

Aplicación de la Metodología OOWS y el Marco Ágil Scrum en el Diseño del “Chat Corporativo”

Application of the OOWS Methodology and the Agile Framework Scrum in the Design of the “Corporate Chat”

Miriam López Sanluis^{ID}, Elizabeth Cuatecontzi Cuahutle^{ID},
María Guadalupe Medina Barrera^{ID}, María Janaí Sánchez Hernández^{ID}
y José Juan Hernández Mora^{ID}

División de Estudios de Posgrado e Investigación, Tecnológico Nacional de México, Instituto Tecnológico de Apizaco
Av. Instituto Tecnológico 418, San Andrés Ahuashuatepec, 90491 Tzompantepec, Tlaxcala, México
[*m23370005@apizaco.tecnm.mx](mailto:m23370005@apizaco.tecnm.mx)

PALABRAS

CLAVE:

Chat corporativo, metodología OOWS, Ingeniería de software, SCRUM, lenguaje de modelado unificado (UML)

RESUMEN

Se implementó una solución de mensajería empresarial segura y eficiente para agilizar la comunicación interna de una organización. Se aplicó el modelo OOWS para el diseño conceptual y el marco de trabajo ágil SCRUM para la gestión de su desarrollo. OOWS permitió obtener una representación detallada de los requerimientos funcionales y no funcionales del sistema, mientras que SCRUM garantizó un desarrollo iterativo, adaptable y en el tiempo establecido. Para la aplicación se implementaron controles de acceso basados en roles para asegurar la seguridad y confidencialidad de la información. La arquitectura del sistema se basó en un modelo vista-controlador, el desarrollo se realizó con las tecnologías Tailwind CSS, Laravel y MySQL como gestor de base datos. La aplicación ofrece funcionalidades avanzadas como gestión de usuarios, creación de grupos y salas de chat, así como la posibilidad de compartir archivos de forma controlada, garantizando la confidencialidad de la información. Gracias a esta nueva herramienta, se logró agilizar la comunicación, facilitar la colaboración entre equipos y garantizar la seguridad de la información compartida, mejorando así la productividad y satisfacción en la empresa. Además, se identificaron oportunidades de mejora, como la necesidad de escalar el sistema para integrar más usuarios. Como trabajo futuro, se propuso migrar el chat a una plataforma Cloud Computing para aprovechar las ventajas como la escalabilidad, alta disponibilidad y seguridad que estas tecnologías ofrecen.

KEYWORDS:

Corporate chat, OOWS methodology, Software engineering, SCRUM, Unified Modeling Language (UML)

ABSTRACT

A secure and efficient enterprise messaging solution was implemented to streamline internal communication within an organization. The OOWS model was applied for the conceptual design, and the SCRUM agile framework was used for managing its development. OOWS provided a detailed representation of the system's functional and non-functional requirements, while SCRUM ensured an iterative, adaptable development process within the established timeframe. Role-based access controls were implemented in the application to ensure the security and confidentiality of the information. The system architecture was based on a Model-View-Controller (MVC) pattern, and development was carried out using Tailwind CSS, Laravel, and MySQL as the database management system. The application offers advanced features such as user management, the creation of groups and chat rooms, and the ability to share files in a controlled manner, ensuring information confidentiality. Thanks to this new tool, communication was streamlined, team collaboration was facilitated, and the security of shared information was ensured, ultimately improving productivity and satisfaction within the company. Additionally, improvement opportunities were identified, such as the need to scale the system to support more users. As future work, it was proposed to migrate the chat application to a Cloud Computing platform to take advantage of benefits such as scalability, high availability, and enhanced security offered by these technologies.

• Recibido: 31 de agosto de 2024 • Aceptado: 15 de diciembre de 2024 • Publicado en línea: 1 de febrero de 2025



1. INTRODUCCIÓN

En ingeniería de software, un sistema web se define como cualquier programa que los usuarios pueden utilizar accediendo a un servidor web por Internet o red interna, usando un navegador para interactuar con él [1].

Además, un sistema web es clave para la comunicación con clientes y proveedores de servicios. Las plataformas permiten obtener información sobre productos y servicios, solicitar cotizaciones, realizar contacto directo, dar información empresarial, buscar datos, ventas y marketing, presentación global y muchas funciones que facilitan una interacción con los clientes, que buscan mayor comodidad [2].

La comunicación efectiva en las organizaciones es esencial para el éxito y la eficiencia operativa. Los sistemas de chat han surgido como herramientas vitales para facilitar la interacción entre las personas de una organización, promoviendo la colaboración y la rápida toma de decisiones.

Esta propuesta se centró en el desarrollo de un chat corporativo para mejorar la comunicación interna en una organización dedicada al desarrollo de software, mediante la gestión de usuarios, grupos, y salas, además de garantizar la confidencialidad y seguridad al compartir documentos propios de la empresa.

El sistema propuesto permitió la edición y eliminación de usuarios y salas, así como el intercambio de recursos esenciales como documentos e imágenes. Una característica clave de esta solución fue implementar controles de acceso basados en roles, lo que garantizó que los usuarios accedan solo a la información y a las salas pertinentes a su rol específico. Este enfoque no solo optimizó el rendimiento del sistema, sino que también fortaleció la seguridad y la confidencialidad de la información compartida.

Para su realización se planteó aprovechar las herramientas disponibles en la ingeniería de software para optimizar su diseño y desarrollo. En la primera sección se presentan ejemplos de algunos desarrollos web de sistemas de chat, en la segunda se muestra la aplicación de la metodología OOWS para el diseño conceptual del sistema y la aplicación del marco de trabajo ágil SCRUM para gestionar el desarrollo del sistema y en la tercera se presentan los resultados obtenidos.

2. TRABAJOS RELACIONADOS

Vinay Kumar et al., [3], desarrollaron una aplicación de chat en Internet basada en tecnologías modernas de desarrollo web como WebSocket y Spring Boot. Esta aplicación, destinada tanto a usuarios comunes como a entornos corporativos, facilita la comunicación en tiempo real sin necesidad de programas clientes adicionales, utilizando la transmisión de datos punto a punto para chats de voz y video. Emplea multithreading para gestionar múltiples clientes simultáneamente, la aplicación permite la mensajería de texto, chats grupales y la transferencia de datos sin restricciones de tamaño. Los autores subrayan que el protocolo WebSocket permite una comunicación bidireccional eficiente entre el servidor y el cliente, mejorando así la experiencia del usuario y la funcionalidad de la aplicación.

Cherckesova et al., [4], presentaron un software complejo que combina un chat corporativo seguro con un sistema de prevención de fugas de datos (DLP) para proteger la información en empresas industriales. Este sistema, desarrollado utilizando C#, PHP y JavaScript, fue implementado sobre una base de datos MySQL y está destinado a ser utilizado por empleados dentro de una organización para la comunicación y el intercambio de archivos relacionados con tareas laborales. El sistema incluye un monitor de seguridad que detecta

y previene amenazas internas mediante la búsqueda de palabras y frases prohibidas en las comunicaciones de los empleados. Los resultados mostraron una mejora significativa en la detección de incidentes y la prevención de fugas de información, destacando la relevancia de proteger la información no solo de amenazas externas, sino también de intrusos internos.

Kumar et al., [5], diseñaron una aplicación de chat web multiusuario que utiliza Node.js y Socket.io, esta permite establecer salas, entrar en ellas para discusión y salir de ellas sin dejar ningún dato porque no es destinado a mantener registros de las conversaciones de los clientes. La aplicación se implementa y se centra en una interfaz de usuario simple que permite el servicio de mensajes de texto o chat para amigos o compañeros de trabajo.

El estudio de Dong et al., [6], examina el impacto del chat en vivo en las decisiones de compra en el contexto de las plataformas de subcontratación en línea. El estudio ofrece implicaciones prácticas para el diseño y uso de herramientas de chat en vivo en plataformas de subcontratación en línea. Al comprender cómo las diferentes dimensiones de las señales afectan a los consumidores con diferentes niveles de experiencia, las plataformas pueden personalizar sus estrategias de comunicación para satisfacer mejor las necesidades de los diversos segmentos de clientes. Este trabajo amplía el conocimiento sobre cómo las interacciones de chat en vivo afectan las decisiones de compra en plataformas de subcontratación en línea. Al integrar tanto las dimensiones afectivas como informativas de las señales de chat y considerar la experiencia previa del consumidor, proporciona una base sólida para futuras investigaciones y desarrollo de estrategias efectivas de comunicación en plataformas digitales.

Daniel & Cabot [7], presentan un marco basado en modelos llamado Xatkit, diseñado

para la creación y gestión de chatbots de manera independiente de la plataforma. El estudio destaca la evolución del marco Xatkit, que inicialmente fue desarrollado para simplificar la implementación de chatbots sin necesidad de un profundo conocimiento técnico sobre las plataformas de destino. Los resultados muestran que Xatkit ha facilitado la creación de chatbots complejos mediante un lenguaje específico de dominio (DSL) y un motor de tiempo de ejecución que automatiza la lógica de conversación definida. Esta experiencia es útil para otros proyectos que buscan desarrollar soluciones basadas en modelos a nivel industrial.

Los trabajos estudiados permitieron comprender como podría funcionar y estructurarse un chat para la empresa que requería la aplicación, pero integrando herramientas metodológicas y tecnologías al alcance de la misma.

3. METODOLOGÍA

Para el desarrollo del chat corporativo propuesto, se empleó el modelo OOWS (Object-Oriented Web Solutions) para el modelado conceptual y la metodología ágil SCRUM para la gestión del desarrollo del sistema.

La combinación de OOWS y SCRUM proporcionó una estructura y flexible para el desarrollo del chat corporativo, garantizando tanto un diseño conceptual sólido como una implementación adaptativa y eficiente. Esta unión permitió abordar eficazmente los desafíos técnicos y organizativos del proyecto, solucionando las necesidades de comunicación interna de la organización.

3.1. Modelado Conceptual con OOWS

OOWS (Object Oriented Web System) es una metodología de modelado específica para el desarrollo de sistemas web. Se caracteriza por su enfoque en la creación de interfaces que combinan texto y elementos gráficos, y

por su énfasis en el análisis profundo de la navegación del sitio web [8].

La metodología permite capturar los requisitos funcionales y no funcionales del sistema, garantizando una comprensión integral y precisa de las necesidades del usuario y del negocio [9].

El modelo OOWS se basa en el método basado en modelos orientados a objetos (OO-Method) y presenta tres modelos que se elaboran de manera secuencial.

1. **Modelo de usuarios:** Clasifica los diferentes tipos de usuarios que interactúan con el sistema y explica las relaciones entre ellos.
2. **Modelo navegacional:** Establece la visibilidad del sistema en términos de datos y funcionalidades, así como las rutas válidas dentro de la estructura del sistema para cada tipo de usuario, la especificación comienza desde una perspectiva general y se va detallando específicamente para cada tipo de usuario.
3. **Modelo de presentación:** Define los requisitos de presentación para los elementos identificados por el modelo de navegación, manteniendo una conexión estrecha con este último, este modelo utiliza los contextos de navegación para describir la interacción del usuario con el sistema [9].

La aplicación de OOWS en este proyecto incluyó la creación de diagramas de casos de uso, diagramas de clases y modelos de navegación que detallan la interacción de los usuarios con el sistema, así como la estructura y comportamiento de los componentes del chat corporativo.

Análisis de requerimientos

El sistema de chat corporativo debe cumplir con los siguientes requerimientos desde tres perspectivas: Usuario, Agente de Soporte, y Administrador.

Usuario:

- Funcionalidad Básica: Recibir mensajes de soporte, no iniciar conversaciones.
- Interfaz requerida: Ventana adaptable, notificaciones visuales y sonoras, historial de chat con fecha y hora, campo para escribir y enviar mensajes, y un apartado para compartir imágenes.
- Diseño: Adaptable a diferentes tamaños de pantalla (PC y Tablet), con opción de expandir a pantalla completa.

Agente de Soporte:

- Funcionalidad Avanzada: Iniciar conversaciones, buscar usuarios específicos (directorio), verificar estado de los usuarios, finalizar conversaciones y recibir notificaciones.
- Herramientas requeridas: Filtro de usuarios, etiqueta de estado del usuario, botón para finalizar chat con confirmación, y cierre automático de chats inactivos.
- Interfaz requerida: Similar al usuario, pero con la adición de un botón para finalizar el chat y la posibilidad de compartir documentos.

Administrador:

- Gestión Integral: Agregar usuarios a salas de chat, activar/desactivar chats por grupo, crear y editar usuarios, grupos y salas, configurar permisos y roles, administrar archivos, y generar reportes estadísticos.
- Estadísticas: Histórico de chats detallado, incluyendo fecha, hora, usuarios involucrados y tiempo de atención.

En resumen, se busca implementar un sistema de chat que sea fácil de usar para los usuarios, eficiente para los agentes de soporte y altamente configurable para los administradores.

Aplicación del modelado OOWS

Se muestran algunos de los diagramas obtenidos para el modelo conceptual del sistema.

A. Diagramas de caso de uso

El modelado de los casos de uso permitió modelar el comportamiento funcional general del sistema, y a la especificación de las funcionalidades que ofrece a sus usuarios y cómo se accede a ellas.

Agente de soporte: Es un empleado de la empresa que hace uso del sistema de chat para brindar soporte a los usuarios en la empresa. Los agentes de soporte comienzan su interacción con el usuario al "Iniciar chat", lo que establece la primera conexión directa. Después, pueden "Enviar mensaje", lo que les permite dar respuestas e información requerida por el usuario de la empresa. fluido y continuo de comunicación.

Simultáneamente, el agente de soporte también "Recibe mensaje", permitiendo un intercambio. Finalmente, una vez resueltas todas las consultas, el agente de soporte procede a "Cerrar chat", marcando el fin de la sesión de soporte. Estas acciones forman la estructura básica del servicio de atención, asegurando que cada interacción sea manejada de manera eficiente y efectiva.

El diagrama de casos de uso de la Figura 1 identifica los actores principales en el contexto del sistema web desde el Agente de soporte, detalla las acciones requeridas para cada uno y una visión comprehensiva de las operaciones necesarias para el funcionamiento del sistema.

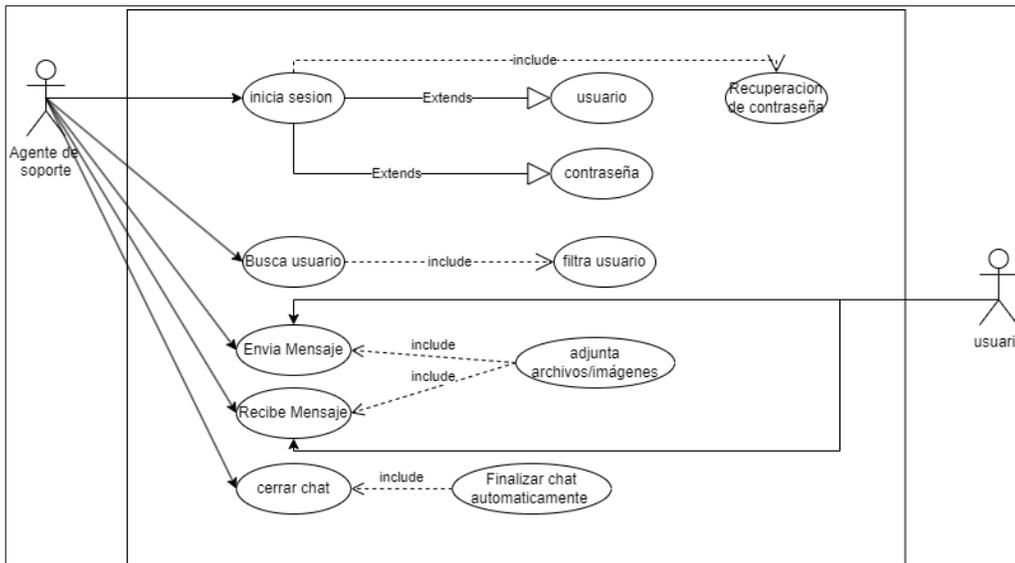


Figura 1. Diagrama de casos de uso desde la perspectiva del agente de soporte de la aplicación Web (Elaboración propia).

B. Modelo de clases

El diagrama general de clases del sistema se presenta en la Figura 2, ilustra la estructura completa del sistema, incluyendo sus clases, atributos y métodos que conforman la aplicación web.

A continuación, se describen las principales clases y sus relaciones con el agente de soporte:

Usuario: Los usuarios son individuos externos que interactúan con el sistema de chat para comunicarse con los agentes de soporte. Estos usuarios pueden iniciar chats, enviar y recibir mensajes y cerrar chats. Su interacción con los agentes es crucial para proporcionar soporte y asistencia en tiempo real.

Agente de soporte: Los agentes de soporte son los empleados de soporte que gestionan las consultas y problemas presentados por los

usuarios. Pueden buscar usuarios externos, iniciar chats con ellos y gestionar toda la comunicación mediante el envío y recepción de mensajes.

Los agentes de soporte desempeñan un papel esencial en la interacción con los usuarios, proporcionando soluciones y asistencia efectiva.

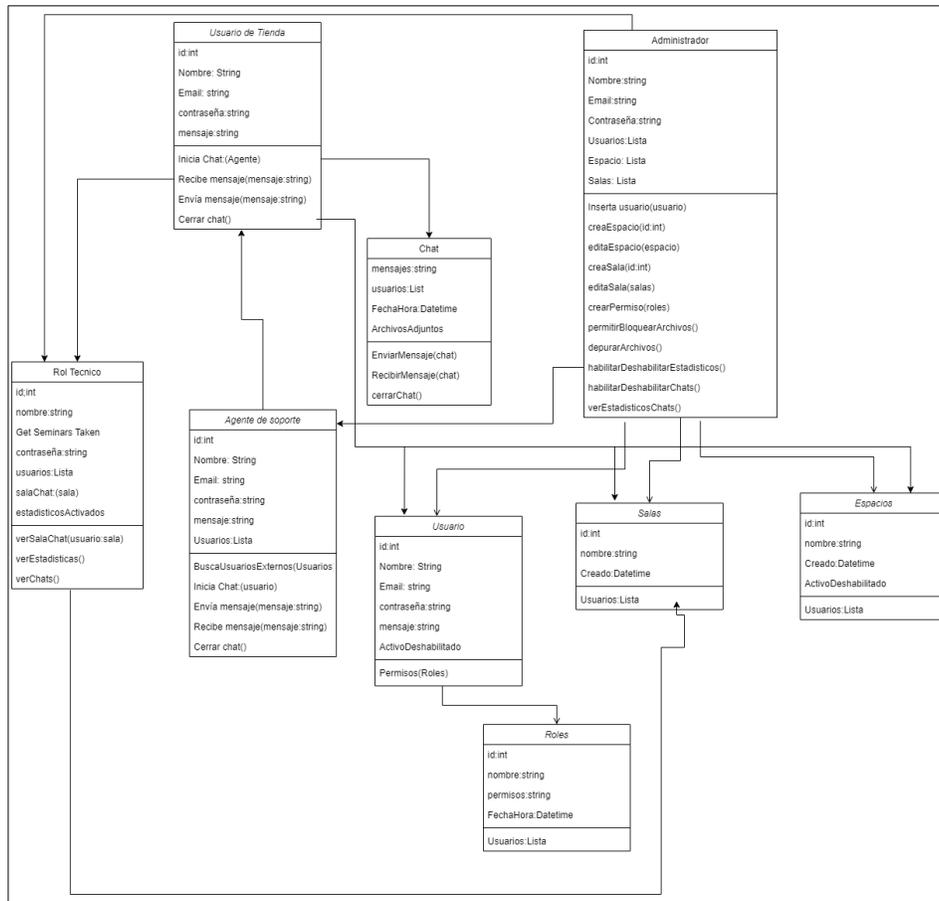


Figura 2. Diagrama de clases general de la aplicación Web (Elaboración propia).

Chat: La clase Chat es la pieza central del sistema, representando la conversación entre los usuarios, tanto agentes de soporte como usuarios. Esta clase contiene los mensajes intercambiados, los usuarios participantes, la fecha y hora de los mensajes, y los archivos adjuntos. Además, incluye métodos para enviar, recibir y cerrar chats, facilitando así la gestión y el flujo de la comunicación entre los usuarios y los agentes de soporte.

Relaciones:

- Usuario interactúa con Rol Técnico para obtener soporte técnico.

- Agente de soporte interactúa con Usuario a través de la clase Chat para gestionar la comunicación.
- Usuario tiene asignados Roles que definen sus permisos y accesos dentro del sistema.
- Salas y Espacios contienen listas de Usuarios y son administrados por Administrador.

En resumen, los agentes de soporte interactúan con los usuarios a través de la clase Chat, gestionando las consultas y problemas presentados por estos usuarios externos. Esta interacción se realiza mediante el inicio de chats, el envío y recepción de mensajes, y el cierre de las conversaciones una vez resueltas las consultas.

C. Modelo de procesos

El proceso general que un usuario debe seguir para comunicarse con otros colaboradores de la empresa se puede representar gráficamente. Esto incluye desde el inicio de la sesión, la división de actividades según los roles y permisos asignados al administrador, agente y usuario, y la concurrencia de las actividades

necesarias para establecer la comunicación con colaboradores internos y externos. Además, se muestra el momento en el que el flujo de control secuencial se une para la comunicación interna de la empresa. En la figura 3, se ilustra el proceso llevado a cabo por los diferentes roles del sistema, detallando cada actividad realizada para facilitar la comunicación entre los colaboradores de la empresa.

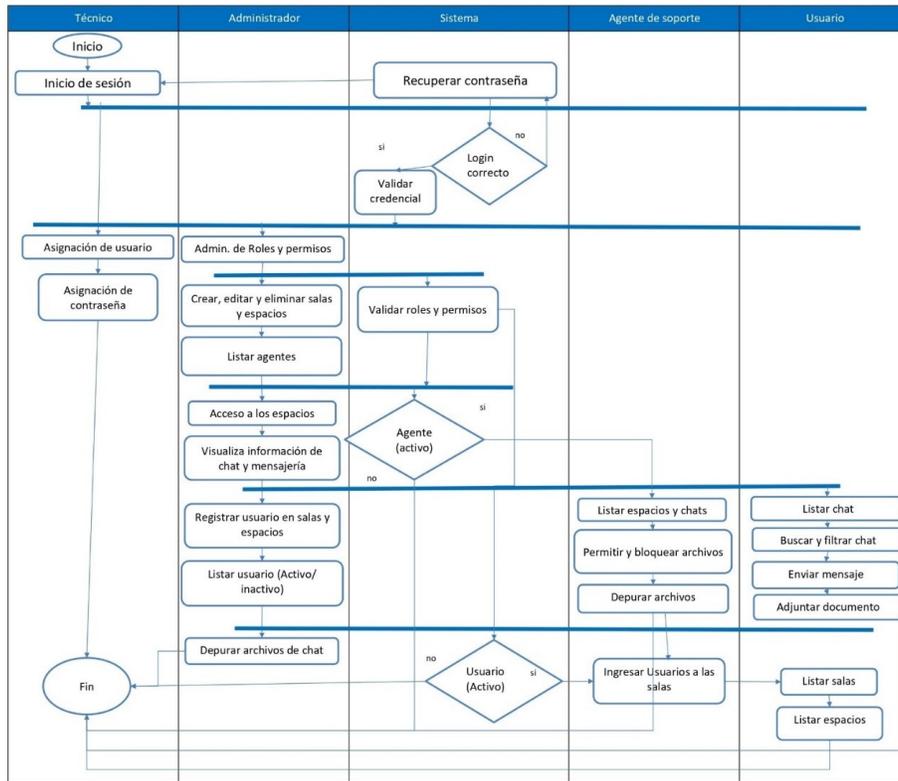


Figura 3. Modelo de procesos general de la aplicación Web (Elaboración propia).

D. Diagrama de estados

En la Figura 4, se presenta el diagrama de estados del "Chat Corporativo" desde la perspectiva del agente. Este diagrama proporciona una visualización detallada de las transiciones de estados que experimenta el sistema en respuesta a eventos particulares, ofreciendo así una comprensión clara y estructurada del comportamiento dinámico del chat desde la perspectiva del agente.

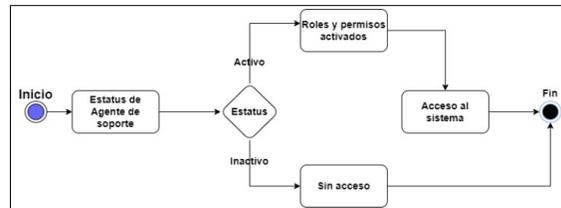


Figura 4. Diagrama de estados de la aplicación Web desde la perspectiva del agente de soporte (Elaboración propia).

El diagrama de estados ilustra la transición entre dos estados clave para los agentes dentro de un sistema: activo e inactivo. En el estado activo, el agente puede interactuar

plenamente con el sistema, teniendo acceso completo a todos los recursos y funciones disponibles. En contraste, en el estado inactivo, el agente se encuentra restringido y no puede interactuar con el sistema ni acceder a sus recursos o funciones.

E. Modelo de usuarios

La Figura 5 representa el modelo de usuarios del "Chat Corporativo". Este diagrama identifica los roles y la jerarquía de acceso dentro de la aplicación, resaltando cómo los distintos tipos de usuarios pueden interactuar con el sistema.

Los usuarios son:

- Agentes instanciables: Son los usuarios que la aplicación del Chat Corporativo reconoce y autoriza mediante sistemas de autenticación. Estos agentes pueden ser:
- Registrados: Son usuarios que necesitan identificarse para utilizar el chat corporativo. Están representados en el diagrama con un símbolo de candado, lo cual indica que su acceso está protegido y requiere autenticación.



Figura 5. Modelo de usuarios de la aplicación Web (Elaboración propia).

Se identificaron tres tipos de usuarios: administrador, agente de soporte y usuario. Al autenticarse con un usuario y contraseña, se produce un cambio de rol, lo que altera tanto su interacción como sus vistas. Por lo tanto, los usuarios: administrador, agente de soporte y usuario son considerados agentes abstractos, y cada uno posee propiedades de navegación distintas. Esto implica que tanto el administrador como el agente de soporte y el usuario tendrán sus propios mapas de navegación.

F. Modelo de navegación

La Figura 6 presenta el mapa de navegación diseñado específicamente para el usuario y agente de soporte, definiendo la interfaz del sistema para este perfil. Este mapa otorga al agente de soporte acceso a cuatro nodos de navegación distintos (página de inicio, acceso, usuarios y chat). Cada nodo representa un contexto de navegación que ofrece una perspectiva de la información y funcionalidades definidas en el diagrama de clases. Así el mapa de navegación se muestra con los contextos de navegación secuenciales disponibles para usuarios autenticados.

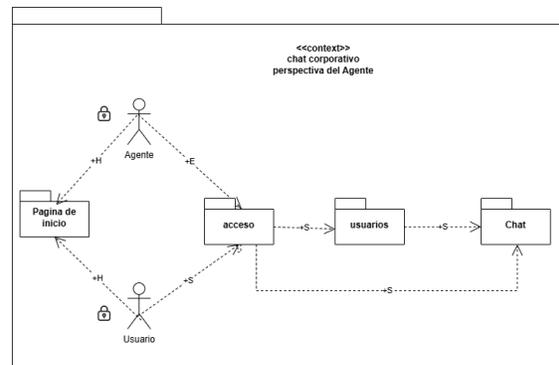


Figura 6. Modelo de navegación de la aplicación Web desde la perspectiva del agente de soporte (Elaboración propia).

En el contexto "Agente de soporte", se identifica un arco clasificado como "Inicio de Aplicación" y otro como "Exploración". Además, se establecen dos relaciones de tipo secuencia que delinean la secuencia específica de interacciones entre los distintos elementos del sistema.

G. Modelo de navegación de contexto

En la Figura 7, se presenta el "modelo de navegación de contexto" diseñado para el "Chat Corporativo" desde la perspectiva de Agente de soporte. Este modelo ofrece una representación visual estructurada de la organización de la navegación en el sistema, teniendo en cuenta diferentes contextos de interacción. Cada elemento en este modelo refleja la adaptación de la interfaz y las opciones de navegación según el lugar y la

situación específica en la que se encuentra el usuario dentro de la plataforma de chat. Este enfoque permite una experiencia de usuario más intuitiva y personalizada, facilitando la exploración y la interacción fluida con las diversas funcionalidades del "Chat Corporativo".

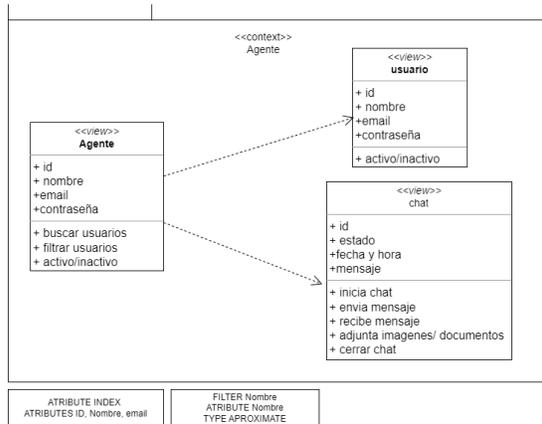


Figura 7. Modelo de navegación de contexto de la aplicación Web desde la perspectiva del agente de soporte (Elaboración propia).

La metodología OOWS fue fundamental para el desarrollo exitoso de la aplicación web. Dado que el equipo trabajaba de manera remota, inicialmente se consideró que tanto la comunicación como el desarrollo podrían ser desafíos significativos.

3.2. Desarrollo del Sistema Chat corporativo bajo el marco de trabajo Ágil SCRUM

Para la implementación del sistema se adoptó la metodología ágil SCRUM, reconocida por su capacidad de gestionar proyectos de desarrollo de software de manera iterativa e incremental. SCRUM facilitó la adaptación continua a los cambios en los requisitos del proyecto y promovió la entrega periódica de funcionalidades operativas, lo que permitió obtener retroalimentación constante de los usuarios y stakeholders [10].

En este proyecto se identificaron tres sprints, cada uno duró 4 semanas y abarcó tareas y funcionalidades a implementar manteniendo el proyecto alineado con los

objetivos y expectativas del usuario final. El proceso realizado se muestra en la figura 8.

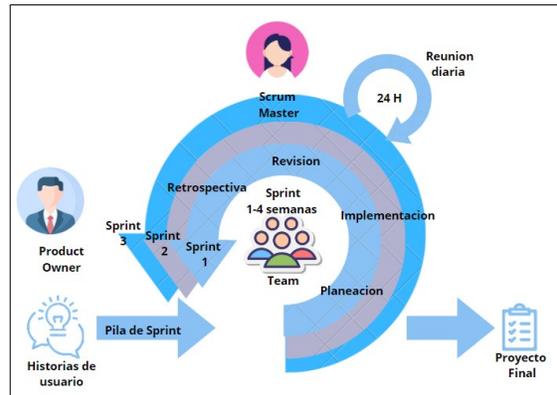


Figura 8. Visualización del progreso de SCRUM en sistema Chat Corporativo (Elaboración propia).

A. Roles Clave

El equipo de trabajo se integró con ocho personas, y de acuerdo con el enfoque de SCRUM se identificaron y asignaron los roles y responsabilidades definidas en la metodología: Product Owner (Propietario del Producto), SCRUM Máster, y Equipo de Desarrollo (Team). El equipo conformado de forma interdisciplinaria trabajó de forma colaborativa y en línea para cumplir con los objetivos del sprint.

B. Proceso SCRUM

El equipo de trabajo realizó las tareas de identificación y priorización de las historias de usuario, las cuales desarrollaron cada una las funcionalidades requeridas para el chat corporativo. Estas historias se detallaron y se almacenaron en la pila de producto.

Antes de iniciar cada sprint, se seleccionaron las historias de usuario prioritarias de la pila de producto que se convierten en tareas concretas para cada integrante del equipo. Estas tareas conformaron la pila de sprint.

Cada sprint tuvo una duración fija 4 semanas. Durante este periodo, el equipo de desarrollo trabajó en las tareas definidas en la pila de sprint para desarrollar cada una de las

funcionalidades del chat corporativo.

Todos los días, el equipo se reunió durante 15 minutos para discutir el progreso, los planes y los impedimentos que se tuvieron. Estas reuniones aseguraron la transparencia y la coordinación continua entre cada uno de los miembros del equipo.

Durante el sprint, el equipo desarrolló las funcionalidades del chat corporativo, realizando tareas de programación, pruebas y revisión. Este proceso incluyó el desarrollo de características como la capacidad de iniciar sesión, enviar y recibir mensajes, compartir archivos e imágenes, y gestionar la interacción entre usuarios y agentes.

Al finalizar cada sprint, el equipo presentó el trabajo completado al Product Owner y a otras partes interesadas. Durante esta revisión, se demostró el incremento del producto, es decir, las nuevas funcionalidades implementadas en el chat corporativo.

Después de la revisión de cada sprint, el equipo reflexionó sobre el proceso y su desempeño, identificando aspectos que funcionaron bien y áreas de mejora. Esta retroalimentación se utilizó para ajustar y optimizar futuros Sprints.

Antes de comenzar un nuevo sprint, se realizó una sesión de planificación en la cual el equipo seleccionó las historias de usuario de la pila de producto y se definieron las tareas a realizar en el próximo sprint. Esta planeación garantizó que el equipo estuviera alineado con las prioridades y objetivos del proyecto.

C. Resultados del proyecto final

Al finalizar los Sprints, se completó el desarrollo del chat corporativo. En el producto final se incorporaron todas las funcionalidades y mejoras definidas en las historias de usuario, cumpliendo con los objetivos y expectativas del usuario final. El

enfoque iterativo e incremental de SCRUM permitió un desarrollo fuera flexible y adaptativo, asegurando que el chat corporativo se construyera de acuerdo con las necesidades cambiantes y prioritarias del cliente.

3.3. Resultados obtenidos del desarrollo del “Chat Corporativo”

El diseño, desarrollo e implementación del chat corporativo con las metodologías OOWS y SCRUM dio resultados positivos en varios aspectos clave del sistema. A continuación, se detallan los principales hallazgos y beneficios observados:

La funcionalidad de inicio de sesión como se muestra en la figura 9, fue optimizada para garantizar un acceso rápido y seguro al sistema, pues se implementaron medidas de seguridad, como la autenticación. Lo cual resultó en una reducción significativa en los problemas de acceso reportados por los usuarios.

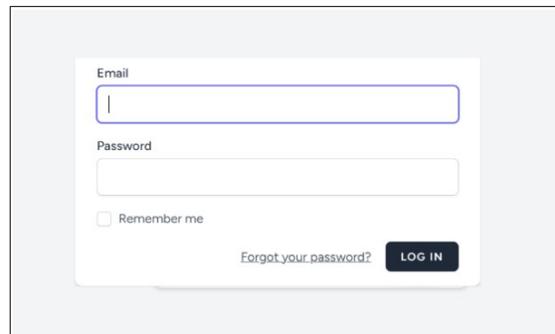


Figura 9. Interfaz de inicio de sesión del Chat Corporativo (Elaboración propia).

La figura 10 presenta la interfaz de usuario diseñada para la creación de nuevos usuarios dentro del sistema chat corporativo.

La interfaz es intuitiva y coherente, facilitó el proceso de registro de nuevos usuarios. Los campos obligatorios y las validaciones implementadas garantizan la calidad de los datos ingresados. Del lado izquierdo de la pantalla, se muestra una lista de usuarios actualmente registrados en el sistema. Los

usuarios están organizados alfabéticamente y pueden ser filtrados o buscados mediante un cuadro de búsqueda. Cada usuario está identificado por su nombre y un avatar con sus iniciales.

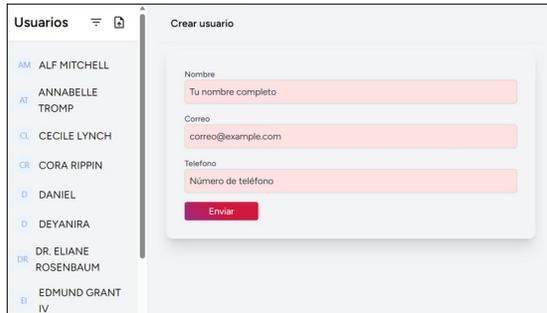


Figura 10. Interfaz de creación de usuarios del Chat Corporativo (Elaboración propia).

En la figura 11 del lado derecho, se encuentra la sección de importación de usuarios. Esta sección proporciona una interfaz sencilla para cargar el archivo Excel que contiene los datos de los usuarios a importar.

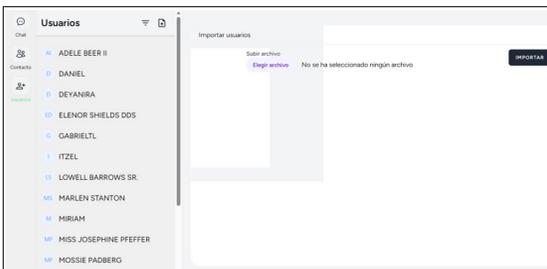


Figura 11. Interfaz de importación de usuarios del Chat Corporativo (Elaboración propia).

La interfaz de creación de una nueva sala de chat como se muestra en la figura 12 es una funcionalidad clave dentro del sistema de chat corporativo. Esta pantalla está diseñada para facilitar la configuración de nuevas salas donde los usuarios pueden comunicarse de manera grupal.

La interfaz presentada en la figura 13 corresponde a un formulario diseñado para la creación de nuevos espacios virtuales dentro de una plataforma de chat corporativo. Esta funcionalidad es esencial para permitir a los usuarios personalizar y

configurar sus propios entornos de colaboración de manera efectiva.



Figura 12. Interfaz de creación de una sala del Chat Corporativo (Elaboración propia).

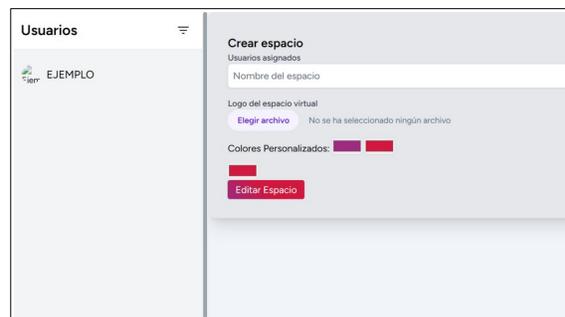


Figura 13. Interfaz de la Creación de espacios del Chat Corporativo (Elaboración propia).

4. ESTUDIO DE SATISFACCIÓN

Una vez implementado el sistema de chat corporativo, se aplicó una encuesta de satisfacción entre los usuarios de la empresa, con el fin de evaluar aspectos clave como la seguridad, confiabilidad, disponibilidad y estabilidad, velocidad, funcionalidad, diseño de la interfaz de usuario, además de identificar la satisfacción y percepción de los empleados respecto al funcionamiento del sistema, su efectividad en la comunicación interna y detectar posibles áreas de mejora. Las preguntas fueron diseñadas con un formato de medición tipo Likert abarcando los siguientes aspectos.

1. Seguridad del Sistema
 - ¿Te sientes seguro al utilizar el chat corporativo en términos de protección de la información y privacidad de los datos?
2. Confiabilidad del Chat

- ¿Qué tan estable consideras que es el chat corporativo en su funcionamiento continuo sin interrupciones?
3. Disponibilidad y Estabilidad
¿Con que frecuencia has experimentado interrupciones o problemas de conexión al usar el chat corporativo?
 4. Velocidad de Respuesta
¿Cómo consideras que es la respuesta del sistema al enviar y recibir mensajes?
 5. Funcionalidad y Características
¿En qué medida el chat corporativo cubre todas las necesidades de comunicación que tienes en tu trabajo diario?
 6. Diseño e Interfaz de usuario
¿Cuál es la calificación que tiene el diseño y la interfaz de usuario del chat corporativo?
 7. Envío de Archivos y Multimedia
¿Has tenido alguna dificultad al enviar o recibir archivos multimedia a través del chat corporativo?
 8. Funcionalidad de las Salas
¿Cómo evaluarías la confiabilidad y el rendimiento de las salas de chat en términos de su capacidad para permitir la comunicación fluida y la gestión de conversaciones?
 9. Opinión General
¿Cómo evaluarías tu satisfacción general con el chat corporativo en cuanto a seguridad, confiabilidad y desempeño?

Al analizar las respuestas obtenidas en la encuesta, se puede concluir que el sistema de chat corporativo ha cumplido con los principales requisitos de calidad establecidos. Los usuarios han reportado buenos niveles de satisfacción en términos de seguridad y confiabilidad, lo que indica que el sistema protege adecuadamente la información y ofrece una plataforma segura para la comunicación interna.

Además, la funcionalidad general del chat, incluida la capacidad de enviar archivos multimedia, ha sido evaluada como positiva, lo que demuestra que cumple con las necesidades de comunicación diaria de los usuarios. La interfaz del sistema también ha

sido bien recibida, destacando su facilidad de uso y diseño intuitivo

El sistema ha destacado las expectativas y ha brindado una experiencia eficiente y confiable a los usuarios. Esto confirma que los estándares de calidad establecidos en la implementación del chat corporativo se han alcanzado de manera satisfactoria.

5. CONCLUSIONES Y TRABAJOS FUTUROS

En este estudio, se presentó el proceso de desarrollo de un sistema chat corporativo diseñado para mejorar la comunicación interna en una organización. Se mostró la aplicación de la metodología OOWS para el modelado del sistema y el marco ágil SCRUM, se logró construir una solución que cumple con los requisitos funcionales y no funcionales establecidos y en el tiempo requerido.

El chat corporativo ofrece una solución integral para las empresas, brindando medidas de seguridad robustas, control sobre los datos en servidores internos y cumplimiento con normativas de protección de datos, eliminando preocupaciones sobre la privacidad al evitar servidores de terceros, como WhatsApp. Los administradores tienen la capacidad de gestionar el acceso de los empleados, controlando permisos y restringiendo el acceso cuando un empleado deja la empresa.

Además, facilita la organización de la comunicación al permitir la creación de canales o grupos específicos para departamentos, proyectos o temas, mejorando la eficiencia en el flujo de trabajo.

Al mantener separada la comunicación profesional de la personal, reduce distracciones y previene confusiones, mientras que su capacidad de escalar con el crecimiento de la empresa permite gestionar grandes volúmenes de usuarios y conversaciones. En última instancia, la

empresa tiene un control total sobre los datos generados, lo que le permite proteger su información sensible, auditar su uso y garantizar la confidencialidad.

Sin embargo, para hacer frente a un crecimiento de usuarios en la empresa y garantizar una alta disponibilidad, se propone como trabajo futuro migrar el chat corporativo a una plataforma de Cloud Computing como AWS, Microsoft Azure o Google Cloud Platform. Se espera que esta migración permita escalar los recursos, reducir los costos de mantenimiento, mejorar la seguridad, la flexibilidad y alta disponibilidad de los datos mediante las robustas medidas de seguridad ofrecidas por Cloud Computing.

En conclusión, el desarrollo de este chat corporativo representa un aporte significativo para mejorar la comunicación y organización en la empresa contribuyendo a la colaboración y la productividad. Sin embargo, es necesario continuar investigando y desarrollando nuevas funcionalidades para optimizar aún más esta herramienta y adaptarla a las necesidades cambiantes de las organizaciones..

REFERENCIAS

- [1] Velarde Paredes G, Pilco Quitiu MI. *Análisis comparativo de metodologías para el desarrollo de la aplicación web del control de las prácticas pre-profesionales de la EIS de la ESPOCH* [tesis de grado]. Riobamba (Ecuador): Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica, Escuela de Ingeniería en Sistemas; 2014 [consultado 30 de julio de 2024]. Disponible en: <http://dspace.esPOCH.edu.ec/handle/123456789/3576>
- [2] Valverde F, Panach JI, Aquino N, Pastor Ó. Hacia un Modelo de Interacción Abstracto para la Definición de Interfaces Multiplataforma. En: Macías Iglesias JA, Granollers Saltiveri T, Latorre Andrés PM, coordinadores. *Actas del VIII Congreso Internacional de Interacción Persona Ordenador (INTERACCIÓN 2007)*. Madrid: Thomson; 2007. p. 251-260. [consultado 30 de julio de 2024]. Disponible en: <https://producciocientifica.uv.es/documentos/5eb09d90299952764112d126>
- [3] Vinay Kumar T, Yathish C, VikeshReddy, Shamanth D, LaxmiRananavare. Internet Chat Application. *International Journal of Advanced Research in Computer Science* [Internet]. 2021 [consultado 30 de julio de 2024];12(2):98-100. Disponible en: <https://www.ijarcs.info/index.php/ijarcs/article/view/6731/5427>
- [4] Cherckesova L, Safaryan O, Reshetnikova I, Nikishina T, Korochentsev D. Corporate chat under DLP-system controlling. *E3S Web of Conferences* [Internet]. 2021 Jun 22;273:08048. *XIV International Scientific and Practical Conference "State and Prospects for the Development of Agribusiness - INTERAGROMASH 2021"*. doi: [10.1051/e3sconf/202127308048](https://doi.org/10.1051/e3sconf/202127308048)
- [5] Kumar M, Thakur V, Gurjar D. Multi-User Web Chat Application using Node.js and Socket.io. *International Journal of Advanced Research in Science, Communication and Technology (IJAR SCT)* [Internet]. 2022 [consultado 30 de julio de 2024];2(4):386-394. Disponible en: <https://ijarsct.co.in/Paper3940.pdf>
- [6] Dong L, Hua Z, Huang L, Ji T, Jiang F, Tan G, Zhang J. The impacts of live chat on service-product purchase: Evidence from a large online outsourcing platform. *Information & Management*. 2024;61(3):103931. doi: [10.1016/j.im.2024.103931](https://doi.org/10.1016/j.im.2024.103931).
- [7] Daniel G, Cabot J. Applying model-driven engineering to the domain of chatbots: The Xatkit experience. *Science of Computer Programming*. 2024 Jan;232:103032. doi: [10.1016/j.scico.2023.103032](https://doi.org/10.1016/j.scico.2023.103032)
- [8] Muñoz Sánchez Y, Alonso-Lavernia MA, Castillo-Pérez I, Martínez-Lazcano V, Gálvez-González F. Desarrollo de un Sitio Web con metodologías de Diseño Hipermedial y de Diseño Dinámico. *Ingenio y Conciencia Boletín Científico De La Escuela Superior Ciudad Sahagún*. 2020;7(13):36-41. doi: [10.29057/escs.v7i13.5263](https://doi.org/10.29057/escs.v7i13.5263)
- [9] Castillo Montes M, Vargas Enríquez J, García Mundo L. Modelado Conceptual de una Aplicación Web usando la Metodología OOWS: Caso Práctico. *TecnoINTELECTO*. 2018;15(12):19-28 [Internet]. [consultado 29 de julio de 2024]. Disponible en: <https://es.scribd.com/document/597541404/2-Modelado-conceptual-de-una-aplicacion-Web-usando-la-metodolgi-a-OOWS>
- [10] Schwaber K, Sutherland J. *Guía de Scrum: La Guía Definitiva de Scrum: Las Reglas del Juego* [Internet]. 2020 [consultado 29 de julio de 2024]. Disponible en: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-Latin-South-American.pdf>

ACERCA DE LOS AUTORES



Miriam López Sanluis completó la Licenciatura en Ingeniería en Tecnologías de la Información y Comunicaciones en el año 2019 en el Instituto Tecnológico de Apizaco. Posteriormente, en 2023, ingresó a la Maestría en Sistemas Computacionales en la misma institución, especializándose en la línea de Ingeniería de Software. A lo largo de su formación académica, se involucró activamente en el desarrollo e implementación de soluciones tecnológicas, participando en diversos proyectos de desarrollo de software.



Elizabeth Cuatecontzi Cuahutle es Licenciada en Informática por el Instituto Tecnológico de Apizaco. Maestra en Dirección de Ingeniería de Software por el Instituto de Estudios Universitarios. Docente de tiempo completo en el Depto. de Sistemas y Computación del Tecnológico Nacional de México Campus Apizaco. Miembro del consejo de la Maestría en Sistemas Computacionales del TecNM Campus Apizaco. Área de interés: ingeniería de software y sistemas distribuidos. Actualmente realiza los estudios de doctorado en el programa de doctorado en ciencias de la ingeniería.



María Guadalupe Medina Barrera es Doctora en Planeación Estratégica y Dirección de Tecnología por la Universidad Popular Autónoma del Estado de Puebla (UPAEP). Realizó estudios de Maestría en Ciencias en Ciencias Computacionales en el Centro

Nacional de Investigación y Desarrollo Tecnológico (CENIDET) y estudios de Licenciatura en Informática en el Instituto Tecnológico de Tepic, Nayarit, México. Actualmente, cuenta con el reconocimiento como candidata en el Sistema Nacional de Investigadores del CONAHCYT, además del reconocimiento al Perfil Deseable y de ser miembro del cuerpo académico "Sistemas de Información", ambos reconocidos por PRODEP. Sus áreas de trabajo son: Gestión y Desarrollo de Proyectos de Software, Automatización de Procesos, Interfaces Humano Computadora y Reconocimiento de Patrones.



María Janái Sánchez Hernández es Licenciada en Informática en el año 2001 y Maestra en Ciencias en Ciencias de la Computación en 2005, ambas carreras cursadas en el Instituto Tecnológico de Apizaco. Sus intereses académicos son la Ingeniería de Software, el uso de las metodologías ágiles, inteligencia artificial aplicada a la ingeniería de software, la deuda técnica y el desarrollo de software. Desde 2004 es docente del área de Sistemas y Computación del TecNM Apizaco impartiendo diversas signaturas, y desde 2015 es colaboradora de la Maestría en Sistemas Computacionales del Instituto Tecnológico de Apizaco.



José Juan Hernández Mora es Ingeniero en Computación por la Universidad Autónoma de Tlaxcala. Tiene el grado de Maestro en Ciencias en Ciencias Computacional es por el Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET), de Cuernavaca, Morelos y Doctor en Excelencia Docente por la Universidad de los Ángeles. Es Profesor con Perfil Deseable por parte del PRODEP, es líder del cuerpo académico "Sistemas de Información" y nivel

de candidato del SNII del CONAHCYT. Sus líneas de investigación incluyen: Ingeniería de Software, Desarrollo de Aplicaciones de Tecnologías de la Información, Procesamiento Digital de Imágenes (PDI), Redes Neuronales Artificiales (RNA).