

TECNOLÓGICO NACIONAL DE MÉXICO

Instituto Tecnológico de Culiacán



Opción:

TITULACIÓN INTEGRAL

Build A Chat: Aplicación web para la creación de asistentes virtuales conversacionales institucionales

ALUMNO(A):

Emilio Payán Ortega

CARRERA:

Ingeniería en Sistemas Computacionales

Director de Tesis:

Dr. Aldo Uriarte Portillo

No. DE CONTROL:

20170777

Culiacán, Sinaloa, junio de 2026



Instituto Tecnológico de Culiacán

DIVISION DE ESTUDIOS PROFESIONALES
ASUNTO: **Autorización de Impresión**

Culiacán, Sinaloa, 16/06/2026 15:24

C. EMILIO PAYAN ORTEGA
PASANTE(S) DE LA CARRERA DE INGENIERIA EN SISTEMAS COMPUTACIONALES
P R E S E N T E.

De acuerdo con fallo emitido por la Comisión Revisora Integrada por:

Cédula Profesional

Presidente:	Doctor en Ciencias de la Ingeniería Aldo Uriarte Portillo	13158766
Secretario:	Maestría en Ciencias en Ciencias Computacionales Arcelia Judith Bustillos Martínez	13118410
Vocal:	Maestría en Administración de Negocios, Área Finanzas Víctor Manuel Bátiz Beltrán	6715966

Y considerando que reúne los requisitos establecidos en la Opción: **TITULACIÓN INTEGRAL: TESIS PROFESIONAL " BUILD A CHAT"**, damos a usted nuestra autorización para que proceda a imprimir su trabajo profesional.

Sin otro particular, quedo de usted.

ATENTAMENTE
Excelencia en Educación Tecnológica®

CONCEPCION MENDOZA ROSALES
LIC. CONCEPCION MENDOZA ROSALES
JEFA DE LA DIVISION DE ESTUDIOS PROFESIONALES

 **Educación** | 
Secretaría de Educación Pública
INSTITUTO TECNOLÓGICO DE CULIACÁN
DEPARTAMENTO DE
DIVISIÓN DE ESTUDIOS PROFESIONALES



2026
año de
Margarita
Maza



AGRADECIMIENTO

Agradezco a la institución del Instituto tecnológico de Culiacán por ser quien aceptó como empresa para el desarrollo de este proyecto teniendo el respaldo de los diferentes docentes de esta institución para la resolución de dudas y dar un seguimiento óptimo.

Agradezco al Dr. Aldo Uriarte Portillo por ser quien me dio entrada a este proyecto que estuve desarrollando, por darme la capacitación para el desarrollo, la guía para cualquier duda que tuve y una retroalimentación activamente para poder terminar el proyecto de forma exitosa.

Agradezco a la M.C. Arcelia Judith Bustillos Martínez por ayudarme y darme guía desde el lado administrativo, ser quien me fue ayudando durante todo este tiempo incentivando mi trabajo para dar lo mejor de mí y que pudiera finalizar este proyecto de manera exitosa.

Agradezco al Dr. Víctor Manuel Bátiz Beltrán por brindar apoyo al desarrollo del proyecto resolviendo dudas a mediados del desarrollo brindando herramientas que expandieron el alcance del proyecto.

Build A Chat: Aplicación web para la creación de asistentes virtuales conversacionales

RESUMEN

Este trabajo presenta el desarrollo de Build A Chat, una plataforma web para crear y administrar asistentes virtuales institucionales orientados al apoyo de consultas académicas y administrativas. El proyecto surge a partir de una problemática observada en el Instituto Tecnológico de Culiacán, donde parte de la información sobre trámites, fechas, pagos, becas, reinscripciones y otros procesos escolares se encuentra distribuida en diferentes medios, lo que puede generar confusión, pérdida de tiempo y consultas repetitivas al personal administrativo.

Para atender esta necesidad, se desarrolló una aplicación web que permite a un usuario administrativo crear asistentes virtuales, organizar información mediante un árbol de conocimiento, revisar conversaciones y actualizar contenidos. Por su parte, el usuario general puede interactuar con el asistente a través de una interfaz conversacional similar a la de aplicaciones de mensajería. La plataforma integra herramientas como Flask, Vue.js, Socket.IO, PostgreSQL y Google Gemini, lo que permite combinar gestión de usuarios, almacenamiento de mensajes, comunicación en tiempo real e inteligencia artificial generativa.

El desarrollo se realizó bajo un enfoque tecnológico aplicado, con una estrategia iterativa e incremental basada en módulos funcionales. Como resultado, se obtuvo una primera versión de la plataforma con funciones de registro, inicio de sesión, creación y edición de asistentes, consulta conversacional, historial de mensajes y panel administrativo. Build A Chat se plantea como una herramienta de apoyo para mejorar la organización de la información institucional y facilitar una primera orientación a los estudiantes, sin sustituir la atención directa del personal administrativo.

Palabras clave: asistente virtual, Build A Chat, chatbot, aplicación web, inteligencia artificial generativa, comunicación institucional.

CONTENIDO

1	INTRODUCCIÓN	1
1.1	Antecedentes	2
1.2	Descripción del problema	3
1.3	Hipótesis	6
1.3.1	Hipótesis Nula (H0)	6
1.3.2	Hipótesis Alternativa (H1)	7
1.4	Objetivos	7
1.4.1	Objetivo general	7
1.4.2	Objetivos específicos	7
1.5	Justificación	8
1.6	Alcance	10
1.7	Contribución Técnica	10
1.8	Estructura de la tesis:	11
2	FUNDAMENTO TEÓRICO	13
2.1	Aplicaciones web	13
2.1.1	Arquitectura básica de una aplicación web	15
2.1.2	Comunicación en tiempo real con WebSocket/Socket.IO	16
2.2	Asistentes virtuales conversacionales	18
2.2.1	Chatbot en contextos educativos e institucionales	21
2.3	Inteligencia Artificial	22
2.3.1	Aprendizaje automático	24
2.3.2	Aprendizaje profundo	26
2.3.3	Procesamiento del lenguaje natural en asistentes conversacionales	27

2.4	Inteligencia artificial generativa.....	27
2.5	Experiencia de usuario en interfaces conversacionales	30
2.6	Trabajos relacionados	32
2.6.1	Asistentes virtuales en la educación	33
2.6.2	Asistentes virtuales que implementan inteligencia artificial.....	34
3	METODOLOGÍA	37
3.1	Enfoque metodológico	37
3.2	Requerimientos funcionales y de calidad	38
3.3	Arquitectura general del proyecto	38
3.4	Herramientas de desarrollo	41
3.5	Interfaces de Usuario.....	42
3.6	Diseño de la Base de datos	46
3.7	Casos de Uso	47
3.8	Estrategia de pruebas y validación	49
4	RESULTADOS.....	51
4.1	Cumplimiento de requerimientos	51
4.2	Evaluación con usuarios	52
5	CONCLUSIONES	63
5.1	Limitaciones	65
5.2	Trabajo futuro.....	65
6	COMPETENCIAS DESARROLLADAS	66
7	FUENTES DE INFORMACIÓN	68

Índice de Figuras

Figura 1 - Línea de tiempo de asistentes virtuales	20
Figura 2 - Tipos de aprendizaje automático	25
Figura 3 - Asistente virtual creado para WhatsApp	34
Figura 4 - Arquitectura general del proyecto	39
Figura 5 - Arquitectura multicapa.....	40
Figura 6 - Interfaz principal de Build A Chat	43
Figura 7 - Inicio de Sesión	43
Figura 8 - Crear nuevo Chatbot	44
Figura 9 - Crear un chatbot.....	45
Figura 10 - Modificar un chatbot	45
Figura 11 - Estructura de un chatbot	46
Figura 12 - Modelo Relacional de la Base de datos	47
Figura 13 - Casos de uso de un usuario estándar	48
Figura 14 - Casos de uso de un usuario Administrador	48
Figura 15 - Nodos del Aspirante	54
Figura 16 - Nodos oferta educativa	54
Figura 17 - Nodos del estudiante.....	55
Figura 18 - Mapa de Conocimiento de un asistente virtual	56
Figura 19 - Ejemplo de Conversaciones	57
Figura 20 - Participantes.....	58
Figura 21 - Nodos para una prueba piloto hecha en un Restaurant	59
Figura 22 - Mapa de Conocimiento del asistente virtual para el restaurant.....	60
Figura 23 - Sugerencias del menú.....	61
Figura 24 - Platillo más solicitado	62

Índice de Tablas

Tabla 1 Situaciones observadas.....	5
Tabla 2 - Criterios de experiencia de usuario.....	31
Tabla 3 - Trabajos relacionados	36
Tabla 4 - Actividades desarrolladas en la creación de Build A Chat.....	38
Tabla 5 - Requerimientos Funcionales	38
Tabla 6 - Casos de prueba	50
Tabla 7 - Tabla de cumplimiento de requerimientos funcionales	51
Tabla 8 - Participantes	53

1 INTRODUCCIÓN

En la vida universitaria es muy común que los estudiantes se enfrenten a dudas relacionadas con trámites, fechas o requisitos institucionales. Frases como “no sabía que tenía que inscribirme en esa página”, “no me enteré de la fecha límite de pago” o “no encontré la información de la beca” reflejan situaciones que, aunque parecen simples, pueden afectar directamente la experiencia académica del alumno. En ocasiones, dichas dudas han provocado retrasos, pérdida de tiempo, confusión, frustración e incluso la falta de aprovechamiento de oportunidades importantes.

Esta situación muestra una problemática recurrente en muchas instituciones educativas: la información se encuentra dispersa en varios medios oficiales, como el sitio web, las redes sociales, comunicados, oficinas o grupos informales de estudiantes. Como consecuencia, el solicitante debe buscar en varios sitios diferentes, interpretar instrucciones administrativas y asegurarse de poder cumplir con los requisitos en tiempo y forma, que no siempre se presentan de forma clara o accesible.

En el contexto actual, donde los estudiantes están acostumbrados a recibir información de manera rápida mediante herramientas digitales, resulta necesario contar con soluciones que faciliten la comunicación entre la institución y su comunidad estudiantil. La falta de un medio centralizado puede generar confusión tanto para los alumnos como para el personal administrativo, ya que muchas preguntas se repiten constantemente, sobre todo en periodos de inscripción, reinscripción, pagos, becas o cierre de semestre.

A partir de esta necesidad surge Build A Chat, un proyecto enfocado en el desarrollo de un asistente virtual educativo apoyado en inteligencia artificial generativa. La finalidad del sistema es ofrecer una plataforma sencilla y accesible que permita consultar información académica y administrativa del Instituto Tecnológico de Culiacán. A través de este asistente, se busca que los estudiantes puedan obtener orientación sobre temas como convocatorias, registros, evaluaciones, fechas de pago y otros procesos institucionales.

Una característica de la plataforma Build A Chat es que permite a un usuario administrador generar y personalizar asistentes virtuales de acuerdo con las necesidades de la institución. Esto

significa que la información puede organizarse mediante una estructura previamente definida, de modo que las respuestas estén relacionadas con los procesos reales que los estudiantes consultan con mayor frecuencia. De esta manera, el sistema no pretende sustituir por completo la atención humana, sino funcionar como un primer punto de apoyo para resolver dudas comunes y canalizar los casos que requieran atención más específica.

Además, el proyecto busca ofrecer una experiencia de uso cercana a las aplicaciones de mensajería que los estudiantes utilizan en su vida diaria. La intención es que la interacción con el asistente sea sencilla, clara y familiar, evitando que el usuario tenga que aprender a manejar una plataforma complicada. Más allá del desarrollo técnico, Build A Chat responde a una necesidad práctica: ayudar a que los estudiantes encuentren información confiable y oportuna sin depender siempre de acudir presencialmente a una oficina o revisar múltiples canales institucionales.

1.1 Antecedentes

El proyecto Build A Chat nace como una propuesta para que una institución educativa pueda generar y gestionar su propio asistente virtual de una manera flexible. El propósito radica en no depender de una solución cerrada o cuya modificación sea compleja, sino que el personal administrativo tenga la posibilidad de estructurar y organizar la información que desea compartir tanto con los público en general y estudiantes; a su vez, y actualizarla cuando sea necesario mediante una plataforma web sencilla e intuitiva.

De esta manera, el asistente no se limita a un conjunto de preguntas y respuestas, sino que puede adaptarse a diferentes trámites y servicios o necesidades existentes en la institución. Esto resulta relevante porque la información institucional cambia con el paso del tiempo: algunas convocatorias se actualizan, se cierran, algunas fechas para realizar un trámite tienen vigencia y los procesos administrativos pueden variar de un periodo escolar a otro. Dicho lo anterior, contar con una herramienta web moderna, intuitiva y editable permitirá mantener la información organizada y disponible para los usuarios.

Diversos estudios han revisado y dado seguimiento a la evolución y al impacto de los asistentes virtuales en el campo de la educación. Por ejemplo, (Gómez & Ríos, 2022) señalan que estos sistemas

han mejorado con el tiempo y han demostrado ser rentables como apoyo para la atención de los usuarios. Sin embargo, reconocen que todavía existen retos relacionados con lograr conversaciones más naturales, precisas y adaptadas a cada contexto educativo. En su investigación, destacan tres elementos esenciales: la comprensión del lenguaje natural, el manejo del diálogo y generación de respuestas pertinentes. Estos elementos ayudan a comprender y argumentar por qué un asistente virtual no debe limitarse a solo mostrar información, sino también interpretar mejor lo que el usuario pretende consultar.

Por otra parte, (Essel, Vlachopoulos, Tachie-Menson, Johnson, & Baah, 2022) evaluaron el uso de un chatbot como apoyo educativo y encontraron que los estudiantes que interactuaron con este tipo de herramienta obtuvieron mejores resultados que quienes trabajaron únicamente con el instructor. Aunque el contexto del estudio se enfoca más en el aprendizaje, resulta útil para este proyecto porque muestra que un chatbot puede servir como apoyo cuando muchas personas necesitan orientación al mismo tiempo. Esto se relaciona con Build A Chat, ya que en periodos de alta demanda los estudiantes suelen repetir dudas similares sobre trámites, fechas o requisitos.

Con base en lo revisado, se puede identificar que los asistentes virtuales representan una opción viable para apoyar la atención a estudiantes y mejorar la forma en que se comunica la información institucional. En el caso del Instituto Tecnológico de Culiacán, esta propuesta se toma como punto de partida para desarrollar una plataforma que concentre en un solo espacio la información académica y administrativa más consultada. Además, se busca que el contenido no dependa únicamente del desarrollador del sistema, sino que pueda ser actualizado por personal autorizado de la institución cuando existan cambios en fechas, trámites, convocatorias o requisitos.

Con esto, Build A Chat busca responder a una necesidad concreta del entorno escolar: facilitar que los estudiantes encuentren orientación sin depender siempre de una atención presencial o de buscar información en varios medios.

1.2 Descripción del problema

El Instituto Tecnológico de Culiacán cuenta con diversas áreas de atención para los estudiantes, entre ellas servicios escolares, becas, titulación, servicio social, pagos, reinscripciones y posgrado.

Aunque esta distribución permite atender distintos procesos académicos y administrativos, en la práctica también puede generar confusión, especialmente cuando los estudiantes no saben con claridad a qué área acudir o con quién dirigirse para resolver una duda relacionada con algún trámite o servicio.

Esta situación también se presenta en los medios digitales. La información académica y administrativa se encuentra distribuida en diferentes medios, como el sitio web oficial, redes sociales institucionales, plataformas académicas como Ámbar o comunicados que se emiten. Como consecuencia, los estudiantes, principalmente los de nuevo ingreso, enfrentan dificultades para localizar información relacionada con inscripciones, trámites o solicitar un Kardex de calificaciones. Esto puede provocar confusión, pérdida de tiempo, retrasos en trámites e incluso no completar correctamente algún proceso de la institución o participación en convocatorias e incluso que algún procedimiento no se complete correctamente.

Por otra parte, el personal administrativo constantemente recibe consultas, sobre todo en épocas de alta afluencia de estudiantes como al inicio o a finales de semestre, inscripción a un curso de verano, en periodo de carga académica y para concluir el proceso de egreso y titulación de la institución. En muchos casos, los estudiantes tienen muchas dudas que pueden resolverse con información clara centralizada y disponible en todo momento. Sin embargo, al no existir un medio único de consulta, los estudiantes recurren directamente a oficinas, correos electrónicos, mensajes o redes sociales, lo que incrementa la carga de atención del personal y hace que los tiempos de respuesta sean variables.

Durante el desarrollo del proyecto se tomaron en cuenta comentarios y experiencias de estudiantes del ITC que han tenido dificultades al buscar información sobre trámites o procesos escolares.

Como parte de una exploración diagnóstica inicial, se recuperaron comentarios informales de estudiantes y observaciones realizadas durante el desarrollo del proyecto. Aunque esta etapa no tuvo el alcance de un estudio estadístico, permitió identificar situaciones recurrentes relacionadas con la dispersión de información, la falta de claridad en los trámites y la necesidad de acudir a distintas áreas para resolver una misma duda.

Estas observaciones surgieron principalmente a partir de conversaciones informales con compañeros y de situaciones que el equipo logró identificar dentro de la institución. A pesar de que en esta etapa no se aplicó una encuesta formal, no significa que no fuese posible reconocer dificultades que se presentan de manera constante en los trámites en la información que solicitan los interesados. Entre ellas destaca la poca claridad para efectuar exitosamente una solicitud de información o un proceso, la información se encuentra distribuida en diversos medios y la necesidad de acudir a diferentes áreas para poder completar un trámite completo, como lo puede ser el de una titulación (ver **Tabla 1**).

Tabla 1 Situaciones observadas

Situación observada	Ejemplo de duda frecuente	Área relacionada	Consecuencia
Información dispersa	“¿Dónde reviso los requisitos de titulación?”	Titulación	El estudiante acude a varias oficinas
Dudas en reinscripción	“¿Cuándo se abre la carga académica?”	Servicios escolares	Retraso o confusión
Falta de canal único	“No sé si revisar Facebook, Ámbar o la página”	Comunicación institucional	Pérdida de tiempo

Durante estas conversaciones, los estudiantes relataron que en repetidas ocasiones no logran conocer todos los requisitos indispensables para concluir un proceso o que la información que les proporcionan no siempre está completa o está desactualizada. También se pudo identificar que trámites como el proceso de reinscripción, la solicitud de constancias de estudios, la liberación de las residencias profesionales, el servicio social o el proceso de titulación generan dudas, pues no existe una guía clara que indique los pasos a seguir con precisión. Este tipo de situaciones resaltan la importancia de poder utilizar una herramienta capaz de proporcionar información institucional de forma ordenada, accesible y fácil de entender.

Con base en lo anterior, surge Build A Chat, una plataforma web cuyo propósito está orientado a la creación de asistentes virtuales institucionales personalizables con el fin de que estos asistentes funcionen como un punto de partida que permita atender a las consultas de información relacionadas tanto con procesos académicos como administrativos, a través de una interfaz conversacional amigable para el usuario final.

El asistente virtual tendrá la capacidad de brindar respuestas a preguntas frecuentes, proporcionar fechas importantes en las que se deben realizar trámites como titulación o reinscripción, dar a conocer todos los requisitos para el proceso de cargas académicas, información para aspirantes de nuevo ingreso y otros servicios institucionales.

Además, al integrar técnicas de inteligencia artificial generativa, se pretende que las respuestas proporcionadas sean claras y cercanas a la forma en que los estudiantes expresan sus dudas o comentarios. Con ello, Build A Chat busca disminuir la frecuencia con la que se realizan consultas presenciales para dudas básicas, reducir la repetición de información sobre temas recurrentes y ofrecer un espacio de consulta organizado y accesible para la comunidad estudiantil.

Para valorar si Build A Chat contribuye a mejorar esta situación, se propone considerar algunos indicadores, como la cantidad de consultas atendidas por el asistente virtual, el tiempo promedio de respuesta, el número de dudas que todavía requieren apoyo del personal administrativo y la opinión de los usuarios sobre la claridad de la información recibida. Estos elementos permitirían tener una primera aproximación sobre la utilidad del sistema y sobre los aspectos que podrían mejorarse después de su implementación.

1.3 Hipótesis

Aunque este proyecto se centra en el desarrollo de una plataforma web, también es importante revisar si su uso puede ayudar a mejorar la forma en que los estudiantes consultan información académica y administrativa dentro del ITC. En este apartado se plantean las siguientes hipótesis, tomando como referencia aspectos que pueden medirse durante la evaluación del sistema, como el tiempo de respuesta, el número de consultas atendidas por el asistente virtual, las dudas que todavía requieren apoyo del personal administrativo y la opinión de los usuarios sobre la claridad de la información recibida.

1.3.1 Hipótesis Nula (H0)

La implementación de una plataforma web para la creación de un asistente virtual institucional basado en inteligencia artificial no provoca una mejora observable en el acceso a la información institucional ni la atención de consultas sobre trámites de los estudiantes del ITC.

1.3.2 Hipótesis Alternativa (H1)

La implementación de una plataforma web para la creación de un asistente virtual institucional basado en inteligencia artificial mejora el acceso a la información institucional y la atención de consultas sobre trámites de los estudiantes del ITC, al ofrecer respuestas más claras, rápidas y organizadas, además de reducir la cantidad de dudas repetitivas dirigidas al personal administrativo.

Para revisar estas hipótesis, se propone evaluar el funcionamiento del sistema mediante pruebas funcionales y una valoración con usuarios, considerando aspectos como la claridad de las respuestas, el tiempo de atención, la facilidad de uso y la cantidad de consultas que pueden resolverse desde el asistente virtual.

1.4 Objetivos

A partir de la problemática identificada, en primera instancia, se define el objetivo general del proyecto y posteriormente se establecen los objetivos específicos con la finalidad de orientar el desarrollo de la plataforma Build A Chat, buscando definir de manera clara lo que se pretende lograr durante el proyecto.

1.4.1 Objetivo general

Desarrollar una aplicación web con funcionalidad de asistente virtual institucional que apoye el acceso a información académica y administrativa del ITC, mediante una interfaz interactiva que permita atender consultas frecuentes, organizar contenidos y facilitar la comunicación entre estudiantes, docentes y personal administrativo.

1.4.2 Objetivos específicos

- Analizar las necesidades de consulta de información académica y administrativa de los estudiantes del ITC, con el fin de identificar los procesos, servicios y dudas más frecuentes que deberá atender la plataforma.
- Diseñar la arquitectura de una aplicación web modular para la creación, administración y consulta de asistentes virtuales institucionales, considerando la estructura cliente-servidor, el

modelo de datos, los roles de usuario y la organización de información mediante un árbol de conocimiento.

- Desarrollar los módulos principales de la plataforma, incluyendo registro, inicio de sesión, creación de asistentes, edición del árbol de conocimiento, consulta con el asistente virtual y monitoreo desde el panel administrativo.
- Integrar un componente de inteligencia artificial generativa que permita al asistente virtual interpretar consultas en lenguaje natural y generar respuestas relacionadas con la información institucional disponible.
- Implementar un panel administrativo para gestionar la información del asistente virtual, revisar conversaciones y actualizar contenidos de manera organizada, con el propósito de apoyar la atención de consultas académicas y administrativas.
- Validar el funcionamiento de la plataforma mediante pruebas funcionales, revisión de requerimientos y evaluación con usuarios, considerando aspectos como tiempo de respuesta, consultas atendidas, claridad de la información y facilidad de uso.

1.5 Justificación

Este proyecto surge de la necesidad de mejorar la forma en que los estudiantes del Instituto Tecnológico de Culiacán consultan información académica y administrativa. En muchos casos, para resolver una duda sobre trámites, fechas, pagos, becas o procesos escolares, los estudiantes deben revisar diferentes medios, acudir a oficinas, enviar correos o buscar publicaciones en redes sociales. Esto puede provocar confusión, pérdida de tiempo y una mayor dependencia del personal administrativo para resolver dudas que suelen repetirse con frecuencia.

Build A Chat propone una plataforma web que permite crear asistentes virtuales institucionales personalizados. La intención no es sustituir la atención humana, sino ofrecer una primera orientación a los estudiantes y apoyar al personal administrativo en la atención de preguntas frecuentes. De esta manera, el sistema puede ayudar a que los usuarios encuentren información básica sobre trámites, requisitos, convocatorias y otros procesos escolares sin tener que acudir directamente a una oficina para cada consulta.

Una de las ventajas principales de la plataforma es que puede estar disponible en cualquier momento. Esto resulta útil para estudiantes que necesitan consultar información fuera del horario laboral o que no saben exactamente a qué área dirigirse. Además, al incorporar inteligencia artificial generativa, el asistente puede responder en lenguaje natural, lo que permite una interacción más sencilla y parecida a una conversación común.

Desde el punto de vista institucional, el proyecto puede apoyar la organización de la información y mejorar los canales de atención. Al contar con una plataforma administrable, el personal autorizado puede actualizar contenidos, modificar respuestas y organizar la información de acuerdo con las necesidades de cada área. Esto ayuda a evitar que los estudiantes dependan de información incompleta, desactualizada o dispersa en varios medios.

El proyecto también tiene valor técnico, ya que integra diferentes elementos de la ingeniería en sistemas computacionales, como el desarrollo de una aplicación web, gestión de usuarios, almacenamiento de conversaciones, creación de asistentes virtuales, comunicación en tiempo real e inteligencia artificial generativa. Por ende, Build A Chat representa una oportunidad para aplicar conocimientos técnicos en una solución relacionada con una problemática real del entorno educativo.

Finalmente, aunque Build A Chat se plantea inicialmente para el Instituto Tecnológico de Culiacán, su diseño no se limita únicamente a esta institución. Al tratarse de una plataforma que permite organizar información y configurar asistentes virtuales de acuerdo con distintas necesidades, podría adaptarse posteriormente a otros espacios educativos o administrativos que enfrenten problemas similares de comunicación y atención a usuarios.

El proyecto tiene una importancia práctica, ya que busca facilitar el acceso a la información; institucional, porque puede apoyar la organización de procesos y la atención de consultas frecuentes; y tecnológica, al proponer el uso de una herramienta digital basada en asistentes virtuales. En conjunto, la propuesta pretende ofrecer una forma más ordenada, accesible y cercana de conectar a los estudiantes con la información que necesitan para realizar sus trámites o resolver sus dudas.

1.6 Alcance

El alcance de este proyecto contempla el desarrollo de una plataforma web que permita crear, administrar y utilizar asistentes virtuales institucionales. En esta primera etapa, la plataforma estará orientada al Instituto Tecnológico de Culiacán y tendrá como finalidad apoyar la consulta de información académica y administrativa relacionada con trámites, fechas importantes, pagos, becas, titulación, reinscripción y otros procesos que forman parte de la vida estudiantil.

Para esta versión inicial se consideran dos tipos de usuario: el usuario administrativo y el usuario general. El usuario administrativo será quien tenga acceso a la gestión interna de la plataforma. Entre sus funciones se encuentran iniciar sesión, crear y modificar asistentes virtuales, organizar la información mediante un árbol de conocimiento, actualizar los contenidos disponibles y revisar las conversaciones realizadas por los usuarios. Esto permitirá que la información del asistente pueda mantenerse actualizada conforme cambien los procesos o necesidades de la institución. Por su parte, el usuario general podrá consultar la información disponible a través del asistente virtual e interactuar mediante una interfaz conversacional.

El sistema busca reunir en un solo espacio información que actualmente puede encontrarse distribuida en diferentes medios institucionales, como páginas web, redes sociales, comunicados o plataformas académicas. No obstante, la plataforma funcionará como una herramienta de apoyo para la consulta y orientación de los usuarios, sin sustituir los sistemas oficiales existentes ni los procesos administrativos establecidos por la institución.

Esta etapa se limita al diseño, desarrollo y validación funcional de la plataforma Build A Chat. Se consideran módulos como autenticación de usuarios, gestión de asistentes virtuales, árbol de conocimiento, historial de conversaciones, interacción con inteligencia artificial generativa y panel administrativo.

1.7 Contribución Técnica

La principal contribución técnica de este proyecto consiste en el desarrollo de una plataforma web modular para crear y administrar asistentes virtuales conversacionales. A diferencia de un chatbot fijo, Build A Chat se caracteriza por permitir que un usuario administrativo organice y personalice la

información del asistente mediante un árbol de conocimiento, sin necesidad de modificar directamente el código fuente del sistema.

El sistema integra dos interfaces principales. La primera está dirigida al usuario general, quien puede realizar consultas mediante una interfaz similar a la de aplicaciones de mensajería, con el propósito de que la interacción sea familiar y fácil de usar. La segunda está dirigida al usuario administrativo, quien puede gestionar los contenidos del asistente, revisar conversaciones y dar seguimiento a las dudas que requieran atención humana.

Desde el punto de vista tecnológico, el proyecto integra diferentes componentes del desarrollo web, como autenticación de usuarios, arquitectura cliente-servidor, base de datos, comunicación en tiempo real, almacenamiento de mensajes, panel administrativo e inteligencia artificial generativa para apoyar la generación de respuestas en lenguaje natural.

Además, la plataforma aporta una forma más ordenada de administrar información institucional, ya que permite concentrar contenidos en un solo espacio y actualizarlos cuando sea necesario. Build A Chat se diseña como una herramienta tecnológica alternativa que busca disminuir la dispersión de la información, asistir y responder a consultas repetitivas, así como mejorar la forma en cómo se orienta a los usuarios dando seguimiento en procesos académicos o administrativos que soliciten.

1.8 Estructura de la tesis:

Este trabajo se organiza de la siguiente manera: en el capítulo 1 se presenta la introducción de la propuesta, donde se abordan los antecedentes, la problemática identificada, se dan a conocer las hipótesis, se presentan los objetivos, la justificación, el alcance y la contribución técnica de la propuesta. En el capítulo 2 se presentan los fundamentos teóricos que sustentan el proyecto, abordando y definiendo temáticas como aplicaciones web, asistentes virtuales, inteligencia artificial generativa, arquitectura de software y las tecnologías utilizadas para el desarrollo de Build A Chat. Subsecuentemente, en el capítulo 3 se describe el procedimiento seguido para efectuar el análisis, el diseño, el desarrollo y la implementación de la plataforma web. En el capítulo 4 se discuten los resultados obtenidos, incluyendo las pruebas realizadas y evidencias del funcionamiento del sistema.

Finalmente, en el capítulo en el 5 se discuten las conclusiones, las limitaciones y los trabajos futuros de este proyecto, el cómo este proyecto puede seguir una vez terminada esta tesis.

2 FUNDAMENTO TEÓRICO

En este capítulo se abordan los conceptos principales que dan sustento al desarrollo de la plataforma Build A Chat. En primer lugar, se presentan las aplicaciones web, así como su arquitectura básica, debido a que el proyecto se plantea como una solución accesible desde un navegador y organizada mediante componentes de frontend, backend y base de datos.

Posteriormente, se muestra la comunicación en tiempo real mediante WebSocket y Socket.IO, esta funcionalidad permite que la comunicación entre el usuario, el asistente virtual y el administrador sea prácticamente inmediata. Asimismo, se revisan los asistentes virtuales conversacionales y su aplicación en contextos educativos e institucionales, tomando en cuenta su utilidad para atender preguntas frecuentes, orientar a los usuarios y facilitar el acceso a la información. Además, se incluye una revisión general sobre inteligencia artificial e inteligencia artificial generativa, ya que estas tecnologías permiten que el sistema interprete consultas y genere respuestas en lenguaje natural. Finalmente, se aborda la experiencia de usuario en interfaces conversacionales y se presentan trabajos relacionados que ayudan a contextualizar la propuesta dentro de soluciones similares.

2.1 Aplicaciones web

Las aplicaciones web en base a la investigación (Camazón, 2010) son: son mecanismos que utilizan los usuarios para comunicarse y son desarrolladas en diferentes lenguajes de programación; estos son formas de crear aplicaciones software que se ejecutan en un equipo informático. Las aplicaciones intercambian datos en redes de ordenadores. Por ejemplo, en Internet, la forma de intercambiar datos entre aplicaciones se realiza por medio de reglas denominadas protocolos que establecen la comunicación entre clientes y servidores; los clientes son los usuarios que, mediante aplicaciones como los navegadores web, se comunican con otros ordenadores e interactúan con ellos. Además de los protocolos, los estándares también forman parte de los servicios web. Los estándares son especificaciones disponibles públicamente para lograr una tarea específica. Un ejemplo de estándares son los del W3C (World Wide Web Consortium, 2004); este tipo de organizaciones son comités responsables de la arquitectura y reglamentación de los servicios web, y su objetivo principal

es la creación de estándares web y pautas. Desde 1994, el W3C ha publicado más de ciento diez estándares, denominados Recomendaciones del W3C.

Los servicios web comunican diferentes aplicaciones software independientemente de las plataformas sobre las que se instalen, la capacidad para comunicar múltiples sistemas diferentes se realiza por medio de protocolos y estándares abiertos.

En el esquema de funcionamiento básico de un servicio web intervienen:

- Clientes: son quienes hacen las peticiones (navegador web).
- Servidores: programas que se ejecutan continuamente en un ordenador, a la espera de que les lleguen las solicitudes de los clientes y responden mediante una página web (servidor web).
- Aplicaciones de software: mecanismos que utilizan los clientes y los servidores para comunicarse (navegador web).
- Lenguaje de programación: forma de crear las aplicaciones software (C++).
- Red de ordenadores: conjunto de ordenadores y/o dispositivos conectados por medio de cables, señales, ondas o cualquier otro método de transporte de datos que comparten información (Internet).
- Protocolos: conjunto de reglas utilizadas en la comunicación entre los clientes y los servidores HTTP (HyperText Transfer Protocol por sus siglas en inglés), diseñado para transferir páginas HTML (MDN Web Docs, 2026).
- Estándares: conjunto de reglas normalizadas que describen los requisitos que deben ser cumplidos por un producto, proceso o servicio, con el objetivo de establecer un mecanismo base para permitir que distintos elementos hardware o software que lo utilicen sean compatibles entre sí. Uno de los estándares más conocidos es el HTML (HyperText Markup Language), un lenguaje de marcas utilizado para definir la estructura de los documentos; de esta forma todos comparten el mismo lenguaje

para la representación de la información. Al usar el mismo formato para el desarrollo se consigue que cualquier usuario que interprete dicho lenguaje represente el documento de la web de la misma forma.

2.1.1 Arquitectura básica de una aplicación web

La arquitectura de una aplicación web se trata del cómo es que los distintos componentes de la aplicación interactúan para permitir el funcionamiento de una plataforma en internet. La separación por capas permite organizar mejor las responsabilidades del sistema, distinguiendo la interfaz de usuario, la lógica de negocio y el acceso a datos (Fowler, 2015). Desde la ingeniería de software, una arquitectura modular facilita el mantenimiento, la evolución y la comprensión del sistema durante su ciclo de vida (Sommerville, 2016; Pressman & Maxim, 2020). Generalmente, una aplicación web se divide en tres capas principales: Frontend, Backend y la base de datos.

- Frontend: es la parte visual con la que interactúa el usuario desde el navegador. Su función principal es mostrar la información y permitir la interacción mediante botones, formularios, menús y otras interfaces gráficas. Las tecnologías más utilizadas en esta capa son: HTML: estructura de la página, CSS: diseño y estilos visuales. JavaScript: interactividad y comportamiento dinámico.
- El backend se encarga de la lógica de negocio y del procesamiento de datos. Recibe las solicitudes del frontend, procesa la información y devuelve respuestas adecuadas.
Entre sus funciones principales están:
 - Validar usuarios.
 - Procesar formularios.
 - Gestionar autenticación.
 - Conectar con la base de datos.
 - Ejecutar reglas de negocio.
- La base de datos almacena toda la información necesaria para el funcionamiento de la aplicación, como usuarios, mensajes, productos o configuraciones.

Existen dos tipos principales:

Bases de datos relacionales Organizan la información en tablas relacionadas entre sí.

Ejemplos:

- MySQL
- PostgreSQL

Bases de datos no relacionales: Almacenan información de manera flexible mediante documentos o colecciones. Ejemplo:

- MongoDB

2.1.2 Comunicación en tiempo real con WebSocket/Socket.IO

De acuerdo con su documentación oficial, Socket.IO es una biblioteca de JavaScript que permite la comunicación en tiempo real y bidireccional entre cliente y servidor (Socket.IO, 2026). Se construye sobre el protocolo WebSocket, proporcionando una API sencilla para manejar eventos y datos en tiempo real. Socket.IO A diferencia de las solicitudes HTTP tradicionales, que son unidireccionales, Socket.IO mantiene una conexión persistente, lo que permite al servidor enviar datos al cliente en cualquier momento (Fette & Melnikov, 2011).

La integración de Socket.IO en chatbot permite:

- Comunicación en tiempo real: Los mensajes se transmiten instantáneamente entre el usuario y el chatbot.
- Interactividad mejorada: Permite respuestas inmediatas, mejorando la experiencia del usuario.
- Escalabilidad: Facilita la gestión de múltiples conexiones simultáneas. Por ejemplo, al combinar Socket.IO con frameworks como Express en Node.js, se puede crear un chatbot que responda en tiempo real a las consultas de los usuarios.

Socket.IO también se puede integrar con Python utilizando bibliotecas como `pythonsocketio` y frameworks como Flask. `python-socketio.readthedocs.io` Esto permite a los desarrolladores aprovechar las capacidades de Python en procesamiento de lenguaje natural y aprendizaje automático para crear chatbots más avanzados.

Chatbots de atención al cliente: Proporcionan respuestas instantáneas a consultas comunes.

Asistentes virtuales: Ofrecen asistencia en tiempo real en aplicaciones web. Aplicaciones educativas: Facilitan el aprendizaje interactivo mediante conversaciones en tiempo real. Un ejemplo práctico es la creación de un chatbot que utiliza Socket.IO para comunicarse con el cliente y una API de procesamiento de lenguaje natural para generar respuestas contextuales.

Socket.IO al tener un funcionamiento que se basa en un sistema de eventos tanto cliente como servidor pueden interactuar y escuchar eventos personalizados, en el caso de los chatbots el cliente es el que estaría emitiendo un evento para que sea escuchado por el servidor y lo pueda procesar para dar como resultado la respuesta a este evento ya sea un mensaje en lenguaje natural o alguna frase predefinida que usa el sistema en caso de no contar con esa consulta en específico, esto nos da ventajas como lo es la transmisión inmediata de los mensajes reduciendo todo lo posible la latencia así como también contar con actualizaciones dinámicas dando capacidad de su uso sin tener que recurrir a recargar la página y una sincronización en tiempo siendo útil en nuestro contexto de ser una plataforma de información en ámbitos educativos o laborales dando soporte al usuario en el momento.

La conexión mediante WebSocket y Socket.IO representa una de las tecnologías fundamentales en el desarrollo de aplicaciones web modernas que requieren interacción continua entre usuarios y sistemas inteligentes. A diferencia de los modelos tradicionales que tienen como base eventos y respuestas independientes, esta tecnología mantiene una conexión activa durante toda la sesión, permitiendo el intercambio constante de información sin interrupciones. En el contexto de los asistentes virtuales, la comunicación en tiempo real es una característica importante, ya que logra que las respuestas del chatbot se entreguen de manera inmediata y que la interacción se perciba como una conversación más natural entre el usuario y el sistema. El protocolo WebSocket fue diseñado precisamente para permitir una comunicación bidireccional entre el navegador y el servidor, evitando que el cliente tenga que estar solicitando información de manera repetida, como ocurre en otros esquemas tradicionales de comunicación web (Fette & Melnikov, 2011).

Socket.IO facilita el desarrollo de aplicaciones que requieren comunicación rápida entre cliente y servidor, ya que trabaja con un modelo bidireccional y basado en eventos. De acuerdo con su

documentación oficial, esta biblioteca permite establecer una comunicación de baja latencia, donde tanto el servidor como el cliente pueden enviar y recibir información cuando ocurre un evento determinado (Socket.IO, 2026). Esto resulta útil en sistemas conversacionales, porque permite manejar mensajes, notificaciones, cambios de estado, indicadores de escritura y otros elementos sin necesidad de recargar la página.

Otra ventaja de este tipo de comunicación es que mejora la experiencia de usuario, ya que el sistema puede responder de forma más fluida y mantener actualizada la interfaz mientras la conversación avanza. En aplicaciones donde el tiempo de respuesta influye en la percepción del usuario, la rapidez del sistema es un aspecto importante, pues las demoras pueden afectar la continuidad de la interacción y generar una sensación de lentitud o poca eficiencia (Nielsen, 1993). Por ello, en un asistente virtual, recibir una respuesta en el momento adecuado puede contribuir a que el usuario perciba el sistema como más útil y confiable.

Además, la arquitectura basada en eventos facilita la integración con otros servicios, como bases de datos, APIs externas o modelos de inteligencia artificial. En el caso de Build A Chat, esta característica permite que el chatbot procese la consulta del usuario, recupere información relacionada con el contexto de la conversación y entregue una respuesta dentro de la misma interfaz. En aplicaciones educativas o empresariales, el uso de tecnologías como WebSocket y Socket.IO puede apoyar la atención inmediata de estudiantes, empleados o clientes, al mantener una comunicación continua y reducir los tiempos de espera durante la interacción. De esta manera, la tecnología no solo optimiza la interacción entre usuario y sistema, sino que también contribuye a la escalabilidad, confiabilidad y eficiencia operativa de las plataformas que dependen de la comunicación en tiempo real.

2.2 Asistentes virtuales conversacionales

Los asistentes virtuales (VA) citando a (Arredondo, TAIPE, & Cruz, 2024) se están volviendo cada vez más comunes, siendo empleados como interfaces principales en dispositivos inteligentes para el hogar, como televisores, altavoces y teléfonos inteligentes, así como en sistemas de entretenimiento en automóviles o robots de servicio. En 2020, se utilizaban alrededor de 4.2 mil millones de VA a nivel mundial, y se proyecta que este número alcanzará los 8.4 mil millones para el 2024. Además, se prevé

que el mercado global de asistentes virtuales alcance los 11 mil millones de dólares para 2025. Esta tendencia refleja el aumento de la eficiencia y la capacidad para manejar interacciones más complejas. Investigaciones similares respaldan esta observación, ya que discuten cómo los avances en PLN afectan la precisión de estos sistemas. Un punto clave es cómo la automatización mediante chatbots y asistentes virtuales puede reducir costos y aumentar la eficiencia en las organizaciones, abordando tareas repetitivas y mejorando la atención al cliente mediante la automatización de procesos. Los autores explican que la IA aplicada a estos sistemas reduce los costos de servicio en un 30% y agiliza la gestión de consultas a gran escala, contribuyendo a una mejor experiencia del cliente y menores gastos operativos (Rouhiainen, 2018). El uso de asistentes virtuales ha cambiado la forma en que las personas interactúan con la tecnología, tanto a nivel personal como comercial. Amazon anunció en 2020 que su asistente virtual, Alexa, estaba conectado a más de 100 millones de dispositivos. Google Assistant, que está disponible en más de 90 países y está disponible en treinta idiomas, maneja millones de interacciones diariamente, lo cual destaca la creciente dependencia de estos asistentes para realizar múltiples tareas diarias. Los asistentes virtuales basados en inteligencia artificial, particularmente aquellos que utilizan técnicas de procesamiento del lenguaje natural, han mejorado la interacción entre empresas y clientes. Gracias a estos avances, es posible brindar respuestas más rápidas y precisas, lo que ha reducido significativamente la carga administrativa en diversas organizaciones. Los avances en PLN y aprendizaje automático muestran como los asistentes virtuales son capaces de adaptarse a los requerimientos de los usuarios.

La evolución de los asistentes desde el chatbot ELIZA, que se limitaba a responder preguntas predeterminadas utilizando patrones de coincidencia de palabras básicos el cual fue el punto de partida para el desarrollo de asistentes virtuales hasta el lanzamiento de ChatGPT por OpenAI el cual transformo significativamente la forma en que se interactúa con la inteligencia artificial. Esta herramienta permite mantener conversaciones fluidas en lenguaje natural, procesando grandes volúmenes de información y generando respuestas coherentes

Los avances tecnológicos han permitido una mejora significativa en la precisión y la capacidad de respuesta de estos sistemas. ChatGPT puede administrar interacciones contextuales complejas y

producir texto coherente comparable al humano, mientras que ELIZA controlaba las conversaciones limitadas a scripts predeterminados. El objetivo principal de este artículo es realizar una revisión cronológica y sistemática de la evolución de los asistentes virtuales, también se mencionan los avances tecnológicos que han impulsado el desarrollo de estos asistentes virtuales. El aprendizaje automático y las redes neuronales profundas, entre otras tecnologías emergentes, han sido cruciales para mejorar la capacidad de los asistentes para manejar interacciones complejas. En particular, se busca dar a conocer el origen de los primeros asistentes virtuales y como evolucionaron hasta llegar a ser asistentes conversacionales capaces de entablar conversaciones naturales y contextuales con los usuarios (ver Figura 1).

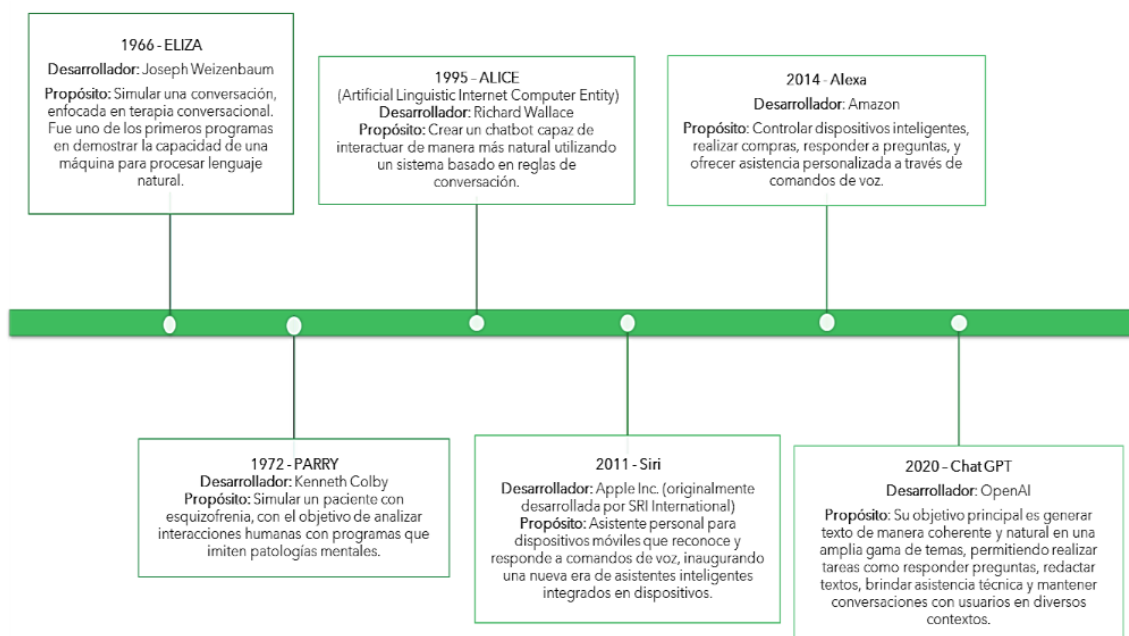


Figura 1 - Línea de tiempo de asistentes virtuales

De acuerdo con su funcionamiento, los chatbots pueden clasificarse en sistemas basados en reglas, sistemas basados en recuperación de información y asistentes apoyados en modelos generativos. Los primeros dependen de flujos definidos previamente; los segundos buscan respuestas en una base de conocimiento; y los terceros generan respuestas en lenguaje natural a partir del contexto proporcionado. Build A Chat retoma elementos de estas categorías, ya que organiza la información

mediante un árbol de conocimiento y utiliza IA generativa como apoyo para formular respuestas más comprensibles.

2.2.1 Chatbot en contextos educativos e institucionales

En el ámbito educativo, los chatbots han comenzado a utilizarse como una herramienta de apoyo para atender dudas frecuentes, orientar a los estudiantes y facilitar el acceso a información institucional de manera más rápida. Estos sistemas permiten ofrecer una primera respuesta al usuario sin que necesariamente tenga que acudir de forma presencial o esperar la disponibilidad inmediata del personal administrativo. De acuerdo con Okonkwo y Ade-Ibijola (2021), los chatbots en educación se han empleado en diferentes contextos como apoyo para la atención, la orientación y el acompañamiento de los estudiantes. Por otra parte, Essel et al. (2022) muestran que un asistente virtual puede apoyar la interacción con estudiantes al ofrecer respuestas automáticas a preguntas frecuentes dentro de un contexto universitario.

En contextos universitarios, los chatbots de preguntas frecuentes pueden apoyar la atención administrativa al responder consultas repetitivas y reducir parte de la carga de trabajo del personal (Lee et al., 2019). Los chatbots generativos en educación superior pueden apoyar procesos estudiantiles y administrativos, aunque requieren lineamientos claros para evitar usos inadecuados o respuestas poco confiables (Chukwuere, 2024).

A diferencia de un sistema de tutoría, un chatbot institucional se enfoca más en orientar sobre procesos administrativos y de gestión escolar. Este tipo de asistente ayuda a resolver dudas relacionadas con trámites, convocatorias, reinscripciones, titulación o medios oficiales de contacto. Su función no es sustituir al personal de la institución, sino servir como un primer punto de apoyo para atender preguntas comunes y, cuando sea necesario, canalizar al usuario con el área correspondiente.

Este tipo de herramienta son muy valiosas porque muchas de las preguntas que realizan los estudiantes se repiten constantemente, sobre todo en periodos de mayor actividad, como el inicio o cierre de semestre, las reinscripciones y los procesos de titulación. Por ello, los chatbots educativos e institucionales pueden ofrecer atención inmediata y disponibilidad continua, lo cual ayuda a mejorar la experiencia del estudiante y a reducir parte de la carga de consultas que normalmente reciben las áreas

administrativas. Okonkwo y Ade-Ibijola (2021) señalan que los chatbots en educación se han utilizado para brindar apoyo, orientación y servicios personalizados tanto a estudiantes como a personal institucional. De manera similar, Chukwuere (2024) destaca que los chatbots generativos en educación superior pueden contribuir al apoyo estudiantil y a la eficiencia administrativa, aunque también requieren lineamientos claros para su uso responsable.

De esta manera, la implementación de un chatbot institucional puede aportar beneficios tanto para los estudiantes como para la institución. Por un lado, los estudiantes cuentan con una herramienta disponible para resolver dudas básicas de forma rápida; por otro, el personal administrativo puede concentrarse en casos más específicos o complejos que requieren atención humana. Sin embargo, para que el chatbot sea realmente útil, es importante que la información proporcionada esté actualizada, sea clara y se mantenga alineada con los canales oficiales de la institución.

2.3 Inteligencia Artificial

En esta sección se presentan conceptos generales sobre inteligencia artificial, aprendizaje automático, aprendizaje profundo y procesamiento del lenguaje natural, con el propósito de explicar su relación con el desarrollo de asistentes virtuales conversacionales.

La inteligencia artificial, es una herramienta que nos puede brindar la ayuda necesaria para poder generar diversos recursos como lo podría ser imágenes, texto, video o código, siendo aplicada esta herramienta en diferentes sectores como lo es educación, medicina, comercio electrónico. En referencia a (Rouhiainen, 2018) la IA es la capacidad de las máquinas para usar algoritmos, aprender de los datos y utilizar lo aprendido en la toma de decisiones tal y como lo haría un ser humano. Sin embargo, a diferencia de las personas, los dispositivos basados en IA no necesitan descansar y pueden analizar grandes volúmenes de información a la vez. Asimismo, la proporción de errores es significativamente menor en las máquinas que realizan las mismas tareas que sus contrapartes humanas.

La idea de que los ordenadores o los programas informáticos puedan tanto aprender como tomar decisiones es particularmente importante y algo sobre lo que deberíamos ser conscientes, ya que

sus procesos están creciendo exponencialmente con el tiempo. Debido a estas dos capacidades, los sistemas de inteligencia artificial pueden realizar ahora muchas de las tareas que antes estaban reservadas sólo a los humanos.

La inteligencia artificial estudia el diseño de agentes capaces de percibir su entorno y actuar de manera orientada al logro de objetivos (Russell & Norvig, 2020).

Las tecnologías basadas en la IA ya están siendo utilizadas para ayudar a los humanos a beneficiarse de mejoras significativas y disfrutar de una mayor eficiencia en casi todos los ámbitos de la vida. Pero el gran crecimiento de la IA también nos obliga a estar atentos para prevenir y analizar las posibles desventajas directas o indirectas que pueda generar la proliferación de la IA.

La IA se puede aplicar en casi todas las situaciones. Éstas son sólo algunas de las aplicaciones técnicas de la IA que están creciendo rápidamente en la actualidad:

- Reconocimiento de imágenes estáticas, clasificación y etiquetado: estas herramientas son útiles para una amplia gama de industrias.
- Mejoras del desempeño de la estrategia algorítmica comercial: ya ha sido implementada de diversas maneras en el sector financiero.
- Procesamiento eficiente y escalable de datos de pacientes: esto ayudará a que la atención médica sea más efectiva y eficiente.
- Mantenimiento predictivo: otra herramienta ampliamente aplicable en diferentes sectores industriales.
- Detección y clasificación de objetos: puede verse en la industria de vehículos autónomos, aunque también tiene potencial para muchos otros campos.
- Distribución de contenido en las redes sociales: se trata principalmente de una herramienta de marketing utilizada en las redes sociales, pero también puede usarse para crear conciencia entre las organizaciones sin ánimo de lucro o para difundir información rápidamente como servicio público.

- Protección contra amenazas de seguridad cibernética: es una herramienta importante para los bancos y los sistemas que envían y reciben pagos en línea

La IA también será capaz de ofrecernos sugerencias y predicciones relacionadas con asuntos importantes de nuestra vida, lo que tendrá su impacto en áreas como la salud, el bienestar, la educación, el trabajo y las relaciones interpersonales. De la misma manera, cambiará la forma de hacer negocios al proporcionar ventajas competitivas a las empresas que busquen entender y aplicar estas herramientas de forma rápida y eficaz. Otro beneficio de la IA es que permitirá que las máquinas y los robots realicen tareas que los humanos consideran difíciles, aburridas o peligrosas, lo que repercutirá a su vez en que el ser humano pueda realizar aquello que antes creía imposible.

La inteligencia artificial estudia el diseño de agentes capaces de percibir su entorno y actuar de manera orientada al logro de objetivos (Russell & Norvig, 2020).

Algunas veces, el término inteligencia artificial tiende a incomodar a la gente, así que se han propuesto algunas alternativas. Un importante experto en IA, Sebastian Thrun, cree que es mejor llamarla «ciencia de datos», una expresión menos intimidatoria que probablemente llevaría a una mayor aceptación entre el público.

2.3.1 Aprendizaje automático

El aprendizaje automático (en inglés, Machine Learning) es uno de los enfoques principales de la inteligencia artificial. En pocas palabras, se trata un aspecto de la informática en el que los ordenadores o las máquinas tienen la capacidad de aprender sin estar programados para ello. Un resultado típico serían las sugerencias o predicciones en una situación particular.

Los primeros ordenadores personales, que estuvieron disponibles para los consumidores a partir de la década de 1980, fueron programados explícitamente para realizar ciertas acciones. Por el contrario, gracias al aprendizaje automático, muchos de los dispositivos que verás en el futuro obtendrán experiencia y conocimientos a partir de la forma en que son utilizados para poder ofrecer una experiencia al usuario personalizada. Ejemplos de ello en la actualidad son la personalización de los sitios de medios sociales como Facebook o los resultados del motor de búsqueda de Google.

El aprendizaje automático usa algoritmos para aprender de los patrones de datos. Por ejemplo, los filtros de spam de correo electrónico utilizan este tipo de aprendizaje con el fin de detectar qué mensajes son correo basura y separarlos de aquellos que no lo son. Éste es un sencillo ejemplo de cómo los algoritmos pueden usarse para aprender patrones y utilizar el conocimiento adquirido para tomar decisiones. El aprendizaje profundo se basa en redes neuronales con múltiples capas que permiten aprender representaciones complejas a partir de grandes volúmenes de datos (Goodfellow et al., 2016).

La Figura 2 muestra a continuación tres subconjuntos del aprendizaje automático que pueden utilizarse: aprendizaje supervisado, no supervisado y de refuerzo.



Figura 2 - Tipos de aprendizaje automático

A continuación, intentaremos explicar la diferencia entre estas tres clases de aprendizaje automático. Imaginemos que hay que organizar 10.000 fotografías y los algoritmos tienen que identificar las fotos en las que aparece un gato.

En el aprendizaje supervisado, los algoritmos usan datos que ya han sido etiquetados u organizados previamente para indicar cómo tendría que ser categorizada la nueva información. Con

este método, se requiere la intervención humana para proporcionar retroalimentación. Volviendo a nuestro ejemplo, enseñaríamos previamente al algoritmo fotos donde apareciera un gato para que luego pudiera identificar imágenes similares.

En el aprendizaje no supervisado, los algoritmos no usan ningún dato etiquetado u organizado previamente para indicar cómo tendría que ser categorizada la nueva información, sino que tienen que encontrar la manera de clasificarlas ellos mismos. Por tanto, este método no requiere la intervención humana. En el ejemplo, los algoritmos tendrían que clasificar ellos mismos todas las fotos en las que apareciera un gato en una categoría.

Por último, con el aprendizaje por refuerzo, los algoritmos aprenden de la experiencia. En otras palabras, tenemos que darles «un refuerzo positivo» cada vez que aciertan. La forma en que estos algoritmos aprenden se puede comparar con la de los perros cuando les damos recompensas al aprender a sentarse, por ejemplo.

2.3.2 Aprendizaje profundo

Una de las aplicaciones más poderosas y de mayor crecimiento de la inteligencia artificial es el aprendizaje profundo (en inglés, Deep Learning). Se trata de un subcampo del aprendizaje automático que se utiliza para resolver problemas muy complejos y que normalmente implican grandes cantidades de datos.

El aprendizaje profundo se produce mediante el uso de redes neuronales, que se organizan en capas para reconocer relaciones y patrones complejos en los datos. Su aplicación requiere un enorme conjunto de información y una potente capacidad de procesamiento. Actualmente, se utiliza en el reconocimiento de voz, el procesamiento del lenguaje natural, la visión artificial y la identificación de vehículos en los sistemas de asistencia al conductor.

Podemos ver un ejemplo claro en las traducciones realizadas en Facebook, que reveló recientemente que, gracias al aprendizaje profundo, había podido efectuar alrededor de 4.500 millones de traducciones diarias. Éstas tienden a ser segmentos de textos cortos, como las actualizaciones de

estado publicadas por los usuarios en sus perfiles. Sin el aprendizaje profundo, sería muy costoso y se requeriría un gran equipo de personas para poder ofrecer el mismo servicio.

En favor de la simplicidad se utiliza principalmente el término inteligencia artificial, a pesar de que muchas veces pueda referirse técnicamente al aprendizaje profundo o al aprendizaje automático.

2.3.3 Procesamiento del lenguaje natural en asistentes conversacionales

Según (Torres & Manjarrés-Betancur, 2020) Los chatbots operan según un conjunto de reglas predefinidas, lo que les permite proporcionar respuestas consistentes y predecibles. Son ideales para tareas como la atención al cliente, la recopilación de datos y la asistencia 24/7, ofreciendo beneficios significativos en términos de costo y simplicidad de implementación. Además, estos asistentes virtuales son altamente flexibles en integraciones con otros sistemas de gestión de datos o atención al cliente, lo que los hace particularmente útiles en entornos corporativos y comerciales. Al seguir reglas estrictas y bien definidas, los asistentes virtuales basados en reglas pueden manejar eficientemente un alto volumen de interacciones, reduciendo la carga de trabajo de los agentes humanos y mejorando la eficiencia operativa. Sin embargo, su capacidad está limitada a los escenarios para los cuales fueron programados. No pueden manejar preguntas fuera de sus reglas predefinidas y carecen de la capacidad para adaptarse a interacciones más complejas o contextuales. Además, los asistentes virtuales son adecuados solo para tareas simples y directas, ya que las interacciones complejas o de múltiples pasos suelen ser difíciles de manejar. Esta simplicidad en su operación los hace útiles para funciones básicas, pero ineficaces en escenarios que requieren una mayor sofisticación en el procesamiento de las interacciones.

2.4 Inteligencia artificial generativa

La inteligencia artificial generativa se puede entender como una tecnología capaz de crear contenido nuevo, como textos, imágenes, fragmentos de código u otros materiales, a partir de la información y los patrones que ha aprendido previamente de grandes cantidades de datos. Dentro de este tipo de inteligencia artificial se encuentran los Modelos de Lenguaje de Gran Escala, conocidos como LLM, los cuales están diseñados para comprender y generar texto en lenguaje natural de una forma coherente. Este tipo de modelos forma parte de una nueva generación de sistemas de inteligencia

artificial que pueden adaptarse a diferentes tareas, como responder preguntas, redactar textos, resumir información o apoyar procesos de búsqueda y orientación (Bommasani et al., 2021).

Gran parte del avance de estos modelos se relaciona con la arquitectura Transformer, propuesta por Vaswani et al. (2017), la cual permitió mejorar el procesamiento del lenguaje mediante mecanismos de atención. Estos mecanismos ayudan al modelo a identificar relaciones entre diferentes partes de una secuencia de texto, lo que facilita generar respuestas más coherentes y relacionadas con el contexto. En términos generales, un LLM no responde a partir de una comprensión humana del lenguaje, sino que predice la continuación más probable de una secuencia con base en los patrones aprendidos durante su entrenamiento.

Estos modelos funcionan a partir de unidades llamadas tokens, que pueden representar una palabra completa, una parte de una palabra o incluso un símbolo. A partir de estos tokens, el modelo interpreta la consulta del usuario y va construyendo una respuesta de acuerdo con el contexto que tiene disponible. No obstante, la calidad de la respuesta no depende únicamente del modelo, sino también de la información que se le proporciona al momento de realizar la consulta. Por lo tanto, el contexto es un elemento importante, ya que ayuda a orientar la respuesta hacia el tema, la intención y las restricciones que el sistema debe respetar.

El contexto que recibe un LLM puede incluir el historial reciente de la conversación, instrucciones específicas y, en algunos casos, información adicional que el sistema le proporciona como referencia. A estas instrucciones se les denomina “prompt”, y funciona como una guía para indicar al modelo qué rol debe asumir, qué tipo de respuesta se espera, qué restricciones puede tener y cuáles son los límites que debe respetar. También existen configuraciones que influyen en el comportamiento del modelo, como la temperatura, la cual permite controlar si las respuestas serán más creativas o precisas y constantes.

En términos generales, cuando se utiliza una temperatura alta, el modelo tiende a generar respuestas más variadas y creativas, aunque también puede volverse menos preciso. En cambio, una temperatura baja favorece respuestas más constantes y controladas, lo cual resulta conveniente cuando se necesita mantener claridad, coherencia y apego a la información disponible.

Los modelos fundacionales se entrenan con grandes volúmenes de datos y pueden adaptarse a diversas tareas, aunque también heredan riesgos relacionados con sesgos, errores y falta de control (Bommasani et al., 2021).

Aunque la inteligencia artificial generativa tiene sus ventajas, también tiene sus riesgos que deben considerarse antes de integrarla en un sistema institucional. Entre ellos destaca que el modelo puede generar respuestas que parecen correctas, pero que en realidad contienen información imprecisa, incompleta o no verificable. Por ello, es necesario definir límites de uso, validar la información que se proporciona y definir qué acciones debe tomar el sistema cuando no cuenta con una respuesta segura.

Uno de los principales es la alucinación, que ocurre cuando el modelo genera una respuesta que parece correcta, pero en realidad es falsa, imprecisa o no puede comprobarse. De acuerdo con Ji et al. (2023), este problema es relevante en los sistemas de generación de lenguaje natural, ya que las respuestas pueden ser fluidas y convincentes, aun cuando no estén sustentadas en información real. También pueden presentarse sesgos, debido a que los modelos pueden reproducir patrones no deseados presentes en los datos con los que fueron entrenados (Bender et al., 2021).

El uso de modelos de lenguaje grandes requiere considerar riesgos asociados con sesgos, datos de entrenamiento y generación de contenido aparentemente fluido, pero no necesariamente confiable (Bender et al., 2021).

Además, en contextos institucionales existe el riesgo de que se maneje información sensible o de que el usuario interprete una respuesta generada por la IA como si fuera una respuesta oficial, aunque no haya sido validada por una persona. Con base en lo anterior, cuando se utiliza inteligencia artificial generativa en un sistema institucional, es importante establecer límites claros, validar la información que se proporciona y definir mecanismos para canalizar los casos que no puedan resolverse automáticamente. Una alternativa para reducir estos riesgos consiste en apoyar las respuestas del modelo con información previamente revisada o recuperada desde fuentes confiables, como ocurre en los enfoques de generación aumentada por recuperación, conocidos como RAG (Lewis et al., 2020).

En el caso del proyecto Build A Chat, la IA generativa se utiliza como un apoyo para interpretar preguntas escritas en lenguaje natural y generar respuestas más comprensibles para el usuario. Sin embargo, su funcionamiento se mantiene limitado al dominio institucional, con el propósito de evitar respuestas fuera de contexto o información no autorizada. Una parte importante del proyecto consiste en que las respuestas del asistente se apoyen en el árbol de conocimiento configurado por el administrador. Esto respalda que la inteligencia artificial no responda de manera completamente abierta, sino que trabaje con información previamente revisada, organizada y actualizada por la institución.

Cuando la pregunta del usuario no se encuentra dentro del contenido disponible o no es lo suficientemente clara, el sistema puede responder de forma controlada. Por ejemplo, puede pedirle al usuario que sea más explícito con su inquietud, puede mostrarle diversas opciones de respuesta, ya sea a través de texto, menús, imágenes, documentos o bien, canalizarlo con un administrador. Con esta estrategia se busca reducir el riesgo de respuestas incorrectas y mantener la confianza en el asistente, aprovechando las ventajas de la inteligencia artificial generativa sin perder el control sobre la información institucional que se ofrece.

En el desarrollo de Build A Chat se decidió no permitir que el modelo generativo respondiera de manera completamente abierta. La respuesta se apoya primero en el contenido configurado por el administrador dentro del árbol de conocimiento. Esta decisión se tomó porque el sistema maneja información institucional, y una respuesta incorrecta sobre fechas, pagos o requisitos podría afectar directamente al estudiante.

2.5 Experiencia de usuario en interfaces conversacionales

La experiencia de usuario, también conocida como UX, en las interfaces conversacionales se relaciona con la forma en que una persona interactúa con un sistema mediante mensajes. En este tipo de interfaces, el usuario no siempre tiene a la vista todos los botones, formularios o menús disponibles, como ocurre en una página web tradicional. Por ello, el sistema debe guiar la conversación de manera clara, sencilla y ordenada.

La usabilidad puede entenderse como el grado en que un sistema permite a usuarios específicos alcanzar objetivos concretos con efectividad, eficiencia y satisfacción en un contexto de uso

determinado (International Organization for Standardization, 2018). La evaluación de usabilidad permite identificar problemas de interacción que pueden afectar la comprensión, eficiencia y satisfacción de los usuarios durante el uso de un sistema (Nielsen, 1993).

En una interfaz conversacional es importante que el usuario sepa qué puede preguntar, cómo continuar y qué hacer cuando el sistema no comprende su solicitud. Para lograrlo, se pueden utilizar elementos de apoyo como botones, sugerencias, mensajes de confirmación o menús de opciones. Estos recursos ayudan a que la interacción sea más fácil de seguir y reducen la posibilidad de que el usuario se sienta perdido durante la conversación.

El diseño de interfaces debe procurar que las acciones disponibles sean comprensibles para el usuario y que el sistema ofrezca retroalimentación clara durante la interacción (Norman, 2013).

En el caso de un asistente institucional, la experiencia de usuario cobra mayor importancia porque los estudiantes suelen buscar respuestas rápidas y concretas sobre trámites, fechas, requisitos o servicios. Si el sistema responde con lenguaje confuso, no entiende la pregunta o no ofrece una forma clara de continuar, el usuario puede perder confianza en la herramienta (ver Tabla 2).

Tabla 2 - Criterios de experiencia de usuario

Criterio UX	Aplicación en Build A Chat
Claridad de respuestas	Respuestas breves, comprensibles y relacionadas con el trámite
Facilidad para iniciar consulta	Interfaz similar a mensajería
Recuperación ante errores	Menús, reformulación o canalización
Canalización humana	Contacto con administrador cuando la consulta no se resuelve

Basado en ello, el diseño del asistente debe procurar que la conversación sea fluida, que existan alternativas cuando ocurre un error y que, si la información no está disponible, el usuario pueda ser orientado hacia otra opción, como consultar un menú, reformular su pregunta o recibir apoyo de una persona encargada.

Claridad de respuestas: Build A chat debe de dar respuestas claras sin que te de algo ambiguo, que te de algo más concreto y directo para ayudarte con tu duda.

Facilidad para iniciar una consulta: Build A chat te permite preguntarle cualquier duda que tengas de manera inmediata solo necesitas escribirlo y en el momento que lo hagas habrás iniciado una consulta con el programa.

Recuperación cuando no entiende: Gracias al api de Gemini 2 al momento de que tengas algún error gramático u ortográfico este si puede entenderlo, pero si le preguntas algo que no tenga en su base de datos te podrá redirigir a algún dato con contacto humano o darte lo más parecido para que puedas verlo en ese momento

Canalización con administrador: El administrador no solo puede crear en cualquier momento un nodo para el árbol de conocimiento del Bot, sino que también este puede interactuar directamente con el usuario que solicito su ayuda

Similitud con aplicaciones de mensajería: al querer darle un ambiente familiar se tomó la decisión de darle un aspecto familiar al Bot darle uno parecido al de Messenger o WhatsApp dando un entorno que todos usamos día a día para lograr un fácil uso de la aplicación.

2.6 Trabajos relacionados

El Instituto Tecnológico de Culiacán es una institución de educación superior que posee una extensa infraestructura repartida en diferentes partes del campus. Aunque esta distribución espacial permite ofrecer una variada gama de servicios académicos, administrativos y de apoyo, también presenta algunas complicaciones para los estudiantes, sobre todo para aquellos que recién ingresan, al tratar de orientarse dentro de las instalaciones o identificar rápidamente las oficinas que satisfacen sus necesidades.

La separación física de los edificios administrativos, departamentos académicos, servicios para estudiantes y otras áreas funcionales puede causar confusión, pérdida de tiempo y una experiencia decepcionante para quienes no conocen bien la estructura del campus. En vista de esta situación, se considera relevante implementar soluciones tecnológicas basadas en inteligencia artificial, como los chatbot conversacionales, que pueden ser integrados en plataformas tanto web como móviles para ofrecer asistencia inmediata, personalizada y accesible en todo momento.

En esta sección, se llevó a cabo una investigación en la cual se exploran trabajos relacionados a la Inteligencia artificial y los chatbot, con el objetivo de adquirir un conocimiento sólido y actualizado sobre los aspectos teóricos relevantes, así como técnicas utilizadas en dichas investigaciones, revisando distintos artículos científicos relacionados.

2.6.1 Asistentes virtuales en la educación

El trabajo realizado por (Essel, Vlachopoulos, Tachie Menson, Johnson, & Baah, 2022) nos comenta que este se centró en el aprendizaje del grupo experimental y en su percepción sobre la interacción con el chatbot. Esta investigación contribuye a la creciente literatura sobre chatbot basados en inteligencia artificial (IA) para mejorar el rendimiento académico de los estudiantes.

Este fue uno de los primeros estudios en Ghana en integrar un chatbot para interactuar con estudiantes universitarios. Asimismo, el estudio proporciona información importante sobre el uso y desarrollo de asistentes virtuales como por ejemplo KNUSTbot un asistente virtual que fue desarrollado en WhatsApp (ver Figura 3) usando las herramientas de meta este tipo de chatbot fueron una las inspiraciones que ayudo a formar el concepto de hacer que el proyecto fuese de fácil entendimiento.

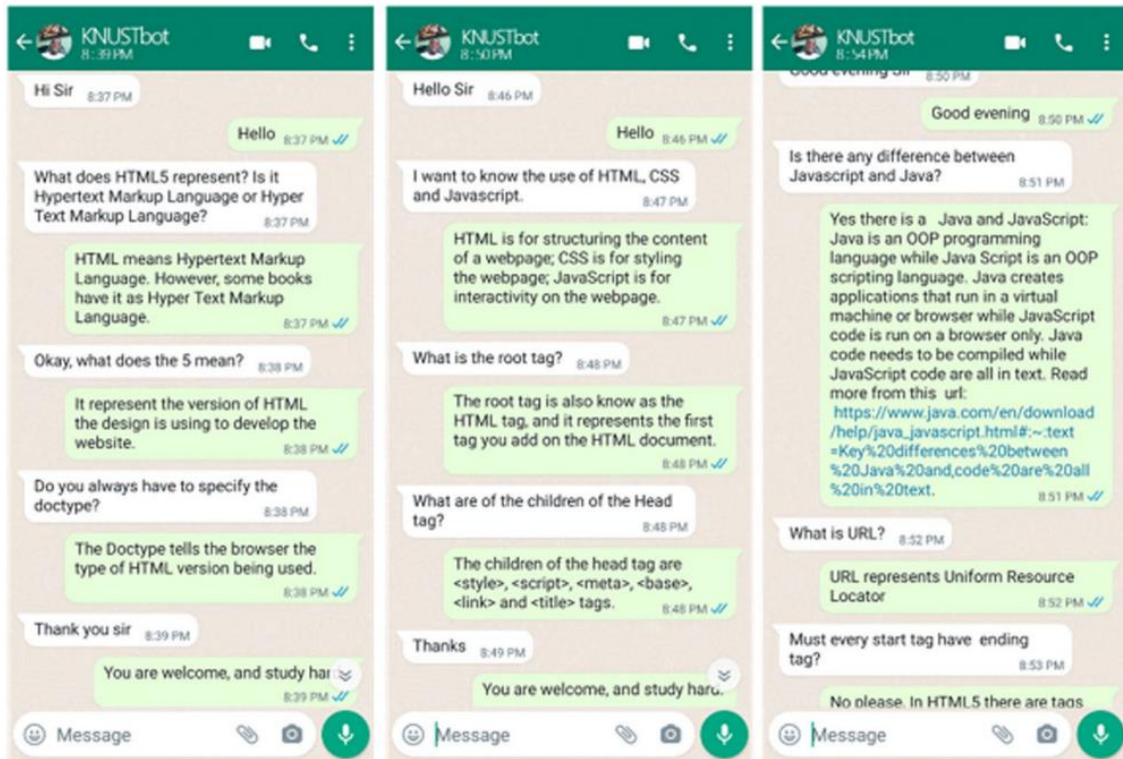


Figura 3 - Asistente virtual creado para WhatsApp

2.6.2 Asistentes virtuales que implementan inteligencia artificial

Además de los chatbots basados en reglas, en los últimos años se han desarrollado asistentes virtuales que incorporan técnicas de inteligencia artificial para mejorar la comprensión del lenguaje natural y hacer que la conversación sea más flexible. A diferencia de los sistemas que solo responden a comandos o preguntas previamente definidas, estos asistentes pueden interpretar distintas formas de expresar una misma necesidad, lo cual resulta pertinente en contextos educativos e institucionales, donde los usuarios no siempre formulan sus dudas de la misma manera. De acuerdo con Okonkwo y Ade-Ibijola (2021), los chatbots en educación han sido utilizados para ofrecer apoyo, orientación y atención automatizada, aunque todavía presentan retos relacionados con la precisión de sus respuestas y la calidad de la interacción.

Para que un chatbot pueda comprender mejor las consultas de los usuarios, algunos sistemas incorporan técnicas de procesamiento del lenguaje natural, como la clasificación de intención y la

extracción de entidades. La clasificación de intención permite identificar qué desea hacer o saber el usuario; por ejemplo, si busca información sobre reinscripción, titulación, pagos o becas. Por otro lado, la extracción de entidades ayuda a reconocer datos específicos dentro de la pregunta, como fechas, nombres de áreas, tipos de trámite o requisitos particulares. Estas técnicas forman parte de los componentes de comprensión del lenguaje natural que permiten que un asistente conversacional interprete con mayor precisión las necesidades del usuario (Ali, 2020).

En el caso de los chatbots educativos, la identificación correcta de la intención del usuario es importante porque permite ofrecer respuestas más relacionadas con la consulta realizada. Si el sistema no interpreta adecuadamente lo que el usuario necesita, puede entregar información poco útil o generar confusión. Okonkwo y Ade-Ibijola (2021) señalan que los chatbots en educación se han utilizado para apoyar la atención, orientación y comunicación con estudiantes, aunque uno de sus retos sigue siendo lograr interacciones más precisas y pertinentes. De forma similar, Rizou et al. (2023) abordan el uso de clasificación de intención y reconocimiento de entidades en agentes conversacionales dirigidos a estudiantes universitarios, destacando su utilidad para atender consultas de manera más eficiente.

En el ámbito institucional, integrar inteligencia artificial en un chatbot puede ayudar a ampliar la cobertura de consultas y evitar que las respuestas sean demasiado rígidas o limitadas. Sin embargo, esta ventaja también requiere mecanismos de control, ya que los modelos de IA pueden generar respuestas incorrectas, incompletas o difíciles de verificar cuando no cuentan con información confiable. Este problema se relaciona con las llamadas alucinaciones, que ocurren cuando un sistema de generación de lenguaje produce información que parece fluida y convincente, pero que no necesariamente es correcta o está sustentada en datos reales (Ji et al., 2023).

Algunos enfoques recomiendan apoyar las respuestas generadas por IA en fuentes de información verificables o en bases de conocimiento previamente definidas. Una estrategia relacionada con este enfoque es la generación aumentada por recuperación, conocida como RAG, la cual combina modelos generativos con mecanismos de recuperación de información para producir respuestas más específicas y fundamentadas en documentos o datos disponibles (Lewis et al., 2020). En el caso de Build A Chat, esta idea se retoma al limitar las respuestas del asistente al contenido del árbol de

conocimiento configurado por el administrador, de modo que la IA trabaje con información organizada, revisada y relacionada con el contexto institucional.

Por ello, en esta sección se revisan trabajos relacionados que han empleado inteligencia artificial en el desarrollo de chatbots, considerando sus enfoques, resultados y principales limitaciones (ver Tabla 3). Esta revisión permite contextualizar la propuesta de Build A Chat, ya que el proyecto busca aprovechar las ventajas de la IA para interpretar consultas en lenguaje natural, pero sin perder el control sobre la información que se proporciona al usuario. Con ello, se pretende que el sistema no solo responda de forma más natural, sino que también conserve precisión, confiabilidad y apego al dominio institucional definido.

Tabla 3 - Trabajos relacionados

Trabajo	Enfoque	Tecnología / método	Aporte	Limitación	Relación con Build A Chat
Essel et al.	chatbot educativo	WhatsApp / asistente virtual	apoyo a estudiantes	contexto académico, no administrativo	inspira interacción simple
Okonkwo y Ade-Ibijola	revisión sistemática	análisis de estudios	identifica beneficios y retos	no desarrolla plataforma específica	justifica uso en educación
Lee et al.	FAQ chatbot universitario	atención administrativa	reduce carga de consultas	limitado a FAQ	se relaciona con consultas institucionales
Build A Chat	plataforma administrable	web + IA + árbol de conocimiento	permite crear asistentes personalizados	depende de contenido actualizado	propuesta de esta tesis

A partir de los trabajos revisados, se observa que la mayoría de los chatbots educativos se enfocan en apoyar procesos de aprendizaje o responder preguntas frecuentes, pero no siempre ofrecen una plataforma administrable para que el personal institucional cree y actualice asistentes sin modificar código. En ese punto se ubica la aportación de Build A Chat.

3 METODOLOGÍA

En este capítulo se describe el proceso seguido para desarrollar la plataforma Build A Chat, desde la identificación de las necesidades iniciales hasta la definición de requerimientos, el diseño, la implementación y la validación del sistema. Debido a que se trata de un proyecto de desarrollo tecnológico, el trabajo se organizó por etapas, con el propósito de construir la aplicación de manera ordenada y facilitar el seguimiento de cada parte del proceso.

Durante el desarrollo se tomaron en cuenta distintos elementos, como la arquitectura del sistema, las herramientas utilizadas, el diseño de la base de datos, las interfaces de usuario, los módulos principales y la estrategia de pruebas. De esta manera, este capítulo no solo explica cómo se construyó la plataforma, sino también las razones que llevaron a tomar ciertas decisiones técnicas durante su elaboración.

3.1 Enfoque metodológico

El presente proyecto se desarrolló bajo un enfoque de desarrollo tecnológico aplicado, ya que su objetivo principal fue diseñar, construir y validar una plataforma web funcional para la creación y administración de asistentes virtuales institucionales. Este enfoque resulta adecuado porque el trabajo no se limita únicamente al análisis teórico del problema, sino que busca proponer una solución concreta que pueda utilizarse en un contexto educativo real.

Para organizar el trabajo se siguió una estrategia iterativa e incremental, en la que el sistema fue construido por módulos funcionales, comenzando con el registro e inicio de sesión de usuarios, seguido por la interacción con el asistente virtual, el monitoreo desde el panel administrativo, la creación y modificación de asistentes virtuales, y finalmente la validación de los requerimientos establecidos. Este enfoque permitió avanzar de manera progresiva, probar cada componente conforme se desarrollaba y realizar ajustes durante el proceso, asegurando que la plataforma respondiera a las necesidades identificadas en el contexto del Instituto Tecnológico de Culiacán. Para sintetizar el proceso seguido durante el desarrollo de Build A Chat, se presenta la Tabla 4, donde se sintetizan las fases, las actividades y productos generados.

Tabla 4 - Actividades desarrolladas en la creación de Build A Chat

Fase	Actividades	Producto generado
Diagnóstico	Observación, revisión de dudas frecuentes	Lista de necesidades
Análisis	Requerimientos funcionales	Tabla de requerimientos
Diseño	Arquitectura, base de datos, casos de uso	Diagramas
Desarrollo	Backend, frontend, chat, IA	Módulos funcionales
Pruebas	TDD, DDT, SBT	Casos de prueba
Evaluación	Pruebas con usuarios	Resultados de usabilidad

3.2 Requerimientos funcionales y de calidad

En la **Tabla 5** se pueden observar algunos requerimientos funcionales y de calidad propuestos para garantizar la funcionalidad y la calidad de la plataforma.

Tabla 5 - Requerimientos Funcionales

Identificador	Descripción	Prioridad
RE-01	El sistema debe permitir el registro de un usuario en el portal.	Media
RE-02	El sistema debe permitir el inicio de sesión de un usuario en el portal.	Media
RE-03	El sistema debe permitir al cliente usar el asistente virtual.	Alta
RE-04	El sistema debe permitir al administrador monitorear la actividad de los clientes.	Media
RE-05	El sistema debe permitir la creación de asistentes virtuales personalizados.	Alta
RE-06	El sistema debe permitir la modificación de los asistentes virtuales.	Alta

3.3 Arquitectura general del proyecto

La arquitectura general de Build A Chat se diseñó bajo un enfoque multicapa, con el propósito de separar responsabilidades y facilitar el mantenimiento del sistema (ver Figura 4). Esta organización permite distinguir la capa de presentación, donde se encuentran las vistas e interacciones del usuario; la capa de lógica y servicios, donde se procesa la autenticación, la gestión de asistentes, la comunicación en tiempo real y la integración con el componente de IA; y la capa de persistencia, encargada de almacenar usuarios, asistentes, nodos del árbol de conocimiento y conversaciones (ver Figura 5). Al estructurar el sistema de esta manera, se favorece la escalabilidad y la claridad del diseño, además de permitir que futuras mejoras se incorporen sin afectar de forma crítica a toda la plataforma.

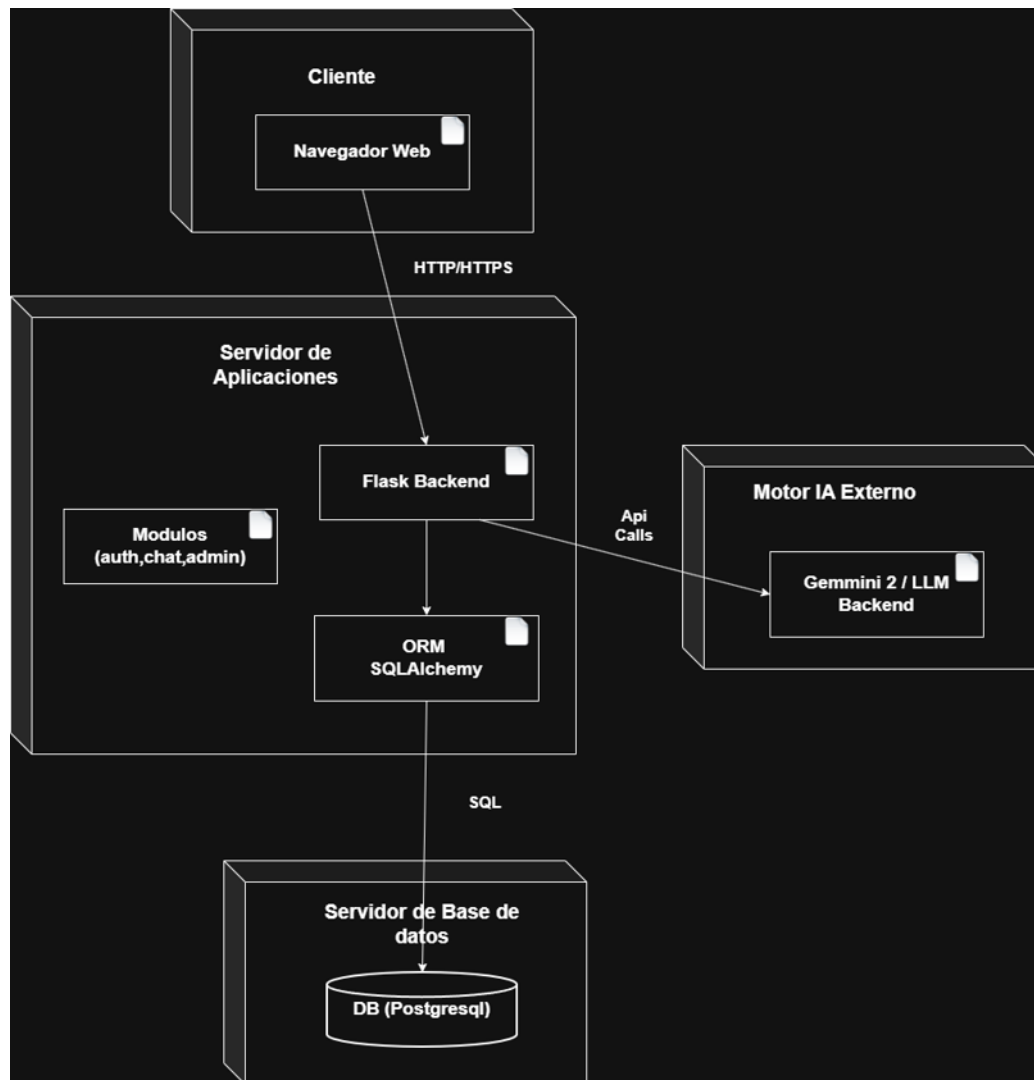


Figura 4 - Arquitectura general del proyecto

a) Capa de presentación (Frontend)

Corresponde a la interfaz accesible desde navegador web o aplicación móvil. Permite al usuario interactuar con el chatbot mediante texto o voz, y recibir respuestas, imágenes, menús y enlaces institucionales.

Se construyó utilizando HTML, CSS y Vue, con un enfoque minimalista y adaptable, pensado para que incluso usuarios con poca experiencia tecnológica puedan utilizarlo sin dificultad.

b) Capa de Lógica y Servicios (Backend):

Desarrollada en Flask y Socket.IO (Python), gestiona la comunicación entre el usuario y el modelo de inteligencia artificial. Sus principales funciones son:

- Registro y monitoreo de conversaciones.
- Conexión directa con el modelo de lenguaje natural.
- Envío y recepción de eventos en tiempo real.
- Sistema de modificación total de árboles y nodos para alimentar a un asistente virtual que da respuestas en lenguaje natural.

c) Capa de persistencia (Base de datos)

Se utilizó PostgreSQL para almacenar usuarios, mensajes, sesiones y chatbots. El sistema cuenta con respaldo automático y control de sesiones, que previene la duplicación o pérdida de datos asegurando la integridad y disponibilidad de los datos.



Figura 5 - Arquitectura multicapa

Integración de Inteligencia Artificial:

El chatbot emplea modelos de lenguaje natural (LLMs) ejecutados mediante la API de Gemini, lo que garantiza la privacidad institucional y una personalización total del sistema. El modelo utilizado permite:

- Comprender el lenguaje humano, incluso con errores ortográficos o abreviaturas.
- Generar respuestas naturales, coherentes y contextualizadas a partir de la información disponible en el árbol de conocimiento.

Humanización de la Respuesta

Más allá de proporcionar datos, el sistema fue diseñado para responder con empatía y tono adaptativo, acercando la experiencia digital a una interacción humana. Por ejemplo: “Parece que estás un poco confundido con el proceso de reinscripción. No te preocupes, te explico paso a paso...”

Este enfoque busca que el chatbot no solo brinde información, sino que también acompañe y oriente al estudiante con un lenguaje amigable.

3.4 Herramientas de desarrollo

Para el desarrollo de Build A Chat se utilizó Flask como Framework principal en el Backend. Flask es un micro Framework de Python que proporciona una base ligera y flexible para la construcción de aplicaciones web. Sobre Flask se integra Socket.IO, una extensión que permite implementar comunicación en tiempo real entre cliente y servidor mediante el protocolo WebSocket, manteniendo una conexión persistente. Para la gestión de concurrencia y evento asíncronos se utiliza gevent, una librería basada en corrutinas que optimiza la ejecución de múltiples conexiones simultáneas sin bloquear el servidor.

Para el Frontend se emplea Vue.JS, Framework de JavaScript para el diseño y base. Se decidió implementar la librería Socket.IO debido a su facilidad, soporte y factibilidad para manejar la comunicación bidireccional en tiempo real con el servidor. Este enfoque permite mantener una arquitectura ligera y de fácil mantenimiento.

Se utilizó PostgreSQL como Sistema Gestor de Base de Datos para realizar la autenticación y registro de usuarios. Finalmente, se integró Google Gemini mediante API para apoyar la interpretación de consultas en lenguaje natural y la generación de respuestas dentro del dominio institucional.

Para la realización de este proyecto y debido a ciertas limitaciones al tener que depender de librerías gratuitas se decidieron utilizar las respectivas versiones de los siguientes programas ya que logra cubrir todas las necesidades y además lo consigue de una manera gratuita las siguientes aplicaciones con las respectivas versiones que logro sacar mi visión adelante fueron: Python 3, Flask 3.1.0, Flask-SocketIO 5.5.1, Vue 3.5.13, PostgreSQL 18, Gemini API y por último el entorno de desarrollo fue utilizando github desktop con las herramientas que te da para modificar código de Visual Studio Code y gracias a su función poder subir cambios al programa en tiempo real así también como también se utilizó la plataforma render para poder alojar tanto la aplicación web así como también la base de datos teniendo ambos entornos en la nube.

3.5 Interfaces de Usuario

Las interfaces de usuario de Build A Chat fueron diseñadas para que tanto el usuario general como el usuario administrador puedan interactuar con la plataforma de forma clara y eficiente. En el caso del usuario general, se priorizó una interfaz conversacional similar a aplicaciones de mensajería, con el fin de reducir la curva de aprendizaje y facilitar el acceso a la información institucional. Para el usuario administrador, se incorporaron vistas específicas orientadas a la gestión de asistentes, edición del árbol de conocimiento y revisión de conversaciones, de manera que la información pueda actualizarse sin necesidad de modificar el código del sistema. En esta sección se describen las principales interfaces de usuario, sus elementos y el flujo de interacción asociado a cada rol. En la Figura 6, se presenta la interfaz principal y en la Figura 7 se puede apreciar la ventana de inicio de sesión.

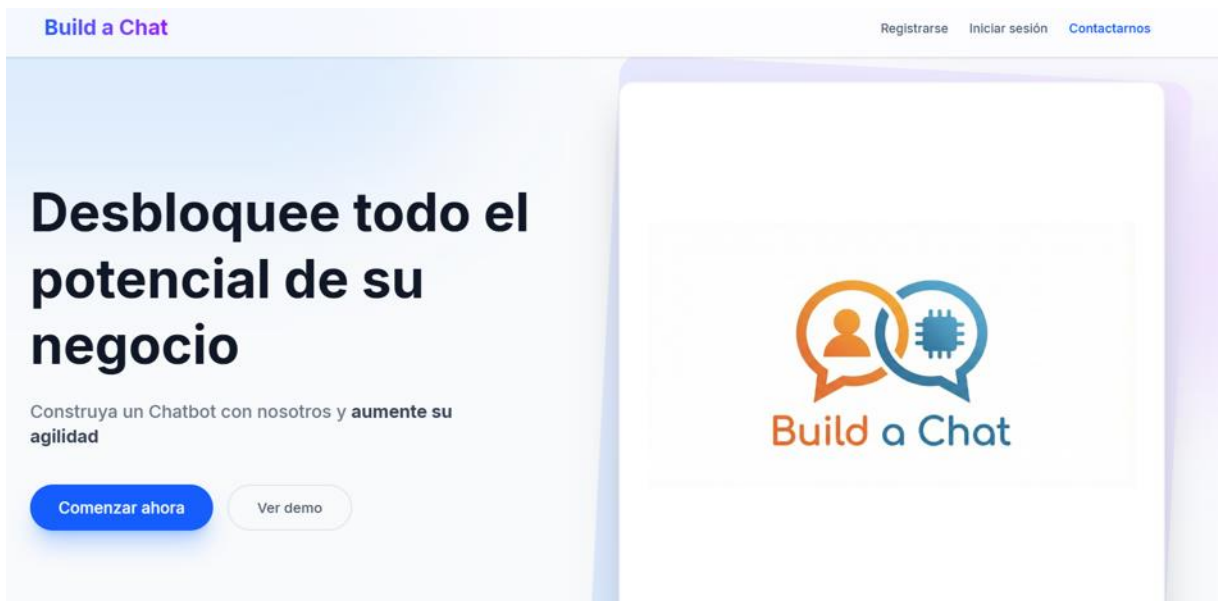


Figura 6 - Interfaz principal de Build A Chat

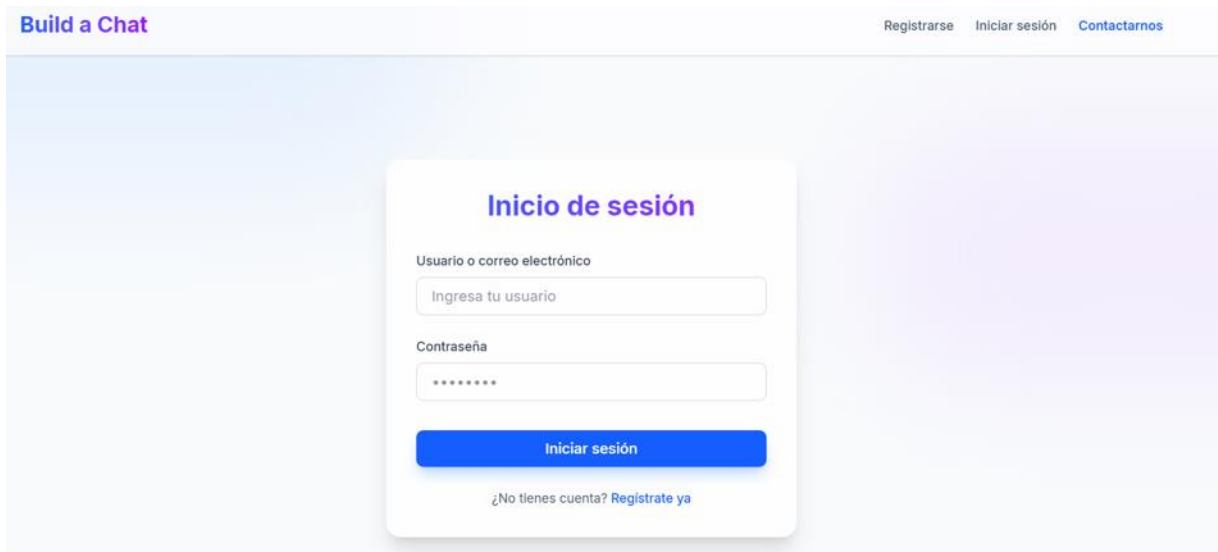


Figura 7 - Inicio de Sesión

En esta interfaz visual, se puede apreciar la ventana para el inicio de sesión, el usuario deberá ingresar un correo electrónico válido, así como una contraseña. Si la plataforma detecta que la contraseña ingresada es considerada “poco segura”, se le muestra un mensaje advirtiéndole al usuario que la cambie y que ingrese una contraseña más robusta. En caso de que el usuario o el correo estén ya asociados a una cuenta, se le informará, y con esto podrá elegir nuevas opciones. Una vez registrada

la cuenta, el sistema permite al usuario iniciar sesión y acceder a las funcionalidades correspondientes según su rol.

Los usuarios normales o generales tienen la capacidad de ver los asistentes virtuales publicados por algunos creadores, como se ve en el portal principal del sitio. Para buscar un chat en específico basta con ingresar el título del chat y la plataforma muestra los asistentes virtuales cuyo nombre coincide o se aproxima al criterio de búsqueda ingresado.

En la interfaz del usuario administrador se incorpora un botón para crear nuevo chatbot (ver Figura 8), funcionalidad que no se muestra a los usuarios generales debido a la diferencia de permisos entre roles.

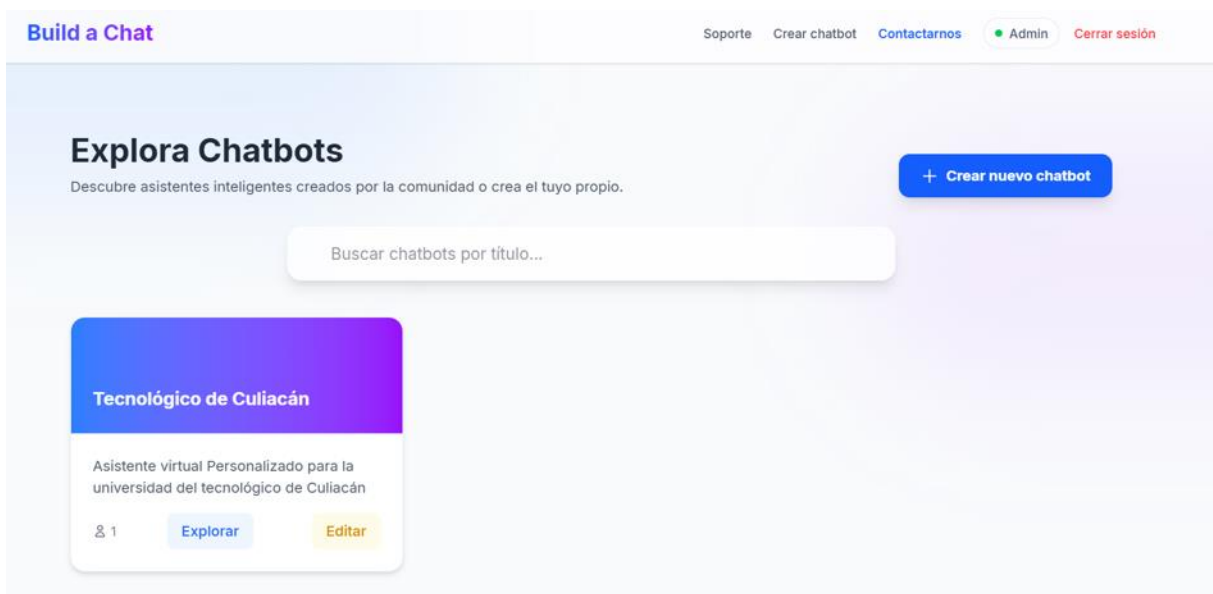


Figura 8 - Crear nuevo Chatbot



Figura 9 -Crear un chatbot

La Figura 9 muestra la ventana para crear y modificar el detalle de los asistentes virtuales. Podemos agregar diferentes niveles por medio de un editor de texto que reconoce texto parecido a JSON con indentación. La plataforma genera una representación visual del mapa de conocimiento del asistente. Además, se podrá ver al lado un ejemplo de la interfaz de inteligencia artificial que se le muestra al usuario final.

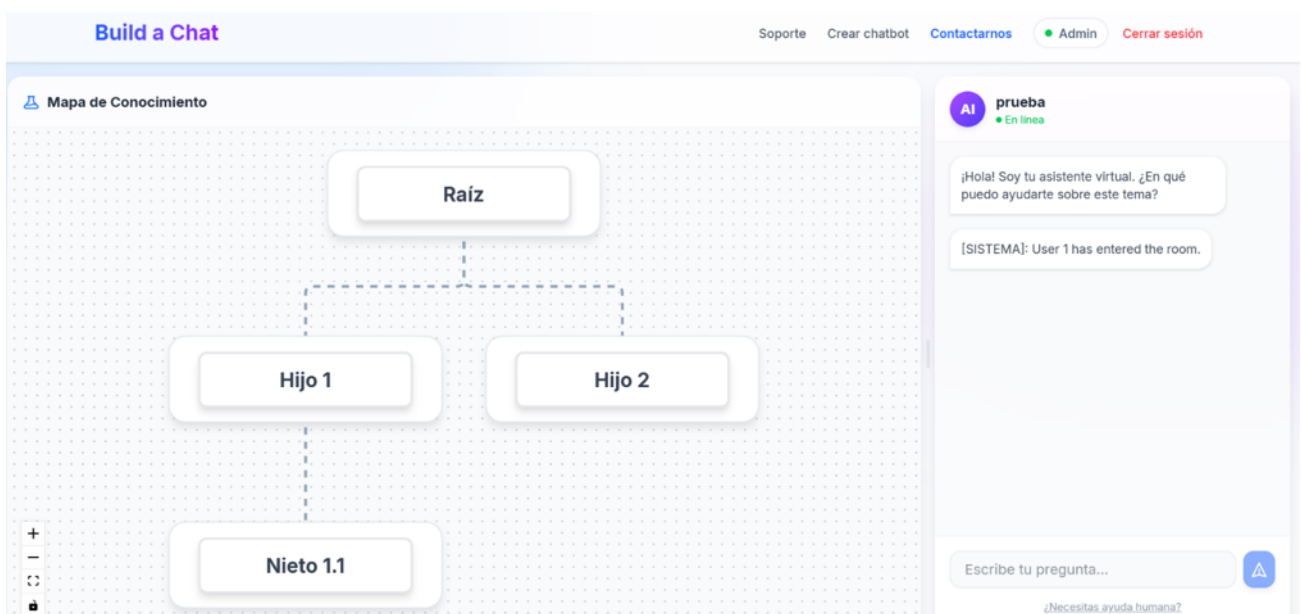


Figura 10 - Modificar un chatbot

Por otra parte, la Figura 11 despliega la vista previa de un chatbot ya estructurado

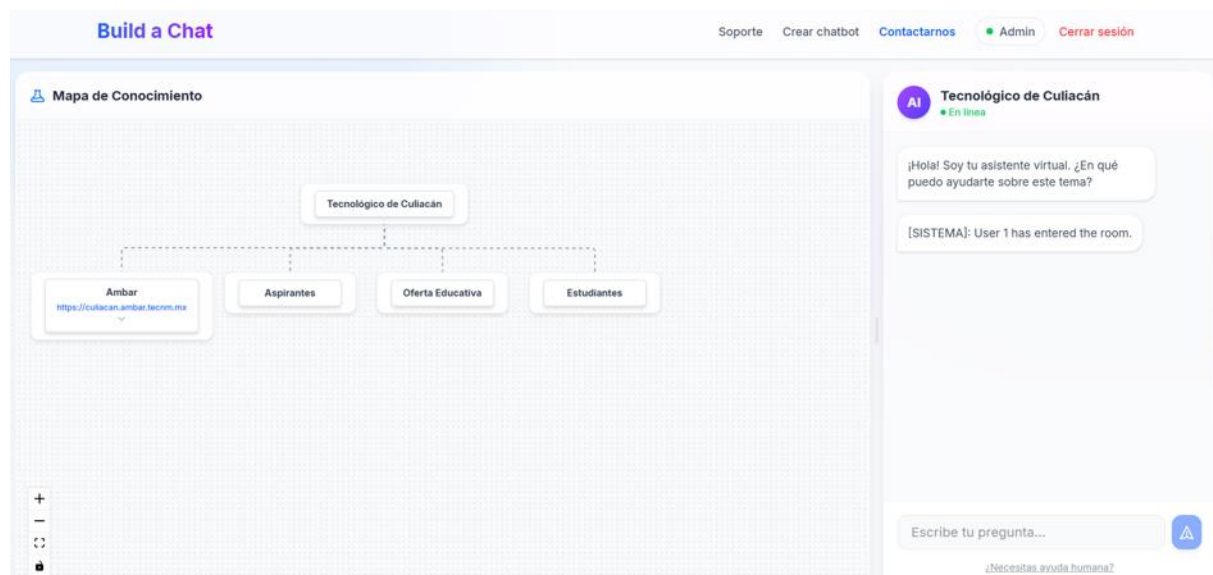


Figura 11 - Estructura de un chatbot

3.6 Diseño de la Base de datos

El diseño de la base de datos de Build A Chat tiene como finalidad almacenar de forma organizada la información necesaria para el funcionamiento de la plataforma. Entre los datos principales se consideran los usuarios, los roles, los asistentes virtuales, los nodos del árbol de conocimiento y el historial de conversaciones.

Debido a que el sistema necesita administrar contenido y, al mismo tiempo, conservar un registro de las interacciones realizadas por los usuarios, se propuso un modelo relacional. Este modelo permite mantener la consistencia de la información, relacionar los registros mediante claves foráneas y facilitar consultas posteriores para revisión, monitoreo o análisis del uso del sistema (ver Figura 12).

En esta sección se presenta el modelo entidad-relación de la plataforma, así como la descripción de las tablas principales y sus relaciones. Con ello se busca explicar cómo se organiza la información dentro del sistema y cómo esta estructura permite crear y administrar asistentes institucionales de manera ordenada y escalable.

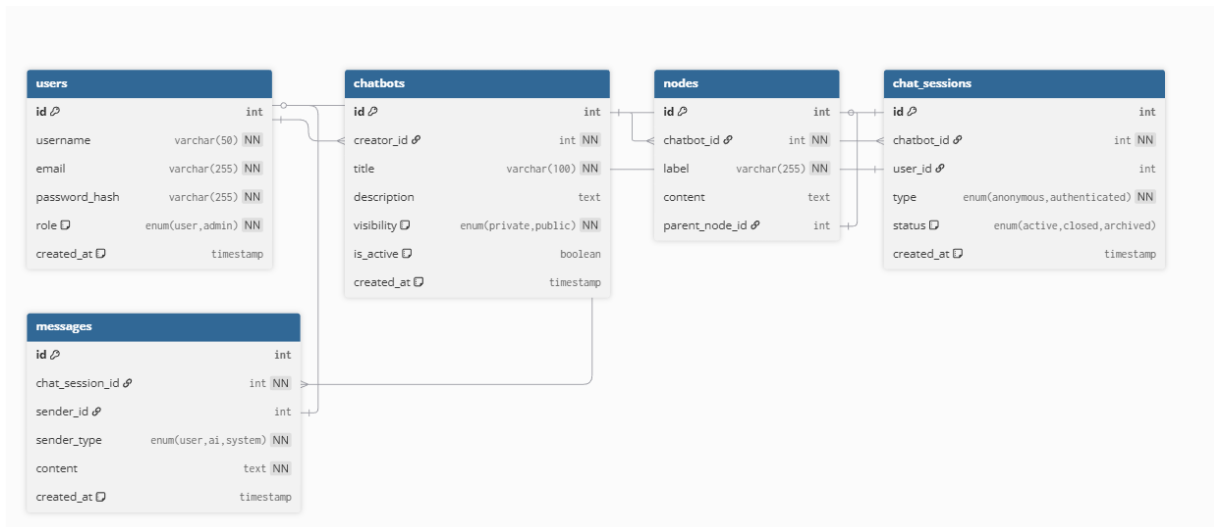


Figura 12 - Modelo Relacional de la Base de datos

3.7 Casos de Uso

Los casos de uso permiten describir de forma clara cómo interactúan los diferentes tipos de usuario con la plataforma. A través de ellos se delimitan las funcionalidades principales del sistema y el comportamiento esperado en cada proceso.

En Build A Chat se identifican dos actores principales: el usuario general y el usuario administrador. El usuario general es quien consulta información mediante el asistente virtual, mientras que el usuario administrador se encarga de gestionar los asistentes, organizar los contenidos y revisar las conversaciones realizadas dentro de la plataforma.

En esta sección se presenta el diagrama general de casos de uso, así como la descripción de los flujos principales y alternos de las funcionalidades más relevantes. Esto permite relacionar los casos de uso con los requerimientos definidos previamente y con la implementación realizada durante el desarrollo del sistema. A continuación, la Figura 13 presenta el diagrama de caso de uso desde el rol de un usuario estándar y la Figura 14 desde el rol de un usuario Administrador.

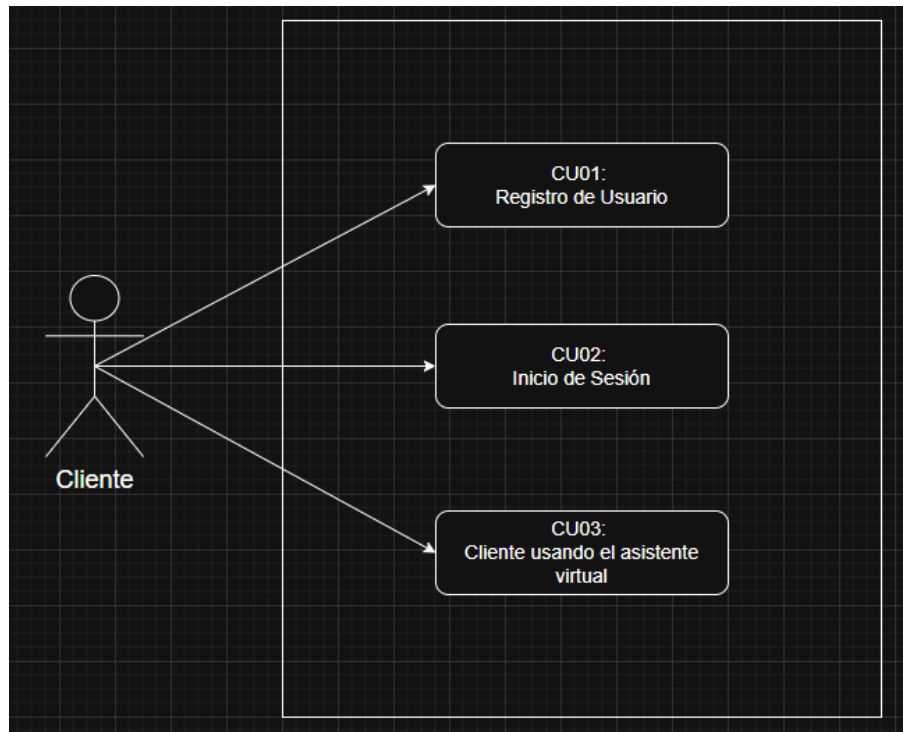


Figura 13 - Casos de uso de un usuario estándar

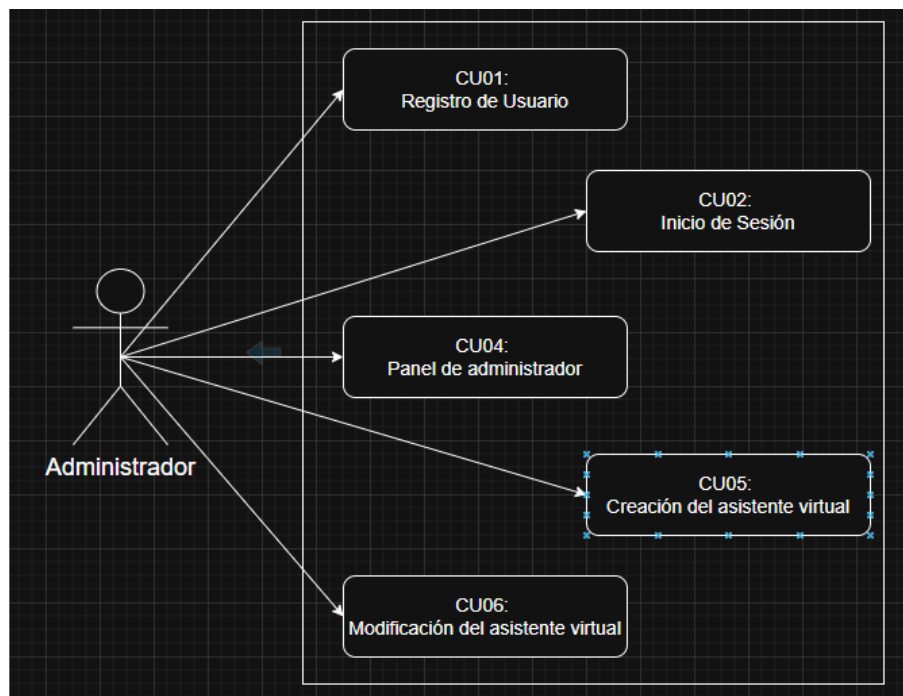


Figura 14 - Casos de uso de un usuario Administrador

3.8 Estrategia de pruebas y validación

Para verificar el funcionamiento correcto de la plataforma, se definió una estrategia de pruebas enfocada en los módulos principales del sistema. Esta estrategia considera tres enfoques: Test Driven Development, Data Driven Testing y State Based Testing. Cada uno de ellos permite revisar el comportamiento del sistema desde una perspectiva diferente y ayuda a detectar errores durante el desarrollo.

Se tomó como referencia el Desarrollo Guiado por Pruebas (Test Driven Development o TDD por sus siglas en inglés) se utilizó para estructurar pruebas sobre componentes críticos desde su construcción (Beck, 2002). Este enfoque consiste en diseñar primero una prueba que inicialmente falla y, posteriormente, implementar el código necesario para que dicha prueba se cumpla. De esta forma, el desarrollo se realiza de manera más controlada y se evita agregar funcionalidades sin comprobar su comportamiento esperado.

En el caso de Build A Chat, este enfoque se orientó principalmente a validar procesos como el registro y autenticación de usuarios, la creación y modificación de asistentes virtuales, la gestión de sesiones y el envío de mensajes. Con ello se buscó asegurar que las funciones esenciales del backend respondieran correctamente antes de avanzar hacia la integración completa de la plataforma (ver Tabla 6).

Las pruebas de software permiten detectar errores y comprobar que el sistema responde de acuerdo con los resultados esperados en distintos escenarios de uso (Myers et al., 2011). La validación permite revisar si el sistema construido responde a las necesidades del usuario y cumple los requerimientos definidos durante el análisis (Pressman & Maxim, 2020; Sommerville, 2016).

Las Pruebas Basadas en Datos (DDT) se utilizó para funciones que requieren validar múltiples combinaciones de datos sin duplicar casos de prueba. Permite ejecutar una misma prueba con distintos valores de entrada, aumentando la cobertura y detección de errores en los escenarios.

Las Pruebas Basadas en Estados (SBT) se usó para validar comportamientos dependientes del estado del sistema, siguiendo los diagramas de secuencia. Verifica que las transiciones sean las correctas y que cada acción responda al estado correspondiente.

Tabla 6 - Casos de prueba

Caso de prueba	Entrada	Resultado esperado	Resultado obtenido	Estado
Registro con correo nuevo	correo válido	cuenta creada	cuenta creada	Aprobado
Registro con correo existente	correo repetido	error controlado	mensaje mostrado	Aprobado
Crear asistente	nombre + árbol	asistente guardado	asistente visible	Aprobado
Consulta al chatbot	pregunta sobre titulación	respuesta del árbol	respuesta correcta	Aprobado

4 RESULTADOS

En este capítulo se presentan los resultados obtenidos durante el desarrollo y validación de Build A Chat. Se incluyen evidencias del cumplimiento de requerimientos, capturas del sistema, pruebas funcionales y resultados derivados de la evaluación con usuarios en los contextos institucional y empresarial.

4.1 Cumplimiento de requerimientos e hipótesis

Con la finalidad de presentar evidencia del funcionamiento de la plataforma Build A Chat, en la Tabla 7 se muestra cada requerimiento definido durante el desarrollo, el resultado obtenido, la evidencia que puede integrarse al documento y el estado actual de implementación. Esta información permite observar de manera ordenada qué módulos fueron desarrollados y cómo contribuyen al funcionamiento general del sistema.

Tabla 7 - Tabla de cumplimiento de requerimientos funcionales

Id	Requerimiento	Resultado obtenido	Evidencia	Estado
RE-01	El sistema debe permitir el registro de un usuario en el portal.	Se implementó un formulario de registro que permite crear una cuenta mediante datos básicos del usuario. El sistema valida que el correo o nombre de usuario no estén previamente registrados y almacena la información en la base de datos.	Captura del formulario de registro, y del mensaje de confirmación.	Finalizado
RE-02	El sistema debe permitir el inicio de sesión de un usuario en el portal.	Se desarrolló el módulo de autenticación, permitiendo que los usuarios ingresen a la plataforma mediante sus credenciales. El sistema distingue el tipo de usuario y muestra las opciones correspondientes según su rol.	Captura de la pantalla de inicio de sesión, validación de credenciales y acceso al panel correspondiente.	Finalizado
RE-03	El sistema debe permitir al cliente usar el asistente virtual.	Se implementó una interfaz conversacional donde el usuario general puede realizar consultas al asistente virtual. El sistema muestra respuestas relacionadas con la información previamente configurada en el árbol de conocimiento.	Captura del chat con el asistente virtual y ejemplo de una consulta respondida correctamente.	Finalizado
RE-04	El sistema debe permitir al administrador	Se desarrolló un panel administrativo que permite visualizar la actividad de los	Captura del panel administrativo mostrando usuarios,	Finalizado

	monitorear la actividad de los clientes.	usuarios y dar seguimiento a las conversaciones. Esto facilita identificar dudas que requieren atención directa del personal administrativo.	conversaciones o historial de mensajes.	
RE-05	El sistema debe permitir la creación de asistentes virtuales personalizados.	Se implementó un módulo que permite al usuario administrativo crear asistentes virtuales, definir su información base y organizar contenidos mediante una estructura editable.	Captura del formulario o interfaz de creación de asistentes virtuales.	Finalizado
RE-06	El sistema debe permitir la modificación de los asistentes virtuales.	Se agregó la funcionalidad para editar asistentes previamente creados, permitiendo actualizar su información, modificar contenidos y ajustar el árbol de conocimiento de acuerdo con las necesidades institucionales.	Captura de la pantalla de edición del asistente virtual y evidencia del cambio reflejado en el chat.	Finalizado
RE-07	El sistema debe almacenar el historial de conversaciones.	Se implementó el almacenamiento de mensajes en la base de datos, permitiendo conservar las conversaciones entre usuarios, asistentes virtuales y administradores para su consulta posterior.	Captura del historial de conversación y registro correspondiente en la base de datos.	Finalizado
RE-08	El sistema debe permitir la comunicación en tiempo real entre usuario y administrador.	Se integró comunicación bidireccional mediante Socket.IO, permitiendo que los mensajes enviados por el usuario y el administrador se actualicen sin necesidad de recargar la página.	Captura de conversación en tiempo real o prueba con dos sesiones abiertas.	Finalizado
RE-09	El sistema debe organizar la información mediante un árbol de conocimiento.	Se desarrolló una estructura jerárquica que permite al administrador organizar temas, subtemas y respuestas del asistente virtual, facilitando la administración de contenidos institucionales.	Captura del árbol de conocimiento y vista del asistente usando esa información.	Finalizado
RE-10	El sistema debe integrar inteligencia artificial generativa para apoyar las respuestas del asistente.	Se incorporó un componente de inteligencia artificial generativa para interpretar consultas en lenguaje natural y generar respuestas más claras y contextualizadas a partir de la información disponible.	Captura de una consulta abierta y respuesta generada por el asistente.	En desarrollo

Sobre las hipótesis gracias a mi investigación y la realización de pruebas con todos los usuarios que fueron seleccionados para probar el sistema todos lo recibieron de buena manera dando comentarios positivos y el como les parece una muy buena idea y revolucionaria para mejorar la calidad de vida en el tecnológico por lo que la hipótesis nula (H_0) queda totalmente anulada ya que si se propone mejoras a todos los usuarios relacionados y nos puede facilitar los procesos en el tecnológico así como también evitar cualquier error o situación extraordinaria que suele afectar a no solo los alumnos si no también al personal administrativo retrasando su trabajo por corrección de errores que se pueden dar por falta de información.

4.2 Evaluación con usuarios

Con el propósito de conocer el desempeño de Build A Chat más allá de su funcionamiento técnico, se realizó una evaluación con usuarios enfocada en identificar su percepción sobre la usabilidad, claridad y utilidad del asistente. Esta evaluación fue importante porque el objetivo de la plataforma no solo es responder consultas, sino también servir como un primer medio de orientación para temas académicos y administrativos.

Durante la evaluación se buscó observar si los usuarios podían realizar consultas comunes, entender las respuestas proporcionadas por el asistente y reconocer hacia qué canal o área podían dirigirse en caso de necesitar atención adicional. De esta manera, la valoración no se centró únicamente en comprobar que el sistema funcionara, sino también en analizar si la interacción resultaba comprensible y útil para los usuarios.

En esta sección se describe cómo se llevó a cabo la evaluación, el perfil de los participantes, los instrumentos utilizados y los resultados obtenidos. Asimismo, se incluyen algunas oportunidades de mejora identificadas a partir de los comentarios y la retroalimentación proporcionada por los usuarios.

Para las pruebas se decidió partirlo en 2 tipos de ambientes diferentes: Institucional y Empresarial, tal como se muestra en la Tabla 8.

Tabla 8 - Participantes

Contexto	Participantes	Perfiles	Aspectos observados	Resultado general
Institucional	50	aspirantes, alumnos, maestros, administrativos	claridad, facilidad de uso, orientación	valoración favorable
Empresarial	40	comensales, empleados, gerencia	consulta de menú, horarios, atención	valoración favorable

Institucional:

Se decidió darle la siguiente estructura al asistente virtual:

Los diversos nodos fueron representados de la forma en la que se puede apreciar en la Figura 15 referente a los aspirantes, en la Figura 16 se para que se pueda apreciar toda la estructura del chatbot de la oferta educativa y en la Figura 17 se visualizan las estructuras de los nodos desde la perspectiva de los estudiantes.

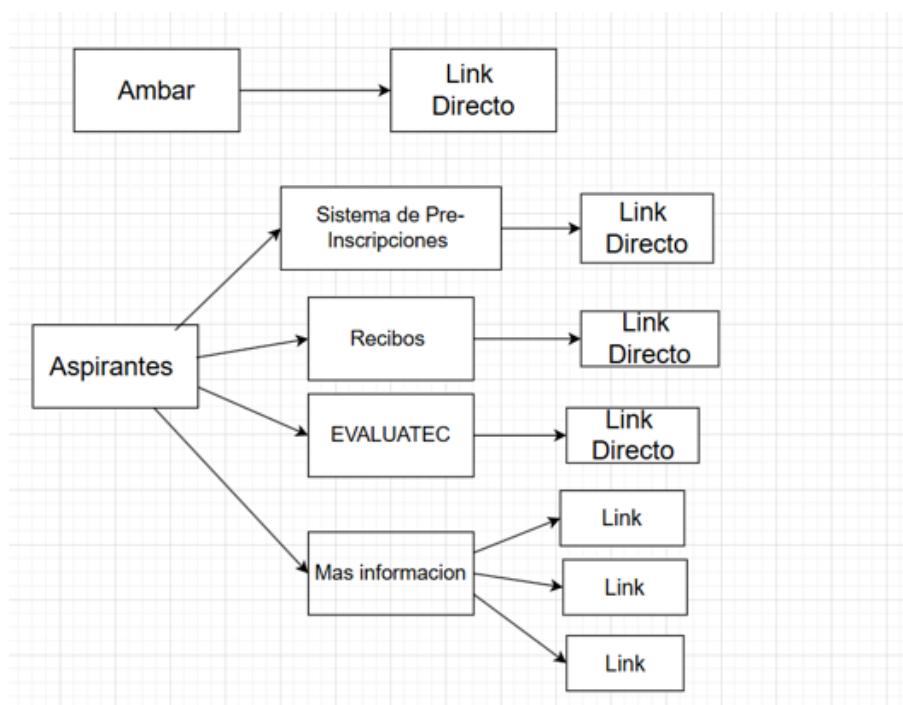


Figura 15 - Nodos del Aspirante

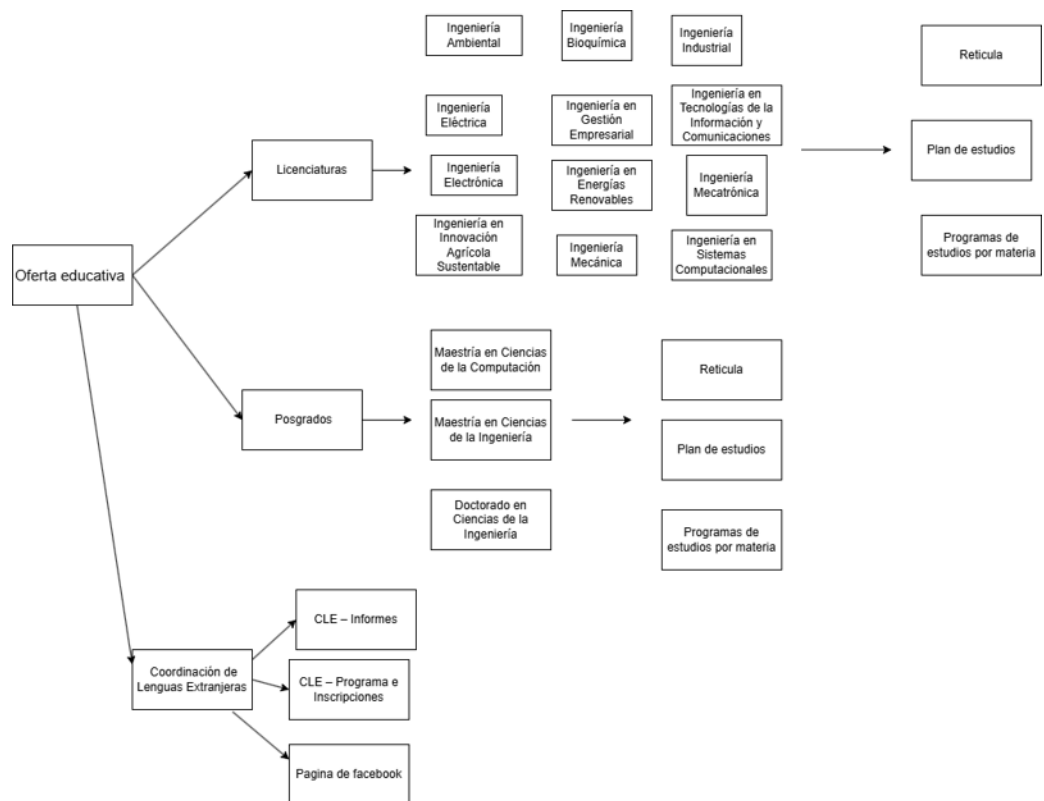


Figura 16 - Nodos oferta educativa

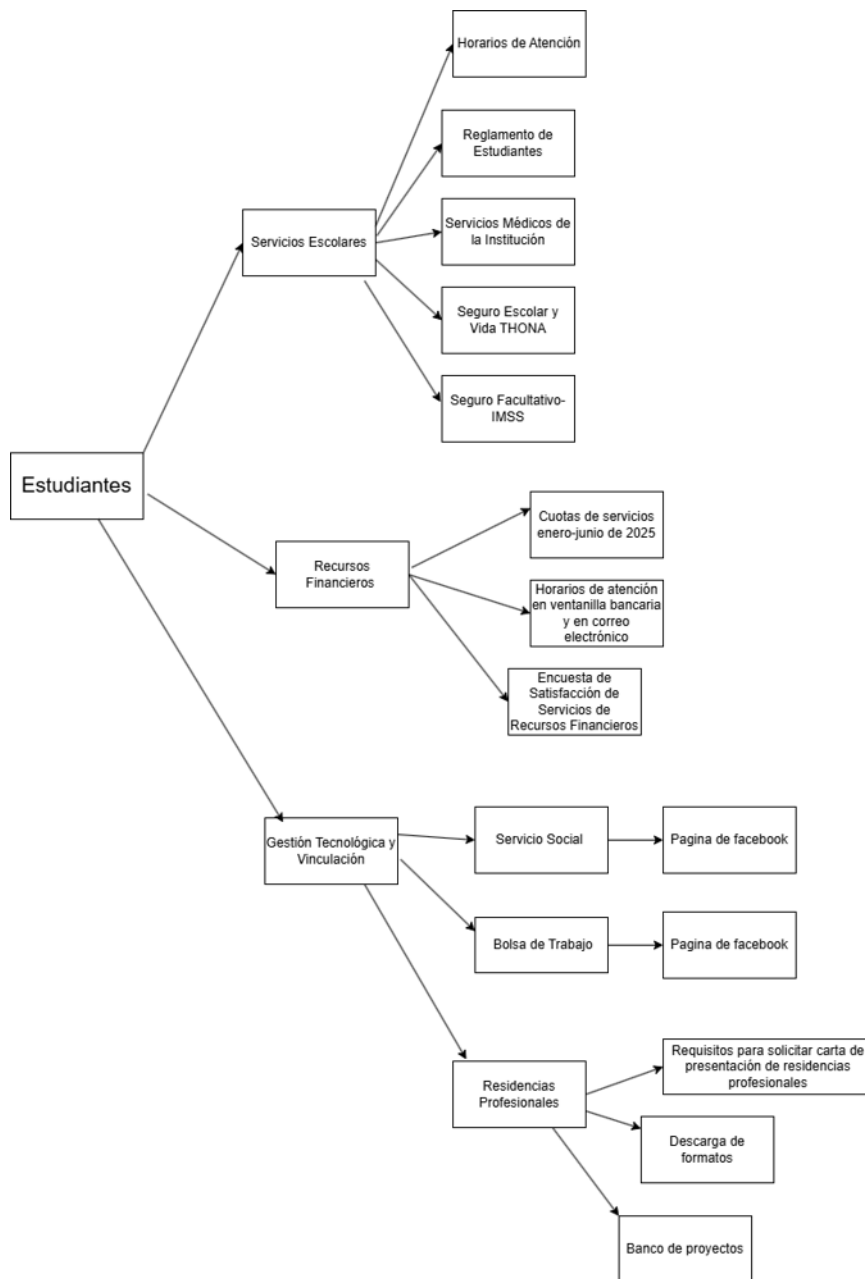


Figura 17 - Nodos del estudiante

Una vez interconectados los nodos en una estructura de árbol, se puede visualizar Figura 18 en la como forma de Mapa de Conocimiento, donde están enlazados los diversos nodos. A lado derecho de la pantalla, se puede visualizar la vista previa de una conversación siguiendo la estructura planeada.

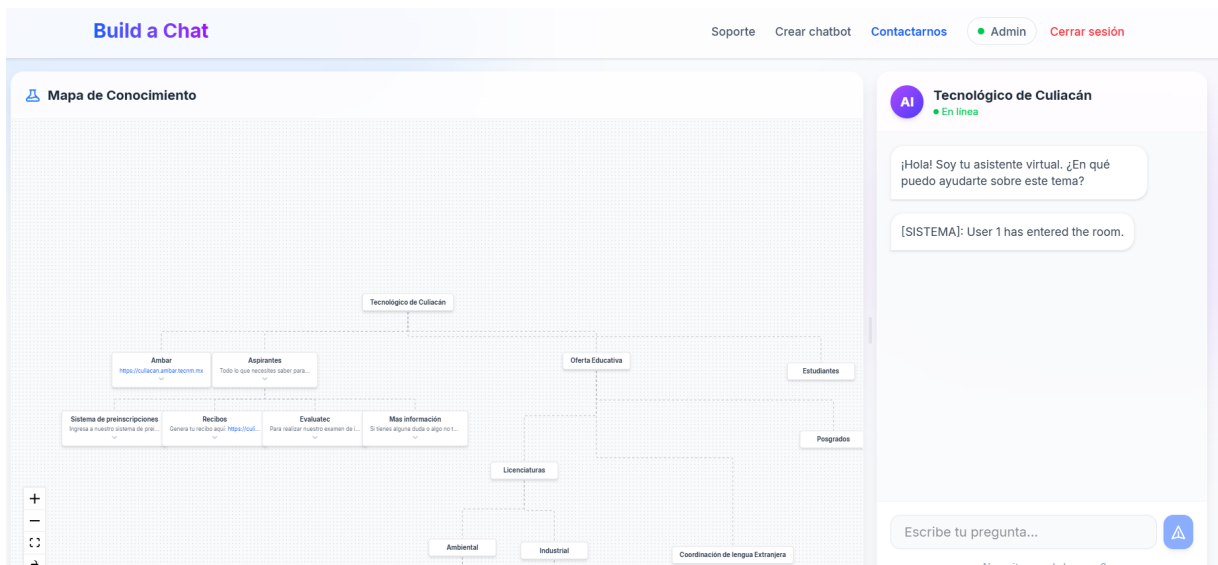


Figura 18 - Mapa de Conocimiento de un asistente virtual

En la Figura 19 se pueden apreciar capturas del flujo de una conversación ejecutadas desde un dispositivo móvil, donde el usuario solicita información de una carrera en especial, Además de solicitar información para inscribirse en una carrera.

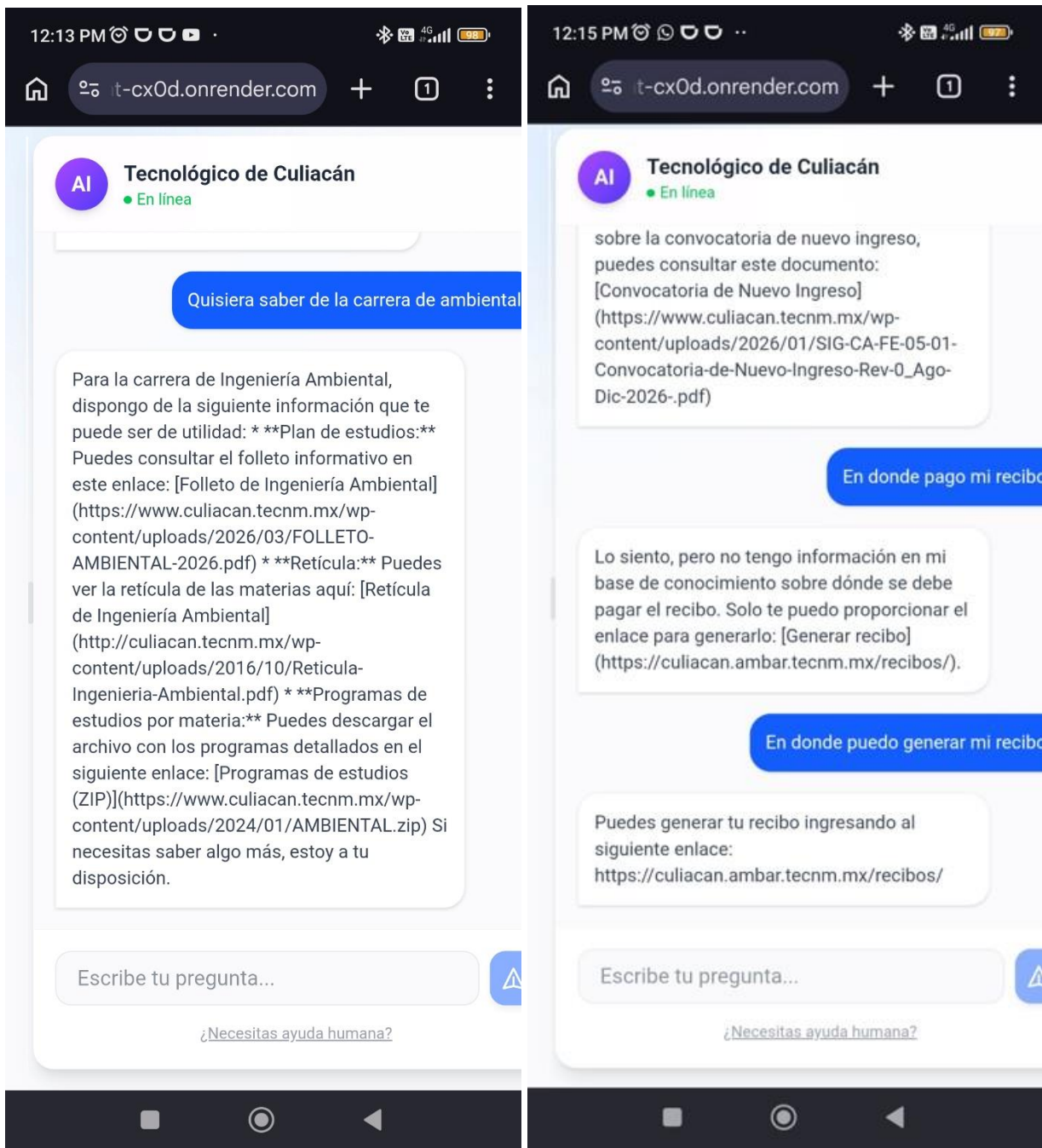


Figura 19 - Ejemplo de Conversaciones

Una vez concluida la fase de desarrollo y de haber realizado algunas pruebas con usuarios, se aplicaron encuestas a los participantes, cuyos roles eran aspirantes, maestros, personal administrativo, y alumnos matriculados en la institución.

La evaluación en el contexto institucional contó con la participación de 50 usuarios con perfiles vinculados con el entorno educativo, con el propósito de valorar el funcionamiento de Build A Chat desde distintas necesidades de consulta. Como se observa en la gráfica, el grupo con mayor participación correspondió a aspirantes a nuevo ingreso, quienes representaron el 37.5% de la muestra. Este perfil resulta relevante porque suele requerir orientación sobre procesos de admisión, requisitos, fechas y medios de contacto. Por otra parte, los alumnos y maestros representaron cada uno el 25% de los participantes, lo que permitió obtener una perspectiva tanto de usuarios que consultan información académica y administrativa como de quienes pueden orientar o acompañar procesos escolares. Finalmente, el personal administrativo representó el 12.5%, aportando una visión relacionada con la gestión de trámites, atención de dudas frecuentes y utilidad del asistente como apoyo para canalizar información institucional.

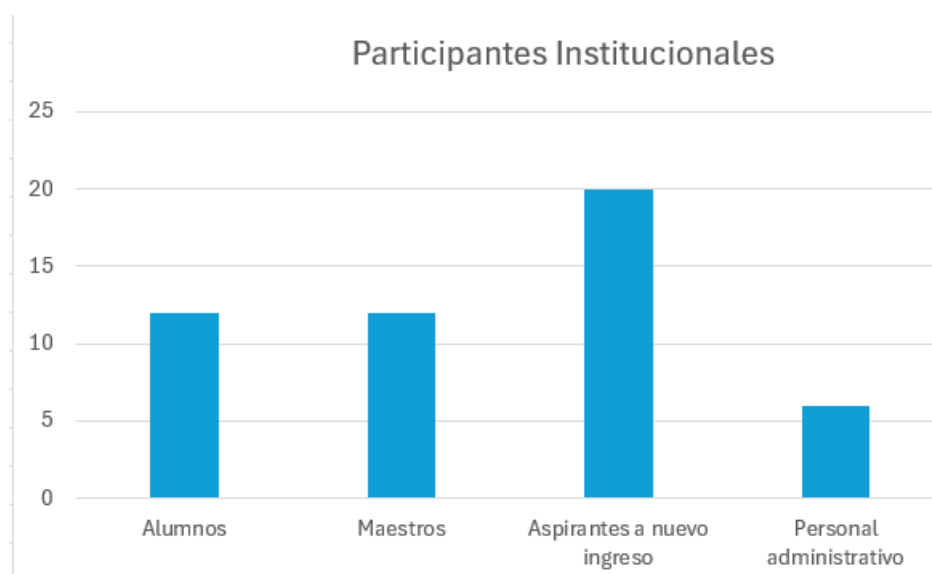


Figura 20 – Participantes Institucionales

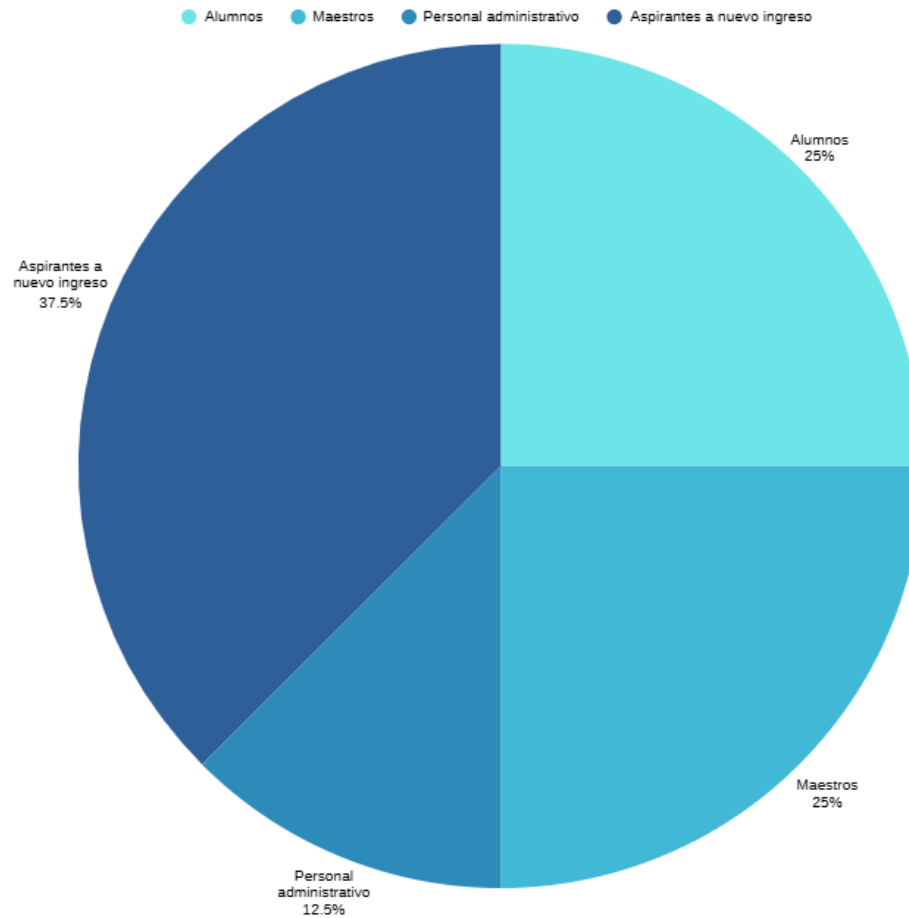


Figura 21 – Grafico de Porcentajes institucionales

Contexto Empresarial

Como parte complementaria para verificar la viabilidad de la implementación de una herramienta de asistente virtual en otro contexto fuera de lo institucional, se realizó una personalización con énfasis en un restaurant bar de la ciudad conocido como Toki Zushi. Esto con la finalidad de poder comprobar si este tipo de herramientas pueden ser replicadas en otros contextos compartiendo un panel administrativo para la gestión. Cabe destacar que se tuvo una participación asertiva por parte de clientes y personal operativo del negocio. Para esta adaptación, se aplicó una estructura de nodos como la que se puede apreciar en la Figura 21.

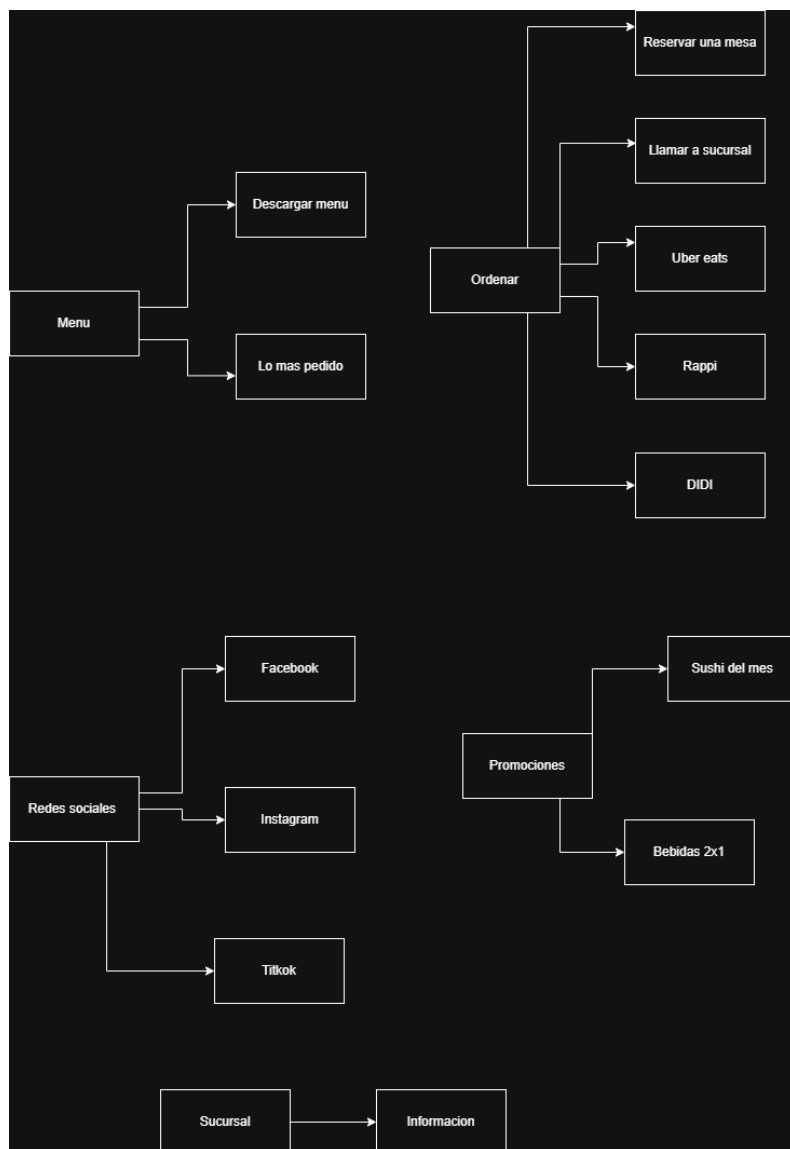


Figura 22 - Nodos para una prueba piloto hecha en un Restaurant

A continuación, se presenta la interfaz principal del mapa del Conocimiento adaptada para esta versión de la aplicación, tal como se aprecia en la Figura 22.

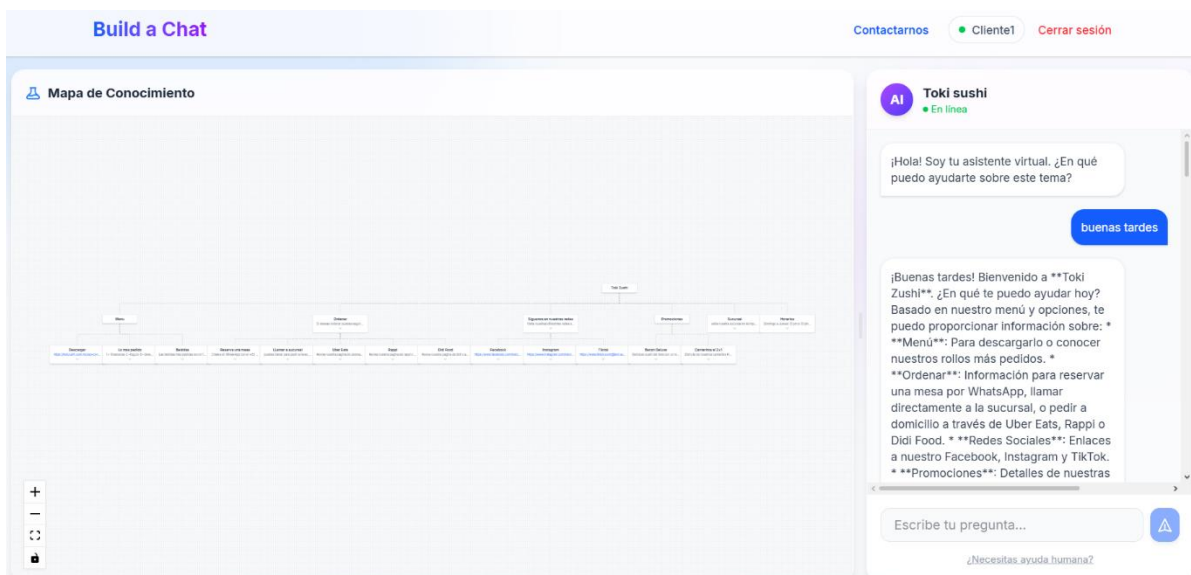


Figura 23- Mapa de Conocimiento del asistente virtual para el restaurant

La evaluación en el contexto empresarial contó con la participación 40 usuarios entre público en general y empleados, con el propósito de observar el funcionamiento de Build A Chat desde diferentes perspectivas. La mayor parte de los participantes correspondió a clientes o comensales, quienes representaron el 65% del total de la muestra. Este grupo permitió valorar la facilidad de uso del asistente desde la experiencia del cliente final. Por otra parte, el 25% estuvo conformado por empleados, quienes aportaron una visión relacionada con la operación diaria del negocio y las dudas frecuentes que pueden presentarse durante la atención. Finalmente, el 10% correspondió a gerentes y personal administrativo, cuya participación permitió identificar la utilidad del sistema para organizar información, dar seguimiento a consultas y apoyar la comunicación interna y externa del establecimiento

La Figura 23 refleja parte del contexto que Build A Chat le proporciona al usuario al solicitar el menú de platillos, donde el asistente virtual le responde con en enlace para acceder al documento, además de emitir sugerencias sobre los platillos sugeridos. Además, en la parte inferior de la conversación se puede apreciar donde el asistente le responde al usuario con los horarios de servicio del restaurante.

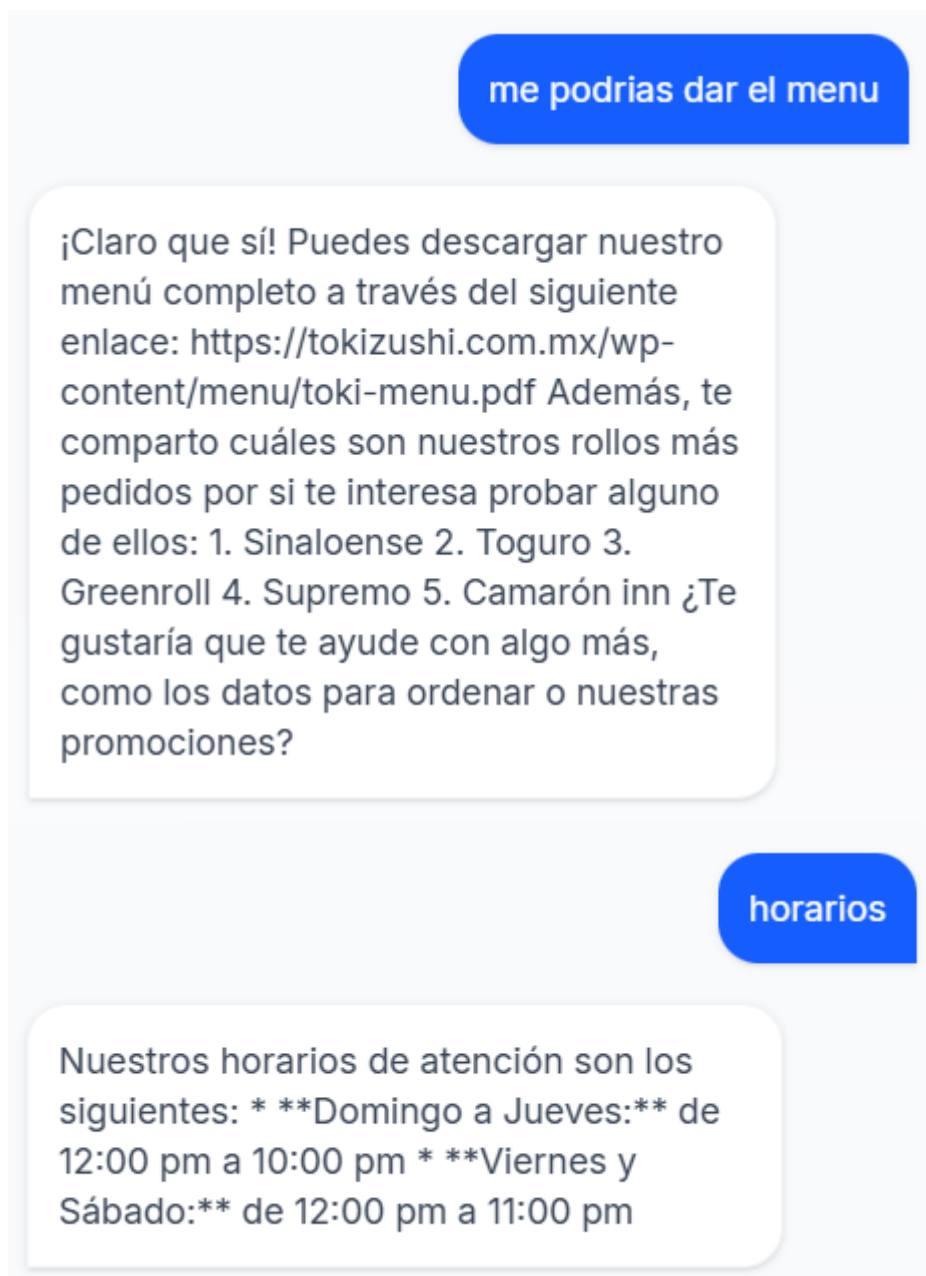


Figura 24 - Sugerencias del menú

Por otra parte, la Figura 24 refleja parte del contexto que Build A Chat le proporciona al usuario al solicitarle de manera confusa que horarios tienen. Sin embargo, el asistente virtual le responde con información de donde comunicarse vía WhatsApp. además, en la parte inferior de la conversación se puede apreciar donde el asistente esta sugiriendo los platillos más de servicio del restaurante.

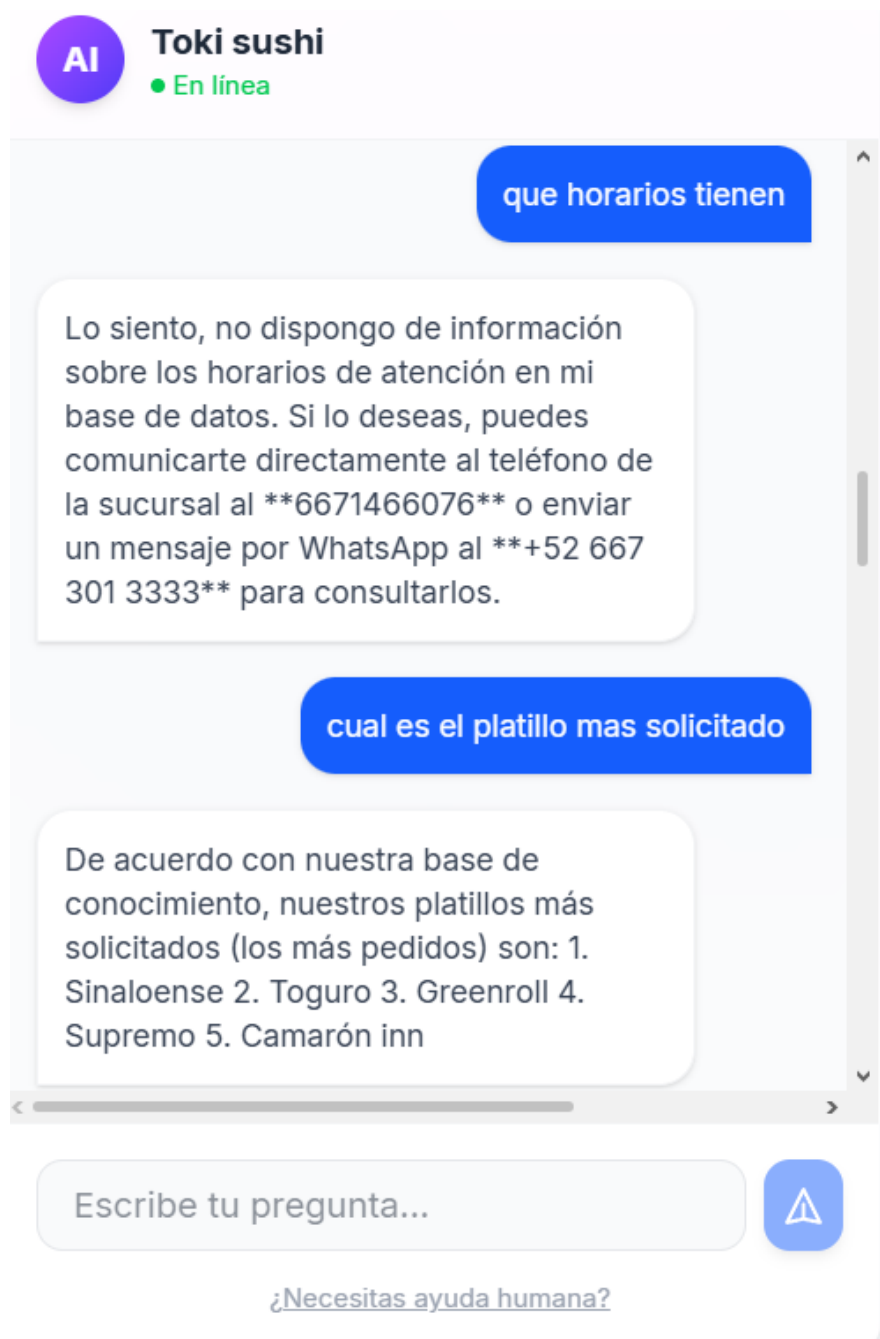


Figura 25 - Platillo más solicitado

La evaluación en el contexto empresarial contó con la participación de 40 usuarios con perfiles vinculados con el entorno empresarial, con el propósito de valorar el funcionamiento de Build A Chat desde distintas necesidades de consulta. Como se observa en la gráfica, el grupo con mayor participación correspondió a los comensales, quienes representaron el 62.5% de la muestra. Este perfil

resulta relevante porque resultan ser los más beneficiados ya que estos suelen consultar precios, objetos del menú, bebidas y promociones. Por otra parte, los empleados representaron el 25% de los participantes, lo que permitió obtener una perspectiva tanto de aquellos quienes son los principales en tratar con los clientes, son los que reciben la mayoría de las preguntas relacionadas al negocio, así como brindar atención al cliente. Finalmente, la gerencia representó el 12.5%, aportando una visión relacionada con la gestión del negocio, atención al cliente y manejo del negocio, así como dudas que suelen tener los clientes apoyando para canalizar información empresarial.

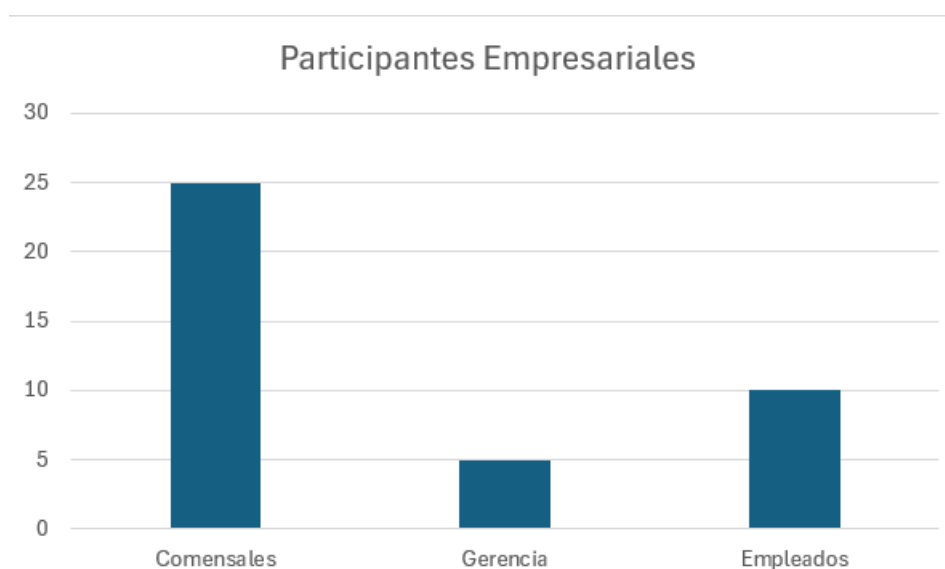


Figura 26- Participantes Empresariales

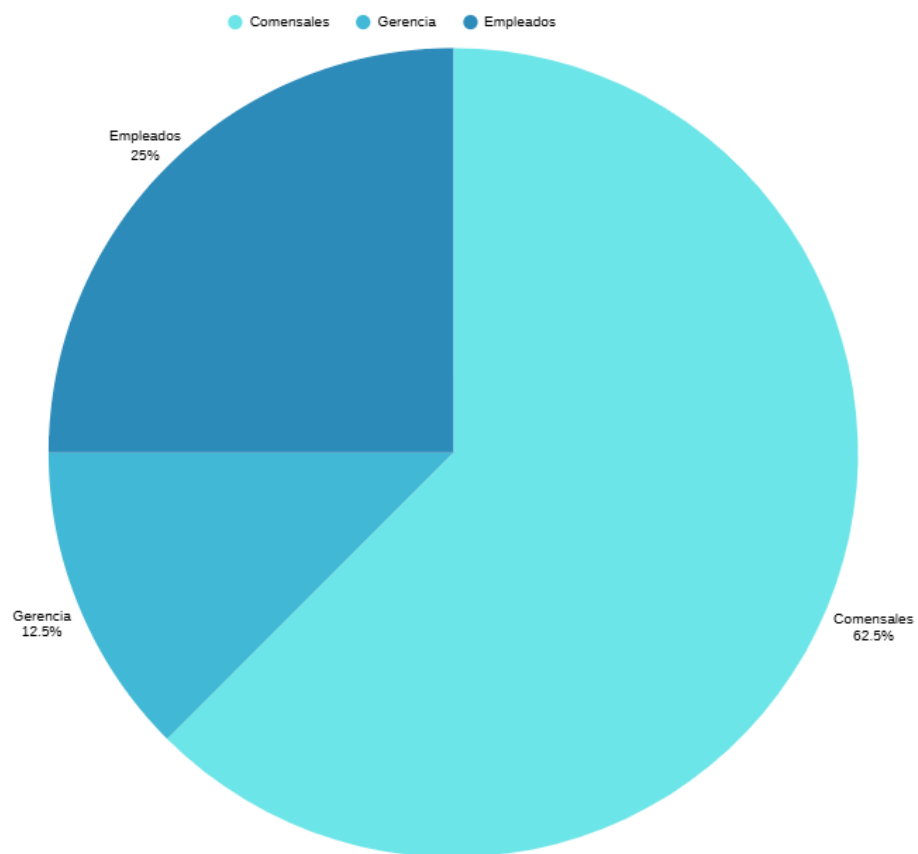


Figura 27- Grafico de porcentajes Empresarial

5 CONCLUSIONES

El desarrollo de Build A chat permitió transformar una idea o una necesidad plenamente identificada en una solución web funcional orientada a la consulta de información institucional. debido a que se quiere dar una solución para evitar la dispersión de información con la amplia cantidad de medios y plataformas donde se suele dar información, convocatorias, etc. Teniendo en cuenta esta necesidad, se desarrolló una aplicación web con una interfaz familiar basada en Meta Messenger, con la finalidad de que el usuario perciba una sensación de familiaridad y una gran facilidad para usarla, los usuarios se adaptaron fácilmente y tanto personas jóvenes como gente mayor la supieron utilizar.

Ya con una plataforma de uso fácil se identificó para el tecnológico que es lo más común que se pregunta se recopilo y se dio todo en una plataforma donde podrían preguntar libremente su duda o inquietud. Gracias a la arquitectura de capas utilizada se pudo separar el desarrollo desde la interfaz como se mencionó dándole algo agradable a la vista y un toque moderno a la idea, la lógica de servicios que es donde se cubrió la necesidad de una estructura cliente-servidor teniendo respuestas en tiempo real tanto del asistente así como lo puede ser el administrador que también genero un apartado extra donde este mismo puede brindar ayuda humana a aquellos que no se sintieron tan cómodos con el asistente virtual o no encontraron lo que buscaba dando una atención única que dejo satisfechos a los usuarios además que otra cosa que permitió fue poder llevar esta idea y lograr un panel para la creación de árboles de conocimiento donde tu creas los nodos que van alimentando a el asistente virtual para dar una mejor experiencia al usuario.

Gracias a las herramientas que se utilizaron y gracias a la implementación de PostgreSQL se permitió por la base de datos el poder no solo guardar las sesiones si no las conversaciones, las sesiones y las preguntas que se le hacían. Con estas funciones se agregaron módulos para los siguientes procesos: Inicio de sesión, registro de usuario, creación de chatbot, lista de usuarios conectados, la exploración de los diversos chatbots creados y el poder guardar cada nodo que se crea para tu asistente virtual así se modifique la información de un nodo, se crean diversos nodos nuevos o se borre actualización desactualizada.

Una vez realizadas las pruebas funcionales del sistema, el funcionamiento de la plataforma no solo logro sorprender a cada una de las personas que utilizaron el asistente virtual sino que también aunque este mismo no tenía todas las respuestas gracias a usar la api de Gemini 2 las respuestas generadas siempre daban algo concreto ejemplo de esto si se preguntaba no te decía que no lo tiene y te ignoraba, este analizaba tu pregunta y la lograba interpretar aun si esta tuviese errores de ortografía o gramática dándote una solución como lo puede ser contacto humano o redirigiéndote a temas similares para poder darte una respuesta satisfactoria. Gracias a las capacidades del sistema, la retroalimentación de los usuarios fue favorable, principalmente en aspectos relacionados con facilidad de uso, rapidez de respuesta y claridad de la información.

Sobre la validación gracias a que se cumplieron todos los requerimientos funcionales se permitió probar la plataforma, revisar que todo esté en orden que no hubiera ningún tipo de error o contratiempo a la hora de realizar las pruebas para los usuarios además de cumplir con todos los objetivos que se propusieron para el proyecto sin dejar de lado alguno o que quedara inconcluso. Los procesos que puede realizar Build A Chat tiene potencial para apoyar en la atención de consultas frecuentes del entorno centralizado en donde este sea implementado ya que con la plataforma se contestan las preguntas recurrentes, se da un medio donde se pueda acceder a toda la información tanto previa como nueva que se pueda ir agregando. Esta validación se fortalece gracias a la evaluación que se hizo con los usuarios como todos terminaron dando una respuesta positiva al programa dando un producto donde se da claridad de información, tiempos de respuesta inmediatos y además una gran facilidad de uso para todo tipo de personas que puedan usar esta plataforma.

En conclusión, Build A Chat se plantea como una herramienta útil para mejorar la comunicación entre estudiantes, docentes y personal administrativo, al centralizar información y facilitar su consulta mediante una plataforma web conversacional. Aunque el proyecto cumple con los módulos principales planteados, aún existen oportunidades de mejora, como la integración directa con sistemas institucionales, la ampliación del contenido disponible en los asistentes, el fortalecimiento de la seguridad y la evaluación con una muestra más amplia de usuarios. Estas mejoras permitirían consolidar la plataforma y aumentar su impacto dentro del entorno educativo.

5.1 Limitaciones

En primer lugar, la plataforma se desarrolló como una primera versión funcional, por lo que su alcance se centra en la creación, administración y consulta de asistentes virtuales institucionales, sin incluir todavía una integración directa con sistemas oficiales del Instituto Tecnológico de Culiacán, como Kardex, pagos, plataformas académicas internas u otros servicios administrativos. Debido a esto, la información que se consulta en el asistente depende de los contenidos previamente capturados y actualizados por el usuario administrativo.

Otra limitación importante se relaciona con la evaluación del sistema. Aunque se contempla la validación mediante pruebas funcionales y revisión de requerimientos, la evaluación con usuarios aún puede fortalecerse con una muestra más amplia de estudiantes y personal administrativo. Esto permitiría obtener datos más precisos sobre la facilidad de uso, la claridad de las respuestas, el tiempo de atención y el impacto real de la plataforma en la reducción de consultas repetitivas.

También debe considerarse que el uso de inteligencia artificial generativa puede presentar limitaciones en la precisión o interpretación de algunas consultas, especialmente cuando la información proporcionada al asistente no está completa, está desactualizada o no se encuentra bien organizada dentro del árbol de conocimiento. Por lo tanto, el sistema requiere una administración constante de los contenidos para mantener la calidad de las respuestas y evitar información incorrecta o ambigua.

Finalmente, al tratarse de una plataforma web que maneja usuarios, conversaciones e información institucional, existen aspectos de seguridad, privacidad y control de acceso que pueden seguir fortaleciéndose en futuras versiones. Si posteriormente se busca conectar la plataforma con sistemas internos o datos sensibles, será necesario implementar mecanismos más robustos de protección, validación de permisos y manejo seguro de la información.

5.2 Trabajo futuro

Para un futuro se quiere expandir de muchas formas ya que este proyecto tiene muchas oportunidades de crecimiento, principalmente una de las cosas que pueden ser integradas que esto es para las respuestas sería algo como un sistema de prioridad, normalmente se suelen dar múltiples respuestas, entonces se darán a las que se tienen más alto tráfico ejemplo si el tecnológico usa

Facebook y Twitter(X) se le dará mayor prioridad a Facebook ya que es en esta plataforma donde suelen darse más noticias, es mucho más usada y cuenta con muchos más seguidores que la otra.

Se propone explorar la integración directa de Build A Chat con sistemas institucionales como Kardex, pagos, plataformas académicas internas u otros servicios oficiales del Instituto Tecnológico de Culiacán. Esta integración dependerá de los permisos, accesos, condiciones técnicas y políticas de seguridad establecidas por la institución.

También se considera como mejora futura ampliar las capacidades del asistente virtual para atender más áreas administrativas, incorporar nuevos tipos de usuarios y fortalecer el seguimiento de conversaciones que requieran intervención humana. Asimismo, podría realizarse una evaluación con una muestra más amplia de estudiantes y personal administrativo para medir con mayor precisión el impacto de la plataforma en la atención de consultas, los tiempos de respuesta y la claridad de la información proporcionada.

Otra línea de mejora consiste en reforzar los mecanismos de seguridad, privacidad y control de acceso, especialmente si en versiones posteriores se conectan datos personales o información académica sensible. De igual manera, se podría optimizar el modelo de inteligencia artificial utilizado, mejorar la precisión de las respuestas y permitir que la plataforma se adapte a otros contextos educativos o institucionales.

6 COMPETENCIAS DESARROLLADAS

Durante el proyecto se reforzó el uso de Flask para organizar rutas y servicios del backend, así como el manejo de Socket.IO para la comunicación en tiempo real entre usuarios y administrador. El trabajo permitió aplicar conocimientos de programación web, bases de datos, arquitectura de software, comunicación en tiempo real e integración de inteligencia artificial generativa, además de reforzar habilidades para la documentación, organización y validación de un sistema funcional.

Una de las principales competencias desarrolladas fue el análisis de requerimientos, ya que fue necesario identificar las necesidades de los usuarios, comprender los problemas de comunicación

institucional y traducirlos en funcionalidades concretas para la plataforma. Esto permitió definir los módulos principales del sistema, como el registro e inicio de sesión, la creación de asistentes virtuales, el árbol de conocimiento, el historial de conversaciones, el chat con el asistente y el panel administrativo.

También se fortaleció la competencia de diseño de software, debido a que se planteó una arquitectura modular basada en capas, separando la presentación, la lógica de servicios y la persistencia de datos. Esta organización facilitó la comprensión del sistema, su mantenimiento y la posibilidad de escalarlo en futuras versiones. De igual manera, se aplicaron conocimientos sobre modelado de datos para estructurar la información relacionada con usuarios, asistentes virtuales, sesiones de chat, mensajes y nodos del árbol de conocimiento.

En el área de desarrollo web, el proyecto permitió aplicar herramientas y tecnologías para construir una plataforma funcional, integrando backend, frontend, base de datos y comunicación en tiempo real. Asimismo, se reforzó la capacidad de implementar interfaces de usuario claras y accesibles, tomando como referencia aplicaciones de mensajería conocidas para facilitar la interacción entre los usuarios y el sistema de inteligencia artificial generativa, utilizada para apoyar la interpretación de consultas en lenguaje natural y la generación de respuestas relacionadas con la información institucional disponible. Esto permitió comprender mejor el papel de la inteligencia artificial dentro de aplicaciones web modernas, así como sus posibilidades y limitaciones en contextos educativos.

Finalmente, el proyecto fortaleció competencias de validación y documentación, ya que fue necesario revisar el cumplimiento de requerimientos, realizar pruebas funcionales y organizar evidencias del funcionamiento de la plataforma. Además, se desarrollaron habilidades de comunicación técnica y trabajo estructurado, al documentar el proceso completo desde la identificación del problema hasta la propuesta de solución, los resultados obtenidos y las posibles mejoras futuras.

7 FUENTES DE INFORMACIÓN

Ali, N. (2020). *Chatbot: A Conversational Agent employed with Named Entity Recognition Model using Artificial Neural Network*.

Arredondo, F. C., Taípe, J. I., & Cruz, M. A. (2024). . Revisión sistemática de la evolución de los asistentes virtuales: Desde los primeros Chatbots hasta la IA conversacional. *Micaela Revista de Investigación-UNAMBA*, 33-39.

Beck, K. (2002). *Test-driven development: By example*. Addison-Wesley.

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). *On the dangers of stochastic parrots: Can language models be too big?* Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, 610–623.

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 610–623.
<https://doi.org/10.1145/3442188.3445922>

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., ... Liang, P. (2021). *On the opportunities and risks of foundation models*. arXiv. <https://arxiv.org/abs/2108.07258>

Camazón, J. N. (2010). *Aplicaciones web*. Editex.

Chukwuere, J. E. (2024). *The future of generative AI chatbots in higher education*. arXiv. <https://arxiv.org/abs/2403.13487>

Essel, H. B., Vlachopoulos, D., Tachie-Menson, A., Johnson, E. E., & Baah, P. K. (2022). The impact of a virtual teaching assistant (chatbot) on students' learning in Ghanaian higher education. *International Journal of Educational Technology in Higher Education*, 19(1), 57.
<https://doi.org/10.1186/s41239-022-00362-6>

Fette, I., & Melnikov, A. (2011). *The WebSocket Protocol* (RFC 6455). Internet Engineering Task Force. <https://doi.org/10.17487/RFC6455>

Fowler, M. (2015, agosto 26). *PresentationDomainDataLayering*. Martin Fowler.
<https://martinfowler.com/bliki/PresentationDomainDataLayering.html>

Gómez, & Ríos. (2022). Asistentes virtuales basados en inteligencia artificial: Revisión sistemática bajo la metodología PRISMA. *Revista Iberoamericana de Tecnología Educativa*, 45–61.

- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
<https://www.deeplearningbook.org/>
- Hong, M., Jiang, D., Wen, J., Su, Z., Li, Y., Sun, Y., Wang, G., & Zhang, C. J. (2026). *RAL2M: Retrieval Augmented Learning-To-Match Against Hallucination in Compliance-Guaranteed Service Systems*.
- International Organization for Standardization. (2018). *ISO 9241-11:2018. Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts*. ISO.
<https://www.iso.org/standard/63500.html>
- Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Chen, D., Dai, W., Chan, H. S., Madotto, A., & Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12), Article 248. <https://doi.org/10.1145/3571730>
- Jurafsky, D., & Martin, J. H. (2026). *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition with language models* (3.^a ed.). Manuscript en línea. <https://web.stanford.edu/~jurafsky/slp3/>
- Lecona-Valdespino, C. N., et al. (2023). *Machine Learning for Intent Classification in an Educational Chatbot*. Research in Computing Science.
- Lee, Y. F., Hwang, G. J., & Chen, P. Y. (2019). *Can chatbots help reduce the workload of administrative officers? Implementing and deploying FAQ chatbot service in a university*. Proceedings of the International Conference on Educational Technology.
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.
<https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html>
- MDN Web Docs. (2010). *Overview of HTTP*. Mozilla Developer Network. De
<https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Overview>
- Myers, G. J., Sandler, C., & Badgett, T. (2011). *The art of software testing* (3.^a ed.). Wiley.
- Nielsen, J. (1993). *Response time limits*. Nielsen Norman Group.
- Nielsen, J. (1993). *Usability engineering*. Academic Press.
- Nielsen, J. (2000). *Why you only need to test with 5 users*. Nielsen Norman Group.
- Nishisako, S., et al. (2025). *Reducing Hallucinations and Trade-Offs in Responses Generated by Artificial Intelligence Chatbots*.

- Norman, D. A. (2013). *The design of everyday things: Revised and expanded edition*. Basic Books.
- Okonkwo, C. W., & Ade-Ibijola, A. (2021). Chatbots applications in education: A systematic review. *Computers and Education: Artificial Intelligence*, 2, 100033. <https://doi.org/10.1016/j.caeai.2021.100033>
- Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9.^a ed.). McGraw-Hill Education.
- Rizou, S., Klonidis, E., Fascista, A., & others. (2023). *Efficient intent classification and entity recognition for university students' chatbot applications*.
- Rouhiainen, L. (2018). *Inteligencia artificial*. Madrid: Alienta Editorial.
- Russell, S. J., & Norvig, P. (2020). *Artificial intelligence: A modern approach* (4.^a ed.). Pearson.
- Socket.IO. (2017). *Introduction*. Socket.IO Documentation. De <https://socket.io/docs/v4/>
- Sommerville, I. (2016). *Software engineering* (10.^a ed.). Pearson.
- Ulrich, K., & Eppinger, S. (2013). *Diseño y desarrollo de productos* (5^a edición ed.). México, D.F.: Mc Graw Hill.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30. <https://arxiv.org/abs/1706.03762>
- World Wide Web Consortium. (2004). *Architecture of the World Wide Web, Volume One*. W3C Recommendation. <https://www.w3.org/TR/webarch/>