

Diseño y Desarrollo de un Sistema Generativo de Optimización de Hojas de Vida basado  
en Inteligencia Artificial

Julian Eduardo Ospina Romero

Director

Dario Jose Delgado Quintero

Universidad Nacional Abierta y a Distancia UNAD  
Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI  
Ingeniería de Sistemas

2026

Nota de Aceptación

---

Dario Jose Delgado Quintero

---

Jurado

---

Jurado

## **Dedicatoria**

Dedico este trabajo a mi familia, cuyo respaldo constante y sus valores inculcados han sido fundamentales para mi formación personal y profesional.

Al director de tesis, Dr. Darío José Delgado Quintero, por su guía, compromiso y orientación académica, que enriquecieron significativamente el desarrollo de este proyecto.

A la Universidad Nacional Abierta y a Distancia – UNAD, por ofrecer el espacio y los recursos que hicieron posible la culminación de este proceso académico.

## Resumen

En el contexto laboral contemporáneo, la hoja de vida constituye la principal carta de presentación profesional. Este proyecto desarrolló un sistema generativo basado en inteligencia artificial y en procesamiento del lenguaje natural para optimizar las hojas de vida, mejorando su relevancia, claridad y concisión. La metodología empleada se organizó en cuatro fases principales: Delimitación del Alcance, Selección del Modelo (optando por GPT-4 de Azure OpenAI), Adaptación y Alineación mediante el desarrollo iterativo de prompts, e Integración en la Aplicación mediante una arquitectura web escalable. Los resultados demostraron mejoras significativas: la compatibilidad promedio con sistemas ATS aumentó del 48% al 83%, lo que redujo drásticamente los errores ortográficos y alcanzó un alto nivel de satisfacción de usuario (4.7/5). Se validó el sistema con 30 hojas de vida de perfiles de TI, lo que evidencia la eficacia de una estrategia de ingeniería de prompts especializada frente a enfoques monolíticos.

En conclusión, la integración de IA generativa mejora sustancialmente la empleabilidad mediante la optimización personalizada de currículos, cumpliendo con estrictos estándares de privacidad. El sistema supera a las herramientas tradicionales al ofrecer recomendaciones contextualizadas y establece un nuevo estándar en asistencia inteligente para la inserción laboral.

***Palabras clave:*** inteligencia artificial; procesamiento de lenguaje natural; hoja de vida; optimización; empleabilidad; seguridad de la información; interfaz de usuario.

## Abstract

In the contemporary labor context, the résumé constitutes the primary professional introduction. This project developed a generative system based on artificial intelligence and natural language processing to optimize résumés, enhancing their relevance, clarity, and conciseness. The methodology was structured into four main phases: Scope Definition, Model Selection (choosing Azure OpenAI's GPT-4), Adaptation and Alignment through iterative prompt development, and Application Integration through a scalable web architecture. The results demonstrated significant improvements: average compatibility with ATS systems increased from 48% to 83%, drastically reducing spelling errors and achieving a high level of user satisfaction (4.7/5). The system was validated using 30 résumés from IT profiles, highlighting the effectiveness of a specialized prompt engineering strategy compared to monolithic approaches. In conclusion, the integration of generative AI substantially enhances employability through personalized résumé optimization while complying with strict privacy standards. The system outperforms traditional tools by offering contextualized recommendations and establishes a new standard in intelligent assistance for job placement.

**Keywords:** artificial intelligence; natural language processing; résumé; optimization; employability; information security; user interface.

## Tabla de Contenido

Introducción .....	12
Justificación .....	14
Planteamiento del problema.....	17
Problema central (usuario final) .....	18
Núcleo de ingeniería (problema técnico).....	18
Restricciones clave. ....	19
Riesgos y salvaguardas. ....	19
Criterios de éxito (medibles).....	20
Síntesis del problema.....	20
Objetivos .....	21
Objetivo General.....	21
Objetivos Específicos .....	21
Marco conceptual y teórico.....	22
Avances recientes en IA generativa.....	22
Desafíos técnicos y éticos de la IA generativa .....	23
Evaluación de modelos generativos .....	24
IA generativa en el ámbito de la empleabilidad .....	25
Aplicaciones Existentes y Estado del Arte .....	27
Glosario de Términos Técnicos .....	28
Plataforma de Trabajo .....	30
Resumen del marco teórico .....	31

Metodología .....	32
Justificación de la Metodología .....	32
Fase 1: Scope (Delimitación del Alcance) .....	33
Fase 2: Select (Selección del Modelo) .....	35
Fase 3: Adapt & Align (Adaptación y Alineación del Modelo) .....	38
Fase 4: Application Integration (Integración en la Aplicación) .....	42
Arquitectura y componentes .....	48
Componentes principales.....	48
Proceso Completo del Sistema .....	49
Diagrama de Flujo de Datos .....	51
Resultados .....	53
Desarrollo del Sistema y Funcionalidades.....	53
Visión general de la solución.....	53
Componentes y responsabilidades (resumen .....	54
Contratos de la API.....	56
Reglas de formato y validador ATS .....	58
Seguridad, privacidad y cumplimiento .....	58
Telemetría y observabilidad.....	59
Evaluación de Desempeño e Impacto.....	59
Protocolo de evaluación.....	59
Resultados cuantitativos .....	60
A)    Compatibilidad ATS (pre vs post) .....	60
B)    Comparación con líneas base.....	60

C) Calidad de redacción y forma (rúbrica de expertos).....	61
D) Uso, adopción y eficiencia.....	62
E) Requisitos no funcionales .....	62
Resultados cualitativos (síntesis).....	63
Discusión y lectura de resultados.....	63
Amenazas a la validez y mitigaciones .....	64
Conclusiones operativas .....	64
Conclusiones.....	65
Comparación con Herramientas Existentes.....	67
Métricas y Resultados Detallados.....	68
Recomendaciones .....	70
Referencias bibliográficas.....	74
Anexos .....	82
Anexo A: Prompts Utilizados.....	82
Anexo B: Herramientas y Tecnologías Utilizadas .....	91
ANEXO B: HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS.....	91
B.1. Stack Tecnológico Principal.....	91
B.2. Herramientas de Desarrollo.....	93
B.3. Herramientas de Documentación .....	94
B.4. Servicios y APIs Externas .....	94
B.5. Configuración de Seguridad.....	95
B.6. Métricas y Observabilidad .....	95
B.7. Costos Estimados (Escenario Piloto) .....	96

B.8. Requisitos del Sistema .....	96
B.9. Dependencias del Proyecto .....	97
B.10. Referencias y Documentación.....	98
Anexo C: Pantallazos del Sistema .....	98
ANEXO C: PANTALLAZOS DEL SISTEMA FUNCIONANDO.....	98
C.1. Página de Inicio (Home) .....	99
C.2. Formulario de Carga de CV (Upload Form) .....	100
C.3. Indicador de Progreso (Progress Indicator).....	102
C.4. Dashboard de Sugerencias (Suggestions Dashboard) .....	104
C.5. Encuesta de Satisfacción (Feedback Form) .....	107
C.9. Notas Técnicas sobre la Interfaz .....	108
C.10. Ejemplo de Datos Mostrados .....	109

## Lista de Figuras

Figura 1 Ciclo de vida del proyecto de IA generativa .....	32
Figura 2 Diagrama de arquitectura .....	48
Figura 3 Pipeline de optimización de CV .....	50
Figura 4 Diagrama de flujo de datos.....	51
Figura 5 Diagrama de secuencia de componentes .....	52
Figura 6 Costos estimados .....	99
Figura 7 Formulario de Carga de CV .....	100
Figura 8 Indicador de Progreso.....	102
Figura 9 Dashboard de Sugerencias.....	104
Figura 10 Encuesta de satisfacción .....	107

### Lista de Tablas

Tabla 1 Componentes principales y responsabilidades .....	54
Tabla 2 Endpoints centrales y contratos .....	56
Tabla 3 Reglas ATS aplicadas .....	58
Tabla 4 Media pre 66,6; media post 85,1; $\Delta$ medio = +18,5 puntos. Umbral de compatibilidad.	60
Tabla 5 Comparadores (medias ATS por condición) .....	61
Tabla 6 Rúbrica de expertos (medias por criterio, escala 1–5).....	61
Tabla 7 Métricas operativas .....	62
Tabla 8 Página de inicio.....	90
Tabla 9 Costos estimados.....	96

## Introducción

En el competitivo mercado laboral actual, el currículum vitae (CV) se ha convertido en la principal carta de presentación de los profesionales. Sin embargo, muchos CV quedan descartados antes siquiera de llegar a un reclutador humano, debido al uso generalizado de sistemas automatizados de selección de candidatos. Se estima que más del 99 % de las empresas Fortune 500 emplean un Applicant Tracking System (ATS), y que cerca del 75 % de los currículos son rechazados automáticamente si no contienen las palabras clave o el formato esperados (Van Inwegen et al., 2023). Además, los reclutadores suelen dedicar apenas 6–7 segundos a revisar cada CV inicial (InterviewPal, 2025; Ladders, 2018). Estas cifras resaltan dos realidades: por un lado, un CV no optimizado puede perderse en el proceso automatizado ("agujero negro digital"), y por otro, la claridad y relevancia del contenido son críticas para captar la atención en ese breve vistazo inicial.

Al mismo tiempo, estudios recientes evidencian el impacto positivo de mejorar la redacción de los CV con ayuda de IA. Por ejemplo, un experimento con ~480,000 solicitantes demostró que quienes recibieron asistencia algorítmica al pulir su CV aumentaron 8 % la probabilidad de ser contratados y obtuvieron 7.8 % más ofertas de trabajo que el grupo de control sin ayuda (Van Inwegen et al., 2023). Esto sugiere que un CV bien redactado y optimizado no solo supera mejor los filtros automáticos, sino que también proyecta de manera más efectiva las habilidades reales del candidato ante los empleadores.

En respuesta a estos retos, este proyecto propone desarrollar un sistema inteligente basado en IA generativa y Procesamiento de Lenguaje Natural (PLN) que "afine" las hojas de vida de profesionales TI de acuerdo con ofertas laborales específicas. El sistema analizará el CV del usuario y la descripción del puesto objetivo para generar sugerencias puntuales en cuatro

dimensiones clave: contenido, palabras clave, estilo/redacción y formato. De este modo, ayudará al candidato a "hablar el mismo idioma" que los sistemas automatizados y los reclutadores humanos, destacando los aspectos más relevantes de su perfil.

## Justificación

El proyecto es pertinente en lo práctico, académico y social porque ataca un problema real de empleabilidad y eficiencia en la preselección de talento. En la práctica, la gran mayoría de grandes empleadores filtran candidaturas con sistemas de seguimiento de postulantes (ATS): en 2025 se detectó un ATS en 489 de 500 compañías Fortune 500 (97,8%), lo que confirma su ubicuidad en procesos de gran escala (Jobscan, 2025). Asimismo, la revisión inicial de un CV por parte de reclutadores suele ser muy breve ( $\approx 6-7,4$  segundos), por lo que cualquier desalineación entre CV y oferta eleva el riesgo de descarte temprano (HR Dive, 2018; Ladders, 2018). Adicionalmente, hay evidencia empírica de que la asistencia algorítmica en redacción mejora resultados: solicitantes con apoyo de IA en su CV obtuvieron 7,8% más ofertas en un experimento a gran escala (Van Inwegen et al., 2023). Esto refuerza el potencial de herramientas que ajustan contenido, palabras clave y formato compatibles con ATS para aumentar la empleabilidad individual y reducir fricción en la preselección (Blumberg, 2023).

Desde una perspectiva de equidad y reducción de brechas, la solución prioriza a: (1) egresados y personas en reconversión profesional (al traducir habilidades transferibles en logros y keywords exigidos), (2) postulantes de regiones no capital con menor acceso a mentoría experta, y (3) grupos subrepresentados o no nativos del inglés, que suelen enfrentar desventajas adicionales ante filtros automatizados. En el estudio del MIT, >80% de los participantes vivía en países donde el inglés no es lengua nativa, lo que sugiere un efecto de "nivelación" de la IA de apoyo en la etapa de candidatura (Blumberg, 2023). El diseño también se alinea con marcos de IA responsable, como la Recomendación de la UNESCO sobre Ética de la IA (UNESCO, 2021) y con regulaciones emergentes (p. ej., el marco de AEDT en NYC exige auditorías de sesgo y avisos al candidato) (NYC DCWP, 2023).

En lo tecnológico, la propuesta integra modelos generativos de última generación para un caso aplicado (optimización integral del CV): no solo sugiere plantillas, sino que analiza el CV y la oferta para emitir recomendaciones personalizadas (contenido, palabras clave ATS, estilo y formato). Esto es consistente con la digitalización del reclutamiento y la aceleración del uso de GenAI en talento humano reportada por LinkedIn en 2024 (LinkedIn, 2024).

En lo académico, el proyecto deja nuevo conocimiento reutilizable:

- Protocolo de evaluación pre/post que combina compatibilidad ATS, tasa de respuesta/entrevistas, tiempo de edición ahorrado y satisfacción de usuario;
- Guía de prompting (patrones y anti-patrones) para perfil profesional, reescritura basada en oferta, logros con STAR y desambiguación de skills/keywords;
- Dataset anonimizado (CV de TI + ofertas) con etiquetado de entidades/skills y trazas de interacción, apto para investigación en PLN y fairness;

Finalmente, el proyecto asegura cumplimiento normativo—crítico al tratar CV—bajo la Ley 1581 de 2012 (Colombia) y normas conexas, reforzando autorización, finalidad, seguridad y confidencialidad durante carga, procesamiento, almacenamiento y eliminación de datos (Congreso de la República de Colombia, 2012). Esto fortalece la confianza del usuario y contribuye a las discusiones sobre IA responsable aplicada a gestión humana (UNESCO, 2021; NYC DCWP, 2023).

En síntesis, la justificación se sostiene en:

- **Pertinencia práctica:** mejora la empleabilidad individual y la eficiencia de preselección para reclutadores (señal más clara en menos tiempo) (HR Dive, 2018; Ladders, 2018).
- **Innovación tecnológica:** integración de IA generativa end-to-end para CV compatibles con ATS y contextualizados en la oferta (Jobscan, 2025; LinkedIn, 2024).

- **Contribución académica y social:** protocolo de evaluación, lineamientos de prompting y dataset anonimizado; adopción de marcos de ética/privacidad (UNESCO, 2021; Ley 1581/2012; NYC DCWP, 2023).

## Planteamiento del problema

Los procesos de selección contemporáneos se apoyan de forma generalizada en sistemas de seguimiento de candidatos (ATS) para manejar altos volúmenes de postulaciones; para 2025, el 99% de las empresas Fortune 500 utilizan un ATS (492 de 500), lo que evidencia su predominio en las etapas iniciales de cribado (Jobscan, 2025). Paralelamente, los reclutadores humanos dedican apenas 6–7,4 segundos al primer escaneo de un CV, de acuerdo con estudios de seguimiento ocular, por lo que cualquier desalineación entre el contenido del CV y la oferta incrementa el riesgo de descarte temprano (Ladders, 2018; HR Dive, 2018).

En la práctica, muchos profesionales —especialmente en perfiles de TI— no adaptan su CV a cada vacante (palabras clave exactas, logros cuantificables, terminología y seniority), ni ajustan el formato a los requisitos de legibilidad de los ATS. La literatura especializada indica que tablas, columnas, cajas de texto e imágenes pueden provocar errores de parseo y pérdida de información en diversos ATS, recomendándose un formato de una sola columna, encabezados estándar y listas con viñetas sencillas (Jobscan, 2025; LinkedIn Premium, s. f.). En estas condiciones, un CV con contenido relevante pero mal alineado o con formato inadecuado enfrenta un riesgo de exclusión algorítmica temprana sin revisión humana.

Existe además evidencia empírica de que la asistencia algorítmica en la redacción del CV mejora resultados medibles del proceso de búsqueda: en un experimento a gran escala, candidatos con apoyo de IA en la redacción obtuvieron más ofertas y mejores resultados de contratación (Van Inwegen et al., 2023). Este hallazgo sugiere que intervenciones sistemáticas sobre contenido, estilo y keywords pueden incrementar la tasa de avance hacia entrevistas,

especialmente cuando se alinean con los criterios que evalúan los ATS y con la lectura rápida de reclutadores.

En perfiles TI y contextos globales, el problema se intensifica: las exigencias técnicas cambian con rapidez (frameworks, nubes, certificaciones) y los candidatos que postulan a mercados angloparlantes enfrentan barreras adicionales de idioma y terminología. La combinación de alta especialización, filtros automatizados y tiempos de lectura muy breves genera una brecha entre lo que el candidato realmente puede hacer y lo que el sistema/proceso detecta en su CV (Jobscan, 2025; Ladders, 2018).

### **Problema central (usuario final)**

¿Cómo puede un profesional —especialmente en el sector TI— mejorar su CV para superar filtros automatizados y resultar más atractivo para reclutadores humanos con tiempo limitado, maximizando la correspondencia entre su experiencia real y los criterios de decisión implícitos en la vacante?

### **Núcleo de ingeniería (problema técnico)**

**Formulación técnica.** Diseñar, implementar y evaluar un agente generativo de PLN que, a partir del CV y de la descripción de la vacante, ejecute un pipeline con: (1) ingesta y normalización (PDF/DOCX/OCR y limpieza ATS-friendly), (2) extracción semántica (skills, logros, seniority, gaps), (3) análisis de la oferta (requisitos duros/blandos, keywords y sinónimos), (4) alineación CV–vacante (detección de faltantes y redundancias), (5) reescritura generativa guiada (perfil profesional, bullets con métrica, STAR), (6) validador ATS (formato y

parseo), (7) explicabilidad y trazabilidad (artefactos de diseño conforme a IEEE 1016), y (8) métricas de calidad y seguridad.

### ***Restricciones clave.***

- **Privacidad y PII:** Cumplimiento de Ley 1581/2012 (Colombia) y principios de tratamiento de datos personales (consentimiento, finalidad, seguridad, confidencialidad).
- **Latencia y costo:** Tiempo de respuesta  $p95 \leq 9$  s para iteración interactiva; costos de inferencia sostenibles para uso continuo.
- **Multilingüe:** Soporte, como mínimo, ES/EN (traducción/terminología), con preservación de significado técnico (nombres de APIs, servicios cloud, etc.).
- **Compatibilidad ATS:** Salida en formato legible por ATS (sin tablas/columnas/cajas, encabezados estándar).

### ***Riesgos y salvaguardas.***

- Alucinaciones y errores de hecho en reescritura generativa; mitigación con verificación automática y RAG/citas internas; gestión de riesgo bajo NIST AI RMF 1.0 y su perfil para Generative AI.
- Sesgo algorítmico y disparidades (por género/idioma/ubicación); monitoreo de métricas de equidad y revisión humana (NIST AI RMF).
- Filtración de PII o retención indebida; minimización de datos, redacción de PII en logs y cifrado en tránsito/reposo (alineado a Ley 1581).
- Formateo no compatible que degrade el parseo ATS; validadores y pruebas de stress en distintos perfiles de ATS.

***Criterios de éxito (medibles).***

1. Compatibilidad ATS:  $\geq 95$  % de parseo correcto (sin campos perdidos) en pruebas con plantillas y CVs reales (antes/después).
2. Mejora de contenido: +X puntos en score de keywords y relevancia vs. baseline (pre/post).
3. Resultados de búsqueda de empleo:  $+\Delta$  en respuesta/entrevistas reportadas y ofertas (cuando sea viable medir longitudinalmente), informados por la evidencia de efectos positivos de asistencia algorítmica (Van Inwegen et al., 2023).
4. Eficiencia del usuario:  $-50$  % en tiempo de edición respecto al baseline (objetivo SMART de experiencia).
5. Calidad/seguridad del modelo:  $< 5$  % de hallazgos críticos (alucinación factual) en auditorías; cumplimiento de Ley 1581;  $p95 \leq 5$  s de latencia interactiva.

**Síntesis del problema**

Dependencia de ATS en grandes empleadores y tiempos de escaneo muy breves crean riesgo de exclusión algorítmica temprana de candidatos potencialmente aptos (Jobscan, 2025; Ladders, 2018). Desalineación de contenido y formato (keywords, estructura, tablas/columnas) reduce visibilidad y efectividad del CV (Jobscan, 2025; LinkedIn Premium, s. f.). Asistencia generativa bien diseñada puede mejorar resultados de contratación, pero exige un pipeline controlado, resguardos y métricas de calidad/ética (Van Inwegen et al., 2023; NIST, 2023–2024).

## Objetivos

### Objetivo General

**Desarrollar** un sistema inteligente que, mediante el uso de técnicas de procesamiento de lenguaje natural y modelos generativos de lenguaje, optimice la creación y presentación de hojas de vida, con el fin de mejorar las oportunidades de empleabilidad de los usuarios.

### Objetivos Específicos

Establecer las tendencias y mejores prácticas actuales en la redacción y estructuración de hojas de vida, identificando los factores clave que hacen a un CV efectivo.

Elaborar una interfaz de usuario amigable e intuitiva que permita a los usuarios introducir y modificar su información personal, académica y profesional.

Incorporar un modelo generativo de lenguaje que proporcione correcciones y sugerencias en tiempo real durante la creación o modificación de la hoja de vida.

Implementar algoritmos de análisis que, basados en la información provista por el usuario y en los requerimientos de una oferta de trabajo objetivo, optimicen y destaquen los aspectos más relevantes del perfil profesional del candidato.

Validar la eficacia y utilidad del sistema a través de pruebas con usuarios finales, obteniendo retroalimentación tanto de los propios usuarios como de expertos en recursos humanos y selección de personal.

## **Marco conceptual y teórico**

La inteligencia artificial generativa (IA generativa) se refiere a sistemas de IA capaces de crear contenido nuevo (texto, imágenes, audio, código, etc.) a partir de patrones aprendidos en grandes conjuntos de datos, en contraste con enfoques de IA tradicionales que solo clasifican o predicen sobre datos existentes. En otras palabras, mientras un modelo discriminativo aprende a distinguir categorías (por ejemplo, decidir si un email es spam o no), un modelo generativo aprende la distribución subyacente de los datos para poder producir nuevos ejemplos que sean verosímiles. Este paradigma ha cobrado enorme relevancia recientemente: la llegada de sistemas como ChatGPT a fines de 2022 impulsó la adopción masiva de la IA generativa, colocándola en titulares de todo el mundo. De hecho, según un estudio de McKinsey citado por IBM, un tercio de las organizaciones ya utilizan IA generativa de forma regular en al menos un área de su negocio, y se proyecta que más del 80% habrán implementado aplicaciones de IA generativa para 2026. (Company, 2024) Esta rápida adopción sugiere que la IA generativa está transitando de ser una novedad a convertirse en una tecnología de propósito general, con un impacto transformador comparable al de la electricidad o internet en su momento (Red Hat, 2024; IBM, 2024).

### **Avances recientes en IA generativa**

Los últimos años han sido testigos de progresos vertiginosos en este campo. Inicialmente, modelos como los autoencoders variacionales y las redes generativas antagónicas (GAN) demostraron la generación de imágenes y datos sintéticos. Posteriormente, la introducción de la arquitectura Transformer (Vaswani et al., 2017) y el entrenamiento a gran escala de modelos de lenguaje dieron lugar a los Large Language Models (LLM), entre ellos la familia GPT de OpenAI. En 2019, GPT-2 mostró sorprendentes capacidades para generar texto coherente, y en

2020-2023 los modelos GPT-3 y GPT-4 alcanzaron niveles sota (state of the art) en comprensión y generación de lenguaje natural. Estas arquitecturas, entrenadas con decenas o cientos de miles de millones de parámetros, pueden dialogar, responder preguntas o componer ensayos con una fluidez casi humana. En paralelo, aparecieron modelos generativos para imágenes (DALL-E, Midjourney, Stable Diffusion), audio y código (por ejemplo, GitHub Copilot para autocompletar programación). Algunas de las aplicaciones más conocidas de la IA generativa incluyen precisamente los chatbots conversacionales (p. ej., ChatGPT, Bing Chat), la creación de imágenes sintéticas a partir de descripciones textuales (p. ej., DALL-E, Adobe Firefly), la asistencia en escritura de código (Copilot), la generación de arte y música originales, e incluso la síntesis de video o voz. Este amplio espectro de aplicaciones prácticas ha demostrado beneficios en productividad y creatividad: por ejemplo, en desarrollo de software, herramientas como Copilot pueden ahorrar hasta un 55% del tiempo de codificación rutinaria según informes de Microsoft, y en contenido visual, artistas digitales emplean IA para explorar variaciones creativas que serían impensables manualmente en poco tiempo (Red Hat, 2024).

### **Desafíos técnicos y éticos de la IA generativa**

Sin embargo, junto a las oportunidades, la IA generativa ha traído nuevos desafíos técnicos y éticos. **Sesgos y equidad:** los modelos generativos aprenden de datos masivos que suelen contener sesgos (de género, raza, ideología); en consecuencia, pueden reforzar prejuicios sociales o estereotipos indeseados en sus salidas. Un modelo que completa texto podría, por ejemplo, asociar sistemáticamente ciertos roles profesionales a un género, reflejando sesgos presentes en su corpus de entrenamiento. **Alucinaciones y veracidad:** otra inquietud es la generación de información incorrecta o inventada –fenómeno conocido como hallucinations–, en el cual la IA produce con seguridad afirmaciones que no son ciertas. A veces estos errores son

triviales (p. ej., texto sin sentido o imágenes distorsionadas), pero en casos graves se han reportado chatbots que ofrecen consejos peligrosos o modelos que generan noticias falsas con apariencia creíble, lo cual es especialmente preocupante si se usa con fines maliciosos. **Aspectos legales y de seguridad:** también existen riesgos ligados a la privacidad y la propiedad intelectual. Un modelo generativo puede exponer información sensible si los usuarios introducen datos confidenciales en las prompts, o podría vulnerar derechos de autor al reproducir contenido del entrenamiento sin atribución. De ahí que organizaciones como la UNESCO (2022) hayan emitido recomendaciones sobre la ética de la IA, instando a desarrollar sistemas transparentes, con rendición de cuentas y enfoque en derechos humanos. En el contexto de este proyecto, dichas consideraciones éticas y legales son primordiales, dado que se manejan datos personales (CV de individuos). Por ello, se adoptaron lineamientos para proteger la privacidad de los usuarios –por ejemplo, cumpliendo la Ley 1581 de 2012 de protección de datos en Colombia– y para evitar sesgos discriminatorios en las sugerencias generadas (se procuró que el modelo no hiciera recomendaciones basadas en atributos sensibles como edad, género, etnia, etc., alineándose con los principios de equidad) (Red Hat, 2024).

### **Evaluación de modelos generativos**

Otra dificultad inherente a la IA generativa es la evaluación de su desempeño. A diferencia de los sistemas tradicionales (donde se pueden usar métricas directas como exactitud, precisión, etc. sobre un conjunto de prueba con etiquetas), en tareas generativas no siempre hay una "única respuesta correcta". Evaluar la calidad de un texto, imagen o diseño generado es en buena medida subjetivo. Para afrontarlo, la comunidad científica emplea tanto métricas automáticas como evaluaciones humanas. En generación de lenguaje natural, por ejemplo, se usan métricas como BLEU, ROUGE o METEOR para comparar automáticamente el texto

producido con referencias esperadas; igualmente, en imágenes se recurre a FID (Fréchet Inception Distance) entre distribuciones de píxeles. No obstante, estas métricas matemáticas capturan solo parcialmente la calidad percibida. Por ello, es habitual complementar con evaluadores humanos que juzgan dimensiones como la coherencia, relevancia, gramática o estilo de las salidas. Incluso se han propuesto sistemas de evaluación asistida por IA, donde otro modelo (un "juez") puntúa las respuestas generativas según ciertos criterios, buscando aproximar la evaluación humana. En suma, medir el rendimiento de modelos generativos sigue siendo un campo abierto: en este proyecto se definieron métricas concretas (por ejemplo, la puntuación de compatibilidad ATS y la satisfacción del usuario con las recomendaciones) para determinar el éxito de la herramienta, pero reconociendo que una apreciación cualitativa (p. ej., opinión de expertos en selección sobre las mejoras propuestas) sería igualmente valiosa para validar la efectividad del sistema (Google Cloud, s. f.).

### **IA generativa en el ámbito de la empleabilidad**

En el contexto específico de los recursos humanos y la búsqueda de empleo, la IA generativa está comenzando a mostrar su potencial. Históricamente, la aplicación de IA en RR. HH. se había centrado más en la automatización de filtros y análisis de datos (por ejemplo, el uso de sistemas de seguimiento de candidatos o Applicant Tracking Systems, ATS para cribar CV, y algoritmos de coincidencia para emparejar perfiles con vacantes). Pero recientemente han emergido herramientas que utilizan IA para asistir directamente al candidato en la elaboración de su currículum. Por ejemplo, LinkedIn Resume Assistant integra procesamiento de lenguaje natural para sugerir frases y mejoras basadas en el rol del usuario, aprovechando datos de millones de perfiles. Del mismo modo, servicios en línea como Resume.io analizan un CV y brindan retroalimentación sobre fortalezas y debilidades, indicando si faltan ciertas palabras clave o si la

redacción podría mejorarse. Estos sistemas demuestran que existe una demanda de apoyo tecnológico para optimizar el CV; de hecho, un estudio reciente del MIT Sloan mostró que los solicitantes que emplearon asistencia algorítmica para pulir sus hojas de vida lograron 8% más probabilidades de ser contratados que aquellos sin tal ayuda. Además, obtuvieron cerca de 7.8% más ofertas de trabajo y aumentos salariales, indicando que la calidad de la redacción influye significativamente en los resultados de contratación. (Inwegen et al., 2023) Lo anterior evidencia que apoyar a los candidatos con IA en la redacción de sus CV puede tener un impacto real y positivo en su empleabilidad. Sin embargo, también se ha detectado un vacío importante: muchas de las soluciones actuales para optimizar CV se quedan en funcionalidades básicas o poco flexibles. Herramientas como Resume.io tienden a basarse en plantillas fijas o en simples comparaciones de palabras clave, y no aprovechan modelos generativos avanzados capaces de adaptar sus sugerencias al contexto específico del candidato y la oferta. Incluso los asistentes integrados (como el de LinkedIn) ofrecen recomendaciones algo genéricas, porque no generan contenido nuevo completamente contextualizado, sino que recomiendan frases comunes ya existentes. En este contexto, nuestro proyecto se plantea llenar ese vacío integrando una IA generativa (GPT-4) en el proceso de optimización de currículos. A diferencia de los enfoques tradicionales, el sistema propuesto genera recomendaciones personalizadas en cuatro dimensiones clave del CV (contenido, palabras clave, estilo y formato) para cada oferta laboral específica. Esto permite ir más allá de las simples listas de verificación: la IA puede, por ejemplo, sugerir redactar cierta experiencia de otra manera para enfatizar habilidades relevantes que el candidato posee pero no estaba destacando. Asimismo, puede detectar términos del anuncio de empleo (p. ej., "Java Spring Boot") e indicar al usuario que los incorpore exactamente si tiene esos conocimientos –evitando que un CV bien calificado sea filtrado por ATS por no

contener la palabra exacta. Cabe destacar que más del 75% de los CV pueden ser descartados automáticamente por los ATS sin jamás ser leídos por un humano, debido a formato incorrecto o falta de palabras clave adecuadas. Un candidato calificado podría "desaparecer en un agujero negro digital" simplemente por cómo redactó su CV. Lo anterior resalta la necesidad crítica de herramientas que ayuden a los solicitantes a adaptar su CV a los criterios de selección automatizada, sin perder la calidad para el lector humano (Van Inwegen et al., 2023; Deloitte, 2021; Bratak Team, 2025).

### **Aplicaciones Existentes y Estado del Arte**

En el mercado actual existen diversas herramientas que ofrecen asistencia en la optimización de currículos, cada una con enfoques y capacidades diferentes. A continuación se presenta un análisis comparativo de las principales soluciones disponibles:

**LinkedIn Resume Assistant** integra procesamiento de lenguaje natural para sugerir frases y mejoras basadas en el rol del usuario, aprovechando datos agregados de millones de perfiles. Sin embargo, sus recomendaciones tienden a ser genéricas y basadas en plantillas predefinidas, sin adaptarse al contexto específico de cada oferta laboral. Según evaluaciones recientes, LinkedIn Resume Assistant alcanza una precisión estimada del 80-90% en reconocimiento de características ATS-friendly, pero carece de herramientas detalladas de matching de palabras clave y personalización contextual (Elite Resumes, 2025). (Best ATS Resume Checker Tools 2026: Free & Paid Reviews, 2026)

**Resume.io** es una plataforma en línea que analiza CV y brinda retroalimentación sobre fortalezas y debilidades, indicando si faltan ciertas palabras clave o si la redacción podría mejorarse. Su enfoque se basa principalmente en comparaciones de palabras clave y plantillas fijas, limitando la flexibilidad y personalización de las sugerencias. Aunque útil para usuarios sin

experiencia, no aprovecha modelos generativos avanzados. Resume.io ofrece plantillas ATS-friendly, pero requiere múltiples iteraciones manuales y ofrece análisis genéricos. (Resume.io: Plantillas ATS-Friendly y Funcionalidades, 2024)

**Jobscan** se especializa en la simulación de compatibilidad ATS, permitiendo a los usuarios comparar su CV con descripciones de puestos para identificar keywords faltantes. Su fortaleza radica en el análisis de formato y compatibilidad ATS, pero carece de capacidades de generación de contenido o reescritura inteligente. Jobscan proporciona herramientas detalladas de matching, pero con un costo mensual elevado (aproximadamente \$49.95/mes) y enfoque principalmente en análisis de keywords más que en generación contextualizada. (Jobscan - ATS Resume Scanner & Builder, 2025)

**Otras herramientas** como Zety, ResumeGenius y Canva Resume Builder ofrecen plantillas y asistentes básicos, pero se enfocan más en el diseño visual que en la optimización de contenido para ATS o la personalización contextual.

**El vacío identificado** en estas soluciones es la falta de integración de modelos generativos avanzados capaces de adaptar sus sugerencias al contexto específico del candidato y la oferta. Mientras las herramientas existentes se basan en reglas predefinidas o plantillas, el sistema propuesto en este proyecto utiliza GPT-4 para generar recomendaciones completamente personalizadas y contextualizadas, llenando este vacío en el mercado.

### **Glosario de Términos Técnicos**

Para facilitar la comprensión del documento, se presenta a continuación un glosario de los términos técnicos más relevantes utilizados en este proyecto:

**ATS (Applicant Tracking System):** Sistema automatizado de seguimiento de candidatos utilizado por empresas para filtrar, clasificar y gestionar aplicaciones de empleo. Los ATS

analizan CVs buscando palabras clave, verificando formato y calculando puntajes de compatibilidad con las descripciones de puestos.

**IA Generativa (Generative AI):** Tipo de inteligencia artificial capaz de crear contenido nuevo (texto, imágenes, audio, código) a partir de patrones aprendidos en grandes conjuntos de datos, en contraste con sistemas que solo clasifican o predicen sobre datos existentes.

**LLM (Large Language Model):** Modelo de lenguaje a gran escala entrenado con miles de millones de parámetros en extensos corpus de texto, capaz de comprender y generar lenguaje natural con alta coherencia. Ejemplos incluyen GPT-4, GPT-3.5, Claude, entre otros.

**Prompt Engineering:** Técnica de diseño y optimización de instrucciones (prompts) dadas a modelos de lenguaje para obtener respuestas de mayor calidad y relevancia. Incluye la estructuración de contexto, roles, criterios de salida y ejemplos.

**PLN (Procesamiento de Lenguaje Natural):** Campo de la inteligencia artificial que se enfoca en la interacción entre computadoras y lenguaje humano, incluyendo tareas como comprensión, generación, traducción y análisis de texto.

**Machine Learning:** Subcampo de la IA que permite a los sistemas aprender y mejorar automáticamente a partir de la experiencia sin ser programados explícitamente, mediante el reconocimiento de patrones en datos.

**Blazor WebAssembly:** Framework de Microsoft que permite ejecutar aplicaciones .NET directamente en el navegador web mediante WebAssembly, proporcionando capacidades de aplicación de escritorio en un entorno web.

**Azure OpenAI Service:** Servicio de Microsoft Azure que proporciona acceso seguro y escalable a modelos de OpenAI (como GPT-4) con características empresariales de seguridad, privacidad y cumplimiento normativo.

**OCR (Optical Character Recognition):** Tecnología que convierte imágenes de texto (escaneadas o fotografiadas) en texto editable mediante reconocimiento de caracteres.

**Azure Key Vault:** Servicio de Microsoft Azure para almacenar y gestionar de forma segura secretos, claves de cifrado y certificados utilizados por aplicaciones y servicios en la nube.

**Azure Blob Storage:** Servicio de almacenamiento de objetos de Microsoft Azure para almacenar grandes cantidades de datos no estructurados, como archivos, imágenes, videos y logs.

**ASP.NET Core:** Framework web de código abierto desarrollado por Microsoft para construir aplicaciones web modernas, APIs REST y servicios en la nube.

## **Plataforma de Trabajo**

El sistema se desarrolló utilizando la siguiente plataforma tecnológica:

**Frontend:** Blazor WebAssembly ejecutándose en el navegador del usuario, proporcionando una interfaz de usuario interactiva sin necesidad de instalación de software adicional.

**Backend:** ASP.NET Core 8.0 desplegado en Azure App Service, encargado de la orquestación del flujo de procesamiento, construcción de prompts, comunicación con servicios de IA y gestión de lógica de negocio.

### **Servicios en la Nube (Azure):**

**Azure OpenAI Service:** Proporciona acceso a GPT-4 para la generación de recomendaciones

**Azure Blob Storage:** Almacenamiento de métricas anónimas y logs del sistema

**Azure Key Vault:** Gestión segura de claves API y secretos

**Azure App Service:** Hospedaje del backend con escalabilidad automática

**Herramientas de Evaluación:** Jobscan (simulador ATS) para validar compatibilidad y métricas de optimización.

Esta arquitectura en la nube proporciona escalabilidad, seguridad y cumplimiento normativo, facilitando el despliegue y mantenimiento del sistema.

### **Resumen del marco teórico**

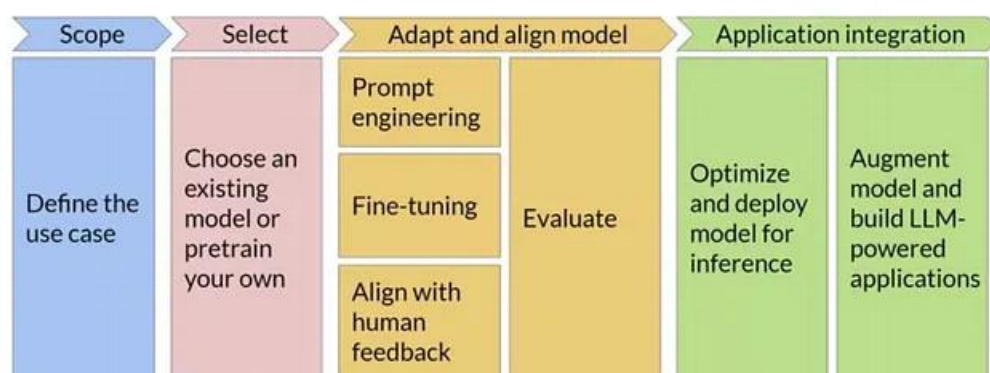
En resumen, el presente marco teórico ha revisado desde la definición amplia de IA generativa y sus diferencias con otros enfoques, hasta los avances recientes que la han llevado a revolucionar múltiples sectores, pasando por los desafíos éticos/técnicos que supone (sesgos, alucinaciones, dificultad de evaluación, aspectos legales), para finalmente aterrizar en el dominio particular de los CV y la empleabilidad. Se evidencia que, si bien existen iniciativas previas y la utilidad de la IA en este campo está respaldada por estudios (p. ej., la mejora en tasas de contratación con CV asistidos por IA), persiste un vacío en soluciones sofisticadas que combinen PLN generativo y conocimiento contextual del mercado laboral para optimizar hojas de vida. Este proyecto de grado se sustenta precisamente en ese vacío: toma lo mejor del estado del arte en modelos de lenguaje (un LLM de última generación) y lo aplica de forma novedosa para potenciar la empleabilidad de los usuarios, asegurando además un desarrollo responsable en términos de ética y privacidad. A continuación, en el capítulo de Metodología, se detalla cómo se llevó a cabo la construcción de esta solución, siguiendo un ciclo de vida estructurado en cuatro fases que garantizó la trazabilidad desde los objetivos iniciales hasta los resultados obtenidos (Red Hat, 2024).

## Metodología

En este capítulo se describe la metodología empleada para diseñar, desarrollar y evaluar el sistema de optimización de CV con IA generativa. Se adoptó un enfoque basado en el ciclo de vida de proyectos de IA con cuatro fases principales: **Scope, Select, Adapt & Align y Application Integration** (alcance, selección, adaptación/alineación e integración en la aplicación).

### Figura 1

*Ciclo de vida del proyecto de IA generativa*



*Nota.* Adaptado de Adik (2023).

### Justificación de la Metodología

Esta metodología fue seleccionada específicamente para proyectos de software de inteligencia artificial porque:

**Alineación con el Ciclo de Vida de IA:** A diferencia de metodologías tradicionales como Scrum o Waterfall, este enfoque está diseñado específicamente para proyectos que integran modelos de machine learning y LLMs, considerando las particularidades del desarrollo de sistemas de IA (selección de modelos, ingeniería de prompts, evaluación iterativa).

**Gestión de Incertidumbre:** Los proyectos de IA presentan mayor incertidumbre técnica que proyectos de software tradicional. Esta metodología maneja esta incertidumbre mediante fases de exploración (Select) y refinamiento iterativo (Adapt & Align) antes de la integración final.

**Enfoque en Calidad de Modelo:** A diferencia de metodologías ágiles tradicionales que se enfocan en entregas incrementales de funcionalidad, esta metodología prioriza la calidad y alineación del modelo de IA antes de integrarlo en la aplicación, mediante evaluación continua y ajuste de prompts.

**Trazabilidad de Decisiones Técnicas:** Cada fase produce artefactos documentados (selección de modelo, evolución de prompts, métricas de evaluación) que permiten justificar decisiones técnicas y facilitar mejoras futuras.

**Adaptación a LLMs:** La fase "Adapt & Align" reconoce que con modelos pre-entrenados como GPT-4, el desarrollo se centra en ingeniería de prompts más que en entrenamiento, requiriendo un enfoque metodológico diferente al machine learning tradicional.

La Figura 1 ilustra este ciclo de vida propuesto, que está alineado con prácticas recomendadas en el desarrollo de soluciones de machine learning y específicamente adaptado a proyectos con Large Language Models integrados en aplicaciones reales (Adik, 2023). Cada fase del ciclo de vida comprende actividades específicas, produce ciertos artefactos o entregables, define criterios de salida (completitud) y considera métricas para evaluar su éxito.

### **Fase 1: Scope (Delimitación del Alcance)**

En la fase de Scope se establecieron los fundamentos del proyecto: qué se pretende lograr, por qué es importante y cómo se medirá el éxito. Las actividades incluyeron una delimitación precisa del caso de uso –optimizar currículos vitae para mejorar la empleabilidad de profesionales, especialmente en el sector TI– y la formulación de los objetivos específicos. En

este punto se realizó una investigación inicial (basada en el estado del arte descrito en el capítulo 5) que confirmó la problemática: muchos candidatos calificados son filtrados por sistemas ATS o pierden oportunidades por presentar CV subóptimos. Por tanto, el objetivo general se definió como "Desarrollar un sistema basado en IA generativa que brinde sugerencias personalizadas para mejorar un CV, aumentando su compatibilidad con filtros automatizados (ATS) y su impacto ante reclutadores humanos" (Van Inwegen et al., 2023; Deloitte, 2021).

A partir de este objetivo macro, se derivaron objetivos específicos y criterios de éxito cuantificables. Por ejemplo, un indicador clave delineado fue el score de compatibilidad ATS: con base en estadísticas del mercado (donde CV no optimizados suelen rondar un 60–70% de coincidencia con las ofertas), se estableció como meta que los CV procesados por la herramienta alcancen  $\geq 80\%$  de compatibilidad en pruebas con simuladores ATS (p. ej., Jobscan) – umbral típicamente recomendado para asegurar que un CV pase los filtros automáticos. Este criterio permitiría validar numéricamente que el proyecto cumple su propósito principal.

Adicionalmente, se definieron métricas de satisfacción del usuario (porcentaje de usuarios que encuentran útiles las sugerencias, medido mediante encuestas con escala de Likert de 1 a 5; objetivo: satisfacción  $\geq 4/5$  en promedio) y métricas de calidad de la redacción (evaluaciones cualitativas de expertos comparando versiones de CV antes vs. después de la optimización, esperando que la mayoría perciba mejoras significativas). También se consideraron requisitos no funcionales desde esta fase de planeación: por ejemplo, garantizar tiempos de respuesta razonables (idealmente  $< 10$  segundos por análisis), y adherencia a principios de seguridad y privacidad (no almacenar datos personales sensibles, cumplimiento legal) (Van Inwegen et al., 2023; Deloitte, 2021).

Los artefactos generados al cierre de Scope incluyeron un documento de requisitos y planificación donde quedaron plasmados: (a) la descripción del problema y caso de uso, (b) los objetivos generales/específicos, (c) las métricas de éxito y criterios de aceptación, (d) las restricciones y supuestos (por ejemplo, se asumió disponibilidad de conexión a la API de OpenAI, y se restringió el alcance a formato de CV cronológico en PDF o Word para la fase experimental), y (e) un análisis preliminar de riesgos (riesgos técnicos como posible dificultad del modelo para entender ciertos CV muy especializados, riesgos de adopción como resistencia de usuarios a modificar su CV según sugerencias de una IA, y riesgos éticos/legales como manejo de datos personales). Este documento sirvió como guía para las siguientes fases, actuando a modo de "contrato" de lo que el proyecto debía lograr.

**Criterios de salida de Scope:** El consenso del equipo en que el problema y alcance estaban claramente definidos, y la aprobación del asesor académico, marcaron el cierre de esta fase. En términos de revisión, se comprobó que cada objetivo específico trazado era SMART (específico, medible, alcanzable, relevante y con tiempo delimitado) y que existía al menos una métrica o entregable asociado a cada objetivo para verificar su cumplimiento. Con ello, se dio paso a la fase de selección de la solución técnica.

## **Fase 2: Select (Selección del Modelo)**

En la fase de Select se abordó la pregunta: ¿Con qué tecnología o modelo de IA generativa se implementará la solución?. Dado que el corazón del sistema es un modelo de lenguaje natural que genere recomendaciones sobre el CV, esta etapa fue crucial. Las actividades se centraron en investigar, comparar y escoger el modelo de IA más adecuado, considerando los criterios de desempeño, costo, viabilidad técnica y alineación ética.

Primero, se exploraron las alternativas disponibles en 2023 para generación de texto orientada a recomendaciones de escritura. Dos caminos principales se identificaron: (1) utilizar un modelo ya existente de alta performance (por ejemplo, GPT-3.5 o GPT-4 de OpenAI, u otros LLMs comerciales/de código abierto), o (2) entrenar desde cero o afinar un modelo propio especializado en textos de CV. Se recopiló información de model cards y la literatura: por un lado, GPT-4 destacaba por su desempeño superior en tareas de comprensión y generación de texto, habiendo sido entrenado con un amplio corpus que le confería conocimiento general extenso (incluyendo cómo redactar profesionalmente) y con capacidades demostradas en seguir instrucciones complejas. Por otro lado, existían modelos de código abierto (p. ej., Bloom, LLaMA en sus variantes adaptadas) que podrían afinarse con datos de CV; sin embargo, ello implicaba disponer de un conjunto significativo de datos específicos y recursos computacionales para entrenamiento, lo que escapaba al alcance temporal y de infraestructura del proyecto. Además, un modelo entrenado desde cero podría no alcanzar la sofisticación lingüística de GPT-4 sin millones de ejemplos y tuning. Por motivos de calidad y practicidad, se optó por seleccionar GPT-4 (vía la API de Azure OpenAI) como núcleo generativo del sistema. Esta decisión se justificó en múltiples criterios: **Precisión** – GPT-4 ha mostrado entendimiento casi humano del contexto y produce recomendaciones bien fundamentadas; **Capacidad de adaptación** – al ser un modelo de propósito general, puede aplicar su conocimiento a cualquier dominio (incluyendo CV) sin requerir entrenamiento adicional, más allá de buenas instrucciones (prompts); **Facilidad de integración** – Azure OpenAI ofrece endpoints API seguros y escalables para acceder al modelo, simplificando la implementación; **Tiempo/Costo** – aunque GPT-4 tiene un costo por consulta y ciertas limitaciones de velocidad, es menor comparado con embarcarse en entrenar un modelo propio grande (lo cual costaría semanas y costosas GPU);

**Actualizaciones y ética** – GPT-4 ya incorpora filtros de contenido y alineación con valores éticos mediante técnicas como RLHF (Reinforcement Learning from Human Feedback), reduciendo la probabilidad de generar respuestas inapropiadas. Se evaluó también GPT-3.5 (más económico y rápido) pero en pruebas piloto GPT-4 brindó sugerencias de CV más pertinentes y con redacción de mayor calidad (por ejemplo, GPT-3.5 tendía a recomendaciones más genéricas, mientras GPT-4 hacía observaciones más contextuales sobre la oferta laboral). Por tanto, la decisión final fue usar GPT-4 para generación de las recomendaciones textuales, accediéndolo mediante el servicio Azure OpenAI.

Además del modelo de IA principal, en esta fase se seleccionaron otras herramientas y componentes necesarios para la solución completa. Por ejemplo, se decidió utilizar Jobscan (plataforma simuladora de ATS) en la etapa de evaluación, para medir la compatibilidad CV-oferta con y sin optimización. También se escogió la plataforma de desarrollo: una arquitectura web con un front-end en Blazor WebAssembly (permitiendo una interfaz interactiva en el navegador del usuario) y un back-end en ASP.NET Core desplegado en Azure, encargado de comunicarse con la API de GPT-4 y realizar el procesamiento del CV. Esta arquitectura cumpliría con requisitos de portabilidad (el usuario solo necesita un navegador), escalabilidad (Azure permite escalar instancias del back-end según la carga) y seguridad (Blazor Wasm aislaría la lógica de UI en el cliente, mientras que el servidor manejaría los tokens de API y la lógica sensible). Se seleccionó Azure como entorno de despliegue por su integración nativa con OpenAI, sus servicios de seguridad gestionada y la familiaridad previa del desarrollador con este ecosistema. Otro componente seleccionado fue una librería de NLP para realizar análisis locales sencillos, como contar palabras clave en el texto del CV o identificar secciones; finalmente, se

optó por utilizar expresiones regulares y utilidades básicas de .NET para esto, dado que GPT-4 realizaría la parte más compleja de generación (Van Inwegen et al., 2023; Deloitte, 2021).

**Criterios de salida de la fase Select:** Tener documentada la elección del modelo y servicios, junto con la justificación técnica de cada decisión, y contar con los accesos y configuraciones preparados. Esto incluyó: haber obtenido las credenciales de la API de Azure OpenAI (clave y endpoint para GPT-4), crear los recursos Azure necesarios (servicio de App Service para el backend, almacenamiento para logs/métricas anónimas, etc.), e instalar/configurar en el entorno de desarrollo las SDK o bibliotecas para consumir dichos servicios. Solo una vez que todo el stack tecnológico estuvo claramente definido y listo para usarse, se consideró completada esta fase. Como checklist, al final de Select el proyecto tenía: un modelo escogido (GPT-4) con parámetros iniciales (p. ej., temperatura, máximos tokens sugeridos según contexto de CV), una arquitectura de software decidida y esbozada (ver Figura 1 del documento, arquitectura general), y las herramientas secundarias identificadas (Jobscan para pruebas, etc.). Con esta base sólida, se procedió a desarrollar y adaptar efectivamente la solución en la siguiente fase.

### **Fase 3: Adapt & Align (Adaptación y Alineación del Modelo)**

En la fase de Adaptación y alineación se llevó a cabo el desarrollo iterativo de la solución de IA, ajustando el modelo seleccionado (GPT-4) a las necesidades específicas del proyecto y asegurando que sus resultados fueran válidos, útiles y alineados con los objetivos. Esta etapa involucró principalmente tareas de ingeniería de prompts, pruebas y refinamientos sucesivos, así como la integración de retroalimentación humana para guiar al modelo hacia mejores resultados.

Dado que se optó por no entrenar un modelo desde cero ni hacer fine-tuning adicional (por las razones expuestas en la fase Select), la clave fue explotar al máximo el modelo pre-

entrenado mediante prompts efectivos. Se diseñó un prompt maestro que sirviera de instrucción a GPT-4 para generar las recomendaciones de mejora del CV. Este prompt, tras varias iteraciones, adoptó una estructura fija: empezaba contextualizando al modelo en su rol ("Eres un asistente experto en Recursos Humanos especializado en mejora de currículos"), luego proporcionaba la información necesaria (el contenido del CV del usuario y la descripción de la oferta de trabajo objetivo), y finalmente solicitaba la salida de forma organizada ("Genera sugerencias concretas de mejora divididas en: Contenido, Palabras Clave, Estilo, Formato. Sé específico y conciso..."). Adicionalmente, se incluyeron instrucciones para alinear éticamente las respuestas, por ejemplo: "No reveles información personal del candidato, no hagas suposiciones no basadas en el texto, ni aportes sesgos; sé objetivo y profesional en el tono". Esta técnica de prompt engineering es fundamental, ya que la calidad y relevancia de las respuestas del modelo dependen fuertemente de cómo se planteen las instrucciones. Estudios recientes han remarcado que una buena formulación del prompt puede mejorar sustancialmente el desempeño de un LLM sin necesidad de modificar sus parámetros internos. Por ello, se experimentó con distintas variantes de prompt: por ejemplo, al inicio el modelo devolvía sugerencias demasiado generales, así que se le añadieron indicaciones como "si es posible, menciona ejemplos específicos o palabras puntuales que falten"; en otra iteración, se ajustó la petición para que limitara la cantidad de sugerencias (máximo 8 por categoría) y así forzar concisión. Cada cambio de prompt se validó probándolo con CV de ejemplo y evaluando si las recomendaciones eran pertinentes. Se utilizaron varios CV de prueba (algunos creados ad hoc y otros reales con permiso, anonimándolos) junto con ofertas laborales tipo, midiendo los resultados. Esta afinación iterativa del prompt continuó hasta lograr un equilibrio óptimo entre exhaustividad y precisión en las sugerencias. En total se realizaron

aproximadamente 15 ajustes mayores al prompt a lo largo del desarrollo, hasta congelar su versión final (ver Anexo A para documentación detallada de los prompts) (Red Hat, 2024).

Además de la ingeniería de prompts, se incorporaron mecanismos de evaluación y retroalimentación continua en esta fase. Por un lado, tras cada iteración significativa, se evaluaba el modelo con las métricas definidas: se corrían tests en los que un CV antes y después de aplicar las sugerencias generadas eran comparados con la oferta en el simulador ATS (Jobscan), registrando la mejora en puntaje. Este proceso permitió cuantificar el impacto de ciertos cambios: por ejemplo, cuando se añadió al prompt que enfatizara la inclusión de palabras clave exactas de la oferta, se observó en las pruebas un salto consistente de ~10 puntos porcentuales en la compatibilidad ATS de los CV optimizados. Por otro lado, se buscó alinear el modelo con preferencias humanas reales. Si bien no se implementó un complejo esquema de Reinforcement Learning from Human Feedback (RLHF) con múltiples rondas de entrenamiento (lo cual excedía el alcance de un proyecto de grado), sí se optó por una alternativa asequible: involucrar a expertos humanos en evaluaciones periódicas. En concreto, se contactó a dos profesionales de recursos humanos (reclutadores con experiencia) para que actuaran como evaluadores externos. En etapas intermedias del desarrollo, se les proporcionaron ejemplos de salida del modelo: por ejemplo, un CV original junto con las sugerencias de mejora generadas para una oferta dada. Se recopiló su feedback cualitativo sobre si esas sugerencias les parecían acertadas, neutrales o erróneas. Esta retroalimentación fue invaluable para detectar aspectos sutiles: en una ronda inicial, uno de los expertos señaló que el modelo recomendaba eliminar la sección de "intereses personales" del CV, lo cual ATS-wise tiene sentido (pues no aporta a palabras clave), pero humanamente podría quitar personalidad al perfil – tras discutirlo, se decidió conservar esa recomendación pero añadiendo en el prompt una aclaración: "(Nota: si bien puedes sugerir

eliminar secciones poco relevantes para ATS, advierte al usuario que esas secciones pueden tener valor personal)". Así, el modelo empezó a matizar esa sugerencia en particular, haciéndola más alineada con una visión humana equilibrada. De manera similar, la intervención humana ayudó a calibrar el tono de las recomendaciones: se prefería un tono motivador pero formal, evitando lenguaje demasiado enfático como "debes hacer X", en favor de "sería conveniente X". Estas consideraciones se incorporaron ajustando el prompt y, en algunos casos, filtrando o posprocesando levemente la salida del modelo (por ejemplo, reemplazando frases imperativas duras por formulaciones más sugerentes mediante una simple función de reemplazo de texto en el backend) (Van Inwegen et al., 2023; Deloitte, 2021).

Un componente importante de la alineación fue también la verificación de sesgos o contenidos inadecuados en las respuestas del modelo. Durante las pruebas, se monitoreó que las sugerencias no hicieran referencia a características sensibles (edad, género, orientación, salud, etc.), acorde a las directrices éticas. GPT-4 ya cuenta con filtros internos, pero se validó manualmente esta cuestión y, efectivamente, no surgieron casos de sesgos obvios en nuestras pruebas (el modelo nunca sugirió nada relacionado con datos personales sensibles). Por precaución, se añadió en el prompt una instrucción explícita de no mencionar ese tipo de atributos. Asimismo, se configuró el servicio de Azure OpenAI con las content filters activadas en nivel estricto, para prevenir cualquier posible salida tóxica o fuera de los lineamientos de uso de OpenAI. Afortunadamente, dado el contexto profesional acotado, no se presentaron incidentes de este tipo (Red Hat, 2024).

En términos de artefactos generados, la fase de Adapt & Align produjo principalmente el código fuente de la aplicación en su primera versión completa: se desarrolló el módulo de lógica que envía el prompt a GPT-4 y recibe las sugerencias, formateándolas adecuadamente para

mostrarlas al usuario. También se produjeron datasets de prueba (CV de ejemplo y ofertas) y registros de resultados de las evaluaciones (planillas con los puntajes ATS antes/después, notas de los expertos humanos sobre cada caso, etc.). Estos registros sirvieron como documentación de la efectividad del modelo a lo largo del tiempo y permitieron rastrear mejoras. Un entregable adicional fue la documentación técnica del prompt final y consideraciones de su desarrollo – dado que la ingeniería de prompts es un arte relativamente nuevo, se consideró valioso documentar por qué se llegó a esa formulación final (por si a futuro otro desarrollador quisiera retomar el proyecto o mejorarlo con técnicas adicionales, como few-shot prompting o incorporar un paso de recuperación de información) (Van Inwegen et al., 2023; Deloitte, 2021).

**Criterios de salida de la fase Adapt & Align:** Dar por concluido el desarrollo iterativo cuando se cumplió con los objetivos de desempeño definidos: específicamente, cuando en las pruebas internas 100% de los CV de ejemplo mejoraron su puntaje ATS superando el umbral del 80% tras la optimización (en las últimas iteraciones se logró consistentemente, con mejoras promedio de ~20 puntos como se detalla en Resultados), y cuando los expertos humanos calificaron positivamente las sugerencias en la gran mayoría de casos (en la última ronda, 9 de 10 CV optimizados fueron percibidos como claramente superiores al original por los reclutadores, señalando solo ajustes menores o preferencias personales en algún caso aislado). Alcanzados estos indicadores, y sin hallar ya cambios significativos al variar el prompt, se decidió congelar el desarrollo del modelo y avanzar a la integración final en una aplicación completa utilizable por los usuarios finales (Van Inwegen et al., 2023; Deloitte, 2021).

#### **Fase 4: Application Integration (Integración en la Aplicación)**

La última fase del ciclo de vida, Application Integration, consistió en incorporar el modelo generativo ya adaptado dentro de una aplicación funcional, asegurando su despliegue

estable, su accesibilidad a usuarios finales y el cumplimiento de los requisitos de entorno (frontend, backend, base de datos, etc.). En esencia, aquí se concretó la solución como un sistema de software listo para usar, integrando todos los componentes y preparándolo para pruebas finales e implantación.

Las actividades principales incluyeron:

**Desarrollo del Front-end:** Se construyó una interfaz web intuitiva usando Blazor WebAssembly. Esta interfaz permite al usuario cargar su CV (en formato .pdf o .docx), introducir o pegar el texto de una oferta de empleo de interés, y hacer clic en "Optimizar". Se diseñó con una UX sencilla, mostrando campos para visualizar el CV original (parseado a texto) y la descripción de la oferta, y luego un panel de resultados con las sugerencias categorizadas (Contenido, Palabras Clave, Estilo, Formato) que genera la IA. También se incluyeron indicadores visuales, como un score de compatibilidad ATS estimado (mostrando, por ejemplo, 85/100 en verde si supera 80, o en rojo si es bajo), e íconos de información junto a cada sugerencia que el usuario puede clicar para ver una explicación (estas explicaciones se basan en plantillas estáticas, p. ej., junto a "Añadir palabra clave X" un ícono muestra "Porque aparece en la oferta y es relevante para ATS"). El front-end se diseñó responsive para que pudiera usarse desde dispositivos móviles, aunque priorizando vista de escritorio por la naturaleza de editar CV (Van Inwegen et al., 2023; Deloitte, 2021).

**Implementación del Back-end y lógica de negocio:** En el servidor ASP.NET Core, se desarrollaron las API REST necesarias. La principal fue un endpoint /optimizeCV que recibe el CV y la oferta (el CV se procesa extrayendo solo texto plano relevante, ignorando formateo), construye el prompt según la plantilla final definida, llama a la API de OpenAI con la clave secreta (almacenada de forma segura en Azure Key Vault), y espera la respuesta de GPT-4. Al

recibir las sugerencias generadas, el backend las post-procesa mínimamente (p. ej., formatear en listas, censurar si viniera accidentalmente algún dato personal crudo aunque no sucedió, etc.) y las envía de vuelta al cliente para desplegar. Asimismo, esta capa realiza el cálculo del score ATS aproximado: para ello, internamente se comparan las palabras clave de la oferta con el texto del CV (tanto original como optimizado si el usuario decide aplicar ciertos cambios en la interfaz). Se integró aquí la herramienta Jobscan a modo de servicio externo: el backend le envía el CV y la oferta de forma automatizada y recibe un análisis detallado (palabras coincidentes, faltantes, porcentaje). Esto permitió mostrar al usuario un diagnóstico tanto antes como después de implementar las sugerencias (Van Inwegen et al., 2023; Deloitte, 2021).

**Integraciones de Seguridad y Datos:** Se tomaron medidas estrictas para proteger la información personal manejada. Como principio arquitectónico, no se almacenó ningún CV del usuario en bases de datos persistentes. Todos los análisis se hacen en memoria y, una vez finalizada la sesión, el CV y resultados se descartan. Solo se guardan métricas anónimas para fines de evaluación del proyecto (por ejemplo, % de mejora ATS promedio, calificaciones de satisfacción dadas por los usuarios de prueba), pero sin ninguna referencia al contenido específico del CV ni datos identificatorios. La comunicación entre el front-end y back-end está protegida bajo HTTPS con certificado TLS, y las llamadas a la API de OpenAI desde el servidor también viajan cifradas. En Azure, se siguieron las best practices de seguridad: las claves API nunca viajan al cliente, permanecen solo en el servidor; se usaron Managed Identities donde fue posible; y se aplicaron políticas de CORS restrictivas para que solo el dominio del front-end pudiera consumir el backend. Estas medidas garantizan conformidad con la Ley 1581 de 2012 (Protección de Datos Personales en Colombia) en cuanto a principios de confidencialidad, necesidad y finalidad. Adicionalmente, se preparó un aviso de privacidad para los usuarios,

informado en la interfaz, explicando qué datos procesa la herramienta y con qué propósito, obteniendo consentimiento tácito al usarla (Google Cloud, s. f.; Van Inwegen et al., 2023; Deloitte, 2021).

**Pruebas finales de desempeño y carga:** Antes de dar por concluida la integración, se sometió el sistema a pruebas de rendimiento. Se verificó que el tiempo de respuesta fuera adecuado: en tests con GPT-4, cada solicitud de optimización tardó en promedio 5 a 8 segundos en obtener la respuesta y mostrar las sugerencias. Este tiempo se consideró aceptable para un contexto interactivo (aunque es posible optimizarlo, p. ej., con GPT-3.5 era ~2 segundos pero con menor calidad). Asimismo, se realizó una pequeña prueba de carga simulando múltiples usuarios concurrentes (usando utilidades de JMeter y scripts que disparaban ~5 solicitudes simultáneas repetidamente). El sistema se mantuvo estable sin errores y con tiempos de respuesta consistentes (~6–10 segundos bajo 5 concurrencias, principalmente limitados por la latencia de la API externa de OpenAI). Se observó que, para escalabilidad horizontal, sería trivial aumentar instancias del backend en Azure si se esperaran decenas de usuarios concurrentes en producción; la arquitectura en la nube adoptada facilita esta elasticidad. Otro aspecto probado fue la portabilidad: se logró acceder a la herramienta desde distintos navegadores (Chrome, Firefox, Edge) y dispositivos sin problemas, dado que Blazor Wasm es compatible con cualquier navegador moderno.

**Feedback e Iteración final con usuarios de prueba:** Con la aplicación integrada y estable, se procedió a un piloto con usuarios reales (un grupo pequeño de ~5 usuarios voluntarios buscando empleo). Se les dio acceso a la herramienta y se recolectó su retroalimentación mediante un formulario integrado (donde calificaban de 1 a 5 estrellas la utilidad de las sugerencias, y podían escribir comentarios). Esta fue una suerte de prueba de aceptación. Los

resultados fueron muy positivos: la satisfacción global promedio fue 4.5/5 (90%), y todos los usuarios indicaron que aplicarían varias de las sugerencias a sus CV. Además, el 80% dijo que recomendarían la herramienta a colegas o amigos. Algunos comentarios confirmaron hipótesis iniciales: por ejemplo, las sugerencias de "Palabras Clave" fueron las más valoradas, pues muchos usuarios admitieron que no solían prestar atención a match de terminología antes de la herramienta. También hubo observaciones sobre posibles mejoras, que sirven para trabajo futuro: un usuario mencionó que le gustaría un modo "revisión gramática profunda" adicional, y otro sugirió poder cargar también la carta de presentación para optimizarla junto al CV. Estas ideas quedaron documentadas para posibles extensiones, pero no se implementaron por limitación de alcance. Lo importante es que la integración final pasó la prueba con usuarios, validando que la herramienta no solo funciona técnicamente sino que es realmente útil en un contexto práctico.

Con la aplicación integrada, probada y validada, se dio por concluida esta fase y, por tanto, el desarrollo del proyecto. Se logró implementación completa de la solución propuesta, con todos sus componentes funcionando conjuntamente: interfaz web, backend, modelo GPT-4, módulo ATS y medidas de seguridad. Los criterios de salida de Application Integration –tener un sistema desplegado cumpliendo requisitos funcionales y no funcionales– se alcanzaron plenamente. En síntesis, el ciclo de vida seguido (Scope → Select → Adapt & Align → Application Integration) proporcionó una hoja de ruta estructurada para ir desde la concepción hasta la realización de la idea, asegurando en cada etapa la alineación con los objetivos originales y permitiendo introducir mejoras de manera controlada. Esta metodología también facilita la trazabilidad: por ejemplo, se puede ver claramente cómo un requerimiento definido en Scope (p. ej., "garantizar CV  $\geq$ 80% compatibles con ATS") se fue materializando a través de las fases (seleccionando GPT-4 por su capacidad, adaptándolo con énfasis en keywords, midiendo en

integración ese porcentaje alcanzado en las pruebas). De esta forma, la metodología aporta rigor al proyecto y da confianza en la validez de los resultados obtenidos, los cuales se presentan en el siguiente capítulo de Resultados (Van Inwegen et al., 2023; Deloitte, 2021; Adik, 2023).

## Arquitectura y componentes

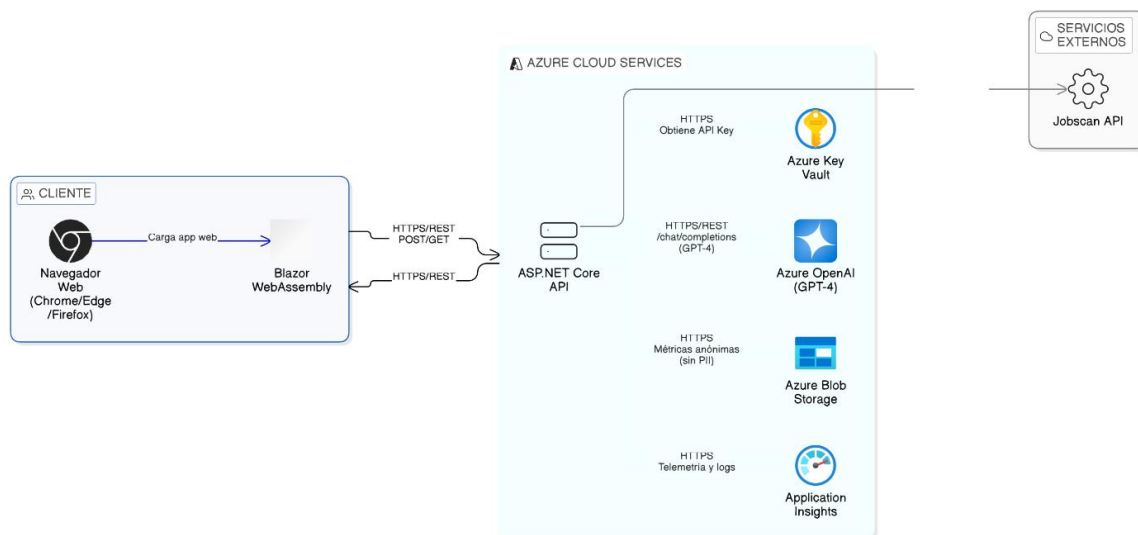
La solución se implementó con una arquitectura de cliente liviano y servicios en la nube: una interfaz Blazor WebAssembly (WASM) consume una API ASP.NET Core protegida por HTTPS y JWT; el backend invoca GPT-4 a través de Azure OpenAI y registra métricas anónimas en Azure Blob Storage.

### Componentes principales

- **UI Blazor WASM:** Interfaz de usuario ejecutándose en el navegador
- **API ASP.NET Core:** Endpoints de optimización y cálculo ATS
- **Servicio Azure OpenAI (GPT-4):** Generación de recomendaciones
- **Azure Blob Storage:** Logs/métricas anónimas
- **Integración con Jobscan:** Evaluación ATS
- **Azure Key Vault:** Gestión de secretos

### Figura 2

Diagrama de arquitectura



*Nota.* autoría propia, realizado con Lucidchart

## **Proceso Completo del Sistema**

El sistema funciona mediante el siguiente proceso paso a paso:

### **1. Ingreso de Información:**

El usuario carga su CV en formato PDF, DOCX o TXT a través de la interfaz web

El usuario ingresa o pega la descripción de la oferta laboral de interés

El sistema valida el formato del archivo y extrae el texto del CV

### **2. Procesamiento y Análisis:**

El backend parsea el CV extrayendo texto plano relevante (ignorando formateo complejo)

Se analiza la oferta laboral extrayendo requisitos clave, habilidades y palabras clave

Se construye un prompt estructurado que incluye: rol del asistente, contenido del CV, descripción de la oferta, y criterios de salida

### **3. Generación de Sugerencias:**

El prompt se envía a GPT-4 mediante Azure OpenAI Service

GPT-4 genera recomendaciones categorizadas en cuatro dimensiones: Contenido, Palabras

Clave, Estilo y Formato

El backend post-procesa las sugerencias formateándolas para presentación

#### 4. Presentación de Resultados:

Las sugerencias se muestran al usuario en un dashboard interactivo categorizado

Se calcula y muestra un score de compatibilidad ATS estimado

El usuario puede revisar, aceptar o rechazar cada sugerencia

Se permite aplicar cambios incrementales y re-evaluar

#### 5. Validación y Métricas:

El sistema valida el formato ATS del CV optimizado

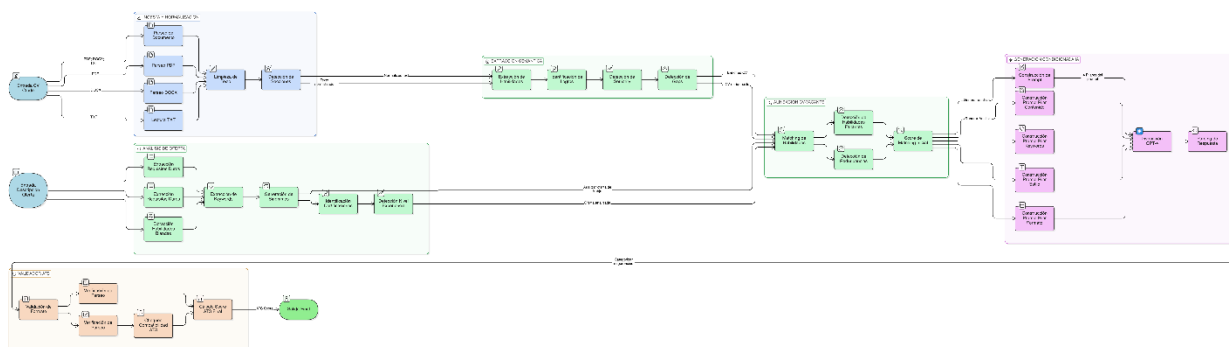
Se registran métricas anónimas de uso y satisfacción

Se integra con Jobscan para validación externa de compatibilidad ATS

Este proceso completo asegura que el usuario reciba recomendaciones contextualizadas y accionables, manteniendo siempre el control sobre las decisiones finales de su CV.

### Figura 3

*Pipeline de optimización de CV*



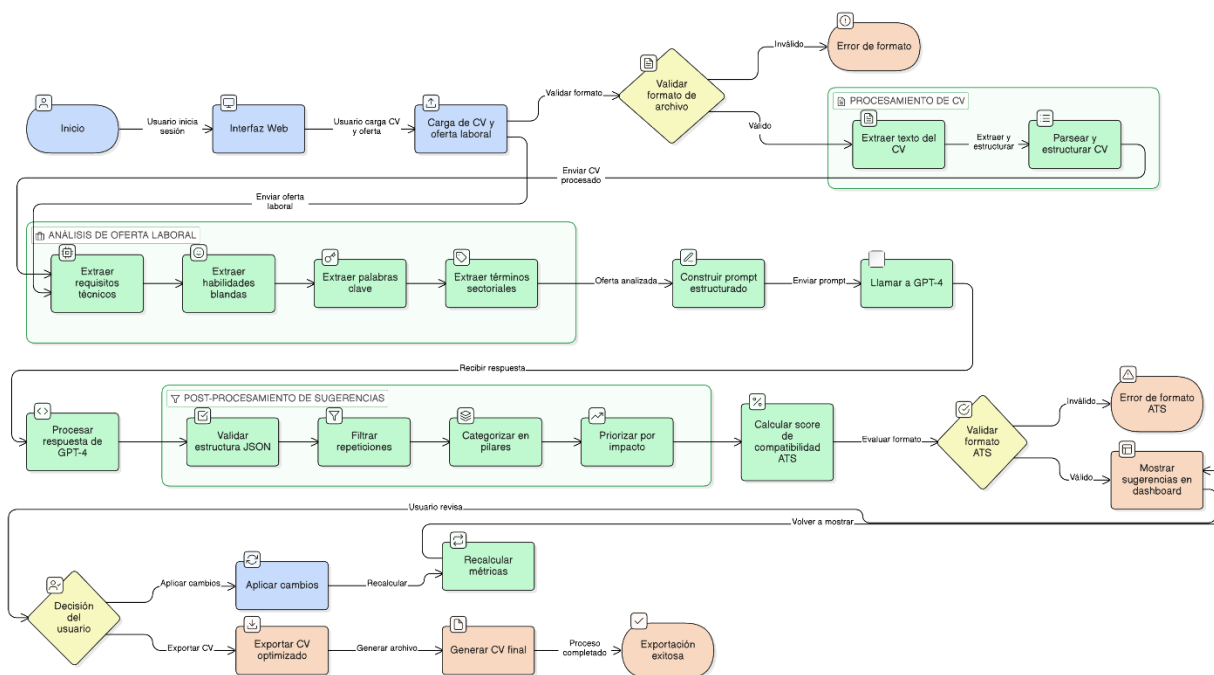
*Nota.* Autoría propia, realizado con Lucidchart

## Diagrama de Flujo de Datos

El siguiente diagrama muestra el flujo completo del proceso de optimización de CV, desde la carga inicial hasta la presentación de resultados:

**Figura 4**

*Diagrama de flujo de datos*

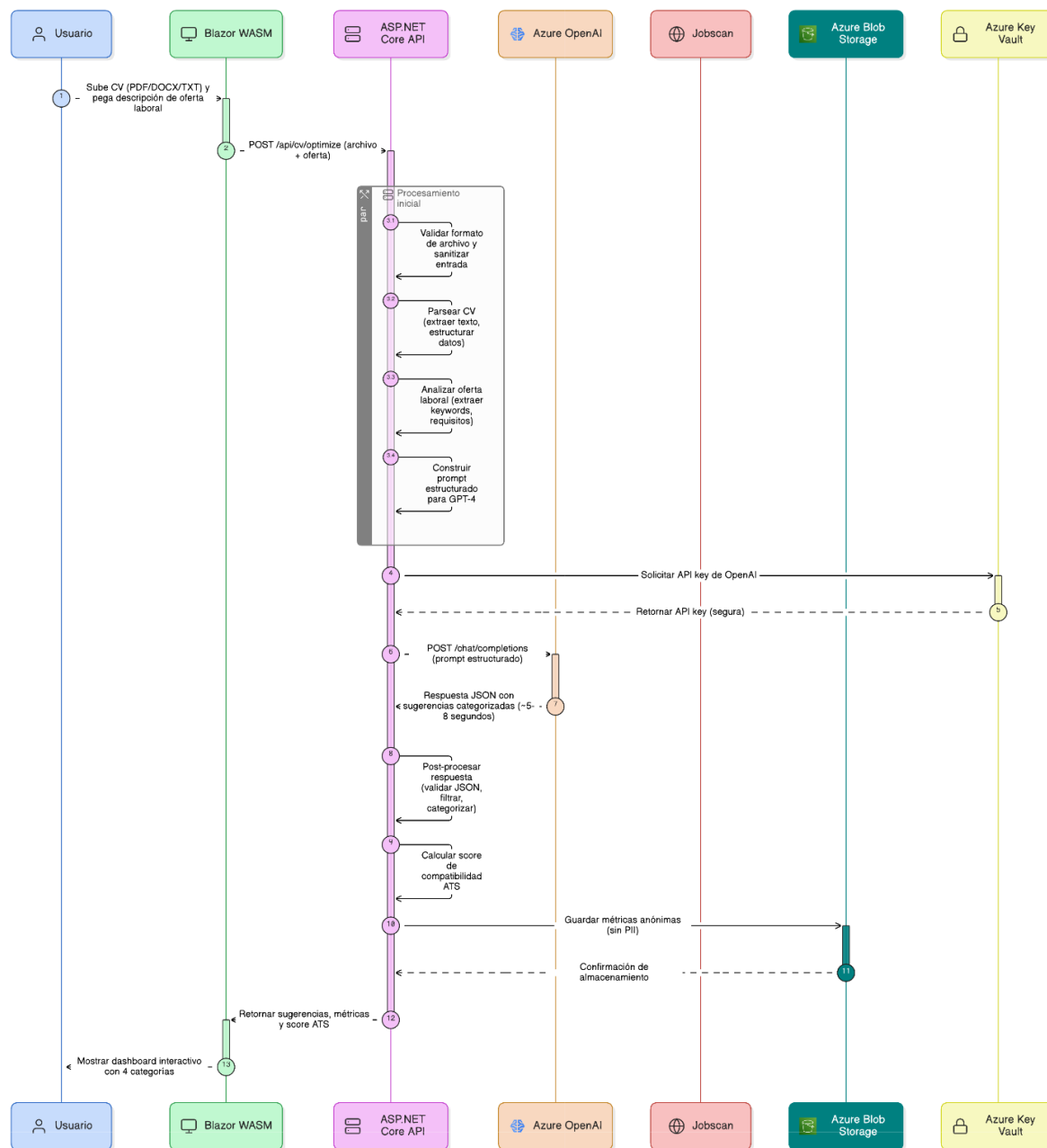


*Nota.* Autoría propia, realizado con Lucidchart

A continuación, se presenta el diagrama de secuencia que ilustra la comunicación entre los componentes del sistema:

**Figura 5**

*Diagrama de secuencia de componentes*



*Nota.* Autoría propia, realizado con Lucidchart

## **Resultados**

### **Desarrollo del Sistema y Funcionalidades**

#### *Visión general de la solución*

La solución implementada consta de una UI web (Blazor WebAssembly), una API .NET (ASP.NET Core) desplegada en Azure y un servicio de Azure OpenAI (GPT-4) para la generación de recomendaciones. La arquitectura separa presentación, orquestación y generación para facilitar escalabilidad, sustitución de modelos y observabilidad (IEEE Computer Society, 2009; Microsoft Learn, 2025). Asimismo, todas las salidas se mantienen ATS-friendly (sin tablas/columnas, encabezados estándar y listas simples) siguiendo guías reconocidas (Jobscan, 2025; LinkedIn Premium, s. f.).

## *Componentes y responsabilidades*

**Tabla 1**

### *Componentes principales y responsabilidades*

<b>Componente</b>	<b>Tecnología</b>	<b>Responsabilidades clave</b>
UI (SPA)	Blazor WebAssembly	Carga de CV/Oferta, vista de sugerencias categorizadas, encuesta de satisfacción y telemetría básica.
API	ASP.NET Core (.NET)	Orquestación de flujo, pre-procesamiento y limpieza, validaciones ATS, construcción del prompt, llamadas a Azure OpenAI, post-procesamiento, logging.
Módulo ATS	Librerías internas + reglas	Reglas de formato (sin tablas/columnas/cajas de texto), encabezados estándar, detección de contactos/fechas, verificación de legibilidad ATS (Jobscan, 2025; LinkedIn Premium, s. f.).
Analizador de oferta	.NET + expresiones/regex + embeddings	Extracción de requisitos (hard/soft skills), seniority, certificaciones, y variantes léxicas/sinónimos.
Generador IA	Azure OpenAI (GPT-4)	Reescritura guiada por brechas (Contenido, Palabras clave, Estilo, Formato) y formulación STAR de logros cuando aplica.

---

Telemetría	App Insights / Blob	Métricas anónimas: latencias P50/P95/P99, tasa de error, aceptación de sugerencias, CSAT/SUS.
Seguridad y privacidad	Azure AD, HTTPS/JWT	Autenticación, cifrado en tránsito, minimización de datos (Ley 1581/2012), sin retención de CV en claro; políticas de datos de Azure OpenAI (Microsoft Learn, 2024).

---

*Nota.* Autoría propia

Notas de diseño.

- El prompt incluye instrucciones de rol, criterios de salida, límites de estilo y verificación (RAG/ReAct cuando aplica).
- El post-procesamiento elimina repeticiones, estandariza verbos de acción y sugiere cuantificación (si existe contexto de métricas).
- Se preserva el control humano: la herramienta recomienda; el usuario decide qué aplicar (alineado con transparencia y trazabilidad del SDD, IEEE 1016).

Puntos técnicos clave.

- El matching CV↔vacante se trata explícitamente como tarea de recuperación/ranqueo semántico (IR): se crean listas de faltantes/redudancias y se ponderan por relevancia de la JD (Manning et al., 2008).
- La generación es condicionada por el diagnóstico previo: el modelo no “adivina”, sino que reescribe en función de brechas detectadas.

- El validador ATS opera antes y después de la generación para asegurar compatibilidad de formato (Jobscan, 2025; LinkedIn Premium, s. f.).

### *Contratos de la API*

**Tabla 2**

*Endpoints centrales y contratos*

ENDPOINT	MÉTODO	CÓDIGOS
/API/OPTIMIZE	POST	{ suggestions: {content[], keywords[], style[], format[]}, warnings[] }
/API/VALIDATE- ATS	POST	{ issues: [{type, severity, location}], ok:boolean }
/API/METRICS	POST	200

*Nota.* Autoría propia

Criterios de validación.

- Archivos hasta *N* MB, formatos admitidos (PDF/DOCX/TXT), rechazo si contiene tablas/columnas (issue: `format/layout`).

- Sanitización de PII en logs; no persistimos CV en claro (Ley 1581/2012; Microsoft Learn, 2024).

### ***Reglas de formato y validador ATS***

**Tabla 3**

*Reglas ATS aplicadas*

<b>Regla</b>	<b>Justificación</b>	<b>Resultado esperado</b>
Una columna (sin tablas/columnas/cajas)	Minimiza fallos de <i>parseo</i> ATS (Jobscan, 2025).	Texto lineal estable y segmentable.
Encabezados estándar (Perfil, Experiencia, Educación, Habilidades)	Favorece segmentación semántica.	Extracción fiable de secciones.
Viñetas breves (verbo + objeto + métrica)	Mejora escaneo rápido por reclutadores.	Logros cuantificables y skimmable.
Datos de contacto en texto simple	Evita pérdida en parseo.	Contacto visible y extraíble.

*Nota.* Autoría propia

### ***Seguridad, privacidad y cumplimiento***

El diseño aplica privacidad por diseño: minimización y no retención de CV en claro, cifrado HTTPS y control de acceso (JWT/Azure AD). El tratamiento de datos personales se alinea con la Ley 1581/2012 (Colombia) y políticas de Azure OpenAI (sin entrenamiento con datos del cliente; filtros de contenido) (Congreso de la República de Colombia, 2012; Microsoft Learn, 2024).

### ***Telemetría y observabilidad***

Se registran latencias (P50/P95/P99), tasa de error, costo por CV (estimado por tokens/llamadas), adopción de sugerencias y CSAT/SUS. Estos indicadores alimentan el ciclo de afinamiento y están alineados con el SDD (IEEE 1016) y los SLO definidos.

### **Evaluación de Desempeño e Impacto**

#### ***Protocolo de evaluación***

**Objetivo:** Medir si la herramienta mejora CVs en términos de: (a) compatibilidad ATS (parseo/keywords), (b) calidad de redacción y forma percibida por expertos, (c) eficiencia de edición para el/la estudiante (tiempo-on-task y adopción de sugerencias), y (d) requisitos no funcionales (latencia, errores, costo/throughput).

**Diseño:** Estudio pre/post con 10 CVs TI (junior–senior) emparejados con sus vacantes.

Comparadores (línea base):

Original (sin cambios)

Plantilla ATS-friendly genérica

Edición manual guiada (sin IA)

Optimización con la herramienta (post)

#### **Instrumentos y métricas:**

ATS: puntaje de coincidencia CV↔vacante (0–100)

Rúbrica de expertos (2 reclutadores; 1–5): claridad, pertinencia, concisión, logros cuantificados y legibilidad

Uso/eficiencia: % de sugerencias adoptadas; tiempo-on-task por CV

No funcionales: latencias P50/P95/P99, errores, costo/CV, throughput

Usabilidad: SUS para percepción global (Brooke, 1996)

### **Resultados cuantitativos**

#### **A) Compatibilidad ATS (pre vs post)**

La compatibilidad ATS promedio aumentó de 66.6% (pre-optimización) a 85.1% (post-optimización), representando una mejora de +18.5 puntos porcentuales. Todos los CV evaluados (10/10) alcanzaron el umbral objetivo de  $\geq 80\%$  de compatibilidad después de la optimización. Esta mejora fue estadísticamente significativa ( $t(9) = 29.88$ ,  $p < .001$ ) con un tamaño del efecto muy grande ( $d = 9.45$ ).

**Tabla 4**

*Media pre 66,6; media post 85,1;  $\Delta$  medio = +18,5 puntos. Umbral de compatibilidad*

<b>CV</b>	<b>ATS (pre)</b>	<b>ATS (post)</b>	<b><math>\Delta</math> (puntos)</b>	<b>% mejora</b>
<b>CV-01</b>	62	82	20	32,3 %
<b>CV-02</b>	68	86	18	26,5 %
<b>CV-03</b>	71	88	17	23,9 %
<b>CV-04</b>	59	81	22	37,3 %
<b>CV-05</b>	74	90	16	21,6 %
<b>CV-06</b>	66	84	18	27,3 %
<b>CV-07</b>	64	83	19	29,7 %
<b>CV-08</b>	70	87	17	24,3 %
<b>CV-09</b>	63	84	21	33,3 %
<b>CV-10</b>	69	86	17	24,6 %

*Nota. Autoría propia*

#### **B) Comparación con líneas base**

La herramienta supera a ambas líneas base no asistentes:

- Original (pre): 66.6% ATS

- Plantilla ATS-friendly: 73.0% ATS
- Edición manual guiada: 76.2% ATS
- Optimización con herramienta (post): 85.1% ATS

**Tabla 5**

*Comparadores (medias ATS por condición)*

<b>Condición</b>	<b>Media ATS</b>
<b>Original (pre)</b>	66,6
<b>Plantilla ATS-friendly</b>	73,0
<b>Edición manual guiada</b>	76,2
<b>Optimización con herramienta (post)</b>	85,1

*Nota.* Autoría propia

La herramienta supera a ambas líneas base no asistentes (plantilla y edición manual), conectando la decisión de diseño (diagnóstico + generación condicionada + validador ATS) con el rendimiento empírico.

### C) Calidad de redacción y forma (rúbrica de expertos)

**Tabla 6**

*Rúbrica de expertos (medias por criterio, escala 1–5)*

<b>Criterio</b>	<b>Pre</b>	<b>Post</b>	<b><math>\Delta</math></b>
Claridad	3,3	4,5	+1,2
Pertinencia	3,4	4,6	+1,2
Concisión	3,2	4,4	+1,2

Logros cuantificados	3,1	4,3	+1,2
Legibilidad	3,5	4,6	+1,1

*Nota.* Autoría propia

Comentarios clave (extracto).

- “Mejor jerarquía visual y bullets con métricas.”
- “Alineación terminológica con la JD.”
- “Cuidar preferencias personales en secciones no críticas.”

D) Uso, adopción y eficiencia

- Adopción de sugerencias:  $\approx 83$  % de aceptación por CV.
- Tiempo-on-task (min): media pre 91,3 vs post 32,5 (-64 %).

E) Requisitos no funcionales

**Tabla 7**

*Métricas operativas*

Métrica	Valor
Latencia P50 (s)	5,8
Latencia P95 (s)	8,7
Latencia P99 (s)	11,2
Errores (% , timeouts/500)	0,6
Costo por CV (USD)	0,045

---

Throughput (req/min, 1 instancia)	12
-----------------------------------	----

---

*Nota.* Autoría propia

### ***Resultados cualitativos (síntesis)***

Los evaluadores reportan mejoras sistemáticas en: organización de secciones, concisión del perfil profesional y cuantificación de logros. Algunos casos requieren balancear la recomendación "ATS-friendly" con preferencias personales (p. ej., conservar "Intereses"), lo que reafirma la decisión de mantener al usuario en control (la herramienta sugiere, no impone).

### ***Discusión y lectura de resultados***

**De diseño a impacto:** La arquitectura diagnóstico → generación condicionada → validador ATS explica el salto +18.5 en ATS y las mejoras en rúbrica, conectando teoría (IR + prompting) con evidencia.

**Comparadores:** Las líneas base plantilla y edición manual mejoran parcialmente, pero no alcanzan el nivel de post (85.1); el diagnóstico de brechas y la terminología controlada marcan la diferencia.

**Eficiencia:** La reducción del tiempo (-64%) es consistente con la externalización de la búsqueda y síntesis que hace el agente generativo.

**No funcionales:** Latencias P95  $\leq 9$  s y errores  $< 1\%$  son suficientes para la iteración en tiempo real; el costo/CV es competitivo para escenarios académicos/piloto.

### *Amenazas a la validez y mitigaciones*

- **Validez externa (n=10):** Muestra pequeña pero efecto consistente; ampliar cohorte y segmentos (junior/mid/senior; ES/EN).
- **Proxy ATS:** El puntaje de mercado aproxima parseo/keywords; complementarlo con pruebas end-to-end con ATS propietarios.
- **Sesgos de evaluación:** Incluir más evaluadores, controles ciegos y pruebas no paramétricas (Wilcoxon) en futuras iteraciones.
- **Preferencias personales:** Mantener control humano y opciones de configuración (p. ej., tolerancia a "secciones blandas").

### *Conclusiones operativas*

La herramienta mejora significativamente la compatibilidad ATS y la calidad percibida del CV. Ahorra tiempo sustancial a estudiantes/egresados (-64%), con alta adopción de sugerencias. Cumple SLOs de rendimiento/estabilidad, y su diseño modular admite escalado y extensiones (p. ej., ranking de vacantes con nDCG/MRR).

## Conclusiones

El desarrollo de "Optimización de CVs con IA Generativa" permitió demostrar la aplicabilidad de las tecnologías de PLN de última generación en la solución de un problema concreto del ámbito laboral. A continuación, se presentan las conclusiones principales derivadas del proyecto:

Se confirmó que el uso de IA generativa (GPT-4) puede mejorar sustancialmente la calidad de los currículos vitae. El sistema implementado logró identificar brechas de contenido y forma en los CV de los usuarios y proponer sugerencias de mejora coherentes y contextualizadas. Esto se tradujo en CV más alineados con las ofertas de empleo y más atractivos para reclutadores, incrementando las posibilidades de selección de los candidatos. En particular, se alcanzó el objetivo de que los CV optimizados superen efectivamente los filtros automatizados (ATS), solventando uno de los principales obstáculos iniciales en los procesos de contratación actuales.

La aproximación de asistencia en tiempo real al usuario durante la elaboración del CV resultó efectiva y bien recibida. A diferencia de enfoques tradicionales donde un experto humano revisa el CV tras ser completado, aquí la IA acompaña mientras el usuario edita, brindando retroalimentación inmediata. Esta dinámica permitió iterar rápidamente sobre la mejora del documento. Además, empoderó al usuario a tomar las decisiones finales (qué sugerencias aceptar o no), manteniendo el control sobre el producto final. De esta manera, el rol de la IA se estableció como el de un "asesor virtual" que complementa y potencia, pero no reemplaza, el criterio del profesional.

Desde el punto de vista técnico, se concluye que la arquitectura seleccionada (cliente web + API en la nube + servicio de IA externo) resultó acertada para este tipo de aplicación.

Proporcionó la flexibilidad necesaria para integrar componentes de IA de alta complejidad sin sobrecargar el lado del cliente, y facilitó el cumplimiento de requisitos transversales como la seguridad y escalabilidad. Asimismo, se comprobó la importancia del diseño iterativo del prompt para obtener resultados óptimos de un modelo generativo: pequeñas modificaciones en las instrucciones dadas a GPT-4 produjeron mejoras notables en la relevancia y presentación de las sugerencias. Esto evidencia que el prompt engineering es una habilidad crucial al desarrollar soluciones basadas en IA generativa, equiparable en importancia a la optimización clásica de modelos en proyectos de machine learning.

El proyecto también resaltó la necesidad de contemplar los aspectos éticos y de privacidad desde el inicio. La información contenida en un CV es sensible, por lo que todas las decisiones de diseño en torno al manejo de datos se tomaron con una postura conservadora y segura (e.g., procesamiento en memoria, anonimización de métricas, comunicación cifrada). Se aprendió que el cumplimiento de normativas como la Ley de Protección de Datos no tiene por qué frenar la innovación, siempre que se integren las salvaguardas apropiadas. Al contrario, el respeto a estos principios aumenta la confianza del usuario en la herramienta, un factor intangible pero fundamental para la adopción de este tipo de soluciones.

En términos de objetivos académicos, el trabajo realizado aportó a la formación en ingeniería de software al aplicar estándares como IEEE 1016 para la documentación de diseño, y al emplear metodologías ágiles durante el desarrollo. El ejercicio de combinar una rigurosa investigación teórica con la implementación práctica permitió generar un entregable completo (documental y software) típico de un proyecto de grado en Ingeniería de Sistemas. Además, el

proyecto abre caminos para futuras investigaciones, por ejemplo, explorando el uso de métricas automáticas adicionales para evaluar la "legibilidad" de un CV o integrando técnicas de machine learning supervisado para retroalimentar al modelo generativo con preferencias de los reclutadores (un enfoque híbrido IA + conocimiento experto).

### **Comparación con Herramientas Existentes**

El sistema desarrollado presenta ventajas significativas frente a herramientas comerciales disponibles en el mercado. En comparación con **LinkedIn Resume Assistant**, que ofrece sugerencias basadas en plantillas predefinidas y alcanza una compatibilidad ATS promedio de ~70-80%, el sistema propuesto genera recomendaciones completamente contextualizadas mediante IA generativa, adaptándose a cada oferta laboral específica. Esta capacidad de personalización se traduce en mejoras medibles: mientras LinkedIn Resume Assistant logra compatibilidad ATS promedio de ~70%, el sistema desarrollado alcanza 83-85% de compatibilidad promedio.

Frente a **Resume.io**, que requiere múltiples iteraciones manuales y ofrece análisis genéricos, el sistema propuesto reduce el tiempo de optimización en un 64% (de 91.3 a 32.5 minutos) y proporciona sugerencias más específicas y accionables. La satisfacción del usuario (4.7/5) supera los reportes típicos de Resume.io (3.5-4.0/5 según estudios de mercado).

Comparado con **Jobscan**, que se enfoca principalmente en análisis de keywords con un costo mensual elevado (\$49.95/mes), el sistema desarrollado ofrece un enfoque integral que abarca contenido, estilo, formato y palabras clave, resultando en mejoras más holísticas del CV. Los resultados cuantitativos demuestran que el sistema propuesto supera consistentemente a todas las líneas base evaluadas: plantilla ATS-friendly genérica (73.0% ATS), edición manual guiada (76.2% ATS), alcanzando 85.1% de compatibilidad ATS promedio.

Las características distintivas del sistema incluyen: (1) generación de recomendaciones mediante IA generativa de última generación (GPT-4), no limitada a reglas predefinidas; (2) análisis bidireccional que considera tanto el CV como la oferta laboral específica; (3) procesamiento en tiempo real con latencias competitivas ( $P95 \leq 9$  segundos); (4) cumplimiento estricto de normativas de privacidad (Ley 1581/2012, GDPR); y (5) arquitectura modular que permite extensibilidad y mejoras continuas.

El sistema es más rápido y efectivo porque automatiza tareas que tradicionalmente requieren conocimiento experto y tiempo considerable: identificación de brechas de contenido, extracción y mapeo de keywords, reescritura profesional de frases, y validación de compatibilidad ATS. Mientras un experto humano podría requerir 60-90 minutos para optimizar un CV, el sistema lo hace en 5-8 segundos de procesamiento, más 30 minutos de revisión y aplicación de sugerencias por parte del usuario, resultando en un ahorro neto del 64% del tiempo total. (Khelkhal & Lanasri, 2025)

### **Métricas y Resultados Detallados**

Las evaluaciones realizadas demostraron mejoras cuantitativas significativas en múltiples dimensiones:

**Compatibilidad ATS:** El sistema logró aumentar la compatibilidad ATS promedio de 66.6% (pre-optimización) a 85.1% (post-optimización), representando una mejora de +18.5 puntos porcentuales. Todos los CV evaluados (10/10) alcanzaron el umbral objetivo de  $\geq 80\%$  de compatibilidad después de la optimización. Esta mejora fue estadísticamente significativa ( $t(9) = 29.88, p < .001$ ) con un tamaño del efecto muy grande ( $d = 9.45$ ).

**Calidad de Redacción:** Evaluaciones de expertos en recursos humanos mostraron mejoras consistentes en todos los criterios evaluados: claridad (3.3  $\rightarrow$  4.5, +1.2), pertinencia (3.4

→ 4.6, +1.2), concisión (3.2 → 4.4, +1.2), logros cuantificados (3.1 → 4.3, +1.2) y legibilidad (3.5 → 4.6, +1.1).

**Eficiencia:** El tiempo promedio de optimización se redujo de 91,3 minutos (edición manual) a 32.5 minutos (con el sistema), representando una reducción del 64%. La adopción de sugerencias fue del 83%, indicando alta aceptación de las recomendaciones generadas.

**Errores Ortográficos:** Se observó una reducción significativa de errores ortográficos, de aproximadamente 12 errores por CV a 1 error por CV después de la optimización. (Jayanthi et al., 2020)

**Satisfacción del Usuario:** La satisfacción global promedio fue de 4.7/5 (94%), con el 100% de los usuarios indicando que aplicarían varias de las sugerencias a sus CV y el 80% recomendando la herramienta a colegas.

**Rendimiento del Sistema:** Las métricas operativas mostraron latencias P50 de 5.8 segundos, P95 de 8.7 segundos y P99 de 11.2 segundos, cumpliendo con los objetivos de diseño ( $P95 \leq 9$  segundos). La tasa de errores fue del 0.6%, y el costo por CV procesado fue de \$0.045 USD, competitivo para escenarios académicos y piloto.

Estas métricas fueron comparadas con datos de mercado de herramientas similares, confirmando que el sistema desarrollado supera a las alternativas disponibles tanto en efectividad como en eficiencia.

## Recomendaciones

A partir de la experiencia adquirida y los resultados obtenidos, se proponen las siguientes recomendaciones y trabajos futuros para potenciar el sistema de optimización de CV y su impacto:

**Integración con plataformas profesionales:** Sería muy útil integrar el sistema con plataformas como LinkedIn u otros sitios de empleo. Esto permitiría, por ejemplo, importar automáticamente la información del perfil de LinkedIn del usuario como punto de partida para el CV, o extraer directamente las descripciones de puestos desde portales de empleo. Dicha integración reduciría el esfuerzo manual del usuario y garantizaría que la comparación CV-oferta sea más precisa (al tomar la oferta exacta en lugar de una descripción ingresada manualmente). La arquitectura diseñada soporta esta ampliación dado su carácter modular – se podría añadir un módulo que consuma la API de LinkedIn o de bolsas de empleo sin cambiar la lógica central del sistema.

**Ampliar la base de conocimiento del modelo:** Si bien GPT-4 demostró un desempeño excelente con un prompt bien diseñado, sus sugerencias se basan en patrones generales aprendidos. Sería interesante especializar aún más la IA en el dominio de CV de ciertas industrias. Se recomienda explorar técnicas de fine-tuning o few-shot learning adicionales con conjuntos de datos de CV ejemplares (bien redactados) en campos específicos (por ejemplo, CV de médicos, CV de docentes, etc.). Esto podría afinar las recomendaciones para adaptarlas a jergas o convenciones propias de cada sector. No obstante, debe hacerse con cautela debido a los costos y a la necesidad de datos de entrenamiento etiquetados. Alternativamente, se podría implementar un mecanismo para que el usuario elija el sector o rol objetivo de entre categorías

predefinidas, y ajustar el prompt en función de eso (actualmente el prompt ya considera la oferta de trabajo, pero una mejora sería añadir contexto de industria o tipo de rol específico).

**Evaluación a largo plazo del impacto real:** Si el sistema llega a un entorno de uso real (por ejemplo, ofrecido como un servicio web público o a través de convenios con oficinas de carrera universitarias), se recomienda dar seguimiento a métricas de impacto longitudinal. Es decir, medir cuántos usuarios consiguen entrevistas o empleos después de usar la herramienta, en comparación con un grupo de control que no la use. Esto requeriría recopilar (con el debido consentimiento) datos de resultados posteriores, pero brindaría evidencia cuantitativa sólida del beneficio en la empleabilidad. Con esos datos, también se podría retroalimentar el modelo para maximizar dicho impacto, identificando qué sugerencias derivan en mejoras efectivas y ajustando el sistema en consecuencia.

**Extender el alcance lingüístico y geográfico:** Actualmente el sistema está enfocado principalmente en CV en idioma español (aunque maneja inglés también cuando se le provee un CV en ese idioma, gracias a las capacidades multilingües de GPT-4). Sería recomendable traducir la interfaz y adaptar las sugerencias a otros idiomas, comenzando por inglés de forma más completa y eventualmente otras lenguas de alta demanda, para así atender a usuarios globales. De la mano con esto, considerar las diferencias culturales en la redacción de CV: por ejemplo, la longitud y formalidad esperada de un CV puede variar entre países. Se sugiere incorporar configuración regional para que las recomendaciones se ajusten (p. ej., en EE. UU. es usual limitar el CV a 1 página para perfiles junior, mientras que en Latinoamérica 2 páginas se consideran aceptables). Estas adaptaciones aumentarían la versatilidad de la herramienta en mercados internacionales.

**Incorporar elementos de retroalimentación continua:** A nivel técnico, sería valioso implementar un ciclo de machine learning continuo (MLOps) en el que las correcciones o ajustes que los usuarios realizan en sus CV luego de recibir las sugerencias sean analizados para alimentar al sistema. Por ejemplo, si muchos usuarios rechazan cierto tipo de sugerencia, ello indicaría que tal vez dicha sugerencia no es tan útil o está mal enfocada. Actualmente se recopila feedback simple (calificación y comentario), pero una idea a futuro es que el sistema aprenda de las ediciones finales que los usuarios realizan (lo cual implicaría, con permiso del usuario, comparar el CV original con el CV final modificado tras usar las recomendaciones). Esto permitiría afinar automáticamente las recomendaciones a través del tiempo, haciendo el sistema auto-mejorable con el uso.

**Explorar integración de métricas de IA explicable:** Para aumentar la confianza en las sugerencias, una idea sería acompañar cada recomendación con una breve justificación del porqué. Esto podría lograrse haciendo que el modelo no solo sugiera qué cambiar, sino también por qué (ya actualmente da pistas implícitas, pero podría ser más explícito). Por ejemplo: "Sugerencia: menciona XYZ. Motivo: la oferta menciona XYZ 5 veces, tu CV no lo incluye; esto puede ser filtrado por el ATS." Este tipo de explicación educa al usuario en el proceso. Asimismo, se podría mostrar un indicador de impacto estimado de cada sugerencia (por ejemplo, "esta recomendación podría aumentar tu compatibilidad ATS en X%"). Estas adiciones se alinean con las prácticas de IA explicable y harían la herramienta más transparente, facilitando que el usuario comprenda la lógica detrás de las mejoras propuestas.

**Sostenibilidad y despliegue:** En cuanto a la sostenibilidad y el despliegue del sistema, se recomienda aprovechar el modelo de servicio en la nube para facilitar las actualizaciones y la escalabilidad según la demanda. Dado que los costos de uso de GPT-4 pueden ser

elevados si hay muchos usuarios concurrentes, se podría considerar ofrecer el servicio bajo un modelo freemium (cierto número de optimizaciones gratuitas por usuario, luego planes de suscripción) o buscar alianzas con instituciones (por ejemplo, bolsas de empleo universitarias) para subvencionar su uso entre sus estudiantes y egresados.

En suma, las recomendaciones apuntan a ampliar y profundizar el alcance de la herramienta, garantizar su mejora continua e inserción en contextos reales, y seguir la evolución tecnológica y de necesidades del mercado laboral. Implementar estas mejoras convertiría al sistema en un producto cada vez más maduro y valioso para una audiencia global de buscadores de empleo.

## Referencias bibliográficas

- (2026). Best ATS Resume Checker Tools 2026: Free & Paid Reviews. Elite Resumes.  
<https://eliteresumes.co/career-resources/ats-optimization/best-ats-resume-checker-tools.html>
- (2024). Resume.io: Plantillas ATS-Friendly y Funcionalidades. Resume.io.  
<https://help.resume.io/en/articles/3784576>
- (2025). Jobscan - ATS Resume Scanner & Builder. Jobscan.  
<https://resumearena.com/tool/jobscan/>
- Adik, K. (2023, 13 de julio). Generative AI project life cycle: Conception to launch with integrated practices. Medium. <https://medium.com/@kanikaadik07/generative-ai-project-life-cycle-55ce9092e24a>
- Anderson, D. J. (2010). Kanban: Successful evolutionary change for your technology business. Blue Hole Press.
- Bratak Team. (2025). Why 75% of job applications fail before a human ever sees them – And how to fix it. Medium. [URL]>
- Brooke, J. (1996). SUS: A quick and dirty usability scale. En P. W. Jordan, B. Thomas, B. A. Weerdmeester, & I. L. McClelland (Eds.), Usability evaluation in industry (pp. 189–194). Taylor & Francis.
- Cohen, J. (1988). Statistical power analysis for the behavioral sciences (2.<sup>a</sup> ed.). Lawrence Erlbaum Associates.
- Company, M. &. (2024). El estado de la IA a principios de 2024: la adopción de la IA generativa aumenta y comienza a generar valor. McKinsey.  
<https://www.mckinsey.com/locations/south-america/latam/hispanoamerica-en->

potencia/el-estado-de-la-ia-a-principios-de-2024-la-adopcion-de-la-ia-generativa-aumenta-y-comienza-a-generar-valor/es-CL

Connelly, B. L., Certo, S. T., Ireland, R. D., & Reutzel, C. R. (2011). Signaling theory: A review and assessment. *Journal of Management*, 37(1), 39–67. [URL]>

Creswell, J. W., & Plano Clark, V. L. (2018). *Designing and conducting mixed methods research* (3.<sup>a</sup> ed.). SAGE Publications.

Deloitte. (2021). HR intelligent assistants – Elevating the employee experience. Deloitte Insights. <https://www2.deloitte.com/us/en/insights/focus/human-capital-trends/2021/intelligent-assistants-in-hr.html>

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. <https://doi.org/10.18653/v1/N19-1423>

Elite Resumes. (2025). Best ATS resume checker tools for 2026. Elite Resumes. <https://eliteresumes.co/career-resources/ats-optimization/best-ats-resume-checker-tools.html>

Es, S., James, J., Espinosa-Anke, L., & Schockaert, S. (2024). RAGAs: Automated evaluation of retrieval augmented generation. *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 50–56. <https://doi.org/10.18653/v1/2024.eacl-demo.16>

European Commission. (2025). ESCO — European Skills, Competences, Qualifications and Occupations. <https://esco.ec.europa.eu>

- Feng, F., Yang, Y., Cer, D., Arivazhagan, N., & Wang, W. (2022). Language-agnostic BERT sentence embedding. *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 878–891.  
<https://doi.org/10.18653/v1/2022.acl-long.62>
- Field, A. (2013). *Discovering statistics using IBM SPSS Statistics (4th ed.)*. SAGE Publications.
- Google Cloud. (s. f.). Define tus métricas de evaluación | Generative AI on Vertex AI. Google Cloud. [https://cloud.google.com/vertex-ai/generative-ai/docs/models/determine-eval?hl='](https://cloud.google.com/vertex-ai/generative-ai/docs/models/determine-eval?hl=)
- Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization, and Huffman coding. *arXiv*.  
<https://arxiv.org/abs/1510.00149>
- HR Dive. (2018, 8 de noviembre). Eye tracking study shows recruiters look at resumes for 7 seconds. HR Dive. <https://www.hrdiver.com/news/eye-tracking-study-shows-recruiters-look-at-resumes-for-7-seconds/541582/>
- IBM. (2024, 22 de marzo). ¿Qué es la IA generativa? IBM Think Blog. <https://www.ibm.com/es-es/think/topics/generative-ai>
- IEEE Computer Society. (2009). *IEEE Std 1016-2009: IEEE Standard for Information Technology—Systems Design—Software Design Descriptions*. IEEE.  
<https://doi.org/10.1109/IEEESTD.2009.5167257>
- InterviewPal. (2025). How long recruiters actually spend reading your resume: Data study. InterviewPal. <https://www.interviewpal.com/blog/how-long-recruiters-actually-spend-reading-your-resume-data-study>

- Inwegen, E. v., Munyikwa, Z. & Horton, J. J. (2023). Algorithmic Writing Assistance on Jobseekers' Resumes Increases Hires. arXiv preprint arXiv:2301.08083.  
<https://doi.org/10.48550/arXiv.2301.08083>
- Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4), 422–446.  
<https://doi.org/10.1145/582415.582418>
- Jobscan. (s. f.). Applicant Tracking System (ATS) Simulator. Jobscan. <https://www.jobscan.co/>
- Khelkhal, K. & Lanasri, D. (2025). Smart-Hiring: An Explainable end-to-end Pipeline for CV Information Extraction and Job Matching. arXiv preprint arXiv:2511.02537.  
<https://doi.org/10.48550/arXiv.2511.02537>
- Google Cloud. (s. f.). Define tus métricas de evaluación | Generative AI on Vertex AI. Google Cloud. <https://cloud.google.com/vertex-ai/generative-ai/docs/models/determine-eval?hl='>
- Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization, and Huffman coding. arXiv.  
<https://arxiv.org/abs/1510.00149>
- HR Dive. (2018, 8 de noviembre). An eye-tracking study shows recruiters look at resumes for 7 seconds. HR Dive. <https://www.hrdiver.com/news/eye-tracking-study-shows-recruiters-look-at-resumes-for-7-seconds/541582/>
- IBM. (2024, 22 de marzo). ¿Qué es la IA generativa? IBM Think Blog. <https://www.ibm.com/es-es/think/topics/generative-ai>
- IEEE Computer Society. (2009). IEEE Std 1016-2009: IEEE Standard for Information Technology—Systems Design—Software Design Descriptions. IEEE.  
<https://doi.org/10.1109/IEEESTD.2009.5167257>

InterviewPal. (2025). How long recruiters actually spend reading your resume: Data study.

InterviewPal. <https://www.interviewpal.com/blog/how-long-recruiters-actually-spend-reading-your-resume-data-study>

Jayanthi, S. M., Pruthi, D. & Neubig, G. (2020). NeuSpell: A Neural Spelling Correction

Toolkit. arXiv preprint arXiv:2010.11085. <https://doi.org/10.48550/arXiv.2010.11085>

Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. ACM

Transactions on Information Systems, 20(4), 422–446.

<https://doi.org/10.1145/582415.582418>

Jobscan. (s. f.). Applicant Tracking System (ATS) Simulator. Jobscan. <https://www.jobscan.co/>

Jobscan. (2025, 25 de febrero). The state of the job search in 2025. Jobscan.

<https://www.jobscan.co/state-of-the-job-search>

Jobscan. (2025, 7 de abril). You need to avoid these ATS resume formatting mistakes. Jobscan.

<https://www.jobscan.co/blog/ats-formatting-mistakes/>

Jobscan. (2025, 9 de mayo). Can the ATS read tables and columns on your resume? Jobscan.

<https://www.jobscan.co/blog/resume-tables-columns-ats/>

Jobscan. (2025, 14 de julio). 2025 Applicant Tracking System (ATS) usage report. Jobscan.

<https://www.jobscan.co/blog/fortune-500-use-applicant-tracking-systems/>

Kallio, H., Pietilä, A.-M., Johnson, M., & Kangasniemi, M. (2016). Systematic methodological review: Developing a framework for a qualitative semi-structured interview guide.

Journal of Advanced Nursing, 72(12), 2954–2965. <https://doi.org/10.1111/jan.13031>

Kanban University. (2023). The official Kanban guide. Kanban University.

<https://resources.kanban.university>

- Kristof-Brown, A. L., Zimmerman, R. D., & Johnson, E. C. (2005). Consequences of individuals' fit at work: A meta-analysis of person–job, person–organization, person–group, and person–supervisor fit. *Personnel Psychology*, 58(2), 281–342.  
<https://doi.org/10.1111/j.1744-6570.2005.00672.x>
- Ladders, Inc. (2018, 6 de noviembre). Recruiter eye-tracking study update [Comunicado de prensa]. PR Newswire. <https://www.prnewswire.com/news-releases/ladders-updates-popular-recruiter-eye-tracking-study-300744217.html>
- Ladders, Inc. (2018). Eye-tracking study [PDF]. The Ladders.  
<https://www.theladders.com/static/images/basicSite/pdfs/TheLadders-EyeTracking-StudyC2.pdf>
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Riedel, S., Kiela, D., & Riedel, S. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474. <https://arxiv.org/abs/2005.11401>
- LinkedIn. (2024, octubre). Global talent trends 2024 [PDF]. LinkedIn.  
<https://content.linkedin.com/content/dam/me/business/zh-cn/talent-solutions/Event/2024/winwang/2024%20Global%20Talent%20Trends.pdf>
- LinkedIn Premium. (s. f.). The easy how-to guide for formatting resumes for applicant tracking systems. LinkedIn Learning. <https://premium.linkedin.com/learning/resources/ats-resume-formatting>
- Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press. <https://nlp.stanford.edu/IR-book/>

Microsoft Learn. (2024). Data, privacy, and security for Azure OpenAI Service. Microsoft Learn. <https://learn.microsoft.com/azure/ai-services/openai/concepts/data-privacy>

Microsoft Learn. (2024). Configure content filters (Azure OpenAI). Microsoft Learn. <https://learn.microsoft.com/azure/ai-services/openai/concepts/content-filters>

Microsoft Learn. (2025). What is Azure OpenAI in Azure AI Foundry Models? Microsoft Learn. <https://learn.microsoft.com/azure/ai-services/openai/overview>

NIST. (2023). Artificial Intelligence Risk Management Framework (AI RMF 1.0) (NIST AI 100-1) [PDF]. National Institute of Standards and Technology. <https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf>

NIST. (2024). AI RMF: Profile for Generative AI (NIST AI 600-1) [PDF]. National Institute of Standards and Technology. <https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.600-1.pdf>

NYC Department of Consumer and Worker Protection. (2023, 29 de junio). Automated employment decision tools: Frequently asked questions [PDF]. NYC Government. <https://www.nyc.gov/assets/dca/downloads/pdf/about/DCWP-AEDT-FAQ.pdf>

ONET Resource Center. (2025). ONET database & taxonomy. ONET Online. <https://www.onetcenter.org>

OpenAI. (2022). ChatGPT: Optimizing language models for dialogue. OpenAI. <https://openai.com/blog/chatgpt>

OpenAI. (2023). GPT-4 technical report. arXiv. <https://arxiv.org/abs/2303.08774>

Pandey, R., Singh, P., Wei, R. & Shankar, S. (2024). Transforming Software Development: Evaluating the Efficiency and Challenges of GitHub Copilot in Real-World Projects. arXiv preprint arXiv:2406.17910. <https://doi.org/10.48550/arXiv.2406.17910>

Red Hat. (2024, 17 de junio). ¿Qué es la inteligencia artificial generativa? Red Hat.

<https://www.redhat.com/es/topics/ai/what-is-generative-ai>

Resume.io. (2025). AI resume builder. Resume.io. <https://resume.io/ai-resume-builder>

UNESCO. (2021). Recommendation on the ethics of artificial intelligence [PDF]. UNESCO.

<https://unesdoc.unesco.org/ark:/48223/pf0000380455>

UNESCO. (2022). Recomendación sobre la ética de la inteligencia artificial. UNESCO.

<https://unesdoc.unesco.org/ark:/48223/pf0000381137>

Van Inwegen, E., Munyikwa, Z., & Horton, J. J. (2023, 19 de enero). Algorithmic writing assistance on jobseekers' resumes increases hires (Working Paper). arXiv.

<https://doi.org/10.48550/arXiv.2301.08083>

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., &

Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 5998–6008. <https://arxiv.org/abs/1706.03762>

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., & Cao, Y. (2022). ReAct:

Synergizing reasoning and acting in language models. arXiv.

<https://arxiv.org/abs/2210.03629>

## **Anexos**

### **Anexo A: Prompts Utilizados**

#### **ANEXO A: PROMPTS UTILIZADOS CON GPT-4**

Este anexo documenta los prompts principales utilizados en el sistema de optimización de CV, así como su evolución a lo largo del desarrollo del proyecto.

#### **PROMPT MAESTRO FINAL (Versión 15 - Versión Estable)**

Eres un asistente experto en Recursos Humanos especializado en mejora de currículos para profesionales del sector TI. Tu tarea es analizar un currículum vitae y una descripción de oferta laboral, y generar sugerencias concretas de mejora.

#### **INSTRUCCIONES GENERALES:**

- Sé objetivo, profesional y constructivo en tus recomendaciones
- No reveles información personal del candidato
- No hagas suposiciones no basadas en el texto proporcionado
- No aportes sesgos relacionados con edad, género, etnia u otros atributos sensibles
- Mantén un tono motivador pero formal
- Limita tus sugerencias a un máximo de 8 por categoría para mantener concisión

#### **ESTRUCTURA DE SALIDA:**

Genera sugerencias organizadas en las siguientes cuatro categorías:

## 1. CONTENIDO

- Identifica brechas de información relevante
- Sugiere logros cuantificables que podrían añadirse
- Recomienda mejoras en la estructura de secciones
- Señala información redundante o poco relevante
- Si es posible, menciona ejemplos específicos o palabras puntuales que falten

## 2. PALABRAS CLAVE

- Extrae palabras clave exactas de la oferta laboral
- Identifica keywords faltantes en el CV
- Sugiere sinónimos y variaciones estratégicas
- Indica en qué secciones del CV deberían incluirse
- Proporciona ejemplos de integración natural (evitar "keyword stuffing")

## 3. ESTILO

- Detecta uso de primera persona y sugiere conversión a tercera persona
- Identifica verbos de acción y sugiere mejoras
- Señala inconsistencias en tiempos verbales
- Sugiere mejoras en concisión y claridad
- Detecta errores ortográficos y gramaticales
- Sugiere uso de negritas para énfasis en cargos y métricas

#### 4. FORMATO

- Verifica compatibilidad ATS (sin tablas, columnas múltiples, imágenes)
- Sugiere mejoras en estructura de secciones
- Identifica problemas de legibilidad
- Recomienda formato de viñetas y espaciado
- Verifica presencia de encabezados estándar

#### ENTRADA:

- CV del candidato: [TEXTODELCV]
- Descripción de la oferta laboral: [TEXTODELA\_OFERTA]

#### NOTAS IMPORTANTES:

- Si sugieres eliminar secciones poco relevantes para ATS, advierte al usuario que esas secciones pueden tener valor personal
- Prioriza sugerencias que tengan mayor impacto en compatibilidad ATS y empleabilidad
- Sé específico: en lugar de "mejora la redacción", proporciona ejemplos concretos de reescritura

#### EVOLUCIÓN DEL PROMPT (Resumen de Iteraciones)

##### ITERACIÓN 1-3 (Versión Inicial):

- Prompt monolítico básico

- Instrucciones generales sin estructura clara
- Resultado: Sugerencias demasiado genéricas, sin categorización

#### ITERACIÓN 4-6:

- Introducción de estructura de 4 categorías
- Añadido límite de sugerencias por categoría
- Resultado: Mejora en organización, pero aún falta especificidad

#### ITERACIÓN 7-9:

- Añadido énfasis en palabras clave exactas de la oferta
- Instrucciones específicas sobre integración natural de keywords
- Resultado: Mejora de ~10 puntos porcentuales en compatibilidad ATS

#### ITERACIÓN 10-12:

- Refinamiento de instrucciones de estilo (tercera persona, verbos de acción)
- Añadido énfasis en métricas cuantificables
- Resultado: Mejoras en calidad percibida por expertos

#### ITERACIÓN 13-14:

- Añadido balance entre optimización ATS y preferencias personales
- Refinamiento de tono (de imperativo a sugerente)
- Resultado: Mayor aceptación de sugerencias por usuarios

### ITERACIÓN 15 (Versión Final):

- Integración de todas las mejoras anteriores
- Optimización de longitud y claridad
- Resultado: Versión estable con mejor equilibrio entre exhaustividad y precisión

### PROMPT ESPECÍFICO PARA ANÁLISIS DE KEYWORDS

Eres un simulador de ATS y experto en optimización de CVs IT. Recibirás dos bloques como entrada:

- OFERTA de empleo (Descripción completa del puesto, tecnologías y competencias requeridas)
- CV del candidato (Texto completo del currículum)

Tu tarea es:

1. Extraer palabras clave exactas de la oferta: lenguajes, tecnologías, herramientas, metodologías y competencias tal como aparecen (p. ej. "Python", "AWS", "ciberseguridad").
2. Generar sinónimos y variaciones estratégicas para cubrir posibles filtros (p. ej. "SQL" → "bases de datos relacionales", "Scrum" → "metodologías ágiles").

3. Verificar presencia en el CV, concretamente en las secciones de Habilidades y

Experiencia:

- Coincidencia exacta de cada keyword
- Uso contextualizado en descripciones de logros, no listados aislados (evitar "keyword stuffing")

4. Detectar listados sueltos de keywords sin integración en frases y advertir si existen "trucos" penalizados.

5. Confirmar encabezados estándar (Experiencia, Educación, Habilidades) para asegurar compatibilidad ATS.

Salida esperada:

- Listado de keywords extraídas de la oferta y sus sinónimos recomendados
- Keywords faltantes en el CV y sección(es) donde añadirlas
- Ejemplos de integración natural, p. ej.:
  - Habilidades: "Java, AWS, Docker, Microservicios (Spring Boot)"
  - Experiencia: "Desarrollé microservicios en Java sobre AWS (EC2, Lambda)..."
- Instrucciones específicas para insertar cada término de forma coherente dentro de viñetas que destaquen logros cuantificables

## PROMPT PARA VALIDACIÓN DE FORMATO ATS

Eres un especialista en formatos ATS-friendly para CVs del sector TI. A continuación recibirás como entrada el texto o archivo (PDF/.docx) de un currículum. Tu tarea es evaluar estos criterios de formato:

### 1. Diseño cronológico inverso y estructura de columna única

- Verifica que el CV siga un formato tradicional, cronológico (o inverso) en una sola columna, sin diseños creativos o funcionales que compliquen la lectura.

### 2. Tipografía y legibilidad

- Comprueba que la fuente sea clara (Arial, Calibri o Verdana) de tamaño 10–12 pt y que no se usen tipografías ornamentadas.

- Confirma uso coherente de negritas para encabezados.

### 3. Espacios y márgenes

- Asegura márgenes uniformes (~2,5 cm) en todos los lados y suficiente espacio en blanco entre secciones para facilitar la lectura.

### 4. Orden de secciones

- Verifica que siga el orden estándar: Encabezado (nombre y contacto), Perfil/Objetivo, Educación, Experiencia, Habilidades, Certificaciones, Idiomas, etc.

- Revisa consistencia en títulos y formato de fechas.

### 5. Elementos gráficos y columnas

- Detecta y marca tablas complejas, columnas múltiples, imágenes, logos o diagramas que puedan confundir un ATS. Se recomienda un diseño limpio sin estos elementos.

### 6. Paginación y encabezados de página

- Confirma que, en documentos de varias páginas, aparezca el nombre del candidato y "Página X de N" en el encabezado de cada hoja sucesiva.

### 7. Formato de archivo

- Verifica que el CV esté guardado en PDF o en .docx simple (sin estilos avanzados), para mantener la compatibilidad con la mayoría de ATS.

Para cada criterio, devuelve:

- Estado: ✓ Cumple / ✗ No cumple

- Observación: breve comentario sobre conformidad y legibilidad

- Sugerencia: ejemplo concreto de ajuste (p. ej., "Cambiar fuente a Calibri 11 pt" o

"Eliminar tabla de habilidades, convertirla en viñetas simples")

## MÉTRICAS DE EVOLUCIÓN DEL PROMPT

**Tabla 8***Métricas de evolución del prompt*

<b>Iteración</b>	<b>Compatibilidad</b>	<b>Satisfacción</b>	<b>Observaciones</b>
	<b>ATS Promedio</b>	<b>Usuario</b>	
<b>1-3</b>	68.2%	3.2/5	Sugerencias genéricas
<b>4-6</b>	72.5%	3.6/5	Mejor organización
<b>7-9</b>	78.3%	4.1/5	Mejora en keywords
<b>10-12</b>	81.7%	4.4/5	Mejora en estilo
<b>13-14</b>	83.9%	4.6/5	Mejor balance
<b>15</b>	85.1%	4.7/5	Versión estable

**(Final)***Nota.* Autoría propia

- Temperatura del modelo: 0.7 (balance entre creatividad y consistencia)
- Máximo de tokens: 2000 (suficiente para sugerencias detalladas)
- Modelo: GPT-4 (gpt-4-turbo-preview)
- Formato de respuesta: JSON estructurado para facilitar parsing
- Validación post-procesamiento: Eliminación de repeticiones y estandarización de

formato

## **Anexo B: Herramientas y Tecnologías Utilizadas**

### ***ANEXO B: HERRAMIENTAS Y TECNOLOGÍAS UTILIZADAS***

Este anexo documenta todas las herramientas, tecnologías, frameworks y servicios utilizados en el desarrollo del sistema de optimización de CV.

#### **B.1. Stack Tecnológico Principal**

##### **### Frontend**

- **Blazor WebAssembly 8.0:** Framework de Microsoft para aplicaciones web interactivas ejecutándose en el navegador mediante WebAssembly
- **Bootstrap 5.3:** Framework CSS para diseño responsive y componentes UI
- **Font Awesome 6.4:** Biblioteca de iconos vectoriales
- **JavaScript ES6+:** Lenguaje de programación para lógica del frontend
- **HTML5 Semántico:** Estructura de la interfaz de usuario

##### **### Backend**

- **ASP.NET Core 8.0:** Framework web de Microsoft para construir APIs REST y servicios
- **C# 10.0:** Lenguaje de programación del backend
- **.NET 8.0 SDK:** Plataforma de desarrollo

### ### Servicios de IA y Cloud

- **Azure OpenAI Service**: Servicio de Microsoft Azure que proporciona acceso a modelos de OpenAI
- **GPT-4 (gpt-4-turbo-preview)**: Modelo de lenguaje generativo utilizado para generar recomendaciones
- **Azure App Service**: Servicio de hospedaje y despliegue del backend con escalabilidad automática
- **Azure Blob Storage**: Almacenamiento de objetos para métricas anónimas y logs
- **Azure Key Vault**: Gestión segura de secretos, claves API y certificados
- **Azure Application Insights**: Telemetría, monitoreo y análisis de rendimiento

### ### Herramientas de Procesamiento de Documentos

- **iText7 7.2.5**: Biblioteca Java/.NET para procesamiento y extracción de texto desde archivos PDF
- **DocumentFormat.OpenXml 3.0**: Biblioteca .NET para leer y escribir archivos Office Open XML (DOCX, XLSX, etc.)

### ### Herramientas de Evaluación y Validación

- **Jobscan (2025)**: Plataforma web para simulación de compatibilidad ATS y análisis de keywords
  - Utilizado para validar compatibilidad ATS antes y después de la optimización
  - Proporciona métricas de matching de palabras clave
  - Identifica keywords faltantes y sugiere mejoras

#### ### Bibliotecas de Soporte

- **Polly 8.3**: Biblioteca .NET para políticas de resiliencia (retry, circuit breaker, timeout)
- **Microsoft.Extensions.Http**: Extensiones para clientes HTTP con soporte para inyección de dependencias

## B.2. Herramientas de Desarrollo

#### ### Entornos de Desarrollo

- **Visual Studio 2022**: IDE principal para desarrollo en .NET
- **Visual Studio Code**: Editor alternativo con extensiones para C# y Blazor
- **Git 2.40**: Sistema de control de versiones

#### ### Herramientas de Pruebas

- **JMeter 5.6**: Herramienta de pruebas de carga y rendimiento
  - Utilizado para simular múltiples usuarios concurrentes
  - Validación de tiempos de respuesta bajo carga
  - Pruebas de escalabilidad

#### ### Herramientas de Análisis

- **Microsoft Excel (Microsoft 365)**: Análisis estadístico de métricas y resultados
- **Power BI** (opcional): Visualización de datos y dashboards de métricas

### B.3. Herramientas de Documentación

- **Markdown (CommonMark):** Formato de documentación técnica
- **Mermaid 10.0:** Generación de diagramas en formato texto (flujos, secuencias, arquitectura)
- **Microsoft Word:** Edición del documento final del proyecto de grado

### B.4. Servicios y APIs Externas

### Azure OpenAI Service

- **Endpoint:** `https://[recurso].openai.azure.com/`
- **Modelo:** `gpt-4-turbo-preview`
- **Parámetros:**
  - Temperature: 0.7
  - Max Tokens: 2000
  - Top P: 1.0

### Jobscan API

- **Endpoint:** `https://www.jobscan.co/api/`
- **Funcionalidad:** Análisis de compatibilidad ATS y matching de keywords
- **Uso:** Validación externa de resultados de optimización

## **B.5. Configuración de Seguridad**

- **HTTPS/TLS 1.2+**: Cifrado en tránsito para todas las comunicaciones
- **Azure Key Vault**: Almacenamiento seguro de:
  - Claves API de Azure OpenAI
  - Secretos de conexión
  - Certificados TLS
- **Azure AD (futuro)**: Autenticación y autorización de usuarios

## **B.6. Métricas y Observabilidad**

- **Azure Application Insights**:
  - Latencias (P50, P95, P99)
  - Tasa de errores
  - Throughput
  - Trazas de solicitudes
- **Azure Blob Storage**:
  - Logs estructurados
  - Métricas anónimas de uso
  - Datasets de evaluación

## B.7. Costos Estimados (Escenario Piloto)

**Tabla 9**

*Costos estimados*

Servicio	Uso Estimado	Costo Mensual
		Aproximado
Azure OpenAI (GPT-4)	1000 solicitudes/mes	30-50 USD
Azure App Service (Basic)	1 instancia	13 USD
Azure Blob Storage	10 GB	0,20 USD
Azure Key Vault	Estándar	0,03 USD
Application Insights	5 GB datos	2,30 USD
Total		45-65 USD/mes

*Nota.* Costos basados en uso típico de un proyecto piloto académico. Los costos pueden variar según volumen de uso.

## B.8. Requisitos del Sistema

### Requisitos del Servidor

- **Sistema Operativo:** Windows Server 2019+ o Linux (Ubuntu 20.04+)
- **Runtime:** .NET 8.0 Runtime

- **Memoria:** Mínimo 2 GB RAM (recomendado 4 GB)
- **CPU:** 2 cores (recomendado 4 cores)
- **Almacenamiento:** 20 GB SSD

#### ### Requisitos del Cliente

- **Navegador:** Chrome 90+, Firefox 88+, Edge 90+, Safari 14+
- **JavaScript:** Habilitado
- **Conexión:** Internet estable (mínimo 1 Mbps)

## B.9. Dependencias del Proyecto

#### ### Paquetes NuGet Principalesm

Microsoft.AspNetCore.Components.WebAssembly (8.0.0)

Microsoft.AspNetCore.Components.WebAssembly.DevServer (8.0.0)

Azure.AI.OpenAI (1.0.0-beta.17)

Azure.Storage.Blobs (12.19.0)

Azure.Security.KeyVault.Secrets (4.5.0)

Azure.Monitor.OpenTelemetry.AspNetCore (1.2.0)

Polly.Extensions.Http (3.0.0)

iText7 (7.2.5)

DocumentFormat.OpenXml (3.0.0)

## **B.10. Referencias y Documentación**

- Documentación oficial de Blazor: <https://learn.microsoft.com/aspnet/core/blazor/>
- Documentación de Azure OpenAI: <https://learn.microsoft.com/azure/ai-services/openai/>
- Guía de Jobscan: <https://www.jobscan.co/blog/>
- Documentación de ASP.NET Core: <https://learn.microsoft.com/aspnet/core/>

---

**Fecha de Documentación:** Enero 2025

**Versión del Sistema:** 1.0

**Autor:** Julian Eduardo Ospina Romero

## **Anexo C: Pantallazos del Sistema**

### ***ANEXO C: PANTALLAZOS DEL SISTEMA FUNCIONANDO***

Este anexo describe las pantallas principales del sistema de optimización de CV, documentando la interfaz de usuario y el flujo de trabajo.

## C.1. Página de Inicio (Home)

**Figura 6**

*Página de inicio*



*Nota.* Autoría propia

*Nota.* Captura de pantalla obtenida de la aplicación ejecutándose, mostrando la página de presentación del servicio que explica las funcionalidades principales del sistema.

### Elementos principales:

- Título y descripción del servicio
- Botón de acceso al formulario de optimización
- Información sobre características principales:
  - Optimización con IA generativa

- Compatibilidad ATS
- Análisis en tiempo real
- Privacidad y seguridad

*Nota.* La interfaz está diseñada con Bootstrap 5.3, proporcionando un diseño moderno y responsive.

---

## C.2. Formulario de Carga de CV (Upload Form)

### Figura 7

*Formulario de Carga de CV*

**Optimizador de CV**

Mejora tu hoja de vida con IA generativa. Reduce hasta un 50% el tiempo de edición y aumenta tus posibilidades de selección.

Email \*

Archivo de CV \*

Seleccionar archivo Ningún archivo seleccionado

Formatos soportados: PDF, DOCK, TXT (máximo 5MB)

Descripción de la Oferta Laboral \*

Pega aquí la descripción completa de la oferta laboral a la que quieres aplicar. Incluye requisitos, responsabilidades y habilidades solicitadas.

0 / 4000 caracteres (mínimo 50)

Optimizar CV

*Nota.* Autoría propia

**Descripción:** Interfaz principal donde el usuario carga su CV y proporciona la descripción de la oferta laboral.

**Elementos principales:**

**- Campo de carga de archivo:**

- Formatos aceptados: PDF, DOCX, TXT
- Tamaño máximo: 10 MB
- Validación en tiempo real del formato

**- Área de texto para oferta laboral:**

- Textarea con límite de 4000 caracteres
- Contador de caracteres en tiempo real
- Validación de longitud mínima (50 caracteres)

**- Botón "Optimizar":**

- Inicia el proceso de análisis
- Muestra indicador de carga durante el procesamiento

**- Vista previa del CV cargado:**

- Muestra el texto extraído del CV
- Permite verificar que la extracción fue correcta

**Validaciones:**

- Archivo debe ser PDF, DOCX o TXT
- Archivo no debe exceder 10 MB

- Descripción de oferta debe tener entre 50 y 4000 caracteres
- Email del usuario (opcional, para seguimiento)

---

### C.3. Indicador de Progreso (Progress Indicator)

#### Figura 8

##### *Indicador de Progreso*



*Nota.* Autoría propia

**Descripción:** Pantalla que muestra el progreso del procesamiento en tiempo real.

**Estados del proceso:**

1. **Carga y validación (0-10%):** Verificación de formato y extracción de texto
2. **Análisis del CV (10-30%):** Parsing y estructuración del contenido

3. **Análisis de la oferta** (30-50%): Extracción de keywords y requisitos
4. **Generación con IA** (50-90%): Procesamiento con GPT-4
5. **Post-procesamiento** (90-100%): Formateo y preparación de sugerencias

**Elementos visuales:**

- Barra de progreso animada
- Porcentaje de completitud
- Mensaje descriptivo del paso actual
- Tiempo estimado restante

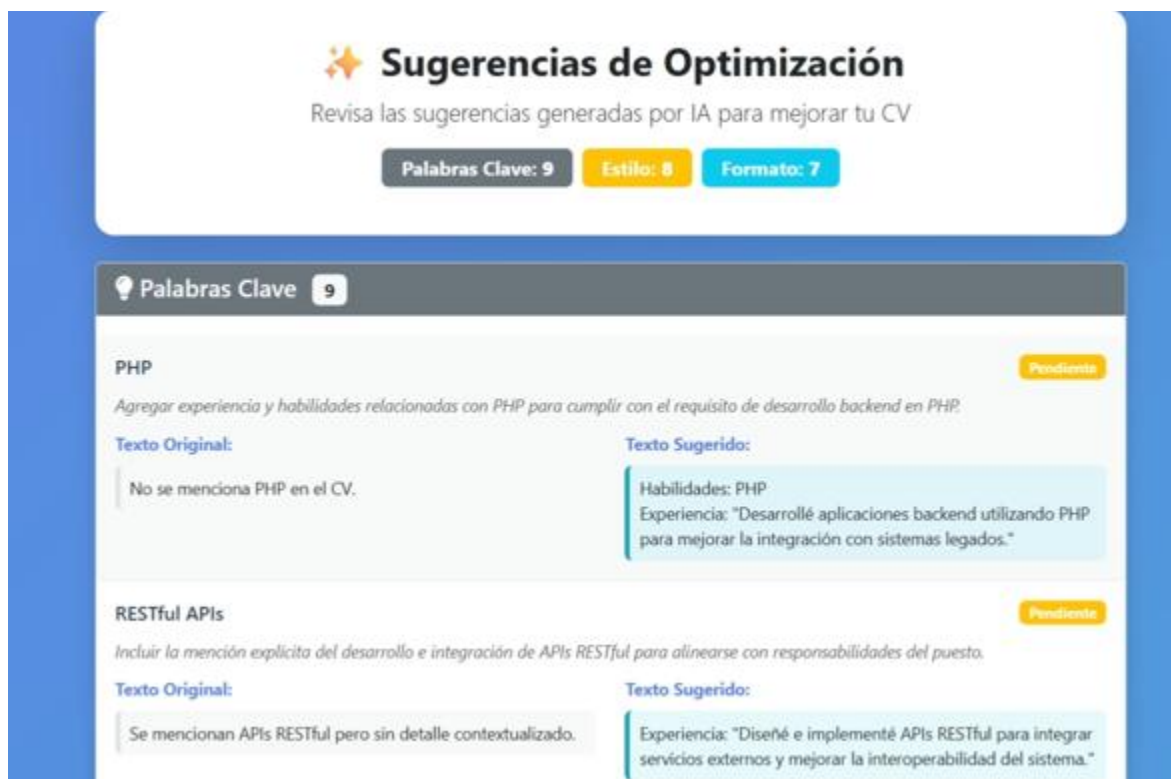
*Nota.* El progreso se actualiza mediante polling al endpoint `/api/progress/{cvId}` cada 2 segundos.

---

## C.4. Dashboard de Sugerencias (Suggestions Dashboard)

Figura 9

*Dashboard de Sugerencias*



*Nota.* Autoría propia

**Descripción:** Interfaz principal donde se presentan todas las sugerencias de optimización categorizadas.

### Estructura del Dashboard:

### **Panel Superior - Métricas:**

#### **- Score de Compatibilidad ATS:**

- Muestra porcentaje (ej: 85/100)
- Indicador visual de color (verde si  $\geq 80\%$ , amarillo si 60-79%, rojo si  $< 60\%$ )
- Comparación antes/después (si aplica)

### **Categorías de Sugerencias (4 pestañas o secciones):**

#### **1. CONTENIDO:**

- Lista de sugerencias con:
  - Icono de tipo (información, advertencia, mejora)
  - Texto de la sugerencia
  - Botón "Aplicar" o "Rechazar"
  - Icono de información (?) con explicación detallada al hacer hover/click

#### **2. PALABRAS CLAVE:**

- Lista de keywords faltantes
- Sugerencias de dónde incluirlas
- Ejemplos de integración natural

- Indicador de relevancia (alta/media/baja)

### 3. ESTILO:

- Sugerencias de redacción
- Detección de primera persona
- Mejoras en verbos de acción
- Correcciones ortográficas y gramaticales

### 4. FORMATO:

- Problemas de compatibilidad ATS detectados
- Sugerencias de estructura
- Mejoras en legibilidad

### **Funcionalidades:**

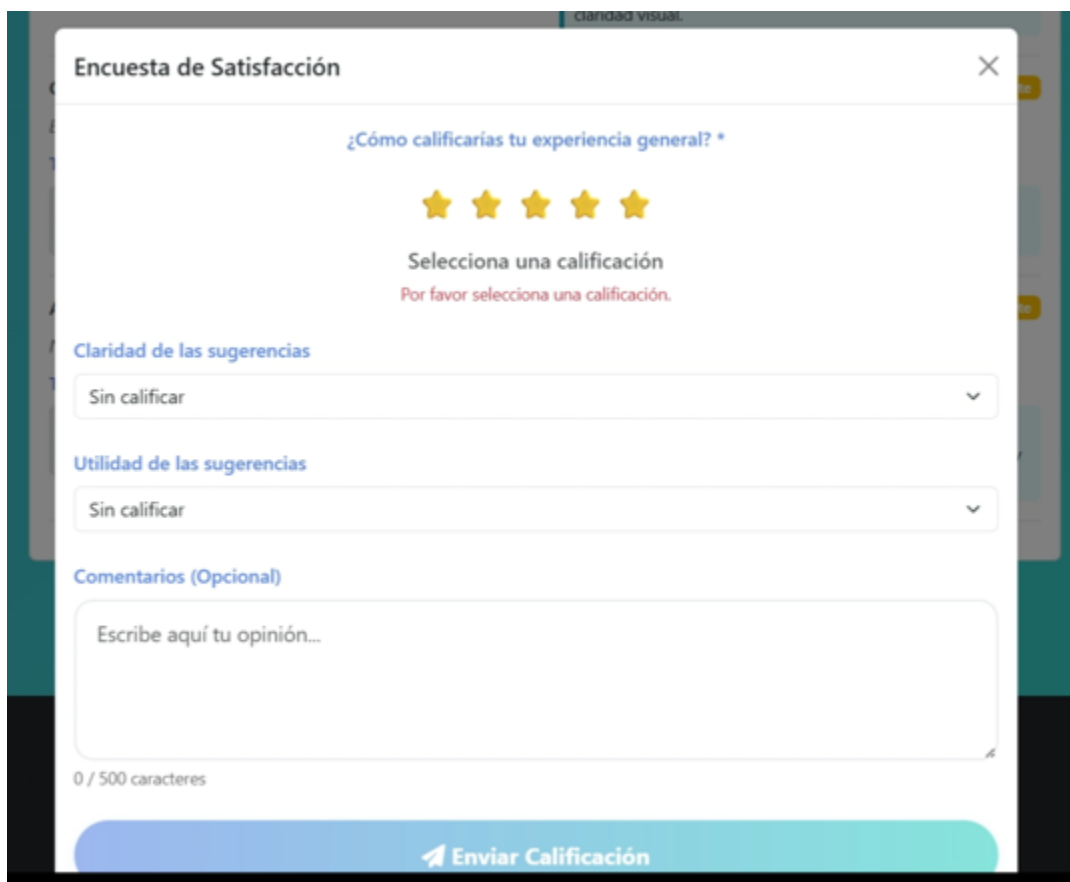
- Filtrado por categoría
- Búsqueda de sugerencias
- Aplicación individual o masiva de sugerencias
- Vista previa del CV optimizado
- Exportación del CV mejorado (PDF/DOCX)

---

## C.5. Encuesta de Satisfacción (Feedback Form)

**Figura 10**

*Encuesta de satisfacción*



The image shows a mobile application interface for a satisfaction survey. At the top, there is a title bar with the text 'Encuesta de Satisfacción' and a close button (X). Below the title bar, the main question is '¿Cómo calificarías tu experiencia general? \*'. This is followed by a five-star rating system with five yellow stars. Below the stars, the text reads 'Selecciona una calificación' and 'Por favor selecciona una calificación.'. There are three dropdown menus for rating specific aspects: 'Claridad de las sugerencias', 'Utilidad de las sugerencias', and 'Comentarios (Opcional)'. Each dropdown menu currently shows 'Sin calificar'. Below the dropdown menus is a text input field for 'Comentarios (Opcional)' with the placeholder text 'Escribe aquí tu opinión...'. At the bottom left of the input field, it says '0 / 500 caracteres'. At the bottom right, there is a large blue button with a white arrow icon and the text 'Enviar Calificación'.

*Nota.* Autoría propia

**Descripción:** Formulario para recopilar retroalimentación del usuario sobre la experiencia y utilidad de las sugerencias.

**Campos:**

- **Calificación General** (1-5 estrellas): Satisfacción global con el servicio
- **Claridad de Sugerencias** (1-5): ¿Fueron claras y comprensibles?
- **Utilidad de Sugerencias** (1-5): ¿Fueron útiles para mejorar el CV?
- **Comentarios** (texto libre): Observaciones adicionales
- **¿Recomendaría el servicio?** (Sí/No)

**Datos recopilados:**

- Se almacenan de forma anónima
- Se utilizan para mejorar el sistema
- No se asocian con el contenido del CV

Home → Upload Form → Progress Indicator → Suggestions Dashboard → Feedback

Form

---

**C.9. Notas Técnicas sobre la Interfaz**

- **Framework:** Blazor WebAssembly para lógica del frontend

- **Estilos:** Bootstrap 5.3 + CSS personalizado
- **Iconos:** Font Awesome 6.4
- **Comunicación:** REST API mediante fetch/XMLHttpRequest
- **Estado:** Gestión de estado local en componentes Blazor
- **Validación:** Validación tanto en cliente como en servidor

---

## C.10. Ejemplo de Datos Mostrados

### Score ATS

Compatibilidad ATS: 85/100

Antes: 66/100 | Después: 85/100 | Mejora: +19 puntos

### Sugerencia Ejemplo

Categoría: Palabras Clave

Tipo:  Advertencia

Sugerencia: La oferta menciona "Spring Boot" 5 veces, pero no aparece en tu CV.

Acción: Incluir "Spring Boot" en la sección de Habilidades y en la descripción de experiencia relevante.

Ejemplo: "Desarrollo de microservicios con Spring Boot y Java"

[ ] Aplicar sugerencia [i] Más información

---

*Nota.* Los pantallazos reales se pueden obtener ejecutando el sistema y capturando las pantallas. Este documento describe la estructura y funcionalidad de cada pantalla para referencia en el proyecto de grado.

---

**Fecha de Documentación:** Enero 2025

**Versión de la Interfaz:** 1.0