

Universidad Siglo 21



Trabajo Final de Grado. Prototipado Tecnológico

Carrera: Licenciatura en Informática

Generación de sitios web con chatbot e inteligencia artificial

Autor: Ivan Esteban Nesteruk

Legajo: VINF06264

Formosa, noviembre de 2024

Índice

Resumen	5
Abstract	6
Título	7
Introducción	7
<i>Antecedentes</i>	7
<i>Descripción Del Área Problemática</i>	9
Justificación	10
Objetivo General del Proyecto	10
Objetivos Específicos del Proyecto	10
Marco Teórico Referencial	11
<i>Dominio Del Problema</i>	11
<i>TICs</i>	11
<i>Competencia</i>	12
Diseño Metodológico	13
<i>Herramientas Metodológicas</i>	13
<i>Herramientas De Desarrollo</i>	13
<i>Recolección De Datos</i>	13
<i>Planificación Del Proyecto</i>	14
Relevamiento	16
<i>Relevamiento Estructural</i>	16
<i>Relevamiento Funcional</i>	16
Procesos de Negocios	18

Diagnóstico y Propuesta	19
<i>Propuesta</i>	21
Objetivos, Límites y Alcances del Prototipo	22
<i>Objetivo Del Prototipo</i>	22
<i>Límites</i>	22
<i>Alcances</i>	22
Descripción del Sistema	22
<i>Product Backlog</i>	22
<i>Historias De Usuario</i>	23
<i>Sprint Backlog</i>	26
<i>Estructura De Datos</i>	28
<i>Prototipos De Interfaces De Pantallas</i>	30
<i>Diagrama De Arquitectura</i>	36
Seguridad	37
Análisis de Costos	38
Análisis de Riesgos	40
Conclusiones	46
Demo	46
Referencias	47

Índice de tablas

Tabla 1: Competencia.....	12
Tabla 2: Desarrollo de Prototipo Rápido.....	20
Tabla 3: Creacion de Componentes Web.....	20
Tabla 4: Usuario Final en la Creación de Componentes Web.....	20
Tabla 5: Product Backlog.....	22
Tabla 6: Historia de Usuario 1.....	24
Tabla 7: Historia de Usuario 2.....	24
Tabla 8: Historia de Usuario 3.....	24
Tabla 9: Historia de Usuario 4.....	25
Tabla 10: Historia de Usuario 5.....	25
Tabla 11: Historia de Usuario 6.....	26
Tabla 12: Historia de Usuario 7.....	26
Tabla 13: Sprint Backlog.....	27
Tabla 14: Análisis de Costos de Desarrollo.....	39
Tabla 15: Análisis de Costos Operativos.....	39
Tabla 16: Resumen de Precios.....	40
Tabla 17: Análisis de Riesgos.....	41
Tabla 18: Riesgos Identificados del Proyecto.....	41
Tabla 19: Matriz de Riesgos.....	42
Tabla 20: Análisis Cuantitativo de Riesgos.....	43
Tabla 21: Análisis Cuantitativo y Grado de Exposición.....	43
Tabla 22: Plan de Contingencia.....	45

Índice de ilustraciones

Ilustración 1: Gantt.....	14
Ilustración 2: Gantt Fechas	15
Ilustración 3: Gantt Línea de Tiempo.....	16
Ilustración 4: Procesos de Negocios.....	19
Ilustración 5: Estructura de Datos	29
Ilustración 6: Diagrama de Entidad Relación	30
Ilustración 7: Interfaz de Pantalla 1	31
Ilustración 8: Interfaz de Pantalla 2.....	32
Ilustración 9: Interfaz de Pantalla 3.....	33
Ilustración 10:Interfaz de Pantalla 4.....	33
Ilustración 11:Interfaz de Pantalla 5.....	34
Ilustración 12: Interfaz de Pantalla 6.....	35
Ilustración 13: Interfaz de Pantalla 7.....	35
Ilustración 14: Interfaz de Pantalla 8.....	36
Ilustración 15: Diagrama de Arquitectura	37
Ilustración 16: Riesgos Detectados	44

Resumen

A pesar de los grandes avances tecnológicos, la creación de sitios web se había convertido en una tarea cada vez más compleja y desafiante, que requería habilidades técnicas avanzadas y un proceso iterativo de diseño y codificación. En este contexto, el proyecto se llevó a cabo con el objetivo de aprovechar el potencial de los modelos de lenguaje de gran escala para simplificar y agilizar este proceso. Se desarrolló una solución innovadora que integraba un chatbot alimentado por un modelo de lenguaje avanzado, capaz de comprender solicitudes en lenguaje natural y generar componentes web. Esta herramienta se complementaba con una interfaz que permitía editar y revisar un historial versionado de cambios e iteración sobre los mismos, facilitando la rápida creación de diseños y estructuras web deseados. El resultado fue una plataforma que hizo más accesible el desarrollo web, permitiendo a usuarios con diversos niveles de experiencia técnica crear sitios web atractivos y funcionales de manera eficiente. Esta solución respondía a la creciente demanda de herramientas web sofisticadas pero fáciles de usar en un mercado digital en constante evolución.

Palabras claves: Inteligencia artificial, chatbot, componente web.

Abstract

Despite major technological advancements, website creation had become an increasingly complex and challenging task, requiring advanced technical skills and an iterative process of design and coding. In this context, the project was carried out with the aim of leveraging the potential of large-scale language models to simplify and streamline this process. An innovative solution was developed, integrating a chatbot powered by an advanced language model capable of understanding natural language requests and generating web components. This tool was complemented by an interface that allowed users to edit and review a versioned history of changes and iterated improvements, facilitating the rapid creation of desired web designs and structures. The result was a platform that made web development more accessible, enabling users with various levels of technical expertise to create attractive and functional websites efficiently. This solution responded to the growing demand for sophisticated yet user-friendly web tools in a constantly evolving digital market.

Keywords: Artificial Intelligence, chatbot, web component.

Título

Generación de sitios web con chatbot e inteligencia artificial

Introducción

El presente proyecto se desarrolló en el ámbito del diseño web, específicamente en la intersección entre la inteligencia artificial y la creación de páginas web. Ante la creciente demanda de herramientas de diseño web más accesibles y la complejidad que enfrentaron los usuarios sin conocimientos técnicos, este proyecto se propuso desarrollar un chatbot con inteligencia artificial. Este asistente virtual tuvo como objetivo guiar a los usuarios a través del proceso de diseño y creación de páginas web, interpretando sus necesidades en lenguaje natural y traduciéndolas en código funcional. A lo largo de su elaboración, el proyecto buscó simplificar significativamente el proceso de diseño web, haciéndolo más intuitivo y accesible para usuarios de todos los niveles de experiencia, al tiempo que ofreció un alto grado de personalización y flexibilidad en los resultados finales.

Antecedentes

En la actualidad una serie de LLM (Large Language Model) incurrieron en el mercado entre ellos ChatGPT (<https://chatgpt.com/>), Llama (<https://llama.com>), v0 (<https://v0.dev/>), que a su vez son un tipo específico de GPT (Generative Pre-trained Transformer), según Amazon (s.f.) podemos entender que:

Los transformadores generativos preentrenados, comúnmente conocidos como GPT, son una familia de modelos de redes neuronales que utilizan la arquitectura de transformadores y representan un avance clave en la inteligencia artificial (IA) que impulsa las aplicaciones de IA generativa, como ChatGPT. Los modelos GPT permiten a las aplicaciones crear texto y contenido (imágenes, música y más) de manera similar a como lo haría un ser humano y responder a preguntas de forma conversacional. Organizaciones de todos los sectores utilizan modelos GPT e IA

generativa para bots de preguntas y respuestas, resumen de textos, generación de contenidos y búsqueda. (p. 1).

Siguiendo a Amazon nos dice cómo funciona la interpretación de texto para un caso de uso particular:

Enfocado en el caso de uso de generar código como modelos de lenguaje, los modelos GPT pueden entender y escribir código informático en diferentes lenguajes de programación. Dichos modelos pueden ayudar a los alumnos al explicarles los programas de ordenador en un lenguaje cotidiano. Además, los desarrolladores experimentados pueden usar las herramientas GPT para auto sugerir fragmentos de código relevantes. (p. 7).

Estos modelos fueron accesibles mediante una interfaz que permitía el chat con los mismos IBM (s.f.) nos dice que:

Un chatbot es un programa informático que simula la conversación humana con un usuario final. No todos los chatbots están equipados con inteligencia artificial (IA), pero los chatbots modernos utilizan cada vez más técnicas de IA conversacional como el procesamiento del lenguaje natural (PLN) para comprender las preguntas de los usuarios y automatizar las respuestas. (p.1).

Siguiendo a IBM nos cuenta los inicios de los chatbots:

Los primeros chatbots eran esencialmente programas interactivos de preguntas frecuentes, que se basaban en un conjunto limitado de preguntas comunes con respuestas preescritas. Incapaces de interpretar el lenguaje natural, estas preguntas frecuentes generalmente requerían que los usuarios seleccionaran entre palabras clave

y frases sencillas para hacer avanzar la conversación. Estos chatbots rudimentarios y tradicionales son incapaces de procesar preguntas complejas, ni de responder a preguntas sencillas que no hayan sido previstas por los desarrolladores. (p. 11).

Con respecto a los componentes web Mozilla nos dice que:

Los Componentes Web son un paquete de diferentes tecnologías que te permiten crear elementos personalizados reutilizables — con su funcionalidad encapsulada apartada del resto del código — y utilizarlos en las aplicaciones web.

Finalizando, siguiendo a Mozilla nos cuenta el concepto y uso de estos:

Como desarrolladores, sabemos que reutilizar código tanto como sea posible es una buena idea. Esto tradicionalmente no es sencillo para estructuras de marcado personalizadas — piense en el complejo HTML (y sus estilos y scripts asociados) que en ocasiones se han tenido que escribir para renderizar controles de interfaz (UI) personalizados, y ahora usarlos múltiples veces puede crear un caos en la página si no se es cuidadoso.

Descripción Del Área Problemática

La creación de componentes web personalizados implica el uso de diversas tecnologías y lenguajes de programación, un proceso que requiere tiempo y conocimientos específicos por parte de desarrolladores y diseñadores web. Este aspecto del desarrollo afecta a proyectos de software de diferentes escalas, desde pequeñas aplicaciones hasta sistemas empresariales complejos. En el contexto actual, la variedad de frameworks disponibles, la necesidad de escribir código específico para cada componente y el tiempo necesario para visualizar los resultados son factores que influyen en el proceso de desarrollo.

Justificación

Se propuso desarrollar un sistema para generar componentes web mediante una interfaz de chat, con el objetivo de abordar el proceso de creación de interfaces. Se buscó aplicar tecnologías de procesamiento de lenguaje natural y frameworks de desarrollo web para implementar esta herramienta. El propósito fue ofrecer una alternativa en el método de creación de componentes web, dirigida a desarrolladores y diseñadores. Se consideró el potencial impacto en los tiempos y costos de desarrollo, así como en la accesibilidad para crear contenido web. El proyecto se planteó como una exploración de nuevas posibilidades en el campo del desarrollo web, con posibles implicaciones para empresas de desarrollo, instituciones educativas y usuarios en general. La propuesta se basó en la observación de las prácticas actuales en el desarrollo web y en las tendencias tecnológicas emergentes en el campo de la inteligencia artificial y el procesamiento del lenguaje natural.

Objetivo General del Proyecto

Crear un sistema de chatbot con inteligencia artificial que pueda interpretar y responder consultas en lenguaje natural a través de una página web, mantener una conversación con el usuario para recopilar sus requisitos y preferencias a la vez que se itera y almacena el progreso sobre un componente, generar automáticamente código HTML y CSS para construir páginas web personalizadas sin necesidad de conocimientos previos de programación.

Objetivos Específicos del Proyecto

Con base en el objetivo general planteado, los objetivos específicos para este proyecto son los siguientes:

1. Evaluar las opciones disponibles de chatbot y las actividades típicas al momento de desarrollar una web para obtener el mejor resultado al momento de interactuar con el mismo.

2. Desarrollar un flujo de conversación estructurado que permita al bot recopilar sistemáticamente la información necesaria sobre estilo, contenido y funcionalidades deseadas para la página web del usuario.
3. Diseñar un sistema de almacenamiento de iteraciones y versiones de los componentes creados, que permita al usuario y al chatbot revisar el historial de cambios y ajustes realizados, asegurando la coherencia en el proceso de diseño y edición.

Marco Teórico Referencial

Dominio Del Problema

Para comenzar a abordar los temas pertinentes primero hay que saber que es una página web “Una página web es un documento electrónico accesible a través de internet que contiene información organizada y presentada de forma visual.” (Godaddy, 2024). Siguiendo definiremos que es la inteligencia artificial. “La inteligencia artificial (IA) es la capacidad de una máquina construida, como una computadora, para simular o duplicar tareas humanas cognitivas.” (Cloudflare, 2024). A continuación, siguiendo a Cloudflare, definiremos que es un chatbot o bot conversacional. “Los Bots conversacionales son programas informáticos diseñados para interactuar con los seres humanos en conversaciones de voz o de texto.” Para finalizar continuando con Cloudflare definiremos que son los grandes modelos de datos “Un modelo lingüístico grande (LLM) es un tipo de programa de inteligencia artificial (IA) que puede reconocer y generar texto, entre otras tareas”.

TICs

A continuación, se examinan las tecnologías que han respaldado este proyecto.

React: “La biblioteca para interfaces de usuario web y nativas” (React 2024.).

PostgreSQL: “Es un potente sistema de base de datos relacional de objetos de código abierto” (Postgresql 2024.).

Typescript: “TypeScript es un lenguaje de programación fuertemente tipado que se basa en JavaScript y le brinda mejores herramientas a cualquier escala.” (Typescript 2024).

VScode: “Visual Studio Code es un editor de código fuente ligero pero potente que se ejecuta en el escritorio y está disponible para Windows, macOS y Linux.” (VScode 2024.).

Nextjs: “Next.js es un marco de React para crear aplicaciones web de pila completa. Se utiliza React Components para crear interfaces de usuario y Next.js para funciones y optimizaciones adicionales.” (NextJs 2024.).

LLM: “Los modelos de lenguaje de gran tamaño, también conocidos como LLM, son modelos de aprendizaje profundo muy grandes que se preentrenan con grandes cantidades de datos. El transformador subyacente es un conjunto de redes neuronales que consta de un codificador y un decodificador con capacidades de autoatención” (Amazon 2024.).

Competencia

La tabla a continuación presenta una visión general de las diversas empresas que se consideran como posibles competencias.

Tabla 1: Competencia

Empresa	Fuente	Generación de componentes web	Chat	Exportar proyecto	Exportar código
OpenAI	https://chatgpt.com/	No	Si	No	No
Anthropic	https://claude.ai/	No	Si	No	No
Vercel	https://v0.dev/	Si	Si	No	Si

Fuente: Elaboración propia

Diseño Metodológico

Herramientas Metodológicas

Para el desarrollo de este proyecto, se ha considerado fundamental la elección de una metodología ágil que permita una gestión eficiente y flexible del proceso de desarrollo para esto se escogió Scrum que se define como “un marco de administración que los equipos utilizan para organizarse por cuenta propia y trabajar en aras de alcanzar un objetivo común.” (Amazon s.f).

Herramientas De Desarrollo

En el desarrollo del proyecto se utilizaron diferentes tecnologías, presentes tanto en la capa de presentación, Front End, como en la capa de acceso a datos, Back End. En ambos casos se compartió el lenguaje de tipado estático Typescript, se eligió para nuestro editor de código VScode el cual nos permite obtener herramientas de autocompletado y gestión de archivos. En el caso de nuestra capa de presentación se usó Nextjs, un marco de trabajo basado en ReactJs para crear las diferentes interfaces de nuestra página web y añadir reactividad entre componentes HTML. Para nuestra persistencia de datos se utilizó la base de datos Postgresql. En nuestra capa de datos se utilizó un Large Language Model (LLM) llamado llamagpt para lograr el chatbot y crear componentes HTML. Así, el conjunto de estas tecnologías posibilitó la creación de nuestro proyecto.

Recolección De Datos

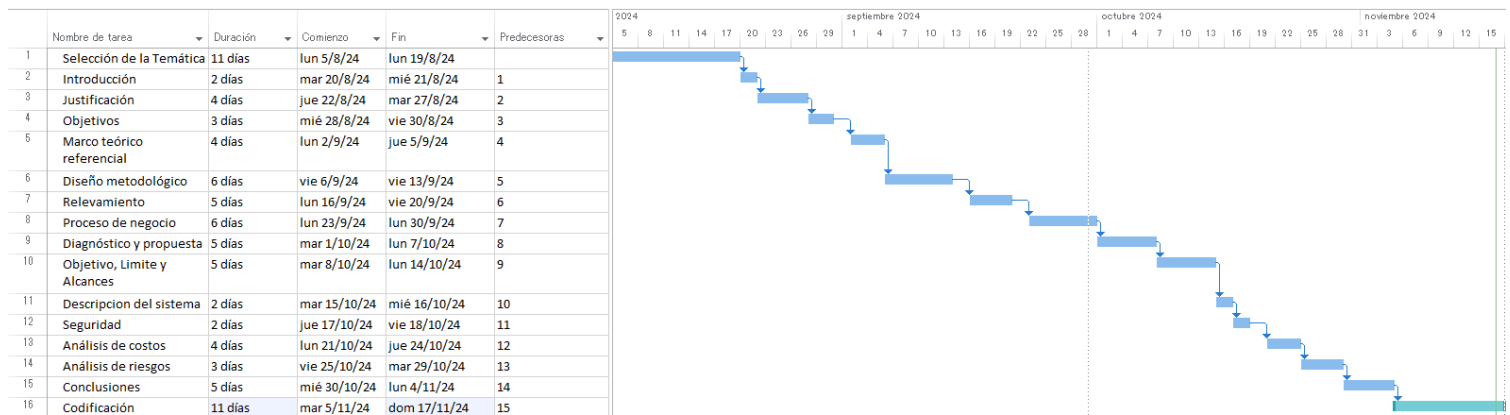
Desde el punto de vista del proyecto, la técnica de observación ayudó a percibir el estado actual de la inteligencia artificial y el desarrollo web. Este método permitió identificar las tendencias actuales en la integración de IA en herramientas de desarrollo web, así como las prácticas comunes de los desarrolladores al crear componentes web. La observación se centró en cómo los profesionales interactúan con las tecnologías existentes, las dificultades

que enfrentan en el proceso de desarrollo, y las áreas donde la asistencia de IA podría ser más beneficiosa.

Planificación Del Proyecto

El plan de actividades se detalla en el siguiente diagrama de Gantt.

Ilustración 1: Gantt



Fuente: Elaboración propia.

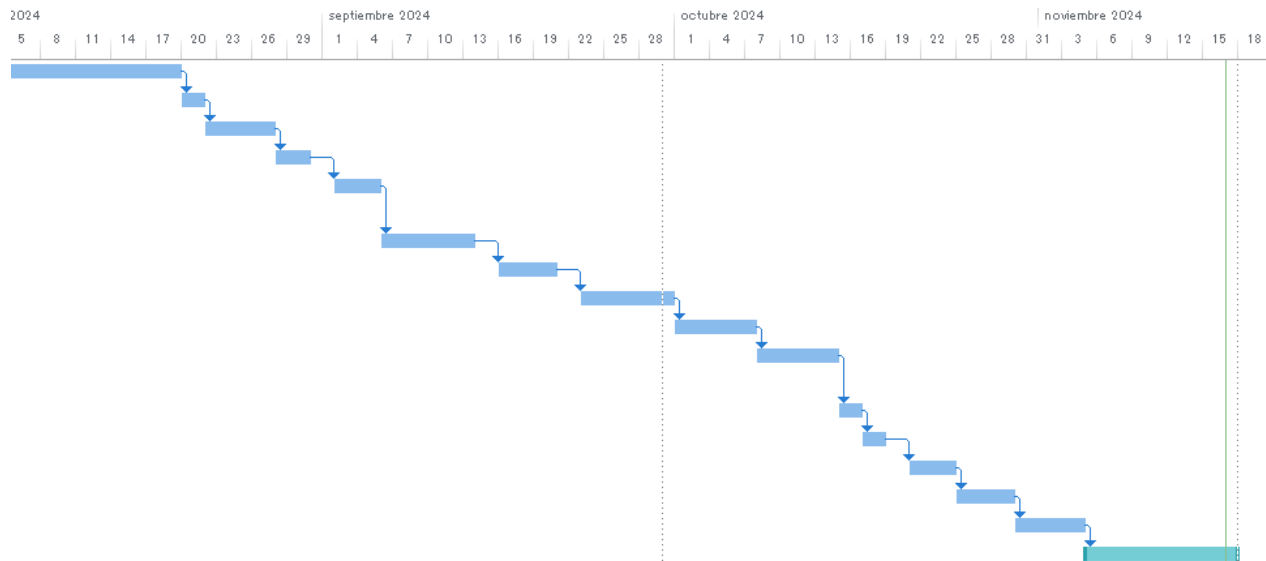
Para su mayor apreciación se divide el Gantt en dos secciones.

Ilustración 2: Gantt Fechas

	Nombre de tarea ▼	Duración ▼	Comienzo ▼	Fin ▼	Predecesoras ▼
1	Selección de la Temática	11 días	lun 5/8/24	lun 19/8/24	
2	Introducción	2 días	mar 20/8/24	mié 21/8/24	1
3	Justificación	4 días	jue 22/8/24	mar 27/8/24	2
4	Objetivos	3 días	mié 28/8/24	vie 30/8/24	3
5	Marco teórico referencial	4 días	lun 2/9/24	jue 5/9/24	4
6	Diseño metodológico	6 días	vie 6/9/24	vie 13/9/24	5
7	Relevamiento	5 días	lun 16/9/24	vie 20/9/24	6
8	Proceso de negocio	6 días	lun 23/9/24	lun 30/9/24	7
9	Diagnóstico y propuesta	5 días	mar 1/10/24	lun 7/10/24	8
10	Objetivo, Limite y Alcances	5 días	mar 8/10/24	lun 14/10/24	9
11	Descripcion del sistema	2 días	mar 15/10/24	mié 16/10/24	10
12	Seguridad	2 días	jue 17/10/24	vie 18/10/24	11
13	Análisis de costos	4 días	lun 21/10/24	jue 24/10/24	12
14	Análisis de riesgos	3 días	vie 25/10/24	mar 29/10/24	13
15	Conclusiones	5 días	mié 30/10/24	lun 4/11/24	14
16	Codificación	11 días	mar 5/11/24	dom 17/11/24	15

Fuente: Elaboración propia.

Ilustración 3: Gantt Línea de Tiempo



Fuente: Elaboración propia.

Relevamiento

Relevamiento Estructural

Por tratarse de usuarios remotos no existe una ubicación geográfica y estructura específica para definir.

Relevamiento Funcional

Una vez analizados los datos recolectados a través de las técnicas de recopilación, se concluye que, actualmente, no se utiliza una estructura formal para los procesos funcionales en el desarrollo web. Por ello, este relevamiento se basa en la particularidad de las personas que utilizan la web, herramientas de desarrollo y los agentes que los rodean. A continuación, se detallan los principales actores identificados en este ecosistema:

Desarrolladores Web: Profesionales encargados de la creación y mantenimiento de sitios y aplicaciones web. Incluye tanto a desarrolladores front-end como back-end.

Diseñadores UX/UI: Responsables de la experiencia del usuario y la interfaz visual de los productos web.

Usuarios Finales: Individuos que interactúan con los sitios y aplicaciones web desarrollados.

A continuación, se presentan los procesos relevados.

Proceso: Desarrollo de Prototipo Rápido

- Roles:
 - Desarrollador Web
 - Diseñador UX/UI
- Pasos: comienza con el diseñador UX/UI creando bocetos iniciales basados en los requerimientos del proyecto. El desarrollador web utiliza estos bocetos para generar un prototipo básico. Tras refinar el código y añadir funcionalidades, el diseñador UX/UI revisa el prototipo y sugiere ajustes. Este ciclo de desarrollo, revisión y ajuste entre el diseñador UX/UI y el desarrollador web se repite hasta que se logra un prototipo que cumpla con los objetivos y especificaciones del proyecto.

Proceso: Creación de Componentes Web

- Roles:

- Desarrollador Web
- Diseñador UX/UI
- Pasos: conceptualización por parte del diseñador UX/UI, quien define el propósito y funcionalidad del componente. Luego, se desarrollan mockups y prototipos visuales. El desarrollador web planifica la estructura técnica y crea una versión inicial del componente. Se integran los estilos y la interactividad, seguido de una fase de optimización del código. El componente se somete a pruebas exhaustivas de funcionalidad, compatibilidad y accesibilidad. Finalmente, se documenta su uso y se realiza una revisión conjunta entre diseñador y desarrollador antes de su integración en el sistema.

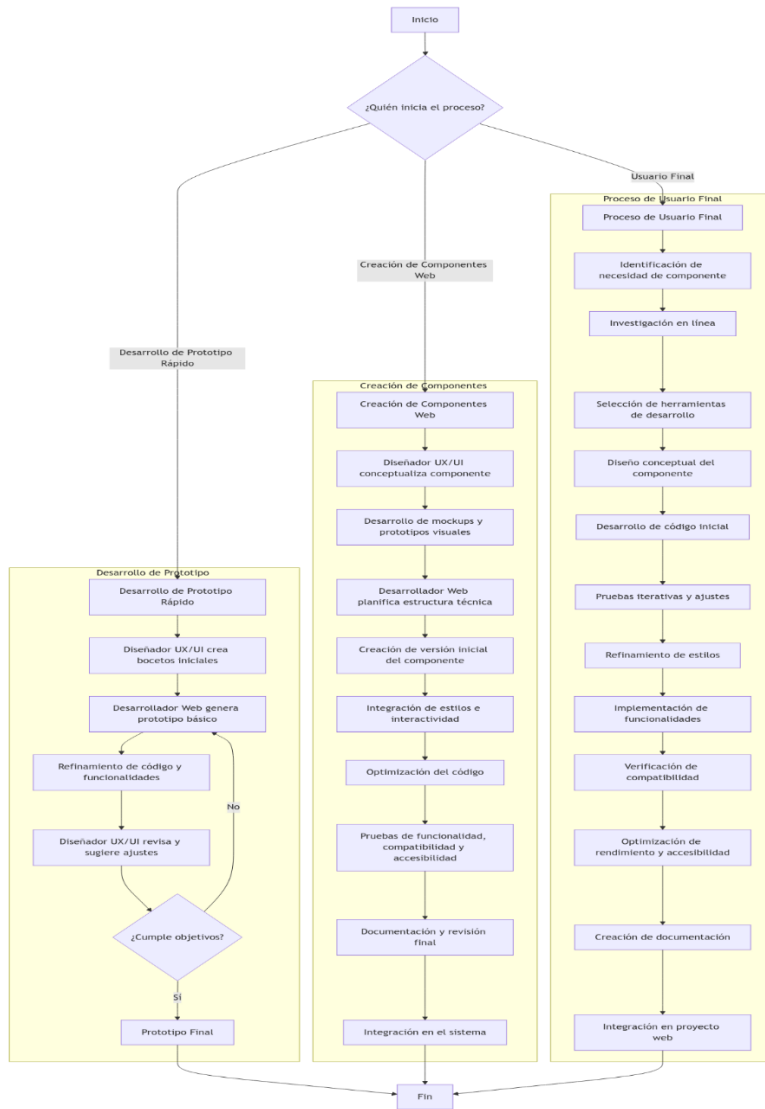
Proceso: Usuario Final en la Creación de Componentes Web

- Roles:
 - Usuario Final
- Pasos: la identificación de la necesidad de un componente específico, seguido de una investigación en línea sobre componentes similares. El usuario selecciona las herramientas de desarrollo adecuadas y realiza un diseño conceptual del componente. Luego, procede a desarrollar el código inicial, involucrando HTML, CSS y JavaScript, y realiza pruebas iterativas para ajustar el funcionamiento. El usuario refina los estilos, implementa las funcionalidades requeridas y verifica la compatibilidad en diferentes plataformas. Finalmente, optimiza el rendimiento y la accesibilidad del componente, crea documentación básica y lo integra en su proyecto web más amplio.

Procesos de Negocios

Se exhibe en un flujograma que representa los procesos relevados.

Ilustración 4: Procesos de Negocios



Fuente: Elaboración propia.

Diagnóstico y Propuesta

Luego del relevamiento de nuestros procesos, a continuación, se exponen las problemáticas halladas.

Tabla 2: Desarrollo de Prototipo Rápido

Nombre del proceso: Desarrollo de Prototipo Rápido	
Problemas	Causas
1. Interpretación incorrecta de los bocetos iniciales por parte del desarrollador web.	<ul style="list-style-type: none"> - Falta de detalles en los bocetos. - Comunicación insuficiente entre diseñador y desarrollador. - Ausencia de un lenguaje común para describir elementos de diseño.
2. Retrasos en la generación del prototipo básico	<ul style="list-style-type: none"> - Subestimación de la complejidad de los requerimientos. - Falta de priorización clara de las características a implementar. - Interrupciones frecuentes en el flujo de trabajo del desarrollador.

Fuente: Elaboración propia.

Tabla 3: Creacion de Componentes Web

Nombre del proceso: Creacion de Componentes Web	
Problemas	Causas
1. Fallos en pruebas de compatibilidad y accesibilidad	<ul style="list-style-type: none"> - Definición insuficiente de los estándares de compatibilidad y accesibilidad. - Falta de herramientas o conocimientos para realizar pruebas exhaustivas. - Omisión de escenarios de prueba importantes.
2. Desalineación entre el propósito definido y la funcionalidad implementada	<ul style="list-style-type: none"> - Comunicación imprecisa entre diseñador y desarrollador. - Falta de documentación clara sobre los requisitos funcionales. - Cambios no comunicados en la conceptualización inicial.

Fuente: Elaboración propia.

Tabla 4: Usuario Final en la Creación de Componentes Web

Nombre del proceso: Usuario Final en la Creación de Componentes Web	
Problemas	Causas
1. Dificultad para identificar la necesidad específica del componente	<ul style="list-style-type: none"> - Falta de claridad en los requisitos del proyecto. - Limitado conocimiento de las posibilidades de los componentes web. - Ausencia de un análisis detallado de las funcionalidades requeridas.
1. Errores en el desarrollo del código inicial	<ul style="list-style-type: none"> - Conocimientos insuficientes de HTML, CSS y JavaScript. - Falta de un enfoque estructurado en el desarrollo. - Copia y pegado de código sin entender su funcionamiento.
2. Dificultades en el refinamiento de estilos	<ul style="list-style-type: none"> - Comprensión limitada de CSS avanzado. - Falta de atención a los detalles de diseño. - Inconsistencia en la aplicación de estilos.

Fuente: Elaboración propia.

Propuesta

El sistema que se desarrolló contribuyó a la generación de sitios webs con componentes personalizables y reutilizables facilitando la construcción visual de la web para usuarios menos expertos y para aquellos más experimentados que buscan una alternativa a su flujo habitual de trabajo, utilizando un agente virtual para atender a los usuarios dando una respuesta diferente según la particularidad de la consulta realizada, con una construcción visual, y facilitando un resultado ventajoso sobre la competencia que no cumplen con estas características.

Objetivos, Límites y Alcances del Prototipo

Objetivo Del Prototipo

Desarrollar un prototipo web con un chatbot que genere y exporte componentes web reutilizables para la construcción de nuevos sitios exportables.

Límites

Desde el inicio de una nueva generación de un componente por parte del usuario hasta su exportación final en la creación de una nueva web.

Alcances

- Creación de cuenta
- Generación del componente web
- Registro de componentes generados
- Personalización del componente web
- Exportación de componentes web
- Exportación de un nuevo sitio web

Descripción del Sistema

Product Backlog

A continuación, se detalla el product backlog utilizado.

Tabla 5: Product Backlog

ID	Historias de usuarios	Prioridad	Puntos de historia	Dependencias
HU-01	Registro del usuario	Media	5	

HU-02	Ingreso del usuario	Media	5	HU-001
HU-03	Iniciar conversación	Alta	10	HU-002
HU-04	Modificar resultado	Alta	7	HU-003
HU-05	Guardar componente personalizado	Media	6	HU-004
HU-06	Visualizar biblioteca de componentes	Media	5	HU-005
HU-07	Exportar sitio web generado	Alta	9	HU-003, HU-004, HU-005

Fuente: Elaboración propia.

Historias De Usuario

A continuación, se detallan las Historias de Usuario identificadas para el proyecto, cada una describiendo una funcionalidad específica desde la perspectiva del usuario final:

Tabla 6: Historia de Usuario 1

ID	HU-001	Nombre	Registro del usuario
Descripción	Como usuario, quiero registrarme en el sistema para poder iniciar sesión		
Criterios de aceptación	<ol style="list-style-type: none"> 1. Dado un correo electrónico que ya se encuentre registrado, cuando este sea ingresado por el usuario, entonces, el sistema mostrará un aviso de error. 2. Dada una contraseña no alfanumérica y/o menor a 6 dígitos, cuando esta sea ingresada, entonces, el sistema le avisará de la restricción. 3. Dado un campo incompleto cuando el usuario intenta registrarse, entonces, el sistema dará aviso de que todos los campos son requeridos. 		
Prioridad	Media	Puntos de historia	5

Fuente: Elaboración propia.

Tabla 7: Historia de Usuario 2

ID	HU-002	Nombre	Ingreso del usuario
Descripción	Como usuario registrado, quiero iniciar sesión en el sistema para acceder a mi cuenta y mis proyectos.		
Criterios de aceptación	<ol style="list-style-type: none"> 1. Datos credenciales válidas, cuando el usuario las ingresa, entonces el sistema permite el acceso y redirige al dashboard. 2. Datos credenciales inválidas, cuando el usuario las ingresa, entonces el sistema muestra un mensaje de error y no permite el acceso. 3. Dado un usuario que ha olvidado su contraseña, cuando solicita restablecerla, entonces el sistema envía un correo con instrucciones para crear una nueva contraseña. 		
Prioridad	Media	Puntos de historia	5

Fuente: Elaboración propia.

Tabla 8: Historia de Usuario 3

ID	HU-003	Nombre	Iniciar conversación
Descripción	Como usuario, quiero iniciar una conversación con el agente virtual para recibir asistencia en la creación de mi sitio web.		
Criterios de aceptación	1. Dado un usuario autenticado, cuando selecciona la opción de iniciar chat, entonces se abre una ventana de conversación con el agente virtual.		

	2. Dada una pregunta del usuario, cuando el agente virtual la procesa, entonces proporciona una respuesta. 3. Dada una solicitud de ayuda específica, cuando el usuario la expresa, entonces el agente virtual ofrece opciones o sugerencias relacionadas con esa solicitud.		
Prioridad	Alta	Puntos de historia	10

Fuente: Elaboración propia.

Tabla 9: Historia de Usuario 4

ID	HU-004	Nombre	Modificar resultado
Descripción	Como usuario, quiero poder modificar los resultados generados por el agente virtual para ajustarlos a mis preferencias específicas.		
Criterios de aceptación	1. Dado un componente web generado, cuando el usuario solicita modificarlo, entonces el sistema muestra opciones de edición. 2. Dadas las modificaciones realizadas por el usuario, cuando las confirma, entonces el sistema actualiza el componente y muestra el resultado. 3. Dada una modificación no válida, cuando el usuario intenta aplicarla, entonces el sistema muestra un mensaje de error explicando por qué no es posible.		
Prioridad	Alta	Puntos de historia	7

Fuente: Elaboración Propia.

Tabla 10: Historia de Usuario 5

ID	HU-005	Nombre	Guardar componente personalizado
Descripción	Como usuario, quiero guardar los componentes web que he personalizado para poder reutilizarlos en futuros proyectos.		
Criterios de aceptación	1. Dado un componente personalizado, cuando el usuario elige guardarlo, entonces el sistema lo almacena en la biblioteca del usuario. 2. Dada una biblioteca de componentes guardados, cuando el usuario la accede, entonces puede ver y seleccionar sus componentes personalizados. 3. Dado un componente guardado, cuando el usuario lo selecciona para un nuevo proyecto, entonces el sistema lo inserta con todas sus personalizaciones.		
Prioridad	Media	Puntos de historia	6

Fuente: Elaboración propia.

Tabla 11: Historia de Usuario 6

ID	HU-006	Nombre	Visualizar biblioteca de componentes
Descripción	Como usuario, quiero ver una biblioteca de todos los componentes disponibles y personalizados para facilitar la creación de mi sitio web.		
Criterios de aceptación	<ol style="list-style-type: none"> 1. Dado un usuario autenticado, cuando accede a la biblioteca de componentes, entonces ve una lista organizada de componentes disponibles. 2. Dada la biblioteca de componentes, cuando el usuario aplica filtros, entonces ve los componentes que coinciden con los criterios seleccionados. 3. Dado un componente en la biblioteca, cuando el usuario lo selecciona, entonces puede ver una vista previa y detalles del componente. 		
Prioridad	Media	Puntos de historia	5

Fuente: Elaboración propia.

Tabla 12: Historia de Usuario 7

ID	HU-007	Nombre	Exportar sitio web generado
Descripción	Como usuario, quiero poder exportar el sitio web que he creado.		
Criterios de aceptación	<ol style="list-style-type: none"> 1. Dado un sitio web completado, cuando el usuario selecciona la opción de exportar, entonces el sistema genera un archivo comprimido con todos los archivos necesarios. 2. Dada la exportación del sitio, cuando se completa, entonces el sistema proporciona la descarga al sistema. 		
Prioridad	Alta	Puntos de historia	9

Fuente: Elaboración propia.

Sprint Backlog

A continuación, se exhiben los sprints utilizados para el desarrollo del proyecto.

Tabla 13: Sprint Backlog

Sprint	Historia de usuario	ID	Tareas	Prioridad	Estimado	Estado	
1	Registro del usuario	HU-01	Diseñar interfaz de registro	Alta	2 días	Hecho	
			Implementar validación de datos	Baja	2 días	Hecho	
			Crear base de datos de usuarios	Alta	3 días	Hecho	
	Ingreso del usuario	HU-02	Diseñar interfaz de login	Alta	1 día	Hecho	
			Implementar autenticación	Alta	2 días	Hecho	
			Implementar validación de datos	Baja	3 días	Hecho	
2	Iniciar conversación	HU-03	Diseñar interfaz de chat	Alta	2 días	Hecho	
			Implementar conexión con agente virtual	Alta	1 día	Hecho	
			Desarrollar lógica de procesamiento de consultas	Alta	3 días	Hecho	
	Modificar resultado	HU-04	Diseñar interfaz de modificación	Alta	2 días	Hecho	
			Desarrollar lógica de procesamiento de consultas de edición	Alta	3 días	Hecho	
				Diseñar interfaz de guardado	Alta	2 días	Hecho

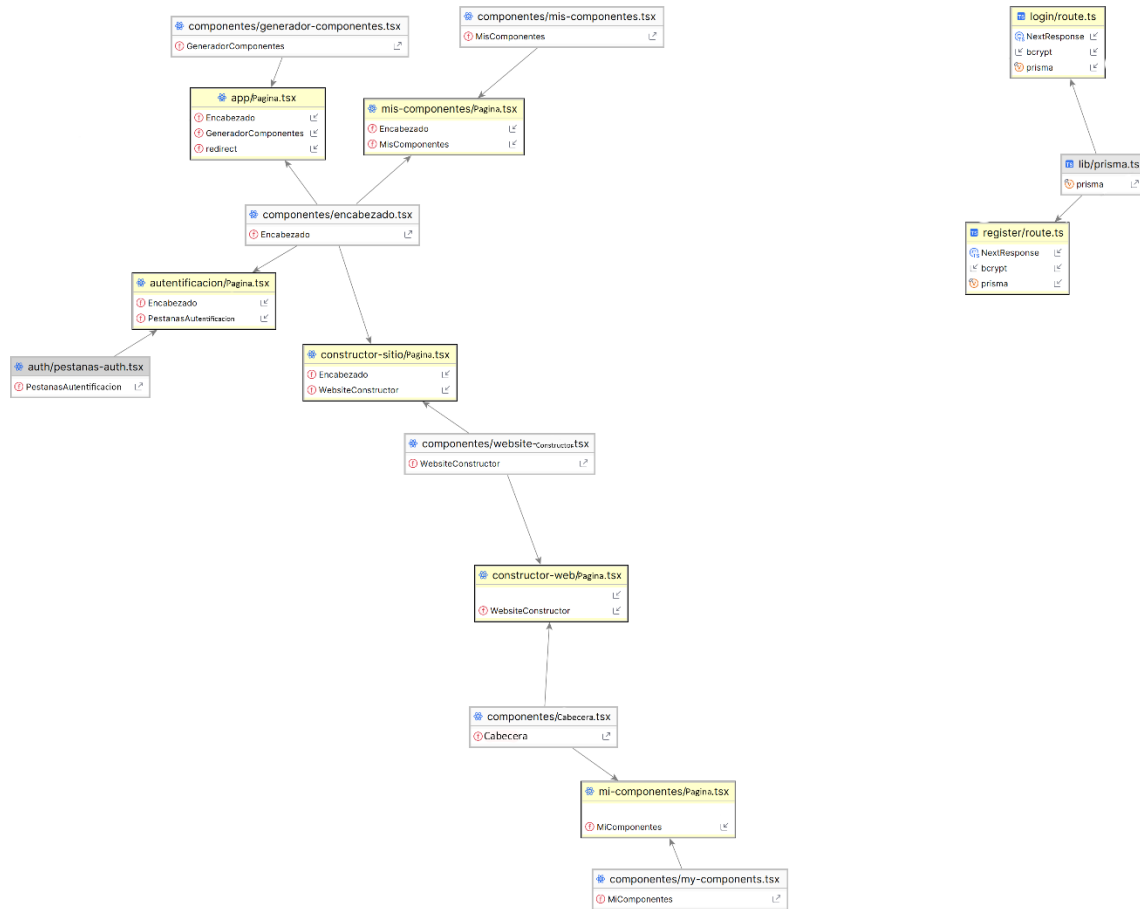
	Guardar componente personalizado	HU-05	Desarrollar lógica de procesamiento de consultas de persistencia	Alta	3 días	Hecho
3	Visualizar biblioteca de componentes	HU-06	Diseñar interfaz de biblioteca	Alta	2 días	Hecho
			Mostrar componentes	Media	2 días	Hecho
	Exportar sitio web generado	HU-07	Diseñar interfaz de exportar	Alta	1 día	Hecho
			Implementar el constructor de sitio	Media	2 días	Hecho
			Implementar exportar	Alta	3 días	Hecho

Fuente: Elaboración propia.

Estructura De Datos

Diagrama de clases. Se representa la estructura del sistema a través de las clases, atributos, operaciones y asociaciones que existen entre los objetos identificados en el sistema.

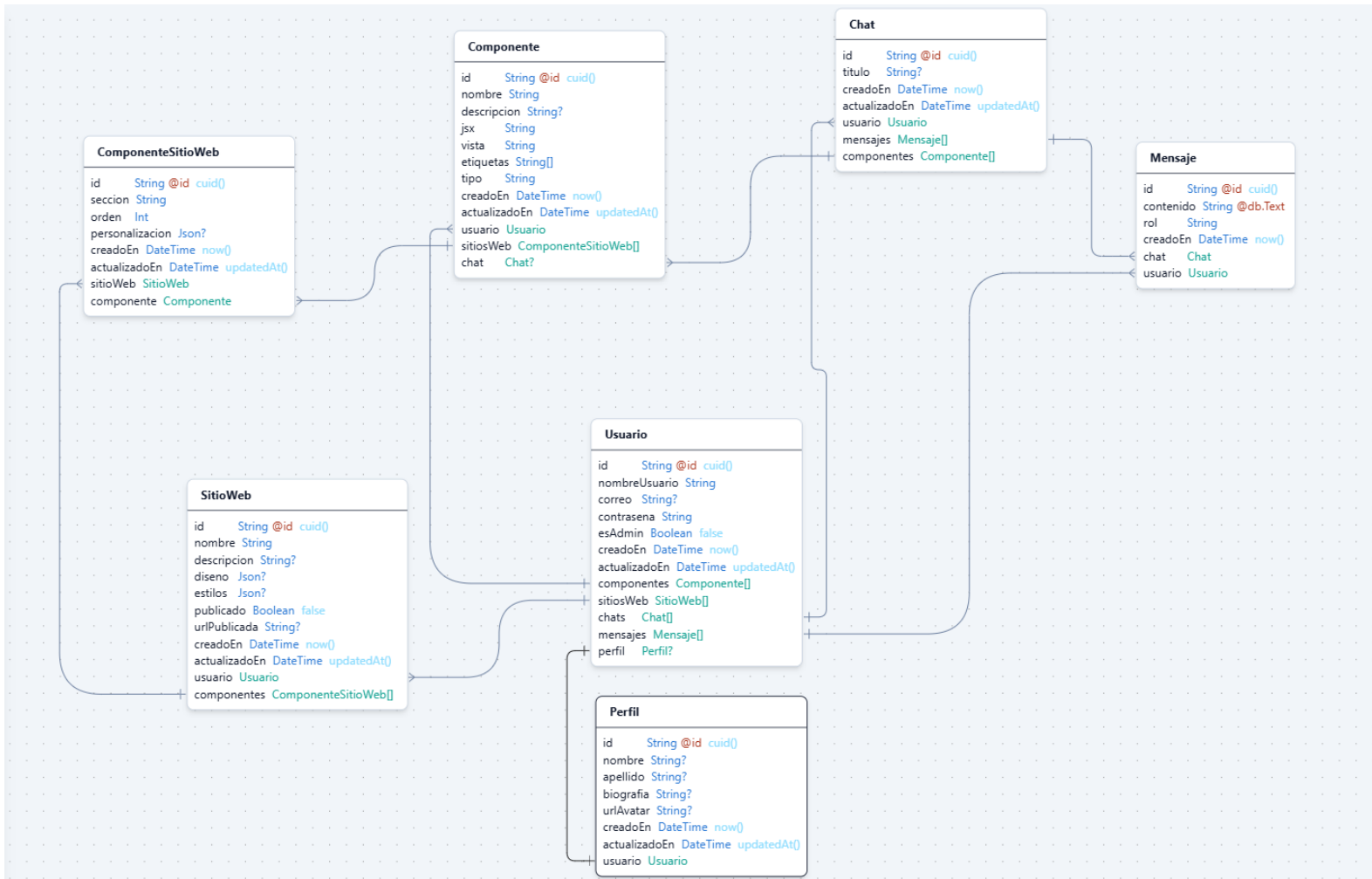
Ilustración 5: Estructura de Datos



Fuente: Elaboración propia.

Diagrama de entidad relación. Se representan las diferentes entidades del sistema y como se relacionan entre sí, junto a sus atributos y claves.

Ilustración 6: Diagrama de Entidad Relación



Fuente: Elaboración propia.

Prototipos De Interfaces De Pantallas

Crear cuenta es un proceso donde el usuario proporciona información personal y credenciales (como nombre, correo electrónico y contraseña) en un formulario. El sistema verifica la unicidad del correo, valida los datos ingresados y, si son correctos, crea una nueva cuenta de usuario en la base de datos. Finalmente, se confirma el registro al usuario.

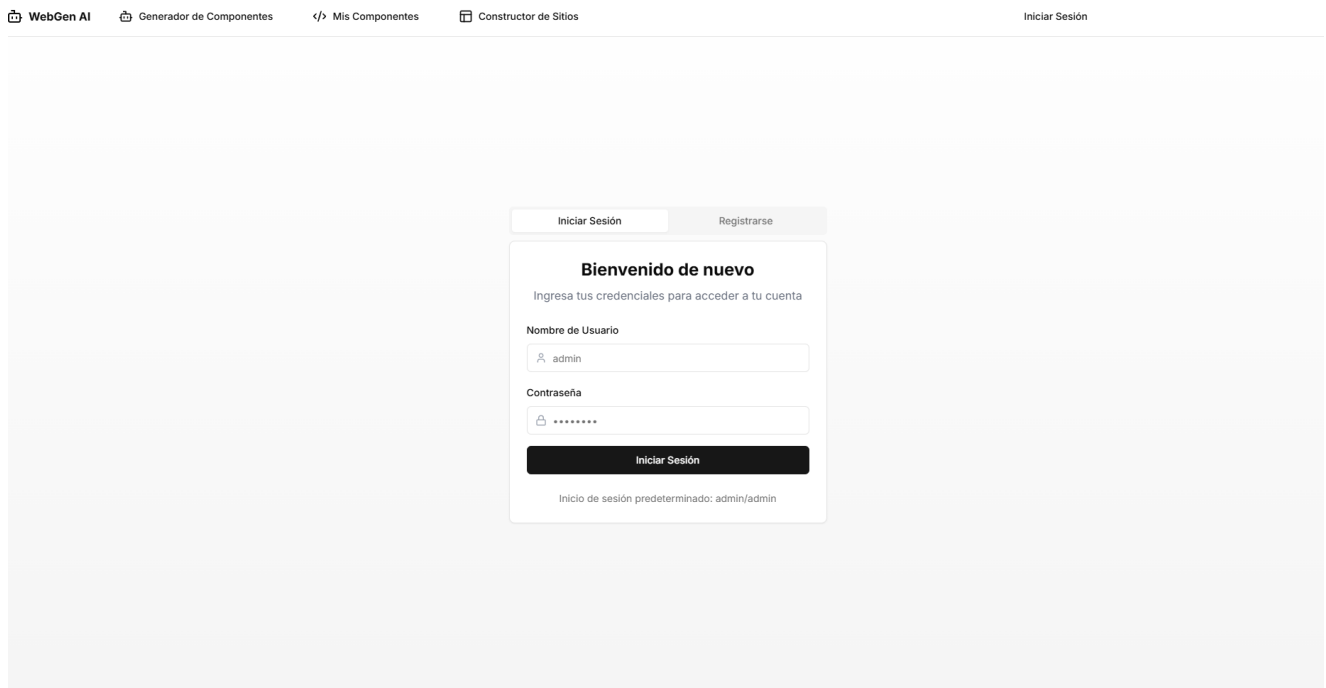
Ilustración 7: Interfaz de Pantalla 1

The screenshot displays a web application interface with a navigation bar at the top containing the following items: 'WebGen AI', 'Generador de Componentes', 'Mis Componentes', 'Constructor de Sitios', and 'Iniciar Sesión'. The main content area features a central form titled 'Crear una cuenta' (Create an account) with the subtitle 'Ingresa tu información para comenzar' (Enter your information to get started). The form includes three input fields: 'Nombre de Usuario' (Username) with the value 'juanperez', 'Contraseña' (Password), and 'Confirmar Contraseña' (Confirm Password). A 'Crear cuenta' button is located at the bottom of the form. The form also has two tabs at the top: 'Iniciar Sesión' (Login) and 'Registrarse' (Register).

Fuente: Elaboración propia.

El login es un proceso de autenticación donde el usuario ingresa sus credenciales (usuario y contraseña) en un formulario. El sistema verifica esta información y, si es correcta, permite el acceso al área personal del usuario.

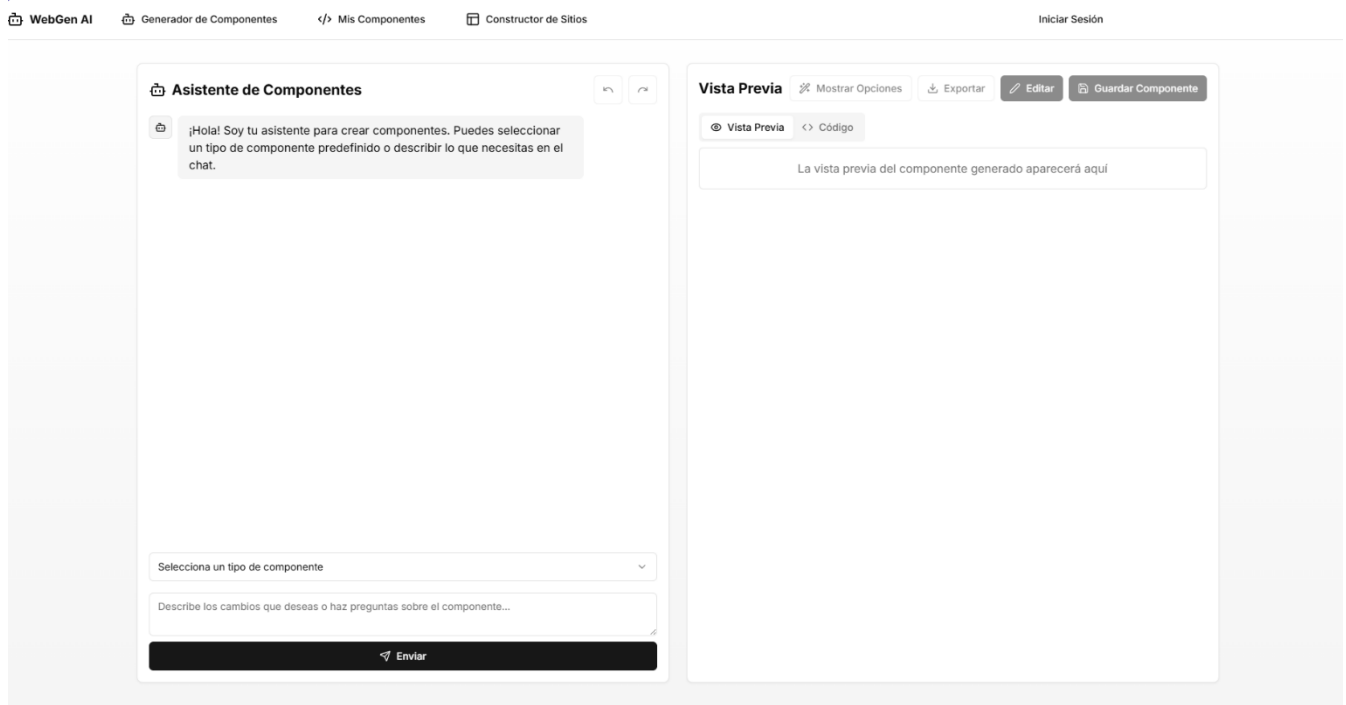
Ilustración 8: Interfaz de Pantalla 2



Fuente: Elaboración propia.

El chat para generar componentes es una interfaz de diálogo donde el usuario describe en lenguaje natural el componente web deseado. El sistema interpreta la solicitud, hace preguntas para clarificar detalles si es necesario, y luego genera automáticamente el código del componente basado en la conversación. El usuario puede revisar, ajustar y aprobar el componente generado a través del chat.

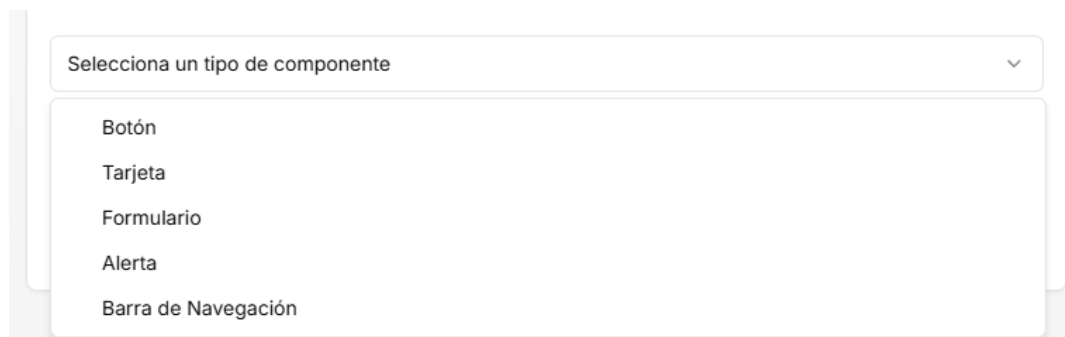
Ilustración 9: Interfaz de Pantalla 3



Fuente: Elaboración propia.

Selección predefinida para la creación de componentes típicos.

Ilustración 10: Interfaz de Pantalla 4



Fuente: Elaboración propia.

Vista previa donde se puede inspeccionar el código, cambiar los colores, la tipografía, el diseño y el contenido además de exportar únicamente el componente en formato html o guardarlo y usarlo en la creación del sitio.

Ilustración 11: Interfaz de Pantalla 5



Fuente: Elaboración propia.

Historial de versiones de cada componente la cual se va creando a medida que se modifica cada uno de ellos.

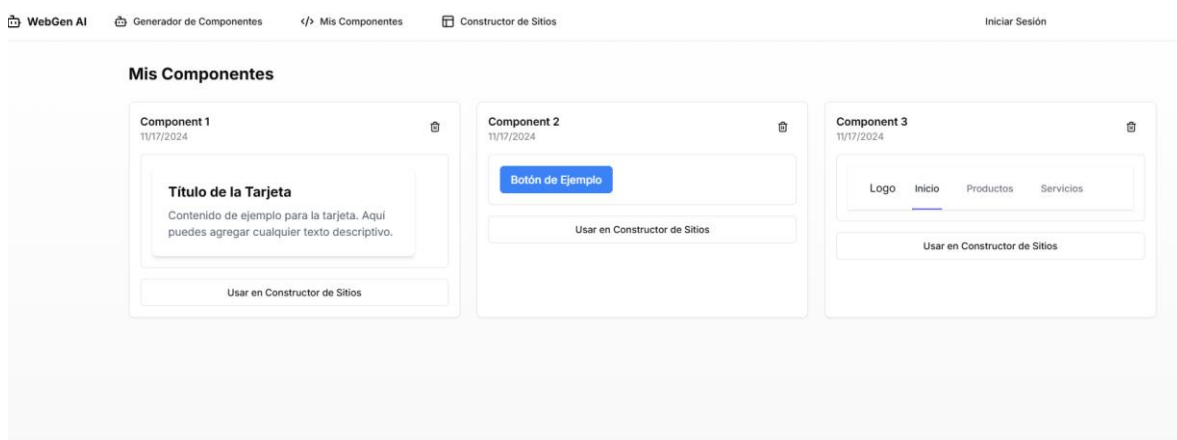
Ilustración 12: Interfaz de Pantalla 6



Fuente: Elaboración propia.

En la Interfaz mis componentes se muestran los componentes creados los mismos pueden ser eliminados o usados de inmediato en el constructor de sitios.

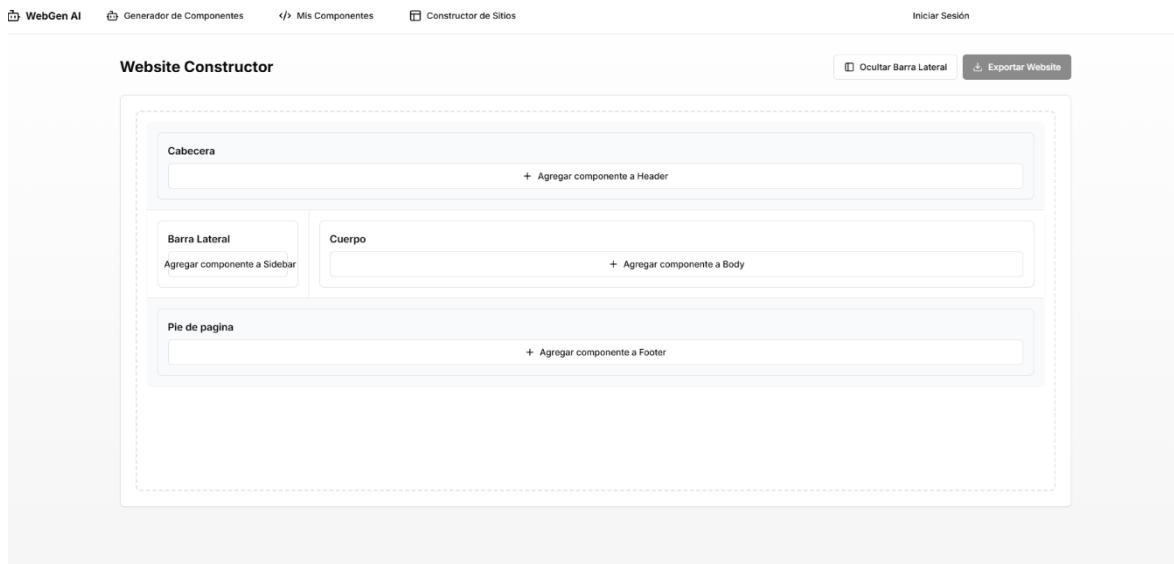
Ilustración 13: Interfaz de Pantalla 7



Fuente: Elaboración propia.

Creacion de página web perteneciente al constructor de sitios, se agrega los componentes generados para luego exportar en un archivo html.

Ilustración 14: Interfaz de Pantalla 8

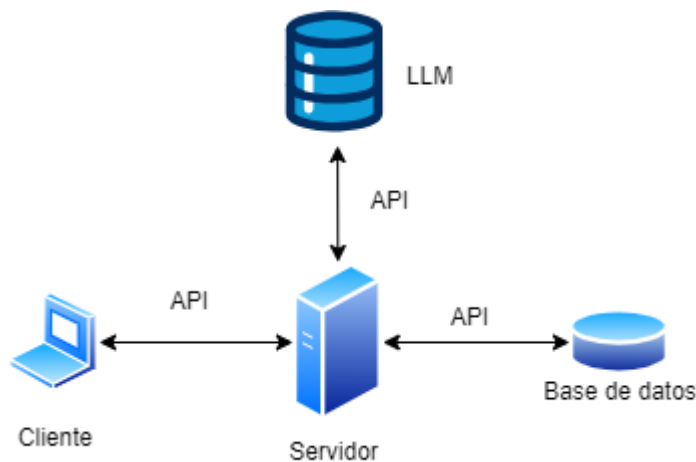


Fuente: Elaboración propia.

Diagrama De Arquitectura

En el siguiente diagrama se puede visualizar la arquitectura del sistema llevada a cabo en el prototipado, la comunicación entre servicios se basa en el uso de API (Interfaz de programación de aplicaciones), donde el cliente se comunica con el servidor y este según sea necesario con el modelo de datos y la base de datos para luego devolver una respuesta.

Ilustración 15: Diagrama de Arquitectura



Fuente: Elaboración propia.

Seguridad

Se detallan a continuación las políticas y actividades que se han definido para garantizar la seguridad para la información que el sistema utiliza.

Acceso a la aplicación

Para usuarios nuevos se les solicita un correo electrónico y un nombre de usuario válido y único para registrarse, para activar su cuenta, debe validar su correo electrónico mediante un código de 6 dígitos alfanuméricos que se le envía a la dirección registrada, la contraseña debe tener un mínimo de 8 caracteres y contener al menos un número, una letra mayúscula y un carácter especial, se requiere cambiar las contraseñas cada 90 días, después de 5 intentos fallidos de acceso, la cuenta se bloquea temporalmente, las contraseñas se almacenan utilizando algoritmos de hash y cifrado de un solo sentido, los privilegios de acceso se asignan según roles y permisos definidos para cada usuario, se mantienen registros de auditoría de todos los intentos de acceso exitosos y fallidos, se implementan mecanismos de autenticación adicionales, como preguntas de seguridad o autenticación de dos factores para accesos desde ubicaciones o dispositivos no reconocidos, las cuentas nuevas

automáticamente reciben un rol de usuario común mientras que las cuentas con permisos de administrador son creadas manualmente de ante mano mediante código sin los pasos que deben seguir los usuarios comunes.

Política de respaldo de información

Se abordan las políticas para respaldar la información de los usuarios y del código de la aplicación.

Nube: La primera copia de los datos se aloja en servicios cloud de Amazon Web Services (AWS). Se realizan respaldos diarios automáticos a las 03:00 horas, manteniendo las últimas 7 copias. Esta copia incluye datos de usuarios, componentes web generados y configuraciones del sistema. La información se encripta utilizando AES-256 y el acceso está restringido mediante políticas de IAM (Identity and Access Management) de AWS.

Servidor local: La segunda copia se almacena en un servidor ubicado en las oficinas del equipo. Este servidor se sincroniza diariamente con la copia en la nube a las 05:00 horas. Se mantienen respaldos de los últimos 30 días. Además de los datos de usuarios y componentes, aquí se almacena la primera copia del código fuente de la aplicación. El acceso físico y lógico al servidor está restringido al administrador de sistemas y al encargado de seguridad.

Almacenamiento externo: La tercera copia de los datos y la segunda del código fuente se guardan en discos duros externos encriptados. Estos se actualizan semanalmente, los lunes a las 08:00 horas, y se almacenan en una caja fuerte ignífuga en una ubicación separada del servidor local. Se conservan respaldos mensuales durante un año. Solo el encargado de seguridad de datos y el director del proyecto tienen acceso a estos discos, que requieren autenticación de dos factores para su acceso.

Análisis de Costos

A continuación, se presentan los costos de recursos humanos y de gastos operativos. Estos fueron calculados el 25 de octubre del 2024 considerando que el proyecto tendrá una duración de 4 meses. Adicionalmente, debemos tener en cuenta que el valor del dólar al día

de la fecha en el Banco de la Nación Argentina es de 1 U\$S = \$967. (Banco de la Nación Argentina [BNA], 2024).

Los valores de los salarios fueron tomados del sitio web del Consejo Profesional de Ciencias Informáticas de la Provincia de Buenos Aires (CPCIBA, 2024).

Tabla 14: Análisis de Costos de Desarrollo

Rol	Salario mensual	Cantidad	Meses totales	Subtotal
Jefe de Proyectos	\$1.827.618,09	1	4	\$7.310.472,36
Analista Programador Senior	\$1.697.430,72	1	4	\$6.789.722,88
Programador IA	\$2.321.547,83	1	4	\$9.286.191,32
Desarrollador Full Stack Developer	\$1.112.996,25	1	4	\$8.475.256,56
Analista Testing de Aplicaciones	\$1.646.481,38	1	3	\$4.939.444,14
Total				\$36.801.087,26

Fuente: Elaboración propia.

A continuación, se exponen los recursos operativos.

Tabla 15: Análisis de Costos Operativos

Recurso	Tipo de Recurso	Fuente	Cantidad	Subtotal
Computadora Portátil Lenovo ThinkPad X1 Carbon 12va Gen (14" Intel)	Hardware	https://www.lenovo.com/ar	5	\$17.419.995

Base de Datos Postgresql	Software	https://www.postgresql.org	1	\$0
Github	Servicio	https://github.com/	1	\$0
LLM llama	Servicio	https://llama.meta.com/	1	\$0
Windows 11	Software	https://www.microsoft.com/	1	\$179.999
Total				\$17.599.994

Fuente: Elaboración propia.

Considerando todos los costos, en la siguiente tabla se presentan los montos totales necesarios para poder llevar a cabo el proyecto.

Tabla 16: Resumen de Precios

Tipo de Costo	Precio Final
Costos de Recursos Humanos	\$36.801.087,26
Costos Software	\$179.999
Costos Hardware	\$17.419.995
Costos Servicio	\$0
Total	\$54.401.081,26

Fuente: Elaboración propia.

Análisis de Riesgos

Todo proyecto conlleva ciertos riesgos inherentes. Identificar y comprender las posibles amenazas es crucial para poder planificar y tomar medidas proactivas que mitiguen estos riesgos potenciales. A continuación, se examinarán los riesgos más relevantes que podrían surgir, sus causas subyacentes y se delinearán planes de contingencia enfocados en abordar aquellos riesgos que podrían tener un impacto más significativo.

Tabla 17: Análisis de Riesgos

ID	Tipo	Riesgo	Probabilidad	Impacto
1	Proyecto	Pruebas incorrectas o escasas	Media	Alto
2	Proyecto	Error de diseño (se debe rediseñar el producto).	Bajo	Alto
3	Proyecto	Pérdida de la información	Media	Medio
4	Proyecto	Vulneración del sistema	Media	Alto
5	Tecnico	La planificación de proyecto es muy optimista	Media	Medio
6	Tecnico	Caída de los servidores	Media	Alto

Fuente: Elaboración propia.

Tabla 18: Riesgos Identificados del Proyecto

Tipo	Riesgo	Causa
Proyecto	Pruebas incorrectas o escasas	Falta de experiencia del encargado de pruebas y el desarrollador
Proyecto	Error de diseño (se debe rediseñar el producto).	El desarrollo de funciones de software erróneas requiere volver a diseñarlas e implementarlas.
Tecnico	Pérdida de la información	Fallas en las copias de seguridad
Proyecto	Vulneración del sistema	Sistema poco seguro.
Proyecto	La planificación de proyecto es muy optimista	Debido a la falta de experiencia, la

		planificación no es realista.
Tecnico	Caída de los servidores	Falta de mantenimiento en los servidores.

Fuente: Elaboración propia.

Una vez que se han identificado los riesgos del proyecto, se evaluarán las probabilidades de que ocurran y sus posibles impactos. Para esto, se utilizó la siguiente matriz.

Tabla 19: Matriz de Riesgos

			Impacto				
			Muy bajo	Bajo	Medio	Alto	Muy alto
			1	2	3	4	5
Probabilidad	Muy alta	0,9	0,9	1,8	2,7	3,6	4,5
	Alta	0,7	0,7	1,4	2,1	2,8	3,5
	Media	0,5	0,5	1	1,5	2	2,5
	Baja	0,3	0,3	0,6	0,9	1,2	1,5
	Muy baja	0,1	0,1	0,2	0,3	0,4	0,5

Fuente: Elaboración propia.

Análisis cuantitativo ordenado por su probabilidad de ocurrencia de mayor a menor.

Tabla 20: Análisis Cuantitativo de Riesgos

Riesgo	Probabilidad de ocurrencia	Impacto
Pruebas incorrectas o escasas	78%	3
Error de diseño (se debe rediseñar el producto).	58%	4
Pérdida de la información	55%	4
Vulneración del sistema	40%	3
La planificación de proyecto es muy optimista	25%	2
Caída de los servidores	10%	5

Fuente: Elaboración propia.

Se calcula la exposición al riesgo.

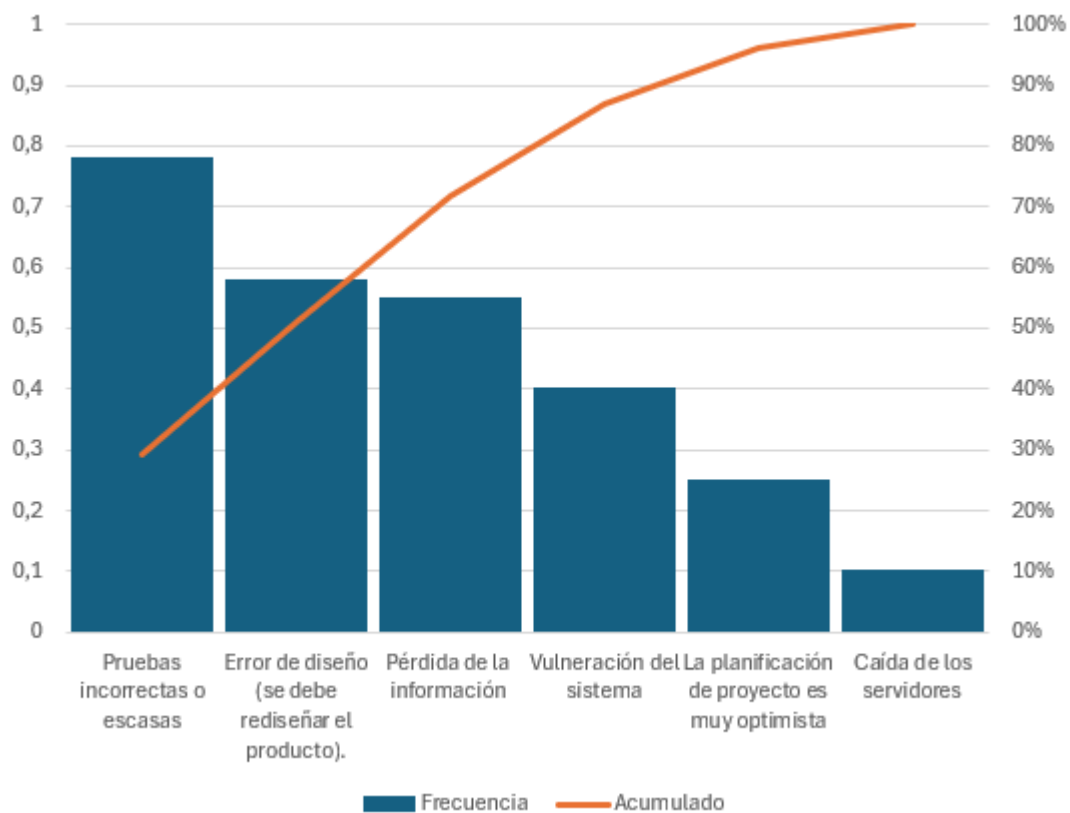
Tabla 21: Análisis Cuantitativo y Grado de Exposición

Riesgo	Probabilidad de ocurrencia	Efecto o impacto	Grado de exposición	Porcentaje	Porcentaje acumulado
Pruebas incorrectas o escasas	78%	3	2,34	25,8%	30%
Error de diseño (se debe rediseñar el producto).	58%	4	2,32	25,6%	53%
Pérdida de la información	55%	4	2,2	24,3%	71%
Vulneración del sistema	40%	3	1,2	13,2%	84%
La planificación de proyecto es muy optimista	25%	2	0,5	5,5%	97%
Caída de los servidores	10%	5	0,5	5,5%	100%

Fuente: Elaboración propia.

En el siguiente gráfico se ordenaron los riesgos detectados según su probabilidad de ocurrencia de mayor a menor. Esto nos sirvió para aplicar el principio de Pareto el cual nos dice que “el 80% de los efectos están provocados por el 20% de las posibles causas” (Universitat de València, s.f.).

Ilustración 16: Riesgos Detectados



Fuente: Elaboración propia.

Plan de Contingencia

Se presenta un plan para los riesgos de mayor exposición.

Tabla 22: Plan de Contingencia

Riesgo	Plan de contingencia
Pruebas incorrectas o escasas	Se implementarán revisiones de calidad, ampliación de pruebas automatizadas y capacitaciones constantes. En caso de errores, se corregirán casos de prueba, se analizarán incidencias y se ejecutarán pruebas adicionales. Revisión periódica y auditorías garantizarán la mejora continua
Error de diseño	Se realizarán revisiones técnicas exhaustivas, validaciones con usuarios y pruebas de concepto. En caso de ocurrencia, se corregirá el diseño rápidamente y se analizarán los impactos. Auditorías periódicas asegurarán mejoras continuas y reducirán futuras fallas.
Pérdida de la información	Se implementarán respaldos automáticos y encriptación de datos. En caso de pérdida, se restaurará la información desde copias de seguridad y se investigará la causa. Auditorías regulares garantizarán la integridad de los datos y la efectividad del plan.

Fuente: Elaboración propia.

Conclusiones

El presente proyecto ha dado como resultado el desarrollo de una página web con un chatbot con inteligencia artificial impulsado por un gran modelo de datos con el objetivo de desarrollar una herramienta innovadora que puede comprender solicitudes en lenguaje natural y generar componentes web reutilizables que facilite y agilice el proceso de creación de sitios web y su posterior exportación para que sea utilizable. La problemática abordada fue la necesidad de contar con herramientas que, mediante el uso de lenguaje natural, permitan simplificar y acelerar el proceso de creación de sitios web. En este sentido, muchos diseñadores y desarrolladores web se enfrentan al desafío de optimizar tiempos y recursos en la creación de componentes reutilizables y exportables que puedan adaptarse a distintos contextos sin perder eficiencia ni calidad en el diseño. En términos de resultados el proyecto no solo cumplió con los objetivos propuestos, sino que también abre la posibilidad de futuras mejoras e implementaciones, como la integración de una mayor variedad de estilos y funciones avanzadas en los componentes generados.

Desde una perspectiva profesional, este proyecto me permitió adquirir conocimientos valiosos en el campo de la inteligencia artificial aplicada, así como experiencia práctica en el desarrollo de aplicaciones web.

En el ámbito personal, este desafío me brindó la oportunidad de expandir mis horizontes, explorar nuevas tendencias tecnológicas y enfrentar los retos inherentes al desarrollo de soluciones innovadoras que implementan inteligencia artificial.

Demo

En el siguiente enlace se podrá descargar el prototipo desarrollado para su ejecución, como así también, el código fuente de la aplicación y un instructivo para su puesta en marcha.

https://1drv.ms/f/s!AhYzxhIlcP9WgsBC8hr8p4_KkEKN8w?e=JsLcSs

Referencias

- Amazon Web Services. (s.f.). *¿Qué es GPT?* Recuperado el 2 de octubre de 2024 de <https://aws.amazon.com/es/what-is/gpt/>
- Amazon Web Services. (s.f.). *¿Qué es Scrum?* Recuperado el 2 de octubre de 2024 de <https://aws.amazon.com/es/what-is/scrum/>
- Amazon. (s.f.). Recuperado el 6 de octubre de 2024 de <https://aws.amazon.com/es/what-is/large-language-model/>
- Banco de la Nación Argentina [BNA]. (s.f.). Cotización de divisas. Recuperado el 11 de octubre de 2024 <https://www.bna.com.ar/>
- Cloudflare. (s.f.). *¿Qué es un bot conversacional?* Recuperado el 2 de octubre de 2024 de <https://cloudflare.com/es-es/learning/bots/what-is-a-chatbot/>
- Cloudflare. (s.f.). *¿Qué es un modelo lingüístico (LLM) grande?* Recuperado el 2 de octubre de 2024 de <https://www.cloudflare.com/es-es/learning/ai/what-is-large-language-model/>
- Cloudflare. (s.f.). *What is artificial intelligence (AI)?* Recuperado el 2 de octubre de 2024 de <https://cloudflare.com/learning/ai/what-is-artificial-intelligence/>
- Consejo Profesional de Ciencias Informáticas de la Provincia de Buenos Aires (CPCIBA, s.f.). Recuperado el 2 de octubre de 2024 de <https://plataforma.cpciba.org.ar/>
- IBM. (s.f.). *¿Qué es un chatbot?* Recuperado el 2 de octubre de 2024 de <https://www.ibm.com/es-es/topics/chatbots>
- Mozilla. (s.f.). *Web Components* Recuperado el 2 de octubre de 2024 de https://developer.mozilla.org/es/docs/Web/API/Web_components
- Nextjs. (s.f.). Recuperado el 15 de septiembre de 2024 de <https://nextjs.org/docs>
- Postgresql. (s.f.). Recuperado el 6 de octubre de 2024 de <https://www.postgresql.org>
- React (s.f.). Recuperado el 6 de octubre de 2024 de <https://react.dev/>
- Typescript (s.f.). Recuperado el 6 de octubre de 2024 de <https://www.typescriptlang.org/>
- Universitat de València. (s.f.). Recuperado el 15 de noviembre de 2024 de <https://www.uv.es/ceaces/calidad/t3/66.htm>
- Vscode (s.f.). Recuperado el 6 de octubre de 2024 de <https://code.visualstudio.com/docs>