**Universidad Mexiquense del Bicentenario Unidad de Estudios Superiores Jiquipilco**

Universidad Mexiquense del Bicentenario

Unidad de Estudios Superiores Jiquipilco

ingeniería en Sistemas Computacionales

Residencias Profesionales

"Desarrollo del componente Controlador de la Arquitectura de software MVC para la Mesa de ayuda interna de la Dirección General de Personal"

Nombre del Estudiante: Osciel Ordoñez Jimenez

Matricula: 10210090

Asesor Interno: Mtra. en C. Computacionales y Telecomunicaciones Elizabeth Medina Morales

Asesor Externo: Ing. Christopher Delgadillo Ramírez

Fecha de inicio: 01 de septiembre de 2025

Fecha de término: 30 de enero de 2026

Jiquipilco, Edo. de México a 19 de septiembre 2025.

**Índice**

[**Resumen** 2](#_Toc209616495)

[**Abstract** 4](#_Toc209616496)

[**Introducción** 6](#_Toc209616497)

[**Capítulo 1. Antecedentes** 8](#_Toc209616498)

[**1.1 Planteamiento del Problema** 8](#_Toc209616499)

[**1.2 Objetivos** 10](#_Toc209616500)

[**1.2.1 Objetivo general** 10](#_Toc209616501)

[**1.2.2 Objetivos específicos** 11](#_Toc209616502)

[**1.3 Justificación** 11](#_Toc209616503)

[**1.4 Estado del arte** 13](#_Toc209616504)

[**1.5 Alcances y Limitaciones** 16](#_Toc209616505)

[**1.5.1 Alcances** 16](#_Toc209616506)

[**1.5.2 Limitaciones** 17](#_Toc209616507)

[**Capítulo II. Fundamentos teóricos** 19](#_Toc209616508)

[**Capítulo III. Metodología** 19](#_Toc209616509)

[**3.1 Sprint Planning** 19](#_Toc209616510)

[**3.2 Sprint Backlog** 19](#_Toc209616511)

[**3.3 Sprint Review** 19](#_Toc209616512)

[**3.4 Sprint Retrospective** 19](#_Toc209616513)

[**3.5 Finished Work** 19](#_Toc209616514)

[**3.6 Resultados** 19](#_Toc209616515)

[**3.7 Recomendaciones** 19](#_Toc209616516)

[**Bibliografía** 19](#_Toc209616517)

[**Anexos** 19](#_Toc209616518)

[**A.** **Manual de Usuario** 19](#_Toc209616519)

[**B.** **Manual Técnico** 19](#_Toc209616520)

[**C.** **Documentación Legal** 19](#_Toc209616521)

# **Resumen**

**Abstract**   

# **Introducción**

# **Capítulo 1. Antecedentes**

## **1.1 Planteamiento del Problema**

En la actualidad, la Dirección General del Personal enfrenta dificultades para gestionar de manera eficiente los incidentes, solicitudes y problemas técnicos que surgen en su operación diaria. La falta de un sistema centralizado para registrar, priorizar y dar seguimiento a las solicitudes de soporte provoca retrasos en la atención, pérdida de información, duplicación de esfuerzos y una disminución considerable en la satisfacción de los usuarios, quienes en este caso son los servidores públicos.

Asimismo, soporte técnico suele brindarse de manera reactiva, sin contar con un historial claro de incidencias que permita identificar áreas de mejora o patrones de fallas recurrentes. Esta situación limita la capacidad de la Dirección General del Personal para tomar decisiones informadas, optimizar recursos humanos y materiales, y garantizar la continuidad de los servicios que resultan esenciales para el desempeño institucional.

De igual forma, la ausencia de una herramienta automatizada para la gestión de tickets y el seguimiento de incidentes también genera dificultades en la comunicación entre los servidores públicos y el personal de la dirección de sistemas. Como consecuencia, se producen confusiones, tiempos de resolución prolongados y, en muchos casos, impactos negativos en la productividad. Esta falta de procesos estandarizados y de un repositorio histórico de casos impide consolidar un conocimiento organizacional que favorezca la resolución de problemas recurrentes, lo que provoca una dependencia excesiva de la experiencia individual del personal técnico y una menor capacidad de respuesta ante situaciones críticas.

Por otro lado, los servicios que se atienden dentro del área de soporte son variados y abarcan necesidades que van desde incidencias técnicas hasta requerimientos administrativos. Sin embargo, al no encontrarse centralizados dentro de un sistema único, su atención se vuelve más compleja, menos eficiente y con mayores posibilidades de retrasos o pérdida de información.

En este escenario, resulta evidente la necesidad de desarrollar una Mesa de Ayuda que centralice la recepción, registro, clasificación, asignación y seguimiento de las solicitudes, con el fin de ofrecer un servicio más ágil, transparente y eficiente. Asimismo, la incorporación de reportes e indicadores dentro del sistema permitirá generar información estratégica para la toma de decisiones, mejorar la experiencia de los servidores públicos y contribuir de manera significativa a la continuidad operativa de la institución.

Para lograr este objetivo, la arquitectura propuesta se apoya en el patrón **Modelo-Vista-Controlador (MVC)**. Esta división del trabajo resulta esencial porque permite separar las responsabilidades del sistema de forma ordenada: el Modelo gestiona la lógica de datos y las reglas de negocio, la Vista presenta la información al usuario de manera clara y accesible, y el Controlador administra la comunicación entre ambos. Gracias a esta estructura, se favorece el trabajo en equipo, se asegura la escalabilidad del proyecto y se facilita el mantenimiento futuro, ya que cada integrante puede enfocarse en un componente particular sin afectar el resto del sistema.

Dentro de esta arquitectura, el **Controlador** adquiere una importancia fundamental. Este módulo será el encargado de gestionar la lógica de los procesos, de tal forma que pueda coordinar de manera ordenada el flujo de información entre la base de datos, los técnicos y la interfaz de usuario. Es a través del Controlador que se recibirán las solicitudes de soporte, se validarán los datos ingresados, se clasificará cada caso de acuerdo con la categoría y prioridad correspondiente, y se asignará la atención al técnico adecuado. De esta manera, el Controlador garantizará que cada ticket cuente con un ciclo de vida completo, documentado y con trazabilidad desde su creación hasta su cierre.

Por otra parte, las causas que originan este problema radican en la falta de un sistema centralizado que permita controlar el flujo de tickets, lo que genera que cada área resuelva los incidentes de forma aislada. A ello se suma la ausencia de estandarización en los procesos de atención, la carencia de registros históricos que documenten los problemas más frecuentes, la limitada comunicación entre usuarios, técnicos y administradores, y la dependencia de la experiencia individual en lugar de contar con repositorios comunes.

En consecuencia, los efectos derivados de estas deficiencias se traducen en retrasos en la atención de incidencias, usuarios insatisfechos por la falta de respuestas rápidas o soluciones completas, duplicación de esfuerzos al atender un mismo problema desde distintos frentes, pérdida de productividad por los tiempos prolongados de inactividad en equipos o servicios, y dificultades en la toma de decisiones por la carencia de reportes consolidados. También se observa una escasa trazabilidad en el manejo de tickets, lo que limita la transparencia, obstaculiza la auditoría y frena la posibilidad de implementar procesos de mejora continua.

En conclusión, la falta de un Controlador eficiente dentro de un sistema de Mesa de Ayuda no solo compromete la calidad del servicio ofrecido a los servidores públicos, sino que también afecta el funcionamiento integral de la institución. La implementación de este componente permitirá centralizar, ordenar y automatizar los procesos de atención, generando un impacto positivo tanto en la satisfacción de los usuarios como en la eficiencia y la sostenibilidad operativa de la Dirección General del Personal.

## **1.2 Objetivos**

### **1.2.1 Objetivo general**

* Desarrollar un sistema de Mesa de Ayuda que permita registrar, organizar, dar seguimiento y resolver de manera eficiente las incidencias técnicas de los usuarios de la Dirección General de Personal, fortaleciendo la comunicación con el área de soporte técnico y optimizando la calidad del servicio.
* Desarrollar el componente Controlador del sistema de Mesa de Ayuda, encargado de gestionar la comunicación entre las interfaces de usuario y la lógica de negocio, mediante la validación de solicitudes, la administración de rutas y la coordinación de procesos de registro, clasificación, priorización y seguimiento de tickets, garantizando un flujo de información confiable, ordenado y eficiente dentro de la Dirección General de Personal.

### **1.2.2 Objetivos específicos**

* Definir y desarrollar los controladores del sistema que gestionen la comunicación entre las vistas de la Mesa de Ayuda y los modelos de datos, asegurando un flujo de información eficiente y seguro.
* Implementar rutas que permitan registrar, actualizar, consultar y dar seguimiento a los tickets de soporte, de acuerdo con las categorías de incidencias establecidas (redes, equipos, licencias, impresoras, correos y biopads).
* Garantizar la validación y control de datos enviados por los usuarios desde la interfaz, evitando inconsistencias en la información almacenada en la base de datos Oracle.
* Integrar métricas de desempeño en los controladores, para que la información sobre tiempos de resolución, tickets pendientes y satisfacción de usuarios pueda ser consultada y graficada en reportes.

## **1.3 Justificación**

La implementación de una Mesa de Ayuda representa una solución estratégica para optimizar la gestión de soporte técnico dentro de la Dirección General de Personal. Este sistema interno no solo mejora la eficiencia en la atención de incidencias y solicitudes, sino que también establece un canal de comunicación claro y documentado entre el área de soporte y las distintas unidades de la Dirección, asegurando la trazabilidad de los servicios y garantizando la integridad de los procesos.

Por otra parte, la elección de la arquitectura Modelo-Vista-Controlador (MVC) responde a la necesidad de separar de manera ordenada las responsabilidades del sistema: el Modelo gestiona la lógica de datos y las reglas de negocio, la Vista presenta la información al usuario final mediante interfaces claras e intuitivas, y el Controlador administra la comunicación entre ambos. Esta división facilita el trabajo en equipo, asegura escalabilidad y favorece el mantenimiento futuro del sistema, al mismo tiempo que permite que cada integrante del proyecto se especialice en un componente particular.

En cuanto al ámbito tecnológico, el sistema se fundamenta en un entorno moderno y robusto que garantiza estabilidad y eficiencia. Estará desarrollado en Java, utilizando IntelliJ IDEA como entorno de programación, lo que permite aprovechar librerías, frameworks y APIs que faciliten la implementación. Para la presentación de la información se emplean HTML y CSS, mientras que la gestión de datos se realiza en una base de datos Oracle, administrada con SQL Developer. Esta combinación de tecnologías asegura un alto nivel de integración, seguridad y rendimiento, además de alinearse con los estándares profesionales que demanda la institución.

Desde la perspectiva organizacional, el desarrollo de esta Mesa de Ayuda contribuye a optimizar el uso de recursos, reducir costos derivados de ineficiencias y fortalecer la calidad del servicio interno. En un contexto donde la rapidez, la trazabilidad y la transparencia son factores clave, contar con un sistema estructurado bajo estas tecnologías se convierte en una necesidad estratégica que impacta directamente en la productividad y competitividad institucional.

Asimismo, el sistema se concibe con un enfoque de mejora continua, lo que permitirá incorporar nuevas funcionalidades orientadas a la automatización de procesos, el autoservicio y la integración con plataformas institucionales clave. De esta manera, la Mesa de Ayuda se consolida como una herramienta esencial para garantizar eficiencia, calidad y adaptabilidad en la prestación de servicios tecnológicos.

Finalmente, dentro de la arquitectura MVC, el componente Controlador desempeña un papel fundamental al gestionar la interacción entre las interfaces de usuario y la lógica de negocio. Su función principal es recibir las solicitudes generadas en la Vista, validarlas y dirigirlas hacia el Modelo para su procesamiento, garantizando coherencia en el flujo de información. Además, administra la asignación de rutas, el control de estados de los tickets y la comunicación con los diferentes módulos del sistema. Una correcta implementación del Controlador asegura que las operaciones de registro, clasificación, priorización y seguimiento de incidencias se realicen de manera ágil, confiable y ordenada, consolidándose como el eje que mantiene la integridad del proceso de atención dentro de la Mesa de Ayuda.

## **1.4 Estado del arte**

El análisis del estado del arte permite identificar avances y enfoques en la aplicación del patrón Modelo-Vista-Controlador (MVC), aportando un marco de referencia para la construcción de la Mesa de Ayuda. Esta revisión hace posible reconocer ventajas, limitaciones y aportaciones de proyectos previos que fortalecen el diseño de un sistema robusto y escalable. En este documento se abordará el **Controlador**, componente encargado de gestionar la lógica de los procesos, coordinar el flujo de información entre la base de datos, los técnicos y la interfaz de usuario, así como garantizar la trazabilidad y el correcto funcionamiento de las operaciones dentro de la Mesa de Ayuda.

En **2020**, se desarrolló el proyecto *“Aplicación del patrón de diseño MVC utilizando Yii2 framework en el desarrollo del módulo de consulta externa del sistema integral hospitalario Rovirosa”*. Este trabajo tuvo como propósito implementar un sistema que separara claramente la lógica de negocio, la interfaz de usuario y el controlador, en un entorno hospitalario donde el registro de pacientes, consultas y diagnósticos era complejo y propenso a errores. El controlador jugó un papel central en la comunicación entre los formularios médicos y la base de datos, asegurando que los datos fueran validados antes de almacenarse y facilitando el flujo de información entre los distintos módulos. La aportación de este proyecto a nuestro caso es que evidencia cómo la implementación de un controlador dentro del patrón MVC puede mejorar la confiabilidad de los datos, reducir duplicidades y optimizar procesos en escenarios críticos. Esto resulta relevante para la Mesa de Ayuda, donde también se requiere garantizar precisión y trazabilidad en el manejo de tickets y solicitudes de soporte. Hernández, J. (2020).

En **2021**, se realizó la investigación *“Impacto del patrón modelo vista controlador (MVC) en la seguridad, interoperabilidad y usabilidad de un sistema informático”*. El objetivo fue analizar de qué manera el patrón MVC, y específicamente los controladores, contribuyen a aumentar la seguridad en el acceso a datos, la interoperabilidad entre sistemas y la usabilidad de las plataformas. Los resultados mostraron que la capa de controladores puede convertirse en una barrera de seguridad al validar entradas y salidas, además de servir como punto de integración entre sistemas heterogéneos. Para nuestro proyecto, la principal aportación radica en entender que el controlador no solo coordina acciones, sino que también puede robustecer la seguridad de la Mesa de Ayuda, evitando inyecciones de datos, accesos indebidos y mejorando la experiencia de los servidores públicos al interactuar con el sistema. Esto abre la posibilidad de que nuestro desarrollo no se limite a organizar flujos, sino que también integre buenas prácticas de seguridad y escalabilidad. Paredes, L., & Gómez, F. (2021).

En **2022**, Microsoft publicó la guía *“Introducción a ASP.NET Core MVC”*, un tutorial oficial orientado al desarrollo de aplicaciones web escalables. En este proyecto, el controlador ocupa un rol primordial, ya que recibe las solicitudes de los usuarios, procesa la lógica de negocio y entrega respuestas a la vista. La documentación explicó buenas prácticas para estructurar controladores, manejar rutas y aplicar validaciones de manera eficiente. Este aporte es relevante para nuestro desarrollo porque establece un marco metodológico probado y ampliamente utilizado en la industria. Asimismo, ofrece ejemplos prácticos que pueden servir de referencia al implementar el componente Controlador en la Mesa de Ayuda, garantizando un diseño ordenado, seguro y con capacidad de crecimiento a largo plazo. Microsoft. (2022).

En **2023**, se desarrolló la tesis *“Implementación del patrón MVC en sistemas académicos para la mejora de la trazabilidad de procesos”* en universidades peruanas. El proyecto consistió en rediseñar plataformas académicas bajo MVC, donde el controlador se encargaba de enlazar módulos de matrícula, notas y reportes, garantizando coherencia en los datos. Uno de los hallazgos principales fue que centralizar la lógica en los controladores redujo inconsistencias, facilitó la trazabilidad y permitió la escalabilidad del sistema sin afectar la experiencia del usuario. Para nuestra Mesa de Ayuda, este aporte es esencial porque demuestra que los controladores pueden organizar procesos complejos, garantizando integridad y consistencia en el manejo de tickets, manuales, encuestas y reportes de soporte. Además, pone en evidencia que la correcta implementación del patrón MVC no solo responde a necesidades técnicas, sino también a requerimientos de gestión y control institucional. Ramos, M. (2023).

En **2024**, se presentó el trabajo *“Aplicación del patrón MVC en sistemas de tickets de soporte técnico usando Spring Boot y Oracle”*. Este desarrollo se enfocó en el diseño de una Mesa de Ayuda digital para una institución pública, donde el controlador centralizaba la recepción de tickets, los clasificaba por prioridad y los asignaba a técnicos de soporte. Además, permitía generar reportes automáticos y enviar notificaciones al usuario sobre el estado de su incidencia. La aportación de este proyecto a nuestro trabajo es directa, ya que aborda un caso muy similar al que nos ocupa. Sus resultados demuestran que los controladores en sistemas de soporte son el núcleo que garantiza la trazabilidad, la transparencia y la eficiencia en la resolución de incidencias, lo que refuerza la pertinencia de su implementación dentro de nuestra arquitectura. García, R. (2024).

Finalmente, en **2025**, se difundió el artículo *“Guía práctica para el uso de controladores en Spring Boot con arquitectura MVC”*. Este material se centró en entornos empresariales donde los controladores coordinan peticiones HTTP, gestionan datos de forma estructurada y permiten la integración con microservicios. El aporte más significativo fue mostrar ejemplos prácticos para escalar sistemas sin perder mantenibilidad, incorporando estrategias de modularidad y separación de responsabilidades. Para nuestro proyecto, la contribución es clara: los controladores son la pieza que permite articular diferentes módulos de la Mesa de Ayuda (tickets, técnicos, manuales, encuestas), asegurando que trabajen como un todo coherente y funcional. Además, ofrece evidencia de cómo este componente puede convertirse en la base para una futura expansión hacia arquitecturas más complejas, como microservicios o integraciones con herramientas de inteligencia artificial. Microsoft. (2025).

## **1.5 Alcances y Limitaciones**

### **1.5.1 Alcances**

En primer lugar, el presente proyecto tiene como alcance la implementación de una plataforma tecnológica de Mesa de Ayuda que centralice la recepción, gestión y resolución de incidentes y solicitudes en la Dirección General de Personal. El sistema busca optimizar los procesos de atención mediante la automatización del registro de tickets, la asignación de prioridades y la generación de reportes de desempeño, favoreciendo así la eficiencia en la resolución de problemas y la mejora continua del servicio. Además, contempla la integración con una base de datos que resguarde la información histórica, otorgando trazabilidad a los procesos y facilitando la toma de decisiones basadas en métricas objetivas.

Por otra parte, entre los alcances específicos del sistema se incluyen:

* La implementación de un sistema centralizado para el registro y seguimiento de tickets que abarque incidencias técnicas (fallas de equipo, problemas de red), solicitudes de servicio (instalación de software, creación de cuentas institucionales), mantenimiento y resguardo de activos (reportes de equipos dañados, mantenimientos preventivos), problemas recurrentes que permitan identificar causas raíz, cambios en infraestructura (actualizaciones, migraciones, configuraciones) y consultas o asesorías relacionadas con sistemas internos;
* La gestión de prioridades y estados de los tickets, clasificando las solicitudes según su urgencia e impacto (crítico, alto, medio, bajo) y su etapa en el ciclo de vida (abierto, en proceso, en espera, resuelto, cerrado);
* La generación de reportes estadísticos con indicadores como tiempos de resolución, incidencias recurrentes y cumplimiento de acuerdos de servicio, orientados a la evaluación del desempeño y a la identificación de áreas de mejora;
* El diseño responsivo de la plataforma, adaptable a computadoras, laptops, tabletas y dispositivos móviles, asegurando accesibilidad y usabilidad uniforme en distintos entornos.

De manera específica, el componente **Controlador** tiene como alcance administrar la lógica de los procesos internos de la Mesa de Ayuda. Su función será coordinar la comunicación entre la base de datos, los técnicos y la interfaz de usuario, asegurando que cada solicitud se valide, se clasifique por prioridad y categoría, y se asigne al técnico adecuado. Asimismo, este componente garantizará la trazabilidad del ciclo de vida de los tickets, desde su registro hasta su cierre, además de la generación de métricas relacionadas con tiempos de atención y carga de trabajo, fortaleciendo la capacidad de análisis y mejora continua del sistema.

### **1.5.2 Limitaciones**

En cuanto a las limitaciones generales, la eficacia de la Mesa de Ayuda dependerá de la calidad de los datos registrados por los usuarios y de la disponibilidad del personal técnico para dar seguimiento a los incidentes. Asimismo, su desempeño estará condicionado por la estabilidad de la conexión a internet, la capacidad del servidor y la compatibilidad con los sistemas existentes en la organización. La plataforma no garantiza la resolución inmediata de los problemas, pues su función principal es organizar, canalizar y dar trazabilidad a los procesos de atención.

De igual forma, entre las limitaciones específicas se encuentran:

* La cobertura de soporte restringida a los servicios definidos por la Dirección General de Personal, como incidencias de red, mantenimiento de equipos, gestión de licencias y software institucional, soporte a impresoras y periféricos, administración de correos electrónicos y dispositivos biométricos. No se consideran, en esta primera etapa, integraciones con sistemas externos ni servicios ajenos al ámbito de la Dirección.
* La dependencia de la infraestructura tecnológica institucional, por lo que fallos de hardware, caídas de red o insuficiencia en la capacidad de los servidores pueden afectar el desempeño del sistema.

Por otro lado, en el caso del componente **Controlador**, sus limitaciones se relacionan con la dependencia de la información recibida desde la interfaz y con la capacidad de la base de datos para responder de manera oportuna. Si bien gestiona el flujo de procesos, no puede garantizar por sí mismo la resolución de los incidentes, ya que requiere del trabajo de los técnicos y de la disponibilidad de recursos tecnológicos. Su desempeño también puede verse comprometido por errores en la programación de validaciones, fallas en la integración con otros módulos o restricciones de la infraestructura tecnológica, lo que impactaría en la eficiencia y trazabilidad de los tickets.

# **Capítulo II. Fundamentos teóricos**

# **Capítulo III. Metodología**

## **3.1 Sprint Planning**

## **3.2 Sprint Backlog**

## **3.3 Sprint Review**

## **3.4 Sprint Retrospective**

## **3.5 Finished Work**

## **3.6 Resultados**

## **3.7 Recomendaciones**

# **Bibliografía**

# **Anexos**

## **Manual de Usuario**

## **Manual Técnico**

## **Documentación Legal**