

**Universidad Peruana de Ciencias Aplicadas**

**INFORME DEL TRABAJO FINAL (TB2)**

1ACC0184 - Complejidad Algorítmica

**Sección: 155**

**Profesor:** Cesar Enrique, Salas Arbaiza

**Carrera:** Ingeniería de Software

**Tema:** Desarrollo de un software basado en algoritmos complejos para optimizar rutas de intervención ante zonas de riesgo vinculados al cambio climático en Lima metropolitana.

de Lima - Perú

**Integrantes:**

| Alumnos: | |
| --- | --- |
| Código | Nombres y apellidos |
| U202317269 | Bonifacio Jaramillo, Samuel Jesus |
| U202116401 | Ruiz Madrid, Billy Jake |
| U202222942 | David Alexander Perez Garcia |

**2025-1**

**Tabla de Contenido**

# 

**1. Descripción del problema:** Redactar la descripción y fundamentación del problema citando fuentes.

**2. Descripción del conjunto de datos (dataset):** Redactar las características y origen de los datos motivo de análisis.

**3. Propuesta:** Redactar el objetivo de la propuesta, así como la(s) técnica(s) y metodología(s) utilizada(s).

**4. Diseño del aplicativo:** Describir los procesos del diseño del aplicativo considerando las etapas de la ingeniería de software o considerando el análisis de algoritmos según sea el caso.

**5. Validación de resultados y pruebas:** Describir las entradas y salidas, interpretación de resultados y pruebas.

**6. Conclusiones:** En un párrafo redactar las conclusiones del trabajo, especificando la(s) técnica(s) utilizadas y, de ser el caso, el trabajo a futuro.

**7. Referencias bibliográficas:** Según APA 7.0

# 1. Descripción del Problema

El Perú enfrenta una problemática crítica debido a su alta exposición y vulnerabilidad frente a desastres naturales intensificados por el cambio climático, afectando profundamente su desarrollo social, económico y ambiental. En Lima Metropolitana, eventos como lluvias extremas, huaicos e inundaciones han generado severos impactos en zonas urbanas y periféricas, especialmente en distritos como Chosica, San Juan de Lurigancho y Ate, donde las condiciones topográficas complejas, la expansión urbana desordenada y la limitada infraestructura vial agravan la situación (MINAM, 2013).

Estas condiciones dificultan significativamente la capacidad de respuesta rápida por parte de instituciones de emergencia como bomberos, centros de salud y defensa civil. La falta de rutas claras, el congestionamiento vial y la escasa planificación territorial generan cuellos de botella críticos durante eventos de emergencia, lo que retrasa la llegada de ayuda y aumenta los riesgos para la población afectada.

El Objetivo de Desarrollo Sostenible (ODS 13), "Acción por el Clima", llama a adoptar medidas urgentes para fortalecer la resiliencia ante desastres relacionados con el clima (UN, 2015). En ese marco, se vuelve imprescindible diseñar sistemas tecnológicos que permitan planificar rutas de intervención más eficientes, considerando tanto la red vial como los puntos de riesgo y los centros de respuesta.

El presente trabajo propone una solución algorítmica basada en el uso de grafos ponderados y datos geoespaciales extraídos de OpenStreetMap y la plataforma SIGRID del CENEPRED. En particular, se emplea el algoritmo de Bellman-Ford para calcular rutas óptimas desde centros de respuesta hacia zonas de riesgo, permitiendo encontrar caminos mínimos considerando el tiempo de viaje como peso principal. Esta elección responde a la capacidad del algoritmo para manejar grafos con pesos negativos y a su aplicabilidad en redes dirigidas con ponderaciones realistas como las del tráfico vial.

Además, se aplica el algoritmo de flujo máximo de Edmonds-Karp (una variante del algoritmo de Ford-Fulkerson) para asignar eficientemente los centros de respuesta a múltiples zonas de riesgo, maximizando la cobertura institucional ante emergencias simultáneas. Este enfoque permite modelar y resolver situaciones donde los recursos son limitados y se requiere tomar decisiones óptimas bajo restricciones.

La optimización algorítmica de rutas de emergencia con soporte geoespacial se convierte así en una herramienta clave para reducir los tiempos de respuesta, priorizar zonas críticas y mejorar la eficiencia operativa ante escenarios de desastre (Alamoudi et al., 2024).

# 2. Descripción del conjunto de datos (dataset):

**2.1 Origen de los datos y motivo de análisis:**

**Origen de datos para el dataset “Caminos de Lima”:**  
 Este conjunto de datos corresponde a la red vial de Lima Metropolitana y fue extraído desde la plataforma BBBike en formato shapefile. Contiene información detallada sobre la estructura de las vías, incluyendo geometría lineal, nombre de las calles y tipo de vía. A partir de esta información se construyó una representación en forma de grafo, donde los nodos representan intersecciones viales y las aristas corresponden a los tramos de carretera que las conectan. Esta representación es fundamental para simular desplazamientos urbanos y analizar rutas óptimas en el contexto de movilidad y accesibilidad. El grafo resultante supera los 1500 nodos, lo que permite realizar análisis topológicos complejos, detectar cuellos de botella y evaluar la eficiencia de la red vial.

Link del dataset:<https://download.bbbike.org/osm/bbbike/Lima>

**Origen de datos para el dataset “Instituciones de respuesta ante desastres”:**  
 Este conjunto de datos contiene la ubicación geográfica de instituciones de respuesta ante desastres naturales, incluyendo estaciones de bomberos, centros de salud y otras entidades de primera atención. Extraído desde la plataforma SIGRID del CENEPRED, el archivo en formato shapefile tipo punto incluye atributos clave como el tipo de institución, su nombre y su capacidad operativa. Esta información fue incorporada al análisis geoespacial como nodos especiales dentro de la red, permitiendo evaluar el acceso y la cercanía de estas instituciones frente a diferentes escenarios de emergencia. Además, facilitó la identificación de zonas con baja cobertura institucional para proponer estrategias de mejora en la respuesta ante desastres.

Link del dataset:<https://sigrid.cenepred.gob.pe/sigridv3/>

**Origen de datos para el dataset “Puntos de alto riesgo a desastres naturales”:**  
 Este conjunto de datos describe puntos críticos de alto riesgo ante desastres naturales en Lima, tales como zonas vulnerables a sismos, inundaciones o deslizamientos. Obtenido también desde la plataforma SIGRID del CENEPRED, se presenta en formato shapefile tipo punto, e incluye atributos como el tipo de riesgo, nivel de amenaza y estimaciones de población potencialmente afectada. Estos puntos fueron integrados al modelo geoespacial para evaluar la exposición de la infraestructura y la población, así como para planificar rutas de evacuación y estrategias de mitigación. Su análisis permitió identificar zonas prioritarias donde es fundamental mejorar la infraestructura o la cobertura de servicios de emergencia.

Link del dataset:<https://sigrid.cenepred.gob.pe/sigridv3/>

DataShape 1: Ubicación de Centros de Respuesta

| Nombre | Departamento | Provincia | Distrito | Direccion | Capacidad Operativa | Fecha de Publicación |
| --- | --- | --- | --- | --- | --- | --- |
| Adolfo Martin King | Lima | Callao | Callao | ELMER FAUCETT | - | 11/4/22 19:00 |
| Antonio Alarco Espinoza | Lima | Callao | Ventanilla | Av.Panama Mz K Ltt | 33 | 11/4/22 19:00 |
| Ate | Lima | Lima | Ate | Sin datos | 152 | -19:00 |
| Barranco | Lima | Lima | Barranco | Sin datos | Sin dato | -19:00 |

**Motivo de Análisis:** Este dataset, que incluye la ubicación geográfica de instituciones de respuesta ante desastres como estaciones de bomberos y centros de salud , se analiza para determinar la distribución espacial y la cobertura de estos servicios de emergencia.

**Fundamentación:** Al integrar esta información en la red de caminos, es posible calcular los tiempos de respuesta y las rutas de acceso. Esto permite identificar áreas con baja o nula cobertura institucional en caso de una emergencia, lo que es crucial para una planificación efectiva.

DataShape 2: Zonas de Riesgo por Desastres Naturales

|  | ID Documento | Nombre | Tipo Documento | Año | Ciudad | Editorial |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 16145 | Informe de evaluación del riesgo por peligro sismo en el cerro de la Milla | EVAR | 2023 | Lima | SIGRID |
| 2 | 17803 | Informe de evaluación de riesgo de desastres por derrumbe y caída | EVAR | 2024 | Lima | SIGRID |
| 3 | 16302 | Informe de evaluación de riesgo de desastres por sismo en el AAHH | EVAR | 2022 | Lima | SIGRID |
| 4 | 17802 | Informe de evaluación de riesgo de desastres por derrumbe y caída | EVAR | 2024 | Lima | SIGRID |
| 5 | 14877 | Informe de evaluación de riesgo por licuación de suelos | EVAR | 2021 | Lima | SIGRID |
| 6 | 18047 | Informe de evaluación de riesgo por inundación fluvial | EVAR | 2024 | Lima | SIGRID |

**Motivo de Análisis:** Este conjunto de datos describe puntos críticos de alto riesgo ante desastres naturales en Lima, como zonas vulnerables a sismos, inundaciones o deslizamientos. Se analiza para identificar las áreas más vulnerables dentro del entorno urbano.

**Fundamentación:** La integración de estos puntos en el modelo espacial permite evaluar el grado de exposición de la infraestructura y la población, así como planificar rutas de evacuación y estrategias de mitigación efectivas. Su análisis es fundamental para priorizar intervenciones y mejorar la infraestructura o la cobertura de servicios de emergencia en zonas críticas.

DataShape 3: Red vial

|  | Departamento | Provincia | Distrito | Nombre | Estado | Codigo Ruta | Tipo Superficie | Jerarquia | Longitud (Km) | Observacion | Fuente logo |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | Callao | Callao | Callao | [EMP.PE](http://emp.pe)-20B (Av.Miguel Grau(ex…)) | Mala | L -400 | 1 | RN | 0.23 | Ancho: > 5.5 | MTC |
| 2 | Callao | Callao | Callao | Av. Miguel Grau(ex…)) | 1 | L - 100 | 1 | RD | 2.26 | Ancho: > 5.5 | MTC |
| 3 | Callao | Callao | Callao | Av. Miguel Grau(ex…)) | 1 | L - 100 | 1 | RD | 0.16 | Nro carril: 3 | MTC |
| 4 | Callao | Callao | Callao | Emp. PE-20 (Ovalo) | Regular | L - 100 | 1 | RN | 0.38 | Ancho: > 5.5 | MTC |
| 5 | Callao | Callao | Callao | Av. Miguel Grau(ex…)) | 1 | L - 100 | 1 | RD | 0.48 | Ancho: > 5.5 | MTC |

**Motivo de Análisis:** Este dataset corresponde a la red vial de Lima Metropolitana, extraído de la plataforma BBBike. Permite modelar la red de transporte terrestre como un grafo.

**Fundamentación:** La representación de la red vial como un grafo es esencial para simular desplazamientos, calcular rutas óptimas y analizar la conectividad urbana. Esto es clave para comprender cómo se puede acceder a diferentes zonas, especialmente en contextos de emergencia, facilitando la identificación de cuellos de botella y la evaluación de la eficiencia de la red vial.

**2.2 Relación con el grafo:**

La estructura de grafo es fundamental en el diseño e implementación del sistema propuesto. La red vial de Lima Metropolitana se modela como un **grafo dirigido y ponderado**, donde:

* **Los nodos** representan intersecciones o puntos clave del mapa (coordenadas geográficas con latitud y longitud).
* **Las aristas** representan los tramos de vía entre estos puntos, cada una asociada a pesos que indican **la distancia (en metros)** y **el tiempo estimado de viaje (en segundos)**.

Además, los **nodos de respuesta** (instituciones como bomberos, centros de salud) y los **nodos de riesgo** (zonas vulnerables a desastres) son georreferenciados y se asocian dinámicamente con los nodos más cercanos dentro del grafo vial mediante una búsqueda espacial de mínima distancia. De esta manera, cada punto de respuesta o riesgo se conecta con la red de caminos existente, permitiendo su inclusión en el análisis algorítmico.

Esta modelación permite aplicar algoritmos clásicos de teoría de grafos sobre una estructura espacialmente coherente y realista.

Dos algoritmos principales operan sobre este grafo:

1. **Bellman-Ford**, aplicado para encontrar el **camino más corto (óptimo)** entre un nodo de respuesta y un nodo de riesgo, minimizando el tiempo de viaje. Se emplea sobre un subgrafo reducido geográficamente para mejorar la eficiencia.
2. **Algoritmo de Flujo Máximo (Edmonds-Karp)**, utilizado para asignar múltiples centros de respuesta a múltiples zonas de riesgo, maximizando el número de atenciones posibles. Aquí, el grafo se expande con nodos virtuales de fuente y sumidero, y las capacidades modelan la disponibilidad de intervención.

Gracias a estas representaciones, el sistema puede ofrecer decisiones estratégicas de intervención en tiempo real, optimizando recursos y priorizando rutas en escenarios de desastre.

**2.3 Visualización del dataset:**

Se utilizó la herramienta Gephi para subir los dos archivos CSV, SIGRID\_RESPUESTA.csv y SIGRID\_RIESGO.csv. Para visualizar un grafo dirigido en Gephi por cada conjunto de nodos (respuesta y riesgo).Se observa la estructura de los nodos siguiente:

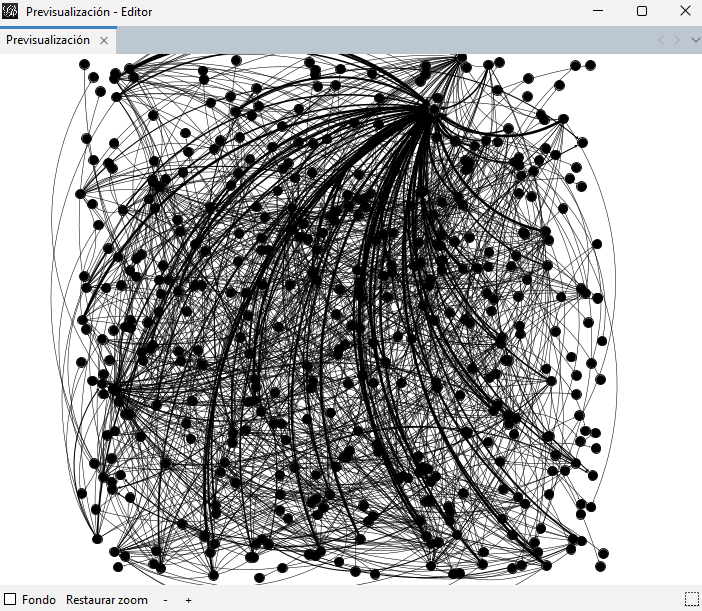
Link del repositorio que contiene los datos csv y los archivos gephi

https://github.com/GRUPO-8-COMPLEJIDAD-ALGORITMICA/backend-fast-api

**Centros de Respuesta:**



**Zonas de riesgo:**



# 3. Propuesta

# 3.1 Objetivo General

Desarrollar un sistemacapaz de optimizar la asignación y el trazado de rutas de intervención ante desastres naturales en Lima Metropolitana, utilizando algoritmos de Bellman-Ford para determinar rutas óptimas en función del tiempo de viaje y el algoritmo de flujo máximo (Edmonds-Karp) para asignar eficientemente los centros de respuesta a las zonas de riesgo, a partir de datos geoespaciales integrados desde redes viales y puntos críticos de emergencia.

## 3.2 Objetivos Específicos

* **Modelar la red vial de Lima Metropolitana como un grafo dirigido y ponderado**, utilizando datos geoespaciales provenientes de OpenStreetMap y almacenados en una base de datos PostGIS, donde los nodos representan intersecciones y las aristas representan tramos de vía con atributos como longitud y tiempo de viaje.
* **Implementar el algoritmo de Bellman-Ford** para calcular rutas óptimas entre puntos de respuesta y zonas de riesgo, priorizando el tiempo de viaje como criterio de optimización y garantizando resultados incluso en presencia de posibles pesos negativos o condiciones de tráfico dispares.
* **Aplicar el algoritmo de flujo máximo (Edmonds-Karp)** para asignar eficientemente múltiples centros de respuesta a múltiples puntos de riesgo, maximizando la cobertura institucional bajo restricciones de capacidad y conectividad.
* **Desarrollar un backend en FastAPI** que exponga endpoints para registrar puntos geoespaciales, consultar rutas óptimas y gestionar asignaciones de flujo, facilitando su consumo por aplicaciones externas como sistemas de monitoreo o interfaces web.
* **Optimizar la eficiencia computacional de los algoritmos implementados** mediante el uso de subgrafos geográficos basados en un radio dinámico entre puntos de interés, reduciendo así la carga de procesamiento sin comprometer la precisión de los resultados.
* **Integrar cálculos adicionales como distancia total, tiempo estimado y velocidad promedio** para cada ruta, proporcionando métricas claras y útiles para la toma de decisiones en contextos de emergencia.

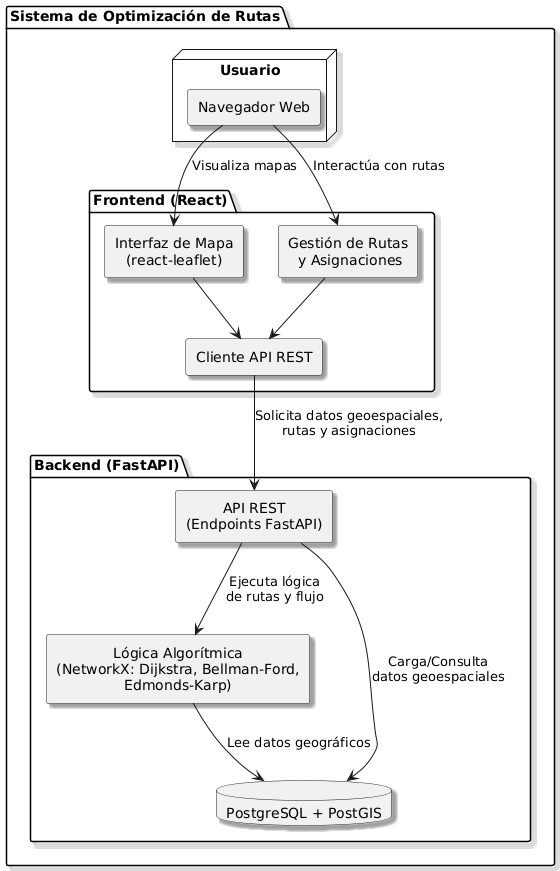
# 3.3 Técnicas y metodologías a usar:

1. **Modelado de Grafos Dirigidos Ponderados:** Se representa la red vial como un grafo dirigido donde los nodos corresponden a intersecciones viales (extraídas desde una base de datos PostGIS), y las aristas representan los tramos de carretera, ponderados según el tiempo estimado de viaje (travel\_time) y la longitud del tramo (length).
2. **Algoritmo de Bellman-Ford:** Utilizado para encontrar la ruta más corta entre un punto de respuesta y un punto de riesgo. Se eligió Bellman-Ford por su capacidad para trabajar con grafos dirigidos y su tolerancia a pesos negativos (aunque no se usan en este caso). Este algoritmo se implementa mediante networkx.bellman\_ford\_path().
3. **Algoritmo de Flujo Máximo (Edmonds-Karp):** Se utiliza para asignar múltiples puntos de respuesta a múltiples puntos de riesgo maximizando el número de atenciones posibles. Implementado con networkx.maximum\_flow(), este método permite modelar la asignación de recursos limitados (ej. ambulancias, bomberos) de forma óptima.
4. **Subgrafo Geográfico Dinámico:** Para reducir la carga computacional, se aplica una técnica de filtrado geográfico mediante subgrafos creados dinámicamente según el radio entre los puntos de interés (respuesta y riesgo), limitando la búsqueda de caminos a una región relevante.
5. **Cálculo de Métricas Complementarias:** Se calculan valores adicionales como la distancia total (en metros), el tiempo estimado (en segundos/minutos), y la velocidad promedio (en km/h) para cada ruta generada. Esto proporciona una evaluación más completa y realista del trayecto.
6. **Desarrollo Backend con FastAPI:** Se utiliza FastAPI como framework principal para exponer endpoints RESTful, que permiten registrar puntos, generar rutas óptimas, obtener rutas más largas y asignar flujos. Todo el backend es integrable con una base de datos PostgreSQL + PostGIS.
7. **Persistencia Geoespacial en PostGIS:** Se almacenan los puntos de respuesta y riesgo en tablas espaciales usando geometrías POINT con SRID 4326. Esto permite realizar consultas espaciales precisas y una integración eficiente con librerías geográficas.

## 4. Diseño del aplicativo

El diseño del aplicativo se basa en los principios de la ingeniería de software y se estructura siguiendo un enfoque modular, escalable y orientado a servicios. El sistema está compuesto por dos grandes componentes: un frontend desarrollado en React para la interfaz de usuario y un backend construido con FastAPI que implementa la lógica algorítmica y maneja la persistencia de datos mediante PostgreSQL con extensión PostGIS.

#### **4.1 Etapas de desarrollo**

1. **Análisis de requisitos** Se identificaron dos tipos principales de usuarios: planificadores de emergencia y personal técnico. Las funcionalidades clave incluyen:  
   * Registro de puntos de riesgo y centros de respuesta.
   * Visualización geoespacial de nodos.
   * Consulta de rutas óptimas y análisis de asignación de flujo.
   * Interacción con los algoritmos para la toma de decisiones.
2. **Diseño de arquitectura  
   **
   * **Frontend (React):**
     + Implementa mapas interactivos mediante react-leaflet.
     + Permite al usuario registrar, visualizar y consultar rutas y asignaciones.
     + Consume los endpoints del backend vía API REST.
   * **Backend (FastAPI):**
     + Expone endpoints para la carga de datos, consulta de caminos y ejecución de asignación de flujo.
     + Implementa lógica algorítmica usando NetworkX para Bellman-Ford y Edmonds-Karp.
     + Utiliza PostgreSQL/PostGIS para almacenamiento geoespacial.
3. **Modelado algorítmico**
   * El mapa de Lima Metropolitana se representa como un **grafo dirigido y ponderado**, donde los nodos son puntos geográficos y las aristas son tramos viales con peso en tiempo y distancia.
   * Se utilizan algoritmos clásicos:  
     + **Bellman-Ford** para rutas óptimas entre pares de nodos.
     + **Edmonds-Karp** para asignar rutas entre múltiples nodos de riesgo y respuesta maximizando la cobertura.
   * Se optimiza el rendimiento aplicando subgrafos locales según cercanía geográfica para reducir complejidad computacional.
4. **Implementación**
   * Se desarrolló una base de datos espacial en PostgreSQL con extensiones GIS.
   * Se cargaron datasets reales desde SIGRID (CENEPRED) y OpenStreetMap.
   * Se implementaron funciones reutilizables, cacheadas y desacopladas para cada operación del sistema.
5. **Pruebas y validación**
   * Se realizaron pruebas funcionales con escenarios reales (por ejemplo, rutas desde estaciones de bomberos a zonas vulnerables).
   * Se validaron los resultados visualmente a través del frontend.
6. **Despliegue**
   * Se prevé su despliegue en entorno local para pruebas de campo y su posterior adaptación a nube si se requiere escalabilidad.

#### **4.2 Ventajas del diseño propuesto**

* Separación clara entre lógica de negocio (algoritmos) y presentación (UI).
* Uso de tecnologías modernas, ligeras y de alto rendimiento.
* Facilidad para integrar nuevas fuentes de datos o algoritmos adicionales.
* Adaptabilidad a distintas zonas geográficas mediante carga dinámica de grafos.

### **4.3 Requisitos del Aplicativo**

#### **4.3.1 Requisitos Funcionales**

* RF1. El sistema debe permitir registrar puntos de riesgo (zonas vulnerables) con sus coordenadas geográficas.
* RF2. El sistema debe permitir registrar puntos de respuesta (estaciones de bomberos, centros de salud, etc.).
* RF3. El sistema debe visualizar en un mapa interactivo los nodos registrados y la red vial.
* RF4. El sistema debe calcular la ruta óptima (más corta en tiempo) entre un nodo de respuesta y un nodo de riesgo usando el algoritmo de Bellman-Ford.
* RF5. El sistema debe calcular la ruta más larga factible entre los mismos nodos para análisis comparativo.
* RF6. El sistema debe calcular una asignación de flujo máximo entre múltiples nodos de respuesta y riesgo usando el algoritmo de Edmonds-Karp, retornando los caminos y el número máximo de atenciones posibles.
* RF7. El sistema debe permitir al usuario seleccionar dinámicamente los nodos de interés para ejecutar los algoritmos.
* RF8. El sistema debe mostrar los resultados (rutas, tiempos, distancias, velocidades) de forma visual y textual.

#### **4.3.2 Requisitos No Funcionales**

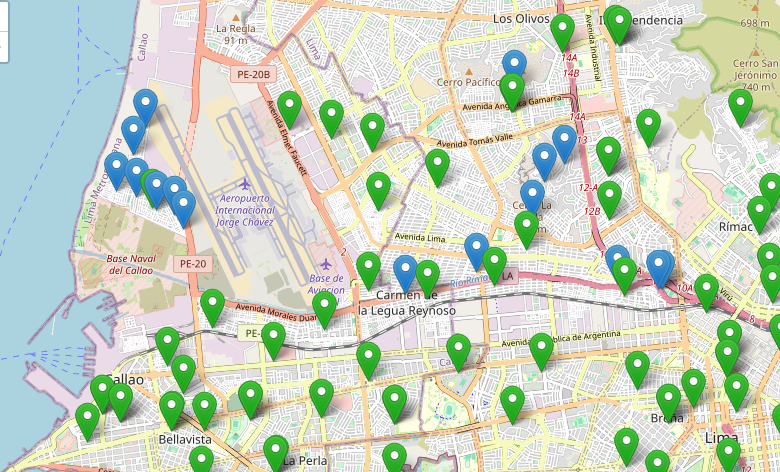
* RNF1. El sistema debe tener tiempos de respuesta aceptables (menos de 3 segundos para rutas individuales y menos de 10 segundos para flujos múltiples en zonas densas).
* RNF2. El backend debe estar desarrollado en FastAPI para garantizar rendimiento, escalabilidad y facilidad de integración.
* RNF3. El frontend debe estar desarrollado en React y ser responsive.
* RNF4. El sistema debe utilizar una base de datos espacial con PostgreSQL + PostGIS para gestionar datos geográficos.
* RNF5. El sistema debe garantizar la precisión geográfica con proyecciones adecuadas (EPSG:4326 / EPSG:32718).
* RNF6. El código debe ser modular, reutilizable y mantener buenas prácticas de programación.
* RNF7. El sistema debe ser portable, permitiendo ejecución local en entornos con recursos moderados.
* RNF8. Debe aplicarse CORS correctamente para habilitar la comunicación entre backend y frontend.

### **4.4 Diseño de la interfaz de usuario:**

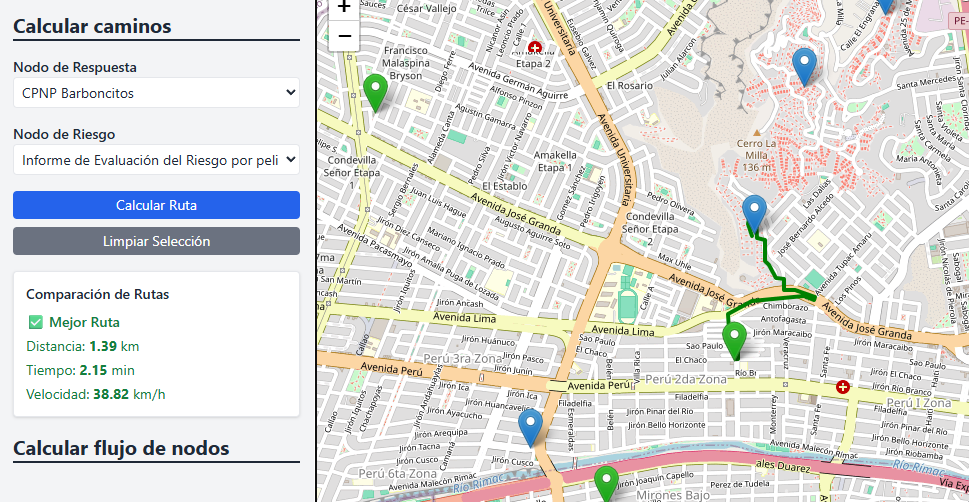
La interfaz de usuario ha sido desarrollada utilizando **React** y la biblioteca **React Leaflet**, lo que permite una visualización geoespacial interactiva e intuitiva. Está orientada a facilitar la toma de decisiones durante situaciones de emergencia, mostrando rutas óptimas, nodos críticos y caminos alternativos sobre la red vial de Lima Metropolitana.

#### **Componentes y funcionalidades clave:**

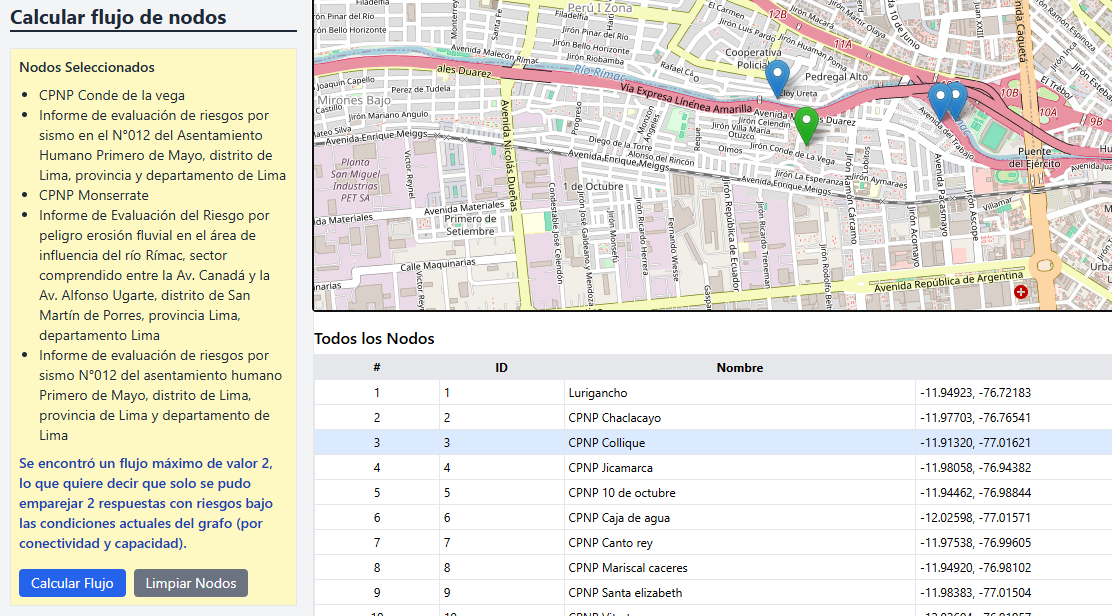
* **Mapa interactivo (Leaflet)**
  + Visualización base: mapa de OpenStreetMap.
  + **Marcadores personalizados**:  
    - **Nodos de riesgo** (icono rojo): zonas vulnerables ante desastres.
    - **Nodos de respuesta** (icono verde): instituciones que atienden emergencias.
  + **Popups contextuales**: al hacer clic en un marcador, se despliega un menú con opciones:  
    - “Usar como nodo de riesgo/respuesta para caminos”: selecciona el nodo para análisis de rutas.
    - “Usar como nodo de riesgo/respuesta para flujo”: agrega el nodo al conjunto para análisis de flujo máximo.



* **Lógica de rutas**
  + Si se selecciona un nodo de respuesta y uno de riesgo, se consulta al backend para:  
    - **Ruta más corta** (algoritmo Bellman-Ford): se traza en **verde sólido**.
    - **Ruta más larga** (peor caso dentro del cutoff): se traza en **rojo punteado**.
  + Ambos caminos se renderizan como Polyline sobre el mapa, mostrando visualmente el trayecto.



* **Gestión de nodos para flujo máximo**
  + El usuario puede marcar múltiples nodos de respuesta y de riesgo usando toggleNodo.
  + Estos nodos se almacenan en el estado global para construir las asignaciones en el backend mediante el algoritmo de **flujo máximo (Edmonds-Karp)**.



#### **Experiencia del usuario**

* **Dinamismo y reactividad**: el mapa se centra automáticamente (RecenterMap) según la selección del usuario.
* **Usabilidad**: botones accesibles, colores diferenciados y estructura clara.
* **Acceso rápido a funciones críticas**: desde el popup del marcador, el usuario puede configurar todo sin navegar entre vistas.

#### **Ventajas de este diseño:**

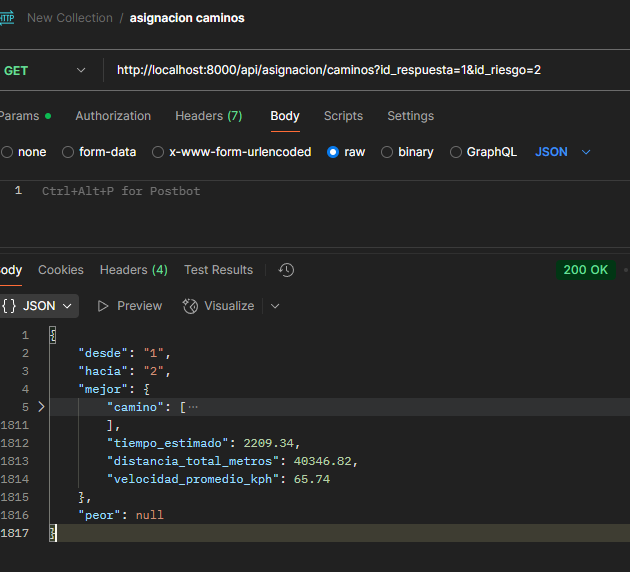
* Permite operar directamente sobre el mapa sin formularios complejos.
* Proporciona **feedback visual inmediato** de las rutas.
* Optimiza la toma de decisiones bajo presión, como en casos de desastres naturales.
* Es escalable para nuevas funcionalidades, como mostrar zonas de congestión o simular rutas bloqueadas.

## 5. Validación de resultados y pruebas

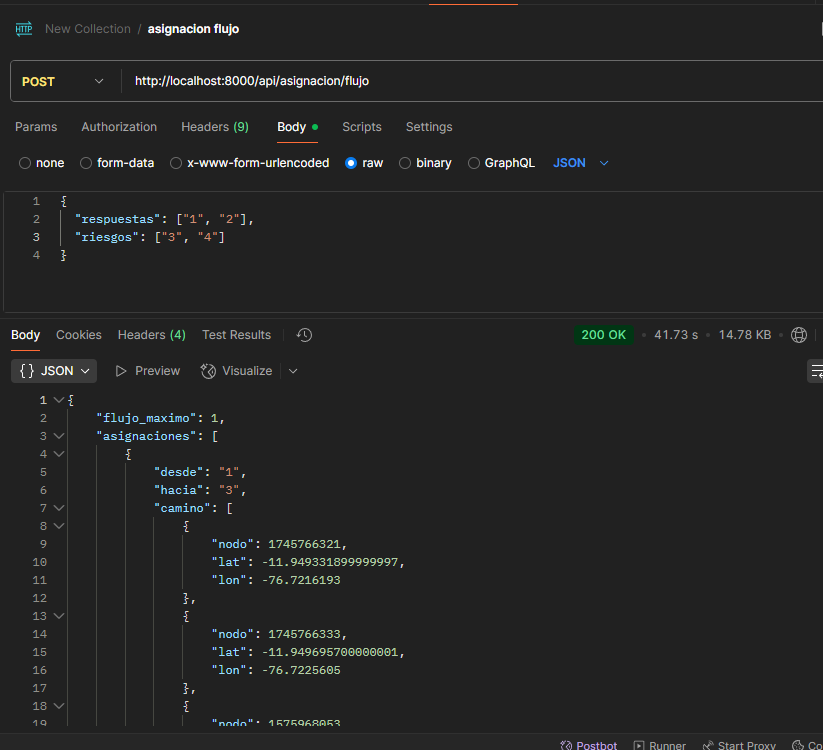
#### **5.1 Entradas del Sistema**

El sistema consume los siguientes tipos de entradas:

* **Puntos georreferenciados**:  
  + Nodos de **riesgo**: zonas vulnerables ante desastres (latitud, longitud, nombre, id).
  + Nodos de **respuesta**: instituciones de atención de emergencias (latitud, longitud, nombre, id).
* **Petición de rutas**:  
  + GET /api/asignacion/caminos?id\_respuesta=X&id\_riesgo=Y
    - Entrada: ID de un nodo de respuesta y uno de riesgo.
    - Salida: camino más corto, camino más largo si existe, tiempo estimado, distancia, velocidad.



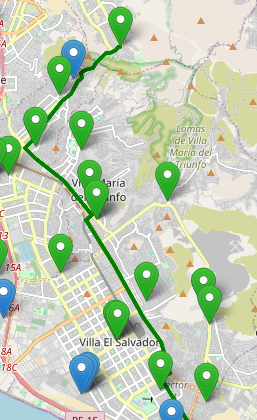
* **Petición de asignación de flujo**:  
  + POST /api/asignacion/flujo  
    - Entrada: JSON con listas de IDs de nodos de respuesta y riesgo.
    - Salida: emparejamientos óptimos según flujo máximo y caminos correspondientes.



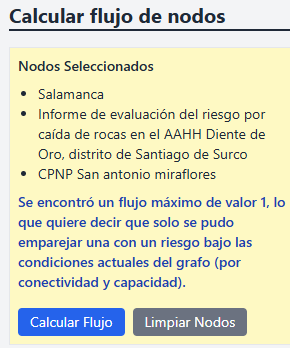
#### **5.2 Salidas Esperadas**

* **Rutas**:  
  + Camino óptimo (verde): obtenido por Bellman-Ford.
  + Camino más lento (rojo): camino más costoso en tiempo (cutoff de profundidad).
  + Datos complementarios: distancia total (m), tiempo estimado (s), velocidad promedio (km/h).





* **Asignaciones de flujo**:  
  + Lista de emparejamientos respuesta-riesgo que maximizan la cobertura institucional.



#### **5.3 Interpretación de Resultados**

* Una **ruta óptima válida** muestra que la red vial está correctamente modelada, con pesos y conexiones coherentes.
* La **diferencia entre camino óptimo y peor caso** permite identificar zonas de alta congestión o falta de conectividad.
* El **flujo máximo** es un indicador directo de la capacidad del sistema de atención ante emergencias dado un escenario de múltiples demandas.
* La **ausencia de caminos** o errores en asignación indican:  
  + Fallos en la conectividad del grafo.
  + Nodos fuera del alcance geográfico (por eso se usa subgrafo con radio).

#### **5.4 Pruebas Realizadas**

| **Prueba** | **Descripción** | **Resultado Esperado** | **Resultado Obtenido** |
| --- | --- | --- | --- |
| Carga de puntos | Se insertaron 10 nodos de riesgo y 10 de respuesta en la BD | Respuesta HTTP 200 y confirmación de carga | Éxito |
| Consulta de ruta óptima | Se seleccionó un nodo de respuesta y riesgo cercanos | Devuelve polyline verde y datos de ruta | Correcto |
| Consulta de ruta peor | Se probaron rutas con muchas bifurcaciones | Devuelve polyline roja con mayor duración | Correcto |
| Asignación de flujo | 5 nodos de respuesta vs 5 de riesgo | Flujo máximo entre 0 y 5 | Asignación según conectividad |
| Subgrafo geográfico | Nodos separados >10 km | El sistema descarta rutas no alcanzables | Optimiza sin errores |
| Nodos inválidos | Se solicitó ruta con un ID inexistente | Error HTTP 404 con mensaje claro | Controlado |

#### **5.5 Herramientas de Prueba**

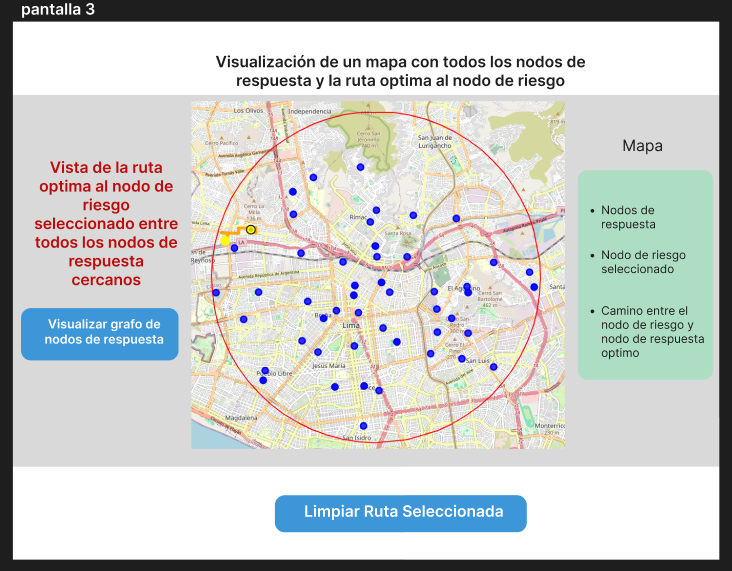
* **Postman** y **cURL**: para pruebas directas de endpoints.
* **Consola del navegador** (React): para verificar correcta visualización de rutas.

# 

# Applications Wireframes

# 

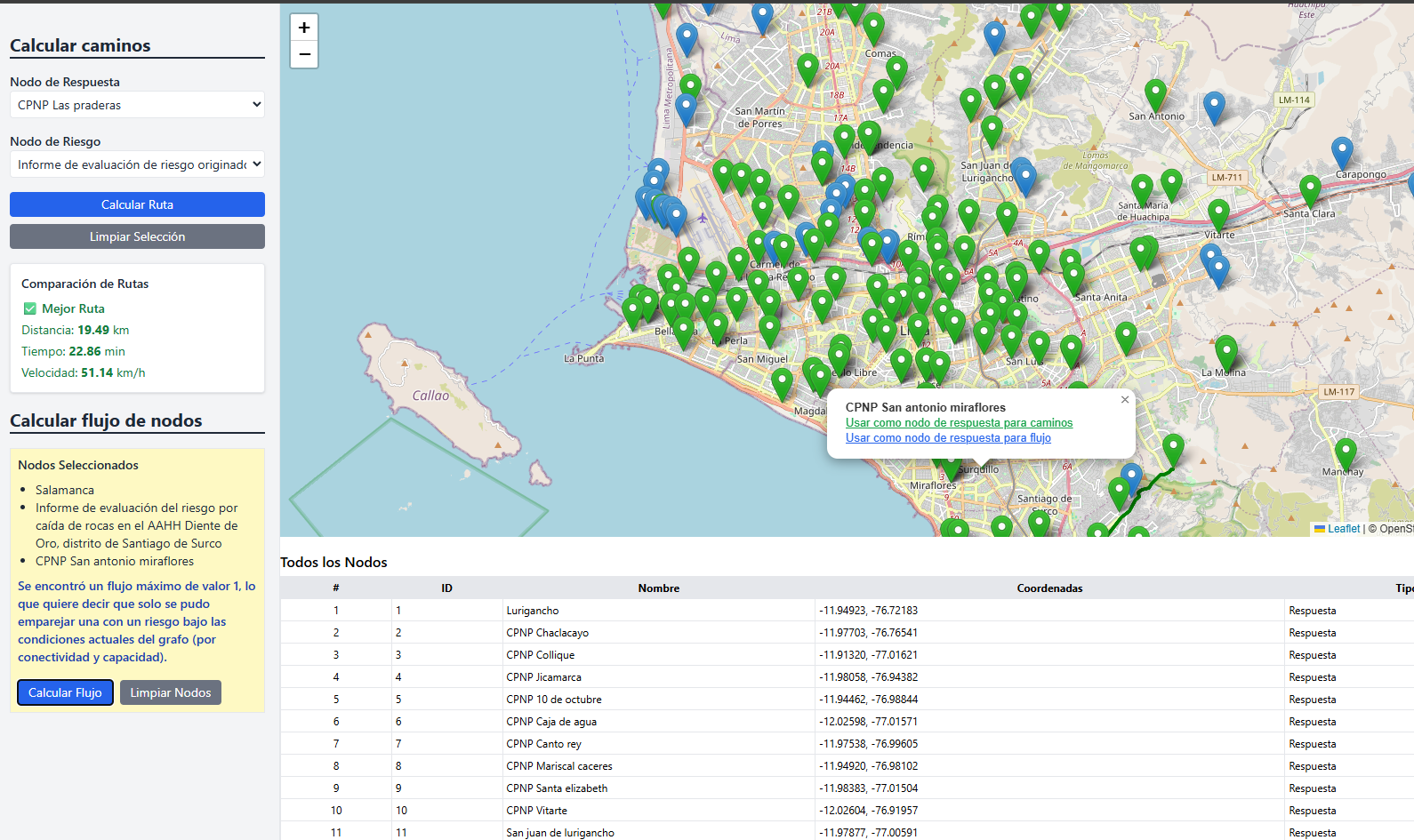




# Applications Mockup

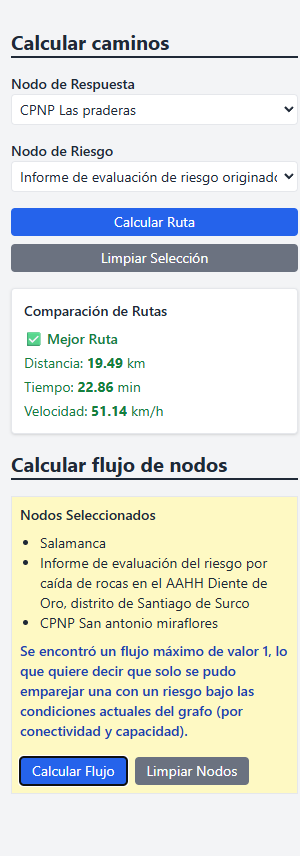
###### Figura 1

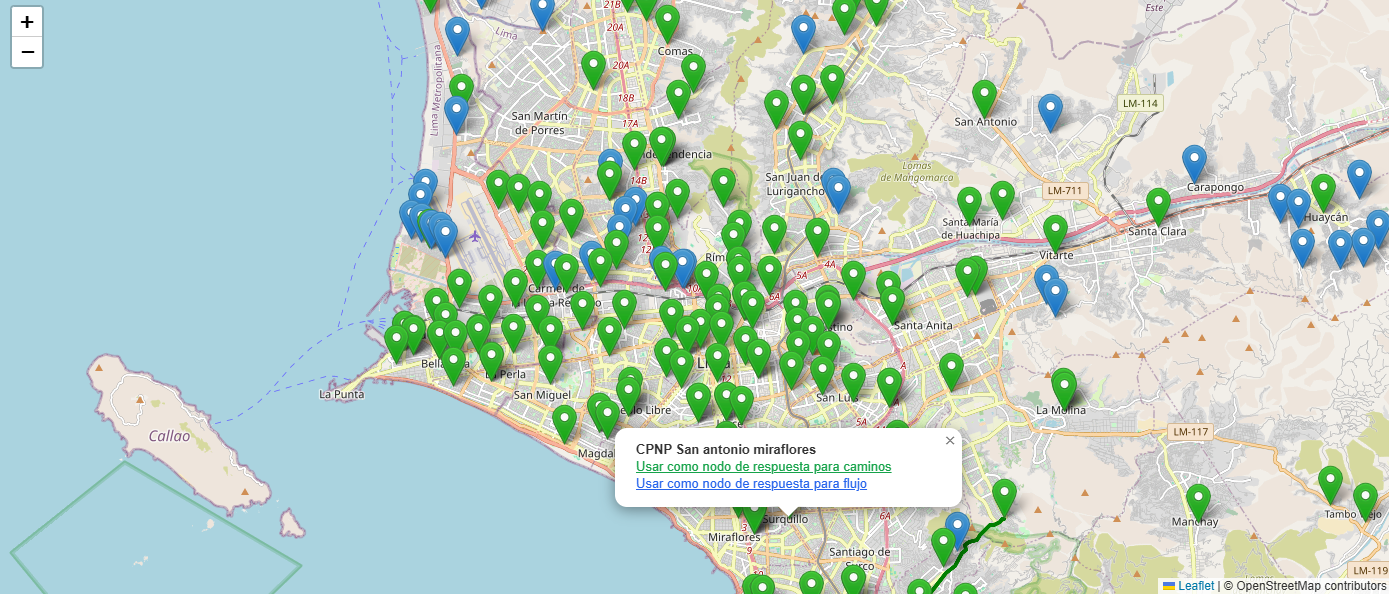
*Pantalla de inicio a la aplicación. Muestra las opciones para ver el mapa principal, búsqueda de riesgo y panel de control*



###### Figura 2

*Panel de control de rutas (información)*





###### Figura 3

*Mapas de Riesgo, Respuesta y caminos. Todos estos sin ninguna configuración inicial.*

**

###### Figura 4

*Búsqueda de caminos, a través del mapa. El usuario puede darle click a cualquier par de nodos para encontrar caminos.*

La interfaz de la aplicación presenta un diseño intuitivo, destacando un mapa interactivo central que permite la optimización de rutas y visualizar puntos de riesgo vinculados al cambio climático en Lima metropolitana. Incluye un panel de configuración con opciones para recalcular rutas, seleccionar kilometraje, medir la distancia, entre otros.

Su diseño responsivo garantiza accesibilidad, empleando una paleta de colores moderna y un estilo minimalista que prioriza la funcionalidad. Adicionalmente, la interfaz muestra la distancia total de las rutas calculadas y cuenta con validaciones que aseguran datos precisos, ofreciendo una solución eficiente para el manejo de rutas ante desastres naturales.

# 6. Conclusiones

* Se logró modelar eficazmente la problemática de la respuesta a desastres en Lima Metropolitana como un grafo ponderado. Este modelado integró de manera exitosa la infraestructura vial, las ubicaciones de las instituciones de emergencia y las zonas de riesgo, lo que permitió una representación abstracta pero precisa del entorno real para el análisis algorítmico.
* La implementación del algoritmo de Dijkstra resultó fundamental para la optimización de rutas, lo que se traduce en una reducción potencial de los tiempos de intervención en escenarios de emergencia. Esta optimización tiene un impacto directo y positivo en la capacidad de respuesta de los equipos de auxilio y en la efectividad de las operaciones ante desastres.
* El uso de datos geoespaciales reales, obtenidos de plataformas como BBBike y SIGRID del CENEPRED, fue crucial para la construcción de un grafo vial robusto y la identificación precisa de puntos críticos y centros de respuesta. Esta integración de datos permitió que los análisis de conectividad, accesibilidad y vulnerabilidad fueran representativos de las condiciones urbanas de Lima.
* El diseño de una interfaz de visualización interactiva, utilizando herramientas como Folium, es un componente clave para facilitar la toma de decisiones. Al presentar las rutas óptimas, las zonas de riesgo y las entidades de respuesta de manera clara y accesible en un mapa, se mejora la comprensión de la situación y la eficiencia en la coordinación de las intervenciones.
* A futuro, se identifica la oportunidad de ampliar la funcionalidad del sistema mediante la incorporación de condiciones dinámicas del tráfico, la priorización de emergencias y la visualización de rutas en tiempo real. Estas mejoras permitirían una adaptación más precisa a los escenarios cambiantes y una optimización continua de la respuesta ante desastres.

# Anexos

Enlace de Figma, en la cual se desarrolló el modelo preliminar de la interfaz esperada a utilizar en el proyecto:

<https://www.figma.com/design/VlbQltpzBq69YzLvmWebm6/Comp.-Algoritmica?node-id=0-1&t=wWMuvVikRxD3bNOi-1>

Enlace a los repositorios de Github, donde se organizó y documentó todo el código del proyecto a través de distintas versiones:

<https://github.com/orgs/GRUPO-8-COMPLEJIDAD-ALGORITMICA/repositories>

# Referencias bibliográficas

Alamoudi, O. & Al-Hashimi, M. (2024). On the Energy Behaviors of the Bellman-Ford and Dijkstra Algorithms: A Detailed Empirical Study. Journal of Sensor and Actuator Networks, 13(5), 67.

<https://www.mdpi.com/2224-2708/13/5/67>

Bonifacio, S., Ruiz, B. & Perez, D. (2025a.). frontend & backend [Repositorio en GitHub]. https://github.com/orgs/GRUPO-8-COMPLEJIDAD-ALGORITMICA/repositories

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introducción a los algoritmos* (3ra ed.). MIT Press.

<https://enos.itcollege.ee/~japoia/algorithms/GT/Introduction_to_algorithms-3rd%20Edition.pdf>

Longley, P. A., Goodchild, M. F., Maguire, D. J., & Rhind, D. W. (2015). *Sistemas de información geográfica y ciencia*. John Wiley & Sons.<https://es.scribd.com/document/433229127/Paul-a-Longley-Michael-F-Goodchild-David-J-Maguire-David-W-Rhind-Geographic-Information-Systems-and-Science-Wiley-2005-1>

Ministerio del Ambiente (MINAM). (2013). Cambio climático y desarrollo sostenible en el Perú. <https://www.minam.gob.pe/cambioclimatico/wp-content/uploads/sites/11/2013/10/CDAM0000323.pdf>

Naciones Unidas. (s.f.). Objetivo 13: Acción por el clima. Recuperado el 9 de mayo de 2025, de <https://www.un.org/sustainabledevelopment/es/climate-change-2/>

Zhu, Y., Li, H., Wang, Z., Li, Q., Dou, Z., Xie, W., Zhang, Z., Wang, R., & Nie, W. (2022). Optimal evacuation route planning of urban personnel at different risk levels of flood disasters based on the improved 3D Dijkstra’s algorithm. Sustainability, 14(16), 10250. <https://doi.org/10.3390/su141610250>