12](#_j62fqkaw7p1u)

[1.4. Objetivo general 12](#_ecyhhakq64qt)

[**1.4.1. Objetivos específicos 13**](#)

[1.5. Justificación 13](#_tfq5oapfgop8)

[1.6. Organización del documento 14](#_qku3khy9imqn)

[**Capítulo 2. Estado del arte 15**](#)

[2.1 Machine Learning para modelar la migración humana 15](#_gspjrgnro2in)

[2.2 Algoritmo para calcular la ruta más segura y óptima 16](#_lltlvzjicoyw)

[2.3 Mapping Nyumbani 17](#_j9xgtu934sca)

[2.4 Identificación de zonas de vulnerabilidad a la violencia contra mujeres 18](#_o065c28u1rez)

[**Capítulo 3. Marco Teórico 19**](#)

[3.1Migración 19](#_xyef2f99hr9b)

[3.2 Índice de Criminalidad 19](#_tu3599suptx)

[3.3 Rutas 19](#_kyje52sz7h9m)

[3.4 Algoritmos 19](#_mrntgb12h20w)

[3.5 Inteligencia Artificial 19](#_37m8dl27hmv6)

[3.6 Machine Learning 20](#_qndbb52a2iol)

[3.6.1 Aprendizaje Supervisado 20](#_bxk8xhlgwwk9)

[3.6.2 Aprendizaje no supervisado 21](#_ezqvwhok3bli)

[3.6.4 Clusters 22](#_5hm0psxaczkx)

[K-Means Clustering 22](#_1td5yea72qtm)

[DBSCAN (Density-Based Spatial Clustering) 23](#_ec85n5ean7wb)

[Clustering Jerárquico 24](#_o7mmevjiu4ye)

[Algoritmos de rutas. 25](#_2p2zlbyahp3q)

[3.7 Algoritmo de A\* 25](#_kus2wai1u8py)

[Algoritmo Dijkstra 26](#_pf2ynvxxrtqf)

[3.7 Base de Datos 27](#_j58nh2w5xei1)

[3.7.1 Tipos de bases de datos 27](#_dm89ijwidt2s)

[3.7.1.1 Relacional 27](#_koeuttrc409d)

[3.7.1.2 No Relacional 28](#_qn5uymxwtk4j)

[3.8 Plataforma de desarrollo 29](#_wa2pti18p0st)

[3.8.1 Tipos de plataformas 30](#_wrcs3zqvrtik)

[3.8.2 App móvil 31](#_e2z7onfrl4hm)

[3.8.2.1 Como se construye 31](#_z18dp5lx6a6t)

[3.8.2.2 Fases de desarollo 31](#_k6g5s9bzcosk)

[3.8.2.3 Metodología para aplicaciones móviles. 32](#_o6skn7vhiq8w)

[3.8.4 Api 32](#_9sa95lcrqgfp)

[**Capítulo 4. Análisis 34**](#)

[**4.2 Reglas de negocio 34**](#)

[4.3 Requerimientos 35](#_67gckmp4wtx1)

[4.3.1 Requerimientos funcionales 35](#_e8vidvgz4lu)

[4.3.2 Requerimientos no funcionales 36](#_4wdf343ho6c)

[**Capítulo 5. Diseño 37**](#_g6wutiyx9d2l)

[5.1 Diagrama de cliente-servidor 37](#_xxbquz13y2bd)

[5.1.1 Diagrama de actividades 38](#_6nfvgacr6b5v)

[5.2 Casos de uso 39](#_n94q5kil1ezc)

[5.2.1 Diagrama de Casos de uso 39](#_12iw9xt5fuz5)

[**5.2.2 Actores 39**](#)

[5.2.3 Especificación de Casos de Uso 40](#_30j0zll)

[Fuentes de datos 54](#_4y167agqfj76)

[Datos de Criminalidad 54](#_f5c88grmzlxy)

[Datos de ONG’ s 56](#_wckbx0yzrp1l)

[Procesamiento de datos 57](#_mmeslw58oaqb)

[5.4 Modelo de datos 61](#_qguwkknupaka)

[5.4.1 Tabla Comparativa de Bases de Datos 63](#_la2nxiumgnrw)

[5.4.2 Ventajas y desventajas de las Bases de Datos 64](#_3w05x7qgabei)

[Comparativa de lenguajes de programación 67](#_jx5upjkz293s)

[Comparativa de algoritmos de rutas 68](#_t3zp76jvpfe3)

[Algoritmos de Agrupamiento 70](#_2k7zqxf09iiv)

[5.3 Desarrollo de la arquitectura del Prototipo de App Móvil 71](#_s2hcuk5tmkx9)

[.5.3.3 Diagrama de Secuencia de Prototipo de Aplicación Móvil 71](#_onqyvlg3gngp)

[5.3.5 Tabla Comparativa de Diseño de app 77](#_2kl8slctb9vy)

[5.3.6 Conexión para prototipo 77](#_n0yqjiixxzd1)

[5.3.7 Apis para prototipo 78](#_j8azbianlkg5)

[5.3.8 Mecanismo de actualización y visualización de api 79](#_cakvuzx5ii54)

[5.3.8 Mockups 80](#_j7928u6jxxms)

[**Capítulo 6. Desarrollo e Implementación 88**](#)

[6.1 Análisis de factibilidad 88](#_cepiq382wk7s)

[6.1.1 Factibilidad técnica 88](#_gpi41mq2q81u)

[6.2 Factibilidad Operativa 89](#_kak7fxua9y0f)

[6.3 Factibilidad Económica 91](#_29zx7772k75d)

[6.4 Riesgos 92](#_rjgnmfw29inv)

[**Capítulo 8. Conclusiones y trabajo a futuro 95**](#)

[**Capítulo 9. Referencias 96**](#)

##### Índice de tablas

[Tabla 2.4.1 Tabla comparativa 18](#_gx5vbxeau2dn)

[Tabla 5.2.1 Caso de Uso CU1: Crear Cuenta 40](#_8zxp61h2ipwt)

[Tabla 5.2.1 Caso de Uso CU1: Trayectorias. 41](#_muimw250gueh)

[Tabla 5.2.2 Caso de Uso CU2: Login 42](#_z4airkh5t48g)

[Tabla 5.2.2 Caso de Uso CU2: Trayectorias 42](#_85ubsxwz0f5q)

[Tabla 5.2.3 Caso de Uso CU3: Visualizar Mapas 43](#_x05s5jbg0ugw)

[Tabla 5.2.3 Caso de Uso CU3: Trayectorias 44](#_drzm017epnh2)

[Tabla 5.2.4 Caso de Uso CU4: Visualizar información de la ONG 44](#_ocse84i7pxjf)

[Tabla 5.2.4 Caso de Uso CU4: Trayectorias 45](#_eitxa1ruw0tt)

[Tabla 5.2.5 Caso de Uso CU5: Visualizar rutas 46](#_4fz88t609ln5)

[Tabla 5.2.5 Caso de Uso CU5: Trayectoria 46](#_xc5v5ytd30gq)

[Tabla 5.2.6 Caso de Uso CU6: Visualizar ONG’s en el sistema 47](#_6td5v3gln1qf)

[Tabla 5.2.6 Caso de Uso CU6: Trayectorias 47](#_peidjgvx8r3o)

[Tabla 5.2.8 Caso de Uso CU7: Actualización de ruta y servicios BD](#_xga0a9myijcm) 48

[Tabla 5.2.8 Caso de Uso CU7: Trayectorias](#_6zw5uj4i1vok) 48

###### Índice de figuras

[Figura 3.2 Machine Learning en la industria (Machine Learning En La Industria, 2025) 20](#_5b1bpicsdz9l)

[Figura 3.6.1.1 Modelo de aprendizaje Supervisado (Luna, 2018) 20](#_vv4zukuv76ab)

[Figura 3.6.1.2 Modelo de aprendizaje No Supervisado (Luna, 2018) 21](#_3wcedzsrgvq0)

[Figura 3.6.1.3.1 Diagrama de aprendizaje reforzado (Luna, 2018) 22](#_py80xgrb5n4h)

[Figura 3.7.1.1 Tablas de Bases de datos relacionales (Olivera, 2022) 25](#_oetmejhfprd3)

[Figura 3.7.1.2 Base de datos no relacional (Castelo, 2023) 26](#_jq2uphfaeeuw)

[Figura 3.8 Tipos de Plataformas (Nunez, 2024) 27](#_18t18o9e0vel)

[Figura 4.1 Diagrama de Componentes de la Arquitectura del sistema 36](#_5du7aq69wtkx)

[Figura 4.1.3.1 Diagrama de Casos de Uso 38](#_edukd9h6hc0m)

[Figura 4.3.4.1 Diagrama de Clases de la BD 51](#_czc5y5qvmy0v)

[Figura 4.3. Diagramas de Flujo del Prototipo de App 53](#_d466iwuns7e9)

[Figura 4.3. Diagramas de Secuencia del Prototipo de App 54](#_y9wk2rba6g4h)

# Abreviaturas

Una lista con las abreviaturas más utilizadas en el documento.

| **Abreviatura** | **Significado** |
| --- | --- |
| ACNUR | Alto Comisionado de las Naciones Unidas para los Refugiados |
| BBVA | Banco Bilbao Vizcaya Argentaria |
| BD | Base de Datos |
| C++ | Evolución del lenguaje C (“C incrementado”) |
| CNDH | Comisión Nacional de los Derechos Humanos |
| CSS | Cascading Style Sheets (Hojas de Estilo en Cascada) |
| CRP | Capacity Requirements Planning (Planificación de Requerimientos de Capacidad) |
| DT‑HPA | Desarrollo y Transferencia de Hábitos y Prácticas de Autocuidado |
| ERP | Enterprise Resource Planning (Planificación de Recursos Empresariales) |
| HPA | Hábitos y Prácticas de Autocuidado |
| HTML | HyperText Markup Language (Lenguaje de Marcado de Hipertexto) |
| IBM | International Business Machines |
| INEGI | Instituto Nacional de Estadística, Geografía e Informática |
| Java | Lenguaje de programación Java (derivado de una isla en Indonesia) |
| JSON | JavaScript Object Notation |
| LAMP | Linux, Apache, MySQL, PHP (stack de tecnologías para desarrollo web) |
| NoSQL | “Not Only SQL” (bases de datos no relacionales) |
| ONG | Organización No Gubernamental |
| OIM | Organización Internacional para las Migraciones |
| PHP | Hypertext Preprocessor (Preprocesador de Hipertexto) |
| Py | Python (lenguaje de programación) |
| SELECT | Palabra reservada de SQL para extraer datos (“select”) |
| SESNSP | Secretariado Ejecutivo del Sistema Nacional de Seguridad Pública |
| SQL | Structured Query Language (Lenguaje de Consulta Estructurada) |
| SO | Sistema Operativo |
| UML | Unified Modeling Language (Lenguaje Unificado de Modelado) |
| UI | User Interface (Interfaz de Usuario) |
| UIX | User Interface/User Experience (Interfaz de Usuario / Experiencia de Usuario) |
| VACUUM | Comando SQL para limpieza y optimización de bases de datos |
| XML | Extensible Markup Language |

# Capítulo 1 Fundamentos de la Investigación

## Introducción

La migración a través de México hacia Estados Unidos representa un fenómeno de creciente complejidad y magnitud. Datos de la Organización Internacional para las Migraciones (OIM) revelan que, durante el primer trimestre de 2024, se registraron 360,146 eventos de personas en situación migratoria en México, lo que refleja un incremento del 28% en comparación con el mismo periodo del año anterior (Organización Internacional para las Migraciones, 2024). Este flujo, impulsado por factores estructurales como la violencia, la pobreza extrema en países centroamericanos y la inestabilidad política, enfrenta a los migrantes a riesgos sistemáticos durante su trayecto. Según el Instituto Nacional de Estadística y Geografía (INEGI), el 60.7% de la población mexicana percibió en 2024 un deterioro en la seguridad pública a nivel nacional, con porcentajes aún más alarmantes en entidades federativas clave para el tránsito migratorio, como Chiapas y Veracruz (Llanos Reynoso, 2023).

En el contexto actual, el uso de tecnologías avanzadas, como la Inteligencia Artificial y el Aprendizaje Máquina (*Machine Learning* por como se conoce en inglés), permite desarrollar soluciones innovadoras para mitigar estos riesgos y optimizar la asistencia a los migrantes. Estudios recientes, como el análisis basado en la Teoría de Grafos, han identificado patrones críticos en las rutas migratorias. Por ejemplo, se documentaron 545 puntos de servicio Organización no Gubernamentales (ONG’s) distribuidos en 183 nodos geográficos, incluyendo ciudades estratégicas como Tapachula y Monterrey, así como zonas de alto riesgo asociadas a violencia o presencia de estaciones migratorias (Llanos Reynoso, 2023). No obstante, estas rutas intersectan áreas con índices delictivos preocupantes: en Chiapas, la tasa de secuestros alcanza los 14.2 casos por cada 100,000 habitantes, mientras que en Veracruz, el 32.5% de los delitos reportados corresponden a robos (Gobierno de México, 2024). A ello se suman condiciones ambientales extremas, como temperaturas de hasta 50°C en regiones desérticas, que incrementan la vulnerabilidad de los migrantes a deshidratación y muerte (Hoover, 2019).

Para abordar estos desafíos, este proyecto propone el desarrollo de un prototipo de aplicación móvil basada en *Machine Learning*, diseñada para ofrecer información sobre una ruta con el menor índice de criminalidad de forma mensual con ubicaciones de albergues y puntos de asistencia humanitaria. A diferencia de plataformas tradicionales de navegación, esta aplicación integrará datos de criminalidad y ubicaciones de albergues y comedores con el único fin proporcionar recomendaciones de rutas a migrantes en tránsito.

El uso de algoritmos de aprendizaje automático permitirá la detección de zonas de riesgo con base en datos históricos y para poder proporcionar una ruta de menor riesgo en robos y secuestros hacia una Organización no Gubernamental (ONG) que se enfoque en brindar asilo o comida.

Las iniciativas existentes para abordar estos desafíos presentan limitaciones significativas. Organizaciones internacionales, como la Organización Internacional de Migrantes (OIM) y la Agencia de la ONU para los Refugiados (ACNUR) centran sus esfuerzos en refugiados o contextos de emergencia, dejando fuera a migrantes en tránsito que no cumplen con estos perfiles (ACNUR, n.d.). La propuesta de esta investigación busca cerrar esta brecha mediante la aplicación de tecnologías avanzadas, mejorando la seguridad y la calidad de la información disponible para los migrantes.

## Planteamiento del problema

La migración en las fronteras sur y este de México se ha intensificado significativamente en los últimos años, registrándose un flujo de 550,085 migrantes provenientes de países distintos a México, según datos recientes del INEGI y del International Rescue Committee. A lo largo de su trayecto, estas personas enfrentan condiciones adversas, incluyendo riesgos de violencia, desinformación y falta de acceso a recursos básicos. La ausencia de una solución digital centrada en sus necesidades representa una brecha que este proyecto busca atender, muchas de las situaciones adversas que un migrante en tránsito puede enfrentar en su trayecto son:

* **Violencia y explotación:** Los migrantes son víctimas frecuentes de delitos como el tráfico de personas, explotación laboral y violaciones de derechos humanos (Hoover, 2019).
* **Vulnerabilidad y riesgos:** Los migrantes en tránsito por la frontera sur y este de México enfrentan numerosos peligros que ponen en riesgo su seguridad y bienestar. Estas amenazas incluyen la violencia, el tráfico de personas, la explotación laboral y violaciones a los derechos humanos. La falta de información confiable sobre rutas con baja tasa de criminalidad y recursos disponibles agrava esta situación, exponiéndolos a condiciones peligrosas durante su trayecto (Hoover, 2019). Además, los migrantes desconocen las zonas con alta incidencia delictiva y carecen de herramientas tecnológicas accesibles para planificar su trayecto de manera segura.
* **Condiciones climáticas extremas:** En regiones donde las temperaturas alcanzan los 50°C, la falta de acceso a agua potable y aumenta el riesgo de deshidratación y muerte. “Según el Manual para Migrantes” de la CDNH, cruzar en temporadas de calor incrementa significativamente el peligro de fallecimientos en áreas remotas (Hoover, 2019).
* **Falta de redes de apoyo:** Muchos migrantes enfrentan aislamiento social y dificultad para acceder a asistencia humanitaria, donde lamentablemente no conocen los servicios de los que son acreedores, incrementando su vulnerabilidad tanto física como emocional (Hoover, 2019). Aunque existen iniciativas como el trabajo de ONGs y organizaciones gubernamentales, su alcance es limitado, dejando a muchos migrantes sin la protección y recursos necesarios para un tránsito seguro.

A partir de una investigación preliminar, se identificó una limitada oferta de aplicaciones móviles en México orientadas específicamente a brindar asistencia a personas migrantes en tránsito. Hasta el momento, son escasas las herramientas tecnológicas que integren de manera sistemática información sobre rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad, así como la localización de albergues, y comedores comunitarios.

## Propuesta de solución

Para abordar la problemática identificada en la sección anterior, se plantea el desarrollo de un prototipo de aplicación móvil como una herramienta tecnológica integral que combine *Machine Learning*, bases de datos actualizadas mensualmente con datos del Secretariado Ejecutivo del Sistema Nacional de Seguridad Pública y un sistema de visualización de mapas con rutas. Este prototipo permitirá a los migrantes en tránsito acceder a información y rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad, servicios humanitarios y zonas de riesgo.

## Objetivo general

Desarrollar un prototipo de aplicación móvil que genere rutas entre organizaciones no gubernamentales (ONG), con el fin de permitir a los migrantes en tránsito desde la frontera sur hasta la frontera norte de México acceder a trayectos con un menor índice de criminalidad en robos y secuestros, mediante herramientas de machine learning y una base de datos que considere la incidencia de la criminalidad según fuentes oficiales.

### Objetivos específicos

* **Analizar e implementar algoritmos de *Machine Learning*** para generar rutas y un mapa que considere el índice de criminalidad por municipio, analizando datos de robos y secuestros, así como la ubicación de servicios humanitarios u ONG’ s para los migrantes en tránsito.
* **Diseñar y desarrollar una base de datos**, para almacenar información sobre la criminalidad en robos y secuestros por estado y municipio, así como los servicios humanitarios u ONG’ s enfocadas en albergues y comedores.
* **Desarrollar un prototipo de aplicación móvil para la visualización de un mapa** que muestre las rutas generadas, localice en el mapa los centros de ayuda humanitaria, como albergues y comedores y muestre las zonas de riesgo por municipio.

## Justificación

La problemática migratoria en la frontera sur de México demanda una intervención urgente y especializada. Según la Organización Internacional para las Migraciones (Organización Internacional para las Migraciones, 2024) y la Unidad de Política Migratoria (Gobierno de México, 2024), el flujo de migrantes ha aumentado considerablemente en el primer trimestre de 2024, lo que se traduce en situaciones de alta vulnerabilidad para quienes transitan por territorios con altos índices delictivos. Esta situación se agrava al considerar que los migrantes enfrentan amenazas constantes, como el tráfico de personas, la explotación laboral y la violencia, tal como se señala en el análisis realizado por Llanos Reynoso (Llanos Reynoso, 2023).

El impacto negativo de esta situación se refleja también en las percepciones ciudadanas, ya que un alto porcentaje de la población reconoce que la seguridad pública se deteriorará en el corto plazo (Llanos Reynoso, 2023). Además, el Manual para Migrantes (Hoover, 2019) resalta que las condiciones extremas, como las temperaturas elevadas y la falta de recursos básicos, aumentan el riesgo de deshidratación y otros problemas de salud, haciendo indispensable la implementación de medidas preventivas y de asistencia oportuna.

En este contexto, la propuesta del prototipo de aplicación móvil se plantea su enfoque en la identificación de rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad que utiliza *Machine Learning* y una base de datos especializada para generar rutas entre organizaciones no gubernamentales (ONG) distribuidas en todo el país. Esta innovación tecnológica planteada facilita la integración de datos históricos y actuales sobre criminalidad actualizados mensualmente para reflejar las condiciones del entorno.

Por otro lado, aunque existen iniciativas de rescate y protección, como los Grupos Beta de Protección a Migrantes (Instituto Nacional de Migración, 2022), su función se orienta primordialmente a la asistencia en situaciones de emergencia, dejando un vacío en el ámbito preventivo y de planificación segura.

La relevancia del proyecto se sustenta además en el análisis realizado en el estudio “Los caminos de los migrantes que pasan por México: análisis desde la Teoría de Grafos” (Llanos Reynoso, 2023). Dicho estudio identificó rutas y zonas de riesgo mediante técnicas avanzadas, lo que respalda la implementación de algoritmos de *Machine Learning* para la generación de rutas optimizadas.

Asimismo, la comparación con otras organizaciones internacionales, como el International Rescue Committee (Instituto Nacional de Migración, 2022) y el Alto Comisionado de las Naciones Unidas para los Refugiados (ACNUR) (ACNUR, n.d.), pone en evidencia que, aunque estos organismos ofrecen asistencia en emergencias y para refugiados, no cuentan con una herramienta tecnológica que permita a los migrantes tomar decisiones informadas antes de exponerse a zonas de alto riesgo. En consecuencia, el prototipo se posiciona como una solución complementaria, que potencia la labor de estas organizaciones al ofrecer un componente preventivo y de planificación basado en datos precisos y actualizados.

## Organización del documento

**Capítulo 1. Fundamentos de la Investigación**El capítulo redacta los elementos esenciales del prototipo, formulando los fundamentos que respaldan la existencia de la problemática a resolver para ello abordamos la propuesta que sugerimos, los objetivos a alcanzar y la justificación del prototipo. El lector comprende el contexto, la necesidad y el propósito de esta investigación, así como la estructura que seguirá el documento.

**Capítulo 2. Estado del Arte**Se desglosa la revisión crítica de artículos o proyectos especializados que exploran enfoques y algoritmos relevantes para el modelado de rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad para migrantes. Se establecen comparaciones que permiten evaluar fortalezas, debilidades y vacíos en los estudios actuales, apoyando la elección de técnicas en esta propuesta.

**Capítulo 3. Marco Teórico**En este apartado se desglosan los términos más relevantes del proyecto: migración, rutas, criminalidad, *machine learning* y bases de datos. Además, se explican diferentes tipos de aprendizaje automático y algoritmos aplicables, lo que permite construir una base teórica sólida para el desarrollo de la solución propuesta.

**Capítulo 4. Análisis**Este capítulo identifica las necesidades que tendrá el prototipo mediante la definición de las reglas de negocio, requerimientos funcionales y no funcionales. Se estructura el problema desglosando lo que se espera del prototipo, para posteriormente diseñar e implementar una interpretación de sus funciones.

**Capítulo 5. Diseño**Se representan visualmente las ideas del prototipo mediante diagramas de arquitectura UML, casos de uso y modelados de datos. El capítulo da forma estructurada al prototipo a través de esquemas claros que definen cómo interactúan los elementos entre sí, permitiendo prever su comportamiento antes del desarrollo.

**Capítulo 6. Desarrollo e Implementación**Aquí se evalúa su viabilidad desde las perspectivas técnica, operativa y de riesgo, justificando su ejecución, también se explican las decisiones técnicas tomadas durante el desarrollo y cómo se implementa el prototipo propuesto.

**Capítulo 7. Pruebas**Este capítulo desglosa un plan de pruebas bien para verificar que los códigos y programas funcionen óptimamente para el futuro desarrollo del prototipo, se generan casos y escenarios que permiten observar cómo responde el prototipo ante diversas situaciones recopilando resultados y evaluando su confiabilidad.

**Capítulo 8. Conclusiones y trabajo a futuro**Se sintetizan los logros del proyecto, se evalúa el cumplimiento de los objetivos sobre los desafíos enfrentados. Finalmente, se sugieren mejoras y líneas de trabajo futuras para mejorar la solución desarrollada del prototipo.

**Capítulo 9. Referencias**Se listan de manera ordenada todas las fuentes consultadas durante el desarrollo del proyecto.

**Anexos**Incluyen material complementario como, capturas de pantalla o pruebas adicionales que respaldan el contenido del trabajo principal.

# Capítulo 2. Estado del arte

En la última década, las tecnologías de información y de *Machine Learning* y sistemas de geolocalización han permitido el desarrollo de soluciones innovadoras que responden a desafíos complejos en diversas áreas.

Este capítulo explora una revisión de aplicaciones y proyectos que abordan temáticas similares al proyecto propuesto, en el que se pretende desarrollar un prototipo de aplicación móvil capaz de generar rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad entre organizaciones no gubernamentales (ONG) a nivel nacional.

Los trabajos analizados incluyen desde aplicaciones de mapeo ampliamente utilizadas hasta iniciativas tecnológicas orientadas a la asistencia humanitaria y el apoyo en situaciones migratorias, evidenciando tanto sus potencialidades como las limitaciones en el manejo de información en tiempo real y la integración de diversos sistemas. La revisión de la literatura permite identificar oportunidades de mejora, sobre todo en la incorporación de algoritmos predictivos y la validación empírica de datos críticos para la toma de decisiones en entornos dinámicos y de riesgo.

## 2.1 *Machine Learning* para modelar la migración humana

El proyecto *A Machine Learning Approach to Modeling Human Migration* parte de una premisa similar a la que se plantea en este documento: los migrantes, incluso en situaciones de alta vulnerabilidad, suelen portar dispositivos móviles que pueden convertirse en herramientas clave para su seguridad, comunicación y orientación durante el tránsito (Caleb & Bistra, 2018). En este sentido, los autores proponen el desarrollo de un prototipo de aplicación móvil que aproveche los datos generados por los propios usuarios con el fin de identificar eventos de riesgo como extorsiones, robos o actos de violencia y, a partir de ello, generar rutas alternativas que resulten más seguras para los migrantes.

Este proyecto se fundamenta en el uso de datos colaborativos, conocidos como *crowdsourced data*, los cuales son proporcionados en tiempo real por las propias personas usuarias. Este enfoque permite construir un mapeo dinámico de zonas de peligro, ofreciendo advertencias geolocalizadas y adaptativas según los contextos reportados (Caleb & Bistra, 2018). El artículo no profundiza sobre cómo se usa la implementación de técnicas de inteligencia artificial como el *Machine Learning*, sí hace énfasis en la relevancia de integrar análisis geoespacial, recolección estructurada de datos y un diseño accesible de interfaces, pensado especialmente para usuarios con conocimientos tecnológicos limitados.

La propuesta se centra en la inmediatez de los reportes y en la experiencia directa de los migrantes, la aplicación puede reaccionar rápidamente ante cambios en el entorno y proporcionar alternativas que reduzcan la exposición a contextos violentos o de riesgo. Su simplicidad de implementación la hace particularmente adecuada para contextos donde los recursos institucionales son insuficientes o inexistentes, lo cual es una realidad frecuente en muchas zonas de tránsito migratorio (Caleb & Bistra, 2018).

En contraste, el proyecto propuesto en este documento también busca incrementar la seguridad de los migrantes, pero lo hace desde una aproximación distinta basada en el uso de técnicas de *Machine Learning* aplicadas con datos gubernamentales sobre criminalidad en México. Este enfoque ofrece una base verificable, respaldada por fuentes institucionales, lo cual fortalece credibilidad para el modelo. A diferencia del enfoque del proyecto anterior, este proyecto busca priorizar la integración de datos especializados y validados, lo cual permite generar rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad. Además, se contempla la incorporación de la ubicación de organizaciones no gubernamentales (ONG’s) y servicios humanitarios como parte del soporte integral a los migrantes.

Si bien ambos proyectos comparten la preocupación por la seguridad de las personas en movilidad y coinciden en el uso de dispositivos móviles como medio de interacción, presentan diferencias fundamentales en la fuente y naturaleza de los datos que utilizan. El proyecto de Caleb y Bistra (2018) se apoya en el *crowdsourcing*, lo cual permite una actualización dinámica pero con un menor nivel de verificabilidad.

En cambio, el proyecto aquí planteado usa datos de fuentes externas y oficiales, para tener un control sobre los datos de entrada para la aplicación para garantizar que las rutas tienen menor índice de criminalidad de robos y secuestros en el trayecto migrante que atraviesa México. Esta delimitación geográfica responde a necesidades locales con un enfoque más estructurado.

### 2.2 Algoritmo para calcular la ruta más segura y óptima

Un antecedente relevante a considerar en este ámbito es el trabajo titulado *“Algoritmo para calcular la ruta más segura y óptima”*, el cual se centra en la aplicación de algoritmos de búsqueda como *Dijkstra* y *A\** para determinar trayectos que prioricen la seguridad del usuario, combinando factores de distancia, tiempo y riesgo (Andrés Areiza et al., 2022). La comparación con dicho proyecto permite establecer similitudes metodológicas, así como identificar divergencias cruciales en cuanto a objetivos sociales, enfoque técnico y escalabilidad.

Desde una perspectiva técnica, ambos proyectos comparten el propósito de calcular rutas seguras en entornos donde la criminalidad representa un factor determinante. No obstante, mientras que el trabajo previo realiza una implementación basada en pesos asignados a grafos sobre mapas de calles, la propuesta planteada en este trabajo incorpora *Machine Learning* para una adaptación dinámica a los cambios en los patrones de riesgo según la criminalidad de robos y secuestros en méxico. El uso de *Machine learning* representa un avance sustancial, ya que posibilita identificar correlaciones complejas entre variables (como tipo de delito, zonas conflictivas, entre otras), que no pueden ser resueltas únicamente con algoritmos deterministas (Andrés Areiza et al., 2022).

Además, el proyecto titulado *“Algoritmo para calcular la ruta más segura y óptima”* emplea información limitada, ya que no habla de actualizaciones, generado de forma simulada o estática, lo cual restringe su uso en escenarios reales. El prototipo de app planteado en este documento contempla el diseño de una base de datos, actualizada mensualmente, con información oficial sobre robos y secuestros, así como la geolocalización de centros humanitarios. Esta actualización periódica no sólo fortalece la confiabilidad del sistema.

Otra diferencia fundamental radica en el componente de asistencia humanitaria. Mientras el proyecto anterior se concentra exclusivamente en la minimización del riesgo durante el trayecto, la aplicación integra también la localización y evaluación de organizaciones no gubernamentales (ONGs), lo cual añade un valor humanitario significativo. Esta funcionalidad convierte a la app en una herramienta integral para la protección del migrante, y no solamente en una solución de navegación.

Finalmente, el sistema de visualización de mapas también representa un elemento diferenciador. Aunque ambos trabajos contemplan interfaces gráficas para la representación de rutas, el prototipo de aplicación móvil se orienta hacia dispositivos móviles y prioriza la usabilidad para personas en tránsito,

El proyecto *“Algoritmo para calcular la ruta más segura y óptima”* constituye un antecedente técnico para demostrar la viabilidad de la seguridad como criterio de optimización de rutas, mientras que el trabajo terminal planteado se distingue por su enfoque de actualización mensual de datos, su integración con servicios humanitarios y su orientación social. Estas diferencias mejoran la aplicabilidad en el contexto migratorio mexicano.

### 2.3 Mapping Nyumbani

*Mapping Nyumbani* se destaca por su enfoque innovador en la creación de rutas seguras para refugiados que huyen de conflictos o áreas inseguras. Este proyecto utiliza geovallado (alrededor de una ubicación geográfica real) y conocimientos comunitarios para mapear caminos seguros, enviando alertas en tiempo real a los refugiados a través de mensajes de texto sobre posibles peligros en su trayecto. La iniciativa se basa en plataformas de mapeo comunitario como OpenStreetMap y trabaja directamente con refugiados para identificar rutas y zonas de riesgo, implementando zonas geovalladas que, al ser atravesadas, activan notificaciones relevantes en los dispositivos móviles de los refugiados (Samuel Hall, 2023).

Al comparar *Mapping Nyumbani* con el prototipo de aplicación móvil propuesto para generar rutas seguras para migrantes en tránsito por México, emergen similitudes y diferencias significativas en términos de objetivos y tecnologías empleadas.​

Ambos proyectos comparten el objetivo primordial de proporcionar rutas seguras a poblaciones en situación de vulnerabilidad durante sus desplazamientos. Sin embargo, mientras *Mapping Nyumbani* se centra en refugiados que huyen de conflictos en África Oriental, el prototipo mexicano está diseñado para migrantes que transitan desde la frontera sur hasta la frontera norte de México. Esta diferencia geográfica influye en la naturaleza de los riesgos abordados y en las fuentes de datos utilizadas para identificar zonas peligrosas.​

En cuanto a la metodología, *Mapping Nyumbani* utiliza geovallado para enviar alertas en tiempo real a los refugiados cuando ingresan a áreas de riesgo previamente identificadas. Este enfoque permite una comunicación inmediata y directa, esencial para la seguridad en contextos de movilidad constante (Samuel Hall, 2023). Por otro lado, el prototipo mexicano propone el uso de algoritmos de (*Machine Learning*) para analizar datos oficiales sobre criminalidad y generar rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad pero no envía notificaciones si no que cambia de ruta permitiendo una adaptación dinámica a patrones cambiantes de criminalidad mensualmente y la personalización de rutas según perfiles de riesgo específicos.​

En relación con las fuentes de datos, *Mapping Nyumbani* se apoya en plataformas de mapeo comunitario como OpenStreetMap y en la colaboración directa con refugiados para identificar rutas y zonas de riesgo. Este enfoque participativo garantiza que la información refleje las experiencias y percepciones de seguridad de la comunidad (Samuel Hall, 2023). El prototipo mexicano, en contraste, planea desarrollar una base de datos especializada que almacene información sobre zonas de riesgo y servicios humanitarios, actualizada mensualmente con datos oficiales sobre robos y secuestros.

Una diferencia notable entre ambos proyectos es el componente de asistencia humanitaria. Mientras que *Mapping Nyumbani* se enfoca principalmente en la seguridad durante el tránsito, el prototipo mexicano integra la localización y evaluación de organizaciones no gubernamentales (ONGs) que ofrecen servicios de ayuda humanitaria. Esta funcionalidad adicional convierte a la aplicación en una herramienta integral que no solo guía a los migrantes por rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad, sino que también facilita el acceso a recursos esenciales durante su trayecto.​

En términos tecnológicos, *Mapping Nyumbani* opera mediante el envío de mensajes de texto a dispositivos móviles, una solución adecuada dado que muchos refugiados poseen teléfonos básicos sin acceso constante a internet (Samuel Hall, 2023). El prototipo mexicano, al ser una aplicación móvil, presupone el acceso a teléfonos inteligentes con conectividad, lo que puede limitar su alcance entre ciertos segmentos de la población migrante.​

Estas diferencias reflejan las particularidades de los contextos en los que operan y las necesidades específicas de las poblaciones a las que sirven. Ambos proyectos ejemplifican cómo la tecnología puede ser aprovechada para mejorar la seguridad y el bienestar de personas en situaciones de movilidad forzada.

### 2.4 Identificación de zonas de vulnerabilidad a la violencia contra mujeres

En “Using heat maps to identify areas prone to violence against women in the public sphere”, los autores recogen reportes de agresiones a mujeres en Corregidora (Querétaro) y los convierten en puntos sobre un mapa. Luego aplican k‑means, un método que agrupa esos puntos en “paquetes” según lo cerca que estén unos de otros, eligiendo inicialmente diez grupos para equilibrar el tamaño del área y la cantidad de incidentes (Garfas Royo et al., 2020).

Con estos grupos, elaboran un mapa de calor que muestra zonas con más o menos incidentes, y así pueden clasificar cada área como de alto, medio o bajo riesgo. Para comprobar que esos grupos realmente reflejan los lugares problemáticos, visitan veintidós de las zonas con más concentración de casos y hacen observaciones en el terreno; de esa manera, usan k‑means no solo para encontrar los puntos calientes, sino también para decidir dónde ir a verificar y, finalmente, sugerir mejoras en el diseño de las calles que ayuden a prevenir estas agresiones (Garfas Royo et al., 2020).

##### 

##### Tabla 2.4.1 Tabla comparativa

| **Título del Trabajo** | **Uso de Machine Learning** | **Actualización de Datos** | **Integración de Servicios Humanitarios** | **Envío de Alertas** | **Mapas** |
| --- | --- | --- | --- | --- | --- |
| A Machine Learning Approach to Modeling Human Migration | ✔️ | ✔️ | ❌ | ❌ | ❌ |
| Algoritmo para Calcular la Ruta Más Segura y Óptima | ✔️ | ❌ | ❌ | ❌ | ✔️ |
| Mapping Nyumbani | ❌ | ✔️ | ❌ | ✔️ | ✔️ |
| Identificación de zonas de vulnerabilidad a la violencia contra mujeres | ✔️ | ❌ | ❌ | ❌ | ✔️ |
| Prototipo Propuesto | ✔️ | ✔️ | ✔️ | ✔️ | ✔️ |

# Capítulo 3. Marco Teórico

### 3.1Migración

Por migración entendemos los desplazamientos de personas que tienen como intención un cambio de residencia desde un lugar de origen a otro de destino, atravesando algún límite geográfico que generalmente es una división político administrativa. Este fenómeno implica un desplazamiento o movimiento espacial que tiene como objetivo buscar mejores oportunidades de vida de los individuos, ya sea porque en su lugar de origen o de establecimiento actual no existen tales oportunidades (Bueno Sánchez, 2000).

### 3.2 Índice de Criminalidad

El índice de criminalidad, o tasa de criminalidad, es un indicador el cual es medible y sus datos se basan en la frecuencia y gravedad de los delitos en una población o área geográfica. En México estos índices se miden con parámetros como la percepción de los habitantes y las denuncias presentadas en cierto periodo de tiempo, los organismos que recolectan dichos datos son encuestas del Instituto Nacional de Estadísticas y Geografía (INEGI), el Observatorio Nacional Ciudadano y reportes municipales del departamento de justicia e instituciones privadas de seguridad. Este índice de criminalidad se calcula dividiendo el número de delitos entre la población y multiplicando por una base, como 10,000 o 100,000 y solo cuentan aquellos cometidos por mayores de 18 años (Sistema Nacional de información estadística y geográfica, 2025).

### 3.3 Rutas

En el contexto de la generación de rutas, el término "ruta" se define como las líneas, arcos o bordes de conexión que unen un origen con un destino se trata de un elemento geométrico a partir del cual se construyen múltiples líneas y polígonos. Una ruta es una secuencia de segmentos conectados no intersecantes, donde no hay dos segmentos que tengan el mismo punto de inicio ni el mismo extremo; cualquier entidad de línea, como una calle, una autopista, un río o una tubería que tiene un identificador único a través de una “red” (eseri, 2025).

Ruta a través de una red que visita un conjunto de ubicaciones de red especificadas. En el análisis del problema de generación de rutas para vehículos (VRP), una ruta también puede hacer referencia a un vehículo y a sus propiedades y restricciones asociadas (eseri, 2025).

### 3.4 Algoritmos

Un algoritmo informático es un conjunto de pasos para realizar una tarea que está descrita con la suficiente precisión para que una computadora pueda ejecutarla. (Cormen, 2013, 1)

### 3.5 Inteligencia Artificial

Es la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un ser humano (McCarthy, 2007).

La Inteligencia Artificial (IA) tiene múltiples ramas que resuelven problemas específicos haciendo uso de diferentes enfoques (McCarthy, 2007).

### 3.6 *Machine Learning*

El machine learning es la ciencia de desarrollo de algoritmos y modelos estadísticos que utilizan los sistemas de computación con el fin de llevar a cabo tareas sin instrucciones explícitas, en vez de basarse en patrones e inferencias. Los sistemas de computación utilizan algoritmos de *machine learning* para procesar grandes cantidades de datos históricos e identificar patrones de datos. Esto les permite generar resultados con mayor precisión a partir de un conjunto de datos de entrada (Amazon, 2025), este se divide por:

### 3.6.1 Aprendizaje Supervisado

Este tipo de *machine learning* alimenta datos históricos de entrada y salida en algoritmos de aprendizaje automático, con procesamiento entre cada par de entrada/salida que permite al algoritmo cambiar el modelo para crear salidas lo más alineadas posible con el resultado deseado. Entre los algoritmos más utilizados en el aprendizaje supervisado se encuentran las redes neuronales *(neural networks)*, los árboles de decisión *(decision trees)*, la regresión lineal *(linear regression)* y las máquinas de vectores de soporte *(support vector machines o SVM)*. La *figura 3.6.1.1* muestra el funcionamiento del aprendizaje supervisado (Gartner, 2025).

### 

###### Figura 3.6.1.1 Modelo de aprendizaje Supervisado (Luna, 2018)

### 

### 

### 3.6.2 Aprendizaje no supervisado

El aprendizaje supervisado requiere que los usuarios ayuden a la máquina a aprender, el aprendizaje no supervisado *(unsupervised learning)* no utiliza los mismos conjuntos de entrenamiento y datos etiquetados. En su lugar, la máquina busca patrones ocultos en los datos. Este tipo de aprendizaje automático es muy útil cuando se necesita identificar patrones y utilizar los datos para tomar decisiones. Entre los algoritmos más utilizados en el aprendizaje no supervisado se encuentran los modelos de Markov ocultos *(Hidden Markov models)*, k-means, la agrupación jerárquica *(hierarchical clustering)* y los modelos de mezcla gaussiana *(Gaussian mixture models)*. En la figura *3.6.2.1* se muestra el funcionamiento del aprendizaje no supervisado. (Gartner, 2025)

###### Figura 3.6.2.1 Modelo de aprendizaje No Supervisado (Luna, 2018)

Algoritmo de agrupamiento

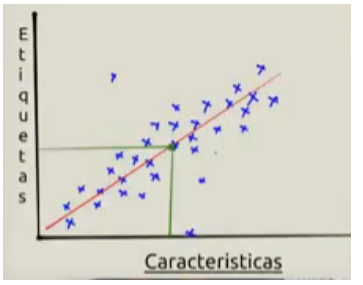
Constituye un amplio conjunto de técnicas para el tratamiento de datos entre las que se puede mencionar: la selección y extracción de características, la clasificación de un objeto en un grupo dado y la división de los datos en grupos (agrupamiento); Dentro de este enfoque, se distingue entre las aproximaciones paramétrica y no paramétrica (Pascual et al., 2007).

* En el enfoque **paramétrico** se parte de la idea de que, antes de observar los datos, ya se tiene cierta noción sobre la “forma” que siguen los distintos grupos o categorías en el espacio en el que se representan los datos. Esta “forma” se expresa mediante funciones matemáticas sencillas a partir de un número fijo de valores o “parámetros”, Las decisiones sobre a qué grupo pertenece un nuevo dato se toman calculando, mediante la función asumida, la probabilidad de que ese dato haya surgido de cada uno de los grupos. De esta manera, las “fronteras” o líneas que separan un grupo de otro están determinadas por esas funciones y por los valores de los parámetros que se ajustaron (Pascual et al., 2007).
* El enfoque **no paramétrico** no formula ninguna suposición previa acerca de cómo “lucen” las distribuciones de los distintos grupos. No se utiliza ninguna función predefinida, ni se reducen las características de los datos a un número pequeño de parámetros.

Las fronteras de decisión se van dibujando a medida que se conocen más datos, sin que exista una forma abstracta definida de antemano (Pascual et al., 2007).

### 3.6.4 Clusters

En este método lo que se espera es un número. No lo ubica en un grupo, sino que devuelve un valor específico. Por ejemplo, el precio de una casa. El algoritmo tiene el precio de diferentes casas, pequeñas, grandes, en el campo, en la ciudad, etc. y por medio de un gráfico de dispersión, puede predecir el precio correcto de una casa en consulta. La siguiente *figura 3.6.4.1* muestra gráficamente el ejemplo mencionado anteriormente (Sandoval, 2018).

*Figura 3.6.4.1 Gráfico del algoritmo de regresión.* (Sandoval, 2018).

### K-Means Clustering

La agrupación de medias K es un algoritmo de aprendizaje no supervisado utilizado para la agrupación en clústeres de datos, que agrupa los puntos de datos no etiquetados en grupos o clústeres. Es uno de los métodos de clustering más populares utilizados en el *Machine learning*. A diferencia del aprendizaje supervisado, los datos de entrenamiento que utiliza este algoritmo no están etiquetados, lo que significa que los puntos de datos no tienen una estructura de clasificación definida (IBM, 2025).

Las medias K son un algoritmo de clustering iterativo basado en centroides que divide un conjunto de datos en grupos similares en función de la distancia entre sus centroides. El centroide, o centro del clúster, es la media o la mediana de todos los puntos dentro del clúster, según las características de los datos.

El objetivo es minimizar la suma de distancias entre los puntos de datos y sus clústeres asignados. Los puntos de datos más cercanos a un centroide se agrupan dentro de la misma categoría. Un valor k más alto, o el número de conglomerados, significa conglomerados más pequeños con mayor detalle, mientras que un valor k más bajo da lugar a conglomerados más grandes con menos detalle (IBM, 2025).

Se asigna cada punto de datos a su centroide más cercano en función de la distancia euclídea; Calcula la media de todos los puntos de cada conglomerado y asigna el centro del conglomerado, o centroide. Este proceso se repite hasta que las posiciones de los centroides hayan alcanzado la convergencia o se haya alcanzado el número máximo de iteraciones (IBM, 2025).

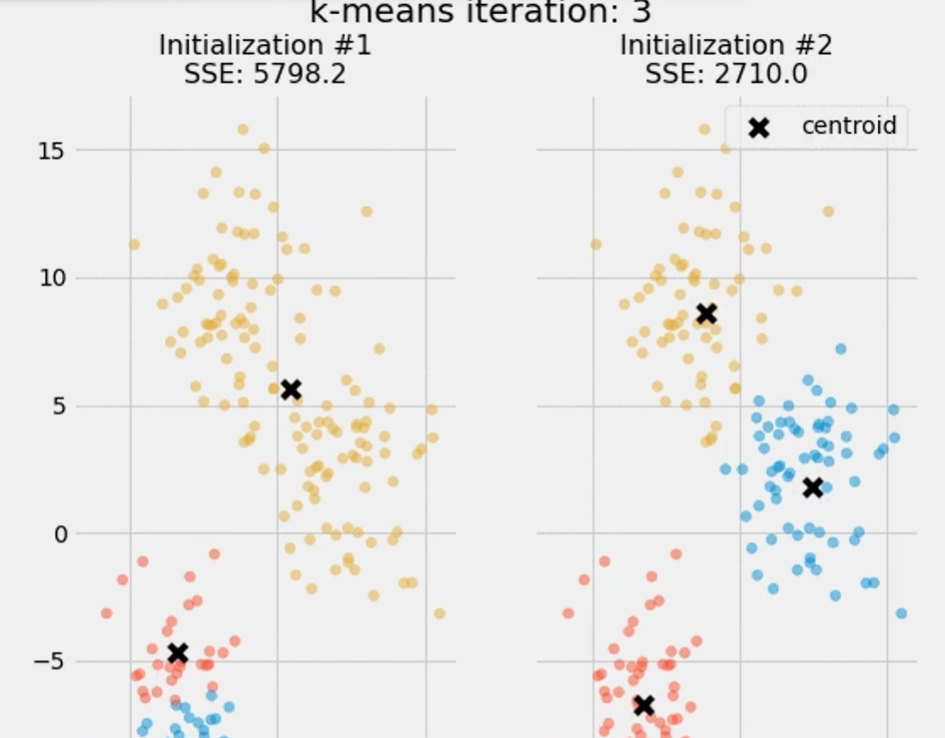


Figura ## Centroides del K means (Ramírez, 2024)

### DBSCAN (Density-Based Spatial Clustering)

Esta herramienta utiliza algoritmos de agrupamiento de aprendizaje automático no supervisado que detectan automáticamente patrones basándose únicamente en la ubicación espacial y la distancia a un número específico de vecinos. Estos algoritmos se consideran no supervisados ​​porque no requieren entrenamiento sobre lo que significa ser un clúster (ArcGIS Pro, 2025).

El algoritmo DBSCAN (Density-Based Spatial Clustering of Applications with Noise) es un método de agrupamiento que forma grupos de datos basándose en la densidad, es decir, en cómo están distribuidos los puntos en el espacio (interactivechaos, 2025).

Este enfoque parte de un concepto central llamado "muestras base" (*core samples*). Una muestra base es un punto de datos que se encuentra en una zona donde hay muchos otros puntos cercanos. Dicho de otra forma, es un punto que está en una región densa. La densidad se determina considerando una distancia máxima (es decir, un límite de cercanía) y un número mínimo de puntos necesarios para que esa zona se considere suficientemente poblada (interactivechaos, 2025).

Además de las muestras base, el algoritmo identifica también "muestras no-base" (*non-core samples*). Estas no están en una zona densamente poblada por sí mismas, pero sí se encuentran cerca de una muestra base. Aunque no cumplan el requisito de tener muchos vecinos, se aceptan dentro del grupo por estar próximas a un punto que sí cumple ese criterio (interactivechaos, 2025).

DBSCAN, es grupo o clúster está formado por un conjunto de muestras base y por todas aquellas muestras no-base que estén suficientemente cerca de alguna de las primeras. Esta cercanía se mide con una función de distancia, que puede ser, por ejemplo, la distancia euclidiana (la distancia "en línea recta").

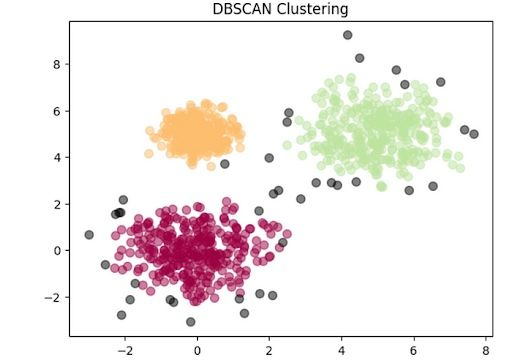


Figura ## DBSCAN Clustering (Yenigün, 2024)

### Clustering Jerárquico

Los métodos jerárquicos construyen una estructura en la que los elementos se agrupan en subconjuntos cada vez mayores hasta que todos pertenecen al mismo conjunto (Expósito Izquierdo et al., 2025).

De esta forma, no se muestra un agrupamiento sino las relaciones de proximidad que existen entre los elementos y una de las mayores dificultades al agrupar elementos es encontrar el número apropiado de clusters (Expósito Izquierdo et al., 2025).

Hay dos formas de agrupar dichos clusters (Expósito Izquierdo et al., 2025):

* Aglomerativos. Inicialmente se forman clusters individuales, cada uno de los cuales contiene a un único elemento. En cada iteración se unen los dos clusters más próximos. El procedimiento analiza cuando solo haya un cluster.
* Divisivos. Se parte de un único cluster al que pertenecen todos los elementos. En cada iteración se escoge un cluster y se divide.

El clustering jerárquico suele representarse a través de un dendograma, que muestra en qué orden se han unido los cluster y cuál es el grado de proximidad que tienen los clusters que se unen (Expósito Izquierdo et al., 2025).

El dendrograma puede usarse para generar diferentes agrupamientos. Para ello, se selecciona un nivel y se poda el dendrograma descartando los hijos de los nodos con nivel igual o superior al nivel seleccionado los nodos, hojas del árbol resultante dan el agrupamiento buscado (eseri, 2025).

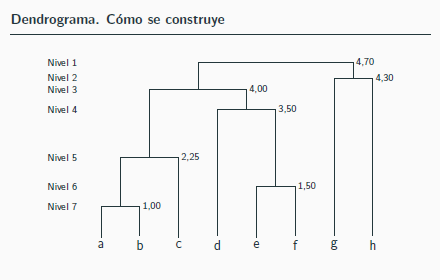


Figura ## Dendograma del clustering Jerárquico (Expósito Izquierdo et al., 2025)

## Algoritmos de rutas.

### 3.7 Algoritmo de A\*

Es un algoritmo de búsqueda inteligente o respaldado por información que busca el camino más corto desde un estado inicial al estado meta a través de un espacio de problema usando una heurística óptima. Como ignorar los pasos más cortos en algunos casos rinde una solución subóptima (LÓPEZ TAKEYAS, 2005).

Pertenece al método de búsqueda preferente por lo mejor (Best First Search). Es muy popular en la planificación de rutas y ha sido exitosamente utilizado en áreas como Inteligencia Artificial y Robótica. A\* Es un algoritmo genérico de búsqueda que utiliza información heurística para determinar cuál es el mejor camino hacia el destino. La información que utiliza es el costo estimado desde el nodo explorado hacia el nodo destino. Esta información es proporcionada por la función llamada función heurística h’(n) (LÓPEZ TAKEYAS, 2005).

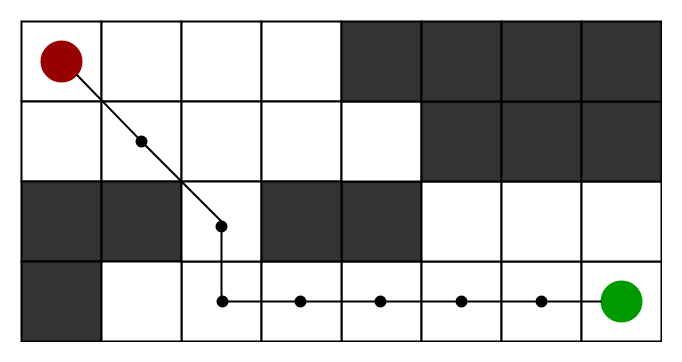
ya que siempre selecciona para expandir el nodo que parece más prometedor de acuerdo con una función de evaluación f(n), definida como *f(n)=g(n)+h(n).*

Donde:

* g(n) es el costo real acumulado desde el estado inicial hasta el nodo n.
* h(n) es la estimación heurística del costo desde el nodo n hasta el estado meta.

La heurística h(n) debe ser admisible, es decir, nunca debe sobrestimar el costo real al objetivo, para garantizar que A\* encuentre la solución óptima. Además, si la heurística también es consistente (o monótona), el algoritmo se comporta de manera más eficiente y evita ciclos.

Este algoritmo ha sido ampliamente utilizado en planificación de rutas, como en sistemas de navegación GPS, donde es fundamental encontrar caminos eficientes evitando zonas de alto tráfico o inseguridad. En el campo de la robótica, A\* Se emplea para permitir a robots autónomos moverse a través de entornos desconocidos o dinámicos, calculando trayectorias que eviten obstáculos mientras minimizan el tiempo de recorrido (Acte, 2025).



###### Figura 3.7.1 Diagrama de A\* (Belwariar, 2024).

### Algoritmo Dijkstra

El algoritmo de Dijkstra es una técnica utilizada en informática y matemáticas para encontrar el camino más corto entre dos puntos en un grafo. Un grafo es una estructura que se usa para representar conexiones entre diferentes elementos, llamados "nodos" o "vértices" (donde cada conexión tiene un costo o peso) (Richaud, 2024).

Un grafo es una estructura compuesta por nodos (vértices) y aristas (conexiones). En el contexto del algoritmo, cada arista tiene un peso que puede representar distancia, tiempo, o cualquier otra magnitud a minimizar (Richaud, 2024).

El funcionamiento del algoritmo de Dijkstra comienza asignando a cada nodo una distancia inicial de infinito, excepto al nodo de origen, cuya distancia se establece en cero. Todos los nodos se marcan inicialmente como no visitados. Luego, se selecciona el nodo no visitado con la menor distancia registrada y se evalúan sus vecinos, calculando la distancia desde el nodo inicial a cada uno de ellos. Si esta nueva distancia es menor que la actualmente registrada, se actualiza. Una vez procesados sus vecinos, el nodo actual se marca como visitado, lo que indica que ya no será evaluado nuevamente. Este proceso se repite, eligiendo siempre el nodo no visitado con la menor distancia, hasta haber visitado todos los nodos o alcanzado el nodo destino. El algoritmo garantiza encontrar el camino de menor costo desde el nodo inicial hacia todos los demás, siempre que los pesos de las conexiones sean positivos (Richaud, 2024).

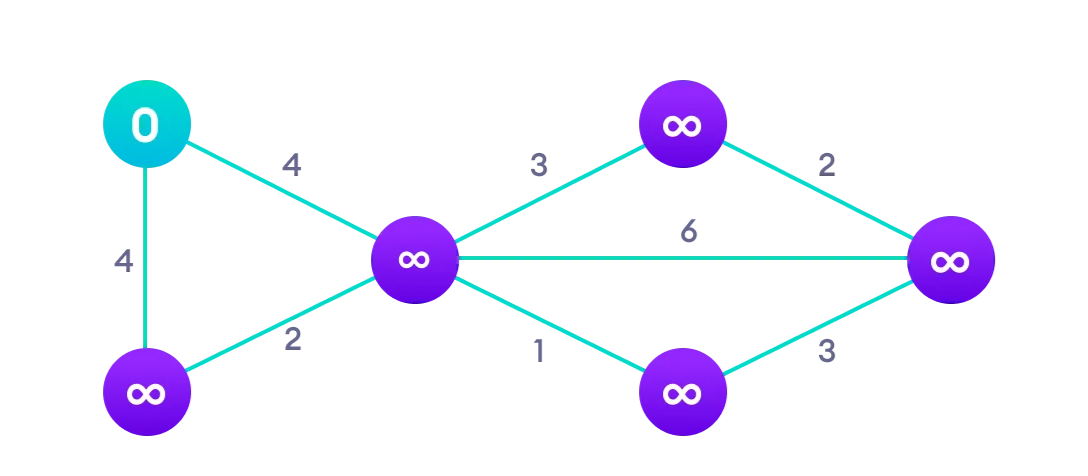


Figura ## Nodos con distancia (Richaud, 2024)

### 3.7 Base de Datos

Una base de datos se define como un conjunto organizado de datos que se almacenan y gestionan de forma sistemática para facilitar su acceso, manipulación y análisis. Esta organización no solo involucra la estructura de los datos, sino también el uso de sistemas de gestión de bases de datos (DBMS) que permiten mantener la integridad, consistencia y disponibilidad de la información. Dichos sistemas soportan operaciones básicas como inserción, actualización, eliminación y consulta de datos, siendo fundamentales en la implementación de aplicaciones empresariales y científicas (Saltos Pérez, 2022).

### 3.7.1 Tipos de bases de datos

Las bases de datos se pueden clasificar según el modelo que utilizan para organizar la información. Las dos categorías principales son:

### 3.7.1.1 Relacional

Las bases de datos relacionales se basan en el modelo propuesto por E. F. Codd. Este modelo organiza la información en tablas compuestas de filas y columnas así como se muestra en la *figura 3.7.1.1*, donde cada tabla representa una entidad del mundo real y las relaciones entre estas entidades se establecen mediante claves primarias y foráneas. El uso del lenguaje SQL (Structured Query Language) es característico de estos sistemas, ya que permite la realización de operaciones complejas y el mantenimiento de la integridad referencial. Los sistemas de gestión de bases de datos relacionales (RDBMS) han demostrado ser especialmente eficientes en entornos donde la estructura de datos es fija y las relaciones entre ellos son bien definidas (Saltos Pérez, 2022).



###### Figura 3.7.1.1 Tablas de Bases de datos relacionales (Olivera, 2022)

Existen ventajas de usar bases de datos relacionales para administrar y almacenar tus datos (Google, 2025):

* **Flexibilidad**

Es fácil agregar, actualizar o borrar tablas, relaciones y hacer otros cambios a los datos cuando lo necesites sin cambiar la estructura general de la base de datos ni afectar las aplicaciones existentes.

* **Cumplimiento de ACID**

Las bases de datos relacionales admiten el cumplimiento de ACID (atomicidad, coherencia, aislamiento y durabilidad) para garantizar la validez de los datos, sin importar si se producen errores, fallas o contratiempos de otro tipo.

* **Facilidad de uso**

Es fácil ejecutar consultas complejas con SQL, lo que permite que incluso los usuarios no técnicos aprendan a interactuar con la base de datos.

* **Colaboración**

Varias personas pueden operar y acceder a datos de manera simultánea. El bloqueo integrado impide el acceso simultáneo a los datos cuando se actualizan.

* **Seguridad integrada**

La seguridad basada en roles garantiza que el acceso a los datos esté limitado a usuarios específicos.

* **Normalización de bases de datos**

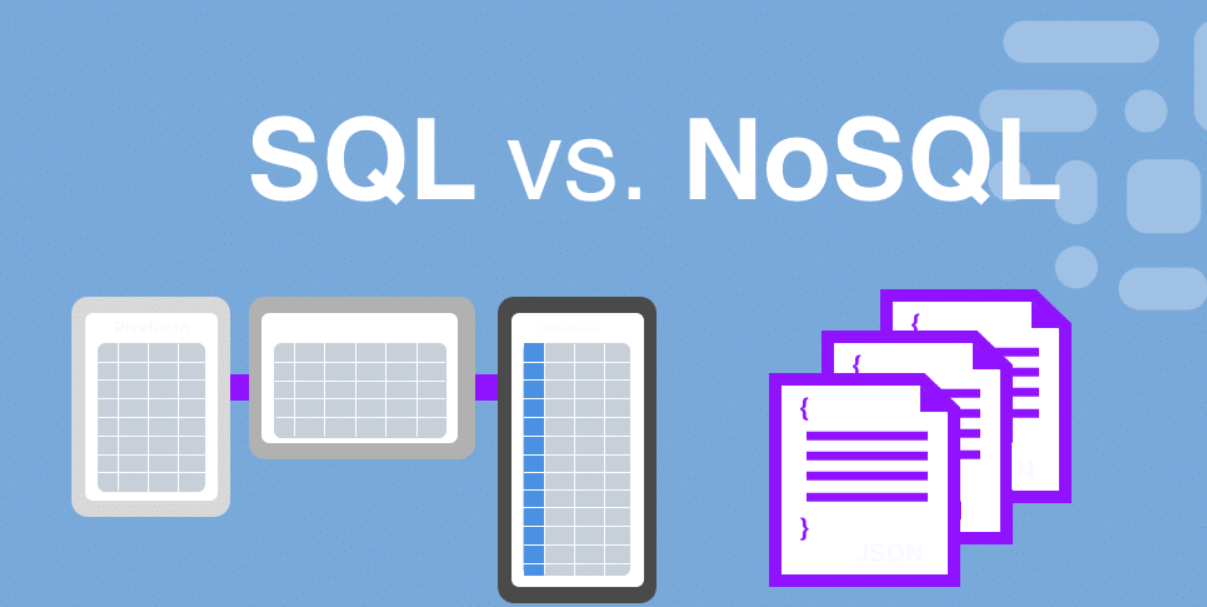
Las bases de datos relacionales usan una técnica de diseño conocida como normalización que reduce la redundancia de los datos y mejora su integridad.

### 3.7.1.2 No Relacional

En contraste, las bases de datos no relacionales, comúnmente denominadas NoSQL, están diseñadas para superar las limitaciones de los sistemas relacionales en escenarios de gran volumen de datos, no se preocupan de la normalización ni de las llaves foráneas, por lo que la integridad de la información está en riesgo. La estructura de la información pasa a un segundo plano y tienen como prioridad la velocidad. Estas bases de datos se agrupan en diversas categorías, tales como (Saltos Pérez, 2022):

* Almacenes clave-valor: optimizados para búsquedas rápidas basadas en una clave única.
* Bases de datos orientadas a documentos: que permiten almacenar datos semi-estructurados en formatos como JSON o XML.
* Bases de datos en columnas: ideales para análisis de grandes volúmenes de datos distribuidos.
* Bases de datos de grafos: orientadas a modelar relaciones complejas entre entidades.
* La flexibilidad y escalabilidad horizontal de las bases de datos NoSQL las convierten en una opción preferente para aplicaciones que requieren un procesamiento rápido y eficiente de datos en entornos de *Big Data* y aplicaciones en tiempo real (Saltos Pérez, 2022).

Podemos ver la notable diferencia de SQL quien relaciona los datos, con NoSQL que se trata de archivos unitarios como se ve en la



###### Figura 3.7.1.2 Base de datos no relacional (salud electronica, 2022)

En entornos de *Big Data*, las bases de datos NoSQL permiten analizar y procesar información en tiempo real, lo que resulta esencial para aplicaciones como la minería de datos, el análisis predictivo y los sistemas de recomendación. La adecuada elección y configuración de una base de datos, ya sea relacional o no relacional, repercute directamente en el rendimiento, la escalabilidad y la seguridad de la solución implementada (Saltos Pérez, 2022). La integración de estas tecnologías es, por tanto, un factor clave en el desarrollo de sistemas informáticos modernos que responden a las demandas de la era digital.

## 3.8 Plataforma de desarrollo

Son herramientas que permiten a los desarrolladores crear aplicaciones y programas de software. Proporcionan un entorno de desarrollo integrado que contiene todas las herramientas necesarias para crear, probar y desplegar aplicaciones.

### 

###### Figura 3.8 Tipos de Plataformas (Nunez, 2024)

### 3.8.1 Tipos de plataformas

Las plataformas se pueden englobar en diferentes grupos en función de su naturaleza.

* **Nativas**

Son aquellas que se pueden desarrollar de modo específico para cada sistema operativo; iOS, Android o Windows Phone.Se adapta a cada SO el lenguaje de desarrollo (Nunez, 2024).

* **Híbridas**

Son aquellas que juntan aspectos de las apps nativas y web, en función de las necesidades exactas. Se desarrollan con los lenguajes de programación Javascript, CSS o HTML, del mismo modo que las aplicaciones web,se adaptan a cualquier SO (Nunez, 2024).

* **Web / HTML5**

Se desarrollan con los lenguajes de programación Javascript, CSS o HTML; tres de los más populares a nivel global. Estas apps son compatibles y se adaptan a todos los sistemas operativos.

* Con HTML5 los programadores y desarrolladores pueden crear aplicaciones basadas en la web que los usuarios pueden ejecutar desde cualquier dispositivo móvil a través del navegador web del propio terminal.
  + **Aplicación móvil**
    - Es el proceso de creación de software para smartphones, tablets y asistentes digitales, más comúnmente para los sistemas operativos Android e iOS (Amazon, 2024).
  + **Aplicación web**
    - Hace referencia a una aplicación a la que se puede acceder desde el navegador de un dispositivo móvil o de escritorio (IBM, 2023).
  + **Aplicación de escritorio**
    - Es un programa de software creado para ejecutarse en un sistema informático con y sin acceso a Internet y navegador (Bustos, 2021).

### 3.8.2 App móvil

Una aplicación móvil, también llamada app móvil, es un tipo de aplicación diseñada para ejecutarse en un dispositivo móvil, que puede ser un teléfono inteligente o una tableta. Incluso si las aplicaciones suelen ser pequeñas unidades de software con funciones limitadas, se las arreglan para proporcionar a los usuarios servicios y experiencias de calidad. (Herazo, 2022)

### 3.8.2.1 Como se construye

El desarrollo de una aplicación móvil implica un proceso integral que abarca desde el diseño de la interfaz de usuario, la programación del sistema y la implementación de funcionalidades, hasta la realización de pruebas y su posterior publicación en plataformas digitales. Este conjunto de actividades tiene como objetivo ofrecer un producto funcional, accesible y con una experiencia de usuario satisfactoria. El desarrollo de aplicaciones móviles requiere una combinación de planificación, codificación, pruebas rigurosas y mantenimiento continuo para garantizar su eficacia y adaptabilidad en distintos entornos operativos (IBM, 2024).

Las aplicaciones móviles se desarrollan generalmente para una o más plataformas operativas, entre las que destacan:

* Android, utilizando principalmente los lenguajes de programación Java o Kotlin (Google Developers Training team, 2025).
* iOS, desarrollado comúnmente en Swift o Objective-C (Developer, 2025).
* Multiplataforma, mediante herramientas como Flutter, React Native o Xamarin, que permiten crear aplicaciones compatibles con múltiples sistemas operativos desde una sola base de código (Flutter, 2025.; Microsoft Build, 2025).

Esta elección tecnológica depende de factores como los recursos disponibles, los requerimientos del proyecto, el rendimiento deseado y la audiencia objetivo.

### 3.8.2.2 Fases de desarollo

El desarrollo de aplicaciones móviles constituye un proceso complejo y estructurado que abarca diversas etapas, desde la planificación inicial hasta el mantenimiento al lanzamiento. Este ciclo tiene como objetivo principal que la aplicación cumpla con los requerimientos técnicos, mantenga un rendimiento y sea sostenible (IBM, 2024. ; Google Developers Training team, 2025).

1. Análisis y planificación
   1. En esta etapa se definen los objetivos del proyecto mediante la identificación de las necesidades del usuario, el análisis de la competencia y la especificación de los requisitos técnicos y funcionales. Una planificación adecuada reduce riesgos y optimiza recursos (IBM, 2024).
2. Diseño de interfaz y experiencia de usuario (UI/UX)
   1. Se elaboran bocetos y se diseña la interfaz considerando principios de usabilidad, accesibilidad y navegación intuitiva. El enfoque centrado en el usuario favorece la aceptación de la aplicación (Interaction Design Foundation, 2025).
3. Desarrollo de la aplicación
   1. Incluye la implementación del frontend (interfaz gráfica) y backend (servidores, bases de datos, lógica del negocio), así como la integración de servicios externos según las necesidades del proyecto (Google Developers Training team, 2025.; IBM, 2024).
4. Pruebas y aseguramiento de calidad
   1. Se realizan pruebas funcionales, de rendimiento, usabilidad y seguridad, así como pruebas de compatibilidad en distintos dispositivos, con el fin de asegurar el cumplimiento de los estándares establecidos (ISO, 2024).
5. Despliegue
   1. La aplicación se lanza en plataformas como Google Play Store o App Store, cumpliendo con los requisitos técnicos, visuales y legales de cada tienda (Google Developers Training team, 2025).
6. Mantenimiento y actualización
   1. Tras su publicación, se realiza un mantenimiento continuo que incluye la corrección de errores, la incorporación de mejoras y la adaptación a nuevas versiones del sistema operativo (IBM, 2024).

### 3.8.2.3 Metodología para aplicaciones móviles.

El desarrollo de aplicaciones móviles puede abordarse mediante distintas metodologías, cada una con enfoques particulares según el tipo de proyecto, el equipo y el entorno de trabajo.

* **Cascada (Waterfall)**
  + Es un modelo lineal y secuencial que requiere completar cada fase antes de avanzar a la siguiente. Es apropiado para proyectos con requisitos bien definidos desde el inicio, ya que facilita una planificación estructurada y controlada (Winston W., 1970).
* **Agile**
  + Metodología iterativa e incremental que promueve la flexibilidad, la colaboración continua y la entrega frecuente de funcionalidades. Permite adaptarse a cambios durante el desarrollo, lo cual resulta útil en contextos dinámicos (Beck et al., 2001).
* **Scrum**
  + Marco de trabajo ágil que estructura el desarrollo en ciclos cortos llamados sprints. Define roles específicos y promueve reuniones diarias para facilitar la comunicación y el seguimiento del progreso (Sutherland & Schwaber, 2020).
* **Kanban**
  + Sistema visual de gestión del trabajo basado en tableros que representan el estado de cada tarea. Es útil para proyectos con flujo de trabajo continuo, ya que permite una supervisión flexible y en tiempo real (Schön et al., 2010).
* **Extreme Programming (XP)**
  + Extreme Programming es una metodología ágil que enfatiza prácticas técnicas como la programación en pareja (pair programming), la integración continua, el diseño simple y la obtención de retroalimentación rápida. Está orientada a equipos altamente colaborativos y a proyectos en los que los requerimientos pueden cambiar con frecuencia (Beck, 1999).
* **Scrumban**
  + Scrumban es una metodología ágil híbrida que combina elementos esenciales de Scrum y Kanban, con el objetivo de optimizar la planificación, ejecución y gestión del desarrollo de software (Ladas, 2009).

### 3.8.4 Api

Las API son mecanismos que permiten a dos componentes de software comunicarse entre sí mediante un conjunto de definiciones y protocolos (Amazon, 2024).

* **Google Maps SDK for Android**
  + Conjunto de herramientas proporcionado por Google que permite integrar mapas interactivos, geolocalización y navegación guiada en aplicaciones Android. Requiere conexión a internet y está sujeto a límites de uso gratuitos (*Documentación De Google Maps Platform*, 2024).
* **Mapbox SDK for Android**
  + Plataforma de desarrollo cartográfico que ofrece mapas personalizables, navegación avanzada y soporte offline. Es flexible, moderna y con modelo de uso gratuito limitado (*Maps SDK | Android Docs | Mapbox*, 2023).
* **OpenStreetMap (OSM)**
  + Plataforma cartográfica de código abierto que proporciona datos geográficos libres y editables, utilizados en aplicaciones de navegación, mapas interactivos y análisis espacial. Su contenido es generado y actualizado colaborativamente por una comunidad global de usuarios (*OpenStreetMap (OSM)*, 2022).

# Capítulo 4. Análisis

Con la finalidad de desarrollar los componentes técnicos y la arquitectura del prototipo de aplicación móvil para migrantes en tránsito desde la frontera sur de México hasta la frontera norte a través de comedores y albergues proporcionados por servicios humanitarios. Se describen elementos clave como la estructura del proyecto, los requerimientos, las tecnologías empleadas y el funcionamiento interno del sistema. Esta evaluación ofrece una perspectiva clara sobre la arquitectura y el razonamiento del sistema, que sirve como base para comprender su ejecución y funcionamiento.

## 4.2 Reglas de negocio

Las reglas de negocio se establecen para definir cómo deben realizarse los procesos de negocio. Pretenden ser instrucciones específicas que describen cómo deben completarse las actividades empresariales. Las reglas de negocio añaden un cierto nivel de control a los procesos empresariales cuando se aplican correctamente, reducen las incoherencias, los malentendidos y los errores (Klippa, 20231) Estos son descritos en la *tabla 4.2.1.*

##### Tabla 4.2.1 Reglas de Negocio

| **RN01** | **Disponibilidad de información** | Solo los usuarios autenticados podrán visualizar rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad El rol Administrador podrá además crear, editar e inhabilitar registros. |
| --- | --- | --- |
| **RN02** | **Frecuencia de actualización** | La base de datos sobre rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad será actualizada mensualmente con datos de criminalidad y datos de la ONG. |
| **RN03** | **Privacidad de los datos** | No se almacenará información sensible, como números telefónicos, lugar de procedencia, año de nacimiento o CURP de los usuarios. |
| **RN04** | **Acceso autorizado** | Los usuarios sólo podrán visualizar la información en permitida, asegurando la integridad de los datos. |
| **RN05** | **Gestión de ONG y centros de ayuda** | Solo las organizaciones verificadas podrán ser registradas en la base de datos, y sus datos deberán actualizarse al menos cada tres meses. |
| **RN06** | **Conectividad y sincronización** | El prototipo de aplicación requerirá conexión a internet periódicamente para acceder a la información y realizar el registro de usuario. |
| **RN07** | **Generación de ruta** | El prototipo genera rutas para minimizar el recorrido a la próxima ONG |
| **RN08** | **Visualización de mapa con trazado de ruta** | El mapa debe centrarse automáticamente en el origen y destino, mostrar la ruta resaltada hacia los marcadores de servicios humanitarios. |
| **RN09** | **Visualización de criminalidad en el mapa** | El mapa debe mostrar los lugares con mayor criminalidad usando una métrica de 3 (Alto medio y bajo) entre cada municipio. |
| **RN10** | **Ficha informativa de la ONG** | Los usuarios pueden desplegar una ficha informativa de la ong al hacer click, con datos como, municipio,estado, tipo (albergue o comedor) y el nombre del centro de ayuda. |

## 4.3 Requerimientos

Los requerimientos del prototipo son esenciales para definir el alcance y las funcionalidades que debe cumplir el prototipo aplicación móvil para migrantes en tránsito. Estos requerimientos guían el proceso de diseño y desarrollo, asegurando que el prototipo cumpla con los objetivos establecidos.

### 4.3.1 Requerimientos funcionales

Son especificaciones detalladas que definen las acciones, comportamientos y funcionalidades que un sistema debe ser capaz de realizar para cumplir con su propósito previsto (Visure solutions, 2025) Como se visualiza en la *tabla 4.3.1.1.*

##### Tabla 4.3.1.1 Requerimientos Funcionales

| ID | Categoría | Descripción |
| --- | --- | --- |
| RF1 | Registro de Usuario | El prototipo debe proporcionar una interfaz de registro con campos para: nombre de usuario, contraseña y confirmación de contraseña. |
| RF2 | Validación de Contraseña | El prototipo debe validar que las contraseñas tengan al menos 8 caracteres, incluyen 1 mayúscula, 1 número y 1 carácter especial, y coincidan con la confirmación. |
| RF3 | Autenticación de usuario. | El prototipo debe permitir el inicio de sesión mediante credenciales (usuario y contraseña) almacenadas en la base de datos, mostrando mensajes claros en caso de error. |
| RF4 | Generación de Rutas | El prototipo debe mostrar rutas hacia la siguiente ONG |
| RF5 | Geolocalización | El prototipo debe actualizar la ubicación del usuario al apretar un botón, para recalcular la ruta a la siguiente ONG |
| RF6 | Actualización de Datos | El prototipo se debe actualizar mensualmente con los datos de criminalidad y ubicaciones de ONG. |
| RF7 | Visualización de ONG | El prototipo debe permitir a los usuarios ver fichas informativas de las ONG’ s en su trayecto. |
| RF8 | Algoritmos de ML | El prototipo debe utilizar K means para agrupar los datos de criminalidad en 3 (Alto, medio y bajo) y mostrar estos en el mapa según el municipio. |
| RF9 | Base de Datos (Criminalidad) | La BD debe almacenar registros actualizados de robos y secuestros por estado, con fechas y zonas de riesgo. |
| RF10 | Base de Datos (ONG) | La BD debe contener ubicación, tipo (albergue/comedor), y el lugar donde se encuentran. |

##### 

### 4.3.2 Requerimientos no funcionales

Se refieren a los atributos de calidad de un sistema que definen cómo funciona en lugar de qué hace. A diferencia de los requisitos funcionales, que especifican las acciones y tareas que debe realizar un sistema, los requisitos no funcionales se centran en las características generales y el comportamiento del sistema en diversas condiciones. Abordan aspectos como el rendimiento, la facilidad de uso, la fiabilidad y la escalabilidad, garantizando que el sistema cumpla con los estándares de calidad y se visualizan en la *tabla 4.3.2.1.*

##### Tabla 4.3.2.1 Requerimientos no Funcionales

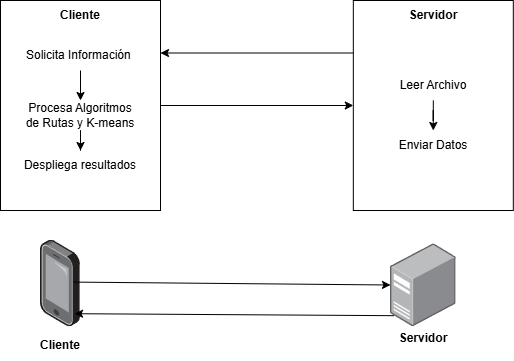
| ID | Categoría | Descripción |
| --- | --- | --- |
| RNF 1 | Rendimiento | El prototipo debe soportar al menos 100 usuarios concurrentes. |
| RNF2 | Compatibilidad | El prototipo debe funcionar en Android. |
| RNF 3 | Tiempo de Respuesta | El prototipo debe generar rutas con conexión estable. |
| RNF4 | Uso de datos | Los datos de ubicación deben solo usarse para cálculo de rutas. |
| RNF5 | Actualizaciones | El prototipo debe recibir actualizaciones mensuales de:  • Datos de criminalidad  • Datos de las ONG’ s |
| RNF 6 | UX/UI | La interfaz del prototipo incorpora iconos, secuencias visuales y textos breves. |
| RNF 7 | Escalabilidad | El prototipo debe estar preparado para mantener su rendimiento ante un aumento de usuarios. |

# Capítulo 5. Diseño

### 5.1 Diagrama de cliente-servidor

El modelo cliente-servidor tiene algunos rasgos característicos. Hay una clara distribución de tareas entre los clientes y los servidores. El servidor es el responsable de proporcionar los servicios. Se encarga de ejecutar los servicios solicitados y entrega la respuesta esperada. El cliente, en cambio, utiliza y solicita los servicios proporcionados. Finalmente, recibe la respuesta del servidor (IONOS, 2023).

En el modelo cliente-servidor, un servidor sirve a varios clientes y, por ende, procesa múltiples peticiones de diferentes clientes. Para ello, presta su servicio de forma permanente y pasiva. Por su parte, el cliente solicita activamente los servicios del servidor e inicia las tareas del servidor como se puede visualizar en la *figura 5.1*.1



###### Figura 5.1.1 Diagrama de Clientes-servidor del prototipo

El diagrama *figura 5.1.1* muestra una arquitectura cliente‑servidor donde la aplicación móvil inicia las solicitudes de los datos procesados, mientras que el servidor se encarga únicamente de suministrar la información en desde la base de datos.

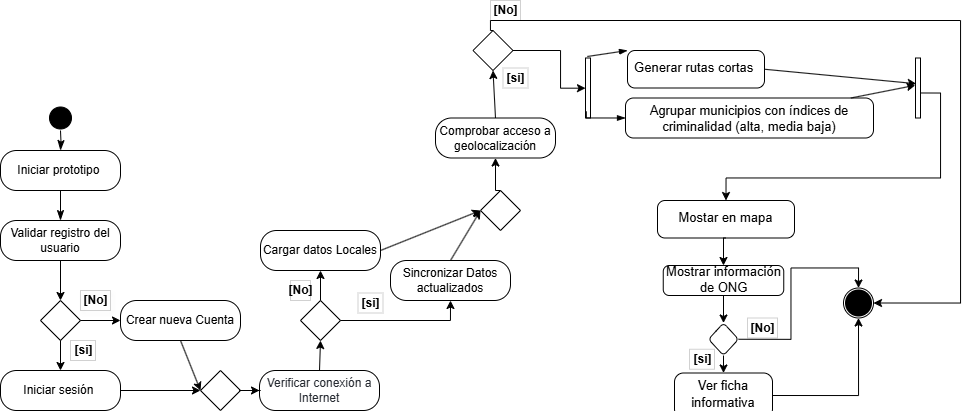
El componente “solicita información” del cliente envía una petición al servidor, el servidor ejecuta el componente “lee archivo”, que accede a las tablas de PostgreSQL y seguidamente invoca “envía datos”, transmitiendo al cliente el conjunto de datos necesario.

Una vez recibida la respuesta, la app móvil “procesa (el algoritmo de búsqueda y el K‑means)”: aplica el algoritmo A\* para el cálculo de rutas óptimas y la técnica K‑means para la agrupación de puntos de riesgo (en alto, medio y bajo) para mostrarlos en el mapa de la república Méxicana. Por último, el cliente despliega los resultados generados en su interfaz de usuario mediante el componente “despliega resultados”.

### 5.1.1 Diagrama de actividades

La UML es un lenguaje de modelado abierto que permite a los desarrolladores ver los productos de su trabajo en diagramas estandarizados. Se trata de una notación independiente de procesos. No es una metodología de desarrollo, UML no dice que hacer primero ni qué hacer después o cómo diseñar un sistema, en cambio, ayuda a visualizar el diseño y la comunicación entre objetos (PMOinformatica, 2024).

El diagrama de actividades es un tipo de diagrama UML (Lenguaje Unificado de Modelado) que plasma el flujo de trabajo o la secuencia de actividades de un sistema, sirve para mostrar cómo se realizan la tareas del programa, en qué orden, y qué decisiones o bifurcaciones se presentan durante el flujo, como se visualiza en la *figura 5.1.1.1*

**

###### Figura 5.1.1.1 Diagrama de actividades

El flujo comienza cuando el migrante inicia la aplicación móvil y, tras la pantalla de bienvenida, se comprueba si ya dispone de una cuenta; en caso contrario se le invita a registrarse y, una vez completado el registro o la autenticación.

La aplicación verifica la conexión a Internet, si está en línea, se sincronizan automáticamente las últimas actualizaciones de rutas, niveles de criminalidad y datos de las ONG; si no, se activa el modo offline cargando la información previamente guardada en el dispositivo.

La app solicita permiso para acceder a la geolocalización; al otorgarse, el prototipo hace dos cosas de forma paralela:

* El prototipo agrupa las regiones según sus índices de criminalidad (alto, medio y bajo) mediante K-means para colorear el mapa que visualiza el usuario.
* El prototipo cula las rutas con los algoritmos como A\* y Dijkstra, priorizando trayectos cortos a la siguiente ONG.

El usuario está en la opción de pulsar sobre cualquier punto de la ONG para desplegar una ficha detallada con descripción del centro de ayuda.

### 5.2 Casos de uso

El modelo de casos de uso describe el comportamiento que tiene el software en respuesta a eventos realizados por actores externos. También, describe acontecimientos iniciados por el sistema hacia los usuarios (PMOinformatica, 2024) y es parte de lo que se desarrollará en este apartado respecto al prototipo propuesto.

### 5.2.1 Diagrama de Casos de uso

El diagrama de casos de uso muestra de forma gráfica las características del software (sus funcionalidades) (PMOinformatica, 2024) observable en la *figura 5.2.1.1*

## 

###### Figura 5.2.1.1 Diagrama de Casos de Uso

### 5.2.2 Actores

Los roles que los usuarios desempeñan en estos casos de uso. A estos roles se les denomina “Actores” y se interrelación entre los elementos del diagrama definiendo sus interacciones específicas con el prototipo (PMOinformatica, 2024). Un mismo actor puede desempeñar varios papeles según el caso de uso con que interactúa.

En el Prototipo propuesto desarrollamos los roles de nuestros actores en las siguientes tablas:

##### 

##### Tabla 5.2.2.1 Descripcion de Actor Usuario

| **Actor: Usuario** | |
| --- | --- |
| **Atributo** | **Descripción** |
| Nombre | Usuario |
| Tipo | Actor primario (interactúa directamente con el prototipo de aplicación móvil) |
| Propósito | Registrarse, iniciar sesión y consultar rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad |
| Responsabilidades | Ingresar datos para registro ([CU1](#_muimw250gueh)), autenticarse , visualizar rutas ([CU5](#_4fz88t609ln5)) y consultar detalles de servicios |
| Interacciones con el prototipo | Solicitar creación de cuenta, enviar credenciales, recibir alertas de peligros en ruta, visualizar mapa, enviar evaluaciones de ONG |
| Precondiciones | El prototipo de aplicación está operativa y hay conexión a internet; usuario no registrado previamente |
| Postcondiciones | Cuenta creada o mensajes de error; sesión iniciada o acceso denegado |

##### Tabla 5.2.2.1 Descripcion de Actor Administrador

| **Actor: Administrador** | |
| --- | --- |
| **Atributo** | **Descripción** |
| Nombre | Administrador |
| Tipo | Actor secundario (gestiona datos y configuración del prototipo) |
| Propósito | Mantener la base de datos de rutas y ONG actualizada, gestionar usuarios, revisar evaluaciones y generar reportes de uso |
| Responsabilidades | Validar y actualizar datos de criminalidad y ONG ([RN02, RN05](#_s4c4093iyarf)), aprobar o inhabilitar ONG ([CU7](#_xga0a9myijcm)). |
| Interacciones con el prototipo | Acceder al panel de administración, lanzar procesos de actualización mensual, modificar datos en BD. |
| Precondiciones | Cuenta de administrador activa; permisos suficientes; conexión a internet |
| Postcondiciones | Datos de rutas y ONG actualizados y usuarios registrados |

### 5.2.3 Especificación de Casos de Uso

En el lenguaje de modelado unificado (UML por sus siglas en inglés), los [**casos de uso**](http://www.utm.mx/~caff/doc/OpenUPWeb/openup/guidances/concepts/use_case_model_CD178AF9.html) son modelos que describen como diferentes tipos de usuarios interactúan con un sistema informático para resolver un determinado problema o necesidades. Por lo tanto, describen los objetivos de los usuarios, las interacciones entre los usuarios y el sistema, así como el comportamiento que debe ejecutar el sistema para satisfacerlos (PMOinformatica, 2024).

Para el modelado del prototipo propuesto se visualizan las siguientes tablas para cada caso de uso visto en la figura 5.2.1.1

| **Caso de Uso CU1: Crear Cuenta** | |
| --- | --- |
| **Versión:** | 0.3 |
| Autor: | Daniela Mendoza Bedolla |
| Supervisa: | Sabrina Michaus Gutierrez y Angel Dehesa Nieves |
| Actor: | Usuario |
| Resumen: | |
| El usuario crea una cuenta en el prototipo de aplicación móvil, almacenando la información en la base de datos para poder acceder posteriormente a las funcionalidades disponibles. | |
| **Propósito:** | |
| Permitir que nuevos usuarios se registren en el prototipo de aplicación proporcionando datos específicos para acceder a las funcionalidades disponibles. | |
| **Entradas:** | |
| Nombre de Usuario | |
| Contraseña | |
| **Origen:** | |
| Usuario a través de la interfaz de registro en el prototipo del prototipo de aplicación móvil | |
| **Salidas:** | |
| Confirmación de creación de cuentas exitosamente | |
| Mensaje de error en caso de fallo | |
| **Destino:** | |
| El usuario recibe una notificación de éxito o error. | |
| Base de datos almacena la nueva cuenta | |
| **Precondiciones:** | |
| El prototipo de aplicación debe ser operativo. | |
| Se debe contar con conexión a internet | |
| **Postcondiciones:** | |
| La cuenta del usuario queda registrada en el prototipo | |
| **Reglas de Negocio:** | |
| [RN03, RN04, RN06.](#_s4c4093iyarf) | |

##### Tabla 5.2.3.1 Caso de Uso CU1: Crear Cuenta

##### 

| **Trayectoria Principal –Crear cuenta** | | | | |
| --- | --- | --- | --- | --- |
| **Paso** | **Descripción** | | | |
| 1 | El usuario abre el prototipo de aplicación y selecciona “Crear Cuenta”. | | | |
| 2 | El prototipo de aplicación muestra el formulario de registro con los campos necesarios. | | | |
| 3 | El usuario ingresa nombre de usuario, contraseña y confirma la contraseña. | | | |
| 4 | Si los datos no presentan duplicidad en el prototipo crea la cuenta RF2 y almacena la información en la base de datos. | | | |
| **Trayectoria Alternativa A – Credenciales incorrectas** | | | | |
| **Paso** | **Descripción** | | | |
| 2A | Nombre de usuario ya existe por ende se muestra el mensaje “El nombre de usuario ya existe” | | | |
| 2A.1 | La confirmación de contraseña no es igual a la contraseña “Contraseña no válida, confirmación incorrecta” | | | |
| 2A.2 | El usuario puede reintentar. | | | |
| **Trayectoria Alternativa B – Error del sistema** | | | | |
| **Paso** | **Descripción** | | | |
| 2B | Si la red falla, el prototipo muestra “Error de red. Verifica tu conexión.” | | | |
| 2B.1 | Si ocurre un fallo en la base de datos, el prototipo muestra “Error del sistema, inténtalo más tarde.” | | | |

##### Tabla 5.2.3.1 Caso de Uso CU1: Trayectorias.

##### 

| **Caso de Uso CU2: Iniciar sesión** | |
| --- | --- |
| **Versión:** | 0.3 |
| **Autor:** | Daniela Mendoza Bedolla |
| **Supervisa:** | Sabrina Michaus Gutierrez y Angel Dehesa Nieves |
| **Actor:** | Usuario |
| **Resumen:** | |
| El usuario registrado ingresa sus credenciales (nombre de usuario y contraseña), el prototipo las verifica contra la base de datos y, si son correctas y activa la sesión. | |
| **Propósito:** | |
| Permitir que los usuarios autenticados accedan al prototipo de aplicación con sus credenciales. | |
| **Entradas:** | |
| Nombre de Usuario | |
| Contraseña | |
| **Origen:** | |
| Usuario realizó el registro en el prototipo del prototipo de aplicación móvil [CU1](#_8zxp61h2ipwt). | |
| **Salidas:** | |
| Acceso exitoso a el prototipo de aplicación | |
| Mensaje de error en caso de credenciales incorrectas. | |
| **Destino:** | |
| Usuario accede a su cuenta y puede navegar en el prototipo de aplicación | |
| **Precondiciones:** | |
| El usuario debe de tener una cuenta registrada, [CU1](#_muimw250gueh). | |
| Debe haber conexión a internet | |
| **Postcondiciones:** | |
| El usuario tiene una sesión activa | |
| **Reglas de Negocio:** | |
| [RN01, RN04.](#_s4c4093iyarf) | |

##### Tabla 5.2.3.2 Caso de Uso CU2: Iniciar seción

##### 

| **Trayectoria Principal – Iniciar sesión** | | | | |
| --- | --- | --- | --- | --- |
| **Paso** | **Descripción** | | | |
| 1 | El Usuario abre el prototipo de aplicación e ingresa nombre de usuario y contraseña. | | | |
| 2 | El prototipo verifica las credenciales. | | | |
| 3 | Si son correctas, se activa la sesión del usuario. | | | |
| 4 | El usuario es redirigido a la pantalla principal del prototipo de aplicación. | | | |
| **Trayectoria Alternativa A – Credenciales incorrectas** | | | | |
| **Paso** | **Descripción** | | | |
| 2A | El prototipo detecta un error en el nombre de usuario o contraseña y muestra mensaje de error “Usuario o Contraseña incorrectos vuelva a intentar” | | | |
| 2A.1 | El usuario puede reintentar. | | | |
| **Trayectoria Alternativa B – Cuenta no registrada** | | | | |
| **Paso** | **Descripción** | | | |
| 2B | El nombre de usuario no existe en la base de datos, muestra mensaje “Usuario no existente” | | | |
| 2B.1 | Se sugiere al usuario registrarse en un mensaje “Comienza tu registro” | | | |
| **Trayectoria Alternativa C – Error del sistema** | | | | |
| **Paso** | **Descripción** | | | |
| 2C | El prototipo presenta un fallo interno. | | | |
| 2C.1 | Se notifica al usuario y se recomienda intentar más tarde con el mensaje “Error del sistema, intentalo más tarde” | | | |

##### Tabla 5.2.3.2 Caso de Uso CU2: Trayectorias

##### 

| **Caso de Uso CU3: Visualizar Mapa** | |
| --- | --- |
| **Versión:** | 0.3 |
| **Autor:** | Sabrina Michaus Gutierrez |
| **Supervisa:** | Dabiela Mendoza Bedolla y Angel Dehesa Nieves |
| **Actor:** | Usuario |
| **Resumen:** | |
| El Usuario puede abrir un Mapa en el prototipo de aplicación móvil, el cual está disponible en todo momento. | |
| **Propósito:** | |
| Permitir a los usuarios visualizar un mapa con rutas con el menor índice de criminalidad, centros de ayuda humanitaria (ONG: albergues, comedores) y zonas de riesgo basadas en datos de criminalidad. | |
| **Entradas:** | |
| Ubicación actual proporcionada por el dispositivo del usuario. | |
| Cargar rutas con el menor índice de criminalidad, centros de ayuda, etc. | |
| **Origen:** | |
| Información sobre criminalidad (robos y secuestros). | |
| Ubicación y detalles de ONG, albergues y comedores. | |
| **Salidas:** | |
| Mapa con rutas generadas por el prototipo. | |
| Ubicación y detalles de centros de ayuda humanitaria. | |
| **Destino:** | |
| Interfaz del prototipo de aplicación donde el usuario puede consultar la información. | |
| **Precondiciones:** | |
| El usuario debe tener acceso al prototipo de aplicación a través de un login [CU2](#_z4airkh5t48g). | |
| La base de datos debe contener información actualizada sobre rutas y centros de ayuda, [CU7](#_xga0a9myijcm). | |
| El usuario debe visualizar la información de las ONG's, [CU4](#_ocse84i7pxjf). | |
| El usuario debe de visualizar las rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad [CU5](#_4fz88t609ln5). | |
| **Postcondiciones:** | |
| El usuario visualiza en el mapa la ruta y los centros de ayuda. | |
| **Reglas de Negocio:** | |
| [RN07, RN08.](#_s4c4093iyarf) | |

##### Tabla 5.2.3.3 Caso de Uso CU3: Visualizar Mapas

##### 

| **Trayectoria Principal – Visualizar Mapas** | | | | |
| --- | --- | --- | --- | --- |
| **Paso** | **Descripción** | | | |
| 1 | El sistema desplegará una solicitud de acceso a la ubicación actual. | | | |
| 2 | El sistema consulta la base de datos y carga los datos. | | | |
| 3 | El sistema selecciona una ruta con el menor índice de criminalidad. | | | |
| 4 | Se despliega la ruta en el mapa. | | | |
| **Trayectoria Alternativa A – Sin conexión a internet** | | | | |
| **Paso** | **Descripción** | | | |
| 2A | El sistema detecta la falta de conexión y arroja un mensaje “Error de red. Verifica tu conexión.” | | | |
| **Trayectoria Alternativa B – Falla en carga de datos** | | | | |
| **Paso** | **Descripción** | | | |
| 3A | Los datos de criminalidad o ubicación de ONG no se pueden recuperar. | | | |
| 3A.1 | El sistema muestra un mapa limitado o sin elementos, arrojando un mensaje “Error de carga de datos” | | | |

##### Tabla 5.2.3.3 Caso de Uso CU3: Trayectorias

##### 

| **Caso de Uso CU4: Visualizar información general de la siguiente ONG** | |
| --- | --- |
| **Versión:** | 0.3 |
| **Autor:** | Angel Dehesa Nieves |
| **Supervisa:** | Dabiela Mendoza Bedolla y Sabrina Michaus Gutierrez |
| **Actor:** | Usuario |
| **Resumen:** | |
| El Usuario puede revisar el estatus de la ONG de su trayecto, para conocer su valoración y si este es albergue o Comedor comunitario | |
| **Propósito:** | |
| Permitir a los usuarios consultar información detallada sobre la ONG disponible en su trayecto. | |
| **Entradas:** | |
| Selección de ONG en el mapa. | |
| **Origen:** | |
| Selección en el mapa. | |
| **Salidas:** | |
| Información sobre la ONG (nombre, ubicación y servicios ofrecidos). | |
| Evaluaciones de otros usuarios. | |
| **Destino:** | |
| Pantalla de información de ONG. | |
| **Precondiciones:** | |
| El centro de ayuda debe visualizarse en el mapa, [CU3](#_x05s5jbg0ugw). | |
| El usuario debe estar autorizado en la plataforma cumpliendo previamente el [CU2](#_z4airkh5t48g). | |
| **Postcondiciones:** | |
| El usuario obtiene datos relevantes de la ONG. | |
| **Reglas de Negocio:** | |
| [RN01, RN05.](#_s4c4093iyarf) | |

##### Tabla 5.2.3.4 Caso de Uso CU4: Visualizar información de la ONG

##### 

| **Trayectoria Principal** | | | | |
| --- | --- | --- | --- | --- |
| **Paso** | **Descripción** | | | |
| 1 | El Usuario accede al mapa. | | | |
| 2 | Selecciona la ONG. | | | |
| 3 | El sistema muestra los datos detallados: ubicación, tipo y nombre | | | |
| 4 | Se muestran evaluaciones de otros usuarios. | | | |
| **Trayectoria Alternativa A – ONG no disponible** | | | | |
| **Paso** | **Descripción** | | | |
| 2A | La ONG seleccionada ya no está activa se muestra el mensaje “La ONG ya no está operativa”. | | | |
| **Trayectoria Alternativa B – Error en la base de datos** | | | | |
| **Paso** | **Descripción** | | | |
| 3A | La información no puede ser recuperada y se muestra un mensaje “Ocurrió un error intentelo mas tarde” | | | |

##### Tabla 5.2.2.4 Caso de Uso CU4: Trayectorias

##### 

| **Caso de Uso CU5: Generación de ruta** | |
| --- | --- |
| **Versión:** | 0.2 |
| **Autor:** | Angel Dehesa Nieves |
| **Supervisa:** | Dabiela Mendoza Bedolla y Sabrina Michaus Gutierrez |
| **Actor:** | Usuario / Administrador |
| **Resumen:** | |
| Tanto el Usuario como el administrador del sistema pueden ver las rutas generadas por el algoritmo de Machine learning. | |
| **Propósito:** | |
| Proporcionar a los usuarios rutas usando algoritmos de *Machine Learning* y datos históricos. | |
| **Entradas:** | |
| Ubicación del usuario. | |
| **Origen:** | |
| Ingreso de la ubicación por el dispositivo a la base de datos. | |
| **Salidas:** | |
| Mapa con rutas basadas en menor índice de criminalidad. | |
| Alertas de seguridad en caso de que la ruta tenga alto índice de criminalidad. | |
| Siguiente ONG cercana en la ruta. | |
| **Destino:** | |
| Pantalla de visualización de rutas. | |
| **Precondiciones** | |
| El usuario debe estar autenticado en la plataforma como en el [CU2](#_85ubsxwz0f5q). | |
| El prototipo de aplicación debe contar con acceso a la base de datos actualizada sobre índices de criminalidad como lo indica el [CU5](#_4fz88t609ln5). | |
| **Postcondiciones:** | |
| El usuario obtiene datos relevantes de la ONG para su trayecto. | |
| **Reglas de Negocio:** | |
| [RN01, RN08.](#_s4c4093iyarf) | |

##### Tabla 5.2.2.5 Caso de Uso CU5: Generación de ruta

##### 

| **Trayectoria Principal** | | | | |
| --- | --- | --- | --- | --- |
| **Paso** | **Descripción** | | | |
| 1 | El Usuario inicia sesión y accede al prototipo de app. | | | |
| 2 | Se solicita su permiso de ubicación de ser necesario, si no el dispositivo arroja ubicación de forma automática. | | | |
| 3 | El sistema procesa la ruta utilizando el algoritmo de ML. | | | |
| 4 | Se genera una ruta basada en menor índice de criminalidad. | | | |
| 5 | Se muestran alertas de seguridad y el centro de ayuda cercano. | | | |
| **Trayectoria Alternativa A – No se encuentra una rutas con servicios humanitarios y un mapa que muestre el índice de criminalidad** | | | | |
| **Paso** | **Descripción** | | | |
| 4A | El sistema no puede generar una ruta con el menor índice de criminalidad, se muestra mensaje “Ocurrió un error en el cálculo de la ruta” | | | |
| **Trayectoria Alternativa B – Error en el algoritmo** | | | | |
| **Paso** | **Descripción** | | | |
| 3A | El algoritmo falla o no responde, Se muestra un mensaje de error “intenta más tarde”. | | | |

##### Tabla 5.2.2.5 Caso de Uso CU5: Trayectoria

##### 

| **Caso de Uso CU6: Visualizar listado de las ONG's en el sistema** | |
| --- | --- |
| **Versión:** | 0.3 |
| **Autor:** | Angel Dehesa Nieves |
| **Supervisa:** | Dabiela Mendoza Bedolla y Sabrina Michaus Gutierrez |
| **Actor:** | Administrador |
| **Resumen:** | |
| El administrador del sistema puede visualizar todas las ONG´s registradas en el sistema. | |
| **Propósito:** | |
| Mostrar en el mapa la ONG cercana según la ruta, todas las ONG’ s se encuentran en la base de datos. | |
| **Entradas:** | |
| Ninguna (se carga automáticamente al entrar en la sección correspondiente). | |
| **Origen:** | |
| Base de datos de ONG’ s. | |
| **Salidas:** | |
| Muestra la siguiente ONG. | |
| El Mapa muestra información de la ONG. | |
| **Destino:** | |
| Pantalla de listado de la BD. | |
| **Precondiciones:** | |
| Se deben haber recopilado datos actualizados sobre ONG 's (Albergues y Comedores) como indica el [CU7](#_xga0a9myijcm). | |
| **Postcondiciones:** | |
| El administrador puede consultar y seleccionar ONG. | |
| **Reglas de Negocio:** | |
| [RN01,RN05, RN08.](#_s4c4093iyarf) | |

##### Tabla 5.2.2.6 Caso de Uso CU6: Visualizar ONG’s en el sistema

##### 

| **Trayectoria Principal – Visualización de ONG por el Administrador** | | | | |
| --- | --- | --- | --- | --- |
| **Paso** | **Descripción** | | | |
| 1 | El Administrador accede al sistema y se autentica. | | | |
| 2 | Selecciona la opción “Visualizar ONG 's”. | | | |
| 3 | El sistema consulta la base de datos de ONG 's. | | | |
| 4 | Se despliega una lista con todas las ONG registradas. | | | |
| 5 | Rectifica el mapa actualizado con su ubicación. | | | |
| **Trayectoria Alternativa A – No hay ONG registradas** | | | | |
| **Paso** | **Descripción** | | | |
| 4A | La base de datos no contiene registros de ONG. | | | |
| 4A.1 | Hay errores en subir o modificar los datos. | | | |
| **Trayectoria Alternativa B – Fallo en carga del mapa** | | | | |
| **Paso** | Descripción | | | |
| 5A | El sistema no puede cargar el mapa. | | | |
| 5A.1 | El sistema no se actualizó correctamente. | | | |

##### Tabla 5.2.2.6 Caso de Uso CU6: Trayectorias

##### 

##### 

##### 

| **Caso de Uso CU7: Actualización de Ruta y Servicios BD** | |
| --- | --- |
| **Versión:** | 0.3 |
| **Autor:** | Sabrina Michaus Gutierrez |
| **Supervisa:** | Dabiela Mendoza Bedolla y Angel Dehesa Nieves |
| **Actor:** | Administrador |
| **Resumen:** | |
| Mensualmente se ingresará a la base de datos por parte del administrador para subir los nuevos datos de criminalidad por estado que el gobierno de méxico ha subido en su portal, así como podrá dar de baja o alta alguna ONG, dependiendo de si sigue operando o no. | |
| **Propósito:** | |
| Permitir la actualización de datos en la base de datos sobre rutas y servicios humanitarios. | |
| **Entradas:** | |
| Ninguna (se carga automáticamente al entrar en la sección correspondiente). | |
| **Origen:** | |
| Algoritmo de Machine Learning para rutas. | |
| Reportes oficiales o fuentes externas sobre criminalidad. | |
| Registros de ONG y centros de ayuda humanitaria. | |
| **Salidas:** | |
| Base de datos actualizada para las nuevas rutas. | |
| Información actualizada sobre criminalidad disponible para consulta. | |
| **Destino:** | |
| Usuarios del prototipo de aplicación que consultan rutas y servicios humanitarios. | |
| Algoritmo de Machine Learning que utilizará los datos de criminalidad actualizados. | |
| **Precondiciones:** | |
| La base de datos debe estar operativa y accesible. | |
| Se deben haber recopilado datos actualizados sobre criminalidad y servicios humanitarios. | |
| **Postcondiciones:** | |
| Los usuarios pueden visualizar rutas y centros de ayuda actualizados en [CU3](#_x05s5jbg0ugw). | |
| Los algoritmos de Machine Learning pueden operar con la información más reciente para la generación de rutas. | |
| **Reglas de Negocio:** | |
| [RN02, RN06,](#_s4c4093iyarf) | |

##### Tabla 5.2.2.8 Caso de Uso CU7: Actualización de ruta y servicios BD

| **Trayectoria Principal – Actualización de ruta y servicios BD** | | | | |
| --- | --- | --- | --- | --- |
| **Paso** | **Descripción** | | | |
| 1 | El Administrador accede al sistema. | | | |
| 2 | El Administrador actualiza los datos. | | | |
| 3 | El sistema recupera los datos más recientes del algoritmo de ML y fuentes oficiales. | | | |
| 5 | Se actualiza la base de datos con las nuevas rutas y datos de criminalidad. | | | |
| 6 | El sistema actualiza los registros de ONG activas e inactivas. | | | |
| 7 | Se muestra un mensaje de confirmación de actualización exitosa. | | | |
| **Trayectoria Alternativa A – Fallo en la conexión con fuentes externas** | | | | |
| **Paso** | **Descripción** | | | |
| 4A | El sistema no puede acceder a las fuentes de datos. | | | |
| 4A.1 | El administrador puede intentarlo nuevamente. | | | |
| **Trayectoria Alternativa B – Error en la base de datos** | | | | |
| **Paso** | **Descripción** | | | |
| 5A | La base de datos no responde o tiene un error. | | | |
| 5A.1 | El sistema detiene la operación. | | | |

##### Tabla 5.2.2.8 Caso de Uso CU8: Trayectorias

## Fuentes de datos

Para crear un sistema que apoye a las personas migrantes en su paso por México, es importante usar datos previamente analizados y seleccionados. Para lograr ese propósito compararemos las fuentes de datos encontradas para el uso del prototipo.

### Datos de Criminalidad

* **Incidencia delictiva (SESNSP)**
  + Los datos abiertos de incidencia delictiva publicados por el SESNSP incluyen cifras estatales y municipales desde 2015 hasta marzo 2025 actualizadas cada mes, así como registros de víctimas y delitos federales del Gobierno de México (Secretariado Ejecutivo del Sistema Nacional de Seguridad Pública, 2025).  
    La página ofrece diccionarios de datos y enlaces directos a archivos en formatos CSV y XLSX, además de un archivo de gran tamaño para descarga mediante Google Drive Gobierno de México.  
    En la página oficial de datos abiertos del gobierno, el SESNSP está registrado como entidad publicante, con 24 conjuntos de datos etiquetados bajo “incidencia delictiva” y formatos principalmente CSV. Este nivel de acceso y documentación garantiza transparencia y facilita la integración en el prototipo (Secretariado Ejecutivo del Sistema Nacional de Seguridad Pública, 2025).
* **INEGI – Incidencia Delictiva**
  + El Instituto Nacional de Estadística y Geografía (INEGI) publica datos abiertos sobre incidencia delictiva a nivel estatal y municipal desde 1997 hasta marzo de 2025, en formatos CSV y XLSX, con diccionarios detallados para cada serie temporal. Esta fuente permite analizar tendencias históricas y comparar diferentes entidades federativas con alta granularidad geográfica (INEGI, 2025).

Tabla ## Comparación de fuentes de datos de Criminalidad

| **Fuente** | **Años Cubiertos** | **Nivel Geográfico** | **Formato de Datos** | **Calidad / Confiabilidad** | **Actualización** | **Características** |
| --- | --- | --- | --- | --- | --- | --- |
| SESNSP | 2015 – marzo 2025 | Estatal y municipal | CSV, XLSX | Alta (metodología actualizada del fuero común) | Mensual | Incluye víctimas, delitos federales, diccionario de datos |
| INEGI | 1997 – marzo 2025 | Estatal y municipal | CSV, XLSX | Alta (institución estadística nacional) | Trimestral o anual | Análisis histórico y comparación interanual |

La documentación del SESNSP (Secretariado Ejecutivo del Sistema Nacional de Seguridad Pública) incluye una actualización de datos mensual que beneficia al prototipo mucho más que el análisis histórico de los datos de criminalidad.

### Datos de ONG’ s

* **Albergues y comedores (BBVA Research)**

El “Mapa 2020 de casas del migrante, albergues y comedores para migrantes en México” de BBVA Research es un PDF portable que lista las principales instalaciones activas en las rutas más transitadas, con georreferenciación y distancias aproximadas en kilómetros entre puntos clave (BBVA, 2020).

La fuente de este directorio incluye datos de Médicos Sin Fronteras (MSF), Organización Internacional para las Migraciones (OIM), REDODEM y CICR, actualizados a marzo 2020 BBVA Research.  
Aunque no se publica en CSV, el PDF puede extraerse y convertirse a formatos tabulares para su análisis y geocodificación (BBVA, 2020).

* **ACNUR México – Directorio de Albergues**

La Agencia de la ONU para los Refugiados (ACNUR) en México ofrece un directorio en línea de albergues y comedores que brindan asistencia humanitaria a personas solicitantes de asilo y migrantes. Este directorio incluye información detallada sobre los servicios ofrecidos, horarios de atención, requisitos de ingreso y, en muchos casos, coordenadas geográficas. Además, se proporciona un mapa interactivo que facilita la localización de estos centros en distintas regiones del país (ACNUR, 2024).

Tabla ## Comparación de datos de ONG’ s

| **Fuente** | **Fecha de Publicación** | **Formato de Datos** | **Referencia Geográfica** | **Contenido Principal** | **Características** |
| --- | --- | --- | --- | --- | --- |
| BBVA Research | marzo 2020 | PDF | Sí | Casas del migrante, albergues, comedores, distancias entre puntos clave | Riesgo de errores en transcripción al convertir PDF a CSV |
| ACNUR México | Actualizado 2024 | Web / Mapa interactivo | Sí | Servicios por albergue, ubicación, requisitos, horarios, servicios | Altamente detallado, ideal para integración con mapas |

Al recopilar información para el prototipo no se encontraron conjuntos de datos en formatos CSV para recopilar los centros de ayuda humanitaria, sin embargo ambas opciones presentadas en la tabla tienen datos en forma de listado con el fin de extraer dicha información, El recopilado de datos de BBVA contiene la latitud y longitud de las ONG’s de albergues y comedores por todo méxico. ACNUR por otro lado tiene un conjunto más pequeño de datos y carece de especificaciones como latitud y longitud, por ello los datos de ONG’ s son recabados de BBVA que se ven en la figura ##.

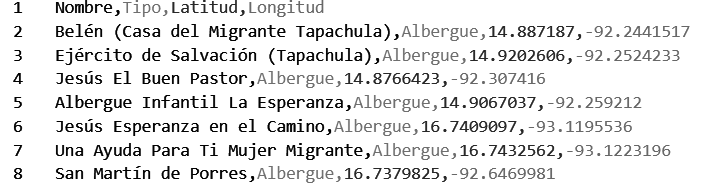


Figura ## CSV de los datos de BBVA (Creación propia)

### Procesamiento de datos

La base de datos utilizada proviene del portal de **Incidencia Delictiva** publicado por el **Secretariado Ejecutivo del Sistema Nacional de Seguridad Pública (SESNSP)**. Esta fuente proporciona información estructurada sobre delitos cometidos en el fuero común en todo el país, organizada por entidad, municipio, año, mes y tipo de delito, vistos en la siguiente tabla ##.

Tabla ## campos de los datos del SESNSP

| **Nº** | **Nombre del Campo** | **Tipo de Dato** |
| --- | --- | --- |
| 0 | Año | int64 |
| 1 | Clave\_Ent | int64 |
| 2 | Entidad | object |
| 3 | Cve. Municipio | int64 |
| 4 | Municipio | object |
| 5 | Bien jurídico afectado | object |
| 6 | Tipo de delito | object |
| 7 | Subtipo de delito | object |
| 8 | Modalidad | object |
| 9 | Enero | int64 |
| 10 | Febrero | int64 |
| 11 | Marzo | int64 |
| 12 | Abril | int64 |
| 13 | Mayo | float64 |
| 14 | Junio | float64 |
| 15 | Julio | float64 |
| 16 | Agosto | float64 |
| 17 | Septiembre | float64 |
| 18 | Octubre | float64 |
| 19 | Noviembre | float64 |
| 20 | Diciembre | float64 |

En total, la base de datos incluye esta cantidad de delitos agrupados por 7 categorías deBien jurídico afectado,40 tipos generales de delito, 55 subtipos y 59 modalidades diferentes como se visualiza en la figura ##.

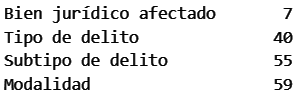


Figura ## Delitos en la tabla (Creación propia)

Como se puede observar en la figura ##, los datos incluyen una amplia variedad de delitos, muchos de los cuales no están directamente relacionados con riesgos frecuentes para las personas en tránsito o migrantes, como es el caso de los delitos ambientales o electorales.

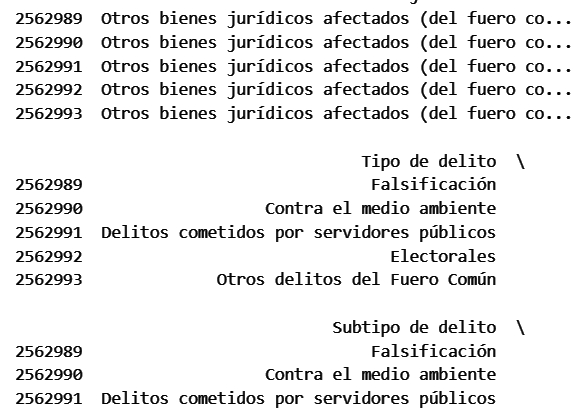


Figura ## Muestra de los datos (Creación propia)

Dado que el tema del prototipo se centra en las personas migrantes en tránsito, se realizó un proceso de depuración de los datos, eliminando aquellos tipos de delitos que no fueron previamente documentados como los principales delitos que enfrentan los migrantes.  
Se seleccionaron únicamente los registros correspondientes a los delitos de:

* + **Robo**, Donde robo se divide en muchos subtipos de delitos mostrados en la figura ##

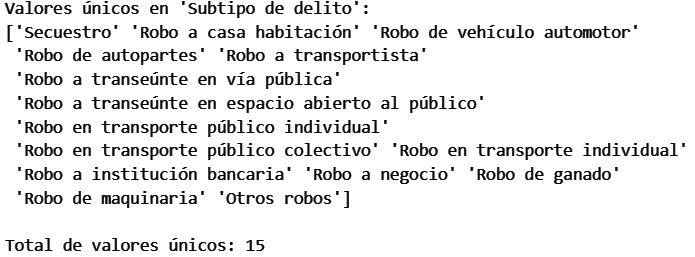


Figura ## Subtipo de delitos (Creación propia)

Donde eliminamos Robo a transportista, Robo a institución bancaria, Robo a negocio y Robo de ganado también solo conservamos sus siguientes modalidades en robo: Con violencia, Sin violencia, Robo de coche de 4 ruedas Con violencia, Robo de coche de 4 ruedas Sin violencia, Robo de motocicleta Con violencia, Robo de motocicleta Sin violencia.

* + **Secuestro,** Secuestro no tiene subtipos sin embargo si poseía modalidad como se muestra en la figura ## todas fueron usadas.

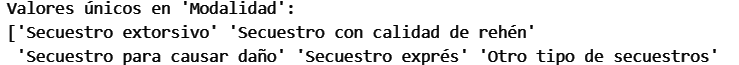


Figura ## Modalidades de secuestro (Creación propia)

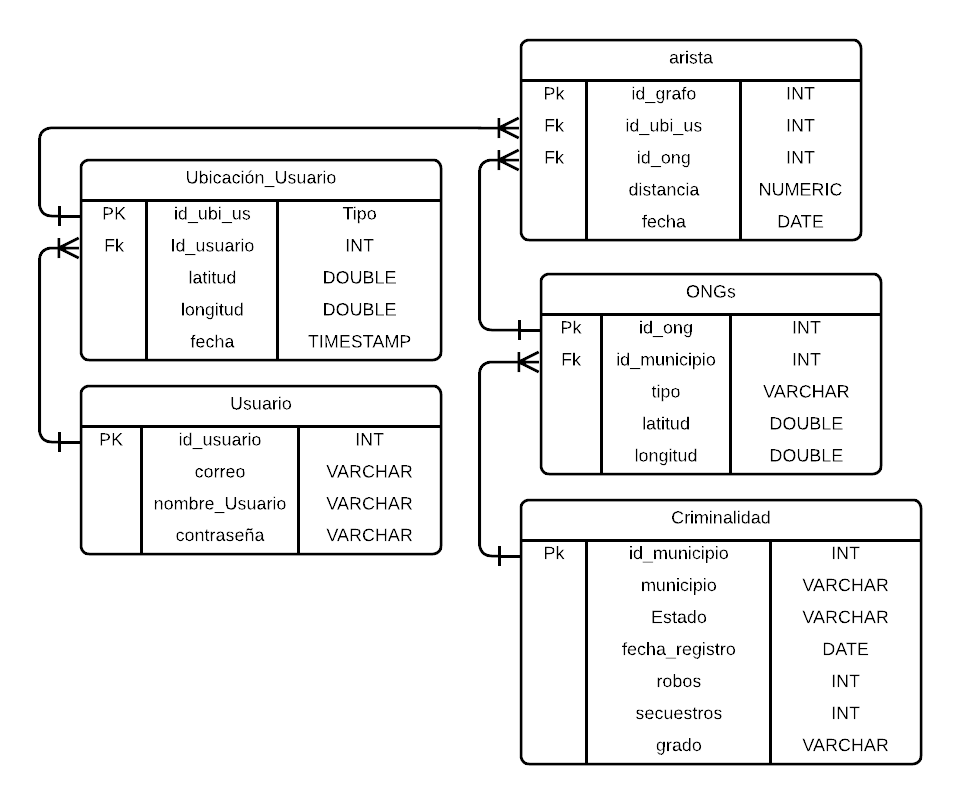
Los datos se filtraron para conservar únicamente los últimos tres meses disponibles al momento de la consulta, con el objetivo de contar con información actualizada y representativa del contexto reciente como se muestra en la tabla ##.

Tabla ## Datos procesados

| **Nombre de la Columna** | **Valores Únicos** |
| --- | --- |
| Año | 1 |
| Clave\_Ent | 32 |
| Entidad | 32 |
| Cve. Municipio | 2,486 |
| Municipio | 2,337 |
| Bien jurídico afectado | 2 |
| Tipo de delito | 2 |
| Subtipo de delito | 10 |
| Modalidad | 11 |
| Febrero | 122 |
| Marzo | 131 |
| Abril | 118 |

### 5.4 Modelo de datos

El modelado de datos se refiere a la organización y dirección que tendrán los datos en una Base de datos para un sistema de software para ello es normal utilizar un diagrama estructural de UML que modela las entidades refiriéndose a las tablas en un diagrama de entidad relación, de una base de datos está muestra sus atributos en columnas y las relaciones entre ellas con claves foráneas o asociaciones. Este diagrama está enfocado en el diseño de datos y se puede observar en la *figura 5.4.1.*



###### Figura 5.4.1 Diagrama de Entidad Relación de la BD

**Tipo de base de datos Relacional**

La estructura de datos permite definir relaciones entre las distintas entidades (por ejemplo, un usuario puede realizar varias valoraciones, y cada ONG puede recibir valoraciones de múltiples usuarios).  
Este modelo relacional facilita la realización de consultas complejas mediante *joins* y asegura la integridad referencial de los datos.

La base de datos se estructura en tablas claramente delimitadas (Usuarios, Criminalidad\_Estados, ONGs, Valoraciones), lo cual es propio de los sistemas relacionales.  
  
Cada tabla incluye un identificador único por ejemplo, *id\_usuario*, *id\_estado*, *id\_ong*, *id\_valoracion* y en el caso de la tabla *Valoraciones*, se establecen relaciones mediante los campos *id\_usuario* e *id\_ong*, que actúan como claves foráneas referenciando a las tablas *Usuarios* y *ONGs*, respectivamente como es observable en la*tabla 5.4.2.*

##### Tabla 5.4.2 Tablas consideradas para la Base de datos

| **Tabla** | **Campo** | **Tipo** | **Descripción** |
| --- | --- | --- | --- |
| **Usuario** | id\_usuario | INT | Identificador del usuario. |
| correo | VARCHAR | Correo electrónico del usuario. |
| nombre\_usuario | VARCHAR | Nombre del usuario. |
| contraseña | VARCHAR | Contraseña del usuario. |
| **Ubicación\_Usuario** | id\_ubi\_us | INT | ID de la ubicación del usuario. |
| latitud | DOUBLE | Latitud de la ubicación del usuario. |
| longitud | DOUBLE | Longitud de la ubicación del usuario. |
| fecha | TIMESTAMP | Fecha en que se registró la ubicación del usuario. |
| **arista** | id\_grafo | INT | ID del grafo o ruta. |
| distancia | NUMERIC | Distancia entre el usuario y la ONG. |
| fecha | DATE | Fecha de registro de la ruta. |
| **ONGs** | id\_ong | INT | Identificador de la ONG. |
| nombre\_ong | VARCHAR | Nombre de la ONG. |
| tipo | VARCHAR | Tipo de servicio de la ONG. |
| latitud | DOUBLE | La actitud de la ONG. |
| longitud | DOUBLE | Longitud de la ONG. |
| **Criminalidad** | id\_municipio | INT | Identificador del municipio. |
| municipio | VARCHAR | Nombre del municipio. |
| estado | VARCHAR | ID o nombre del estado al que pertenece el municipio. |
| fecha\_registro | DATE | Fecha del registro delictivo. |
| robos | INT | Número de robos registrados en el municipio. |
| secuestros | INT | Número de secuestros registrados. |
| grado | VARCHAR | Nivel de criminalidad (alto, medio, bajo). |

### 5.4.1 Tabla Comparativa de Bases de Datos

Para el desarrollo del prototipo de asistencia informativa a migrantes en tránsito en México, es fundamental seleccionar un sistema de gestión de bases de datos (SGBD) que sea capaz de manejar grandes volúmenes de información de forma eficiente. El sistema elegido debe ser relacional, ya que permitirá estructurar y vincular distintos tipos de datos por eso se hace una tabla comparativa de las posibles bases de datos *tabla 5.4.1.1.*

*Tabla 5.4.1.1 comparativa de Bases de datos* (Brown, 2025)

| **Característica** | **PostgreSQL** | **MySQL** | **MariaDB** |
| --- | --- | --- | --- |
| **Escalabilidad** | Alta escalabilidad *vertical* (Añadir más recursos a un solo servidor) y *horizontal* (Distribuir la carga entre múltiples servidores) con extensiones. | Alta (vertical, limitada en horizontal) | Alta (vertical y horizontal con Galera Cluster) |
| **Compatibilidad SQL** | American National Standards Institute Structured Query Language (ANSI SQL) es un conjunto de normas que definen cómo debe funcionar el lenguaje SQL para garantizar *portabilidad entre sistemas de bases de datos*. + extensiones avanzadas | ANSI SQL + dialecto propio | ANSI SQL + compatibilidad con MySQL |
| **Soporte para JSON** | Si, JavaScript Object Notation (JSON) es un formato ligero de intercambio de datos | Sí (JSON desde MySQL 5.7) | Sí |
| **Casos de uso ideales** | Aplicaciones complejas, análisis de datos. | Aplicaciones web. | Reemplazo de datos de MySQL para sistemas distribuidos. |
| **Plataformas soportadas** | Windows, Linux, macOS | Windows, Linux, macOS | Windows, Linux, macOS |

### 5.4.2 Ventajas y desventajas de las Bases de Datos

**PostgreSQL**

**Ventajas:**

1. Mapas avanzados con PostGIS: PostGIS es una extensión que convierte a PostgreSQL en “experto” para manejar datos geográficos (como coordenadas GPS, rutas o zonas). Te permite buscar “todos los puntos dentro de 1 km” o “la ruta más corta entre A y B” sin programar todo desde cero (IBM, 2021).
2. Datos flexibles: JSONB es un tipo de campo que guarda texto en formato JSON (como un objeto en JavaScript) pero lo indexa para que las búsquedas sean rápidas. Esto es útil si, por ejemplo, almacenan configuraciones de usuario o registros variables sin crear columnas nuevas cada vez (IBM, 2021).
3. Transacciones seguras (ACID): ACID conjunto de reglas que garantizan que los cambios en tu base de datos sean fiables:
   1. *Atomici­dad:* se aplican todos los cambios o ninguno.
   2. *Consistencia:* la base de datos pasa de un estado válido a otro válido.
   3. *Isolamiento:* varias operaciones al mismo tiempo no se “meten en medio” unas de otras.
   4. *Durabilidad:* una vez confirmados, los cambios no se pierden aunque falle el servidor.
4. Plugins y funciones extra: Puedes instalar módulos para añadir funcionalidades (por ejemplo, buscadores de texto, algoritmos especiales, nuevos tipos de datos) (IBM, 2021).

**Desventajas:**

1. Curva de aprendizaje más larga: Tiene muchas opciones y parámetros (tuning de rendimiento, mantenimiento de espacio en disco).
2. Mantenimiento periódico: Hay que “limpiar” de vez en cuando (con comandos como VACUUM) los espacios que dejan atrás las filas borradas o cambiadas, para que la base no crezca sin control.
3. Instalación y configuración inicial: Montar un servidor robusto en producción (copias de seguridad, réplicas, monitoreo) es más complejo que con otras bases de datos más “listas para usar” (IBM, 2021).

**MySQL**

**Ventajas:**

1. Muy fácil de empezar: La mayoría de los tutoriales y servicios de hosting ya lo traen preinstalado. Con pocos pasos tienes un servidor listo para meter datos.
2. Buen rendimiento en lecturas: Funciona muy bien cuando tu app “solo” consulta datos (SELECT) sin hacer operaciones complejas de escritura o transacciones muy largas (Erickson, 2024).
3. Ecossistema LAMP: En el mundo Linux + Apache + MySQL + PHP tienes muchísimas guías, foros y herramientas (por ejemplo, cPanel) que simplifican tareas cotidianas (Erickson, 2024).
4. ACID básico con InnoDB: El motor de almacenamiento InnoDB soporta transacciones seguras y bloqueos de fila, suficientes para muchas aplicaciones. (Erickson, 2024)

**Desventajas:**

1. Funciones geográficas limitadas: Tiene soporte GIS (para geopuntos y líneas) pero no tan completo como PostGIS: carece de algunas consultas complejas o índices especializados (Erickson, 2024).
2. Se degrada con mucho volumen: Cuando tu base crece a decenas o cientos de gigas y hay miles de conexiones simultáneas, puede volverse lento si no se ajusta bien.
3. Menos flexibilidad de datos: No hay un tipo JSON tan potente como JSONB las búsquedas dentro de un campo JSON pueden ser más lentas.
4. Menos extensible: Si quieres añadir funcionalidades nuevas (por ejemplo, un algoritmo de encriptado distinto o un índice especializado), es más difícil que en PostgreSQL (Erickson, 2024).

**MariaDB**

**Ventajas:**

1. Clon de MySQL con extras: MaríaDB nació como “otro MySQL”. Mantiene la misma forma de usarlo, pero añade mejoras internas en velocidad y replicación (mariadb., 2025).
2. Replicación multi-master: A diferencia de MySQL clásico, puedes tener varios nodos que aceptan escritura al mismo tiempo y se sincronizan entre sí, facilitando alta disponibilidad (mariadb., 2025).
3. Motores especializados: Además de InnoDB, tiene plugins como ColumnStore (optimizado para análisis de grandes volúmenes de datos) o Aria (más rápido en ciertas lecturas) (mariadb., 2025).
4. Hilos (threads) mejor manejados: Su “thread pool” nativo ayuda a que miles de conexiones no colapsen el servidor (mariadb., 2025).

**Desventajas**

1. Comunidad más pequeña: Aunque crece, tiene menos documentación, ejemplos y plugins de terceros que MySQL o PostgreSQL.
2. Divergencias con MySQL: Algunas nuevas características de MySQL (o viceversa) no están en MaríaDB, lo que puede complicar migrar entre uno y otro en el futuro.
3. Riesgo de “inflar” la base: Dependiendo de la carga, puede acumular más datos de índice o logs que MySQL tradicional si no se configura bien (mariadb., 2025).

Para elegir una base de datos para el prototipo de la app móvil se requiere considerar parámetros como geolocalización y cálculo de rutas en base a diferentes pesos, debes balancear tres cosas principales:

1. **Facilidad de uso**: qué tan simple es instalarla, configurarla y meter datos.
2. **Manejo de mapas y ubicaciones**: capacidad para guardar y consultar datos de puntos.
3. **Escala y rendimiento**: qué tan bien aguanta cuando los datos crecen o hay muchas conexiones al mismo tiempo.

Para ello la mejor opción propuesta es PostgreSQL ya que cumple los puntos necesarios para el desarrollo del prototipo.

**Extensiones esenciales el prototipo en PostgreSQL:**

1. PostGIS
   1. Para trabajar con datos geoespaciales.
   2. Permite almacenar y consultar datos espaciales (coordenadas, distancias, rutas).
   3. Requisitos usa latitud y longitud.
   4. Ideal para encontrar la ONG más cercana a una ubicación.
2. pgRouting
   1. Para aplicar algoritmos de rutas sobre redes de grafos.
   2. Se integra con PostGIS.
   3. Soporta A\*, Dijkstra, etc., sobre tus tablas de nodos y aristas.
   4. Usa la tabla de arista como red de caminos.

## Comparativa de lenguajes de programación

Al desarrollar el prototipo se requiere considerar parámetros entre velocidad y rendimiento por lo que el lenguaje a elegir se reduce entre dos opciones Java y Python.

Java, con su robustez y compatibilidad nativa en Android, ofrece alto desempeño pero requiere más código y tiempo de desarrollo.

Python, en cambio, permite prototipado rápido gracias a su sintaxis sencilla y frameworks, aunque sacrifica optimización. Esta comparación analiza ventajas y desventajas para priorizar la productividad inicial (Python) o escalabilidad y rendimiento (Java). (Amazon, 2025)

Tabla ## Descripciones de Lenguajes

| **Característica** | **Java** | **Python** |
| --- | --- | --- |
| **Paradigma** | Orientado a objetos (OOP), multiplataforma. | Multiparadigma (OOP, procedural, funcional). |
| **Tipado** | Estático y fuerte. | Dinámico y fuerte. |
| **Sintaxis** | Verbosa.  *Ejemplo:* System.out.println("Hola"); | Simple y legible.  *Ejemplo:* print("Hola") |
| **Rendimiento** | Alto (JIT compiler, JVM). | Lento (interpretado, tipado dinámico). |
| **Gestión de memoria** | Automática (Garbage Collector). | Automática (Garbage Collector). |
| **Uso principal** | Aplicaciones empresariales, Android, servidores (Spring, Hibernate). | IA, ciencia de datos (NumPy, TensorFlow), scripting, web (Django, Flask). |
| **Portabilidad** | Alta ("Write once, run anywhere", JVM). | Multiplataforma (interpretado). |
| **Librerías** | Amplias (Java Standard Library, Spring Boot). | Enormes (Pandas, Django, Scikit-learn). |
| **Curva de aprendizaje** | Moderada (OOP estricto, sintaxis rígida). | Baja (sintaxis intuitiva). |
| **Comunidad** | Grande (empresas, open source). | Enorme (academia, startups, IA). |
| **Seguridad** | Alta (JVM con sandbox, bytecode verificado). | Depende del uso (menos vulnerable que C++). |
| **Salario típico** | Alto en empresas tradicionales. | Alto en IA/startups. |
| **Tendencia (2025)** | Estable (empresas, Android). | En crecimiento (IA, datos). |

Tabla ## Ventajas y desventajas de los lenguajes

| **Lenguaje** | **Python** | **Java** |
| --- | --- | --- |
| **Ventajas** | -Sintaxis simple y legible | -Alto rendimiento (JVM optimizada) |
| -Ideal para IA y ciencia de datos | -Tipado estático (menos errores en runtime) |
| -Gran cantidad de librerías | -Ideal para sistemas escalables |
| -Perfecto para prototipo rápido | -Gran soporte para aplicaciones empresariales |
| **Desventajas** | Bajo rendimiento (interpretado) | - Sintaxis verbosa (más código) |
| -Tipado dinámico (errores en runtime) | - Mayor consumo de memoria (JVM) |
| Problemas de concurrencia (GIL) | -Curva de aprendizaje más pronunciada |
| -No es óptimo para sistemas críticos | -Menos productivo en desarrollo rápido |

## Comparativa de algoritmos de rutas

Se compararon dos algoritmos: Dijkstra, A\*, considerando aspectos como su aplicabilidad en contextos de riesgo, capacidad de adaptación a datos dinámicos y utilidad en entornos geoespaciales. Para el desarrollo del prototipo de la aplicación móvil orientada a la generación de rutas para personas en tránsito, se propone la implementación de algoritmos de búsqueda como A\*, Dijkstra y técnicas de agrupamiento como K-Means. Estas herramientas permiten modelar trayectos a partir de información geográfica y de riesgo.

El algoritmo A\* ha sido seleccionado como el principal motor de búsqueda de rutas debido a su eficiencia en la planificación de caminos óptimos, gracias al uso de heurísticas que guían la exploración del grafo hacia el destino. A diferencia de otras técnicas, A\* combina el costo acumulado desde el origen con una estimación del costo restante, lo que reduce significativamente el número de nodos explorados, siendo ideal para dispositivos móviles con capacidades de procesamiento y memoria limitadas. (Sinnot, 2019, 159)

En la tabla ## se muestra la comparativa de heurísticas para determinar la que se va a utilizar en el prototipo de aplicación móvil

Tabla ## Comparativa de Heurísticas

| **Heurística** | **Uso de coordenadas** | **Tipo de espacio ideal** | **Comentario contextual** |
| --- | --- | --- | --- |
| Haversine | ✅ | Geográfico real (lat/lon) | Ideal para distancias entre municipios usando coordenadas |
| Euclidiana | ❌ | Plano cartesiano | Solo válida si usas mapas |
| Manhattan | ❌ | Rejillas tipo ciudad/calles | Solo útil si es un modelo cuadriculado |
| Chebyshev | ❌ | Rejillas con diagonales | Similar a Manhattan, pero con movimiento diagonal |

Dado que la aplicación trabaja con coordenadas geográficas (latitud y longitud) y busca rutas entre municipios en un entorno no estructurado como lo es una cuadrícula, la heurística de Haversine fue seleccionada por su capacidad para estimar distancias sobre la superficie terrestre, utilizando la fórmula del gran círculo. (Sinnot, 2019, 160)

La fórmula de Haversine es una ecuación que se utiliza para calcular la distancia más corta (conocida como distancia del gran círculo) entre dos puntos sobre la superficie de una esfera, dados sus valores de latitud y longitud. Importante en [navegación](https://en.wikipedia.org/wiki/Navigation) , es un caso especial de una fórmula más general en [trigonometría esférica](https://en.wikipedia.org/wiki/Spherical_trigonometry) , la ley de los senos de Havers , que relaciona los lados y ángulos de los triángulos esféricos. (Veness, 2002)

Fórmula:

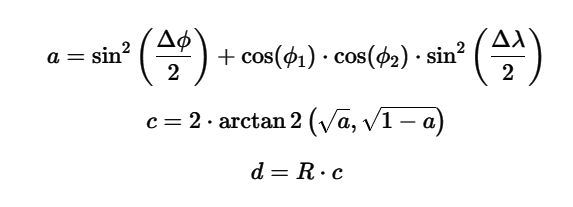
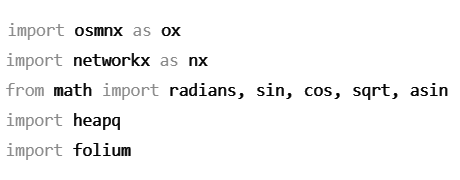


Figura ## Fórmula de Heurística Haversine

* Δϕ=ϕ2​−ϕ1​: diferencia de latitudes (en radianes)
* Δλ=λ2−λ1\Delta \lambda = \lambda\_2 - \lambda\_1Δλ=λ2​−λ1​: diferencia de longitudes (en radianes)
* R: radio de la Tierra (en km ≈ 6371)
* d: distancia entre los dos puntos en kilómetros

Se utiliza el algoritmo de Dijkstra de forma paralela para una nueva ruta, que no requiere una heurística, no se necesita garantizar la búsqueda exhaustiva del camino más corto para aumentar las opciones de los usuarios. (Richaud, 2024)

Se comprobó la funcionalidad del algoritmo de ruta dentro de la ciudad de Guadalajara, utilizando datos de OpenStreetMap. Para ello, se emplea el algoritmo A\*, conocido por ser eficiente en la búsqueda de rutas óptimas cuando se dispone de una función heurística.En este caso, se utiliza como heurística la fórmula de Haversine, que calcula la distancia en línea recta (ortodrómica) entre dos puntos sobre la superficie curva de la Tierra, a partir de sus coordenadas de latitud y longitud. Esta fórmula permite estimar de manera realista la cercanía entre nodos, lo que acelera la búsqueda del camino más corto.



*figura## Bibliotecas importadas, creación propia*

En la figura## se muestran las bibliotecas que se utilizaron.

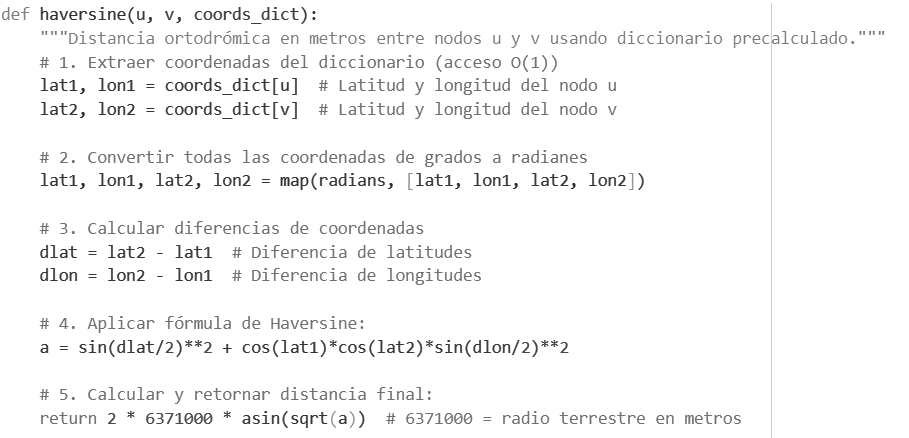
donde:

* osmnx: descarga y maneja redes viales desde OpenStreetMap.
* networkx: trabaja con grafos.
* math: para la fórmula de Haversine.
* heapq: estructura de datos eficiente para el algoritmo A\*.
* folium: crea mapas interactivos.



*figura## Descarga de las carreteras de guadalajara, creación propia*

Se descarga la red de caminos peatonales de Guadalajara y se representa como un grafo (G), donde los nodos son intersecciones y las aristas son calles como se muestra en la figura##.



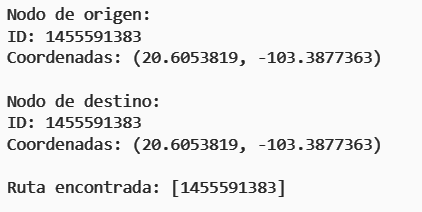
*figura## Cálculo de Heurística Haversine, creación propia.*

Esta función calcula la distancia entre dos nodos (u y v) a partir de sus coordenadas (usadas como heurística para A\*), considerando la curvatura de la Tierra como se muestra en la figura##.



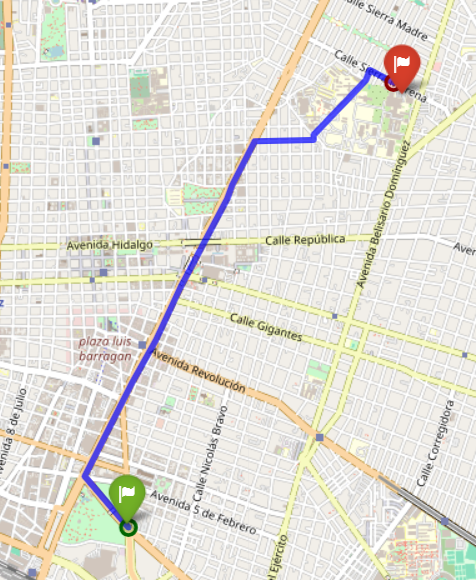
*figura## Creación de diccionario con coordenadas, creación propia.*

Se crea un diccionario que guarda las coordenadas de cada nodo (latitud, longitud) como se muestra en la figura##.



*Figura## Obtención de la ruta calculada con A\*, Creación propia*

Se ejecuta el algoritmo A\* para obtener el camino más corto entre el nodo de origen y el de destino, usando:Longitud física (weight='length') y la Heurística haversine como se maestro en la figura ##.



*Figura## Ruta en OSM, creación propia*

Se muestra la ruta sobre un mapa de la red vial usando ox.plot\_graph\_route como se muestra en la figura##.

### Algoritmos de Agrupamiento

Se comparó K-Means con otros algoritmos de agrupamiento como se muestra en ta tabla## para seleccionar una técnica de agrupamiento no supervisado, con el fin de clasificar municipios según su nivel de riesgo (alto, medio o bajo), utilizando registros de delitos como robos y secuestros ocurridos en los últimos tres meses como ya se hizo en el procesamiento de los datos. Esta clasificación alimenta sirve para la visualización de zonas críticas dentro del mapa.

Tabla ## Comparativa de algoritmos de agrupación.

| **Criterio** | **Descripción** | **K-Means** | **DBSCAN** | **Jerárquico** |
| --- | --- | --- | --- | --- |
| Escalabilidad | Capacidad del algoritmo para manejar y procesar grandes volúmenes de datos. | ✅ | ❌ | ❌ |
| Velocidad | Tiempo de ejecución en datasets grandes. | ✅ | ❌ | ❌ |
| Resultados consistentes | Reproduce los mismos resultados si se usa una buena inicialización. | ✅ | ❌ | ❌ |
| Ideal para uso exploratorio rápido | Bueno para entender patrones generales de los datos de forma ágil. | ✅ | ❌ | ❌ |

K-Means se selecciona por su efectividad en la segmentación no supervisada de datos, permitiendo clasificar municipios con niveles de riesgo (alto, medio o bajo) en función de variables como robos y secuestros reportados en los últimos tres meses gracias al uso de los centroides. Este agrupamiento facilita la generación de mapas que muestran los municipios clasificados con bajo costo computacional para el prototipo de la aplicación móvil como se muestra en la figura ##.

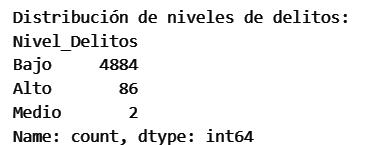


Figura ## Distribución de niveles con K means (Creación propia)

Gracias a la agrupación de los datos creamos una nueva columna que indique el nivel en las tablas de los datos de criminalidad como se muestra en la figura ##.



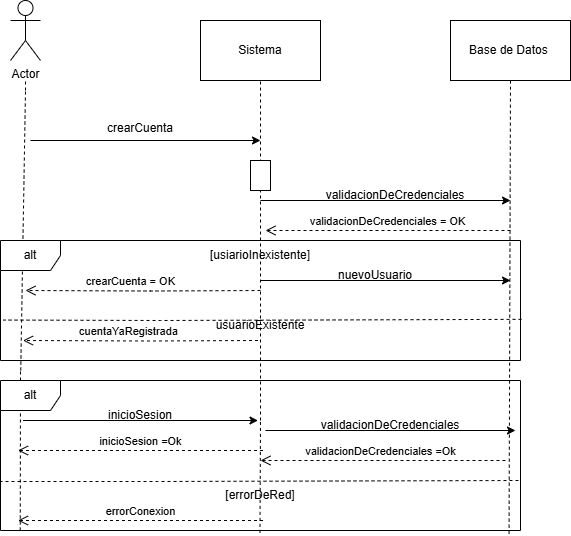
Figura ## Datos con niveles de delitos (Creación propia)

### 5.3 Desarrollo de la arquitectura del Prototipo de App Móvil

En esta sección se define la estructura técnica sobre la cual se construye el prototipo de aplicación móvil, describiendo cómo interactúan los componentes, cómo fluye la información desde la interfaz de usuario hasta la base de datos y los servicios de inteligencia artificial.

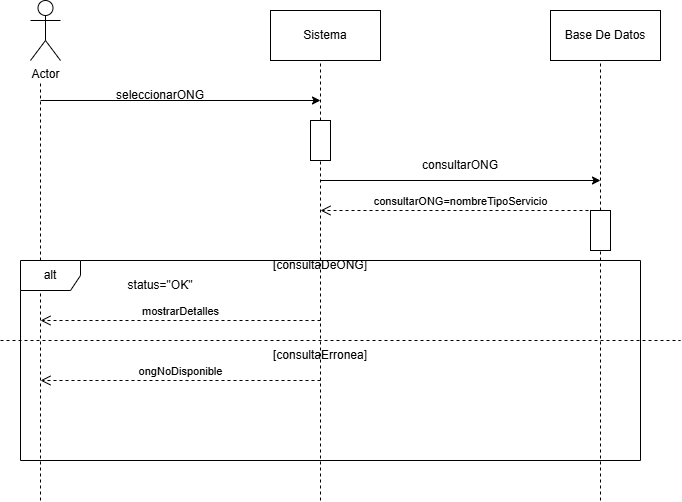
### .5.3.3 Diagrama de Secuencia de Prototipo de Aplicación Móvil

Las *Figuras 5.3.3.1 a 5.3.3.6* muestran los diagramas de secuencia correspondientes al prototipo de la aplicación móvil. Cada uno de estos diagramas representa, de manera estructurada y cronológica, las interacciones entre los distintos componentes funcionales del prototipo, permitiendo visualizar el flujo de mensajes y la lógica de ejecución en diversos escenarios de uso.



*Figura 5.3.3.1 Diagramas de Secuencia de Proceso de creación de cuenta y validación de credenciales.*

1. **Crear cuenta:** Verifica si el usuario ya existe en la base de datos.
   1. Si no existe: Incluir un nuevo usuario y muestra “Cuenta creada”.
   2. Si existe: Muestra “Nombre ya registrado” y sugiere iniciar sesión.
2. **Validar contraseña:** efectúa una consulta (SELECT) para verificación
   1. Resultados posibles
      1. Credenciales válidas → Redirige a la pantalla principal.
      2. Error de red → Muestra “Error de conexión”



*Figura 5.3.3.3 Diagramas de Secuencia de Interacción con marcadores de ONGs en el mapa.*

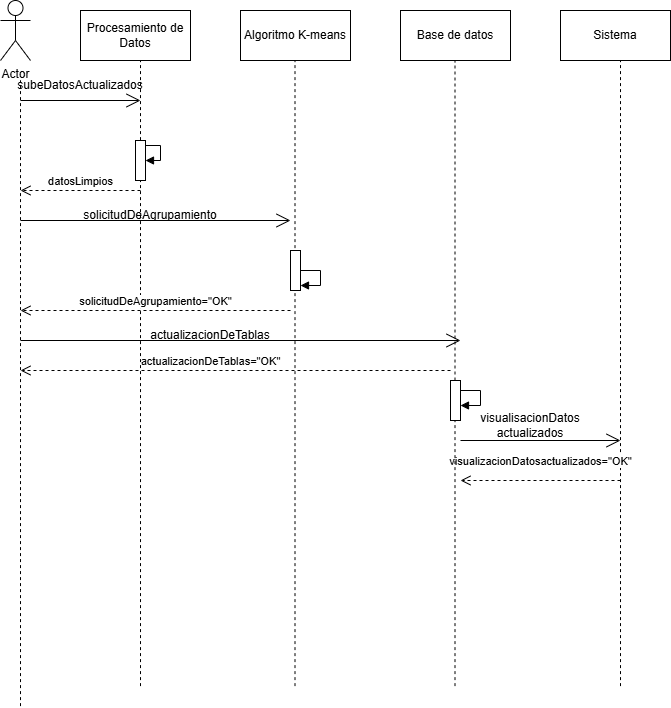
1. **Proceso:**
   1. selección de un marcador de ONG → Muestra ventana como detalles (nombres, tipo, ubicación).
   2. Consulta SQL: SELECT \* FROM ONGs WHERE id=X.
   3. Estados posibles:
      1. ONG activa: Muestra información completa.
      2. ONG inactiva: Muestra “ONG no disponible”

###### 

*Figura 5.3.3.3 Diagramas de Secuencia de Solicitud de rutas y actualización de datos.*

**Funciones**:

1. **Usuario**: Solicita ruta hacia una ONG, confirma ubicación y envía datos.
2. **MLAlgorithm**: Calcula la ruta (origen-destino) usando la base de datos y la muestra en el mapa.
3. **Sistema**: Obtiene datos actualizados de criminalidad para alimentar el algoritmo.



###### Figura 5.4.3.3 Diagramas de Secuencia de Panel de administrador y actualización

**Funcionalidades**:

1. **Administrador**:
   1. Inicia sesión con rol específico.
   2. Sube archivos CSV con datos nuevos (criminalidad, ONGs).
   3. Ejecuta scripts para reentrenar el modelo de ML con nuevos ratings.
2. **Resultado**:
   1. Base de datos y modelo actualizados → Mensaje: "Datos y modelo actualizados".

5.3.4 Metodología

En el desarrollo de un prototipo de aplicación móvil con un plazo estimado de desarrollo en cuatro a cinco meses, se ha determinado que la metodología Scrumban representa la opción más adecuada para la gestión del proyecto. Esta metodología híbrida combina las fortalezas del marco estructurado de Scrum con la flexibilidad visual y adaptativa de Kanban, permitiendo un equilibrio entre organización y capacidad de respuesta ante cambios.

Dicha elección se fundamenta en una comparación estructurada entre diversos enfoques de gestión de proyectos utilizados en el desarrollo de software. *La Tabla 5.3.4.1* presenta una tabla comparativa de metodologías, donde se valoran criterios clave como la adaptabilidad, planificación, duración del proyecto, enfoque en la entrega y complejidad de implementación. En este análisis, Scrumban destaca por su adaptabilidad y su capacidad de gestionar el flujo de trabajo de forma visual y eficiente, sin perder la estructura necesaria para mantener el orden del proyecto.

###### Tabla 5.3.4.1 Tabla comparativa de metodologias

| **Metodología** | **Tipo de enfoque** | **Ventajas** | **Desventajas** |
| --- | --- | --- | --- |
| Cascada (Winston W., 1970). | Lineal y secuencial | - Estructura clara  - Facilita la documentación  - Fácil gestión si no hay cambios | - Poco flexible  - No permite retrocesos  - No apta para requisitos cambiantes |
| Agile (Beck et al., 2001). | Iterativo e incremental | - Adaptable a cambios  - Mejora continua  - Alta implicación del cliente | - Requiere disciplina  - Menos documentación formal  - Puede ser difícil de escalar |
| Scrum (Sutherland & Schwaber, 2020). | Ágil con estructura definida | - Entregas regulares  - Facilita la gestión  - Mejora la comunicación interna | - Requiere experiencia previa  - Riesgo de mal uso de roles  - No ideal para proyectos urgentes |
| Kanban (Schön et al., 2010). | Ágil y visual | - Visualiza todo el proceso  - Flexible y adaptable  - Permite cambios en cualquier momento | - Difícil de priorizar  - No define tiempos  - Puede carecer de estructura si no se gestiona bien |
| Extreme Programming (XP) (Beck, 1999). | Ágil y técnico | - Alta calidad de código  - Reducción de errores  - Ideal para cambios frecuentes | - Requiere fuerte compromiso del equipo  - Demanda habilidades técnicas elevadas |
| Scrumban (Ladas, 2009). | Ágil híbrido (Scrum + Kanban) | - Adaptable  - Buena gestión visual  - Menor rigidez que Scrum | - Puede carecer de claridad si no se implementa bien  - Requiere madurez del equipo |

Dado que los proyectos de desarrollo de prototipos de aplicaciones suelen implicar modificaciones en los requerimientos, ya sea por retroalimentación de usuarios, condiciones técnicas o decisiones estratégicas, es indispensable contar con un enfoque ágil que no solo tolera, sino que favorezca la adaptación constante. En este sentido, Scrumban permite mantener una visión clara del progreso sin sacrificar la flexibilidad operativa, lo cual resulta fundamental en un proyecto con tiempos definidos y entregables progresivos.

# 5.3.5 Tabla Comparativa de Diseño de app

En el desarrollo del prototipo de migrantes en tránsito en México es importante diseñar una app que tenga rendimiento, para ello se muestra la *tabla 5.4.3.1* de comparaciones de herramientas para desarrollar una app que con la opción más viable.

*Tabla 5.3.4.1 comparativa de apps para el diseño UI/UX*

| **Herramienta** | Android Studio (Android, 2025. ) | Xcode (Apple Developer, 2025) | Flutter(Flutter, 2025) | Ionic + Capacitor (GetApp, 2025) |
| --- | --- | --- | --- | --- |
| **Costo** | Gratis | Gratis (requiere Mac) | Gratis | Gratis (Pro opcional) |
| **Lenguaje principal** | Java / Kotlin | Swift / Objective-C | Dart | HTML / CSS / JS |
| **Plataformas** | Solo Android | Solo iOS | Android, iOS, Web, Desktop | Android, iOS |
| **Rendimiento** | Código 100% nativo, compilado directamente al sistema Android. | Máxima eficiencia, uso directo de APIs nativas de Apple. | Compila código nativo con su propio motor gráfico. Ligeramente inferior a nativo puro. | Corre en WebView (como una mini app web dentro del teléfono), por eso es más limitado. |
| **Escalabilidad** | Soporta arquitectura modular y CI/CD, ideal para proyectos grandes. | Permite testing, automatización y escalamiento, pero limitado a iOS. | Soporta arquitecturas como BLoC, paquetes propios y externos; ideal para crecer. | Puede crecer, pero no es ideal para apps complejas en animaciones o datos en tiempo real. |

En cuanto al costo de publicación de la aplicación, se decidió lanzar inicialmente en la plataforma Android debido a su mayor alcance entre usuarios en tránsito y a su bajo costo de entrada. Publicar en la Google Play Store requiere un pago único de $25 USD, lo cual resulta accesible dentro del presupuesto personal del desarrollador. Por el momento, no se considera el lanzamiento en Apple App Store, ya que esta exige una suscripción anual de $99 USD, lo que representa un gasto más elevado y constante que no es viable en esta fase del proyecto (ABAMOBILE, 2024).

### 5.3.6 Conexión para prototipo

Para este prototipo, se eligió Kotlin como lenguaje principal en la capa de interconexión, ya que permite una implementación más clara y eficiente de tareas como llamadas HTTP, manejo de JSON y procesos asincrónicos, especialmente al integrarse con la librería Retrofit.

| **Característica** | Kotlin (*Kotlin and Android*, 2023.; *Connect to the Network | Connectivity*, 2023) | Java (*Connect to the Network | Connectivity*, 2023,; Android, 2025.) |
| --- | --- | --- |
| **Compatibilidad** | Total compatibilidad con librerías y código Java | Total compatibilidad con Android Studio |
| **Concisión** | Código más corto y limpio | Código más largo y repetitivo |
| **Manejo de errores null** | Sistema seguro de tipos (Null Safety) integrado | Mayor riesgo de errores por NullPointerException |
| **Asincronía** | Corrutinas para tareas asincrónicas simples y claras | Requiere manejo de hilos (Threads) o AsyncTask |
| **Modernidad** | Lenguaje moderno, expresivo, recomendado por Google | Más antiguo, pero estable |
| **Facilidad para Retrofit** | Integración más sencilla con corrutinas y estructuras modernas | Requiere más código para tareas similares |
| **Curva de aprendizaje** | Leve curva inicial para quienes vienen de Java | Más conocida entre desarrolladores veteranos |
| **Desventajas** | - Curva de aprendizaje para quienes vienen de Java  - Menos documentación en algunos casos | - Código más verboso  - Mayor riesgo de errores de null  - Más trabajo para tareas asíncronas |

Se eligió Kotlin como lenguaje principal para la capa de interconexión con el backend, ya que permite construir el prototipo de aplicación más mantenible y moderna, particularmente en el manejo de respuestas provenientes de una API REST, como lo pueden ser predicciones generadas por un modelo de aprendizaje automático o datos obtenidos desde una base externa. Asimismo, la integración con librerías especializadas como Retrofit y Gson o Moshi resulta más fluida y natural en Kotlin, lo cual reduce la complejidad del código, mejora la legibilidad y contribuye a acelerar el proceso de desarrollo.

### 5.3.7 Apis para prototipo

Tras hacer un análisis de diversas opciones se ha optado por el uso de OpenStreetMap (OSM) como datos geográficos abiertos, colaborativa y de uso libre, que permite gran adaptabilidad sin costos asociados ni restricciones comerciales.

La siguiente tabla comparativa presenta un análisis técnico entre OSM, Google Maps SDK y Mapbox SDK

*Tabla 5.3.7.1 comparativa de apis para prototipo*

| **Nombre** | **Licencia / Costo** | **Ventajas** | **Desventajas** |
| --- | --- | --- | --- |
| OpenStreetMap (OSM) (OSM), 2022). | Libre / Gratuito | - Gratuito y libre de licencias.  - Personalizable.  - Funciona offline (con librerías).  - Actualizado por la comunidad.  - Compatible con Kotlin (vía osmdroid). | - No incluye navegación guiada por defecto.  - Cobertura variable en regiones poco documentadas.  - Requiere integración adicional para funciones avanzadas. |
| Google Maps SDK (Documentación De Google Maps Platform, 2024). | Gratuito con límites / Comercial | - Alta precisión y cobertura global.  - Navegación guiada y tráfico en tiempo real.  - Integración sencilla con Android.  - Actualizaciones frecuentes. | - Requiere clave API y está sujeto a límites de uso gratuitos.  - No permite personalización profunda.  - No funciona offline nativamente. |
| Mapbox SDK (Maps SDK | Android Docs | Mapbox, 2023). | Gratuito con límites / Comercial | - Mapas vectoriales personalizables.  - Navegación offline disponible.  - Compatible con Kotlin.  - Interfaz moderna y flexible. | - Tiene límites gratuitos y planes comerciales.  - La documentación puede ser menos accesible para principiantes.  - Algunas funciones avanzadas requieren pago. |

### 5.3.8 Mecanismo de actualización y visualización de api

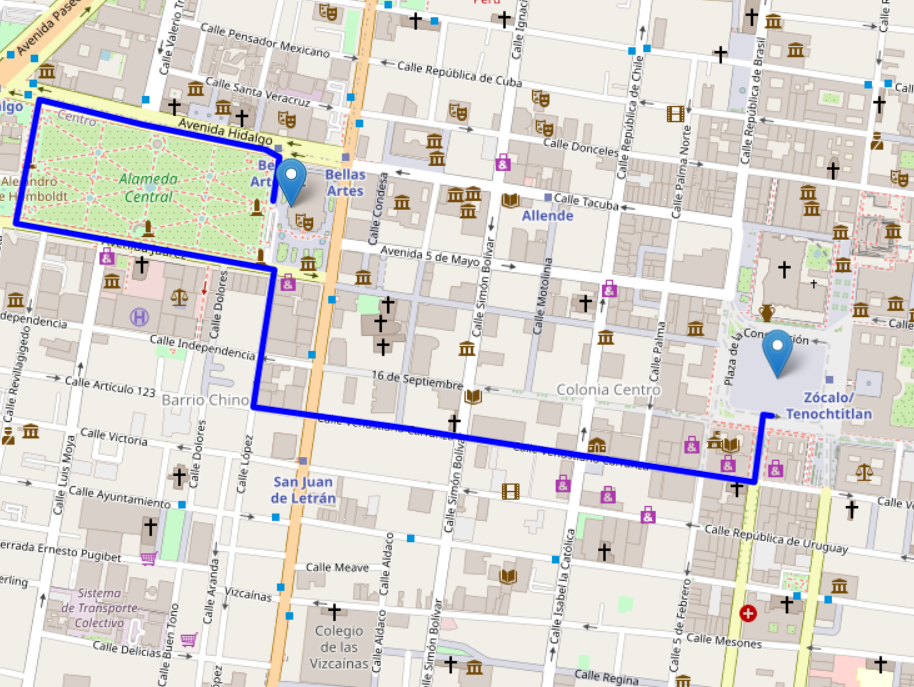
Mecanismo de actualización mensual de polígonos desde OSM:

Actualizar los polígonos geográficos provenientes de OpenStreetMap de forma automatizada una vez al mes, verificando si ha pasado suficiente tiempo desde la última sincronización al momento de iniciar el prototipo.

1. Registro de la última actualización
   1. Se guarda localmente, mediante SharedPreferences, la fecha de la última actualización exitosa.
2. Verificación del tiempo transcurrido
   1. Al iniciar la app, se comprueba si han pasado 30 días desde la última actualización.
3. Descarga de datos desde OSM
   1. Si se cumple el intervalo, se realiza una solicitud HTTP para obtener los polígonos actualizados desde OpenStreetMap.
4. Procesamiento y visualización
   1. Se extraen las coordenadas del archivo JSON/GeoJSON recibido y se dibujan los polígonos con la librería osmdroid.
5. Actualización de la referencia local
   1. Al finalizar el proceso, se registra la nueva fecha para futuras validaciones.

Visualización de la API

Se llevó a cabo una prueba de visualización utilizando la API de rutas para verificar su funcionamiento básico. En esta prueba, se simula una ruta en transporte en la Ciudad de México desde un punto A "Parque Central" hasta un punto B "Hospital General".

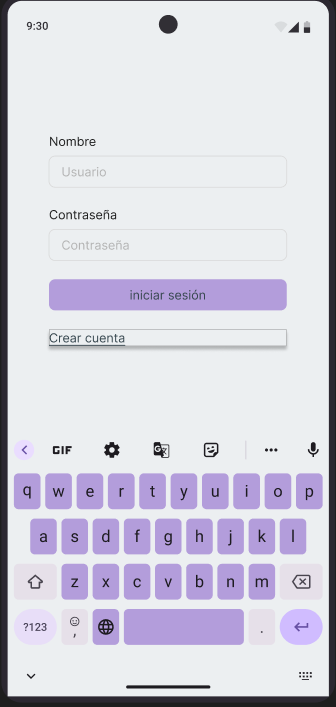


*Figura 5.3.7.2 Visualización de la api (Creacion propia)*

### 5.3.8 Mockups

El mockup es un prototipo (ya sea de una página web, diseño o producto) que muestra cómo funciona un objeto en el mundo real. Sobre todo, se emplea para mostrar una idea a los prospectos, clientes y compañeros de trabajo sin invertir mucho tiempo, dinero y esfuerzos de diseño (Walsh, 2023).

En la primera pantalla (*Figura 5.3.8.1*) se presenta el inicio de sesión a la prototipo.



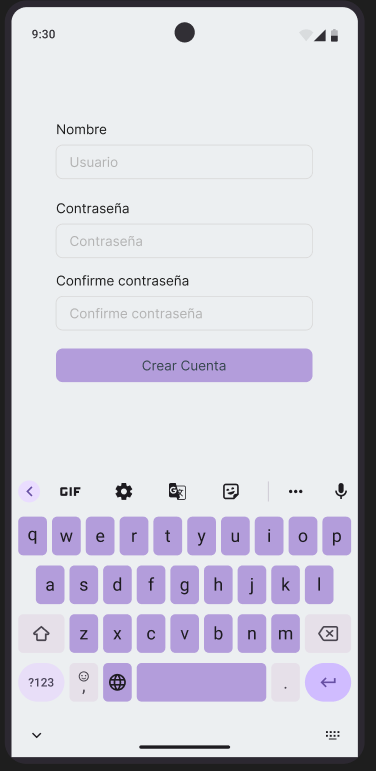
*Figura 5.3.8.1 Pantalla inicial.*

En la segunda pantalla (*Figura 5.3.8.2*) se mostrará mensaje de error al intentar iniciar sesión, si el usuaria no existe el sistema debe crear su cuenta

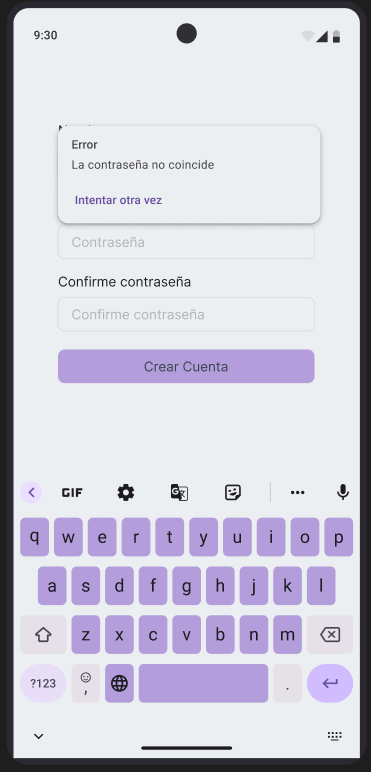
## 

*Figura 5.3.8.2 Pantalla de mensaje de error*.

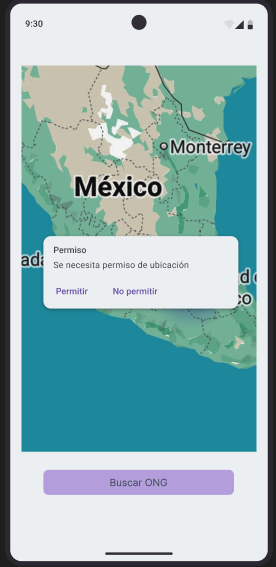
En la tercera pantalla (*Figura 5.3.8.3*) se muestra la pantalla de creación de cuenta



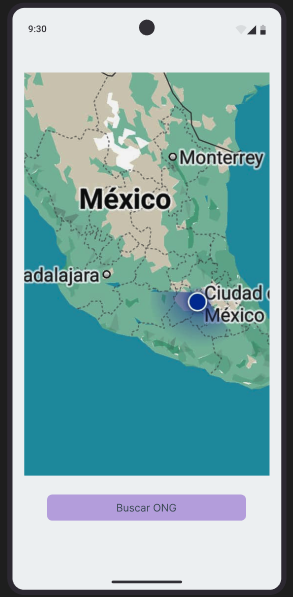
*Figura 5.3.8.3 Pantalla de Creación de cuenta.*

**

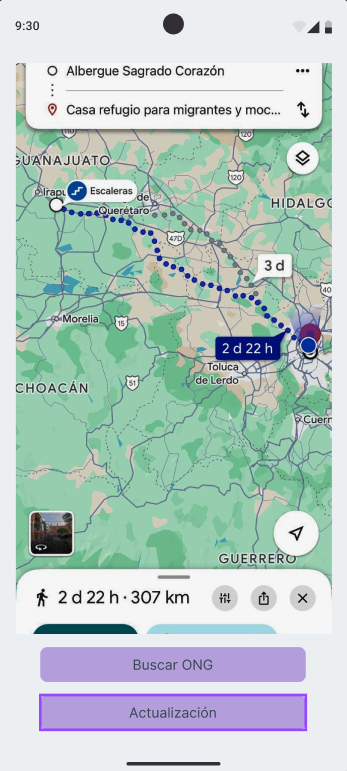
*Figura 5.3.8.4 Pantalla de contraseña no coincide.*



*Figura 5.3.8.5 Pantalla de Solicitud de ubicación.*



*Figura 5.3.8.6 Visualización del mapa.*

**

*Figura 5.3.8.7 Visualización de ruta*

# Capítulo 6. Desarrollo e Implementación

### 6.1 Análisis de factibilidad

El análisis de factibilidad se refiere a la evaluación de la viabilidad técnica, económica y operativa del proyecto del prototipo de aplicación móvil de asistencia a migrantes en tránsito por México. Se establecen la duración estimada, el tamaño del proyecto, los costos, la disponibilidad tecnológica y los aspectos de recopilación de información de los usuarios que estén dispuestos a usar el software, así como qué métricas se usarán para saber si está funcionando como se tiene previsto.

### 6.1.1 Factibilidad técnica

Para la factibilidad técnica se consideran tanto los recursos de hardware como las herramientas de software y las capacidades del equipo humano. El proyecto dispone de un equipo de escritorio cuyo costo de adquisición se refleja en la *tabla 6.1.1.1.*

| **Herramienta** | **CPU** | **RAM** | **Almacenamiento** |
| --- | --- | --- | --- |
| PostgreSQL (postgresql, 2025) | Mínimo: 1 núcleo  Recomendado: Multi-núcleo moderno | Mínimo: 1 GB  Recomendado: 2 GB o más | Mínimo: 10 GB  Depende del tamaño de la base de datos |
| Android Studio (developer, 2025) | Mínimo: Intel i3 o equivalente  Recomendado: Intel i5-8400 o superior | Mínimo: 8 GB  Recomendado: 16 GB | Mínimo: 8 GB  Recomendado: 30 GB (preferiblemente SSD) |
| Random Forest y A\* (Pham Dinh et al., 2023) | Multi-núcleo (idealmente 4 núcleos o más) | Mínimo: 8 GB  Recomendado: 16–32 GB | Depende del tamaño del dataset; SSD recomendado |

*Tabla 6.1.1.1 Software del equipo*

Estas máquinas cumplen con los requisitos mínimos para ejecutar entornos de emulación, gestores de base de datos, algoritmos de *Machine Learning* además de herramientas de control de versiones como Git.

El equipo de desarrollo, conformado por tres integrantes uno especializado en la programación móvil y diseño de interfaces, otro para la administración de bases de datos, y un último para desarrollar la lógica en *Machine learning* para la generación de la ruta. La infraestructura tecnológica disponible y el perfil técnico del personal aseguran que el proyecto pueda llevarse a cabo sin necesidad de adquirir nuevos equipos o licencias de software adicionales.

Parte de la factibilidad técnica permite determinar si el prototipo propuesto puede ser implementado con las herramientas tecnológicas disponibles y si estas cubren adecuadamente los requerimientos funcionales y de procesamiento del proyecto.

Los entornos de desarrollo para la bases de datos, modelos de inteligencia artificial y entorno deben cubrir una capacidad de respuesta adecuada ante los escenarios necesarios, para eso las herramientas han sido previamente evaluadas y las escogidas son resumidas en la *tabla 6.1.1.2* donde se desglosan las principales herramientas para desarrollar las funciones del sistema:

| **Herramienta / Tecnología** | **Función Principal** | **Justificación Técnica** |
| --- | --- | --- |
| **Android Studio** | Entorno de desarrollo para la aplicación móvil | Permite la creación de apps nativas en Android, ideal para dispositivos económicos y con amplia adopción. |
| **PostgreSQL** | Gestión de base de datos relacional | Sistema robusto, de código abierto, con soporte para estructuras complejas y consultas eficientes. |
| **K means** | Algoritmo de clustering no supervisado | Agrupa puntos de datos en k (clústeres) según similitud de características, permitiendo etiquetar los datos y generar mapas con puntos específicos. |
| **A\*** | Algoritmo de búsqueda heurística de rutas óptimas | Combina costo real (g) y heurística (h) para acelerar la búsqueda de caminos cortos en grafos. |
| **Dijkstra** | Algoritmo de grafo para hallar ruta de coste mínimo | Calcula la ruta más corta en grafos con pesos no negativos. |

*Tabla 6.1.1.2 Sotfware para el desarrollo*

Todas las herramientas seleccionadas son de código abierto o de libre uso, lo que beneficia la factibilidad económica del proyecto, ya que no implican costos de licencia asegurando el desarrollo del sistema sin incurrir en gastos adicionales por software, sosteniendo la factibilidad técnica y financiera del prototipo.

### 6.2 Factibilidad Operativa

La factibilidad operativa consiste en el análisis de los recursos productivos, incluidos los humanos, necesarios para la realización de un proyecto (Rus, 2020).

Para evaluar la factibilidad operativa, es esencial examinar los posibles roles para los flujos de trabajo, las ineficiencias de proceso y las necesidades de capacitación para ello se presenta la *tabla 6.2.1*

| **Rol** | **Justificación / Tareas clave** | **Habilidades esenciales** | **Sueldo** |
| --- | --- | --- | --- |
| Desarrollador Móvil | • Crear la interfaz de usuario (UI) clara y navegable.  • Implementar mapas interactivos, rutas y capas de ayuda humanitaria.  • Asegurar funcionamiento offline. | • React Native / Flutter  • Integración con APIs de mapas (Leaflet, Mapbox)  • Gestión de almacenamiento local (SQLite, Realm) | $12,500  (hireline, 2025) |
| Ingeniero de Machine Learning | • Diseñar y entrenar modelos que predigan “índice de riesgo” según datos históricos.  • Evaluar desempeño y ajustar parámetros.  • Automatizar pipelines de ML. | • Python, TensorFlow / PyTorch  • Preprocesamiento de datos (pandas, NumPy)  • Experiencia en estructurar pipelines (Airflow) | $7,499  (hireline, 2025) |
| Desarrollador de Base de Datos | • Modelar esquemas para criminalidad, ONG y rutas.  • Asegurar integridad y rapidez en consultas geoespaciales.  • Planificar actualizaciones mensuales. | • PostgreSQL/PostGIS o MongoDB  • Diseño normalizado y de índice | $12,500  (hireline, 2025) |

*Tabla 6.2.1 Personas para el desarrollo del prototipo*

Para garantizar la disponibilidad, escalabilidad y rendimiento de los servicios , es imprescindible contar con servidores que ofrezcan un equilibrio entre cómputo, memoria y almacenamiento. A continuación en la *tabla 6.2.2* se describen opciones comunes en entornos cloud, seguidas de una comparativa de sus características y costos.

*Tabla 6.2.2 Servicios de servidores.*

| **Proveedor** | **Plan Gratuito** | **Plan Básico (MXN/mes)** | **Almacenamiento inicial (Gratuito)** | **Características clave** |
| --- | --- | --- | --- | --- |
| **Railway.app** | 500 MB, 1,000 horas al mes de ejecución | Desde $96.59 | 500 MB | - PostgreSQL administrado  - Escalamiento automático  - Implementación rápida |
| **Render.com** | 256 MB RAM, 100 MB de almacenamiento | Desde $135.22 | 256 MB RAM / 100 MB disk | - PostgreSQL administrado  - Respaldos automáticos  - Panel de administración en línea |
| **Supabase** | 500 MB PostgreSQL, 1 GB archivos, 50,000 requests | Desde $482.93 | 500 MB | - PostgreSQL administrado  - Servicios de autenticación incluidos |
| **ElephantSQL** | 20 MB con PostgreSQL en plan "Tiny Turtle" | Desde $193.17 | 20 MB | - PostgreSQL administrado  - Panel web para gestión  - Soporte para conexiones seguras |

La decisión de escoger **Railway.app** para PostgreSQL es el almacenamientogratuito y costo accesible frente a otras alternativas del mercado.

En comparación con otros servicios como Render.com, Supabase o ElephantSQL, Railway.app ofrece 500 MB de almacenamiento gratuito y hasta 1,000 horas de ejecución al mes, lo cual es más que suficiente para el desarrollo, pruebas y primera etapa de producción del proyecto (railway, 2025).

Railway.app proporciona una instancia de PostgreSQL administrada, con capacidades de escalamiento automático, lo cual permite que el sistema se adapte a un mayor volumen de usuarios sin necesidad de intervención manual.

El plan básico de Railway.app comienza desde aproximadamente $96.59 MXN/mes, lo que lo posiciona como una de las opciones más económicas en el mercado (railway, 2025).

### 6.3 Factibilidad Económica

En esta sección se presenta un desglose de los costos fijos asociados al proyecto, que incluyen servicios básicos, infraestructura y otros gastos de equipo indispensables para el funcionamiento adecuado del desarrollo del prototipo. Estos costos se describen en las tablas , la cual permite identificar las partidas económicas que se deben contemplar mensualmente o bimestralmente para mantener operativa la infraestructura tecnológica y los espacios de trabajo.

Se detallan los costos de hardware, así como los gastos fijos necesarios para mantener el desarrollo operativo.

*Tabla 6.1.1.2 Hardware del equipo*

| **Artículo** | **Cantidad** | **Costo** |
| --- | --- | --- |
| **Computadora de escritorio** | 1 | $14,000 |
| **Laptop** | 1 | $25,000 |
| **Laptop** | 1 | $10,000 |

Estos gastos incluyen servicios básicos como renta de oficina o espacio de trabajo, consumo de energía eléctrica, agua potable, servicios de internet y telefonía, así como mantenimiento general

*Tabla 6.1.1.2 Costos Fijos*

| **Categoría** | **Descripción** | **Costo Estimado (MXN)** | **Frecuencia** |
| --- | --- | --- | --- |
| Agua potable | Servicio de agua mensual | $400 | Bimestral |
| Energía eléctrica | CFE, según consumo | $800 | Bimestral |
| Internet | Paquete de 100–200 Mbps | $600 | Mensual |
| Teléfono celular | Plan individual o familiar | $500 | Mensual |
| Renta o hipoteca | Vivienda u oficina | $4000 | Mensual |
| Mantenimiento | Reparaciones/limpieza | $100 – $300 | Mensual |

### 6.4 Riesgos

La evaluación de riesgos se enfoca principalmente en **i**dentificar, analizar y priorizar eventos que podrían amenazar el cumplimiento de los objetivos de alcance, tiempo, costo y calidad.

A continuación se presenta la *tabla 6.2.1* de riesgos para el prototipo, dividido en riesgos técnicos y operativos y del equipo de trabajo. Cada riesgo incluye su categoría, probabilidad, impacto, plan de gestión y nivel de riesgo.

**Riesgos Técnicos y Operativos**

| **Categoría** | **Riesgo** | **Probabilidad** | **Impacto** | **Gestión del riesgo** | **Nivel de riesgo** |
| --- | --- | --- | --- | --- | --- |
| **Datos** | Falta de precisión en los datos de criminalidad | Alta | Alta | Validar datos con múltiples fuentes confiables; usar datos históricos y actuales | Alto |
| **Base de datos** | Problemas en la actualización de la base de datos | Media | Alta | Implementar cron jobs y monitoreo automático; respaldo frecuente | Alto |
| **Conectividad** | Dificultad de acceso a la app por conectividad limitada | Alta | Media | Incluir modo offline y sincronización cuando haya conexión | Moderado |
| **Algoritmos** | Errores en los algoritmos de Machine Learning | Media | Alta | Probar modelos con datos de prueba; validación cruzada y revisión manual | Alto |
| **Fuentes externas** | Fiabilidad de las fuentes de datos (ONG y gobierno) | Alta | Alta | Seleccionar fuentes reconocidas; documentar la procedencia de los datos | Alto |
| **Usuarios** | Falta de conocimiento de los migrantes sobre la app | Alta | Media | Campañas informativas en albergues y redes sociales | Moderado |
| **Ambiente** | Condiciones climáticas extremas | Baja | Media | Asegurar que la app funcione sin conexión prolongada y con interfaz simple | Bajo |
| **Seguridad** | Problemas de seguridad informática o vulnerabilidades | Media | Alta | Usar cifrado, control de acceso, actualizaciones de seguridad y auditorías periódicas | Alto |
| **Costos** | Costos de mantenimiento y actualización elevados | Media | Media | Planificar presupuesto a largo plazo y buscar apoyos institucionales | Moderado |

*Tabla 6.3.1 Riesgos Técnicos y Operativos*

Los riesgos del equipo de trabajo giran en torno a factores humanos y organizacionales que pueden afectar la productividad, la comunicación y el compromiso del equipo, estos se miden en la *tabla 6.2.2.*

**Riesgos del Equipo de Trabajo**

| **Categoría** | **Riesgo** | **Probabilidad** | **Impacto** | **Gestión del riesgo** | **Nivel de riesgo** |
| --- | --- | --- | --- | --- | --- |
| **Motivación** | Falta de interés en el proyecto por parte de algunos integrantes | Media | Alta | Definir objetivos claros y dividir tareas según intereses y habilidades | Alto |
| **Responsabilidad** | Bajo nivel de responsabilidad en la entrega de tareas | Alta | Alta | Establecer seguimiento semanal y consecuencias por incumplimiento | Alto |
| **Organización** | Desigual distribución de trabajo dentro del equipo | Alta | Media | Revisión constante de tareas y rotación equitativa | Moderado |
| **Participación** | Falta de participación en reuniones y toma de decisiones | Media | Media | Establecer horarios definidos y delegar funciones con rotación | Moderado |
| **Proactividad** | Poca proactividad para resolver problemas | Media | Media | Fomentar la creatividad e innovación con reconocimiento a aportes individuales | Moderado |
| **Compromiso** | Abandono del proyecto por parte de algún integrante | Baja | Alta | Plan de respaldo para redistribuir tareas rápidamente | Moderado |
| **Personalidad** | Diferencias de personalidad que afecten el trabajo en equipo | Media | Media | Comunicación abierta y mediación por parte de un líder neutral | Moderado |
| **Conflictos** | Falta de respeto o actitudes negativas entre integrantes | Baja | Alta | Establecer reglas de convivencia y resolución de conflictos | Moderado |
| **Problemas externos** | Problemas familiares/personales que afecten el rendimiento | Media | Media | Flexibilidad y apoyo entre integrantes; comunicación continua | Moderado |
| **Desacuerdos** | Dificultad para resolver desacuerdos profesionalmente | Media | Media | Crear acuerdos de resolución; reuniones de conciliación | Moderado |
| **Competencia interna** | Competencia o rivalidad dentro del equipo | Baja | Media | Promover colaboración en lugar de competencia | Bajo |
| **Empatía** | Falta de empatía y apoyo entre integrantes | Baja | Media | Dinámicas de integración y reuniones de feedback positivo | Bajo |

*Tabla 6.3.2 Riesgos del equipo de trabajo*

# Capítulo 8. Conclusiones y trabajo a futuro

**Conclusión**

Nuestra investigación preliminar justifica el desarrollo futuro de un prototipo de aplicación móvil orientado a brindar apoyo informativo a personas migrantes en tránsito. El procesamiento de datos para la base de datos donde se analizó el CSV y se eliminaron los datos no requeridos etiquetando los niveles de criminalidad (alta, media y baja) y se hizo otro CSV con la localización de servicios humanitarios (albergues y comedores), con la selección de la base de datos PostgleSQL.

Para el cálculo de trayectos, se comprobó la funcionalidad de los algoritmos de rutas como A\* con heurística Haversine y Dijsktra, permitiendo establecer recorridos entre organizaciones no gubernamentales (ONG). Asimismo, se empleó un algoritmo de *clustering* (*K-mean*s) para agrupar zonas según su nivel de incidencia delictiva.

Se realizó una selección de una API conforme a las necesidades del prototipo así proponiendo la selección de *OpenStreetMap*, que tiene compatibilidad con *AndroidStudios* que fue la herramienta seleccionada. Aunque se trata de una versión preliminar, la estructura del prototipo permite establecer una base técnica para futuras mejoras, centradas en dispositivos con sistema operativo Android.

**Trabajos a futuro**

Como siguiente etapa, se contempla la integración de todos los elementos desarrollados y seleccionados a lo largo del proyecto, con el objetivo de consolidar un prototipo funcional. Se trabajará en la vinculación de la base de datos con los algoritmos de rutas (A\* con Haversine y Dijkstra) permitiendo establecer recorridos entre organizaciones no gubernamentales (ONG) y *K-Means* para agrupar zonas según su nivel de incidencia delictiva, permitiendo que la aplicación procese e interprete la información. Finalmente el prototipo deberá permitir la interacción directa del usuario con la interfaz móvil, de manera que pueda consultar rutas y la visualización de un mapa que retrate los municipios según su índice de criminalidad.

# Capítulo 9. Referencias

ABAMOBILE. (2024). *¿Cómo publicar una app en las tiendas de descarga? ¿Cuánto cuesta?* Cómo publicar una app en las tiendas de descarga | ABAMobile. <https://abamobile.com/web/como-publicar-app-tiendas-descarga/>

ACNUR. (2024). *ACNUR, la Agencia de la ONU para los Refugiados*. ACNUR. <https://www.acnur.org/mx/>

Acte. (2025). *Algoritmo A\* en IA: Introducción, Implementación, Pseudocódigo*. acte. <https://www.acte.in/a-algorithm-in-ai-article>

Amazon. (2024). *¿Qué es una API? - Explicación de interfaz de programación de aplicaciones*. AWS. <https://aws.amazon.com/es/what-is/api/>

Amazon. (2024). *¿Qué es una aplicación web?* Amazon Web Services. <https://aws.amazon.com/es/what-is/web-application/>

Amazon. (2025). *¿Qué es el machine learning? - Explicación sobre el machine learning empresarial*. AWS. Retrieved May 15, 2025, from <https://aws.amazon.com/es/what-is/machine-learning/>

Anderson, D. J., & Carmichael, A. (2016). *Kanban: Successful Evolutionary Change for Your Technology Business*. Kanban. <https://www.tigalia.com/wp-content/uploads/2020/12/Essential-Kanban-Condensed-Spanish.pdf>

Andrés Areiza, S., Maria Cano, S., Serna, A., & Andres Riascos, J. (2022). Algoritmo para calcular la ruta más segura y óptima. *osf preprints*. <https://doi.org/10.31219/osf.io/wd73v>

Android. (2025). *Cómo diagnosticar fallas por errores en código nativo*. Cómo diagnosticar fallas por errores en código nativo. <https://source.android.com/docs/core/tests/debug/native-crash?hl=es-419>

Apple Developer. (2025). *Xcode*. Apple Developer. <https://developer.apple.com/xcode/>

ArcGIS Pro. (2025). *Cómo funciona la agrupación en clústeres basada en densidad*. esri. <https://pro.arcgis.com/en/pro-app/3.3/tool-reference/spatial-statistics/how-density-based-clustering-works.htm>

BBVA. (2020). *Mapa de casas del migrante, albergues y comedores en las principales rutas de migración por México, 2020*. BBVA. <https://www.bbvaresearch.com/wp-content/uploads/2020/02/Mapa_2020_Albergues_Migrantes_Portable.pdf>

Beck, K. (1999). *Extreme Programming Explained: Embrace Change.* Extreme Programming Explained: Embrace Change. <https://books.google.com.mx/books?hl=es&lr=&id=G8EL4H4vf7UC&oi=fnd&pg=PR13&dq=Beck,+K.+(1999).+Extreme+Programming+Explained:+Embrace+Change.+Addison-Wesley.&ots=jcwKyvhQzj&sig=xpLbVsCeM08QmC7Qi34GGo8RpBk#v=onepage&q&f=false>

Beck, K., Beedle, M., & Cockburn, A. (2001). Manifesto for Agile Software Development. <https://agilemanifesto.org/>

Belwariar, R. (2024, July 30). *A\* Search Algorithm*. GeeksforGeeks. Retrieved May 16, 2025, from <https://www.geeksforgeeks.org/a-search-algorithm/>

Borrego, Á. (2021, 11 25). *Clasificadores bayesianos*. Introducción a la ciencia de datos para estudiantes de Información y Documentación. <https://bookdown.org/angelborrego/ciencia_datos/>

Brown, F. (2025). *11 BEST Free SQL Database Software*. guru99. <https://www.guru99.com/free-database-software.html>

Brown, F. (2025). *Los 11 mejores programas gratuitos de bases de datos SQL*. guru99. <https://www.guru99.com/free-database-software.html>

Bueno Sánchez, E. (2000). Migración: una visión general. In *Definiciones y conceptos sobre la migración*. <https://d1wqtxts1xzle7.cloudfront.net/39717010/capitulo1-_Migracion-libre.pdf?1446737337=&response-content-disposition=inline%3B+filename%3DCapitulo1_Migracion.pdf&Expires=1744266036&Signature=AMZQVcBqkfGjYy3yNj2ITnFEuTA7-r9KH-nUE-8ePLEvqdaCGfzYhgF~RXZNoZK>

Bustos, S. (2021). *Desarrollo de Aplicaciones Web vs. Aplicaciones de Escritorio*. Codster. <https://codster.io/blog/desarrollo-software/desarrollo-de-aplicaciones-web-vs-escritorio/>

*Calculate distance and bearing between two Latitude/Longitude points using haversine formula in JavaScript*. (n.d.). Movable-type.co.uk. Retrieved June 18, 2025, from <https://www.movable-type.co.uk/scripts/latlong.html>

Caleb, R., & Bistra, D. (2018). A Machine Learning Approach to Modeling Human Migration. *CrossMark*. <https://doi.org/10.1145/3209811.3209868>

Castelo, M. (2023). *Modelado de datos NoSQL y Firebase*. NoCode StartUp. <https://nocodestartup.io/es/modelado-de-datos-nosql-firebase/>

*Connect to the network | Connectivity*. (2023). Android Developers. <https://developer.android.com/training/basics/network-ops/connecting>

Cormen, T. H. (2013). *Algorithms Unlocked*. Cambridge, MA.

developer. (2025). *Cómo instalar Android Studio*. Android Developers. Retrieved May 19, 2025, from <https://developer.android.com/studio/install?hl=es-419>

Developer. (2025). *Swift*. Apple Developer. https://developer.apple.com/swift/

devx. (2023). *Base de datos híbrida*. devx. <https://www.devx.com/terms/hybrid-database/>

*Documentación de Google Maps Platform*. (2024). Google for Developers. <https://developers.google.com/maps/documentation?hl=es-419>

Erickson, J. (2024, August 29). *MySQL: qué es y cómo se usa*. Oracle. Retrieved May 12, 2025, from <https://www.oracle.com/mx/mysql/what-is-mysql/>

eseri. (2025). *Definición de ruta*. eseri. <https://support.esri.com/es-es/gis-dictionary/search?q=Ruta>

Expósito Izquierdo, C., Expósito Márquez, A., López Plata, I., Melián Batista, B., & Moreno Vega, J. M. (2025). *Clustering jerarquico*. Universidad de La Laguna.

Flutter. (2025). Flutter - Build apps for any screen. <https://flutter.dev/>

Garfas Royo, M., Parikh, P., & Belur, J. (2020). *Using heat maps to identify areas prone to violence against women in the public sphere*. Garfas Royo et al. <https://doi.org/10.1186/s40163-020-00125-6>

Gartner. (2025). *Understand 3 key types of machine learning*. <https://www.gartner.com/smarterwithgartner/understand-3-key-types-of-machine-learning>

GetApp. (2025). *Ionc*. Ionic: precios, funciones y opiniones | GetApp México 2025. <https://www.getapp.com.mx/software/2056178/ionic#:~:text=Ionic%20ofrece%20los%20siguientes%20planes,Prueba%20gratis%3A%20Disponible>

Gobierno de México. (2024). *Política Migratoria del Gobierno de México*. portales.segob.gob.mx. <https://portales.segob.gob.mx/es/PoliticaMigratoria/Panorama_de_la_migracion_en_Mexico>

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Google. (2025). *Autenticación de IAM*. Google Cloud. <https://cloud.google.com/sql/docs/mysql/iam-authentication?utm_source&hl=es-419>

Google. (2025). *¿Qué es una base de datos relacional (RDBMS)?* Google Cloud. Retrieved May 15, 2025, from <https://cloud.google.com/learn/what-is-a-relational-database?hl=es-419>

Google Developers Training team. (2025). *Cómo crear tu primera app para Android*. Android Developers. <https://developer.android.com/codelabs/basic-android-kotlin-compose-first-app?hl=es-419#0>

Herazo, L. (2022). *¿Qué es una aplicación móvil?* Anincubator. <https://anincubator.com/que-es-una-aplicacion-movil/>

hireline. (2025). *Perfil y Salario de un Perfil de ingeniero en Inteligencia Artificial en México en 2025*. Hireline. Retrieved May 15, 2025, from <https://hireline.io/mx/enciclopedia-de-perfiles-de-tecnologia/ingeniero-en-inteligencia-artificial>

hireline. (2025). *Perfil y sueldo de un Perfil de Desarrollador Backend en México 2025*. hireline. <https://hireline.io/mx/enciclopedia-de-perfiles-de-tecnologia/desarrollador-backend#:~:text=El%20promedio%20del%20salario%20neto,2025%20es%20de%20%2431%2C220.00%20MXN>.

hireline. (2025). *Perfil y sueldo de un Perfil de Desarrollador Móvil en México 2025*. hireline. <https://hireline.io/mx/enciclopedia-de-perfiles-de-tecnologia/desarrollador-movil#:~:text=El%20promedio%20del%20salario%20neto,2025%20es%20de%20%2435%2C168.00%20MXN>.

Hoover, R. (2019). *MANUAL PARA LOS MIGRANTES*. cndh. <https://www.cndh.org.mx/sites/default/files/documentos/2019-05/manual-migrantes.pdf>

Huet, P. (2023, April 13). *Qué son las redes neuronales y sus aplicaciones*. OpenWebinars. Retrieved May 15, 2025, from <https://openwebinars.net/blog/que-son-las-redes-neuronales-y-sus-aplicaciones/>

IBM. (2021). *¿Qué es PostgreSQL?* IBM. Retrieved May 12, 2025, from <https://www.ibm.com/mx-es/topics/postgresql>

IBM. (2023). *Desarrollo de aplicaciones móviles.* IBM. <https://www.ibm.com/mx-es/topics/mobile-application-development>

IBM. (2024). *What Is Mobile Application Development?* IBM. <https://www.ibm.com/think/topics/mobile-application-development>

IBM. (2025). *¿Qué es la agrupación en clústeres k-means?* IBM. Retrieved June 3, 2025, from <https://www.ibm.com/mx-es/think/topics/k-means-clustering>

INEGI. (2025). *Incidencia delictiva*. INEGI. <https://www.inegi.org.mx/temas/incidencia/?utm_source>

Instituto Nacional de Migración. (2022). *Grupos Beta de Protección a Migrantes*. gob.mx. <https://www.gob.mx/inm/acciones-y-programas/grupos-beta-de-proteccion-a-migrantes>

Interaction Design Foundation. (2025). *Mobile User Experience (UX) Design*. What is Mobile User Experience (UX) Design? <https://www.interaction-design.org/literature/topics/mobile-ux-design>

interactivechaos. (2025). *Tutorial de Machine Learning*. interactivechaos. <https://interactivechaos.com/es/manual/tutorial-de-machine-learning/tutorial-de-machine-learning>

International Rescue Committee. (n.d.). *Comité Internacional de Rescate (IRC)*. rescue. <https://www.rescue.org/somos-irc>

IONOS. (2023, January 31). *¿Qué es el modelo cliente-servidor? Pros y contras*. IONOS. Retrieved June 17, 2025, from <https://www.ionos.mx/digitalguide/servidores/know-how/modelo-cliente-servidor/>

ISO. (2024). *Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality model overview and usage*. ISO/IEC 25002:2024. <https://www.iso.org/standard/78175.html>

Klippa. (2023). *¿Qué son las reglas de negocio y por qué son importantes?* Klippa. <https://www.klippa.com/es/blog/informativo/reglas-de-negocio/>

*Kotlin and Android*. (2023). Android Developers. <https://developer.android.com/kotlin>

Ladas, C. (2009). *Scrumban: Essays on Kanban Systems for Lean Software Development*. Scrumban-essays on kanban systems for lean software development. <https://books.google.es/books?hl=es&lr=&id=SQFdAgAAQBAJ&oi=fnd&pg=PA7&dq=Ladas,+C.+(2009).+Scrumban:+Essays+on+Kanban+Systems+for+Lean+Software+Development.+Modus+Cooperandi+Press.&ots=cb71YJVzSj&sig=fe9jWoCstxdvIcXE-i6jC8LwItA#v=onepage&q=Ladas%2C%20C.%20>

Li, Y., Ming Su, L., & Liang Li, W. (2012). *Hierarchical Path-Finding Based on Decision Tree*. Springer Nature Link. <https://link.springer.com/chapter/10.1007/978-3-642-31900-6_32>

Llanos Reynoso, L. F. (2023). Los caminos de migrantes que pasan por México: análisis desde la Teoría de Grafos. In *Migraciones Internacionales* (Vol. VOL. 14, p. 29). El colegio de la Frontera Norte. <https://doi.org/10.33679/rmi.v1i1.2581>

LOPEZ TAKEYAS, B. (2005). *ALGORITMO A \**. INSTITUTO TECNOLÓGICO DE NUEVO LAREDO. <https://nlaredo.tecnm.mx/takeyas/Apuntes/Inteligencia%20Artificial/Apuntes/tareas_alumnos/A-Star/A-Star(2005-II-B).pdf>

Luna, J. (2018, February 8). *Tipos de aprendizaje automático. La Inteligencia Artificial (IA) está en… | by Javier Luna Gonzalez | SoldAI*. Medium. Retrieved April 23, 2025, from <https://medium.com/soldai/tipos-de-aprendizaje-autom%C3%A1tico-6413e3c615e2>

*Machine Learning en la Industria*. (2025). Home - ATRIA. Retrieved April 23, 2025, from <https://atriainnovation.com/>

*Maps SDK | Android Docs | Mapbox*. (2023). Maps SDK | Android Docs | Mapbox. <https://docs.mapbox.com/android/maps/guides/>

mariadb. (2025). *MariaDB en resumen*. mariadb. <https://mariadb.org/es/>

McCarthy, J. (2007). *WHAT IS ARTIFICIAL INTELLIGENCE*. stanford. <http://jmc.stanford.edu/articles/whatisai/whatisai.pdf>

Microsoft Build. (2025). *Documentación de Xamarin*. Conectar, codificar y crecer. <https://learn.microsoft.com/en-us/previous-versions/xamarin/>

Ng, A. (n.d.). *The Elements of Statistical Learning*. Penn Arts & Sciences. Retrieved May 9, 2025, from <https://www.sas.upenn.edu/~fdiebold/NoHesitations/BookAdvanced.pdf>

Nunez, L. (2024). *Tipos de aplicaciones, características, ejemplos y comparativa*. EMMA. <http://emma.io/blog/tipos-aplicaciones-caracteristicas-ejemplos/>

Olivera, A. (2022, November 7). *Bases de datos relacionales | Alura Cursos Online*. Alura Latam. <https://www.aluracursos.com/blog/base-de-datos-relacional>

*OpenStreetMap (OSM)*. (2022). OpenStreetMap-Tools for Android. <https://github.com/osmdroid/osmdroid>

Organización de las Naciones Unidas (ONU). (2018). *Migración internacional | Naciones Unidas*. the United Nations. <https://www.un.org/es/global-issues/migration>

Organización Internacional para las Migraciones. (2024). BOLETÍN DE ESTADÍSTICAS MIGRATORIAS PARA MÉXICO. *OIM ONU Migración*. <https://mexico.iom.int/sites/g/files/tmzbdl1686/files/documents/2024-07/oim-boletin-estadisticas-migratorias-mexico-t12024.pdf>

Pascual, D., Pla, F., & Sánchez, S. (2007). *Algoritmos de agrupamiento*. <https://agrarias.campus.mdp.edu.ar/pluginfile.php/129/course/section/1500/Los%20algoritmos%20de%20agrupamientos.%20Pascual%2C%20D.%20%2C%20Pla%2C%20F.%20y%20S.%20S%C3%A1nchez.%20Cap%C3%ADtulo%20del%20libro%20M%C3%A9todos%20inform%C3%A1ticos%20avanzados.pdf>

Pham Dinh, T., Pham-Quoc, C., Ngoc Thinh, T., Do Nguyen, B. K., & Cong Kha, P. (2023). A flexible and efficient FPGA-based random forest architecture for IoT applications. *Internet of Things*, *Vol. 22*(100813). <https://doi.org/10.1016/j.iot.2023.100813>

PMOinformatica. (2024, August 6). *Diagrama de casos de uso: Definición - La Oficina de Proyectos de Informática*. PMOInformatica. Retrieved April 29, 2025, from <https://www.pmoinformatica.com/2021/02/diagrama-de-casos-de-uso-definicion-elementos-ejemplos-como-hacer.html?utm_source=chatgpt.com>

postgresql. (2025). *Documentation: 17: 17.1. Requirements*. PostgreSQL. Retrieved May 19, 2025, from <https://www.postgresql.org/docs/current/install-requirements.html>

railway. (2025). *Pricing Plans*. Railway Docs. Retrieved June 5, 2025, from <https://docs.railway.com/reference/pricing/plans>

Ramírez, L. (2024, October 30). *Algoritmo k-means: ¿Qué es y cómo funciona?* IEBS. Retrieved June 5, 2025, from <https://www.iebschool.com/hub/algoritmo-k-means-que-es-y-como-funciona-big-data/>

React Native. (2025, April 14). *Introduction · React Native*. React Native. <https://reactnative.dev/docs/getting-started>

Richaud, A. (2024). *¿Qué es y para qué sirve el algoritmo Dijkstra?* blog antonio-richaud. <https://antonio-richaud.com/blog/archivo/publicaciones/20-dijkstra.html>

Rus, E. (2020, September 1). *Factibilidad operativa*. Economipedia. Retrieved May 15, 2025, from <https://economipedia.com/definiciones/factibilidad-operativa.html>

*ruta | Definición | Diccionario de la lengua española | RAE - ASALE*. (n.d.). Diccionario de la lengua española. Retrieved April 23, 2025, from <https://dle.rae.es/ruta?m=form>

Saltos Pérez, L. A. (2022). Estudio comparativo entre bases de datos relacional y no relacional. <http://dspace.utb.edu.ec/handle/49000/11669>

saludelectronica. (2022). *Base de datos SQL vs NO SQL*. saludelectronica. <https://saludelectronica.com/base-de-datos-sql-vs-no-sql/>

Samuel Hall. (2023). *Mapping Nyumbani*. medium. <https://medium.com/samuel-hall-stories/mapping-nyumbani-the-project-creating-pathways-to-safe-spaces-for-refugees-c06d68140bd0>

Sanchez, S. (2025). *Diagrama de árbol de decisión: Cómo hacerlo + 8 ejemplos*. DolarApp. Retrieved May 15, 2025, from <https://www.dolarapp.com/es-MX/blog/freelancer-tips/que-es-diagrama-arbol-decision-ejemplos>

Sandoval, L. J. (2018). *ALGORITMOS DE APRENDIZAJE AUTOMÁTICO PARA ANÁLISIS Y PREDICCIÓN DE DATOS*. REDICCES. Retrieved April 24, 2025, from <http://redicces.org.sv/jspui/bitstream/10972/3626/1/Art6_RT2018.pdf>

Secretariado Ejecutivo del Sistema Nacional de Seguridad Pública. (2025). *Datos Abiertos de Incidencia Delictiva En esta págin*. gob.mx. <https://www.gob.mx/sesnsp/acciones-y-programas/datos-abiertos-de-incidencia-delictiva?utm>

Sinnot. (2019, abril 29). Virtues of the Haversine. *Journal of Geographic Information System*, *68*(2), 159. <https://www.scirp.org/reference/referencespapers?referenceid=2500014>

Sistema Nacional de información estadística y geográfica. (2025). *Catálogo Nacional de Indicadores*. Catálogo Nacional de Indicadores. Retrieved April 24, 2025, from <https://www.snieg.mx/cni/escenario.aspx?idOrden=1.1&ind=6200028526&gen=989&d=n>

Sutherland, J., & Schwaber, K. (2020). Scrum Guides: Home. https://scrumguides.org/

Veness, C. (2002, agosto 1). *Calcular distancia, rumbo y más entre puntos de latitud/longitud*. Scripts de Movable Type. <https://www.movable-type.co.uk/scripts/latlong.html>

Visure solutions. (2025). *Requisitos funcionales y no funcionales (con ejemplos)*. Visure Solutions. Retrieved May 15, 2025, from <https://visuresolutions.com/es/requirements-management-traceability-guide/functional-vs-non-functional-requirements/>

Walsh, D. (2023, September 28). *Mockup: qué es, para qué sirve y ejemplos inspiradores*. Blog de HubSpot. <https://blog.hubspot.es/website/mockup-que-es#que-es>

Williams, A. (2012). *When to Use NoSQL: A Guide for Beginners*. Butter CMS. <https://buttercms.com/blog/when-to-use-nosql/>

Wilson, R. J. (1996). *Introduction to Graph Theory*. Longman.

Winston W., R. (1970). *NAGING THE DEVELOPMENT OF LARGE SOFTWARE SYSTEMS*. Proceedings of IEEE WESCON. <https://www.praxisframework.org/files/royce1970.pdf>

Yenigün, O. (2024, March 11). *DBSCAN Clustering Algorithm Demystified*. Built In. Retrieved June 5, 2025, from <https://builtin.com/articles/dbscan>

Zhou, Z.-H. (2021). *Machine Learning* (S. Liu, Trans.). Springer Nature Singapore.