



UNIVERSIDAD AUTÓNOMA METROPOLITANA
Unidad Azcapotzalco

División de Ciencias Básicas e Ingeniería
Licenciatura en Ingeniería en Computación



**CyADBot: Asistente Virtual para la Mejora del Acceso a Información
Académica en CyAD**

Proyecto Tecnológico
Segunda versión
Trimestre 2025-Primavera

Alumno

Miguel Angel Ruiz Ruiz
2202000639
al2202000639@azc.uam.mx

Asesor

Dr. César Benavides Alvarez
Profesor Asociado
Departamento de Electrónica
cesarbenavides@azc.uam.mx

Coasesora

Dra. Angeles Belém Priego Sánchez
Profesora Titular
Departamento de Sistemas
abps@azc.uam.mx

13 de agosto de 2025

Declaratoria

En caso de que el Comité de Estudios de la Licenciatura en Computación apruebe la realización de la presente propuesta, otorgamos nuestra autorización para su publicación en la página de la División de Ciencias Básicas e Ingeniería.



Miguel Angel Ruiz Ruiz
Alumno

Dr. César Benavides Alvarez
Asesor

Dra. Angeles Belém Priego Sánchez
Coasesora

1. Introducción

En el contexto universitario, el acceso rápido y claro a la información académica es fundamental para que los estudiantes puedan realizar trámites, cumplir con fechas importantes y comprender los procesos institucionales. En la División de Ciencias y Artes para el Diseño (CyAD) de la Universidad Autónoma Metropolitana, Unidad Azcapotzalco, estas necesidades son constantes dentro de su comunidad estudiantil.

Sin embargo, los canales tradicionales de atención como el correo electrónico o las consultas presenciales suelen presentar demoras en la respuesta, saturación en periodos de alta demanda y falta de atención fuera del horario laboral.

Una alternativa funcional ante estas limitaciones es el uso de asistentes virtuales o chatbots, programas informáticos que simulan la conversación humana y pueden integrar técnicas de procesamiento de lenguaje natural para interpretar preguntas complejas y generar respuestas automatizadas [1]. Además, la adopción de un chatbot de preguntas frecuentes puede reducir hasta en un 80 por ciento las solicitudes repetitivas al personal administrativo, mejorando tanto la rapidez como la calidad del servicio [2].

Con base en lo anterior, este proyecto propone el desarrollo de CyADBot, un asistente virtual que brindará respuestas directas a las dudas más comunes del estudiantado de CyAD a través de un formato conversacional accesible desde el sitio web de la División.

Entre los beneficios esperados se encuentran mejorar la atención a estudiantes, reducir la carga de trabajo en áreas administrativas, garantizar disponibilidad permanente de información básica y fomentar el uso de soluciones digitales para optimizar los servicios educativos dentro de la División. De esta forma, CyADBot busca facilitar el acceso equitativo a la información académica y contribuir a la transformación digital de la División.

2. Justificación

El alumnado de CyAD enfrenta con frecuencia dificultades para acceder a información académica esencial relacionada con procesos, requisitos o fechas importantes. Estas dudas suelen surgir fuera del horario laboral, lo que limita el acceso a una atención oportuna y confiable.

CyADBot se propone como una solución efectiva para mejorar este panorama. A través de una herramienta interactiva accesible en cualquier momento, se facilita la obtención de respuestas claras, estructuradas y actualizadas sin depender de la intervención directa del personal administrativo.

Esto permite optimizar recursos, disminuir errores por desinformación y fomentar una comunicación más eficiente y accesible en la comunidad de CyAD mediante el aprovechamiento de tecnologías digitales sostenibles y fáciles de mantener.

Además de su funcionalidad práctica, CyADBot promueve el uso responsable de la tecnología en beneficio de la comunidad estudiantil al incorporar desde su diseño principios de seguridad y privacidad. Entre ellos, se incluyen buenas prácticas como evitar la solicitud o el almacenamiento de información personal sensible, garantizar que las respuestas provengan únicamente de fuentes académicas validadas y mantener un control interno que asegure que la interacción se mantenga pertinente, segura y libre de riesgos innecesarios.

Con estas medidas, se refuerza la confianza en el uso de la herramienta y se sientan las bases para futuros desarrollos institucionales con un enfoque responsable, seguro y sostenible.

3. Objetivos

3.1. Objetivo general

Implementar un asistente virtual que brinde atención automatizada a las dudas académicas más comunes del alumnado de CyAD mediante una base de datos de preguntas frecuentes y contenidos institucionales para mejorar la rapidez y la eficiencia en la atención sin depender del personal administrativo.

3.2. Objetivos específicos

- Diseñar una arquitectura de base de datos en **MySQL** que optimice la recuperación de información académica y minimice los tiempos de respuesta, mediante el uso de índices y consultas optimizadas, para gestionar de forma eficiente preguntas frecuentes y documentos institucionales.
- Desarrollar un módulo de interpretación de consultas en **Python** utilizando el framework **Flask** e implementando técnicas de **Procesamiento de Lenguaje Natural (PLN)** con la biblioteca **spaCy**, con el fin de identificar la intención del usuario y extraer palabras clave que permitan localizar información académica relevante.
- Implementar en **Python** un servicio de auditoría que registre automáticamente, en una tabla de **MySQL**, las consultas no resueltas de forma anonimizada, para facilitar su revisión periódica y la mejora continua de la base de conocimiento por parte del personal administrativo.
- Construir una interfaz conversacional en **HTML**, **CSS** y **JavaScript**, empleando **Fetch API** para la comunicación asíncrona con el servidor, que proporcione al usuario una experiencia clara e intuitiva desde el sitio web oficial de la División CyAD.

4. Trabajos relacionados

4.1. Desarrollo de un asistente virtual (ChatBot) para mejorar el acceso a la información recurrente por los estudiantes de Instituciones de Educación Superior [3].

Este proyecto desarrolló un ChatBot para estudiantes universitarios con el objetivo de agilizar consultas frecuentes sobre inscripciones y calificaciones en períodos de alta demanda. Se hizo una investigación descriptiva usando encuestas y entrevistas para conocer las necesidades. Se aplicó Scrum para avanzar de forma flexible y productiva. El bot se implementó como una app móvil tipo chat, creada con Ionic y Dialogflow de Google.

4.2. Diseño, Desarrollo e Implementación de una Asistente Virtual Para la Resolución de Dudas Sobre los Procesos Académicos de la Universidad Politécnica Salesina - Sede Cuenca Utilizando Inteligencia Artificial y Procesamiento del Lenguaje Natural [4].

Este proyecto desarrolló un asistente virtual para procesos académicos en universidades, con el fin de automatizar el sistema de peticiones, quejas, reclamos y sugerencias (PQRS) y mejorar la eficiencia en la atención de solicitudes. Utilizó IBM Watson Assistant para procesar lenguaje natural y comprender mejor las consultas. El ChatBot se implementó como una app móvil multiplataforma usando Ionic, facilitando la gestión y liberando recursos humanos para tareas más importantes.

4.3. UAMBot: ChatBot de asistencia para CAMVIA [5].

Este proyecto surgió por la contingencia del COVID-19, que incrementó la necesidad de herramientas para atender procesos de forma remota, especialmente en universidades. Se desarrolló un chatbot para ayudar a estudiantes de la Universidad Autónoma Metropolitana (UAM) unidad Azcapotzalco a resolver dudas sobre el uso del Campus Virtual Azcapotzalco (CAMVIA). La solución se basó en módulos de consulta, análisis y respuesta, que permiten recibir preguntas, analizarlas y generar respuestas precisas. Finalmente, se probó su funcionamiento e integración con Moodle, ya que CAMVIA se basa en esta plataforma.

4.4. Chatbot que facilita la información en la Facultad de Ingenierías de la Universidad Simón Bolívar [6].

Este proyecto implementó un asistente virtual (chatbot) para la Facultad de Ingenierías de la Universidad Simón Bolívar, con el fin de resolver consultas frecuentes de estudiantes que antes saturaban la secretaría. Usaron inteligencia artificial y lenguaje natural para entender preguntas y responder como una persona en tiempo real, logrando información más precisa y ahorro de tiempo para dedicarlo a tareas más importantes.

4.5. Chatbot para atención de procesos académicos de grado, caso de estudio Subdecanato FIS: análisis de herramientas de chatbot, diseño y construcción del backend [7].

Este proyecto desarrolló el backend de un chatbot para la Facultad de Ingeniería de Sistemas (FIS) de la Escuela Politécnica Nacional (EPN), buscando mejorar la atención a consultas académicas de forma más rápida y eficiente. Se analizaron herramientas, se diseñó una arquitectura clara y se implementó una API REST con Kanban como método de trabajo para lograr un sistema flexible y fácil de ampliar.

4.6. Diseño de prototipo de chatbot en Hangouts para la resolución de preguntas frecuentes de la Universidad Distrital Francisco José de Caldas [8].

Este proyecto implementó un chatbot desplegado en Google Hangouts para atender consultas frecuentes de la comunidad universitaria a través de cuentas institucionales. Utilizó la librería spaCy para el procesamiento de lenguaje natural y técnicas de machine learning, y se diseñó una arquitectura modular que facilita la administración y la actualización continua de la base de conocimiento.

Las similitudes y diferencias con el proyecto propuesto, se presentan en la Tabla 1.

Tabla 1. Comparación cualitativa de los trabajos relacionados con el proyecto.

Ref.	Similitudes	Diferencias
[3]	■ Orientado a estudiantes de educación superior	■ Estará implementado en una plataforma Web
	■ Ofrece información académica frecuente	■ Usará documentos institucionales para la búsqueda de respuestas
	■ Reduce carga administrativa	■ Registra consultas sin respuesta

Continuación de la Tabla 1

Ref.	Similitudes	Diferencias
[4]	<ul style="list-style-type: none"> ■ Resolver dudas académicas comunes ■ Automatizan procesos repetitivos ■ Usan procesamiento de lenguaje natural 	<ul style="list-style-type: none"> ■ Usará documentos institucionales para la búsqueda de respuestas ■ Se usara spaCy NLP básico en vez de IBM Watson Assistant
[5]	<ul style="list-style-type: none"> ■ Análisis de preguntas con NLP. ■ Chat para consultas académicas ■ Automatiza procesos frecuentes 	<ul style="list-style-type: none"> ■ Usará documentos institucionales para la búsqueda de respuestas ■ Registra consultas sin respuesta
[6]	<ul style="list-style-type: none"> ■ ChatBot para atención estudiantil ■ Mejora tiempos de respuesta ■ Análisis de preguntas con NLP. 	<ul style="list-style-type: none"> ■ No se usará IBM Watson ■ El backend es Python+Flask
[7]	<ul style="list-style-type: none"> ■ Gestionan dudas académicas estudiantiles ■ Usan Python + Flask backend ■ Uso de procesamiento lenguaje natural 	<ul style="list-style-type: none"> ■ Busca en documentos institucionales ■ Filtra preguntas irrelevantes automáticamente
[8]	<ul style="list-style-type: none"> ■ Chatbot para resolver preguntas frecuentes universitarias ■ Usa spaCy para procesamiento de lenguaje natural ■ Arquitectura modular con módulo de actualización 	<ul style="list-style-type: none"> ■ Se desplegara en una pagina Web ■ Tendrá un registro automático de preguntas no resueltas.

5. Descripción técnica

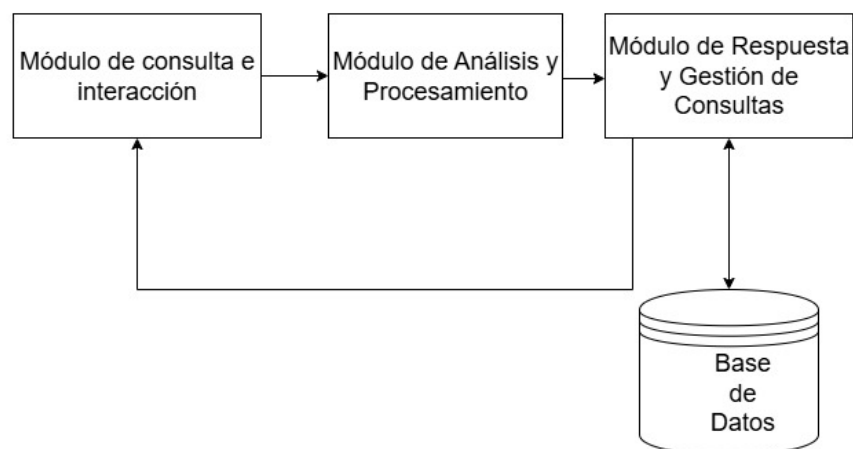


Figura 1. Diagrama de bloques del proyecto

La arquitectura de CyADBot se compone de tres módulos principales que colaboran de forma integrada para ofrecer una experiencia conversacional completa, como se ilustra en la Figura 1. Cada módulo desempeña un papel clave en el flujo de atención automatizada.

El **Módulo de Consulta e Interacción** es el punto inicial de contacto con el usuario. Se implementa en el sitio web institucional de la División CyAD mediante un botón flotante que, al activarse, despliega una ventana de chat. Desde esta interfaz, desarrollada con **HTML** [9], **CSS** [10] y **JavaScript** [11], el estudiante puede enviar consultas en lenguaje natural. La comunicación con el servidor se realiza usando **Fetch API** [12], enviando los mensajes en formato **JSON** [13] a un servidor **Flask** [14] sin recargar la página. Este módulo también integra elementos visuales como indicadores de estado (por ejemplo, “escribiendo...”) para mantener una experiencia más cercana a la interacción humana.

El **Módulo de Análisis y Procesamiento** recibe la consulta y la procesa con un pipeline de **spaCy** [15], que incluye la limpieza del texto, la normalización a minúsculas y la eliminación de *stop words*. Posteriormente, el sistema aplica reconocimiento de entidades nombradas para identificar términos clave relacionados con el contexto académico (como códigos de materia, trámites o fechas relevantes). Un clasificador supervisado determina la intención de la consulta, asignándola a categorías predefinidas. Además, este módulo incluye un filtro de contenido que descarta mensajes irrelevantes o ajenos al ámbito académico, evitando su almacenamiento. Para proporcionar contexto adicional, se recurre a un glosario y a información obtenida de documentos institucionales oficiales (archivos PDF y páginas HTML) previamente indexados, asegurando que la información provenga de fuentes internas y confiables.

Finalmente, el **Módulo de Respuesta y Gestión de Consultas** accede a una base de datos **MySQL** [16] que contiene una tabla de preguntas frecuentes (FAQs) y un repositorio indexado de documentos institucionales. Primero, el sistema intenta encontrar coincidencias exactas o aproximadas en las FAQs utilizando consultas SQL parametrizadas y operadores `LIKE`. Si no se obtiene una respuesta satisfactoria, se consulta el índice de documentos para extraer fragmentos relevantes. En todos los casos, la respuesta se genera en lenguaje natural, manteniendo un formato conversacional que explique la información solicitada sin limitarse a mostrar enlaces. Para garantizar la seguridad y prevenir que código malicioso se ejecute dentro del navegador del usuario, se utiliza la biblioteca **MarkupSafe** [17], la cual evita que el texto que escribe el usuario pueda alterar o dañar la página. Toda la comunicación entre cliente y servidor se realiza bajo el protocolo **HTTPS** [18] para cifrar los datos transmitidos.

En conjunto, estos módulos permiten que CyADBot funcione como un asistente virtual capaz de dialogar con el usuario, interpretando sus preguntas y entregando respuestas claras, contextuales y redactadas en lenguaje natural, evitando limitarse a mostrar enlaces. Cuando sea pertinente, las respuestas incluirán vínculos de referencia a documentos oficiales de la División CyAD, siempre provenientes de fuentes internas como archivos PDF y páginas obtenidos del repositorio institucional y previamente indexados para su consulta.

6. Especificación técnica

El desarrollo del proyecto se llevará a cabo utilizando un conjunto de herramientas y tecnologías que garanticen un funcionamiento eficiente y seguro. Se empleará **Python** [19] como lenguaje principal; el framework **Flask** [14] para el servidor web y la **API REST** [20]; se utilizarán técnicas de Procesamiento de Lenguaje Natural (PLN) implementadas con la biblioteca **spaCy** [15]; **MySQL** [16] como sistema de gestión de bases de datos relacional; **HTML** [9], **CSS** [10] y **JavaScript** [11] para la construcción de la interfaz de usuario; **Fetch API** [12] sobre **HTTPS** [18] para la comunicación asíncrona cliente-servidor en formato **JSON** [13]; y **MarkupSafe** [17] para la saniti-

zación de entradas. El entorno de desarrollo será **Visual Studio Code** [21] con control de versiones en **Git** [22]. No se utilizarán servicios remotos externos ni conexiones a dispositivos externos.

La interfaz conversacional consiste en un botón flotante en la esquina inferior derecha del sitio web de la División CyAD, que despliega una ventana de chat embebida donde cualquier usuario (sin iniciar sesión) puede redactar preguntas en lenguaje natural. Durante la interacción, el sistema muestra estados de envío, respuesta y posibles errores de conexión.

Para cada módulo del sistema se establecen los siguientes criterios de finalización:

- El módulo de consulta e interacción debe abrir correctamente la ventana de chat desde el botón flotante, enviar los mensajes en formato JSON al servidor y mostrar las respuestas o mensajes de error sin necesidad de recargar la página.
- El módulo de análisis y procesamiento debe limpiar la entrada con el pipeline de spaCy, detectar entidades nombradas, clasificar la intención en las categorías definidas con una precisión mínima del 85 % sobre un conjunto de pruebas, y aplicar un filtro de contenido que descarte consultas no pertinentes antes de registrarlas.
- El módulo de respuesta y gestión de consultas debe realizar correctamente las búsquedas en la tabla de preguntas frecuentes y en el índice de documentos institucionales, registrar las consultas no resueltas y construir respuestas en lenguaje natural, presentadas en HTML, que mantengan un formato conversacional y no se limiten a mostrar enlaces.

Se implementarán medidas básicas de protección de datos mediante la sanitización de entradas y el uso exclusivo de HTTPS para garantizar la confidencialidad e integridad de las interacciones. CyADBot no almacena información personal, permite consultas anónimas y no sustituye los canales formales de atención institucional, sino que los complementa como herramienta de apoyo para resolver dudas frecuentes de forma ágil y segura.

Adicionalmente, se realizarán pruebas de carga, estrés y vulnerabilidad. Las primeras evaluarán el rendimiento con múltiples usuarios simultáneos; las segundas someterán el sistema a una demanda superior a la habitual para comprobar su límite y capacidad de recuperación; y las últimas verificarán que solo se permita conexión segura, que se limiten solicitudes repetidas desde un mismo origen, que los mensajes de error no revelen información interna, y que el texto mostrado en el chat se procese siempre como contenido seguro.

Al concluir el proyecto de integración se entregará a la Coordinación de Estudios de Ingeniería en Computación una carpeta digital que incluirá el reporte final del proyecto en un archivo PDF (sin restricciones)¹, el código fuente del proyecto en un archivo comprimido (sin restricciones)². Además, la sección de apéndices del reporte final contendrá al menos un listado del código fuente desarrollado.

7. Calendario de actividades

Las actividades a realizar durante el Trimestre 2025-P en la UEA Proyecto de Integración en Ingeniería en Computación I (1100113) se presentan en la Tabla 2.

¹Debe poder visualizarse sin solicitar contraseña

²Debe poder descomprimirse sin solicitar contraseña

Tabla 2. Listado de actividades a realizar durante el Trimestre 2025-P.

No.	Actividad	Horas	Entregable
1	Preparar el repositorio de documentos institucionales y FAQs para indexación.	18	Colección de documentos organizados y validados.
2	Modelar la base de datos MySQL y generar los scripts de inicialización (ER y tablas).	18	Diagrama ER y scripts SQL.
3	Desarrollar el módulo NLP con spaCy: limpieza de texto, extracción de entidades y clasificación de intenciones.	30	Código del módulo NLP con su documentación.
4	Implementar la API REST en Flask, que incluya endpoints para consulta, filtrado y sanitización.	20	Servicio API funcional y documentado.
5	Construir la interfaz web del chatbot: botón flotante, ventana de chat y comunicación con Fetch API.	20	Prototipo integrado en el sitio web de CyAD.
6	Configurar la lógica de respuestas: consultas a FAQs, búsqueda en índice documental y registro de “no resueltas”.	22	Código de la gestión de respuestas y casos de prueba.
7	Establecer el entorno de pruebas unitarias, de integración y gestionar el control de versiones con Git.	18	Entorno de pruebas operativo e informe de resultados.
8	Ejecutar pruebas de seguridad y rendimiento (carga, estrés y vulnerabilidades) y aplicar ajustes.	20	Informe de pruebas técnicas y recomendaciones.
9	Refactorizar y optimizar el código según los resultados de las pruebas técnicas y de integración.	22	Código optimizado.
10	Redactar el reporte final del proyecto de integración.	10	Reporte final en PDF
Total de horas:		198	

8. Factibilidad y Estimación de Costos

8.1. Factibilidad Técnica

El desarrollo de CyADBot es factible debido a que se cuenta con experiencia en el desarrollo web utilizando HTML y CSS, así como en la implementación y gestión de bases de datos relacionales con MySQL. Aunque no se tiene amplia experiencia en Python, JavaScript ni procesamiento de lenguaje natural (PLN), se está tomando la capacitación necesaria para cubrir estas áreas y así cumplir con los objetivos en los tiempos establecidos.

El proyecto se basa en herramientas y tecnologías de código abierto o de uso común, por lo que no requiere recursos especiales ni software con licencia.

Por lo anterior, dados los conocimientos, recursos y planificación actuales, se considera que las

actividades detalladas en el calendario se realizarán en tiempo y forma conforme al plan propuesto.

8.2. Factibilidad Operativa

Una vez finalizado, CyADBot ofrecerá una interfaz gráfica amigable e intuitiva que facilitará la interacción de los usuarios con el sistema. Su disponibilidad continua permitirá que el alumnado resuelva sus dudas académicas comunes de manera inmediata, reduciendo la carga administrativa y mejorando la eficiencia en la atención.

El diseño modular y escalable del sistema facilitará futuras actualizaciones y adaptaciones a nuevas necesidades o tecnologías. Asimismo, se espera que la aceptación por parte de la comunidad estudiantil sea alta, dado que el chatbot utiliza un formato conversacional familiar y accesible desde la página web oficial de CyAD.

De esta forma, CyADBot contribuirá a optimizar los servicios educativos y la comunicación dentro de la División, favoreciendo una experiencia más satisfactoria para estudiantes y personal administrativo. Además, gracias a su arquitectura flexible, se contempla la posibilidad de replicar esta solución en otras Divisiones académicas o incluso en otras universidades, lo que representa un valor agregado significativo para su escalabilidad y aprovechamiento institucional.

9. Estimación de Costos

La estimación de costos del proyecto se muestra en la Tabla 3.

La mayoría de los recursos ya están disponibles por parte del alumno, por lo que los costos son estimaciones de valor de mercado. No se requiere inversión institucional directa para su ejecución.

Tabla 3. Estimación de costos del proyecto.

Descripción	Costo mensual (MXN)	Costo trimestral (MXN)
Laptop HP 255 G10 Ryzen 5 7530u 16GB RAM 512GB SSD 15.6 [23]	\$8,998.00	\$8,998.00 (Pago único)
Servidor de hosting web básico [24]	\$219.99	\$659.97
Salario de desarrollador de Inteligencia Artificial [25]	\$27,000.00	\$81,000.00
Salario de desarrollador FullStack [26]	\$15,000.00	\$45,000.00
Internet Izzi 150 megas [27]	\$389.00	\$1,167.00
Electricidad [28]	\$170.49	\$511.47
Costo total	\$51,777.48	\$137,336.44

Los asesores se responsabilizan de guiar al alumno y de que todos los recursos mencionados en la Factibilidad Técnica estarán disponibles para el alumno, de modo que el proyecto de integración se pueda concluir en tiempo y forma.

Dr. César Benavides Alvarez
Asesor

Dra. Angeles Belém Priego Sánchez
Coasesora

Referencias

- [1] IBM. (2025, Febrero) ¿qué es un chatbot? [Online]. Available: <https://www.ibm.com/mx-es/think/topics/chatbots>
- [2] LookFor.AI. (2025, Julio) Faq chatbot: beneficios, casos de uso y precios. [Online]. Available: <https://www.lookfor.ai/es/blog/faq-chatbot-benefits-use-cases-and-prices>
- [3] O. León Granizo and M. León Granizo, “Desarrollo de un asistente virtual (chatbot) para mejorar el acceso a la información recurrente por los estudiantes de instituciones de educación superior,” *Ecuad. Sci. J*, vol. 4, no. 2, pp. 111–116, sep 2020.
- [4] P. M. Gonzalez Arias, “Diseño, desarrollo e implementación de una asistente virtual para la resolución de dudas sobre los procesos académicos de la universidad politécnica salesiana – sede cuenca utilizando inteligencia artificial y procesamiento de lenguaje natural,” Trabajo de Titulación, Universidad Politécnica Salesiana, Cuenca, Ecuador, 2022. [Online]. Available: <https://dspace.ups.edu.ec/handle/123456789/22027>
- [5] A. León Vargas, “Uambot: Chatbot de asistencia para camvia,” Proyecto terminal, División de Ciencias Básicas e Ingeniería, Universidad Autónoma Metropolitana Azcapotzalco, México, 2022.
- [6] G. Casseres, J. Cuao, M. Londoño, L. Obredor, S. Orozco, and P. Sánchez, “Dchatbot que facilita la información en la facultad de ingenierías de la universidad simón bolívar,” *Investigación y Desarrollo en TIC*, vol. 9, no. 2, pp. 18–25, jul 2018. [Online]. Available: <https://revistas.unisimon.edu.co/index.php/identific/article/view/3480>
- [7] A. Lozano Estrada, “Chatbot para atención de procesos académicos de grado, caso de estudio subdecanato fis : análisis de herramientas de chatbot, diseño y construcción del backend,” Trabajo de Integración Curricular, Facultad de Ingeniería en Sistemas Informáticos y de Computación (FIS), Escuela Politecnica Nacional, Ecuador, 2024. [Online]. Available: <http://bibdigital.epn.edu.ec/handle/15000/25514>
- [8] Mass López, Pilar Stephany and Orjuela Castillo, Juan José, “Diseño de prototipo de Chatbot en Hangouts para la resolución de preguntas frecuentes de la Universidad Distrital Francisco José de Caldas,” Tesis de Especialización en Ingeniería de Software, Universidad Distrital Francisco José de Caldas, 2020, director: Meza Álvarez, Joaquín Javier. [Online]. Available: <https://repository.udistrital.edu.co/items/f94ae73f-afc0-4d41-ae29-e0885876840b>
- [9] World Wide Web Consortium (W3C), “Html standard,” 2025. [Online]. Available: <https://html.spec.whatwg.org>
- [10] —, “Cascading style sheets (css) — w3c,” 2025. [Online]. Available: <https://www.w3.org/Style/CSS>
- [11] Mozilla Developer Network, “Javascript guide — mdn web docs,” 2025. [Online]. Available: <https://developer.mozilla.org/docs/Web/JavaScript>

- [12] —, “Fetch api — mdn web docs,” 2025. [Online]. Available: https://developer.mozilla.org/docs/Web/API/Fetch_API
- [13] T. Bray, “The javascript object notation (json) data interchange format,” 2014. [Online]. Available: <https://www.rfc-editor.org/rfc/rfc7159>
- [14] Pallets Projects, “Flask documentation,” 2025. [Online]. Available: <https://flask.palletsprojects.com>
- [15] Explosion AI, “spacy — industrial-strength natural language processing in python,” 2025. [Online]. Available: <https://spacy.io>
- [16] Oracle Corporation, “Mysql 8.0 reference manual,” 2025. [Online]. Available: <https://dev.mysql.com/doc>
- [17] Pallets Projects, “Markupsafe documentation,” 2025. [Online]. Available: <https://markupsafe.palletsprojects.com>
- [18] Rescorla, Eric, “Rfc 2818 — http over tls,” 2000. [Online]. Available: <https://www.rfc-editor.org/rfc/rfc2818>
- [19] Python Software Foundation, “Python programming language,” 2025. [Online]. Available: <https://www.python.org>
- [20] R. T. Fielding, “Architectural styles and the design of network-based software architectures,” 2000. [Online]. Available: https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm
- [21] Microsoft Corporation, “Visual studio code,” 2025. [Online]. Available: <https://code.visualstudio.com>
- [22] Software Freedom Conservancy, “Git — distributed version control system,” 2025. [Online]. Available: <https://git-scm.com>
- [23] HP. (2025, jul) Laptop hp 255 g10, ryzen 5 7530u, 16gb ram, 512gb ssd, 15.6” fhd display. [Online]. Available: https://articulo.mercadolibre.com.mx/MLM-2257133259-laptop-hp-255-g10-ryzen-5-7530u-16gb-ram-512gb-ssd-156-_JM
- [24] Hostinger. (2025, Julio) Precios de hosting web en México. [Online]. Available: <https://www.hostinger.com/mx/precios>
- [25] Glassdoor. (2025, Agosto) Sueldos para desarrollador de inteligencia artificial en ciudad de México. [Online]. Available: https://www.glassdoor.com.mx/Sueldos/ciudad-de-m%C3%A9xico-desarrollador-de-inteligencia-artificial-sueldo-SRCH_IL.0,16_IM1176_KO17,57.htm
- [26] —. (2025, Agosto) Sueldos para desarrollador full stack jr en México. [Online]. Available: https://www.glassdoor.com.mx/Sueldos/desarrollador-full-stack-jr-sueldo-SRCH_KO0,27.htm
- [27] Izzi. (2025, Julio) Paquetes de internet y telefonía para tu hogar. [Online]. Available: <https://www.izzi.mx/paquetes/goo/br/2p/>
- [28] CFE Contigo. (2025, Julio) Simulador de consumo de luz cfe. [Online]. Available: <https://appcfecontigo.com.mx/simulador-consumo-cfe/>