

Desarrollo de un prototipo de aplicación de entrenamiento para entrevistas laborales, dirigido tanto a la teoría como a la práctica.

Nilson Andrés Cuero Ocoro
Sebastián Pérez Bastidas

Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación
Ingeniería de Sistemas

Universidad del Valle
Sede Tuluá

25 de octubre de 2025

Desarrollo de un prototipo de aplicación de entrenamiento para entrevistas laborales, dirigido tanto a la teoría como a la práctica.

Nilson Andrés Cuero Ocoro
Código:2067570
`nilson.cuero@correounivalle.edu.co`

Sebastián Pérez Bastidas
Código:2067808
`sebastian.perez.bastidas@correounivalle.edu.co`

Director:
Migue Ángel Askar Rodríguez
`miguel.askar@correounivalle.edu.co`

Facultad de Ingeniería
Escuela de Ingeniería de Sistemas y Computación
Ingeniería de Sistemas

Universidad del Valle
Sede Tuluá

25 de octubre de 2025

Índice

1. Introducción	8
1.1. Formulación del Problema	8
1.1.1. Descripción del Problema	8
1.2. Objetivos	10
1.3. Resultados esperados	11
1.4. Metodología	12
1.4.1. Metodología de Investigacion	12
1.4.2. Metodología de Gestión de Actividades	13
2. Justificacion y declaracion de alcance	14
2.1. Justificación	15
2.2. Alcance del Proyecto	16
2.2.1. Declaración del Alcance	16
3. Marco Referencial	18
3.1. Marco Teórico	18
3.1.1. Teoría de la Evaluación Basada en Competencias	18
3.1.2. Teoría del procesamiento de la información	18
3.1.3. Teoría del Procesamiento del Lenguaje Natural (PLN)	19
3.1.4. Estado del Arte	20
3.1.5. Antecedentes	21
3.2. Marco Conceptual	24
3.2.1. Evaluador de entrevistas simuladas basado en IA	24
3.2.2. Modelos y Técnicas de Aprendizaje Automático	26
4. Diseño y prototipo	27
4.1. Mockups	29
4.2. Requisitos de software	33
4.2.1. Requisitos funcionales	33
4.2.2. Requisitos no funcionales	35
4.3. Casos de uso	36
4.4. Base de Datos	37
4.5. Stack tecnológico	39
4.5.1. Frontend	39
4.5.2. Backend	39
5. Implementación	40
5.1. Descripción general del proceso de implementación	40
5.1.1. Problemas técnicos y soluciones implementadas	40
5.2. Integración de la Inteligencia Artificial	42
5.3. Generación y envío de Prompts a la Inteligencia Artificial	44
5.3.1. Resultado final de la implementacion	45

6. Pruebas	46
6.1. Plan de pruebas	46
6.1.1. Pruebas funcionales	46
6.1.2. Pruebas de usabilidad	48
6.2. Análisis preliminar	50
6.3. Perfil de los encuestados	50
6.4. Diseño de la encuesta	50
6.5. Resultados de la encuesta	51
6.5.1. Resultados obtenidos del proyecto	62
6.5.2. Análisis e importancia de la encuesta	62
7. Conclusiones y Trabajos Futuros	64
7.1. Conclusiones	64
7.2. Trabajos Futuros	66
8. Anexos	71

Índice de figuras

1.	Diagram de arquitectura	28
2.	Mockup inicio de aplicacion	30
3.	Mockup login	31
4.	Mockup area de estudiantes	31
5.	Mockup area entrevistadores	32
6.	Mockup registro de usuario	32
7.	Mockup recuperar contraseña	32
8.	Diagrama de casos de uso	36
9.	Modelo relacional	38
10.	Diagram de arquitectura tecnica del sistema	43
11.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	51
12.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	52
13.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	53
14.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	54
15.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	55
16.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	56
17.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	57
18.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	58
19.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	59
20.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	60
21.	Encuesta sobre la usabilidad y funcinabilidad de la aplicacion	61

Índice de tablas

1.	Resultados esperados	11
2.	Comparación Antecedentes.	22

Resumen

El presente documento expone el desarrollo de un prototipo de aplicación de entrenamiento para entrevistas laborales, que también será nombrado en este documento como InterviewSim, una aplicación web diseñada para simular entrevistas técnicas mediante la integración de inteligencia artificial. Su propósito principal es servir como herramienta de preparación para estudiantes de Ingeniería de Sistemas o áreas afines, permitiéndoles fortalecer sus habilidades técnicas y mejorar su desempeño en procesos de selección reales.

Se aborda de manera estructurada todas las fases del proceso de desarrollo: desde el planteamiento del problema y los objetivos, hasta la implementación técnica, pruebas y análisis de resultados. Se detalla el diseño del sistema basado en una arquitectura cliente-servidor, donde el frontend fue implementado con React, el backend con Node.js y Express, y la base de datos con MongoDB. Además, se explica la integración de la API de OpenAI, encargada de generar preguntas, respuestas y retroalimentación personalizada según el nivel de dificultad y el tipo de entrevista.

Finalmente, el documento presenta los resultados de las pruebas de usabilidad y funcionalidad, aplicadas a 20 usuarios, junto con su respectivo análisis estadístico y cualitativo. Los datos obtenidos evidencian una alta aceptación del sistema, destacando la facilidad de uso, la claridad de la interfaz y la utilidad de la retroalimentación automática como elementos diferenciadores del proyecto.

Palabras clave: Inteligencia artificial, Aprendizaje automático, Procesamiento del lenguaje natural, Interacción persona-computadora, Gestión del desarrollo de software

1. Introducción

1.1. Formulación del Problema

1.1.1. Descripción del Problema

En la industria tecnológica, las entrevistas para puestos de programación suelen incluir evaluaciones rigurosas tanto de conocimientos teóricos como de habilidades prácticas en resolución de problemas. Estas entrevistas evalúan la competencia técnica de un candidato, la comprensión de la informática y las capacidades de resolución de problemas. Sin embargo, los candidatos suelen enfrentarse a una serie de obstáculos en su preparación para estas entrevistas, lo que afecta su confianza y rendimiento negativamente[1]. Con frecuencia, presentan segmentos de codificación en vivo donde los candidatos deben articular su proceso de pensamiento y diseñar soluciones óptimas centradas en la eficiencia y la conservación de la memoria.

Los candidatos a ingenieros de software suelen participar en entrevistas técnicas en la pizarra como parte de una evaluación de contratación. Durante estas sesiones, los candidatos escriben código mientras piensan en voz alta mientras trabajan para encontrar una solución, bajo la atenta mirada de un entrevistador. Si bien las entrevistas técnicas deberían permitir una evaluación imparcial e inclusiva de la capacidad de resolución de problemas, sorprendentemente, las entrevistas técnicas pueden ser, en cambio, un procedimiento para identificar a los candidatos que mejor manejan y migrar el estrés causado únicamente por ser examinados por un entrevistador (ansiedad por el desempeño)[2].

Las empresas se preocupan mucho por contratar al desarrollador equivocado, pero pasan por alto el alto costo de dejar pasar buenos candidatos. Por ejemplo, rechazar a un desarrollador con más de 10 años de experiencia y muchas contribuciones notables en proyectos de código abierto que no puede desempeñarse bien en una pizarra frente a un entrevistador porque sufre el síndrome del impostor. Además, la mayoría de los desarrolladores no son consistentes con su desempeño en todas las entrevistas técnicas a las que se someten[3].

Otra de las dificultades es que los candidatos suelen enfrentarse a niveles de estrés muy altos al momento de presentar una entrevista laboral real, lo que puede llevar a manifestar el síndrome del impostor, esto le causa a los candidatos inseguridad, falta de confianza en ellos mismos y como consecuencia, impacta negativamente en su rendimiento.

No se han llevado a cabo investigaciones formales sobre el PI en ingenieros de software en general, a pesar de que sus consecuencias pueden contribuir a trastornos de salud mental, como la depresión y el agotamiento. Este estudio describe una encuesta que investiga el alcance de los sentimientos de impostor en los ingenieros de software, considerando aspectos como el género, la raza/etnia y los roles. Además, se investiga la influencia del PI en su productividad percibida. El instrumento de la encuesta se diseñó utilizando un enfoque basado en la teoría e incluyó preguntas demográficas, una escala de PI validada internacionalmente (CIPS) y preguntas para medir la productividad percibida con base en los constructos del marco SPACE. La encuesta se envió a empresas que operan en varios sectores comerciales.

El análisis de datos utilizó bootstrap con remuestreo para calcular intervalos de confianza y pruebas de significación estadística de Mann-Whitney para evaluar las hipótesis. Recibimos respuestas de 624 ingenieros de software distribuidos en 26 países. Los resultados del bootstrap revelan que un 52,7 % de los ingenieros de software experimentan niveles frecuentes o intensos de IP y que las mujeres sufren en una proporción significativamente mayor (60,6 %) que los hombres (48,8 %) [4]. Estudios previos han identificado el PI como un fenómeno psicológico que afecta especialmente a profesionales de alto rendimiento[5], con un impacto significativo en la productividad y el bienestar emocional [6], [7]. En el ámbito de la ingeniería de software, se ha observado que el PI está asociado con una menor confianza en las habilidades técnicas y un mayor nivel de estrés [8].

Algunas de las dificultades comunes a las que se enfrentan son la retroalimentación limitada, esto debido a que los candidatos practican por su cuenta y es difícil evaluar objetivamente su rendimiento o recibir retroalimentación detallada en áreas clave como la eficiencia y precisión de su código. Los candidatos pueden cometer errores frecuentes en la lógica o eficiencia del código sin recibir correcciones detalladas que les ayuden a entender sus fallos y así poder mejorar, esto no afecta solamente la calidad de su código, sino también la capacidad de adaptarse y poder pensar de forma crítica bajo presión.

Los participantes de Hacker News informan de varias preocupaciones y percepciones negativas sobre las entrevistas, incluida su falta de relevancia para el mundo real, el sesgo hacia los desarrolladores más jóvenes y la exigencia de un compromiso de tiempo. Los participantes informan de que estas entrevistas provocan ansiedad y frustración innecesarias, lo que les obliga a aprender normas arbitrarias, implícitas y oscuras [9].

Para entender si las entrevistas de programación, tal como se realizan hoy en día, pueden inducir estrés que afecte significativamente el desempeño, realizamos un ensayo controlado aleatorio con 48 estudiantes de Ciencias de la Computación, comparándolos en entornos privados y públicos con pizarras blancas. Descubrimos que el desempeño se reduce a más de la mitad, simplemente por ser observados por un entrevistador. También observamos que el estrés y la carga cognitiva eran significativamente mayores en una entrevista técnica tradicional en comparación con nuestra entrevista privada. En consecuencia, los entrevistadores pueden estar filtrando a los candidatos calificados al confundir la evaluación de la capacidad de resolución de problemas con un estrés innecesario [2].

La combinación de estos factores genera una baja autoconfianza, y los candidatos a menudo dudan de sus propias habilidades al momento de enfrentarse a situaciones técnicas en tiempo real[10], [11]. En las entrevistas reales, los candidatos enfrentan condiciones que combinan alta presión, un tiempo limitado, y la necesidad de justificar sus decisiones en cada paso. Al no tener un entorno de práctica que simule estas condiciones, muchos candidatos pueden encontrar difícil replicar el ritmo y la mentalidad necesaria para sobresalir en la entrevista, lo cual puede llevar a problemas de rendimiento en las situaciones reales.

1.2. Objetivos

Objetivo General

Desarrollar un prototipo de una aplicación web que permita la práctica de entrevistas a candidatos a empleos en el área de programación.

Objetivos Específicos

1. Recopilar ejercicios prácticos y teóricos al nivel de una entrevista.
2. Entrenar una inteligencia artificial para la creación de ejercicios prácticos, de modo que alcancen el estándar para preguntas de entrevistas.
3. Diseñar la arquitectura y la interfaz de la aplicación web con el fin de simular entrevistas técnicas en el campo de la programación.
4. Implementar la aplicación incorporando la inteligencia artificial entrenada y los ejercicios recopilados.
5. Evaluar la aplicación respecto a su usabilidad y funcionalidad a través de pruebas de usuario.

1.3. Resultados esperados

Se planea crear una aplicación que integre preguntas tanto prácticas como teóricas, organizadas de acuerdo en como se muestra la siguiente tabla .

Objetivo General	Objetivo Específico	Producto Esperado
Desarrollar el prototipo de aplicación web de entrevista donde se pueda emplear la práctica de preguntas tanto técnicas y teóricas utilizando una IA y conocimiento entregado por profesionales en el tema	Recopilar de ejercicios prácticos y teóricos al nivel de los de una entrevista.	Una base de datos organizada que contenga ejercicios prácticos y teóricos categorizados por nivel de dificultad en los diferentes temas que contempla la programación, más un documento con el analisis y resumen de los datos encontrados.
	Entrenar una inteligencia artificial para la creación de ejercicios prácticos, de modo que alcancen el estándar para pregunta de entrevista.	Un modelo de IA entrenada que proporcione preguntas de calidad por niveles de dificultad, así como una muestra de precisión y relevancia en la calidad de las preguntas generadas.
	Diseño de la estructura e interfaz de una aplicación web con el fin de simular completamente entrevistas técnicas en el campo de la programación.	Un documento que presenta diseños y mockup, centrado en proporcionar una experiencia amigable que haga que el uso de la aplicación sea sencillo y accesible para los usuarios, se entrega un link de acceso publico sin permisos de edición para consultar los diseños de la aplicación.
	Implementar aplicaciones web que integren modelos de inteligencia artificial previamente entrenados, con el objetivo de generar dinámicamente preguntas teóricas y ejercicios prácticos de programación.	Un prototipo que combine el front-end y el back-end, incorporando características avanzadas como inteligencia artificial y un banco de preguntas creado en colaboración con profesores expertos en la materia.
	Implementar un proceso de evaluación integral para una aplicación web con enfoque en analizar su usabilidad y funcionalidad a través de pruebas de usuario.	Resultados obtenidos de la aplicación por medio de usuarios reales para recolectar feedback de modo que la aplicación cumpla con lo planteado.

Tabla 1: Resultados esperados
Fuente: Elaboración propia.

1.4. Metodología

1.4.1. Metodología de Investigación

Para evaluar el alcance y la madurez tecnológica de las actividades de Investigación, Desarrollo Tecnológico e Innovación (I+D+i) asociadas al proyecto, se adoptará el enfoque de Niveles de Madurez Tecnológica (TRL)[12], un marco conceptual desarrollado por la NASA y posteriormente adoptado por el Ministerio de Ciencia e Investigación de Colombia[13].

El Modelo de Investigación en Ingeniería del Software se compone de tres fases principales[14]:

1. Investigación y desarrollo inicial: Se define el área de trabajo del proyecto identificando necesidades de la industria o tendencias tecnológicas emergentes mediante diagnósticos, revisiones bibliográficas y pruebas de concepto en laboratorio.
2. Investigación aplicada: Se desarrolla y aplica la tecnología en entornos reales, identificando y adaptando posibles fallas del modelo para su implementación en empresas.
3. Transferencia: Se amplía el impacto de las nuevas tecnologías más allá de las empresas participantes mediante cursos, conferencias y licenciamiento, conectando con los niveles de madurez tecnológica y extendiendo los resultados obtenidos.

1.4.2. Metodología de Gestión de Actividades

La gestión de actividades dentro del proyecto resulta esencial para garantizar una ejecución organizada, eficiente y alineada con los objetivos establecidos. Con el fin de optimizar la planificación, el seguimiento y la distribución de tareas entre los integrantes del equipo, se adoptó una metodología ágil que permitiera visualizar el progreso de cada fase del desarrollo y facilitar la toma de decisiones. En este contexto, se usó una herramienta flexible y orientada a la mejora continua, que contribuye al control efectivo del flujo de trabajo y a la priorización de actividades de forma dinámica y colaborativa.

Metodología Kanban

Kanban es una herramienta de gestión de flujo de trabajo visual, versátil y adaptable, que organiza y monitorea actividades del equipo mediante un tablero visual, columnas de proceso, tarjetas de tareas y límites de trabajo en curso. Facilita la visualización del progreso, la identificación temprana de desafíos y la toma de medidas correctivas. Sus principios promueven la mejora continua y la eficiencia operativa, limitando el trabajo en curso para evitar la sobrecarga y mantener la calidad. Implementar Kanban mejora la eficiencia, productividad, colaboración y comunicación interna, permitiendo una respuesta ágil a cambios en los requisitos del proyecto y las prioridades del cliente[15].

Jira se presenta como una excelente opción para aplicar la metodología Kanban gracias a su capacidad para manejar de manera efectiva proyectos ágiles mediante herramientas como tableros visuales adaptables, listas de tareas y sprints. Estas funcionalidades facilitan una asignación clara de responsabilidades y un seguimiento exhaustivo del progreso del equipo [16]. Los beneficios de Jira también se extienden a su flexibilidad para adaptarse a diversas metodologías ágiles y su habilidad para garantizar la transparencia en el avance del proyecto a través de tableros visuales personalizables. Además, promueve una colaboración efectiva al centralizar la comunicación y el seguimiento de tareas dentro del equipo, lo cual mejora significativamente la gestión del ciclo de vida del proyecto y potencia la productividad del equipo [17]

2. Justificación y declaración de alcance

En el contexto del actual mercado laboral, la preparación para las entrevistas se ha vuelto fundamental para alcanzar los puestos anhelados. Por lo tanto, las entrevistas laborales se convierten en una prueba no solo de conocimiento, sino también de capacidad para resolver problemas bajo presión, así como de habilidades en la práctica real para resolver problemas de manera eficiente y mostrar confianza en sus conocimientos. Dado que el proceso de entrevista puede resultar intimidante, una plataforma de práctica que sea interactiva y adaptada a las necesidades individuales puede ser de gran ayuda para que los candidatos mejoren sus competencias en este ámbito.

Debido a la creciente necesidad de profesionales capacitados en la industria tecnológica, resalta la necesidad de herramientas que ayuden a los candidatos a mejorar sus habilidades y confianza antes de enfrentar entrevistas reales. Este problema es aún más relevante cuando consideramos el creciente número de graduados en áreas de tecnología relacionadas con la programación que ingresan al mercado cada año. La competencia es intensa y muchos candidatos pueden estar bien capacitados para esto, pero pueden fallar por no estar bien preparados para las exigencias del proceso de selección. Este problema afecta tanto a las empresas que están en busca de talentos cualificados, como a los candidatos que se enfrentan a procesos de selección donde sus habilidades y verdaderas capacidades pueden no verse reflejadas con precisión. En base a esto, el objetivo principal de este trabajo de grado es diseñar y construir un prototipo de una aplicación web que permita a los usuarios simular una entrevista de trabajo. La herramienta utilizando inteligencia artificial, se centrará a proporcionar recomendaciones que ayudarán a los candidatos a identificar y mejorar sus habilidades. En este documento se presentan los detalles del proyecto organizados en varias secciones clave. En primer lugar, se expone la descripción del problema, justificando la necesidad del desarrollo de la herramienta y detallando los desafíos que enfrentan los candidatos y las empresas en los procesos de selección. Luego, se aborda el marco referencial, que incluye los fundamentos teóricos y el estado del arte, destacando las limitaciones de las soluciones existentes y cómo este proyecto busca superarlas. Posteriormente, se describen los objetivos del proyecto y su alcance, especificando las metas generales y específicas, así como los resultados esperados. También se detalla la metodología empleada para el desarrollo del prototipo, incluyendo las tecnologías seleccionadas, los enfoques de investigación y las estrategias de gestión adoptadas. Finalmente, se presentan los cronogramas, presupuestos y el impacto ambiental asociado al proyecto, ofreciendo una visión integral de su planificación, ejecución y sostenibilidad.

2.1. Justificación

La creación de una aplicación web para entrevistas técnicas surge como respuesta a la necesidad de mejorar y agilizar el proceso de evaluación de habilidades tanto teóricas como prácticas en el campo de la programación. La motivación detrás de este proyecto, es mitigar la dificultad que afrontan algunos candidatos en el campo de la programación, para llevar a cabo sus entrevistas técnicas y prácticas de manera efectiva. Este proyecto se enfoca en los desafíos que enfrentan los usuarios, ya sean estudiantes, quienes buscan demostrar sus conocimientos y habilidades, como también profesores, que deben evaluar a los estudiantes de manera precisa y estructurada en un entorno académico cada vez más competitivo. La amplia variedad de habilidades y conocimientos que se piden en algunas entrevistas, desde conceptos teóricos hasta la implementación práctica en diferentes lenguajes de programación, puede resultar abrumadora para algunos. Esta diversidad, junto con la falta de herramientas que integren elementos de estudio tanto teóricos como prácticos, a menudo puede conducir a evaluaciones poco precisas o fragmentadas. Esto puede generar desánimo o ansiedad por no poder responder algunas de las preguntas propuestas por el evaluador. En el contexto de la formación académica y profesional, donde el dominio de las habilidades técnicas es fundamental, la necesidad de una plataforma que facilite la evaluación objetiva tendría relevancia. Con el uso de técnicas avanzadas de inteligencia artificial, la aplicación promete adaptarse a las necesidades específicas de cada usuario, brindando recomendaciones personalizadas que incrementen la efectividad del aprendizaje y la preparación técnica. En un contexto educativo que exige cada vez más habilidades técnicas, esta herramienta se posiciona como un aliado crucial para el éxito de los usuarios. En conclusión, el desarrollo de una aplicación web para entrevistas técnicas representa una solución innovadora y necesaria para enfrentar los desafíos actuales en la evaluación de habilidades en programación.

2.2. Alcance del Proyecto

2.2.1. Declaración del Alcance

El propósito del proyecto consiste en desarrollar un prototipo de plataforma web que permita la simulación de entrevistas técnicas, específicamente dirigidas a candidatos en el área de programación. Estas entrevistas están diseñadas para evaluar tanto conocimientos teóricos como habilidades prácticas. Los tipos de entrevistas técnicas que se incluirán son:

Evaluación Teórica:

Se llevarán a cabo preguntas orientadas a evaluar el conocimiento o dominio sobre estructuras de datos, algoritmos, bases de datos, paradigmas de diseño de software y fundamentos de ciencias computacionales.

El propósito de esta evaluación es medir la comprensión de los conceptos abordados y la habilidad para aplicar las teorías en contextos prácticos.

Codificación en Vivo:

Los candidatos deberán resolver tareas de programación en un Entorno de Desarrollo Integrado (IDE), lo que les permitirá escribir, ejecutar y depurar programas durante la entrevista de evaluación.

El sistema evaluará aspectos tales como la eficiencia, la corrección y la claridad del código generado.

Resolución de Problemas Técnicos:

Se busca que los candidatos aborden casos prácticos y problemas del mundo real que requieren la formulación de soluciones técnicas, basadas en la aplicación de principios de ingeniería de software.

La plataforma permitirá a los usuarios seleccionar el tipo de entrevista según sus necesidades, con niveles de dificultad ajustables para principiantes, intermedios y avanzados. También incluirá un módulo de retroalimentación personalizada, generada automáticamente por inteligencia artificial.

El diseño del sistema incluirá:

Front-End: Mediante una interfaz que sea intuitiva, accesible y que cumpla con los estándares de usabilidad y de calidad externa (ISO/IEC25010)[18].

Back-End: Que albergue la lógica, la seguridad y la gestión de datos de toda la plataforma web, cumpliendo los estándares de calidad interna y de protección de datos(ISO/IEC 27001 y Ley 1581 de 2012)[19], [20].

El objetivo principal de este prototipo es proporcionar a los postulantes un conjunto integral de herramientas que les permita practicar entrevistas técnicas. Esto busca reducir la ansiedad que tales situaciones generan y, al mismo tiempo, mejorar su competitividad en el mercado laboral.

3. Marco Referencial

3.1. Marco Teórico

El desarrollo de aplicaciones de entrevistas con inteligencia artificial se construye y fundamenta en una variedad de teorías que guían la creación del sistema de evaluación más efectivo, justo y éticos. Estas plataformas combinan lo último en tecnología de procesamiento de información y aprendizaje autónomo para la simulación de entrevistas laborales, dando a los candidatos una experiencia que permita la evaluación de forma práctica y personalizada. Algunas de las principales teorías que se involucran en la creación de estas plataformas son el procesamiento de información y el procesamiento del lenguaje natural (PLN). Que permite a la IA interpretar y analizar repuestas de usuarios en tiempo real. Otras, como la teoría de la evaluación basada en competencias para una evaluación precisa y objetiva, la unión de estas teorías da la base para la creación de una aplicación bien estructurada.

3.1.1. Teoría de la Evaluación Basada en Competencias

La teoría del procesamiento de la información (IPT) es un proceso que explica cómo los seres humanos absorben y aplican el conocimiento en el entorno a través de actividades psicológicas internas como la percepción sensorial, la atención, el reconocimiento, la transformación y la memoria, lo que se conoce como procesamiento de la información. Los psicólogos de esta escuela intentan asumir la estructura y el proceso del sistema cognitivo humano, incluida la esencia de la investigación sobre la cognición humana, la atención, la percepción, la comprensión, la memoria, la resolución de problemas, etc. [21]. Se ha convertido en la corriente principal de la psicología y la educación analizar científicamente el proceso cognitivo y la estructura cognitiva de los individuos involucrados en diversas actividades mentales (como la percepción, la memoria y la resolución de problemas) para comprender y predecir el comportamiento individual [22]. Por lo tanto, la enseñanza eficaz inevitablemente debe prestar atención a las habilidades mentales y los procesos mentales de los candidatos.

3.1.2. Teoría del procesamiento de la información

La preparación y la determinación de la elegibilidad de los estudiantes son dos preocupaciones principales en el proceso de colocación laboral. Un proceso típico de reclutamiento en un campus implica evaluar a los candidatos estudiantiles en varias competencias [23] como aptitud y habilidades blandas, conocimiento de la materia, historial de desempeño académico, etc. Entre estas, el conocimiento de la materia, también conocido como competencia de dominio (CD), juega un papel crucial en la evaluación de la elegibilidad de un estudiante para el reclutamiento. El conocimiento sobre la elegibilidad para un trabajo preferido ayuda a los estudiantes en la preparación para la prueba de colocación, las pruebas de preselección y las entrevistas. En la mayoría de las situaciones, la evaluación de la competencia laboral (CJ) se basa principalmente en el Promedio de Calificaciones Acumuladas (CGPA) obtenido

por un estudiante en las materias de estudio relevantes; esto es seguido por una prueba escrita y/o una entrevista oral. Los criterios de CGPA no siempre capturan con precisión el conocimiento asimilado de la materia que posee un estudiante, para calificar para un trabajo específico. Diferentes trabajos requieren excelentes puntajes de evaluación de desempeño en diferentes dominios. Por ejemplo, los estudiantes que generalmente son inteligentes o creativos en aplicaciones prácticas del conocimiento (por ejemplo, trabajo de laboratorio, programación de computadoras), no necesariamente son inteligentes teóricamente y tienden a obtener calificaciones más bajas en las materias. A pesar de esto último, terminan teniendo una carrera exitosa cuando se les brindan oportunidades compatibles. De manera similar, los estudiantes que tienen un alto rendimiento académico, pueden fallar en las entrevistas de trabajo debido a una experiencia laboral práctica inadecuada. Cada intento fallido en una entrevista de trabajo degrada la autoeficacia [24] del estudiante. La falta de preparación para una oportunidad laboral causa ansiedad por la elección [25] , que es una de las principales razones de la indecisión profesional, observada en los recién graduados (en la escuela secundaria, la universidad o las instituciones vocacionales). La gran mayoría de los estudiantes encuentran el proceso de entrevista una experiencia desafiante, debido a la tutoría incompetente, junto con una preparación deficiente para la evaluación de colocación laboral [26] . Los estudiantes necesitan una forma fácil y rápida de autoevaluar su conocimiento del dominio y estar preparados para una oportunidad laboral determinada. Tales situaciones se pueden remediar si se les ofrecen a los estudiantes evaluaciones que brinden retroalimentación personalizada detallada sobre sus fortalezas y debilidades.

3.1.3. Teoría del Procesamiento del Lenguaje Natural (PLN)

Con el amanecer de la era de Internet, especialmente en el área de sistemas móviles [27] , el área de sistemas integrados [28] y la tecnología moderna de multiprocesadores [29] , la civilización ha experimentado cambios profundos y rápidos que estamos experimentando hoy más que nunca [30] . La investigación de PNL, en particular, no ha evolucionado al mismo ritmo que otras tecnologías en las últimas décadas de años. PNL es un campo importante en la ciencia informática y la inteligencia artificial, que tiene como objetivo estudiar varias teorías y métodos que pueden lograr una comunicación efectiva entre humanos y computadoras con lenguaje natural. El uso del lenguaje natural para comunicarse con las computadoras tiene una importancia de aplicación práctica muy importante, así como una importancia teórica revolucionaria. Más recientemente, se ha demostrado que la aplicación de tecnología preentrenada lleva la cantidad de tareas de PNL al nivel de vanguardia.

3.1.4. Estado del Arte

En un panorama en el que dominar el arte de las entrevistas de trabajo es sinónimo de éxito profesional, el simulador de entrevistas surge como una herramienta transformadora que revoluciona la forma en que las personas perfeccionan sus habilidades para las entrevistas. Además de ofrecer asesoramiento teórico, este simulador sirve como un puente dinámico que proporciona un entorno virtual seguro, donde los solicitantes de empleo pueden transmitir el conocimiento teórico en una aplicación práctica. Al reconocer la brecha que a menudo existe entre la orientación convencional para entrevistas y la aplicación de habilidades en el mundo real, el Simulador de entrevistas se convierte en un catalizador para la mejora. Va más allá de lo convencional, al permitir a los usuarios personalizar su experiencia, ofreciendo una gran variedad de opciones para personalizar escenarios de entrevistas, seleccionar temas relevantes para su campo y calibrar niveles de dificultad en consonancia con sus aspiraciones profesionales. Esta personalización no solo mejora la relevancia de la práctica, sino que también aborda los desafíos únicos que las personas pueden enfrentar en sus dominios profesionales específicos. La eficacia del simulador de entrevistas se basa en su inteligencia artificial (IA). Los usuarios se benefician de una retroalimentación inmediata en su desempeño de la entrevista, debido a que cada pregunta teórica mal contestada tendrá su corrección, mencionando cuál era la correcta y el porqué lo es. Este mecanismo de retroalimentación permite a los usuarios realizar una autoevaluación, identificando las fortalezas y las áreas de mejora. Como herramienta para la preparación de entrevistas, el Simulador se destaca como un elemento innovador en el competitivo mercado laboral. No solo ofrece una visión general del proceso de entrevistas, sino que también ofrece una ruta integral hacia el éxito. En un panorama en el que conseguir el puesto deseado depende de la capacidad de una persona para superar entrevistas de manera eficaz, el Simulador de entrevistas surge como un poderoso aliado, que brinda a las personas las habilidades y la confianza necesarias para sobresalir en el ámbito profesional en constante evolución.[31]

3.1.5. Antecedentes

Ahora ha evolucionado el proceso de contratación en puestos que utilizan tecnología. Para utilizar este tipo de tecnología, necesitamos un panel de expertos calificados. Hay algunas plataformas de codificación que son populares en este segmento, como Leet Code, Hacker Rank y Code Forces, que son algunas de las plataformas en las que puede realizar codificación. Hoy en día, los solicitantes tienen acceso a casi todos los recursos, ya que están disponibles en Internet, a través de los cuales pueden practicar todo tipo de preguntas de codificación y pueden mejorar sus habilidades. Las estructuras de datos y los algoritmos son el enfoque principal de todos los solicitantes a la hora de aumentar sus habilidades y para ellos hay cursos en línea disponibles como Coursera, edX y Udemy. Con el conocimiento básico de las estructuras de datos y algoritmos de los cursos, los solicitantes pueden establecer una buena base en estos temas. El formato de la entrevista ha evolucionado demasiado y la brecha entre el conocimiento teórico y el conocimiento práctico es enorme, pero al igual que las plataformas de codificación, también tenemos algunas plataformas como Pramp y Interviewing.io para este tipo de servicios [32] , [33] . Los servicios brindan a los estudiantes todo tipo de conocimientos antes de ir a una entrevista real. Estos servicios tienen expertos que ayudan mucho a los estudiantes. Además de los recursos de Internet para estructuras de datos y algoritmos, también tenemos libros como “Cracking the Coding Interview” de Gayle Laakmann McDowell y sitios web como Geeks-for-Geeks que brindan una amplia información sobre estructuras de datos y algoritmos. Los candidatos también comparten sus experiencias de entrevistas, tipos de preguntas realizadas, temas específicos de la empresa en plataformas como Glassdoor y Blind. En conclusión, la necesidad de personas con experiencia para realizar este tipo de entrevistas de alto nivel está provocando una evolución en la preparación para las entrevistas técnicas.

La siguiente tabla proporciona un análisis comparativo de diversas plataformas que se utilizan para la preparación de entrevistas técnicas en programación. Las plataformas evaluadas incluyen Pramp, LeetCode, Interviewing.io y HackerRank, cada una destacándose por sus características y enfoques particulares.

Plataformas	Idioma Disponible	ESTADO	Costo	Pros	Contras
Pramp	Inglés	Vigente	Freemium	Simulación en tiempo real, Retroalimentación entre pares	No son evaluaciones profesionales, sin entrevistas personalizadas
LeetCode	Inglés, Chino	Vigente	Freemium	Gran base de datos, elegir la dificultad a tu gusto	No ofrece simulaciones completas, la retroalimentación es estandarizada
Interviewing.io	Inglés	Vigente	Pago	Entrevistas con profesionales, Evaluación más precisa	No accesible para todos, mayormente usado por gente con experiencia
HacKerRank	Inglés	Vigente	Freemium	Gran cantidad de problemas, evaluaciones usadas por empresas	Falta de personalización para estudiantes, se enfoca más en lo técnico

Tabla 2: Comparación Antecedentes.
Fuente: Elaboración propia.

Para concluir esta sección, a pesar de la disponibilidad de múltiples plataformas que ofrecen simulaciones y retos técnicos, muchas de ellas presentan limitaciones importantes, tales como la falta de personalización, una retroalimentación poco detallada y una integración deficiente entre elementos teóricos y prácticos. Por ejemplo, herramientas como LeetCode y HackerRank se centran exclusivamente en la resolución de problemas técnicos, mientras que Pramp e Interviewing.io brindan simulaciones que no abarcan adecuadamente las necesidades particulares de los candidatos. Nuestro proyecto se distingue por no ser redundante, ya que fusiona características innovadoras, entre ellas la personalización de entrevistas en función del nivel del candidato, simulaciones elaboradas por educadores y una retroalimentación detallada respaldada por inteligencia artificial. Este enfoque singular aborda las carencias que se encuentran en las plataformas actuales y proporciona un valor añadido significativo al centrarse en la experiencia del usuario y en la mejora constante de sus habilidades, tanto técnicas como teóricas. De este modo, se posiciona como una solución única y necesaria en el ámbito de las entrevistas técnicas.

3.2. Marco Conceptual

3.2.1. Evaluador de entrevistas simuladas basado en IA

En los últimos años, ha aumentado la demanda de herramientas eficientes de preparación para entrevistas, ya que los candidatos buscan mejorar su desempeño en mercados laborales altamente competitivos. Los métodos tradicionales, como las entrevistas simuladas entre pares y los bancos de preguntas genéricos, a menudo carecen de la personalización y la retroalimentación en tiempo real necesarias para mejorar significativamente la preparación de un candidato. Para abordar esta brecha, se ha empleado inteligencia artificial (IA) para crear plataformas dinámicas de entrevistas simuladas que simulan entornos de entrevistas reales y ofrecen retroalimentación personalizada para ayudar a los candidatos a perfeccionar sus habilidades. El sistema de entrevistas con IA desarrollado en este artículo tiene como objetivo ofrecer a los usuarios una experiencia de entrevista simulada realista e interactiva. Utilizando procesamiento del lenguaje natural (PNL) y algoritmos de aprendizaje automático, el sistema formula preguntas relevantes para el contexto basadas en las del usuarios rol laboral elegido. Evalúa las respuestas del usuario en varias dimensiones, como contenido, emoción y confianza, y ofrece comentarios personalizados para mejorar el rendimiento futuro. Este trabajo amplía las contribuciones de Chou et al. (2022) y otros trabajos relacionados, en los que se han empleado plataformas basadas en IA en el análisis del desempeño en entrevistas. Si bien en el pasado se han diseñado otros sistemas para beneficiar a los reclutadores, nuestro sistema está orientado a ayudar a los candidatos a mejorar con la práctica y una evaluación exhaustiva. El modelo de IA evalúa las habilidades de comunicación, el campo de conocimiento y las señales no verbales como el tono emocional y el nivel de confianza y les proporciona una revisión completa que los métodos típicos no pueden proporcionar. La incorporación de IA para escanear señales emocionales y de confianza mientras se realiza una entrevista proporciona una ventaja especial para los solicitantes. Investigaciones anteriores han señalado la capacidad de la IA en entrevistas en línea, especialmente en mediciones métricas cognitivas y emocionales, que son determinantes vitales para el éxito de la entrevista. Nuestro sistema se esfuerza por llenar el vacío brindando a los solicitantes comentarios significativos basados en datos que puedan ayudar a mejorar su desempeño real en las entrevistas[34].

Tipos de aplicaciones de entrevistas

Inteligencia artificial en la transcripción de entrevistas

Las entrevistas, fundamentales para el ejercicio periodístico y la investigación cualitativa, capturan el significado profundo del pensamiento de la humanidad. En 2023, las herramientas de inteligencia artificial (IA) se popularizaron, y vimos que se las empezó a usar en la grabación, transcripción y subtitulado de discursos. El objetivo del estudio es identificar la IA más adecuada para transcribir grabaciones en español y priorizar la completitud de tareas, la eficacia y la eficiencia. La IA seleccionada se aplicará a un corpus de 450 entrevistas cortas, para luego codificarlas y analizar su contenido[35].

Simulador de entrevista de trabajo virtual impulsado por IA que utiliza procesamiento de lenguaje natural

El simulador de entrevista de trabajo virtual impulsado por IA aborda esta brecha mediante el uso de tecnologías avanzadas, en particular el procesamiento del lenguaje natural (PLN), para crear una plataforma de práctica de entrevistas interactiva en tiempo real. Este simulador genera preguntas específicas del rol alineadas con la trayectoria profesional deseada por el usuario y evalúa las respuestas en tiempo real a través de análisis de sentimientos y evaluaciones de coherencia. Ofrece una visión constructiva, retroalimentación personalizada que resalta tanto las fortalezas como las áreas de mejora. El diseño de aprendizaje adaptativo del simulador ajusta continuamente la dificultad de las preguntas en función del rendimiento del usuario, lo que garantiza una progresión fluida. Con el tiempo, aprende de los comportamientos individuales de los usuarios para mejorar su retroalimentación, haciendo así que la experiencia sea cada vez más personalizada y efectiva. Con una interfaz fácil de usar, el sistema no requiere conocimientos técnicos para su funcionamiento. Al combinar practicidad con tecnología de inteligencia artificial de vanguardia, esta innovadora herramienta empodera a los usuarios al inculcarles la confianza y las habilidades necesarias para sobresalir en las entrevistas, cerrando eficazmente la brecha entre la preparación y el empleo. El simulador representa una solución transformadora para la preparación profesional, revolucionando el proceso de preparación de entrevistas[36].

sistema interactivo de preparación de entrevistas basado en IA

Smart Prep, una plataforma interactiva donde los usuarios pueden elegir interactuar con agentes o un equipo profesional para practicar las entrevistas. Durante cada sesión, el personal de Smart Prep y los agentes de IA hacen preguntas específicas del dominio, a las que los usuarios responden escribiendo o diciendo sus respuestas. Luego, los agentes analizan estas respuestas y brindan a los usuarios comentarios constructivos y sugerencias para mejorar sus habilidades de entrevista. Desarrollado por OpenAI ChatGPT y con capacidades de voz y texto, Smart Prep proporciona un entorno virtual y útil para que los usuarios practiquen, mejorando el rendimiento de sus entrevistas. Construido con Svelte para front-end y Flask para back-end, Smart Prep demuestra un potencial significativo como una valiosa herramienta de capacitación en entornos profesionales y educativos[37].

3.2.2. Modelos y Técnicas de Aprendizaje Automático

La mejor manera de prepararse para una entrevista es revisar los diferentes tipos de posibles preguntas que le harán durante la entrevista y practicar cómo responderlas. Un sistema de coaching de entrevistas intenta simular a un entrevistador para proporcionar sesiones simuladas de simulación de prácticas de entrevistas para los usuarios. Los sistemas tradicionales de entrenamiento para entrevistas proporcionan algunos comentarios, incluida la preferencia facial, el movimiento de cabeza, el tiempo de respuesta, la velocidad de habla y el volumen, para que los usuarios conozcan su propio desempeño en la entrevista simulada[38].

Using Generative AI for Interview Simulations

usa ChatGPT (y otros LLMs) para simular entrevistas con sujetos mayores, con el objetivo de mejorar habilidades de investigación, empatía y entrevistas reales[39].

Conversate: Supporting Reflective Learning in Interview Practice

Esta propuesta va más allá de “dar feedback”: permite un diálogo interactivo con el agente IA para que el usuario reflexione, mejore sus respuestas y aprenda activamente[40].

AI-Enhanced Interview Simulation in the Metaverse

Propone un módulo de simulación de entrevista en realidad virtual, con modelos de IA integrados para hacer la experiencia inmersiva más realista[41].

A Smart Interview Simulator Using AI Avatars and Real-Time Feedback Mechanisms

Plataforma que usa un avatar 3D para simular entrevistas, genera preguntas con GPT-3 y da retroalimentación inmediata al usuario durante la simulación[42].

Integrating Resume Analysis, HR Simulation, and Technical Skill Assessment

Combina análisis de currículum con simulador de entrevistas, reconocimiento de voz, detección de emociones con CNN, etc., para evaluación objetiva y retroalimentación[43].

4. Diseño y prototipo

La plataforma InterviewSim se desarrolló empleando una estructura de servidor y cliente, con el objetivo de crear un diseño que sea comprensible, modular y que permita el crecimiento, facilitando así el cuidado y el avance del proyecto. Esta disposición posibilita que cada componente, como la interfaz, la lógica del negocio y la administración de datos, funcione de manera autónoma pero coordinada, lo que optimiza el rendimiento global de la aplicación.

La interfaz de usuario, que constituye el frontend, fue desarrollada en React.js[44], donde los usuarios pueden interactuar con la plataforma de modo intuitivo. A través de esta interfaz, los usuarios pueden participar en entrevistas simuladas, responder a preguntas y recibir comentarios instantáneos del sistema. Este componente se conecta con el servidor mediante API RESTful[45], asegurando una experiencia dinámica, fluida y segura.

El backend, que está construido utilizando Node.js[46] y Express[47], se encarga de la lógica central del sistema. Sus responsabilidades incluyen la autenticación de usuarios, la interacción con los módulos de inteligencia artificial, la gestión de las entrevistas y la elaboración de recomendaciones personalizadas.

La capa de datos se gestionó mediante MongoDB[48], encargada del almacenamiento y recuperación eficiente de la información, incluyendo usuarios, entrevistas, resultados y registros de interacción con la IA. Cada solicitud enviada desde el cliente se procesa en el servidor, donde se ejecutan los algoritmos de análisis y se devuelven resultados personalizados al usuario.

Gracias a esta arquitectura en capas, el sistema logra una alta cohesión interna, flexibilidad para escalar a futuro y una comunicación eficiente entre cliente y servidor. En la figura 1 se muestra el esquema general de la arquitectura del sistema, que ilustra la interacción entre sus componentes principales y el flujo de datos que sustenta el funcionamiento de InterviewSim.

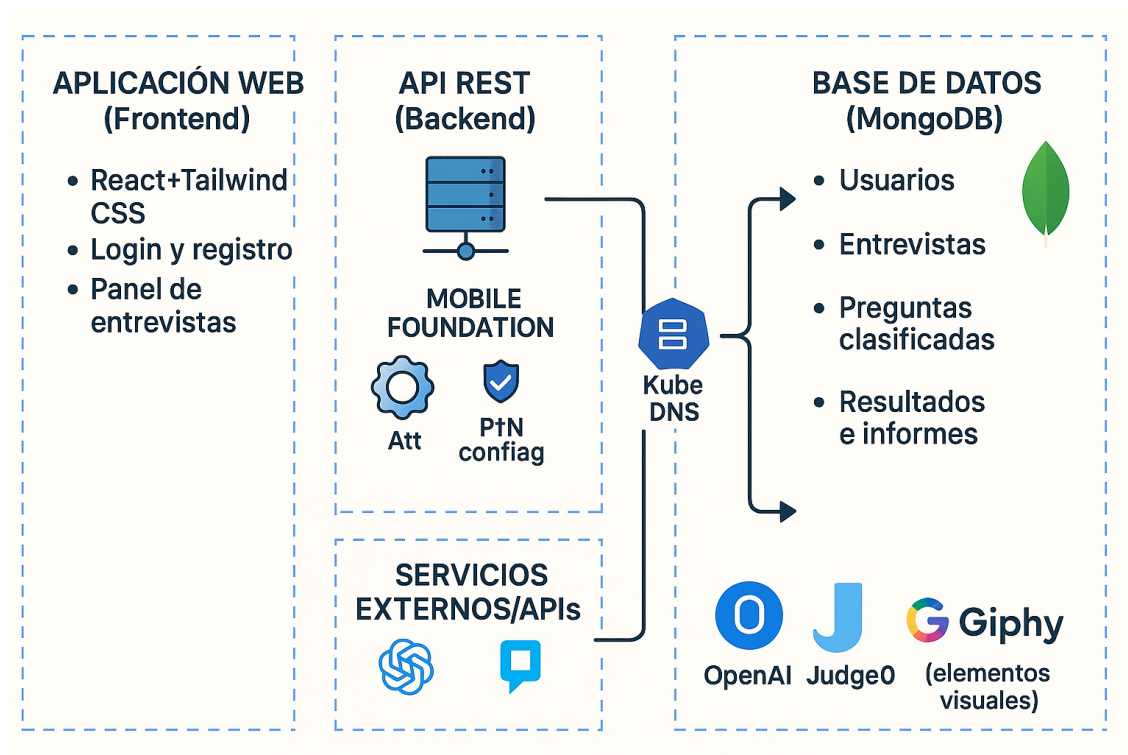


Figura 1: Diagram de arquitectura

Fuente: Elaboración propia.

4.1. Mockups

En el proceso de desarrollo de la interfaz de InterviewSim, se diseñaron prototipos visuales con la finalidad de estructurar las pantallas clave y definir cómo se distribuirían sus características esenciales. Esta metodología se basó en las necesidades identificadas de los estudiantes, quienes buscan mejorar sus habilidades en entrevistas técnicas, además de las tendencias actuales en el diseño de plataformas educativas y tecnológicas. El objetivo primordial fue garantizar una experiencia de usuario que sea intuitiva, actual y sencilla de manejar.

Se dio prioridad a un diseño pulcro y minimalista, con una navegación suave que facilitara al usuario el acceso inmediato a las simulaciones, resultados y sugerencias. Cada pantalla se diseñó con el objetivo de captar la atención del usuario y simplificar la interacción con el sistema de inteligencia artificial responsable de formular preguntas y proporcionar comentarios personalizados.

Los prototipos fueron sometidos a pruebas con usuarios seleccionados, quienes brindaron comentarios sobre la claridad de la interfaz, la organización de los elementos y la facilidad de uso. Gracias a esta retroalimentación, se realizaron mejoras enfocadas en la coherencia visual, la fluidez del flujo de trabajo y la reducción de pasos innecesarios. Además, se consideró la inclusión de tutoriales interactivos para los nuevos usuarios, con el fin de orientar su primera interacción con la plataforma y maximizar la comprensión de sus funcionalidades.

Finalmente, se presentan los diseños finales de la aplicación web en las figuras correspondientes.

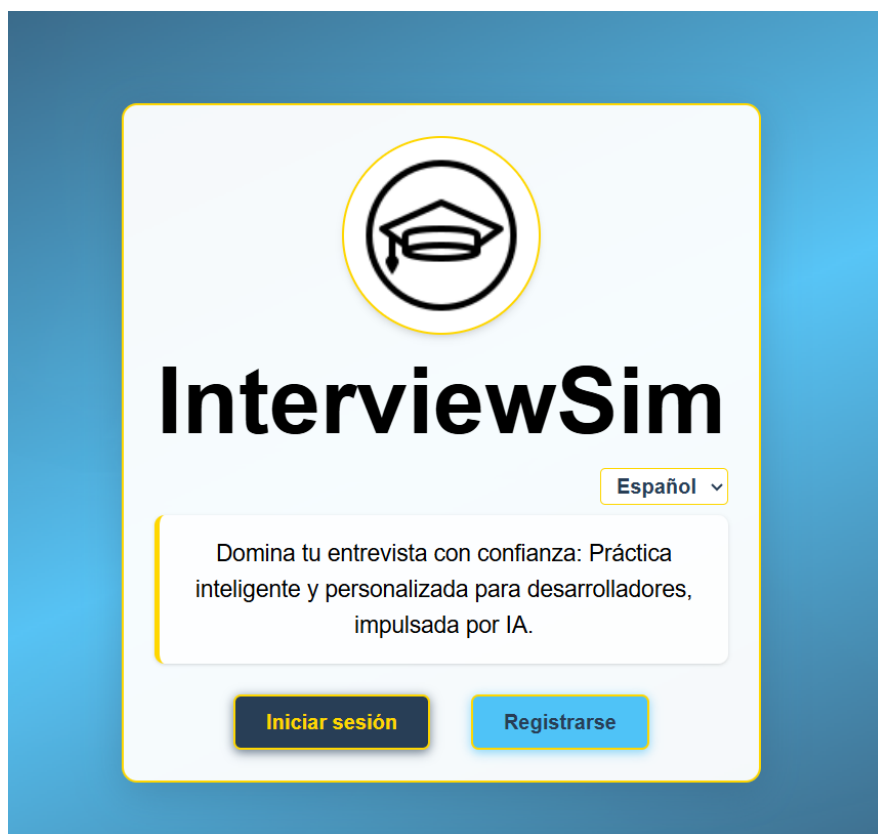


Figura 2: Mockup inicio de aplicación
Fuente: Elaboración propia.

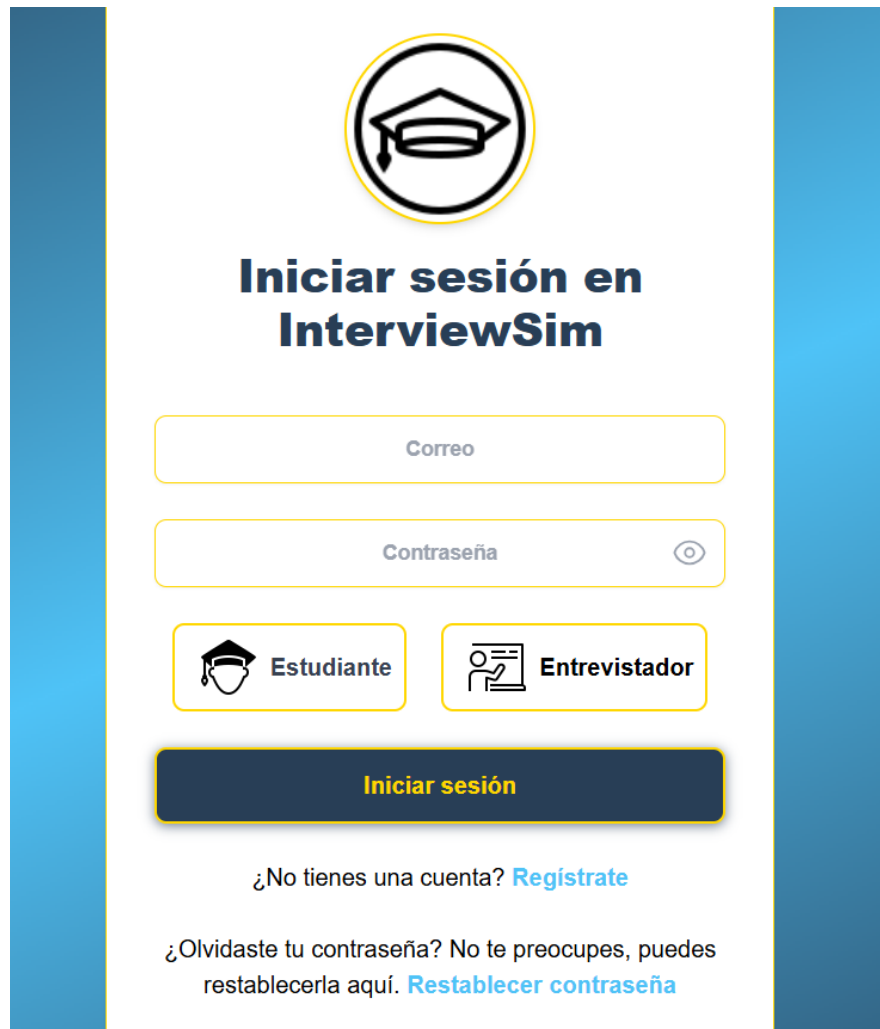


Figura 3: Mockup login
Fuente: Elaboración propia.



Figura 4: Mockup area de estudiantes
Fuente: Elaboración propia.

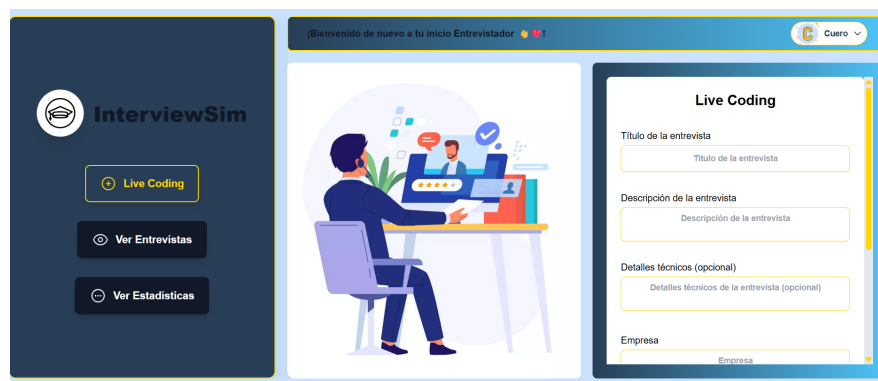


Figura 5: Mockup area entrevistadores
Fuente: Elaboración propia.

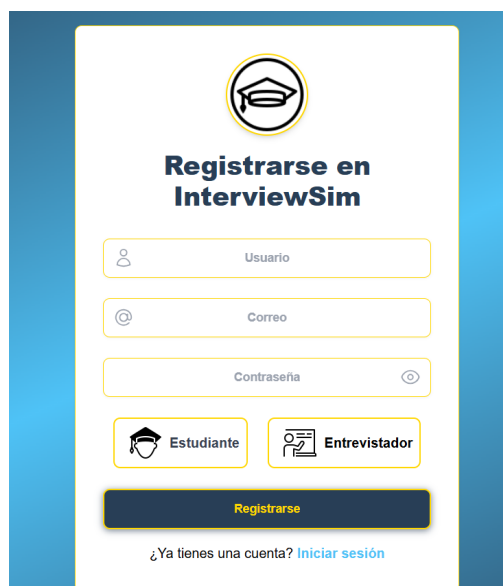


Figura 6: Mockup registro de usuario
Fuente: Elaboración propia.



Figura 7: Mockup recuperar contraseña
Fuente: Elaboración propia.

4.2. Requisitos de software

4.2.1. Requisitos funcionales

En el desarrollo de InterviewSim, es fundamental especificar con precisión los requisitos funcionales, porque esto define de forma clara las funcionalidades y conductas que el sistema necesita llevar a cabo para atender apropiadamente las demandas del usuario. En este proyecto, los requisitos son cruciales para definir las habilidades del sistema en términos de manejar perfiles de usuario con distintos roles, crear preguntas de programación a través de inteligencia artificial, guardar y recuperar entrevistas personalizadas y brindar retroalimentación sobre el rendimiento del participante. Además, consideran la comunicación fluida entre el backend y el frontend, la interacción con la API de OpenAI[49] y el manejo seguro de datos en la base de datos..

En este contexto, los requisitos funcionales se establecen como un elemento central para garantizar el éxito del proyecto, delineando las capacidades clave necesarias para mejorar la experiencia del usuario y cumplir con los objetivos técnicos y estratégicos de la aplicación.

Gestión de usuarios

La aplicación debe permitir el manejo completo de usuarios finales, docentes y administradores, garantizando la autenticación y personalización de la experiencia.

- Registro de nuevos usuarios mediante correo electrónico y contraseña.
- Inicio de sesión y cierre de sesión seguro.
- Actualización de datos personales y académicos en el perfil del usuario.
- Visualización de estadísticas personales y progreso dentro de las entrevistas simuladas.
- Roles diferenciados con distintos niveles de acceso.

Simulación de entrevistas

El sistema debe permitir la creación, ejecución y almacenamiento de simulaciones de entrevistas, tanto automatizadas por IA como dirigidas por docentes.

- Generación dinámica de entrevistas con base en el lenguaje o tema seleccionado.
- Formulación automática de preguntas por la inteligencia artificial.
- Evaluación inmediata con retroalimentación generada por el modelo de IA.
- Almacenamiento de resultados y calificaciones para consulta posterior.

Gestión de entrevistadores y entrevistas guiadas

El sistema debe permitir que entrevistadores creen entrevistas personalizadas.

- Creación de plantillas de entrevistas por parte de los entrevistadores.

- Poder ver qué cantidad de personas respondieron su entrevista.
- Poder ver las entrevistas que ha creado.

Módulo de inteligencia artificial

El sistema debe integrar modelos de procesamiento de lenguaje natural para analizar y evaluar las respuestas de los usuarios.

- Análisis semántico de las respuestas para identificar claridad, coherencia y nivel técnico.
- Generación automática de retroalimentación en tiempo real.

Gestión de base de preguntas

El sistema debe mantener una base de datos estructurada con preguntas clasificadas por nivel, tema y tipo de entrevista.

- Creación, edición y eliminación de preguntas desde el panel administrativo.
- Categorización por lenguaje de programación, dificultad y tema.
- Control de versiones y trazabilidad de modificaciones.

Visualización y experiencia de usuario

El sistema debe proporcionar una interfaz interactiva que facilite la navegación, la visualización de resultados y la interacción con la IA.

- Interfaz dinámica que permita iniciar simulaciones, visualizar resultados y acceder a reportes.
- Presentación visual de resultados mediante gráficos, métricas y recomendaciones.
- Navegación fluida y coherente entre los módulos del sistema.

4.2.2. Requisitos no funcionales

Seguridad

- Los datos personales de los usuarios deben cumplir con las normas de privacidad vigentes.

Rendimiento

- El sistema debe procesar respuestas y generar retroalimentación en menos de 5 segundos.

Disponibilidad

- La plataforma debe estar disponible al menos el 99.5 % del tiempo.

Usabilidad

- La interfaz debe ser intuitiva, con navegación clara y accesible en un máximo de tres pasos por tarea.

Compatibilidad

- La aplicación debe ser compatible con los principales navegadores modernos.

Mantenibilidad

- Debe emplearse control de versiones para registrar cambios y gestionar despliegues.

Escalabilidad

- La arquitectura debe permitir la incorporación de nuevos módulos o funciones sin afectar el rendimiento del sistema.

Notificación y registro de errores

- Los errores deben mostrarse al usuario mediante mensajes amigables, evitando revelar información técnica.

4.3. Casos de uso

Los casos de uso fueron fundamentales para definir y estructurar las interacciones entre los usuarios y el sistema de simulación de entrevistas con inteligencia artificial. A través de estos escenarios, se identificaron los procesos clave que permiten a los usuarios realizar simulaciones, recibir retroalimentación automática y acceder a recomendaciones personalizadas según su desempeño.

Además, se establecieron los distintos roles dentro del sistema, como usuario y entrevistador detallando las acciones disponibles para cada tipo de actor. Esto permitió garantizar que la plataforma ofreciera una experiencia fluida, funcional y adaptada a las necesidades de cada perfil.

La definición de los casos de uso también contribuyó a anticipar excepciones y comportamientos no deseados, fortaleciendo la robustez, seguridad y usabilidad general del sistema.

En la siguiente figura se presenta el diagrama que representa los casos de uso de la aplicación. En este esquema se visualizan las principales interacciones entre los usuarios y los distintos módulos del sistema, destacando funcionalidades como el registro, la autenticación, la simulación de entrevistas, la evaluación automática mediante IA, la gestión de historial y la generación de recomendaciones personalizadas.

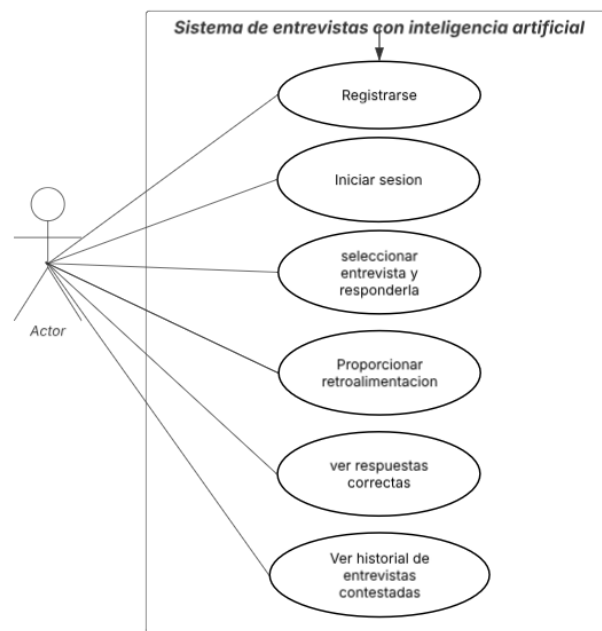


Figura 8: Diagrama de casos de uso

Fuente: Elaboración propia.

4.4. Base de Datos

El modelo de datos se creó con el objetivo de preservar la consistencia entre las diferentes entidades del sistema, facilitando el registro de los datos de los usuarios, el almacenamiento de entrevistas simuladas, el resguardo de las notas obtenidas y la administración de las sugerencias personalizadas. Esta estructura permitió la integración eficaz de los módulos de inteligencia artificial y retroalimentación automática, los cuales se basan en la persistencia y la actualización constante de los datos.

El enlace entre la base de datos y el back-end, creado en Node.js[46] con [47], facilitó la realización de tareas fundamentales como el registro y autenticación de usuarios, la generación y almacenamiento de entrevistas, la adquisición y almacenamiento de resultados de rendimiento y la creación de recomendaciones personalizadas para cada perfil. Esta interacción eficaz entre el servidor y MongoDB[48] aseguró un intercambio de datos continuo, seguro y fiable hacia el front-end, mejorando la experiencia del usuario.

La siguiente figura muestra un esquema relacional modificado para la estructura de datos de InterviewSim, en el que se ilustran las entidades fundamentales y sus vínculos dentro del sistema.

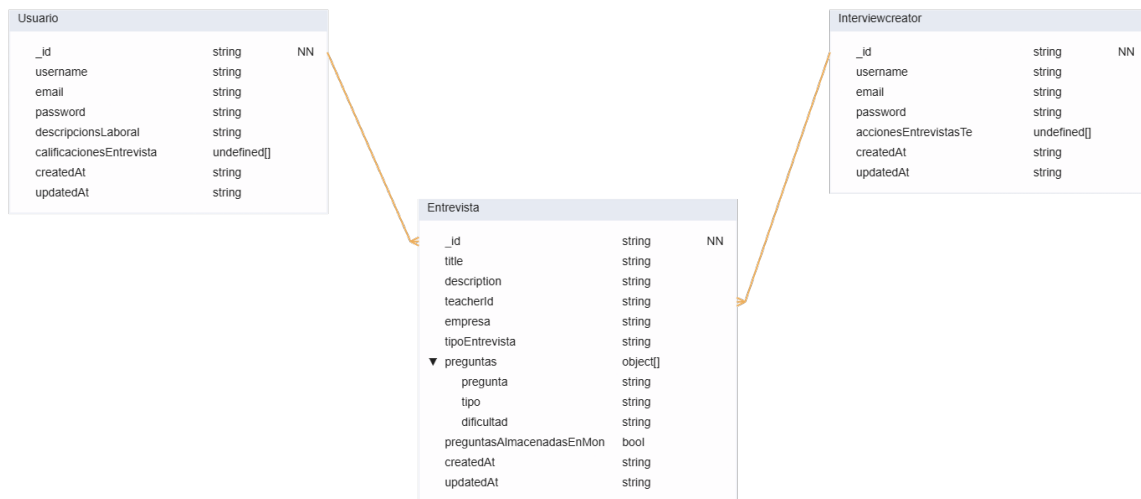


Figura 9: Modelo relacional
Fuente: Elaboración propia.

4.5. Stack tecnológico

4.5.1. Frontend

Para el desarrollo del frontend de InterviewSim, se empleó React v18.3.1[44], una biblioteca de JavaScript[50] ampliamente utilizada para la creación de interfaces de usuario interactivas y dinámicas. Su enfoque basado en componentes permitió construir una estructura modular, donde cada parte de la interfaz cumple una función específica, facilitando la mantenibilidad y escalabilidad del sistema.

El uso del Virtual DOM mejoró significativamente el rendimiento, reduciendo las operaciones innecesarias en el DOM real y optimizando la experiencia del usuario durante la interacción con la plataforma.

El diseño visual se implementó con tailwind CSS[51], y librerías complementarias que garantizan una apariencia moderna, limpia y adaptable a distintos dispositivos. Además, se empleó React Router DOM para gestionar la navegación entre las diferentes vistas, permitiendo una estructura de tipo SPA (Single Page Application), en la que los cambios de sección se realizan sin recargar completamente la página, logrando una experiencia fluida e ininterrumpida.

4.5.2. Backend

El backend fue desarrollado utilizando Node.js[46], un entorno de ejecución para JavaScript[50] que permite construir aplicaciones escalables y eficientes mediante un modelo no bloqueante basado en eventos. Esto resultó ideal para manejar las múltiples solicitudes concurrentes generadas por los usuarios durante sus interacciones con la aplicación.

Para la definición de rutas, gestión de peticiones HTTP y control de la lógica de negocio, se utilizó el framework Express.js, el cual proporcionó una base ligera, flexible y bien estructurada para organizar el flujo de datos entre el cliente, el servidor y la base de datos.

En lo referente al almacenamiento de información, se optó por MongoDB[48], una base de datos NoSQL orientada a documentos, que permitió un manejo ágil y flexible de los datos relacionados con usuarios, entrevistas, respuestas y resultados de análisis. La integración entre el backend y la gestión de la base de datos MongoDB se realizó mediante MongoDB Compass[52], una herramienta gráfica que permite administrar bases de datos NoSQL, realizar consultas y visualizar colecciones de forma interactiva.

Esta combinación tecnológica permitió desarrollar una arquitectura robusta, modular y escalable, asegurando un rendimiento óptimo y una comunicación fluida entre los distintos componentes del sistema InterviewSim.

5. Implementación

5.1. Descripción general del proceso de implementación

La fase de implementación constituyó una de las etapas más críticas del desarrollo de InterviewSim, ya que permitió materializar las funcionalidades diseñadas e integrar los distintos componentes tecnológicos del sistema. Durante este proceso se abordaron tanto aspectos técnicos del desarrollo del backend y frontend, como la conexión con los servicios de inteligencia artificial y las bases de datos. Asimismo, se realizaron múltiples iteraciones de prueba, detección de errores y optimización del código para garantizar la estabilidad, seguridad y coherencia funcional del sistema.

5.1.1. Problemas técnicos y soluciones implementadas

En el transcurso de la implementación de la aplicación InterviewSim, surgieron varios problemas técnicos que demandaron modificaciones en la tanto en el back-end como en el frontend, en la interacción entre las partes del sistema y en la organización del entorno de lanzamiento. a continuación, se detallan las dificultades más relevantes identificadas y las respuestas implementadas en cada situación.

1. Problemas con los textos generados por la IA

En las etapas iniciales del sistema, los textos generados por la inteligencia artificial carecían de concordancia con el contexto de las entrevistas o mostraban fallos en el formato. Esto sucedía por la manera en que estaban formulados los pedidos utilizados para interactuar con el modelo.

Solución: Se revisaron y modificaron los prompts enviados al modelo de IA, ajustando la redacción, el nivel de detalle y las instrucciones para obtener respuestas más precisas y adaptadas al tipo de entrevista. Además, se implementó una validación del texto recibido antes de mostrarlo al usuario.

2. Modificación de variables en el despliegue de la aplicación

A lo largo de la implementación del sistema, aparecieron fallos vinculados a las variables del entorno, particularmente en el enlace con la base de datos y en la disposición de los puertos de operación.

Solución: Se creó un archivo `.env` para centralizar las variables sensibles y se actualizó la configuración de los scripts de inicio tanto en el frontend como en el backend. También se separaron los puertos de ejecución para evitar conflictos al correr ambos entornos simultáneamente.

3. Errores en los botones e interacciones del frontend

Ciertas funcionalidades de los botones no actuaban de manera adecuada o producían acciones repetidas en la interfaz de usuario. Esto era resultado de fallos en los controladores de eventos y en la administración del estado en los componentes de React.

Solución: Se depuraron las funciones asociadas a los botones, optimizando el uso de hooks y corrigiendo las dependencias en los efectos. Además, se validó el correcto uso del Virtual DOM para evitar renderizados innecesarios.

4. Errores con las cookies y el inicio de sesión

Durante las pruebas de autenticación, se presentaron errores relacionados con la lectura y manejo de cookies, lo que impedía mantener la sesión activa o generaba conflictos entre usuarios.

Solución: Se implementó una configuración adecuada de cookies seguras `httpOnly` y `sameSite`, además de verificar el almacenamiento de tokens JWT en el cliente. También se ajustó la lógica de validación en el middleware de autenticación del backend.

5. Errores en el login por roles (estudiante y entrevistador)

El sistema manejaba dos tipos de roles (estudiante y entrevistador), y en algunos casos se presentaban fallos al validar las credenciales o redirigir al usuario correcto.

Solución: Se actualizó la lógica de autenticación en el backend para diferenciar los roles desde el inicio de sesión, generando tokens independientes según el tipo de usuario. Además, se añadieron rutas protegidas específicas para cada rol en el frontend utilizando React Router DOM.

6. Errores simples en el backend

Durante la implementación del servidor, se presentaron errores menores relacionados con rutas mal definidas, controladores incompletos o respuestas sin formato JSON.

Solución: Se realizó una revisión completa de las rutas y controladores, aplicando manejo de errores centralizado con middleware y mensajes de respuesta estructurados. También se usaron pruebas con Postman para validar cada endpoint antes de la integración.

7. Errores 404 y rutas no encontradas

En diferentes fases del desarrollo se produjeron errores 404 al intentar llegar a algunas vistas o puntos finales, tanto en el lado del usuario como en el servidor.

Solución: Se corrigieron las rutas en el enrutador de Express.js y en el archivo de configuración de React Router, garantizando que todas las vistas y recursos tuvieran rutas válidas y actualizadas. Además, se añadió una página personalizada para manejar los errores 404.

8. Problemas al traer preguntas creadas tras el despliegue

Después del despliegue, la aplicación presentaba fallos al intentar mostrar las preguntas almacenadas en la base de datos, lo que impedía cargar correctamente las entrevistas.

Solución: Se revisó la conexión con la base de datos y se ajustaron las consultas en Mongoose para asegurar que las preguntas fueran recuperadas según los filtros definidos. También se implementaron mensajes de error claros para facilitar la detección de fallos durante las pruebas.

En general, la etapa de implementación fue un proceso iterativo que permitió identificar errores, analizarlos y corregirlos mediante estrategias de depuración, pruebas unitarias y revisiones continuas. Gracias a estas mejoras, el sistema alcanzó un funcionamiento estable, con una correcta integración entre frontend, backend e inteligencia artificial.

5.2. Integración de la Inteligencia Artificial

La inteligencia artificial constituye el núcleo funcional del sistema InterviewSim, ya que permite generar preguntas personalizadas y brindar retroalimentación automática a los usuarios durante las entrevistas simuladas. Para este propósito, se implementó la API de OpenAI, una interfaz que ofrece modelos de lenguaje avanzados capaces de procesar texto natural y producir respuestas coherentes, contextualizadas y adaptadas a las necesidades específicas del sistema.

La interacción entre la aplicación y la API se realiza a través del backend, que actúa como intermediario seguro entre la interfaz de usuario y el modelo de inteligencia artificial. Esta capa intermedia garantiza la protección de las credenciales de acceso, el control del consumo de recursos y la validación de las solicitudes antes de enviarlas al modelo. Las peticiones se estructuran en formato JSON, incluyendo información clave como el tipo de entrevista (por ejemplo, de opción múltiple o de escritura de código), el nivel de dificultad (de 0 a 5) y el lenguaje de programación seleccionado. Una vez procesada la solicitud que debe ser puesto en el titulo,descripcion o detalles tecnicos, Luego el modelo genera una respuesta que es recibida, validada y enviada al frontend para su visualización por parte del usuario.

La integración de la inteligencia artificial no solo potenció la funcionalidad general de la aplicación, sino que también aumentó su valor pedagógico. Gracias al uso de la API de OpenAI, InterviewSim logró generar entrevistas adaptativas, mejorando la personalización de la experiencia de práctica para los usuarios. Además, la capacidad del modelo para ofrecer retroalimentación inmediata fortaleció el proceso de aprendizaje, permitiendo a los usuarios identificar sus errores y mejorar sus habilidades técnicas de manera autónoma. En conjunto, la implementación de la inteligencia artificial consolidó a InterviewSim como una herramienta innovadora y eficaz para la preparación de entrevistas técnicas.

En la siguiente figura se ilustra la arquitectura técnica del sistema y su integración con la API de OpenAI, donde se observa la interacción entre el frontend, backend, base de datos y los servicios de inteligencia artificial utilizados para generar preguntas y retroalimentación adaptativa.

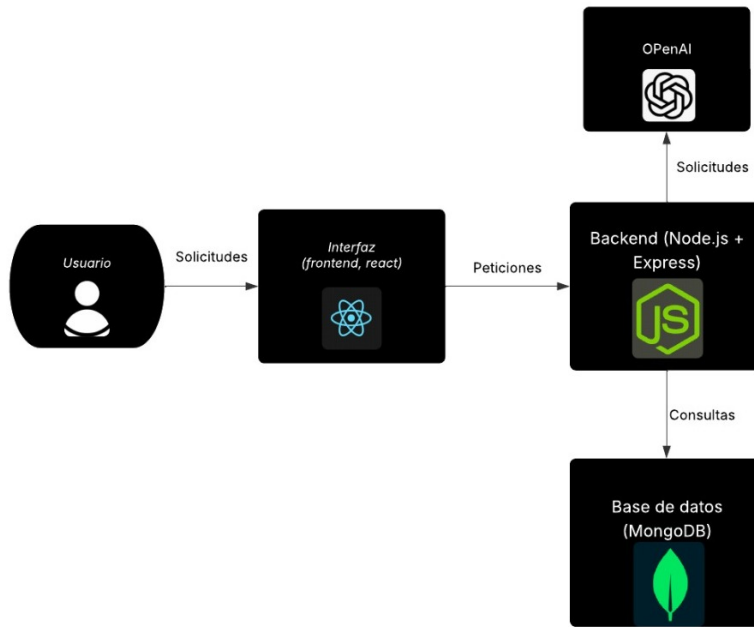


Figura 10: Diagram de arquitectura tecnica del sistema
Fuente: Elaboración propia.

5.3. Generación y envío de Prompts a la Inteligencia Artificial

Se creó un módulo que maneja la comunicación directa con la API de OpenAI en la etapa de implementación. Este módulo es el corazón del sistema para generar entrevistas automáticas. Las instrucciones textuales estructuradas que guían al modelo de lenguaje para generar preguntas que sean coherentes y se ajusten al nivel del usuario se llaman prompts.

El proceso de creación del prompt comienza eligiendo los parámetros que se utilizarán en la entrevista, como el tipo de pregunta, el lenguaje de programación y el nivel de dificultad. El sistema, con base en estos datos, elabora de manera dinámica una instrucción contextual que especifica la meta y la estructura esperada de la respuesta.

Por ejemplo, para una entrevista de nivel intermedio que seria de nivel 2 o 3 en JavaScript, el prompt se genera de la siguiente manera:

“Genera cinco preguntas de programación de nivel nivel sobre lenguaje en formato de opción múltiple, con cuatro opciones, una respuesta correcta y una breve explicación.”

El prompt se manda al servicio de OpenAI, por medio de una petición HTTP, después de ser creado. Para ello, se emplea el modelo gpt-3.5-turbo. La comunicación se lleva a cabo con límites establecidos de temperatura y cantidad máxima de tokens, asegurando un balance entre la precisión y la creatividad.

Se limpia el formato de la respuesta del modelo, se organiza la pregunta y se guarda en la base de datos MongoDB para que el frontend pueda recuperarla y exhibirla en la interfaz del simulador.

Ahora para mostrarlo de mejor manera se presentara un pseudocódigo del flujo de generación de preguntas:

Algorithm 1 GenerarPreguntasIA

Require: nivel, lenguaje, tipoPregunta**Ensure:** listaPreguntas

```
1: if numPreguntas no está definido then
2:   numPreguntas  $\leftarrow$  5
3: end if
4: prompt  $\leftarrow$  “Genera ” + numPreguntas + “ preguntas de programación de nivel ” + nivel
   + “ sobre ” + lenguaje + “ en formato de opción múltiple (A, B, C, D),” + “ indicando
   la respuesta correcta y una breve explicación.”
5: respuesta  $\leftarrow$  LlamarAPI_OpenAI(modelo = “gpt-3.5-turbo”, prompt)
6: if respuesta.estado = 200 then
7:   listaPreguntas  $\leftarrow$  Formatear(respuesta.datos)
8:   GuardarEnMongoDB(listaPreguntas)
9:   return listaPreguntas
10: else
11:   Mostrar(“Error al generar las preguntas de la IA”)
12: end if
```

5.3.1. Resultado final de la implementacion

La aplicación InterviewSim, gracias a esta arquitectura, crea automáticamente cinco preguntas de programación de opción múltiple que se ajustan al nivel y al lenguaje escogidos por el usuario. El flujo entre la API de OpenAI, el backend y la base de datos se integró por completo, obteniendo resultados que son coherentes, consistentes y reutilizables. Esto posibilitó que la funcionalidad principal del simulador se fortaleciera, brindando entrevistas realistas, evaluables y alineadas con las metas de aprendizaje del proyecto.

6. Pruebas

6.1. Plan de pruebas

El plan de pruebas tuvo como propósito evaluar el correcto funcionamiento, seguridad, rendimiento y experiencia de usuario del sistema InterviewSim. Este proceso permitió verificar el cumplimiento de los requisitos funcionales y no funcionales, garantizando la estabilidad y fiabilidad de la plataforma antes de su implementación final.

Se realizaron pruebas de usabilidad y funcionales para confirmar que las características principales —como el registro e inicio de sesión de usuarios, la simulación de entrevistas, la interacción con la inteligencia artificial, la generación de resultados personalizados y la gestión de información en la base de datos— operaran de manera correcta y coherente con los objetivos del sistema. Gracias a este proceso integral, fue posible detectar y corregir errores antes del despliegue final, asegurando que InterviewSim ofreciera una plataforma robusta, segura y con un rendimiento óptimo orientado a una experiencia de usuario satisfactoria.

6.1.1. Pruebas funcionales

Código	PF001
Título	Gestión de perfiles de usuario
Descripción	La aplicación debe permitir el manejo completo de usuarios finales y entrevistadores garantizando la autenticación y personalización de la experiencia: + Registro de usuario. + Visualización del perfil del usuario. + Actualización de datos del perfil. + Inicio de sesión. + Recuperación de contraseña. + Cierre de sesión.
Estado	Aprobado
Responsable	Nilson cuero - Desarrollador

Código	PF002
Título	Simulación de entrevistas
Descripción	El sistema debe permitir la creación, ejecución y almacenamiento de simulaciones de entrevistas, tanto automatizadas por IA como dirigidas por docentes: + Generación dinámica de entrevistas con base en el lenguaje o tema seleccionado. + Formulación automática de preguntas por la inteligencia artificial. + Evaluación inmediata con retroalimentación generada por el modelo de IA. + Almacenamiento de resultados y calificaciones para consulta posterior.
Estado	Aprobado
Responsable	Nilson cuero - Desarrollador

Código	PF003
Título	Gestión de entrevistadores
Descripción	El sistema debe permitir que entrevistadores creen entrevistas personalizadas: + Creación de plantillas de entrevistas por parte de los entrevistadores. + Poder ver qué cantidad de personas respondieron su entrevista: + Poder ver las entrevistas que creado.
Estado	Aprobado
Responsable	Nilson cuero - Desarrollador

Código	PF004
Título	Módulo de inteligencia artificial
Descripción	El sistema debe integrar modelos de procesamiento de lenguaje natural para analizar y evaluar las respuestas de los usuarios: + Análisis semántico de las respuestas para identificar claridad, coherencia y nivel técnico. + Generación automática de retroalimentación en tiempo real.
Estado	Aprobado
Responsable	Nilson cuero - Desarrollador

6.1.2. Pruebas de usabilidad

Código	PU001
Título	Facilidad de registro e inicio de sesión
Descripción	Se evaluará la facilidad con la que un usuario puede registrarse e iniciar sesión en la aplicación. Se verificará que: + Formulario de registro intuitivo y solicite solo información esencial. + Se ofrezca validación en tiempo real para evitar errores en el registro. + Mensajes de error en caso de credenciales incorrectas.. + Se proporcione una opción sencilla para recuperar la contraseña.
Estado	Aprobado
Responsable	Sebastian perez - Desarrollador

Código	PU002
Título	Claridad y coherencia en la navegación
Descripción	Se evaluará que la interfaz sea intuitiva y permita al usuario desplazarse fácilmente entre las distintas secciones de la aplicación. Se verificará que: + Los menús y botones estén claramente identificados y sean coherentes con sus funciones. + La estructura de navegación sea consistente en todas las pantallas. + Los íconos utilizados sean representativos y de fácil comprensión.. + El usuario pueda regresar fácilmente al inicio o sección anterior sin perder información.
Estado	Aprobado
Responsable	Sebastian perez - Desarrollador

Código	PU003
Título	Claridad en los mensajes de retroalimentación.
Descripción	Se evaluará la calidad de la retroalimentación que el sistema ofrece al usuario. Se verificará que: + Los mensajes de éxito, error o advertencia sean claros, visibles y comprensibles. + Se utilicen colores y símbolos coherentes con la naturaleza del mensaje. + Los mensajes se presenten en el momento adecuado, evitando interrumpir la interacción del usuario.
Estado	Aprobado
Responsable	Nilson cuero - Desarrollador

Código	PU004
Título	Evaluación de usabilidad en la simulación de entrevistas
Descripción	Se evaluará la facilidad de uso y comprensión del módulo de entrevistas simuladas. Se verificará que: + El usuario entienda claramente cómo iniciar y finalizar una entrevista. + Las preguntas sean presentadas de forma legible y con tiempos de respuesta adecuados. + Se muestre retroalimentación visual o textual al finalizar la simulación. + El usuario pueda repetir la simulación sin dificultad o pérdida de datos.
Estado	Aprobado
Responsable	Sebastian perez - Desarrollador

6.2. Análisis preliminar

La fase de pruebas del sistema tuvo como objetivo verificar el correcto funcionamiento de la aplicación web InterviewSim y garantizar que sus componentes cumplieran con los requerimientos establecidos durante el desarrollo. En esta etapa se evaluó el desempeño general del sistema, la interacción entre el frontend y el backend, la usabilidad de la interfaz, así como la efectividad de las funcionalidades implementadas para la simulación de entrevistas técnicas. Las pruebas se orientaron a asegurar que la aplicación ofreciera una experiencia fluida, confiable y de valor educativo para los usuarios, quienes constituyen el público objetivo principal del proyecto.

6.3. Perfil de los encuestados

Para la validación de la aplicación InterviewSim se aplicó una encuesta a un determinado número de participantes, principalmente estudiantes de Ingeniería en Sistemas y áreas afines. La mayoría de los encuestados contaba con conocimientos previos en lenguajes de programación como JavaScript, Python o Java, y con experiencia en el uso de plataformas web educativas. Esta población fue seleccionada por representar de manera directa al público objetivo del proyecto, garantizando que la retroalimentación obtenida reflejara la percepción y necesidades reales de los usuarios a los que la aplicación está dirigida.

6.4. Diseño de la encuesta

Para la evaluación de la funcionalidad y usabilidad de la aplicación web InterviewSim, se diseñó una encuesta estructurada con el propósito de recopilar información precisa sobre la experiencia de los usuarios al interactuar con el sistema. El instrumento se elaboró en la plataforma Google Forms y fue aplicado a un grupo de usuarios, quienes representan el público objetivo principal del proyecto.

La encuesta estuvo compuesta por preguntas cerradas y preguntas abiertas, lo que permitió obtener una combinación de datos cuantitativos y cualitativos. Las preguntas cerradas se formularon bajo una escala de valoración tipo Likert, abarcando aspectos como la facilidad de registro e inicio de sesión, la comodidad en la navegación, la claridad de la interfaz del editor de código, la adecuación del nivel de dificultad de las preguntas, la calidad de la retroalimentación automática y el atractivo visual del diseño. Por otra parte, las preguntas abiertas ofrecieron la oportunidad de recibir sugerencias y observaciones directas de los usuarios respecto a posibles mejoras en la plataforma.

El diseño de esta encuesta permitió evaluar de manera integral los componentes técnicos y de experiencia de usuario de la aplicación, identificando fortalezas relacionadas con la funcionalidad y posibles áreas de mejora en términos de accesibilidad, presentación visual y claridad en las recomendaciones ofrecidas. Este enfoque metodológico garantizó que la validación de la aplicación se sustentara en la percepción real de los usuarios finales, proporcionando información relevante para el perfeccionamiento del sistema.

6.5. Resultados de la encuesta

Pregunta: ¿Qué tan fácil fue registrarte e iniciar sesión en la aplicación?

La primera pregunta tuvo como propósito evaluar la accesibilidad inicial del sistema, específicamente los procesos de registro y autenticación de usuarios, los cuales son determinantes para garantizar una experiencia de uso fluida desde el primer contacto con la aplicación.

Según los resultados obtenidos, el 60 % de los encuestados calificó el proceso como “muy fácil” y el 40 % como “fácil”, sin registrarse respuestas en las categorías “regular”, “difícil” o “muy difícil”. Esta tendencia evidencia que la interfaz de ingreso fue percibida como clara, intuitiva y libre de complicaciones técnicas, lo cual favorece la adopción temprana del sistema.

En términos de usabilidad, estos resultados reflejan el cumplimiento de los principios de eficiencia y facilidad de aprendizaje, garantizando que los usuarios, en su mayoría estudiantes de ingeniería o desarrollo de software, pudieran acceder sin obstáculos a las funcionalidades principales de InterviewSim.

20 respuestas

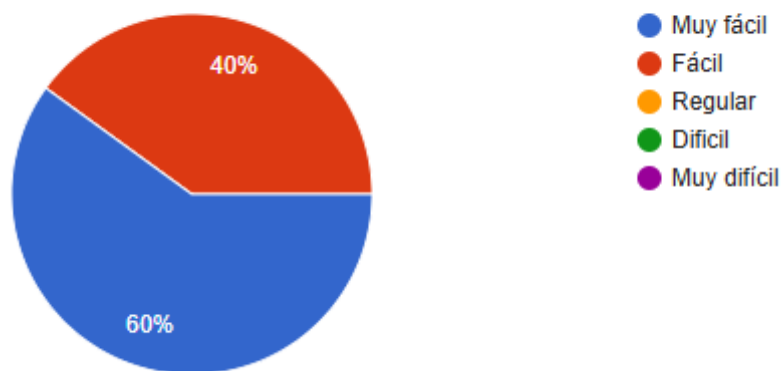


Figura 11: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion
Fuente: Elaboración propia.

Pregunta: ¿Qué tan cómodo te resultó iniciar sesión con correo/contraseña?

Esta pregunta buscó evaluar la percepción de comodidad del usuario frente al método de autenticación implementado en la aplicación, basado en credenciales tradicionales de correo y contraseña. Este aspecto resulta clave para determinar la aceptación del mecanismo de acceso y su impacto en la experiencia general de uso.

Los resultados muestran que el 50 % de los participantes calificó el proceso como “muy cómodo”, el 45 % como “cómodo”, y únicamente el 5 % indicó sentirse “poco cómodo”, sin reportarse respuestas en la opción “nada cómodo”. Estos datos reflejan una alta satisfacción con el sistema de inicio de sesión, evidenciando que la interfaz es clara y el flujo de autenticación resulta familiar y eficiente para la mayoría de los usuarios.

El reducido porcentaje de respuestas “poco cómodo” podría atribuirse a factores externos, como la preferencia por métodos alternativos de ingreso (por ejemplo, autenticación mediante GitHub), más que a fallas de diseño en la plataforma. En general, los resultados validan la adecuación del esquema actual de acceso, garantizando usabilidad y confort en la interacción inicial.

20 respuestas

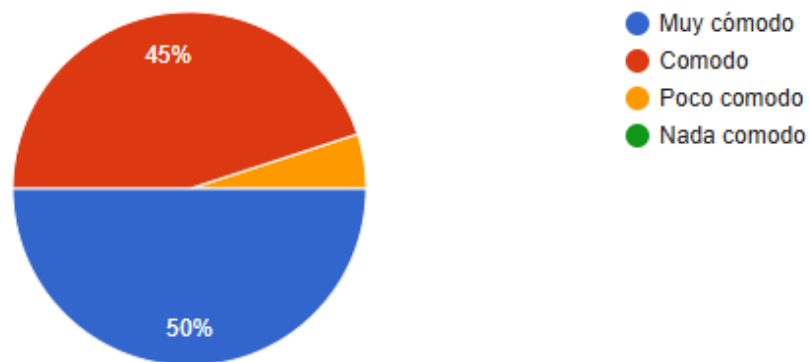


Figura 12: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

Fuente: Elaboración propia.

Pregunta: ¿Qué tan sencilla fue la navegación dentro de la aplicación?

Esta pregunta tuvo como objetivo medir la facilidad de navegación dentro de la aplicación, un aspecto fundamental para garantizar una experiencia de usuario intuitiva y coherente con los principios de diseño centrado en el usuario. La claridad en la organización de menús, botones y secciones contribuye directamente a la eficiencia en el uso de la aplicación.

De acuerdo con los resultados, el 60 % de los encuestados calificó la navegación como “sencilla”, el 25 % como “muy sencilla”, mientras que un 10 % la consideró “poco sencilla” y un 5 % optó por una valoración “neutral”. Estos datos reflejan una valoración predominantemente positiva, indicando que la mayoría de los usuarios logró desplazarse por las diferentes secciones sin dificultad significativa.

El pequeño porcentaje de respuestas negativas podría estar relacionado con la familiarización inicial con la estructura de la plataforma o con diferencias en dispositivos y resoluciones de pantalla. En conjunto, los resultados demuestran que la aplicación presenta una arquitectura de navegación clara y accesible, facilitando la interacción fluida entre las distintas funcionalidades.

20 respuestas

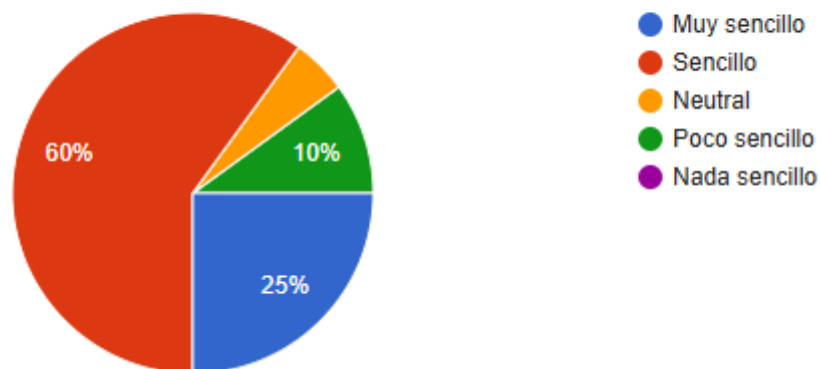


Figura 13: Encuesta sobre la usabilidad y funcionalidad de la aplicación

Fuente: Elaboración propia.

Pregunta: ¿El diseño visual de la aplicación te pareció atractivo y cómodo de usar?

El propósito de esta pregunta fue evaluar la percepción estética y la comodidad visual que ofrece la interfaz de la aplicación. Este aspecto resulta relevante, ya que el diseño visual influye directamente en la satisfacción del usuario, la comprensión de la información y la motivación para interactuar con la plataforma.

Los resultados indican que el 35 % de los encuestados respondió “sí, en parte”, otro 35 % se mantuvo “neutral”, el 25 % calificó “sí, mucho”, y el 5 % manifestó “poco”. Esta distribución revela una tendencia moderadamente positiva, con una mayoría que percibe el diseño como aceptable o atractivo, aunque con espacio para mejoras en cuanto a uniformidad visual o uso de colores y contrastes.

El porcentaje de respuestas neutrales sugiere que, si bien la interfaz cumple con los estándares funcionales, podría beneficiarse de ajustes estéticos y de ergonomía visual que refuercen la sensación de modernidad y coherencia gráfica. En general, los resultados respaldan la eficacia del diseño actual, pero también evidencian la oportunidad de fortalecer la experiencia visual del usuario mediante mejoras en el estilo y la presentación.

20 respuestas

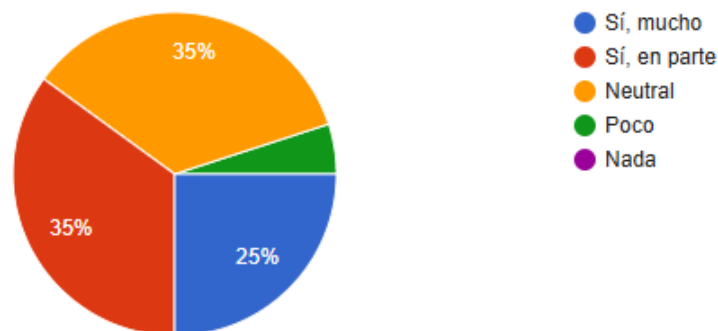


Figura 14: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

Fuente: Elaboración propia.

Pregunta: ¿El nivel de dificultad de las preguntas fue adecuado a tu preparación?

Esta pregunta tuvo como finalidad determinar la pertinencia y equilibrio del nivel de dificultad de las preguntas incluidas en las simulaciones de entrevista, en relación con la formación académica y el dominio técnico de los usuarios. Evaluar este aspecto es esencial para garantizar que la aplicación cumpla con su propósito pedagógico y formativo sin generar frustración o desinterés.

De acuerdo con los resultados, el 50 % de los participantes consideró que el nivel fue “adecuado en parte”, el 35 % lo percibió como “totalmente adecuado”, y el 15 % manifestó una posición “neutral”, sin respuestas en las categorías negativas. Esta tendencia demuestra una valoración globalmente positiva respecto al equilibrio de dificultad, lo que sugiere que el sistema logra ofrecer un reto apropiado para estudiantes de ingeniería en sistemas o desarrollo de software.

No obstante, el hecho de que la mitad de los encuestados haya seleccionado la opción “en parte” indica la conveniencia de ajustar gradualmente la dificultad de las preguntas o personalizarla según el nivel del usuario. En conjunto, los resultados evidencian que la aplicación cumple con el objetivo de evaluar habilidades técnicas reales, manteniendo un grado de exigencia adecuado al perfil académico del público objetivo.

20 respuestas

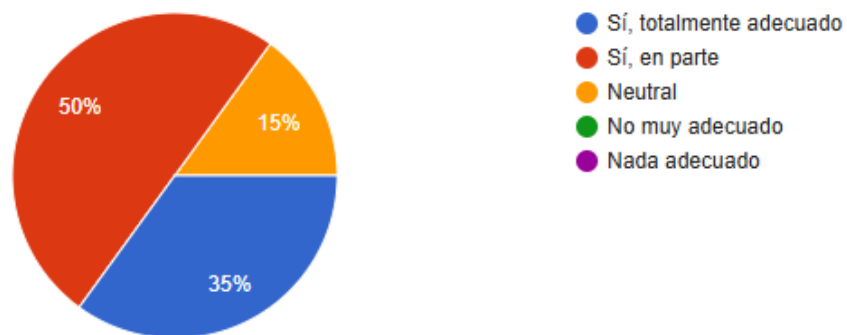


Figura 15: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

Pregunta: ¿La retroalimentación automática sobre tus respuestas fue clara y fácil de entender?

El objetivo de esta pregunta fue evaluar la efectividad y comprensión del sistema de retroalimentación automática implementado en la aplicación, el cual informa al usuario sobre la corrección o error de sus respuestas y, en algunos casos, ofrece explicaciones complementarias. La claridad en este proceso es esencial para promover el aprendizaje autónomo y garantizar que el usuario identifique con precisión las áreas en las que debe mejorar.

Según los resultados obtenidos, el 70 % de los encuestados calificó la retroalimentación como “clara”, el 25 % como “muy clara”, mientras que un 5 % la consideró “nada clara”. Esta distribución refleja una valoración predominantemente positiva, con un 95 % de los participantes percibiendo la retroalimentación como comprensible y útil.

20 respuestas

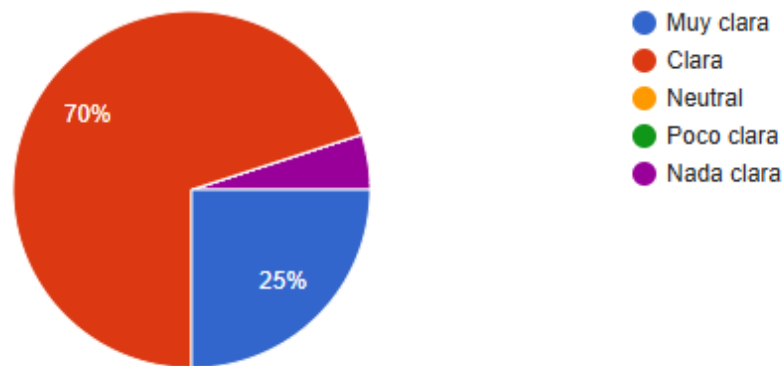


Figura 16: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

Pregunta: ¿Qué tan útiles fueron las recomendaciones personalizadas que recibiste?

Esta pregunta tuvo como finalidad medir la percepción de utilidad de las recomendaciones personalizadas que la aplicación ofrece tras la realización de las entrevistas simuladas. Dichas recomendaciones buscan orientar al usuario en los temas o competencias que debe reforzar, convirtiéndose en un componente esencial para el aprendizaje adaptativo y la mejora continua del desempeño técnico.

De acuerdo con los resultados, el 55 % de los encuestados consideró las recomendaciones “útiles”, el 30 % las calificó como “muy útiles” y un 15 % mantuvo una posición “neutral”, sin registrarse respuestas negativas. Esta distribución evidencia una alta aceptación del sistema de recomendaciones, lo que sugiere que la información proporcionada fue pertinente, clara y percibida como un apoyo real para el progreso académico y profesional del usuario.

El pequeño porcentaje de neutralidad podría reflejar la necesidad de mayor personalización en los mensajes o sugerencias más detalladas según el rendimiento individual. En conjunto, los resultados confirman que el módulo de recomendaciones cumple su propósito de ofrecer orientaciones prácticas y relevantes, contribuyendo de manera directa al proceso de autoevaluación y aprendizaje continuo que promueve la plataforma.

20 respuestas

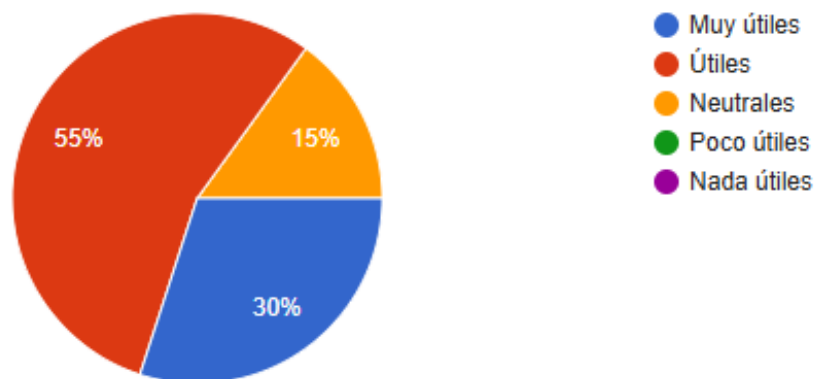


Figura 17: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

Pregunta: ¿Qué tan clara fue la interfaz del editor de código?

El propósito de esta pregunta fue evaluar la claridad y facilidad de uso del editor de código incluido en la aplicación, componente esencial del entorno de práctica, ya que permite al usuario resolver ejercicios de programación directamente dentro de la plataforma. Una interfaz clara en este módulo es determinante para evitar distracciones, reducir errores y favorecer la concentración durante la simulación.

Los resultados muestran que el 45 % de los participantes calificó la interfaz como “clara”, el 30 % como “muy clara” y el 25 % adoptó una posición “neutral”. Esta distribución indica una percepción mayoritariamente positiva, con un 75 % de los usuarios reconociendo que el diseño del editor resulta comprensible y funcional.

El porcentaje de neutralidad sugiere que algunos usuarios podrían requerir mayor familiarización con el entorno de codificación o preferir un diseño más minimalista y con mayor contraste visual. No obstante, en términos generales, los resultados demuestran que el editor cumple adecuadamente su función, proporcionando una interfaz intuitiva y adecuada para la práctica técnica dentro de la simulación.

20 respuestas

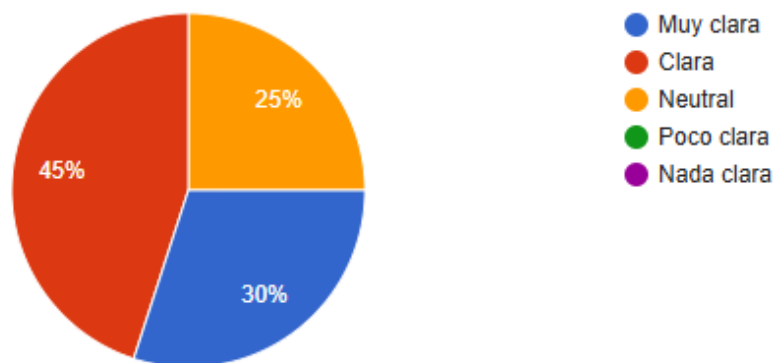


Figura 18: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

Pregunta: ¿Qué tan satisfecho estás con la experiencia general de la aplicación?

Esta pregunta tuvo como propósito medir el grado de satisfacción global de los usuarios frente a su experiencia con la aplicación, integrando aspectos como la usabilidad, la funcionalidad, el diseño visual y la efectividad del aprendizaje percibido. La satisfacción general constituye un indicador clave para determinar la aceptación del sistema y su potencial de adopción por parte del público objetivo.

Los resultados muestran que el 45 % de los encuestados se declaró “satisfecho”, el 20 % “muy satisfecho”, el 30 % se mantuvo en una posición “neutral”, y solo un 5 % indicó estar “poco satisfecho”, sin respuestas en la categoría “nada satisfecho”. Esta distribución evidencia una valoración predominantemente favorable, con un 65 % de opiniones positivas frente al desempeño general de la aplicación.

El nivel de neutralidad registrado podría estar asociado a la percepción de que el proyecto aún se encuentra en fase de desarrollo y podría beneficiarse de ajustes menores en el diseño o la personalización de contenidos. En conjunto, los resultados reflejan que la aplicación logra ofrecer una experiencia satisfactoria, funcional y pertinente, cumpliendo con los objetivos de usabilidad y apoyo al aprendizaje propuestos en su diseño inicial.

20 respuestas

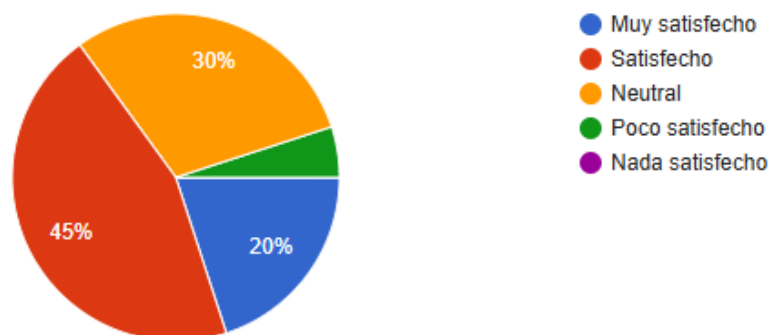


Figura 19: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

Pregunta 10. ¿Sientes que practicar con esta plataforma aumentó tu confianza para entrevistas reales?

Esta pregunta tuvo como finalidad evaluar el impacto formativo y motivacional de Interview-Sim en los usuarios, particularmente en relación con la confianza adquirida para enfrentar entrevistas laborales reales. Este indicador resulta esencial para valorar la efectividad pedagógica de la aplicación, ya que su objetivo principal es preparar a los estudiantes en entornos simulados que reproduzcan situaciones reales del ámbito profesional.

De acuerdo con los resultados, el 45 % de los encuestados respondió “sí, un poco”, el 25 % manifestó “sí, mucho”, el 25 % se mantuvo “neutral” y un 5 % indicó “no mucho”, sin respuestas en la categoría “no”. Esta distribución refleja una tendencia positiva, donde el 70 % de los participantes percibió un aumento, total o parcial, en su nivel de confianza luego de utilizar la plataforma.

Los resultados sugieren que la experiencia de simulación ofrecida por InterviewSim cumple su función principal al reducir la ansiedad y familiarizar al usuario con las dinámicas de una entrevista técnica real. No obstante, el grupo neutral y las pocas respuestas negativas indican la oportunidad de incorporar más escenarios prácticos y retroalimentaciones personalizadas que potencien la seguridad y el desarrollo de habilidades comunicativas. En conjunto, los datos confirman que la aplicación tiene un impacto positivo en la preparación profesional y la autoconfianza.

20 respuestas

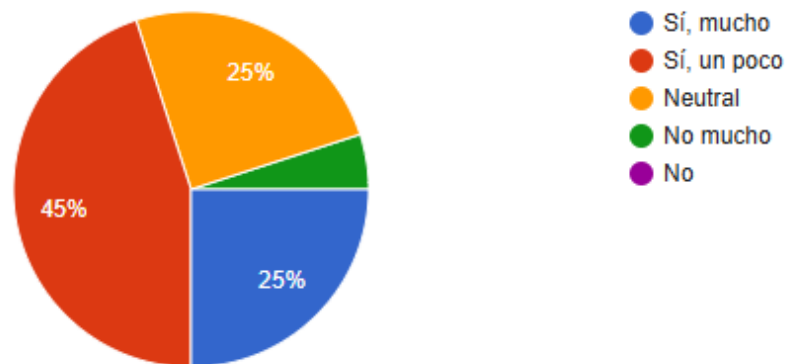


Figura 20: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

Pregunta: ¿Recomendarías la aplicación a otras personas que se preparan para entrevistas?

El objetivo de esta pregunta fue medir la predisposición de los usuarios a recomendar InterviewSim a otras personas interesadas en prepararse para entrevistas laborales en el ámbito de la programación o el desarrollo de software. Este indicador, conocido comúnmente como intención de recomendación, constituye una medida global de satisfacción y confianza en la utilidad del producto, así como del impacto percibido en la comunidad usuaria.

Los resultados muestran que el 50 % de los encuestados respondió “sí, probablemente”, el 40 % “sí, definitivamente” y el 10 % seleccionó “tal vez”, sin registrarse respuestas negativas. Esta tendencia refleja una aceptación general muy alta, donde el 90 % de los participantes manifestó una intención positiva de recomendar la plataforma, lo cual evidencia su grado de satisfacción con la experiencia ofrecida.

La ausencia de respuestas desfavorables confirma que InterviewSim ha logrado consolidar una propuesta funcional y valiosa para su público objetivo, percibida como una herramienta eficaz para la preparación técnica y emocional de entrevistas. En síntesis, los resultados sugieren un alto potencial de adopción y difusión entre estudiantes y futuros profesionales del área de sistemas, consolidando la credibilidad y pertinencia del proyecto en contextos académicos y laborales.

20 respuestas

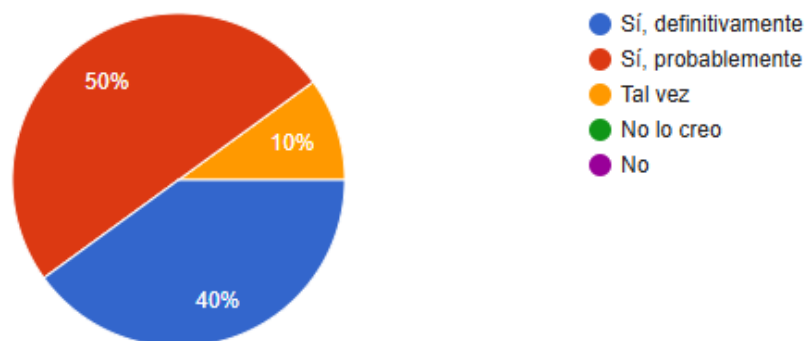


Figura 21: Encuesta sobre la usabilidad y funcinabilidad de la aplicacion

6.5.1. Resultados obtenidos del proyecto

Con el propósito de medir la eficacia y utilidad del sistema de entrevistas con inteligencia artificial desarrollado, se llevó a cabo una fase de pruebas con usuarios reales. Esta etapa fue fundamental para evaluar no solo el rendimiento técnico de la aplicación, sino también la experiencia de uso, el nivel de satisfacción de los participantes y la precisión de las respuestas generadas por el sistema. La prueba se realizó con un grupo de 20 personas seleccionadas de manera intencional para representar distintos perfiles. Algunos tenían experiencia previa en procesos de entrevista, mientras que otros contaban con conocimientos del área. Esta diversidad permitió analizar cómo respondía el sistema ante distintos niveles de preparación y necesidades, aspecto clave para validar su funcionalidad en contextos reales variados. Durante el proceso, cada participante interactuó libremente con la plataforma. Se les solicitó crear su perfil de usuario, seleccionar el tipo de entrevista que deseaban simular, responder las preguntas generadas por la inteligencia artificial y recibir la retroalimentación automática del sistema. No se establecieron restricciones en la interacción, con el fin de simular una experiencia natural y fluida, similar a la de una entrevista real en condiciones normales. Al finalizar la simulación, los usuarios completaron una encuesta estructurada para recopilar sus percepciones. Esta incluyó preguntas cerradas y abiertas relacionadas con la facilidad de uso de la aplicación, la claridad de las instrucciones, la utilidad percibida de las entrevistas simuladas, la coherencia de las preguntas y la pertinencia de la retroalimentación recibida. Un aspecto que destacó especialmente fue el beneficio que brindó el sistema a usuarios con poca experiencia en procesos de entrevista. Gracias a las simulaciones personalizadas, estos pudieron enfocar mejor su preparación, identificar sus fortalezas y debilidades, y comprender con mayor claridad los aspectos clave evaluados durante una entrevista laboral. En este sentido, la aplicación fue vista como una herramienta útil para ahorrar tiempo, reducir la incertidumbre y facilitar la mejora del desempeño en futuros procesos de selección.

6.5.2. Análisis e importancia de la encuesta

El análisis global de la encuesta aplicada al grupo de usuarios permitió obtener una visión integral del desempeño y la percepción general sobre la plataforma InterviewSim. Los resultados evidencian una alta aceptación en términos de usabilidad, funcionalidad y pertinencia, destacándose que la mayoría de los participantes calificó positivamente aspectos como la facilidad de registro, la claridad en la navegación y la efectividad de la retroalimentación automática. Estos indicadores demuestran que la aplicación ofrece una experiencia de uso intuitiva, accesible y alineada con los principios de diseño centrado en el usuario.

Asimismo, los datos muestran una valoración favorable del editor de código y de las recomendaciones personalizadas, lo que confirma que las funciones principales cumplen adecuadamente con su propósito pedagógico. El hecho de que más del 70 % de los usuarios manifestara sentirse más confiado para enfrentar entrevistas reales y que el 90 % indicara que recomendaría la aplicación a otros, refuerza la validez del proyecto como herramienta de preparación profesional dentro del ámbito académico.

La encuesta resultó fundamental para validar empíricamente la efectividad y aceptación del sistema, aportando información cuantitativa y cualitativa que orienta futuras mejoras. Entre las oportunidades identificadas se encuentran la posibilidad de ampliar los niveles de dificultad,

mejorar la personalización de la retroalimentación y optimizar el diseño visual. En conjunto, los hallazgos confirman que InterviewSim cumple su objetivo de fortalecer las competencias técnicas y la confianza de los estudiantes, consolidándose como una plataforma de apoyo innovadora en el proceso de preparación para entrevistas.

7. Conclusiones y Trabajos Futuros

7.1. Conclusiones

El desarrollo de InterviewSim mostró que es posible tener una plataforma que use inteligencia artificial para simular entrevistas técnicas de manera interactiva. Al generar preguntas con inteligencia artificial se creó un sistema para manejar usuarios con diferentes roles y una base de datos que se adapta, se creó una herramienta útil que ayuda a mejorar las habilidades profesionales de los usuarios.

La arquitectura cliente-servidor que se utilizó permitió distinguir claramente entre la lógica del negocio, el manejo de datos y la parte que ve el usuario, asegurando que sea fácil de escalar, mantener y modular en su desarrollo. La mezcla de tecnologías como React, Node.js y MongoDB ofreció una estructura moderna y eficaz para el manejo de la información y proporcionó una experiencia fluida para el usuario.

Durante el proceso de creación, surgieron varios problemas relacionados con el proceso de autenticar a los usuarios según su función (estudiante o entrevistador), el manejo de cookies, la conexión entre la parte visible de la aplicación y el servidor, la interacción con la base de datos y fallos durante la puesta en marcha de la aplicación. No obstante, gracias a una depuración constante, la revisión de las herramientas utilizadas y la implementación de buenas prácticas en el desarrollo, se logró resolver cada uno de estos problemas, alcanzando un sistema que es estable y funcional.

Asimismo, la introducción del componente de inteligencia artificial basado en el modelo GPT-3.5-Turbo de OpenAI facilitó la generación automática de preguntas de opción múltiple, ajustadas al nivel y al idioma elegidos por el usuario. Esto evidencia las capacidades de los modelos lingüísticos en la enseñanza y valoración de habilidades técnicas. La transmisión y recepción de los prompts desde la parte posterior se llevó a cabo a través de solicitudes HTTP específicas, conservando constantes estables en la temperatura y el número máximo de tokens, lo que asegura un balance óptimo entre originalidad y exactitud en las respuestas.

Se pudo observar una integración robusta entre los componentes de la arquitectura del sistema, que fue creada con el método MERN (MongoDB, Express, React y Node.js). El frontend, creado con Vite y React, facilitó una experiencia sin inconvenientes y responsive; el backend, por su parte, fue elaborado en Node.js y gestionó la API de OpenAI, la comunicación con ella, la autenticación y el almacenamiento de datos. La posibilidad de escalar y mantener el sistema se vio favorecida por esta división nítida entre servidor y cliente.

En el periodo de desarrollo, surgieron retos técnicos como corregir errores de conexión con la base de datos, manejar correctamente las cookies seguras y hacer modificaciones en los prompts generados por la inteligencia artificial. No obstante, gracias a la adopción de prácticas adecuadas de programación, a la depuración constante del código y al empleo de variables de entorno protegidas .env, se logró un producto estable, eficaz y con un flujo de datos fiable.

En general, InterviewSim es un progreso importante que une entornos educativos digitales con herramientas de inteligencia artificial, proporcionando a los usuarios una plataforma interactiva, escalable y útil para mejorar su confianza en entrevistas reales y sus capacidades técnicas.

En resumen, InterviewSim es una idea nueva que mezcla educación y tecnología, brindando un espacio accesible y activo para prepararse para entrevistas técnicas. Su diseño modular y la inclusión de herramientas de inteligencia artificial allanan el camino para futuras mejoras y expansiones, estableciéndola como una ayuda tanto académica como profesional.

7.2. Trabajos Futuros

Conforme a los resultados obtenidos y la experiencia adquirida a lo largo del desarrollo del sistema, se reconocen varias áreas de trabajo que podrían expandir las habilidades de InterviewSim en versiones posteriores. Las mejoras mencionadas se dirigen a la optimización técnica y al desarrollo de la funcionalidad de la plataforma:

- **Ampliación del banco de preguntas:** Incorporar una variedad más amplia de temas, grados de dificultad y campos del saber. Además de las preguntas de programación, incluir evaluaciones acerca de algoritmos, estructuras de datos, bases de datos, metodologías ágiles y la solución a problemas lógicos. Esto posibilitaría la creación de simulaciones más completas y adecuadas a varios perfiles profesionales.
- **Optimización de la retroalimentación de la IA:** Perfeccionar la estructura de los *prompts* que se envían al modelo GPT-3.5-Turbo de OpenAI, utilizando métodos de ingeniería del prompt para optimizar la pertinencia, el formato y la coherencia de las preguntas producidas. Se sugiere además la implementación de un sistema de validación automática que analice la calidad de las respuestas proporcionadas por la IA antes de guardarlas en la base de datos.
- **Incorporación de entrevistas por voz:** incorporar un módulo para la síntesis y el reconocimiento de voz con el fin de posibilitar entrevistas orales simuladas. Así, los usuarios tendrían la oportunidad de practicar no solo el contenido técnico, sino también la comunicación efectiva y la expresión verbal en una entrevista real.
- **Sistema de métricas y progreso:** Incluir un panel interactivo que presente estadísticas minuciosas sobre el desempeño del usuario, como la exactitud, la rapidez de respuesta, el progreso por temas y la comparación con resultados pasados. Para ello, librerías de visualización como Chart.js o Recharts podrían ser empleadas.
- **Retroalimentación adaptativa:** Crear un sistema de evaluación automática que examine las respuestas del usuario y ofrezca sugerencias personalizadas. Esto posibilitaría brindar una retroalimentación formativa inmediata, señalando puntos fuertes, débiles y áreas de mejora concretas, lo que favorecería un aprendizaje individualizado.
- **Despliegue multiplataforma:** Crear una versión para móviles de la aplicación usando React Native, conservando la misma estructura de autenticación JWT y de servicios REST. Esta versión incrementaría la capacidad de acceso del sistema y posibilitaría que se llevaran a cabo entrevistas simuladas desde dispositivos iOS o Android.
- **Integración con entornos educativos:** Integrar InterviewSim con plataformas para la gestión del aprendizaje (Learning Management Systems - LMS) como Google Classroom o Moodle, posibilitaría que educadores y tutores puedan supervisar el avance de los alumnos, asignar entrevistas y hacer evaluaciones en el contexto académico.

El añadir estas mejoras posibilitará que InterviewSim se desarrolle como una herramienta más completa, interactiva y ajustada a las exigencias reales de los usuarios, reafirmando así su función como un recurso educativo en la capacitación para entrevistas con asistencia de inteligencia artificial.

Referencias

- [1] A. Sharma, «Rethinking Problem-Solving in Interviews,» *IEEE Software*, vol. 37, n.º 1, págs. 88-94, 2019.
- [2] M. Behroozi, S. Shirolkar, T. Barik y C. Parnin, «Does Stress Impact Technical Interview Performance?» En *Proceedings of the 2020 ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*, New York, NY, USA: Association for Computing Machinery, 2020, ISBN: 978-1-4503-7043-1. DOI: 10.1145/3368089.3409712. dirección: <https://doi.org/10.1145/3368089.3409712>.
- [3] M. Behroozi, «Towards Scientific Study of Technical Interviews Using Eye Tracking,» en *2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 2019, págs. 229-230. DOI: 10.1109/VLHCC.2019.8818900.
- [4] P. Guenes, R. Tomaz, M. Kalinowski, M. T. Baldassarre y M.-A. Storey, «Imposter Phenomenon in Software Engineers,» en *Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS'24)*, Lisbon, Portugal, 2024, págs. 96-106. DOI: 10.1145/3639475.3640114.
- [5] P. R. Clance y S. A. Imes, «The Impostor Phenomenon in High Achieving Women: Dynamics and Therapeutic Intervention,» *Psychotherapy: Theory, Research & Practice*, vol. 15, n.º 3, págs. 241-247, 1978.
- [6] A. Parkman, «The Impostor Phenomenon in Higher Education: Incidence and Impact,» *Journal of Higher Education Theory and Practice*, vol. 16, n.º 1, págs. 51-60, 2016.
- [7] G. Matthews et al., «Emotional Intelligence, Personality, and Task-Induced Stress in a High-Risk Occupation,» *Military Psychology*, vol. 18, n.º suppl. 1, S27-S39, 2006.
- [8] P. Silva, «The Impact of Impostor Syndrome on Software Engineers' Productivity and Wellbeing,» en *Proceedings of the 43rd International Conference on Software Engineering*, 2021.
- [9] M. Behroozi, C. Parnin y T. Barik, «Hiring is Broken: What Do Developers Say About Technical Interviews?» En *2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, 2019, págs. 1-9. DOI: 10.1109/VLHCC.2019.8818836.
- [10] HackerRank. «Simulating Interview Environments. »dirección: <https://www.hackerrank.com/blog/>.
- [11] FreeCodeCamp. «Performance Anxiety in Coding Interviews. »dirección: <https://www.freecodecamp.org/>.
- [12] T. e. I. Ministerio de Ciencia, *2a convocatoria ecosistema científico para la financiación de programas de I+D+i que contribuyan al mejoramiento de la calidad de las instituciones de educación superior colombianas*, oct. de 2016.
- [13] J. A. C. L. y H. F. A. J., «Modelo de investigación en ingeniería del software: Una propuesta de investigación tecnológica,» inf. téc.
- [14] H. Arboleda y J. Chavarriaga, «"LIDIS": Consideraciones para la generación, maduración y aplicación de conocimiento en ingeniería de software,» inf. téc.
- [15] A. Martel. «Un poco de historia sobre Extreme Programming. »dirección: <https://www.antoniomartel.com/2018/12/un-poco-de-historia-sobre-extreme-programming.html>.

- [16] Atlassian. «Jira Software. »dirección: <https://www.atlassian.com/software/jira>.
- [17] S. T. Patel y K. P. Vora, «A Review on Agile Project Management Tools: A Survey,» en *2020 International Conference on Communication Systems, Computing and IT Applications (CSCI)*, 2020, págs. 14-19.
- [18] *Systems and Software Engineering — Systems and Software Quality Requirements and Evaluation (SQuaRE) — System and Software Quality Models*, International Organization for Standardization, 2011.
- [19] *Information technology — Security techniques — Information security management systems — Requirements*, International Organization for Standardization, 2013.
- [20] *Ley 1581 de 2012: Ley de Protección de Datos Personales*, Congreso de la República de Colombia, 2012.
- [21] S. Lin, *Educational Psychology*, W. Na, ed. Taipei: Wu Na, 1999, págs. 113-140.
- [22] X. Zhang, *New Trends in Learning and Teaching*. Taipei: Psychology Press, 2003.
- [23] J. Winterton, F. D.-L. Deist y E. Stringfellow, «Typology of Knowledge, Skills, and Competencies: Clarification of the Concept and Prototype,» Office for Official Publications of the European Communities, inf. téc., 2006.
- [24] B. J. Zimmerman, «Self-efficacy: An essential motive to learn,» *Contemporary Educational Psychology*, vol. 25, n.º 1, págs. 82-91, 2000.
- [25] L. D. Goodstein, «Behavioral theoretical views of counseling,» en *Theories of Counseling*, 1965, págs. 140-192.
- [26] D. P. Saxon y E. A. Morante, «Effective student assessment and placement: Challenges and recommendations,» *Journal of Developmental Education*, vol. 37, n.º 3, 2014.
- [27] H. P. Y. et al., «Enfoque basado en IA para la extracción eficiente de información para contratos de la cadena de suministro,» en *Conferencia IEEE sobre Inteligencia Artificial (CAI)*, 2024, págs. 139-143.
- [28] Y. Z. et al., «Predicción de funciones de proteínas basada en la técnica de atención de gráficos heterogéneos,» *IEEE Journal of Biomedical and Health Informatics*, vol. 28, n.º 4, págs. 2408-2415, 2024.
- [29] S. Kumar, Simran y N. T. Singh, «Mejora de la comprensión contextual en GPT a través del preentrenamiento multimodal,» en *Conferencia Internacional de 2023 sobre la Confluencia de Avances en Robótica, Visión y Gestión de Tecnología Interdisciplinaria (IC-RVITM)*, 2023, págs. 1-7.
- [30] P. Isaranontakul y W. Kreesuradej, «Un estudio sobre el uso de GPT-3 para generar un análisis de sentimiento tailandés del conjunto de datos de tweets de COVID-19,» en *20ª Conferencia Conjunta Internacional sobre Ciencias de la Computación e Ingeniería de Software (JCSSE)*, 2023, págs. 106-111.
- [31] S. A. et al., «El futuro del reclutamiento: uso del aprendizaje profundo para construir robots de entrevistas inteligentes,» en *15ª Conferencia Internacional sobre Tecnologías Informáticas, de Comunicación y de Redes (ICCCNT)*, 2024, págs. 1-6.
- [32] N. Parmar, «Análisis del rendimiento de WebRTC y SIP para videoconferencia,» *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, n.º 9S, 2019.
- [33] C. Jennings, *WebRTC: Comunicación en tiempo real en navegadores*, Disponible en: <https://www.w3.org/TR/webrtc/>.

- [34] S. Rajbhar, S. Shelke, A. Singh, Y. Singh y P. Shinde, «AIced-Prep -AI Based Mock Interview Evaluator,» en *2025 6th International Conference on Data Intelligence and Cognitive Informatics (ICDICI)*, 2025, págs. 1840-1843. DOI: 10.1109/ICDICI66477.2025.11135157.
- [35] V. Yepez-Reyes y J. Cruz-Silva, «Inteligencia artificial en la transcripción de entrevistas,» *Contratexto*, n.º 41, págs. 183-202, 2024, Epub 31-Mayo-2024. Consultado el 15 de octubre de 2025, ISSN: 1025-9945. DOI: 10.26439/contratexto2024.n41.6750. dirección: http://www.scielo.org.pe/scielo.php?script=sci_arttext&pid=S1025-99452024000100183&lng=es&nrm=iso.
- [36] G. S. Rao, M. Jaiganesh, M. A. K, M. A. A. A, N. R y P. K. Parida, «AI Powered Virtual Job Interview Simulator Using Natural Language Processing,» en *2025 8th International Conference on Trends in Electronics and Informatics (ICOEI)*, 2025, págs. 1413-1421. DOI: 10.1109/ICOEI65986.2025.11013504.
- [37] R. M. Marvaniya, A. S. Acharya, D. M. Detroja, V. K. Dabhi y H. B. Prajapati, «Smart Prep: AI Based Interactive Interview Preparation System,» en *2025 4th OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 5.0*, 2025, págs. 1-6. DOI: 10.1109/OTCON65728.2025.11070972.
- [38] «Seguimiento del estado del diálogo y selección de acciones utilizando un mecanismo de aprendizaje profundo para el entrenamiento de entrevistas,» en *Conferencia Internacional sobre Procesamiento del Lenguaje Asiático (IALP) de 2016*.
- [39] J. I. Millen, «Using Generative AI for Interview Simulations to Enhance Student Research Skills in Biology Education,» *Journal of Microbiology and Biology Education*, vol. 26, n.º 2, e00122-25, 2025. DOI: 10.1128/jmbe.00122-25. dirección: <https://doi.org/10.1128/jmbe.00122-25>.
- [40] T. Daryanto, X. Ding, L. T. Wilhelm, S. Stil, K. M. Knutsen y E. H. Rho, «Conversate: Supporting Reflective Learning in Interview Practice Through Interactive Simulation and Dialogic Feedback,» *arXiv*, 2024. DOI: 10.48550/arXiv.2410.05570. eprint: 2410.05570. dirección: <https://arXiv.org/abs/2410.05570>.
- [41] A. B. Nofal et al., «AI-enhanced interview simulation in the metaverse: Transforming professional skills training through VR and generative conversational AI,» *Computers and Education: Artificial Intelligence*, vol. 8, pág. 100347, 2024. DOI: 10.1016/j.caeai.2024.100347. dirección: <https://www.sciencedirect.com/science/article/pii/S2666920X24001504>.
- [42] K. K. Sahani, M. S. Khan, S. Khatwani, S. Gupta y A. Dubey, «A Smart Interview Simulator Using AI Avatars and Real-Time Feedback Mechanisms (AI Avatar for Interview Preparation),» *International Journal of Engineering Technologies and Management Research*, vol. 12, n.º 5, págs. 66-75, 2025. DOI: 10.29121/ijetmr.v12.i5.2025.1618. dirección: <https://doi.org/10.29121/ijetmr.v12.i5.2025.1618>.
- [43] H. Koshti, P. Gosavi, R. Pagar, P. Khairnar y S. Talekar, «AI-Powered Interview Preparation System: Integrating Resume Analysis, HR Simulation, and Technical Skill Assessment,» *Journal of Engineering Research and Reports*, vol. 27, n.º 5, págs. 21-33, 2025. DOI: 10.9734/jerr/2025/v27i51489. dirección: <https://journaljerr.com/index.php/JERR/article/view/1489>.

- [44] I. Meta Platforms, *React: A JavaScript library for building user interfaces*, ver. 18.3.1, Accessed: 2025-10-21, 2025. dirección: <https://react.dev/>.
- [45] R. T. Fielding, «Architectural Styles and the Design of Network-based Software Architectures,» Accessed: 2025-10-21, Tesis doct., University of California, Irvine, 2000. dirección: https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf.
- [46] OpenJS Foundation, *Node.js: JavaScript runtime built on Chrome's V8 engine*, ver. 22.9.0, Accessed: 2025-10-21, 2025. dirección: <https://nodejs.org/>.
- [47] OpenJS Foundation, *Express.js: Fast, unopinionated, minimalist web framework for Node.js*, ver. 4.19.2, Accessed: 2025-10-21, 2025. dirección: <https://expressjs.com/>.
- [48] MongoDB Inc., *MongoDB: The Developer Data Platform*, ver. 7.0, Accessed: 2025-10-21, 2025. dirección: <https://www.mongodb.com/>.
- [49] OpenAI, *OpenAI API Documentation*, Accedido el 21 de octubre de 2025, 2024. dirección: <https://platform.openai.com/docs>.
- [50] Ecma International, *ECMAScript Language Specification*, Accessed: 2025-10-21, 2023. dirección: <https://262.ecma-international.org/14.0/>.
- [51] Tailwind Labs Inc., *Tailwind CSS: A utility-first CSS framework*, ver. 3.4.14, Accessed: 2025-10-21, 2025. dirección: <https://tailwindcss.com/>.
- [52] MongoDB Inc., *MongoDB Compass: The GUI for MongoDB*, Herramienta gráfica utilizada para la gestión y visualización de la base de datos. Accedido el 21 de octubre de 2025, 2024. dirección: <https://www.mongodb.com/products/compass>.

8. Anexos

Debido a la extensión de los anexos, estos se encuentran de manera digital en el repositorio del del proyecto tanto el backend como el fronted.

<https://github.com/NilsonSytaxError39/InterviewSim-Frontend.git>

<https://github.com/NilsonSytaxError39/InterviewSim-Backend.git>

A continuación, se presenta el enlace correspondiente al despliegue de la aplicación. Para realizar pruebas o acceder a la versión funcional del sistema, es necesario comunicarse previamente al correo nilson.cuero@correounivalle.edu.co , ya que dicha cuenta pertenece al responsable del despliegue del proyecto y se requiere su autorización para reactivar el entorno de ejecución. <https://interviewsim-frontend.netlify.app/>

Para la tablas de trabajo que realizamos con metodoliga kanban revisar el siguiente enlace:

Tablero Kanban en Jira