



TRABAJO DE GRADO
Opción Seminario-Diplomado.

Desarrollo de app asistencial con inteligencia artificial para la predicción de zonas de riesgo y atención legal a migrantes
“Alerta Migrante”

Corporación Universitaria Remington.
Facultad de Ingeniería
Ingeniería de Sistemas

Autor: Alejandro David Ojeda Guarnizo

Seminario
2025

Estudiante de Octavo semestre de Ingeniera de sistemas, Uniremington sede Palmira. Email:
alejandro.ojeda.2986@miremington.edu.co

Dedicatoria

Me dedico este trabajo a mí mismo, porque conozco todo lo que me costó llegar hasta aquí. A mis ganas de no rendirme, incluso cuando el camino se volvió difícil. A la disciplina que mantuve, a las noches sin dormir, y a cada pequeño logro que me motivó a seguir.

Agradecimientos

Gracias a todos los que, de una u otra manera, hicieron parte de este camino.

Primeramente a Dios quien ha sido mi motor y fortaleza.

A mis compañeros, por sus ideas, su tiempo, su apoyo y buena energía.

A mi familia, por ser mi refugio y por su amor incondicional

A mis profesores, por su paciencia, dedicación y por enseñarme con pasión y compromiso.

Y, sobre todo, a mí mismo, por nunca rendirme, por seguir creyendo en mis capacidades.

Listado de tabla de Contenido

Resumen	6
Palabras clave	7
Pregunta orientadora	8
Contexto del Problema	8
Estrategias metodológicas para la integración de IA	9
Metodología de búsqueda de la información.....	10
Sustentación teórica de la pregunta.....	12
1. Objetivo general	12
2. Análisis de requerimientos.....	12
2.1. Requerimientos funcionales	12
2.2. Requerimientos no funcionales	13
3. Diseño de la Solución	13
3.1. Arquitectura general	13
3.2. Diseño de la base de datos	14
3.3. Diseño de la interfaz de usuario (UI/UX).....	15
4. Cómo desarrollé mi chatbot con inteligencia artificial.....	20
4.1. ¿Por qué escogí un chatbot con inteligencia artificial?	20
4.2. ¿Con qué herramientas trabajé?	21
4.3. Paso a paso del funcionamiento del bot.....	22
4.4. Parte del código que programé.....	23
4.5. Probando el chatbot desde CMD.....	24
Conclusiones.....	25
Referencias	26

Listado de Tablas

Tabla 1. Estructura lógica de la base de datos	14
--	----

Listados de Figuras

Figura 1: Boceto de la pantalla principal con botón SOS.	15
Figura 2: Diseño del mapa de riesgo con marcadores comunitarios.	16
Figura 3: Interfaz web del chatbot.	17
Figura 4: Diseño de la pantalla de documentos seguros	18
Figura 5: Diseño de la pantalla de Ajustes	19
Figura 6: Página de OpenAI donde cree la cuenta y obtuve la clave API.	20

Figura 7: entorno de Visual Studio Code con los archivos del proyecto abiertos app.py, index.html.	21
Figura 8: interfaz de la app corriendo en el navegador, mostrando un ejemplo de pregunta y respuesta del bot.....	22
Figura 9: Captura del CMD ejecutando el servidor	24

Resumen

En el contexto actual de los Estados Unidos, la comunidad inmigrante enfrenta una creciente vulnerabilidad debido a la intensificación de redadas, el limbo migratorio y la falta de información oportuna y confiable. Este escenario crea una brecha que demanda una respuesta urgente de tecnologías capaces de ofrecer asistencia legal protectora y acción inmediata para emergencias. A la luz de este contexto, nació el proyecto “Alerta Migrante”, una aplicación móvil que busca interceder en nombre de migrantes vulnerables utilizando tecnologías digitales que incluyen inteligencia artificial.

El objetivo principal de este trabajo es contemplar los enfoques estratégicos que presentan desafíos en la construcción de una solución tecnológica socialmente útil que utiliza ingeniería de software y combina IA para proporcionar notificaciones de alerta automáticas, predicción de zonas de riesgo y una interfaz de chatbot de orientación legal. Este problema puede abordarse desde el ámbito de la ingeniería de sistemas para crear un diseño socialmente responsable, éticamente sólido y centrado en el usuario que utilice principios de ingeniería humanitaria.

Se utilizó una metodología cualitativa y aplicada donde se analizan casos, se utilizan herramientas tecnológicas disponibles, se revisan lineamientos éticos respecto al uso de IA, y se realiza construcción progresiva de un prototipo funcional de la aplicación. Durante el desarrollo se consideró tanto el enfoque técnico como el impacto social del proyecto, sus posibilidades y su pertinencia, a lo largo de su desarrollo.

Así, se integró una solución con dos componentes fundamentales: un sistema de predicción de zonas de peligro que se alimenta de reportes ciudadanos y un chatbot legal inteligente migratorio con la capacidad de atender consultas de forma inmediata. Constituye una herramienta de impacto práctico, académico y humanitario que propicia una innovación social por la unión de tecnología y empatía.

Palabras clave: Asistencia migratoria, inteligencia artificial, ingeniería de sistemas, innovación social, tecnología humanitaria

Pregunta orientadora

La integración de sistemas de transformación digital e inteligencia artificial en el contexto de vulnerabilidad social ha mostrado una gran promesa para ofrecer soluciones rápidas y personalizadas. Por ejemplo, en el caso de los migrantes indocumentados en Estados Unidos, las redadas impredecibles, el proceso de detención y la completa falta de información creíble crean escenarios que, en la realidad, son amenazantes para la vida y provocan ansiedad. Esta realidad obliga a explorar enfoques de desarrollo de software que no solo sean técnicamente sólidos, sino también éticos y centrados en el usuario.

La aplicación particular de herramientas de software para sistematizar las brechas que crean las fronteras alrededor de las vidas humanas, Alerta Migrante, fue creada para abordar esta urgente necesidad. Esta aplicación combina ingeniería de sistemas, diseño humanitario e inteligencia artificial para proporcionar tres servicios cruciales: notificaciones automáticas en emergencias, predicción de áreas de alto riesgo utilizando datos generados por ciudadanos, y navegación legal en tiempo real facilitada por chatbots inteligentes para migrantes.

A partir de estas suposiciones, es necesario avanzar en las metodologías de desarrollo que aseguren y empleen adecuadamente estas tecnologías, al mismo tiempo que se garantiza la protección de datos y la usabilidad en situaciones críticas.

Contexto del Problema

Las políticas migratorias son cada vez más hostiles hacia las personas migrantes y su posibilidad de acceso a información legal confiable. Esto, junto a la falta de recursos legales y de información veraz acerca de sus derechos, aumenta la vulnerabilidad, el abuso y la re victimización que enfrentan al ser separados de familiares. Sin herramientas adecuadas, muchas víctimas de abusos o separaciones familiares no cuentan con un canal

inmediato a la ayuda o guía de sus derechos, al canal que contempla una respuesta formal.

El diseño de una aplicación que incorpora tecnología IA conlleva un desafío importante: pensar éticamente desde la recolección geoespacial de datos, su manejo, hasta la construcción de modelos robustos a desarrollar, como el de lenguaje natural y por supuesto, la predicción en teléfonos inteligentes de bajo consumo de recursos.

Estrategias metodológicas para la integración de IA

Durante la revisión de literatura académica y de casos de éxito en asistencia humanitaria, así como en informes de ACNUR, surgen estrategias que constituyen el centro del trabajo a abordar (Russell & Norvig, 2021):

- 1) Permitir la validación temprana del chatbot y el modelo predictivo con usuarios reales mediante el desarrollo iterativo de prototipos.
- 2) El sistema modular permite que cada componente de IA (predicción, chat, notificaciones) se escale y actualice de manera independiente y autónoma.
- 3) Garantizar rendimiento, accesibilidad y confidencialidad de datos a través de pruebas de usabilidad y seguridad continuas.

Con base en estas estrategias, la pregunta orientadora es:

¿Cuáles son las metodologías más efectivas para desarrollar e integrar soluciones de inteligencia artificial en la app Alerta Migrante, de modo que ofrezca asistencia inmediata, predicción de zonas de riesgo y orientación legal segura para personas migrantes?

Metodología de búsqueda de la información

Para responder a la pregunta orientadora sobre las metodologías más efectivas para incorporar inteligencia artificial en la aplicación Alerta Migrante, diseñé una estrategia para la adquisición de información científica, técnica y de campo. El proceso que seguí incluyó cuatro pasos principales:

Selección de palabras clave y operadores booleanos

Partí de términos vinculados al proyecto como: “inteligencia artificial”, “asistencia migratoria”, “chatbot legal”, “predicción de riesgo” y “transformación digital”. Estos conceptos se combinaron con operadores booleanos (AND, OR) para generar consultas como:

- “inteligencia artificial AND asistencia migratoria”
- “chatbot legal OR orientación legal AND migrantes”
- “predicción de zonas de riesgo AND aplicaciones móviles”

Fuentes consultadas y motores de búsqueda

De las bases de datos académicas y de los repositorios de referencia en ingeniería de sistemas y ciencia de datos, prometí usar:

- IEEE Xplore: para artículos sobre algoritmos de IA y sistemas embebidos.
- Scopus y Web of Science: para estudios multidisciplinares de tecnología y derechos humanos.
- Google Scholar: para acceder a tesis, actas de conferencias y literatura gris relacionada con chatbots legales y asistencia social.

Además, consulté informes de ACNUR y ONU para contextualizar aplicaciones móviles en crisis humanitarias.

Definición de criterios de selección y filtrado.

Para filtrar los resultados, en primera instancia, revise las publicaciones relevantes que al menos cumplan con dos criterios:

- 1) Publicación en los últimos cinco años para relevancia actual.
- 2) Presentación de una clara y reproducible metodología para evaluar el rigor científico.
- 3) Enfoque a movilidades o entornos de asistencia social.
- 4) Ética, cuyo impacto es que protejan datos y privacidad.

Análisis y síntesis de la información.

Leí resúmenes y conclusiones para comprobar la veracidad de la relevancia por cada fuente. Posteriormente, realicé la recolección y categorización de conceptos tales como: arquitecturas de chatbot, metodologías de validación de prototipos, y estrategias de recolección de datos ciudadanos, para insertarlas en un cuadro comparativo. Con esta síntesis, el fundamento de parte teórico y práctico que establecí para el diseño y desarrollo de Alerta Migrante, fue que la implementación de IA exigía solvencia técnica, seguridad, y responsabilidad social.

Sustentación teórica de la pregunta

1. Objetivo general

Desarrollar una aplicación móvil que combine ingeniería del software y inteligencia artificial con el objetivo de brindar seguridad y auxilio en tiempo real a migrantes en EEUU.

2. Análisis de requerimientos

Como estudiante de Ingeniería de Sistemas con interés en temas migratorios, realicé mis requerimientos de forma **autodidacta**, en base a:

- **Revisión de experiencias compartidas en redes sociales** de migrantes, donde describían sus miedos y necesidades.
- **Consultas a expertos** y chats profesionales, para validar conceptos básicos de asistencia legal y seguridad.
- **Análisis de estudios en línea** (ONG, blogs especializados), para entender los procesos migratorios más comunes.

2.1. Requerimientos funcionales

1. **Botón SOS con envío instantáneo:** Al pulsar, la app debe tomar la ubicación actual y notificar a contactos de emergencia y a un abogado de turno.
2. **Mapa de riesgo comunitario:** Mostrar, en tiempo real, las zonas de actividad migratoria intensa o redadas, basado en reportes agregados de usuarios.
3. **Chatbot legal inteligente:** Un asistente virtual web que, mediante IA, responda dudas habituales sobre derechos, vías de apelación y contactos consulares.
4. **Directorio dinámico:** Listado filtrable de consulados y abogados cercanos, con distintas opciones.
5. **Bóveda de documentos segura:** Espacio cifrado para guardar pasaportes o visas, accesible con huella o PIN

2.2.Requerimientos no funcionales

1. **Seguridad de datos:** Cifrado AES 256 en reposo en el dispositivo, TLS 1.3 para todas las comunicaciones. La autenticación se realiza utilizando biometría o un PIN.
2. **Rendimiento móvil:** La acción SOS debe completarse en menos de 2 segundos, el mapa debe cargarse en menos de 5 segundos incluso con datos variables.
3. **Alta disponibilidad:** Basado en la nube con arquitectura de escalado automático y tolerancia a fallos para garantizar alta disponibilidad y servicio continuo.
4. **Usabilidad excepcional:** Diseño minimalista, botones grandes y acceso rápido bajo estrés a través de lectores de pantalla y comandos de voz.
5. **Portabilidad y copias offline:** Aplicación nativa de Flutter para dispositivos iOS y Android, funcionalidad con capacidad limitada offline.
6. **Mantenibilidad:** CI/CD junto con un código modular bien definido y una documentación amplia facilitan cambios futuros y pruebas automatizadas.

3. Diseño de la Solución

Con los requerimientos claros, el siguiente paso fue plasmar cómo funcionaría *Alerta Migrante* en la práctica. Mi enfoque fue en que cada elemento de la app debe responder a una necesidad real y encajar en un flujo ágil de uso.

3.1.Arquitectura general

Opté por un **modelo de microservicios** que divide la aplicación en piezas independientes:

- **Servicio SOS:** Atiende la pulsación del botón, captura la ubicación y dispara notificaciones.
- **Servicio de Mapas:** Gestiona los reportes ciudadanos y entrega la capa de riesgo para el mapa.

- **Chatbot:** Alojado en un contenedor separado, procesa consultas en lenguaje natural y devuelve respuestas basadas en IA.
- **Almacenamiento de Documentos:** Un microservicio dedicado al cifrado y acceso a pasaportes o visas.
- **API Gateway:** Punto único de entrada, que enruta llamadas y maneja autenticación.
- **Base de datos relacional:** PostgreSQL para datos de usuarios y alertas

3.2. Diseño de la base de datos

La estructura de datos es clave para que las consultas sean **rápidas** y la información siempre **consistente**. Así quedó el modelo entidad-relación:

Entidad	Campos principales	Relación
Usuarios	id_usuario (PK), nombre, email, rol	1-* Alertas, 1-* Reportes, 1-* Documentos
Alertas	id_alerta (PK), id_usuario (FK), gps, estado	1 alerta genera n notificaciones
Reportes	id_reporte (PK), id_usuario (FK), gps, tipo	Datos para mapa de riesgo
Mensajes	id_mensaje (PK), id_emisor, id_receptor, texto	Chatbot y chat entre usuarios
Documentos	id_doc (PK), id_usuario (FK), ruta, tipo	Bóveda cifrada por usuario

Tabla 1. Estructura lógica de la base de datos

Esta tabla muestra cómo están organizados los datos en el sistema. Cada parte representa una función importante.

3.3. Diseño de la interfaz de usuario (UI/UX)

1. Pantalla principal:

- Gran botón SOS en rojo, siempre fijo en la esquina inferior.
- Barra superior con acceso al mapa y al chatbot



Figura 1: Boceto de la pantalla principal con botón SOS.

2. Mapa de riesgo:

- Iconos de reportes recientes que muestran hora y tipo al pulsar.



Figura 2: Diseño del mapa de riesgo con marcadores comunitarios.

3. ChatBot:

- Ventana web con estilo de mensajería común.
- Campo de texto y sugerencias automáticas.

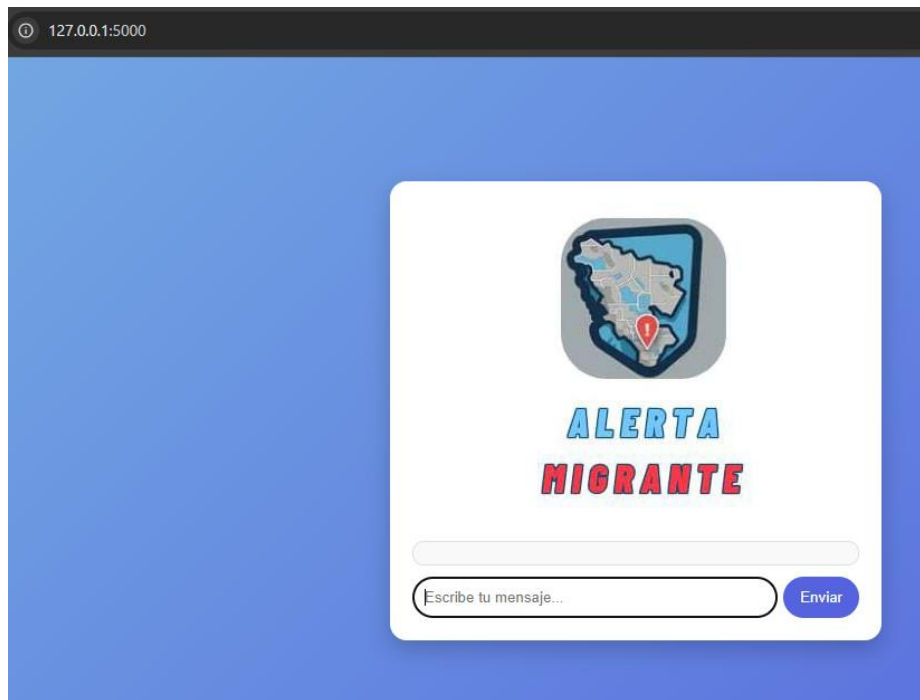



Figura 3: Interfaz web del chatbot.

4. Bóveda de documentos:

- Iconos de pasaporte, visa, residencias.
- Acceso protegido con huella digital o PIN.

Creá una cuenta nueva

¿Ya te registraste? Iniciá sesión acá.




NOMBRE FOTO

Daniel Martínez

NÚMERO DE IDENTIFICACIÓN

PASAPORTE



FECHA DE NACIMIENTO

12/08/1997

CORREO

Daniel030@gmail.com

Continuar

Creá una cuenta nueva

¿Ya te registraste? Iniciá sesión acá.

IDIOMA

Español

PAIS

Colombia

ESTATUS MIGRATORIO

INDOCUMENTADO

UBICACIÓN

ACTIVADA

ESTADO

NEW YORK

Registrarme

Figura 4: Diseño de la pantalla de documentos seguros

5. Módulo de ajustes:

- Configuración personalizada.

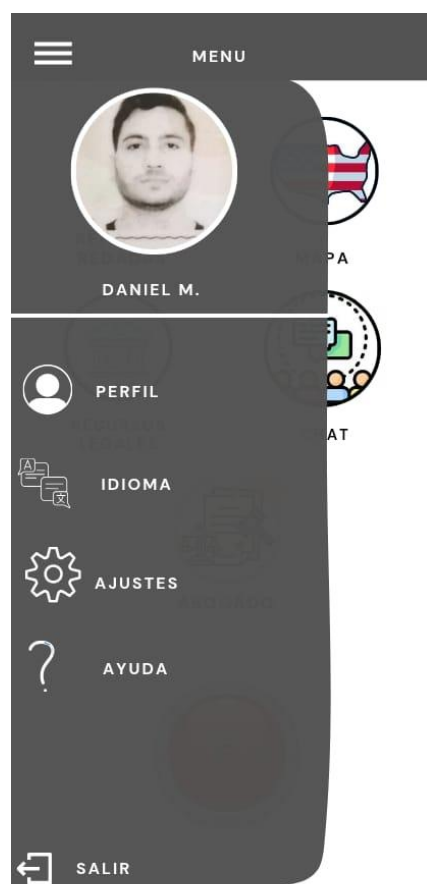


Figura 5: Diseño de la pantalla de Ajustes

4. Cómo desarrollé mi chatbot con inteligencia artificial

Para mi proyecto, quise hacer algo que ayudara a la comunidad. Como ya tenía una app llamada **Alerta Migrante**, le integre un chatbot que sirviera como guía básica para personas que están pasando por procesos migratorios en Estados Unidos. La idea fue crear un bot que responda dudas comunes de forma clara, sencilla y al instante.

4.1. ¿Por qué escogí un chatbot con inteligencia artificial?

Al principio pensé en hacer un bot sencillo, con respuestas programadas, pero luego me di cuenta que no era suficiente. Las personas migrantes tienen situaciones muy diferentes entre sí, y es imposible anticipar todas las preguntas que pueden tener.

Por eso, me incliné por usar inteligencia artificial, específicamente un modelo que entienda **lenguaje natural** y pueda responder como si fuera una persona real. Después de investigar un poco, decidí usar la **API de OpenAI**, que permite conectarse con su modelo ChatGPT y obtener respuestas bien estructuradas.

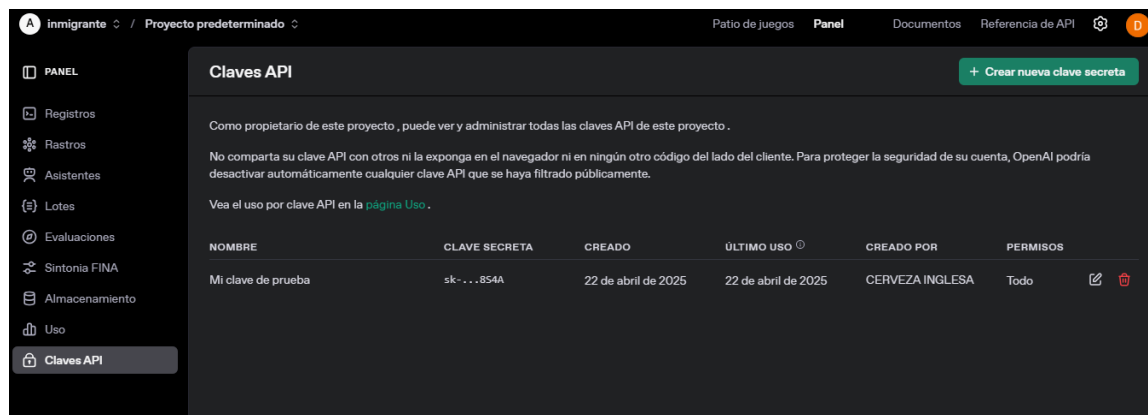


Figura 6: Página de OpenAI donde cree la cuenta y obtuve la clave API.

4.2. ¿Con qué herramientas trabajé?

Elegí usar **Python** porque es un lenguaje que ya manejo y es ideal para trabajar con IA. Usé **Visual Studio Code** como editor, porque es liviano y fácil de usar.

Para conectarme con la IA, utilicé un microservicio que hice con **FastAPI**, es una librería de Python muy útil para crear APIs rápidas y modernas.

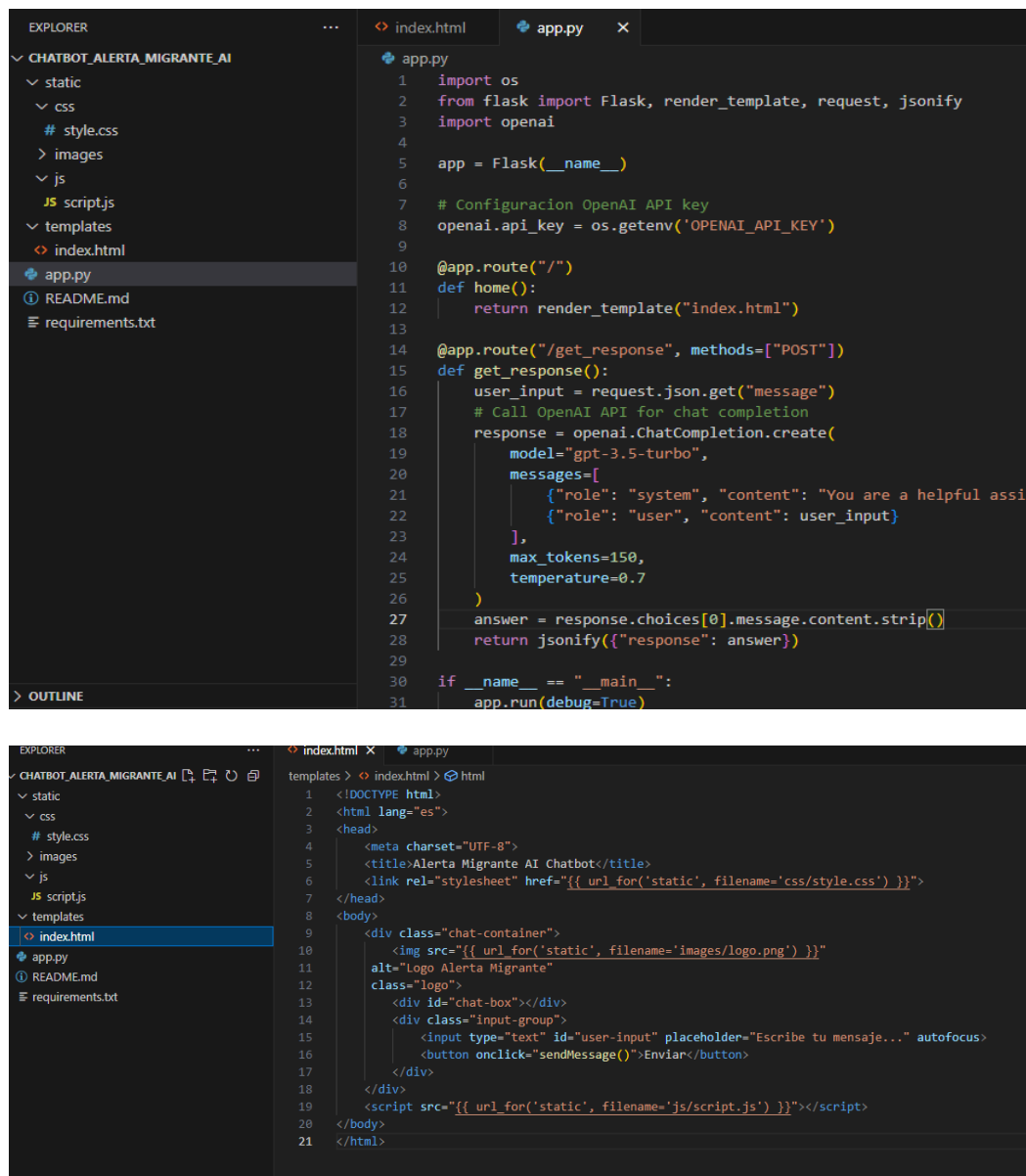


Figura 7: entorno de Visual Studio Code con los archivos del proyecto abiertos app.py, index.html.

4.3.Paso a paso del funcionamiento del bot

Así es como funciona mi chatbot por dentro:

1. El usuario escribe una pregunta en la app (por ejemplo: “¿Qué hago si ICE me detiene?”).
2. Esa pregunta se envía al backend.
3. El backend se conecta con OpenAI usando FastAPI y le pasa la pregunta al modelo.
4. El modelo genera una respuesta y esta aparece de nuevo en la pantalla del usuario.



Figura 8: interfaz de la app corriendo en el navegador, mostrando un ejemplo de pregunta y respuesta del bot.

4.4. Parte del código que programé

Aquí muestro una parte del código que usé para conectar con OpenAI y recibir las respuestas:

```
from fastapi import FastAPI, Request

import openai

app = FastAPI()

openai.api_key = "sk...."

@app.post("/chatbot")

async def responder(request: Request):

    data = await request.json()

    pregunta = data.get("mensaje")

    respuesta = openai.ChatCompletion.create(

        model="gpt-3.5-turbo",

        messages=[

            {"role": "system", "content": "Eres un asistente legal para migrantes. Responde de forma clara, humana y útil."},

            {"role": "user", "content": pregunta}

        ]

    )

    return {"respuesta": respuesta["choices"][0]["message"]["content"]}
```

4.5. Probando el chatbot desde CMD

Después de tener el código listo, fui a la terminal (cmd) y ejecuté el servidor con el comando:

```
C:\Users\USUARIO\Desktop\chatbot_alerta_migrante_ai>python app.py
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 140-991-038
```

Figura 9: Captura del CMD ejecutando el servidor

Conclusiones

Completar este proyecto me hizo darme cuenta de que, cuando se aplica con un enfoque centrado en el ser humano, la inteligencia artificial puede ser una gran ayuda para apoyar a comunidades vulnerables como los migrantes. A través del desarrollo e implementación de un chatbot legal en la aplicación móvil Alerta Migrante, pude crear una tecnología funcional, segura y accesible que proporcionaba respuestas legales básicas a preguntas de manera empática, de forma automática y en tiempo real. Este trabajo demostró que incluso como estudiante, es posible construir herramientas con impacto social de manera digital, si existen metodologías claras, diseño centrado en el usuario y una arquitectura modular que permita el crecimiento y la mejora con el tiempo. La integración de la IA en esta aplicación no solo respondía a un requisito específico, sino que reafirmó la capacidad de la tecnología para servir como un vínculo entre el conocimiento y quienes más lo necesitan.

Referencias

- Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson Education.
- Pérez, A., & Gómez, M. (2020). Implementación de chatbots legales mediante procesamiento de lenguaje natural. *Revista de Ingeniería y Tecnología*, 18(2), 45–52.
- Norman, D. A. (2013). *The design of everyday things* (Revised and expanded edition). Basic Books.
- Chen, L., Xu, Y., & Wang, D. (2019). Risk prediction using machine learning in real-time applications. *International Journal of Data Science*, 5(1), 67–78.
- ACNUR. (2022). *Tecnología e innovación para la protección de personas refugiadas*. <https://www.acnur.org/>
- ONU. (2021). *Informe sobre migración internacional y desarrollo sostenible*. <https://www.un.org/>
- Pressman, R. S. (2014). *Ingeniería del software: Un enfoque práctico* (7a ed.). McGraw-Hill Interamericana.