

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

SISTEMA CLOUD ERP PARA LA GESTIÓN DE RECURSOS Y PROCESOS CLAVES DE ASOCIACIONES UNIVERSITARIAS DE ESTUDIANTES

MÓDULO DE INVENTARIO

**TRABAJO DE INTEGRACIÓN CURRICULAR PRESENTADO COMO
REQUISITO PARA LA OBTENCIÓN DEL TÍTULO DE INGENIERO/A EN
SOFTWARE**

JUAN SEBASTIÁN POSSO CARRASQUILLA

juan.posso@epn.edu.ec

DIRECTOR: PhD. Sandra Patricia Sánchez Gordón

sandra.sanchez@epn.edu.ec

DMQ, agosto 2024

CERTIFICACIONES

Yo, Juan Posso declaro que el trabajo de integración curricular aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

Juan Sebastián Posso Carrasquilla

Certifico que el presente trabajo de integración curricular fue desarrollado por Juan Posso, bajo mi supervisión.

PhD. Sandra Patricia Sánchez Gordón
DIRECTOR

DECLARACIÓN DE AUTORÍA

A través de la presente declaración, afirmamos que el trabajo de integración curricular aquí descrito, así como el (los) producto(s) resultante(s) del mismo, son públicos y estarán a disposición de la comunidad a través del repositorio institucional de la Escuela Politécnica Nacional; sin embargo, la titularidad de los derechos patrimoniales nos corresponde a los autores que hemos contribuido en el desarrollo del presente trabajo; observando para el efecto las disposiciones establecidas por el órgano competente en propiedad intelectual, la normativa interna y demás normas.

JUAN SEBASTIÁN POSSO CARRASQUILLA

SANDRA PATRICIA SÁNCHEZ GORDÓN

VALERY ANTONELLA VALLEJO BARRIONUEVO

CRISTIAN ANDRÉS VERDUGA ACOSTA

JAVIER STEVEN REVELO QUIROZ

AGRADECIMIENTO

A mis padres, gracias por su apoyo incondicional en el transcurso de la carrera universitaria.

A mis dos mejores amigos, gracias por el aguante. A la persona que fue mi equipo y mi mejor amiga desde primer semestre hasta el fin de la carrera, siempre estuvo ahí cuando me tropezaba. No hace falta nombrarlos ellos saben perfectamente quienes son.

Al Javi, un grande.

A los profesores que nos apoyaron en el desarrollo de este trabajo, especialmente al Dr. Victor Velepucha y a la Dra. Sandra Sánchez.

ÍNDICE DE CONTENIDO

CERTIFICACIONES.....	I
DECLARACIÓN DE AUTORÍA.....	II
AGRADECIMIENTO.....	III
ÍNDICE DE CONTENIDO.....	IV
ÍNDICE DE TABLAS	VII
RESUMEN	VIII
ABSTRACT	IX
1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO.....	1
1.1 Objetivo general	1
1.2 Objetivos específicos	1
1.3 Alcance	2
1.4 Marco teórico	3
2 METODOLOGÍA.....	7
2.1 Proceso de desarrollo	7
2.2 Propuesta de heurísticas de usabilidad.....	8
2.3 Propuesta de normas de seguridad	9
2.4 Diseño de procesos.....	10
2.5 Roles del sistema	14
2.6 Descripción de perfiles de usuario	14
2.7 Gestión de requisitos.....	15
2.8 Diseño de la arquitectura	17
2.9 Diseño de base de datos.....	19
2.10 Prototipado.....	20
2.11 Desarrollo e implementación.....	21
2.12 Evaluación de usabilidad.....	27
2.13 Evaluación de seguridad.....	34
3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES.....	44
3.1 Resultados de entrevistas Think Aloud	44
3.2 Resultados encuesta SUS	46
3.3 Resultados encuesta NPS	49
3.4 Resultado encuesta de cumplimiento de heurísticas	50

3.5	Resultados evaluación con SortSite.....	52
3.6	Resultados de métricas servicios cloud	53
3.7	Resultado de Microsoft Defender for Cloud	54
3.8	Resultados SAST	56
3.9	Conclusiones.....	57
3.10	Recomendaciones.....	58
4	REFERENCIAS BIBLIOGRÁFICAS	59
5	ANEXOS.....	62
5.1	Anexo I. Tabla de heurísticas seleccionadas.	62
5.2	Anexo II. Tabla de controles de seguridad seleccionados.	62
5.3	Anexo III. Tabla de roles del sistema	62
5.4	Anexo IV. Perfiles de usuario	62
5.5	Anexo V. Desarrollo detallado del front-end y back-end	62
5.6	Anexo VI: Ejemplos de pruebas de integración y unitarias	63
5.7	Anexo VII. Repositorio que contiene el código fuente del proyecto	64
5.8	Anexo VIII. Tabla de objetivos a evaluar en cada heurística de usabilidad	64
5.9	Anexo IX. Formularios para la evaluación de usabilidad.....	65
5.10	Anexo X. Tabla de preguntas para evaluación de heurísticas.	65
5.11	Anexo XI. Resultado de evaluación de SortSite para las interfaces de productos y proveedores.....	65

ÍNDICE DE FIGURAS

Figura 1. Tablero Kanban en Clickup.	7
Figura 2. Diagrama de proceso de gestión administrativa de FEPON.	11
Figura 3. Diagrama de proceso de ingreso de un producto al inventario.	12
Figura 4. Diagrama de proceso para eliminar producto.	13
Figura 5. Diagrama de proceso venta de producto.	13
Figura 6. Diagrama de proceso de compra de un producto.	14
Figura 7. Parte del backlog de Kanban para el módulo de inventario.	17
Figura 8. Diagrama de arquitectura del sistema.	17
Figura 9. Diagrama base de datos.	20
Figura 10. Prototipo interfaz inventario.	21
Figura 11. Información organizacional.	22
Figura 12. Visualización de los módulos del sistema.	22
Figura 13. Visualización de los planes de aportación.	23
Figura 14. Visualización de los miembros administrativos.	23
Figura 15. Interfaz de inicio de sesión.	23
Figura 16. Interfaz principal de movimientos de inventario.	24
Figura 17. Interfaz principal de productos.	24
Figura 18. Interfaz principal de proveedores.	25
Figura 19. Interfaz de la página de ayuda.	25
Figura 20. Entrevistas think aloud con la presidenta de la federación.	30
Figura 21. Métricas de usabilidad configuradas en SortSite.	33
Figura 22. Roles implementados.	40
Figura 23. Políticas de acceso basado en roles.	41
Figura 24. Configuración SonarCloud.	44
Figura 25. Distribución términos recurrentes.	46
Figura 26. Puntaje promedio por enunciado encuesta SUS.	48
Figura 27. Gráfica de clasificación NPS.	49
Figura 28. Puntaje promedio encuesta de cumplimiento de heurísticas.	51
Figura 29. Resultado del análisis con SortSite para la interfaz de inventario.	53
Figura 30. Métricas de servidor SQL.	53
Figura 31. Métricas de API management.	54
Figura 32. Resumen general de Microsoft Defender for Cloud.	55
Figura 33. Resumen recomendaciones de Microsoft Defender for Cloud.	55
Figura 34. Resultados generales de análisis código estático.	56
Figura 35. Resultados de análisis código estático.	56

ÍNDICE DE TABLAS

Tabla 1. Épicas del componente de información de inventario.	15
Tabla 2. Historia de usuario Adición de un producto al inventario.....	15
Tabla 3. Interacción entre usuarios y módulos del sistema.....	28
Tabla 4. Enunciados de encuesta SUS.	30
Tabla 5 . Rating de puntaje SUS.	31
Tabla 6 . Clasificación encuesta NPS.....	31
Tabla 7. Clasificación de resultados NPS.....	32
Tabla 8. Rango de cumplimiento de heurísticas.	33
Tabla 9. Análisis de riesgo.....	38
Tabla 11. Promedio de puntaje por enunciado encuesta SUS.....	47
Tabla 12. Resultado rating SUS.	48
Tabla 13. Resultado NPS.	49
Tabla 14 . Resultados NPS.	50
Tabla 15. Puntaje promedio y rango de cumplimiento de heurísticas.	51

RESUMEN

Actualmente, la Federación de Estudiantes de la Escuela Politécnica Nacional carece de una plataforma centralizada que permita la captura y gestión eficiente de los productos en su inventario. Para ello, como parte de un sistema Cloud ERP, se propone utilizar un componente de inventario, diseñado para gestionar de manera integral los productos de una asociación estudiantil, diferenciando entre aquellos destinados a la venta y los que son bienes de la asociación. Este componente permite la recepción, verificación y clasificación de productos en sus respectivas categorías, asegurando la correcta actualización de información. Además, facilita la eliminación de productos por venta, donación o descarte, registrando los valores económicos de estas transacciones. La gestión organizada y transparente de este componente no solo mejora la eficiencia operativa, sino que también asegura la integridad de los registros, proporcionando una base sólida para la toma de decisiones informada y estratégica dentro de la asociación estudiantil. El componente tiene una base sólida de usabilidad, ya que sigue una combinación específica de guías de usabilidad seleccionadas especialmente para un sistema ERP. Además, se implementan las prácticas de seguridad óptimas, siguiendo el ciclo de desarrollo seguro del software. Como metodología de desarrollo, se utilizó Kanban, lo que permitió un desarrollo flexible y adaptado a las necesidades cambiantes del producto.

PALABRAS CLAVE: inventario, federación estudiantil, ERP, Kanban, usabilidad, seguridad, Cloud, ERP.

ABSTRACT

Currently, the Federation of Students of the National Polytechnic School lacks a centralized platform for efficient capture and management of inventory products. To address this, as part of a Cloud ERP system, it is proposed to use an inventory component designed to comprehensively manage the products of a student association, distinguishing between those intended for sale and those that are association assets. This component allows for the reception, verification, and classification of products into their respective categories, ensuring accurate information updates. Additionally, it facilitates the removal of products through sale, donation, or disposal, recording the economic values of these transactions. The organized and transparent management provided by this component not only improves operational efficiency but also ensures the integrity of records, providing a solid foundation for informed and strategic decision-making within the student association. The component has a solid usability foundation, following a specific combination of usability guides selected especially for an ERP system. Moreover, optimal security practices are implemented, following the secure software development lifecycle. Kanban was used as the development methodology, allowing for flexible development adapted to the product's changing needs.

Keywords: Inventory, student federation, ERP, Kanban, usability, security, Cloud ERP.

1 DESCRIPCIÓN DEL COMPONENTE DESARROLLADO

El componente de manejo de inventario del sistema Cloud ERP está diseñado para gestionar de manera integral los productos de una asociación estudiantil, diferenciando entre aquellos destinados a la venta y los que son bienes de la asociación. El proceso de ingreso de productos al inventario incluye la recepción, verificación y clasificación del producto en su respectiva categoría. Para los productos destinados a la venta, se añade la información correspondiente al sistema, se actualiza el stock y se registra la información necesaria para su integración con el módulo financiero. En el caso de productos que son bienes no comercializables, se verifica la existencia de una categoría correspondiente; si existe, se etiqueta y se añade la información al inventario, y si no, se crea una nueva categoría y se etiqueta de igual manera.

Este componente también permite eliminar productos del sistema ya sea por venta, donación o descarte, registrando los valores económicos de estas transacciones. De esta manera, se asegura un seguimiento preciso del estado y movimiento de los productos dentro de la asociación, facilitando una gestión eficiente del inventario.

El módulo de inventario garantiza una gestión organizada y transparente, permitiendo un control detallado y actualizado de los productos. Esto no solo mejora la eficiencia operativa, sino que también asegura la integridad de los registros, proporcionando una base sólida para la toma de decisiones informada y estratégica dentro de la asociación estudiantil.

1.1 Objetivo general

Desarrollar un módulo de inventario dentro del sistema Cloud ERP que optimice la gestión de productos de una asociación estudiantil, diferenciando claramente entre bienes destinados a la venta y aquellos que son activos de la asociación, asegurando un seguimiento preciso, actualización eficiente de información y una integración fluida con el módulo financiero, todo ello con el objetivo de mejorar la eficiencia operativa y la transparencia en la administración del inventario.

1.2 Objetivos específicos

- Garantizar un seguimiento preciso y eficiente de los productos en el inventario, facilitando la identificación, clasificación y registro adecuado de los artículos, ya sean destinados a la venta o como activos de la asociación.

- Agilizar y simplificar los procedimientos de ingreso de productos al inventario, asegurando la correcta actualización de la información en el sistema. De igual manera, proporcionar una vía efectiva para eliminar productos vendidos, donados o descartados, manteniendo la integridad y actualidad de los registros.
- Posibilitar un control exhaustivo de las transacciones comerciales asociadas al inventario, permitiendo el registro detallado de compras y ventas.

1.3 Alcance

Fase de requerimientos:

En esta etapa, se lleva a cabo un levantamiento minucioso de requisitos, detallando las necesidades específicas del módulo de inventario. Se realiza un análisis exhaustivo de las funciones y características necesarias para satisfacer las demandas financieras de las asociaciones universitarias.

Fase de diseño:

La fase de diseño se centra en la elaboración de diagramas de procesos fundamentales y prototipos utilizando la plataforma Figma. Además, se desarrolla un diagrama de arquitectura del sistema, proporcionando una visión clara de la estructura y relación de cada del módulo de inventario con los demás módulos del ERP.

Fase de desarrollo:

En esta fase se realiza la codificación de funcionalidades definidas en el diseño, creación de estructuras de datos específicas del inventario, y desarrollo de procesos y flujos de trabajo necesarios para el funcionamiento eficiente del módulo.

Fase de implementación:

La implementación del módulo de inventario incluye la configuración de la comunicación efectiva entre módulos para garantizar la integración en el sistema ERP. Se lleva a cabo una prueba inicial de la funcionalidad del módulo en conjunto con otros componentes para verificar la interoperabilidad y detectar posibles problemas de integración.

Fase de pruebas y evaluación:

Esta fase incluye pruebas de usabilidad, evaluación de rendimiento, pruebas de integración, ajustes y mejoras basados en los resultados obtenidos.

1.4 Marco teórico

En esta sección se presentan los conceptos clave que fundamentan el desarrollo y la implementación de un sistema cloud ERP, proporcionando un marco teórico comprensivo y detallado. Los sistemas cloud ERP representan una evolución significativa en la gestión de recursos empresariales, combinando la eficiencia y la flexibilidad de los entornos en la nube con las capacidades integradas de planificación de recursos empresariales. Para entender a fondo este sistema, es crucial explorar y entender una variedad de tecnologías y metodologías subyacentes que contribuyen a su diseño, desarrollo y operación.

A continuación, se enumeran los términos y conceptos que serán abordados en esta sección:

- Enterprise Resource Planning (ERP)
- Microservices Architecture
- Views Role-Based
- TypeScript
- React
- OAuth 2.0
- C#
- .NET Core
- Figma
- SAP Power Designer
- Azure DevOps
- Azure Data Studio
- Visual Studio Code
- Web App Services
- API Management
- Microsoft Defender for Cloud
- Azure Monitor

ERP

Enterprise Resource Planning (ERP) se refiere a sistemas de software integrados que manejan y automatizan muchos de los procesos empresariales asociados con las operaciones de producción y distribución de una empresa. Estos sistemas permiten la gestión de recursos, operaciones y relaciones con clientes mediante un único sistema cohesivo. el ERP facilita la gestión de productos, incluyendo el seguimiento de inventarios,

la actualización de información y la integración con otros módulos como el financiero para un control completo de los recursos. El objetivo principal es optimizar y mejorar la eficiencia de los procesos empresariales mediante la centralización de la información y la automatización de tareas repetitivas [1].

Microservices architecture

Es un estilo de arquitectura de software que estructura una aplicación como un conjunto de servicios acoplados libremente. Estos servicios son independientes, desplegados de manera autónoma y organizados alrededor de capacidades de negocio específicas, aplicado al módulo de inventario permite desarrollar y desplegar servicios independientes para la gestión de productos, actualización de stock y procesamiento de transacciones, mejorando la escalabilidad y flexibilidad del sistema. Los microservicios facilitan la implementación continua y la adaptación rápida a los cambios en los requisitos del negocio [2].

Views Role-Based

Las vistas basadas en roles son un mecanismo de control de acceso en sistemas informáticos donde los permisos y accesos a ciertos datos y funcionalidades están determinados por el rol del usuario dentro de la organización. Este enfoque garantiza que los usuarios solo puedan acceder a la información y ejecutar las tareas que son relevantes para sus responsabilidades, mejorando así la seguridad y la eficiencia operativa. Al centralizar y simplificar la gestión de permisos, se reducen los riesgos de acceso no autorizado y se mejora la auditoría y el cumplimiento normativo [17].

TypeScript

TypeScript es un lenguaje de programación que extiende JavaScript añadiendo tipado estático opcional. TypeScript permite a los desarrolladores detectar errores antes de ejecutar el código y mejora la productividad mediante características como autocompletado y refactorización [3].

React

React es una biblioteca de JavaScript para construir interfaces de usuario. Desarrollada por Facebook, React permite crear componentes reutilizables que gestionan su propio estado. Es conocida por su eficiencia en la actualización y renderización de los componentes cuando los datos cambian, lo cual permite que la visualización del estado del inventario se actualice en tiempo real en la vista del usuario [4].

OAuth 2.0

OAuth 2.0 es un protocolo de autorización que permite a las aplicaciones obtener acceso limitado a los recursos de usuario en un servidor HTTP. Simplifica el proceso de autorización para servicios web y aplicaciones móviles mediante la delegación de acceso sin compartir credenciales [5].

C#

C# es un lenguaje de programación orientado a objetos desarrollado por Microsoft como parte de su plataforma .NET. Es utilizado para desarrollar una amplia variedad de aplicaciones, desde aplicaciones de escritorio hasta servicios web y aplicaciones móviles [6].

.NET Core

.NET Core es una plataforma de desarrollo modular y multiplataforma para construir aplicaciones modernas, basadas en la nube y de alta rendimiento. Es una versión de .NET diseñada para soportar entornos de ejecución en Windows, Linux y macOS [7].

Figma

Figma es una herramienta de diseño colaborativo basada en la web, que permite a los equipos trabajar juntos en tiempo real en el diseño de interfaces de usuario. Proporciona herramientas para diseño vectorial, prototipado y generación de especificaciones para desarrolladores [8].

SAP Power Designer

SAP Power Designer es una herramienta de modelado y gestión de datos que soporta el diseño de arquitecturas empresariales, análisis de impacto y generación de documentación. Es utilizada para modelar datos, procesos y arquitecturas, y para gestionar cambios en estos modelos [9].

Azure DevOps

Azure DevOps es un conjunto de herramientas para el desarrollo de software que soporta la colaboración en proyectos, integración continua, entrega continua y gestión de versiones. Facilita el desarrollo ágil y mejora la productividad del equipo [10].

Azure Data Studio

Azure Data Studio es un entorno de desarrollo integrado (IDE) diseñado para gestionar, desarrollar y administrar bases de datos SQL en Azure y servidores locales. Ofrece

funcionalidades de administración de datos, desarrollo de consultas y visualización de datos [11].

Visual Studio Code

Visual Studio Code es un editor de código fuente ligero pero potente, desarrollado por Microsoft. Ofrece soporte para depuración, control de versiones Git, resaltado de sintaxis, autocompletado inteligente y extensiones que agregan funcionalidades adicionales [12].

Web App Services

Azure App Service es un servicio de computación basado en la nube para alojar aplicaciones web, APIs RESTful y back-ends móviles. Proporciona escalabilidad automática y alta disponibilidad, simplificando la gestión de infraestructura [13].

API management

Azure API Management es una plataforma que permite a las organizaciones publicar, asegurar, transformar, mantener y monitorear APIs. Facilita la gestión del ciclo de vida de las APIs y asegura el acceso mediante autenticación y autorización [14].

Microsoft Defender for Cloud

Microsoft Defender for Cloud es una plataforma de gestión de la postura de seguridad en la nube y protección contra amenazas. Proporciona herramientas para evaluar, mejorar y monitorear la seguridad de recursos en la nube y locales [15].

Azure monitor

Azure Monitor es un servicio que maximiza la disponibilidad y el rendimiento de aplicaciones y servicios mediante la entrega de una solución completa para la recopilación, análisis y actuación sobre datos de telemetría de entornos en la nube y locales, para el módulo de inventario se utiliza para monitorear el rendimiento y asegurar la disponibilidad continua del sistema. [16].

2 METODOLOGÍA

2.1 Proceso de desarrollo

Kanban es una metodología ágil que se enfoca en la gestión visual del trabajo y la mejora continua del flujo de trabajo. Se originó en la década de 1940 en la industria manufacturera de Toyota y se ha adaptado ampliamente al desarrollo de software y otros sectores. El objetivo principal de Kanban es incrementar la eficiencia, la visibilidad del trabajo en curso y la flexibilidad del equipo.

Kanban ofrece herramientas que apoyan la implementación de la metodología como las que se verán a continuación.

Tablero Kanban: Tablero visual que contiene el flujo de trabajo, este tablero se divide en tres columnas que representan distintos estados del desarrollo. Estos estados son: Pendientes (To do), en proceso (In progress) y por último completadas (Done).

Para el desarrollo del proyecto se usó una herramienta llamada Clickup que proporciona la herramienta del tablero de Kanban y otras funcionalidades adicionales como la de asignación de tareas, visualización de prioridad, estado de las tareas y tiempos de entrega como se puede apreciar en la Figura 1.

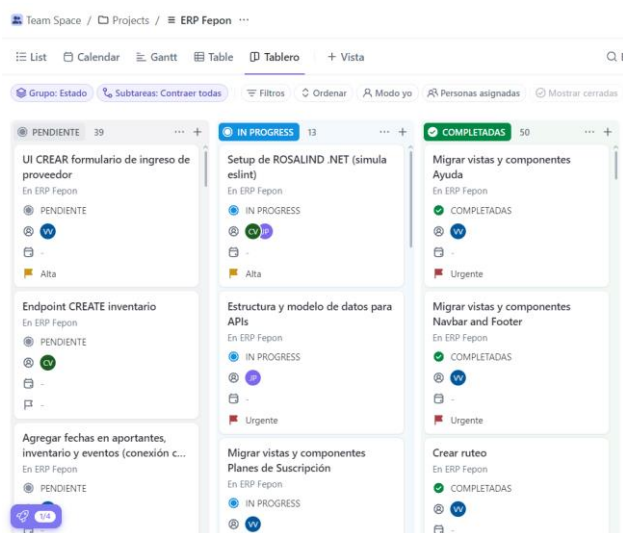


Figura 1. Tablero Kanban en Clickup.

Gracias a esta herramienta, el flujo de trabajo se hace completamente visible para todos los desarrolladores, fomentando la colaboración entre los miembros del equipo, en este caso al ser un módulo de un sistema ERP, se hace necesario visualizar el avance de los demás componentes para realizar la integración correcta con el componente de inventario.

2.2 Propuesta de heurísticas de usabilidad

Para garantizar que el componente de inventario propuesto provea una experiencia al usuario de calidad y cumpla con los principios de usabilidad, se decidió adoptar un enfoque integral utilizando como base cuatro guías de usabilidad ampliamente utilizadas.

Heurísticas de usabilidad de Nielsen

Nielsen provee una lista de diez principios generales para el diseño de interacción. Se les llama "heurísticas" porque son reglas generales y no pautas específicas de usabilidad. Los principios son bastante amplios y se aplican prácticamente a cualquier tipo de interfaz de usuario, incluidas las interfaces basadas en caracteres y las interfaces gráficas. Estos principios se utilizan como base para una inspección sistemática de una interfaz de usuario para encontrar sus problemas de usabilidad [18].

Heurísticas de Colombo & Pasch para una experiencia de usuario óptima

Presenta diez heurísticas para garantizar una experiencia de usuario óptima, derivadas de una inspección crítica de la teoría del flujo y aplicándola al contexto de la interacción humano-computadora. Las heurísticas están destinadas como pautas para los profesionales al diseñar y evaluar productos interactivos [19].

Norma ISO 9241 Parte 110: Principios de Interacción

Esta norma describe los principios de interacción y recomendaciones generales de diseño que son independientes de cualquier técnica de interacción específica y que son aplicables en el análisis, diseño y evaluación de sistemas interactivos. Estos principios de interacción y recomendaciones generales de diseño pueden guiar el desarrollo y la evaluación de interfaces de usuario, lo que conduce a una mejor usabilidad [20].

Evaluación de Bastien & Scapin de Interfaces Humano-Computador

Provee una manera de desarrollar métodos y herramientas que incorporen factores humanos en el proceso de diseño y evaluación de interfaces humano-computadora. Este diseño de criterios se considera como un medio para definir y operacionalizar dimensiones de usabilidad. Además, con respecto a la evaluación, representa una forma de mejorar la completitud y la explicitud del diagnóstico, de estandarizar el formato de la evaluación y de documentar mejor la evaluación. Los criterios ergonómicos de evaluación permiten identificar problemas de usabilidad, mejorar la experiencia de usuario y guiar el proceso de diseño de interfaces [21].

Heurísticas seleccionadas

En base a los estándares seleccionados para brindar una experiencia de usuario de calidad, haciendo un enfoque en las necesidades del sistema cloud ERP y de los usuarios finales que usarán el sistema, se han seleccionado diferentes heurísticas de las normas antes mencionadas, las cuales se detallan en la tabla que se encuentra en el enlace del Anexo I.

2.3 Propuesta de normas de seguridad

Para garantizar la protección adecuada de los datos confidenciales y la integridad del sistema, se adopta un enfoque integral utilizando como base tres guías de seguridad líderes en la industria: la norma ISO/IEC 27001, OWASP Top 10 y CWE Top 25.

Estas guías proporcionan un marco sólido para identificar, mitigar y gestionar los riesgos de seguridad de manera efectiva, asegurando así que sistema ERP cumpla con los estándares más altos de seguridad.

ISO/IEC 27001

Esta norma establece los requisitos para establecer, implementar, mantener y mejorar un sistema de gestión de la seguridad de la información (SGSI). Al seguir las directrices de ISO/IEC 27001, podemos asegurarnos de que nuestro sistema ERP en la nube cumpla con los estándares internacionales de seguridad de la información. Esto implica la adopción de políticas de seguridad claras, la asignación de roles y responsabilidades, la gestión de riesgos y la implementación de controles de seguridad adecuados [22].

OWASP Top 10

Esta lista recopila las diez vulnerabilidades más críticas en aplicaciones web, identificadas por la Open Web Application Security Project (OWASP). Las vulnerabilidades incluidas van desde problemas de inyección de código hasta la exposición de datos sensibles. Al adoptar las recomendaciones de OWASP Top 10, podemos abordar las vulnerabilidades más comunes y peligrosas que podrían afectar a nuestro sistema ERP en la nube, asegurando así su robustez y resistencia ante posibles ataques [23].

CWE Top 25

Esta lista enumera los 25 problemas de seguridad de software más comunes y críticos. Estas vulnerabilidades van más allá de las aplicaciones web y cubren una amplia gama de riesgos, desde inyecciones de código hasta desbordamientos de búfer. Al integrar las

mejores prácticas de CWE Top 25 en nuestro proceso de desarrollo y diseño de software, podemos identificar y mitigar eficazmente las vulnerabilidades que podrían comprometer la seguridad de nuestro sistema ERP en la nube [24].

Controles de seguridad seleccionados

Con base en las tres guías usadas como referentes de seguridad, se han establecido específicamente los siguientes puntos para evaluar la seguridad del Sistema Cloud ERP desarrollado, como se puede observar en la tabla que se encuentra en el enlace del Anexo II.

2.4 Diseño de procesos

El diseño de procesos es fundamental para comprender el funcionamiento actual de la Federación de Estudiantes y cómo puede integrarse el componente de inventario en sus actividades y procesos habituales. La Federación de Estudiantes de la Escuela Politécnica Nacional gestiona una variedad de tareas administrativas a través de diferentes módulos, cada uno de los cuales abarca aspectos cruciales como las finanzas, el inventario, la planificación de eventos y la información organizacional.

Los diagramas de procesos proporcionan una representación visual detallada de estos flujos de trabajo, permitiendo identificar áreas de mejora y optimización mediante la implementación del sistema Cloud ERP. El diagrama que se observa en la Figura 2, muestra como se integran los componentes del ERP.

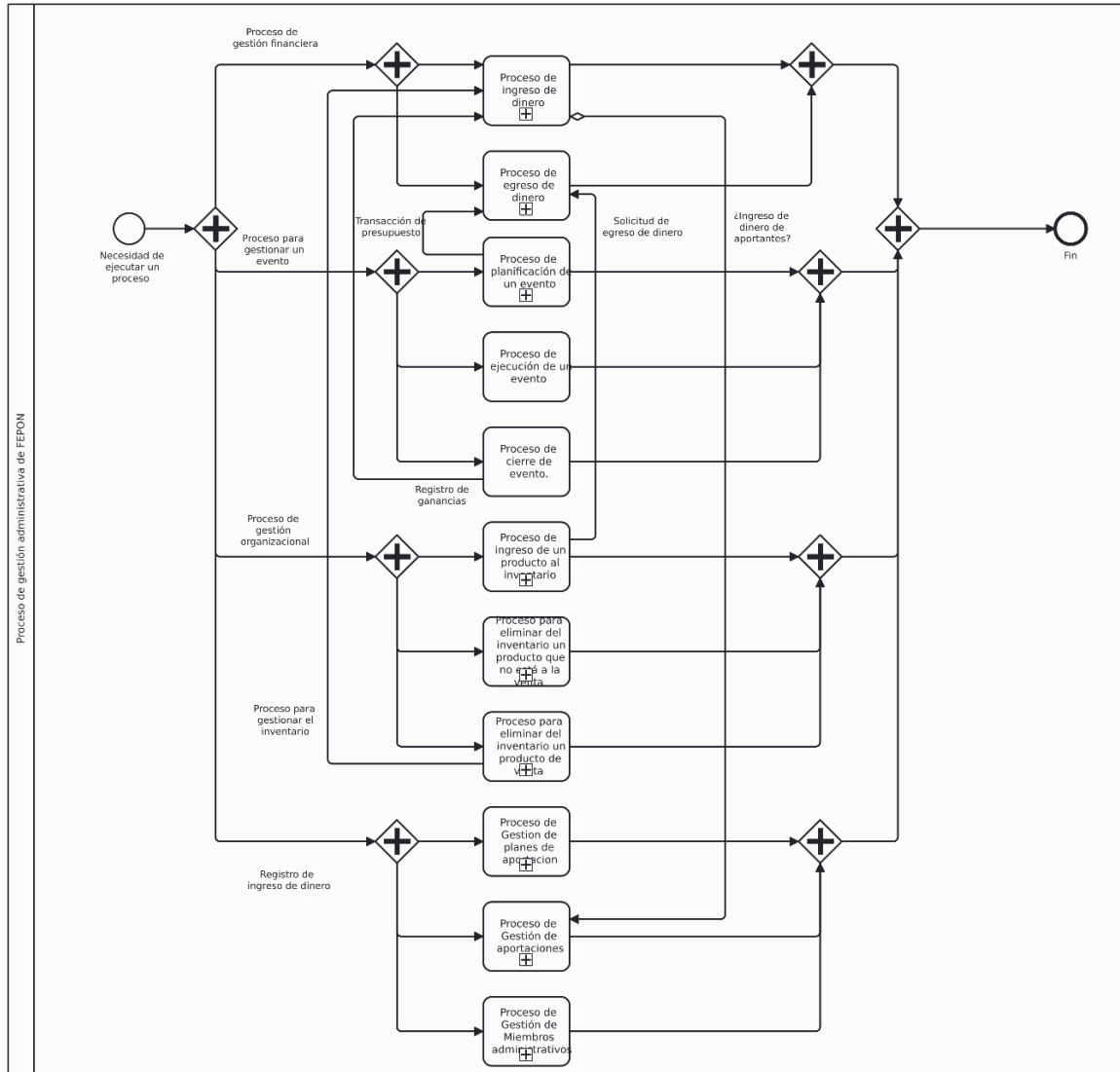


Figura 2. Diagrama de proceso de gestión administrativa de FEPOP.

Para realizar el ingreso de un nuevo producto, el encargado de inventario recibe el producto y verifica la categoría a la que pertenece. Si el producto es de venta, se procede a verificar si el producto ya existe en el sistema. Si el producto ya existe, se aumenta la cantidad de stock del producto en el sistema. Si el producto no existe, se añade la información del nuevo producto en el sistema y se procede a ponerlo a la venta. Si el producto no es de venta, se verifica si la categoría del producto existe en el sistema. Si la categoría no existe, se añade la nueva categoría al sistema y luego se añade la información del producto y se etiqueta. Finalmente, el proceso concluye. El flujo de trabajo asegura que cada producto que ingresa al inventario esté correctamente categorizado, registrado y etiquetado, facilitando su gestión y posterior venta si corresponde. Este proceso se puede observar en la Figura 3.

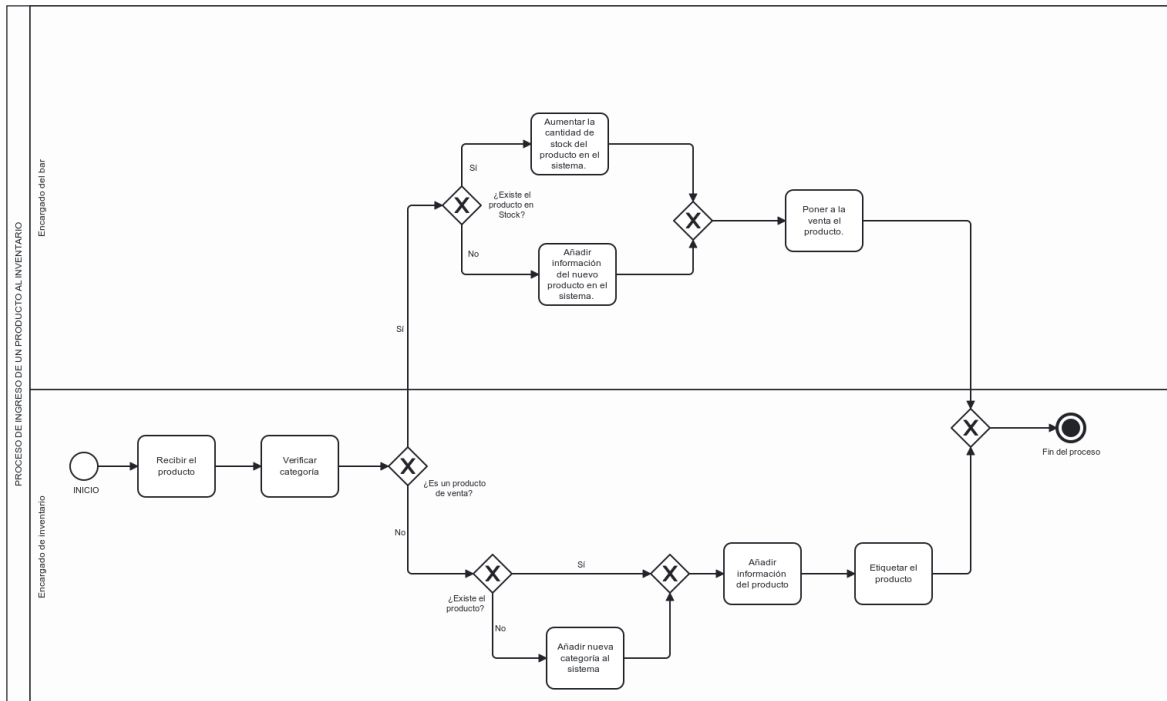


Figura 3. Diagrama de proceso de ingreso de un producto al inventario.

En el diagrama de procesos para eliminar del inventario un producto que no está a la venta se observa el flujo de actividades que se realiza actualmente para manejar productos obsoletos o no vendidos. Para iniciar el proceso, el encargado de inventario verifica el estado del producto. Si el producto se encuentra en buen estado, se procede a buscar lugares donde se pueda donar el producto. Una vez seleccionado el lugar, se entrega el producto a dicho lugar. Si el producto no se encuentra en buen estado, se procede a desecharlo. En ambos casos, ya sea después de la donación o desecho, se cambia el estado del producto a inactivo en el sistema para completar el proceso. El flujo de trabajo de trabajo asegura que los productos no vendidos sean manejados de manera adecuada, ya sea donándolos si están en buen estado o desechándolos si no lo están, y que el sistema de inventario se mantenga actualizado con el estado correcto de cada producto. Este proceso se puede observar en la Figura 4.

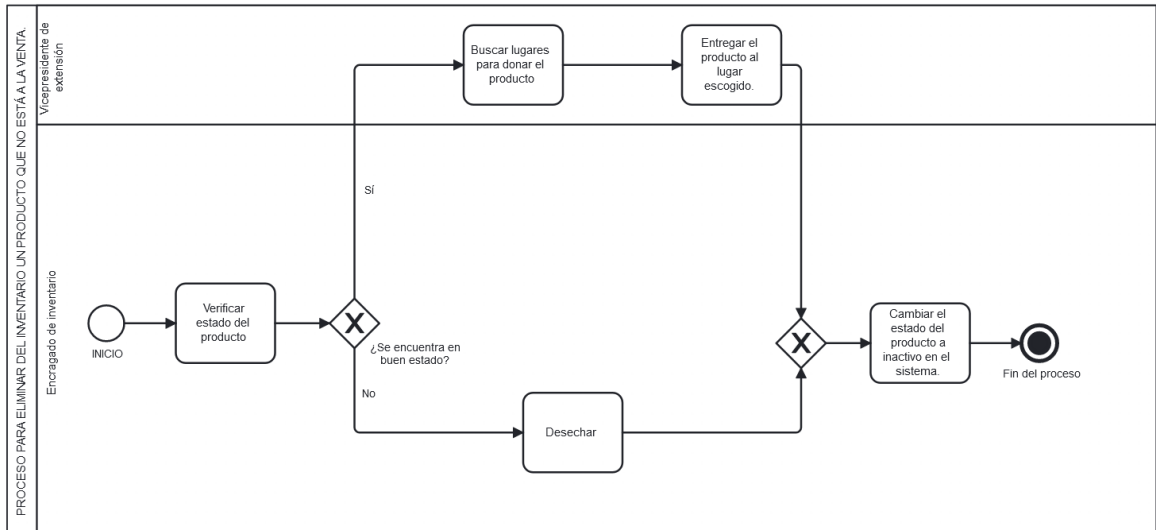


Figura 4. Diagrama de proceso para eliminar producto.

Para iniciar el proceso de venta de un producto, el encargado de inventario verifica la disponibilidad del producto en stock. Si el producto está disponible, se procede a entregarlo al comprador. Luego, se registra el valor del ingreso económico generado por la venta. A continuación, se disminuye la cantidad de stock del producto en el sistema. Finalmente, el proceso concluye. Si el producto no está disponible, se registra la falta de producto, y el proceso también finaliza. El flujo de trabajo asegura que las ventas se registren adecuadamente, el inventario se actualice de manera precisa, y cualquier falta de producto se documente para futuros análisis y acciones correctivas. Este proceso se puede observar en la Figura 5.

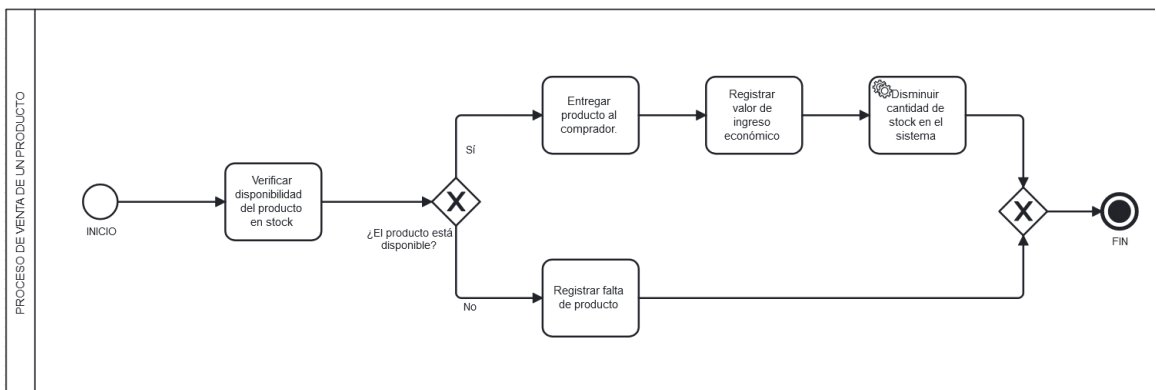


Figura 5. Diagrama de proceso venta de producto.

Por último, para iniciar el proceso de compra de un producto, se contacta al proveedor del producto. Una vez realizado el contacto, se recibe el producto. A continuación, se registra el gasto económico del producto. Luego, se verifica la categoría del producto. Si la categoría ya existe en el sistema, se añade la información del producto. Si la categoría no existe, se añade la nueva categoría al sistema antes de registrar la información del

producto. Finalmente, se guarda el producto en la estantería correspondiente. Este flujo de trabajo asegura que cada producto adquirido sea correctamente registrado, categorizado y almacenado, facilitando su gestión y posterior utilización o venta. Este proceso se puede observar en la Figura 6.

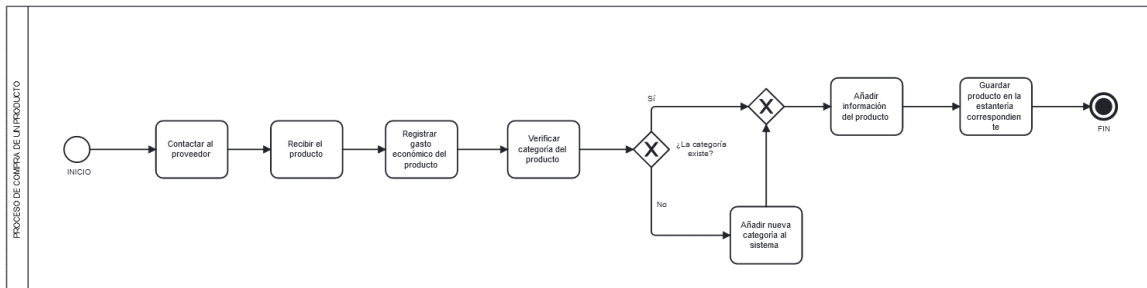


Figura 6. Diagrama de proceso de compra de un producto.

2.5 Roles del sistema

Los roles del sistema se definen en función de las responsabilidades y necesidades que tengan los distintos miembros administrativos dentro de la federación estudiantil. Cada rol accede a diferentes componentes del sistema ERP según sus funciones.

En cuanto al uso del sistema identifica cinco perfiles de usuario, los cuales son: presidente, vicepresidente, director, vocal, y administrador. Estos perfiles de usuario son los encargados de utilizar el sistema y dependiendo de que departamento sea asignado a un miembro administrativo en específico, este tendrá un acceso distinto.

En la tabla que se puede encontrar en el enlace del Anexo III se detalla los roles y como éstos se relacionan con los componentes del sistema.

2.6 Descripción de perfiles de usuario

Es fundamental describir detalladamente los perfiles de usuario para poder enfocar de manera precisa los requisitos del sistema y comprender plenamente las necesidades de cada uno. Conocer los roles y responsabilidades específicos de cada usuario permite diseñar soluciones personalizadas que optimicen su desempeño y faciliten su trabajo diario. Esto no solo mejora la usabilidad del sistema, sino que también garantiza que todas las funciones críticas estén alineadas con los objetivos y expectativas de los diferentes actores dentro de la federación.

Para ello se realiza una descripción detallada de cada perfil de usuario. Se incluyen aspectos clave como las responsabilidades, habilidades de liderazgo, conocimientos técnicos, personalidad, áreas de influencia, nivel de acceso y posibles problemas. Esta

información es esencial para comprender mejor el rol del usuario dentro de la organización y su interacción con los diferentes departamentos y equipos. El detalle del diseño de cada perfil de usuario se puede encontrar en el enlace del Anexo IV.

2.7 Gestión de requisitos

Para la toma de requisitos se realizan del componente de inventario, las épicas se centran en el manejo y registro de movimientos de los productos que forman parte del inventario de la asociación, las mismas están definidas y se muestran en la Tabla 1.

Tabla 1. Épicas del componente de información de inventario.

No	Épica
1	IN-Verificación de categoría del producto
2	IN - Adición de un producto al inventario
3	IN-Visualización de inventario
4	IN-Registro de venta de un producto
5	IN-Compra de un producto
6	IN-Actualización de un producto en el inventario

Posteriormente, cada una de las épicas, se describe con el formato de historia de usuario, junto con sus criterios de aceptación. En la Tabla 2 se muestra el ejemplo para la épica de IN - Adición de un producto al inventario

Tabla 2. Historia de usuario Adición de un producto al inventario.

Descripción	Criterios de aceptación
Como encargado del inventario, quiero tener la capacidad de agregar nuevos productos al sistema. Para mantener actualizado el inventario con la información más reciente sobre los productos disponibles.	<ol style="list-style-type: none"> 1. Debe existir una opción clara y accesible en la interfaz de usuario que permita agregar un nuevo producto al inventario. 2. La aplicación debe solicitar información esencial para el producto, como nombre, descripción, cantidad disponible, precio, y cualquier atributo relevante. 3. Se deben validar los campos obligatorios para garantizar que la

	<p>información es completa antes de agregar el producto al inventario.</p> <p>4. La aplicación debe generar automáticamente un identificador único para cada nuevo producto agregado al inventario.</p> <p>5. La interfaz de usuario debe proporcionar retroalimentación inmediata después de agregar un nuevo producto, confirmando el éxito de la operación o informando sobre posibles errores.</p> <p>6. Se debe garantizar que la información del nuevo producto se refleje correctamente en la lista de inventario.</p> <p>7. Implementar un sistema de registro o historial que registre la fecha y hora de la adición de cada producto al inventario, proporcionando un seguimiento detallado de las actividades realizadas.</p>
--	--

De igual forma, esta épica se divide en las subtarear que sean necesarias para completar la tarea general, estas añaden a los tickets de desarrollo general en el backlog de tickets Kanban definidos en Clickup, como se observa en la Figura 7.

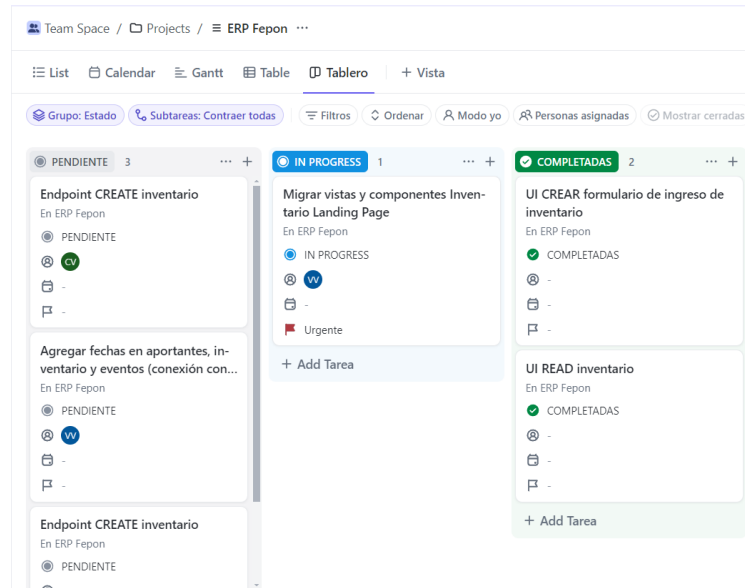


Figura 7. Parte del backlog de Kanban para el módulo de inventario.

2.8 Diseño de la arquitectura

En el diseño de la arquitectura de un sistema ERP en la nube, la adopción de una arquitectura en tres capas es fundamental para asegurar la modularidad, escalabilidad y mantenimiento del sistema.

La arquitectura en tres capas se divide en capa de presentación (front-end), capa de lógica de negocio (back-end) y capa de datos (database). Cada una de estas capas tiene responsabilidades claramente definidas y se comunican entre sí a través de una API Gateway. Este enfoque permite una estructura más "headless", donde el front-end no contiene lógica de negocio y se relaciona con el back-end a través de APIs específicas según sea necesario. El diagrama de la arquitectura se puede observar en la Figura 8.

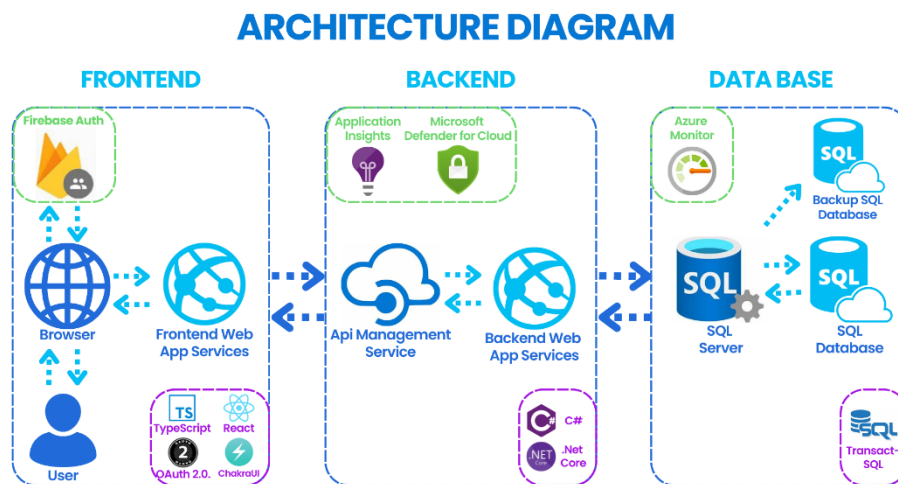


Figura 8. Diagrama de arquitectura del sistema.

Descripción de la arquitectura

1. Front-end:

- **Navegador:** Actúa como el medio de acceso al sistema ERP desde el lado del cliente, enviando solicitudes HTTPS a los servicios de front-end.
- **Servicios de aplicaciones web front-end:** Gestionan la interfaz de usuario basado en roles, manejan las interacciones y se comunican con el back-end a través de la API Gateway. No contienen lógica de negocio.
- **Firebase:** Servicio que gestiona la identidad y el acceso de los usuarios mediante autenticación y autorización, asegurando que solo los usuarios autorizados puedan acceder al sistema y maneja los roles de los usuarios según la lógica del negocio.
- **Registro de aplicaciones:** Administra las aplicaciones registradas, permitiendo la configuración de políticas de acceso y permisos para diferentes aplicaciones, es utilizada para configurar la relación de los usuarios y sus permisos en la aplicación.

2. Back-end:

- **API Gateway (servicio de gestión de API):** Actúa como un punto de entrada unificado para las solicitudes de los clientes, dirigiéndolas a los servicios de back-end apropiados. Proporciona funcionalidades como autenticación, autorización, gestión de tráfico, monitoreo y análisis.
- **Azure API Management:** Plataforma de gestión de API que permite publicar, asegurar, transformar, mantener y monitorizar APIs de manera eficiente.
- **Servicios de aplicaciones web back-end:** Contienen la lógica de negocio y se encargan del procesamiento de las solicitudes, ejecutando las operaciones requeridas y gestionando la comunicación con la base de datos.
- **Application insights:** Herramienta de monitoreo y análisis que proporciona visibilidad sobre el rendimiento de las aplicaciones, permitiendo detectar y diagnosticar problemas rápidamente.
- **Microsoft Defender for Cloud:** Ofrece seguridad y protección a las aplicaciones en la nube mediante la detección de amenazas y la implementación de mejores prácticas de seguridad.

3. Base de datos:

- **Servidor SQL (SQL Server):** Sistema de gestión de bases de datos relacional. Proporciona una plataforma de datos confiable y escalable. Suministra el servicio para gestionar las bases de datos utilizadas en el sistema.
- **Base de datos SQL:** Almacena y gestiona los datos del ERP, incluyendo información de clientes, inventarios, órdenes, entre otros.

- **Backup SQL database:** Servicios y estrategias de copia de seguridad para asegurar la disponibilidad y recuperación de datos en caso de fallos o pérdidas.
- **Azure monitor:** Servicio de monitoreo de infraestructura que proporciona indicadores sobre el rendimiento y la integridad del sistema, permitiendo la detección proactiva de problemas y la optimización de recursos.

La arquitectura en tres capas descrita proporciona una estructura robusta y flexible para el sistema ERP en la nube. La separación de responsabilidades entre front-end, back-end y base de datos, junto con el uso de un API Gateway, permite una mayor modularidad, escalabilidad y seguridad.

Además, la integración de herramientas de monitoreo y seguridad asegura un alto rendimiento y protección de los datos y aplicaciones. Esta arquitectura facilita la implementación de nuevas funcionalidades y la adaptación a las necesidades cambiantes del negocio.

2.9 Diseño de base de datos

Una vez obtenidos los requerimientos y la arquitectura del sistema, se diseña la arquitectura de la base de datos a utilizar en el Sistema Cloud ERP.

El componente de inventario este utiliza siete tablas relacionadas entre sí. Estas relaciones comienzan por la tabla Product la cual almacena información sobre los productos, y está vinculada a la tabla Providers, que representa a los proveedores del producto. También está relacionada con la tabla Category, que indica la categoría a la que pertenece cada producto, y con la tabla State, que muestra el estado actual del producto. Adicionalmente, los movimientos de inventario se registran en la tabla Inventory_Movement, la cual asocia cada movimiento a un producto específico. Estos movimientos pueden o no incluir una transacción, representada en la tabla Transaction, dependiendo del tipo de movimiento asignado en la tabla Inventory_Movement_Type. El diagrama de la base de datos se puede observar en la Figura 9.

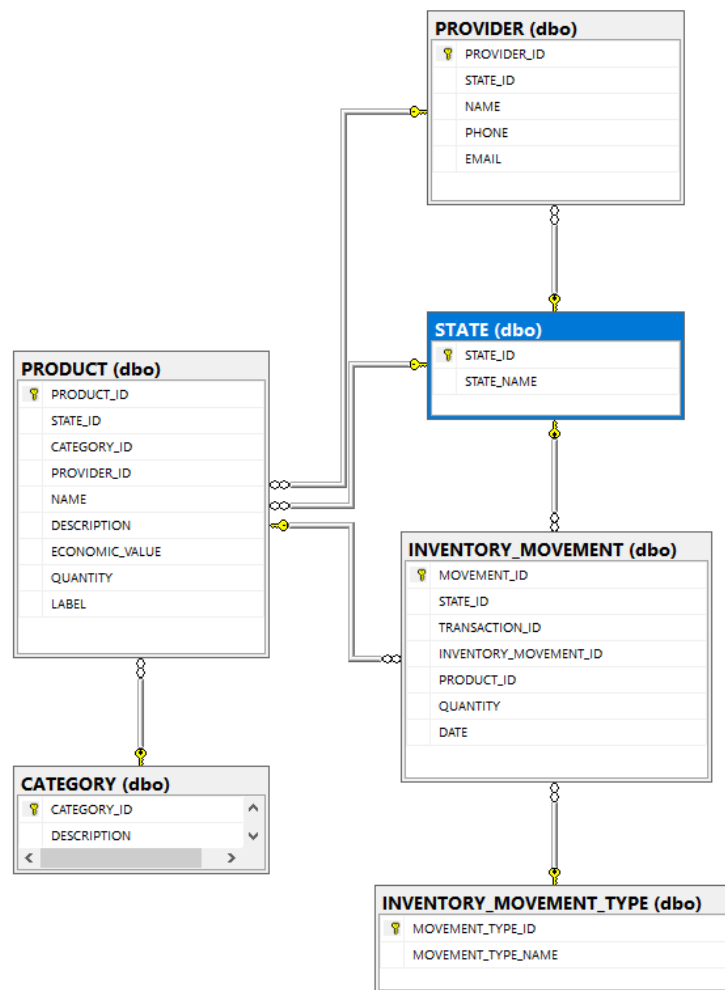


Figura 9. Diagrama base de datos.

2.10 Prototipado

Para desarrollar los prototipos, se utilizó la herramienta Figma, en la cual se realizaron diseños de las interfaces principales. Entre ellas se encuentran el formulario de inicio de sesión y los formularios de ingreso. Además, se utilizaron colores sobrios que se relacionan principalmente con los colores de la marca de la federación que se utilizó como caso de estudio, la Federación de Estudiantes de la Escuela Politécnica Nacional, como lo son el gris y azul. Para el diseño de componentes de las páginas principales de los componentes de información organizacional, eventos, financiero e inventario, se tomaron en cuenta las heurísticas que serán utilizadas durante las pruebas de usabilidad, las cuales se detallaron en la sección previa respectiva. Es por ello por lo que se buscaron diseños atractivos para el usuario, pero que a su vez sean sencillos y, que cuenten elementos y un lenguaje que faciliten el manejo del sistema.

Interfaz inventario

Esta interfaz está destinada a la visualización de los productos existentes en el inventario de la federación. Dentro de los elementos importantes que se pueden observar en esta ventana se encuentra una tabla con el detalle de los productos, su cantidad, categoría, precio, entre otros detalles. También se encuentra los botones para poder descargar la información en formato PDF y para poder añadir un nuevo producto, como se observa en la Figura 10.

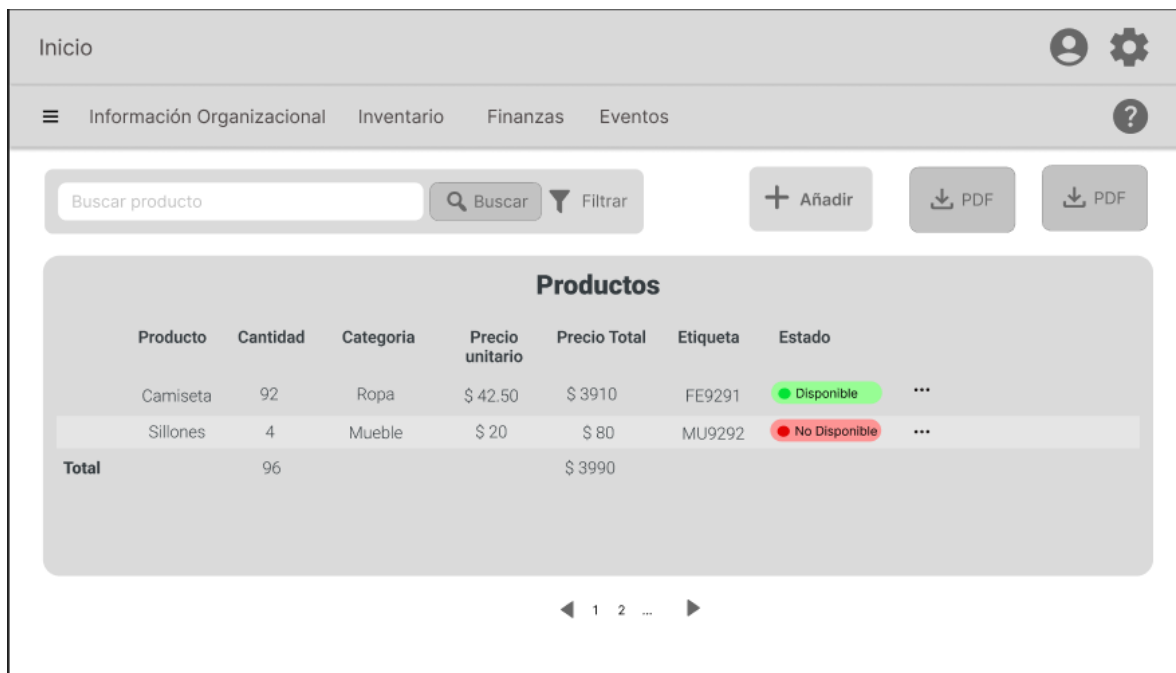


Figura 10. Prototipo interfaz inventario.

2.11 Desarrollo e implementación

En esta sección se detalla el desarrollo de las interfaces de usuario de la aplicación, así como la configuración y el desarrollo de la lógica del servidor. Se explora cómo se diseñaron y desarrollaron las funcionalidades clave, la integración de servicios y la validación de datos, proporcionando una visión comprensiva de las estrategias y herramientas utilizadas para llevar a cabo este proyecto.

Front-end

Para el desarrollo del front-end de la aplicación, se utiliza el framework ReactJS junto con TypeScript, lo que garantiza un desarrollo más seguro y con tipado estático. Además, se emplea ChakraUI como biblioteca de componentes para crear una interfaz de usuario moderna y accesible, y Yup como validador de esquemas para asegurar la correcta validación de los datos en los formularios. Es importante destacar que cada una de las

interfaces tiene autorización basada en el rol del usuario que inicia sesión, mostrando o no ciertas funcionalidades según su rol.

La aplicación consta de seis interfaces principales por las cuales el usuario puede navegar. En primer lugar, se encuentra la página principal de la aplicación, cuya interfaz presenta una barra de navegación, la información organizacional, donde se muestran la visión y misión y el logo de la federación, enlaces de acceso a los diferentes módulos del sistema, un resumen de los planes de aportación existentes, e información de los miembros administrativos de la federación, junto con un pie de página que contiene información de contacto y redes sociales de la misma. Las diferentes secciones de la página principal se pueden observar en las Figuras 11, 12, 13 y 14.



Figura 11. Información organizacional.

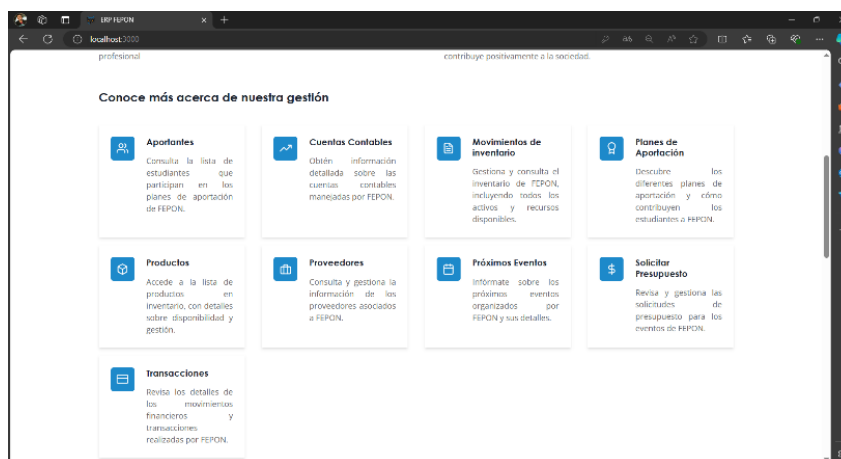


Figura 12. Visualización de los módulos del sistema.

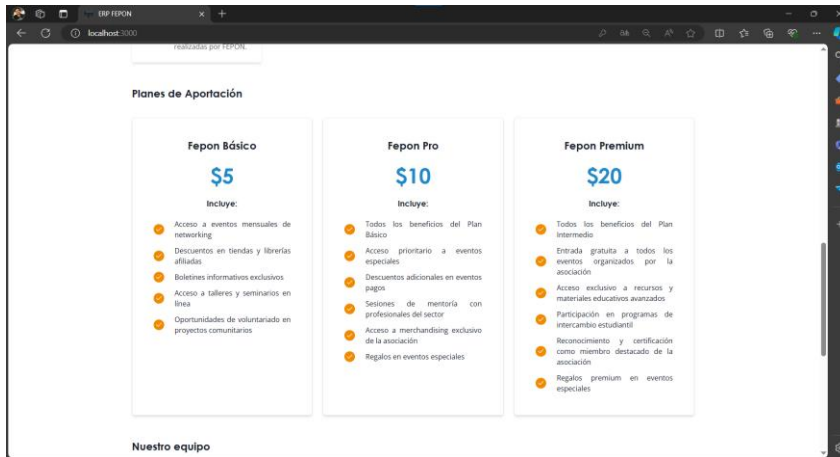


Figura 13. Visualización de los planes de aportación.

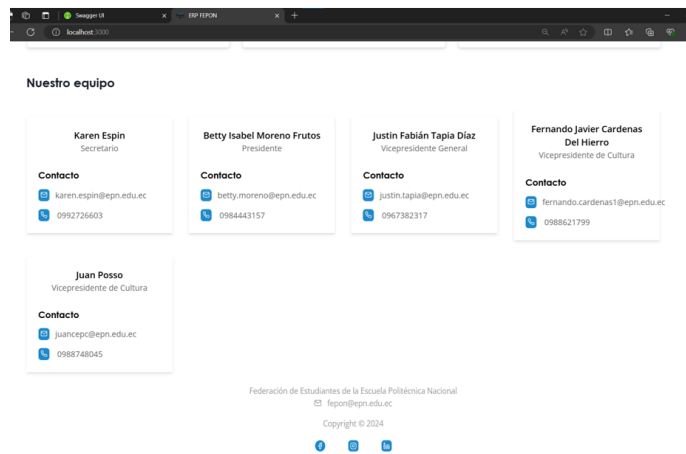


Figura 14. Visualización de los miembros administrativos.

Otra de las interfaces principales es la interfaz de inicio de sesión, la cual cuenta con un formulario donde los usuarios pueden ingresar su correo y contraseña para acceder a la plataforma, como se observa en la Figura 15.

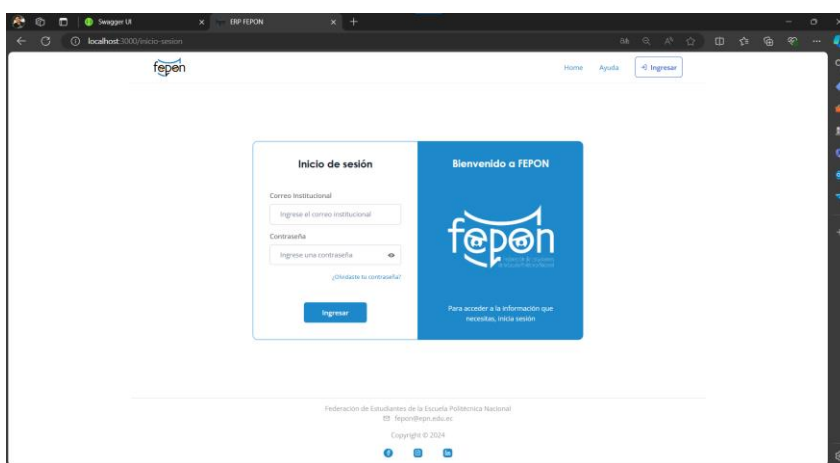


Figura 15. Interfaz de inicio de sesión.

Por otro lado, se encuentran las interfaces que representan el núcleo de la funcionalidad del aplicativo. Estas interfaces muestran las tablas de movimientos, productos y proveedores, cada una con formularios de campos validados que permiten añadir, eliminar o editar información de cada entidad representada, y la funcionalidad de poder descargar la información en Excel. Además, incluyen mensajes de éxito y error que proporcionan al usuario retroalimentación sobre las acciones realizadas en el sistema. En las Figuras 16, 17 y 18 se muestran las interfaces principales de la aplicación.

The screenshot shows the 'Movimientos de Inventario' page. It features a search bar, a '+ Movimiento' button, and a '- Excel' button. Below is a table with the following data:

	PRODUCTO	TIPO DE MOVIMIENTO	CANTIDAD	FECHA
<input checked="" type="checkbox"/>	Librería	VENTA	10	13/06/2024
<input checked="" type="checkbox"/>	Printer	DONACIÓN	8	11/07/2024
<input checked="" type="checkbox"/>	Printer	DONACIÓN	12	24/07/2024
<input checked="" type="checkbox"/>	Laptop	COMPRA	10	13/07/2024
<input checked="" type="checkbox"/>	Laptop	COMPRA	10	13/07/2024
<input checked="" type="checkbox"/>	Laptop	VENTA	10	13/07/2024
<input checked="" type="checkbox"/>	Laptop	VENTA	10	13/07/2024
<input checked="" type="checkbox"/>	Laptop	VENTA	10	13/07/2024

Figura 16. Interfaz principal de movimientos de inventario.

The screenshot shows the 'Productos' page. It features a search bar, a '+ Producto' button, and a '- Excel' button. Below is a table with the following data:

	NOMBRE	CATEGORÍA	DESCRIPCIÓN	PRECIO	CANTIDAD	ETIQUETA	PI
<input checked="" type="checkbox"/>	Librería	Categoría de muebles	Librería de madera con soportes ajustables	100	14	Furniture	P
<input checked="" type="checkbox"/>	carne	Categoría de alimentos	carne	3.5	7	carnet	P
<input checked="" type="checkbox"/>	Maceta	Categoría de muebles	Maceta de barro	10	8	Decoración	P
<input checked="" type="checkbox"/>	Mesa billar	Categoría de muebles	Es una mesa de billar	750	2	FEPON-1234	P

Federación de Estudiantes de la Escuela Politécnica Nacional
fepon@fepon.edu.ec

Figura 17. Interfaz principal de productos.

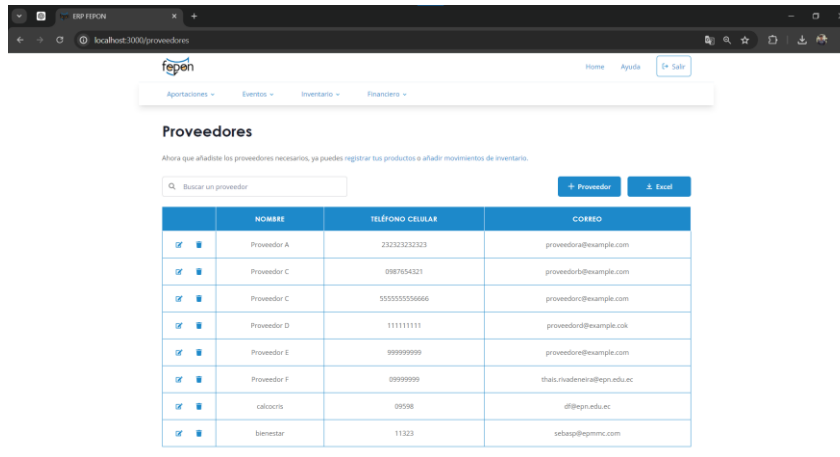


Figura 18. Interfaz principal de proveedores.

Finalmente, se encuentra la interfaz de ayuda, la cual como se muestra en la Figura 19 proporciona recursos y respuestas a preguntas frecuentes para ayudar a los usuarios a entender y utilizar mejor el sistema.

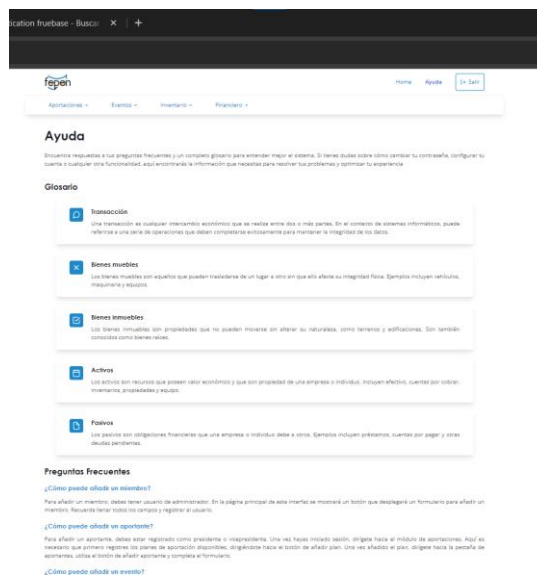


Figura 19. Interfaz de la página de ayuda.

Para obtener una descripción detallada del desarrollo del front-end dirigirse al documento del Anexo V.

Back-end

Para el back-end del aplicativo se desarrolló utilizando .NET Core y se implementaron cinco entidades de dominio, que se detallan a continuación:

- Category.
- InventoryMovement.
- InventoryMovementType.
- Product.
- Provider.

Cada una de estas entidades cuenta con una configuración de mapeo a las tablas de la base de datos para asegurar una correcta relación entre estas y los modelos que describen a las entidades. Además, se desarrollaron controladores para cada entidad que manejan las solicitudes HTTP según los requerimientos específicos. Cada endpoint de estos controladores está protegido por autenticación mediante un token JWT y tiene diferentes niveles de autorización según el caso.

El detalle de la implementación del back-end se encuentra disponible en el Anexo V.

Pruebas unitarias

Las pruebas unitarias se realizaron para cada API definida en el controlador. Estas pruebas utilizan Moq, una herramienta para crear mocks de las dependencias utilizadas en los controladores, y el framework xUnit para estructurar las pruebas. En el Anexo VI, se muestra un ejemplo de prueba para el método POST de la API de Inventory Movement, las demás pruebas de integración se pueden encontrar en el directorio UnitTestTIC del proyecto que se encuentra en el enlace del repositorio del backend del Anexo VII. En este ejemplo, se sigue el patrón Arrange-Act-Assert:

- **Arrange:** se llama a un método que limpia y configura el estado inicial de la base de datos de pruebas.
- **Act:** se ejecuta el método del controlador.
- **Assert:** se verifica que los resultados del método probado se comportan como se espera.

Pruebas de integración.

Para las pruebas de integración se utiliza React Testing Library y Jest, en el código mostrado en el Anexo VI, se verifica que la página InventoryMovementPage se renderiza correctamente. Además, simula un clic en el botón de adición, asegurándose de que se muestra un modal para editar un movimiento de inventario. Las demás pruebas de integración se realizan de manera similar y se encuentran en el directorio `__tests__` del repositorio que se encuentra en el enlace del repositorio del frontend del Anexo VII.

2.12 Evaluación de usabilidad

Para asegurar que el Sistema Cloud ERP cumpla con los estándares de usabilidad esperados por los usuarios finales, se ha diseñado un plan integral de evaluación de usabilidad. Esta sección describe los métodos y herramientas utilizados para llevar a cabo las pruebas, asegurando así una experiencia de usuario óptima y eficiente.

Preparación de la evaluación

1. Definición de objetivos

La definición clara de objetivos es fundamental para una evaluación de usabilidad efectiva. Estos objetivos establecen el marco sobre el cual se evaluará la experiencia del usuario en el Sistema Cloud ERP. Al identificar y definir específicamente qué aspectos de la usabilidad deseamos medir y mejorar, podemos enfocar nuestros esfuerzos en áreas críticas que impacten positivamente la experiencia del usuario y la eficiencia operativa.

Anteriormente se había definido cuatro guías de usabilidad, de las cuales se han seleccionado 13 heurísticas que serán la base de los parámetros a evaluar. Las cuales se muestran en la tabla que se encuentra en el enlace del Anexo VIII.

Con la aplicación de estas heurísticas de usabilidad, se espera obtener resultados positivos en cada una de las evaluaciones aplicadas a los usuarios finales, asegurando así, una buena experiencia del usuario y la capacidad de que puedan ejecutar tareas y escenarios sin conflictos.

2. Identificación de usuarios representativos

Para obtener una evaluación precisa y relevante de la usabilidad del Sistema Cloud ERP, es crucial identificar a los usuarios que mejor representen a la base de usuarios finales.

Cada usuario tiene necesidades y expectativas únicas, por lo que seleccionar una muestra diversa pero representativa nos permitirá obtener resultados valiosos sobre cómo diferentes grupos interactúan con el sistema y perciben su usabilidad.

Anteriormente se hizo la definición de perfiles y roles, detallando cada una de las tareas que pueden desempeñar en el Sistema Cloud ERP, y en concordancia con las actividades desempeñadas en su respectiva organización, sea asociación o federación estudiantil.

Para nuestro caso de estudio, los usuarios seleccionados para participar en las evaluaciones de usabilidad corresponden a miembros activos de la Federación de Estudiantes de la Escuela Politécnica Nacional (FEPON).

A continuación, en la Tabla 3 se detallan los nombres, cargos y módulo con el que interactúa cada uno de los usuarios finales seleccionados.

- Srta. Betty Moreno (presidenta de FEPON 2023-2025)
- Sr. Justin Tapia (vicepresidente de FEPON 2023-2025)
- Srta. Karen Espín (secretaria de FEPON 2023-2025)
- Srta. Pamela Cárdenas (vicepresidenta Financiera de FEPON 2023-2025)
- Srta. Thais Rivadeneira (directora Financiera de FEPON 2023-2025)
- Sr. Fernando Cárdenas (director de Cultura de FEPON 2023-2025)

Tabla 3. Interacción entre usuarios y módulos del sistema.

	Administrador	Financiero	Inventario	Gestión Organizacional	Eventos
Presidenta	X	X	X	X	X
Vicepresidente	X	X	X	X	X
Secretaria				X	
V. Financiera		X	X		
D. Financiera		X	X		
D. de Cultura					X

3. Métodos y herramientas de evaluación:

La elección adecuada de métodos y herramientas de evaluación es esencial para obtener datos significativos y precisos sobre la usabilidad del Sistema Cloud ERP. Cada método utilizado aporta perspectivas únicas que nos ayudarán a entender cómo los usuarios interactúan con el sistema y qué aspectos específicos de la usabilidad necesitan mejorarse.

1. Think aloud

Para aplicar este método de evaluación en primer lugar se define la serie de tareas que el usuario debe realizar en el sistema. Posteriormente se le solicita al usuario que verbalice sus pensamientos a medida que navega por la interfaz de usuario, realizando las tareas antes definidas. Este método permite identificar problemas de usabilidad, confusiones y áreas de mejora en la interfaz de usuario.

A continuación, se detallan las tareas establecidas para los usuarios:

1. Iniciar sesión con credenciales de vicepresidente, presidente o secretaria.
2. Dirigirse a movimientos de inventario en el módulo inventario.
3. Inspeccionar la página de movimientos.
4. Añadir un movimiento de inventario (el usuario debe notar que es necesario añadir primero un producto y un proveedor.
5. Editar un movimiento de inventario.
6. Eliminar el movimiento de inventario.
7. Editar un producto.
8. Eliminar un producto.
9. Editar un proveedor.
10. Eliminar un proveedor.
11. Descargar información de cualquiera de las tablas.

Las entrevistas se llevaron a cabo de manera virtual con las personas que conforman la directiva de la federación de estudiantes. Un ejemplo de ello se muestra en la Figura 20, donde se observa una parte de la entrevista con la presidenta de la federación.

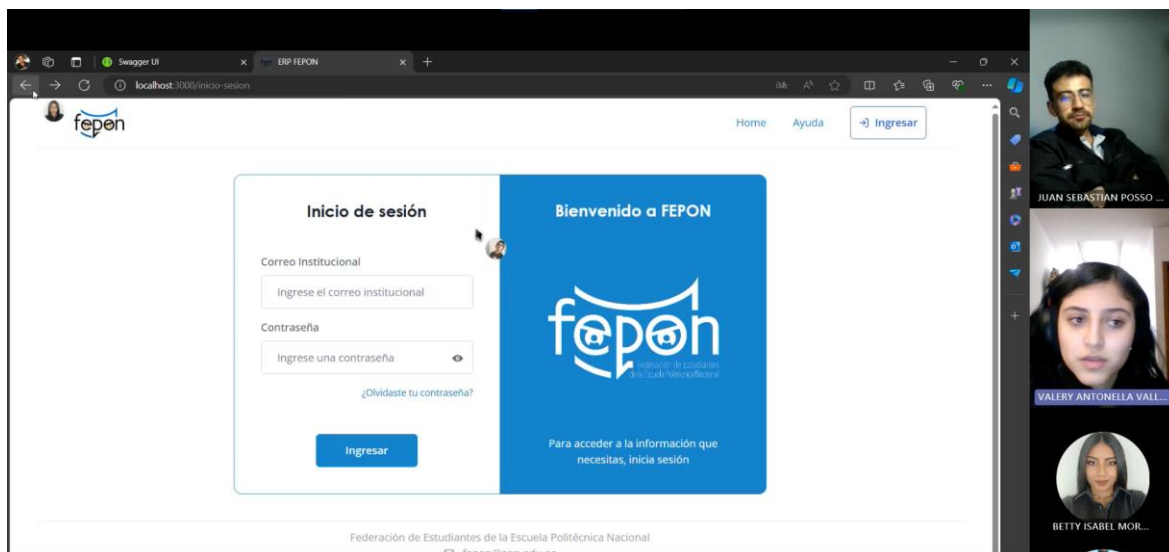


Figura 20. Entrevistas think aloud con la presidenta de la federación.

2. Encuesta SUS

Una vez el usuario ha realizado las tareas antes definidas utilizando el Think Aloud, se realiza la encuesta SUS, la cual es un cuestionario diseñado para medir la usabilidad percibida de un sistema, esta consiste en diez afirmaciones relacionadas con la experiencia del usuario, las mismas que se pueden observar en la Tabla 4.

Tabla 4. Enunciados de encuesta SUS.

Enunciado
Creo que me gustaría usar este sistema con frecuencia.
Encontré el sistema innecesariamente complejo.
Creo que el sistema es fácil de usar.
Creo que necesitaría la ayuda de una persona con conocimientos técnicos para poder usar este sistema.
Encontré que las diversas funciones del sistema estaban bien integradas.
Pensé que había demasiada inconsistencia en este sistema.
Imagino que la mayoría de las personas aprenderían a usar este sistema muy rápidamente.
Encontré el sistema muy engorroso de usar.
Me sentí muy seguro/a usando el sistema.
Necesité aprender muchas cosas antes de poder desenvolverme con el sistema.

Para la evaluación el usuario puntúa las afirmaciones mostradas en la Tabla 4, en una escala del 1 al 5 donde 1 significa “Totalmente en desacuerdo” y 5 “Totalmente de acuerdo”. Posteriormente para obtener el SUS Score se sigue el siguiente proceso:

- Sumar las respuestas de los enunciados impares y después restar 5
- Sumar las respuestas de los enunciados pares y restar ese total a 25
- Sumar ambos resultados y multiplicarlo por 2,5.

Al realizar el proceso mencionado se obtiene un valor de SUS Score entre 0 y 100, este valor categoriza al nivel de usabilidad del sistema de acuerdo con la Tabla 5.

Tabla 5 . Rating de puntaje SUS.

SUS Score	Rating
> 80.3	Excelente
68 – 80.2	Bueno
67	Aceptable
51 – 66	Pobre
< 51	Terrible

Para aplicar la encuesta SUS, se creó un formulario utilizando la herramienta Microsoft Forms, mismo que se encuentra disponible en el enlace del Anexo IX.

3. Net Promote Score (NPS)

Adicionalmente, se utilizó el NPS para evaluar el nivel de lealtad y de satisfacción de los usuarios con el sistema a través de la pregunta: "En una escala del 0 al 10, ¿cuán probable es que recomiende este producto a un amigo o colega?"

En base a la respuesta del usuario, este se clasifica según se observa en la Tabla 6.

Tabla 6 . Clasificación encuesta NPS

Clasificación	Descripción	Puntuación
Promotor	Usuarios satisfechos y leales que probablemente recomendarán tu producto o servicio.	9-10
Pasivo	Usuarios satisfechos pero que no recomendarían el producto.	7-8
Detractor	Clientes insatisfechos que podrían afectar el producto a través de comentarios negativos.	0-6

Posteriormente se calcula el NPS utilizando la siguiente fórmula:

$$nps = \%promotores - \%detractores$$

Posteriormente se realiza una comparación del NPS relativa a otros productos del sector de software, en la cual el promedio varía entre 27 y 70 por lo cual se fija la escala que se observa en la Tabla 7.

Tabla 7. Clasificación de resultados NPS.

Clasificación	Descripción	Puntuación
Grave	El sistema tiene más detractores que promotores. Necesita mejoras urgentes para aumentar la satisfacción del usuario.	< 0
Insatisfactorio	Predominan los detractores sobre los promotores. Es necesario abordar problemas importantes para mejorar la experiencia del usuario.	0 – 27
Moderadamente positivo	Los usuarios están generalmente satisfechos, aunque hay oportunidades de mejora.	28 – 54
Excelente	La mayoría de los usuarios son promotores y están muy satisfechos con el sistema. Existen muy pocas áreas de mejora.	55 - 100

De manera similar a la encuesta SUS, se creó un formulario utilizando la herramienta Microsoft Forms para evaluar el NPS, mismo que se encuentra disponible en el enlace del Anexo IX.

4. Cumplimiento de heurísticas de acuerdo con la percepción de los usuarios.

Finalmente, se realiza un cuestionario para evaluar el nivel de cumplimiento de las heurísticas definidas, tomando en cuenta una escala del 1 al 5, donde 1 es “Muy en desacuerdo” y 5 es “Muy de acuerdo”, las preguntas del cuestionario se pueden observar en la tabla que se encuentra disponible en el enlace del Anexo X.

Una vez que los usuarios completaron el cuestionario, se recopiló las respuestas de cada heurística y se realizó un promedio para evaluar su nivel de cumplimiento en base a la Tabla 8.

Tabla 8. Rango de cumplimiento de heurísticas.

Promedio	Rango de cumplimiento	Descripción
1.0 – 1.9	Muy bajo	El sistema falla significativamente en cumplir con la heurística
2.0 – 2.9	Bajo	El sistema tiene deficiencias importantes en esta heurística.
3.0 – 3.9	Moderado	El sistema cumple con la heurística, pero hay áreas de mejora.
4.0 – 4.9	Alto	El sistema cumple bien con la heurística, existe poco margen de mejora.
5.0	Muy alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.

Finalmente, para evaluar el cumplimiento de heurísticas siguiendo el mismo método realizado para las encuestas anteriores, se creó un formulario utilizando la herramienta Microsoft Forms cuyo enlace se encuentra disponible en el Anexo IX.

5. Evaluación de usabilidad usando SortSite.

Para utilizar SortSite se configuraron métricas específicas basadas en guías de usabilidad reconocidas por SortSite. Este análisis incluyó las Readability Guidelines para garantizar la comprensión del contenido, las Usability.gov: Research-Based Web Design & Usability Guidelines para ofrecer una experiencia de usuario óptima, y las W3C Usability Best Practices para asegurar el cumplimiento de los estándares internacionales de usabilidad. La configuración de estas métricas se muestra en la Figura 21.

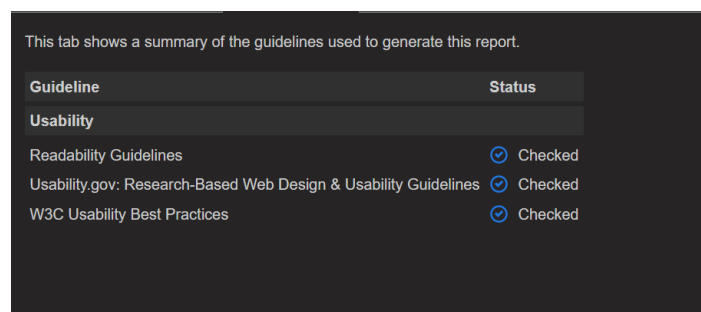


Figura 21. Métricas de usabilidad configuradas en SortSite.

2.13 Evaluación de seguridad

Esta sección abarca un análisis exhaustivo de los riesgos y vulnerabilidades asociados con el Sistema Cloud ERP, desde la identificación y evaluación de activos y amenazas, hasta la implementación de controles y medidas de mitigación. Además, se detalla el diseño de una arquitectura segura que incluye prácticas de autenticación, autorización y encriptación para asegurar cada capa del sistema. El objetivo es proporcionar un enfoque integral para proteger todos los componentes del sistema, garantizando la continuidad del negocio y la seguridad de la información crítica.

Ciclo de vida de desarrollo de software seguro (SSDLC)

Fase 1. Planificación

Evaluación de riesgos

Como primer paso dentro del ciclo de vida de desarrollo de software seguro, se realizó una evaluación de riesgos para identificar las amenazas potenciales y sus impactos.

Identificación de activos

Como primer paso de la evaluación de riesgo, se identificaron los activos que forman parte e interactúan con el Sistema Cloud ERP. En este paso se logra definir con claridad los componentes críticos del sistema en los que se necesita implementar diferentes medidas de protección en base a sus posibles amenazas o vulnerabilidades, para esto se utiliza como base el diagrama de arquitectura.

Front-end

Navegador (Browser): Punto de acceso de los usuarios finales al sistema ERP.

Servicios de aplicaciones web front-end (Front-end web app services): Aloja la interfaz de usuario, responsable de la experiencia del usuario.

Firebase: Gestión de identidades, asegura la autenticación de usuarios.

Back-end

API management service: Gestiona y asegura las APIs que conectan Front-end y back-end.

Servicios de aplicaciones web back-end (Back-end web app services): Contiene la lógica de negocio, crucial para el procesamiento de datos y operaciones.

Base de datos

Servidor SQL (SQL server): Aloja la base de datos principal, almacenamiento de datos críticos.

Base de datos SQL (SQL database): Contiene datos operativos, esencial para las operaciones diarias.

Backup SQL database: Almacenamiento de copias de seguridad, asegura la recuperación de datos en caso de fallo.

Identificación de vulnerabilidades

Como segundo paso de la evaluación de riesgos, se definen las vulnerabilidades que pueden ser explotadas, estas vulnerabilidades también están relacionadas a cada uno de los activos del sistema.

Navegador (Browser)

- **Vulnerabilidades**

- Falta de educación o conciencia sobre phishing entre los usuarios.
- Falta de controles de seguridad adecuados para detectar y bloquear ataques de phishing.
- No tener filtros o mecanismos para prevenir la ejecución de scripts maliciosos (XSS).

Servicios de aplicaciones web front-end

- **Vulnerabilidades**

- Falta de validación adecuada de entradas que podría permitir inyección de código.
- Ausencia de medidas de seguridad contra ataques de fuerza bruta, como bloqueo temporal de cuentas.

Firebase

- **Vulnerabilidades**

- Débiles políticas de gestión de contraseñas, como contraseñas débiles o reutilización de contraseñas.

- Falta de autenticación multifactor (MFA) que podría reducir el riesgo de suplantación de identidad.

API management service

- **Vulnerabilidades**

- Falta de mitigación adecuada contra ataques DDoS, como redes de entrega de contenido (CDN) o servicios de mitigación de DDoS.
- No tener controles de seguridad suficientes para prevenir la inyección SQL a través de APIs.

Servicios de aplicaciones web back-end

- **Vulnerabilidades**

- Implementación incorrecta de mecanismos de autenticación, como sesiones no expiradas o tokens de sesión débiles.
- Errores de configuración que podrían permitir accesos no autorizados a recursos sensibles.

Servidor SQL (SQL Server)

- **Vulnerabilidades**

- Falta de validación de entradas que podría permitir inyección SQL.
- Políticas de seguridad inadecuadas, como permisos excesivos o cuentas de usuario mal configuradas.

Base de datos SQL (SQL Database)

- **Vulnerabilidades**

- Vulnerabilidades conocidas en el sistema de gestión de base de datos que podrían permitir corrupción de datos.
- No tener cifrado adecuado para proteger los datos sensibles almacenados en la base de datos.

Backup SQL Database

- **Vulnerabilidades**

- Falta de cifrado en las copias de seguridad, lo que podría permitir el acceso no autorizado a datos críticos.
- Gestión deficiente de las políticas de retención y acceso a las copias de seguridad.

Matriz de riesgo

Como último paso de la evaluación de riesgo se obtuvo la matriz de riesgo. Herramienta utilizada para evaluar y priorizar los riesgos potenciales que enfrenta el componente del sistema cloud ERP.

Se basa en la evaluación de dos factores clave:

- **Probabilidad:** La probabilidad de que ocurra una amenaza específica. Se evalúa en una escala del 1 al 5, donde 1 representa una baja probabilidad y 5 una alta probabilidad.
- **Impacto:** El impacto que tendría en el sistema si la amenaza se materializara. También se evalúa en una escala del 1 al 5, donde 1 representa un impacto bajo y 5 un impacto alto.

Para realizar la interpretación de la matriz de riesgo, se han definido tres intervalos basados en el resultado de la multiplicación de la probabilidad de la amenaza y el impacto que tendría en el sistema, cuyos resultados se observan en la Tabla 9:

- Riesgo Alto: valores de riesgo entre 16 y 25.
- Riesgo Medio: valores de riesgo entre 9 y 15.
- Riesgo Bajo: valores de riesgo entre 1 y 8.

Tabla 9. Análisis de riesgo.

Amenaza Potencial	Probabilidad (1-5)	Impacto (1-5)	Riesgo = Probabilidad × Impacto
Firestore - Robo de credenciales	4	5	20
Navegador (Browser) - Phishing	4	4	16
API Management Service - Ataques DDoS	3	5	15
Navegador (Browser) - XSS	3	4	12
Servicios Web Front-end - Inyección de código	3	4	12
App Registration - Fuga de claves de API	3	4	12
API Management Service - Inyección SQL a través de APIs	3	4	12
Servicios Web Back-end - Falla de autenticación	3	4	12
Servidor SQL - Inyección SQL	3	4	12
Servidor SQL - Acceso no autorizado	3	4	12
Firestore - Ataques de identidad	3	3	9
App Registration - Configuración incorrecta	2	4	8
Base de Datos SQL - Corrupción de datos	2	4	8
Base de Datos SQL - Fugas de datos	2	4	8
Backup SQL Database - Pérdida de datos	2	4	8
Backup SQL Database - Robo de copias de seguridad	2	4	8
Servicios Web Front-end - Ataques de fuerza bruta	2	3	6
Servicios Web Back-end - Errores de configuración	2	3	6

Fase 2. Diseño

Arquitectura segura

En el diseño de la arquitectura del sistema cloud ERP, la adopción de una arquitectura en tres capas no solo facilita la modularidad y escalabilidad, sino que también permite implementar medidas de seguridad específicas en cada una de las capas, asegurando una protección integral del sistema.

A continuación, se detalla todos los puntos de seguridad abarcados en la arquitectura del sistema.

Puntos de seguridad en la arquitectura

Front-end

- **Firestore: Autenticación y autorización.** Gestiona la identidad y acceso de los usuarios, asegurando que solo los usuarios autenticados y autorizados puedan acceder a las aplicaciones.
- **OAuth 2.0: Tokens de acceso.** Utiliza tokens de acceso con una vida útil limitada para minimizar el riesgo de uso indebido.
- **HTTPS: Encriptación de datos en tránsito.** Asegura que toda la comunicación entre el navegador del usuario y los servicios front-end esté encriptada, protegiendo los datos contra interceptaciones.

Back-end

- **API Gateway (Azure API Management): Autenticación y autorización centralizada.** Gestiona las políticas de seguridad para todas las API, asegurando un control de acceso coherente y centralizado.
- **Rate limiting y throttling:** Previene ataques de denegación de servicio (DoS) al limitar el número de solicitudes que un cliente puede hacer en un período de tiempo determinado.
- **Microsoft Defender for Cloud: Detección de amenazas.** Monitoriza continuamente el entorno en la nube para identificar y responder a posibles amenazas.
- **Application Insights: Monitoreo y diagnóstico.** Permite detectar anomalías en el rendimiento y posibles intentos de intrusión a través del análisis de patrones de tráfico y comportamiento de las aplicaciones.

Base de datos

- **SQL Server y base de datos SQL: Encriptación a nivel de columna.** Permite encriptar datos sensibles a nivel de columna, proporcionando una capa adicional de seguridad.
- **Backup SQL Database: Cifrado de copias de seguridad.** Respaldo de la información en caso de anomalías con la base de datos principal.
- **Azure Monitor: Monitoreo de seguridad.** Proporciona visibilidad continua sobre la integridad y el rendimiento de la infraestructura de la base de datos, permitiendo la detección temprana de actividades sospechosas y vulnerabilidades.

Diseño de control de acceso

El control de acceso en el Sistema Cloud ERP se gestiona mediante Firebase, que proporciona una autenticación robusta y una administración centralizada de permisos. Utilizando el dominio de la organización, se autentica a los usuarios y se les asignan roles específicos que determinan su nivel de acceso y las acciones que pueden realizar dentro del sistema.

Uso de Firebase

Autorización: A través de roles de aplicación, se define y controla el acceso a diferentes funcionalidades del ERP.

Roles implementados

Con base en los roles establecidos en la sección previa, se tienen los registros en Firebase que se detallan en la Figura 22.

	ROLE_ID	ROLE_NAME
1	1	Administrador
2	2	Presidente
3	3	Vicepresidente General
4	4	Director Financiero
5	5	Vicepresidente Financiero
6	6	Vicepresidente de Cultura
7	7	Secretario

Figura 22. Roles implementados.

Estos roles se usan en las políticas de acceso a cada una de las interfaces, negando o aprobando la autorización a roles de acuerdo con las reglas de negocio, como se ilustra en la Figura 23.

```
// Configure Authorization
builder.Services.AddAuthorization(options =>
{
    options.AddPolicy("OrganizationalOnly", policy => policy.RequireRole("Presidente", "Vicepresidente General"));
    options.AddPolicy("AdminOnly", policy => policy.RequireRole("Administrador"));
});
```

Figura 23. Políticas de acceso basado en roles.

Revisión de configuración de seguridad

Previo al despliegue del componente a los servicios en la nube se realizó la revisión de configuración de seguridad, con el enfoca de asegurar que el entorno de producción esté adecuadamente protegido contra amenazas y accesos no autorizados. A continuación, se detallan las principales medidas implementadas:

- **Configuración de seguridad en la nube:** Revisión y aplicación de políticas de seguridad recomendadas por el proveedor de los servicios en la nube, en este caso Azure.
- **Accesos y permisos:** Verificación que solo el personal autorizado tenga acceso al entorno de producción mediante roles y permisos específicos.
- **Herramientas de monitoreo:** Implementación herramientas como Azure Monitor para supervisar la actividad y el rendimiento del sistema.
- **Alertas de seguridad:** Configuración de alertas para eventos sospechosos o intentos de acceso no autorizados.

Codificación segura

Se aplicaron prácticas y guías de seguridad durante todo el desarrollo del software. Esto incluye la validación estricta de todas las entradas de usuario, la gestión adecuada de errores y la aplicación del principio de privilegios mínimos. Siguiendo las recomendaciones de OWASP y realizando revisiones de código periódicas, se minimizan los riesgos de seguridad en el código fuente.

- **Validación de entradas:** Implementar validaciones estrictas para todas las entradas de usuario para prevenir inyecciones SQL, XSS y otros tipos de ataques.
- **Principio de privilegios mínimos:** Escribir código que opere con los menores privilegios necesarios para realizar su función.

Cifrado de datos

Para proteger la confidencialidad e integridad de la información tanto en tránsito como en reposo. Se aseguró que todas las comunicaciones entre el cliente y el servidor estén cifradas mediante HTTPS y TLS. Para la información en reposo, se cifraron datos que son sensibles para la organización utilizando técnica de cifrado AES.

Mantenimiento

Para garantizar la seguridad del sistema a largo plazo y evitar complicaciones en las operaciones diarias, se debe monitorear continuamente varios aspectos configurados previamente utilizando servicios como Azure Monitor, Application Insights y Microsoft Defender for Cloud.

A continuación, se detallan los aspectos clave que se deben monitorear.

Azure monitor

Métricas de rendimiento:

- **CPU, memoria y uso de disco:** Supervisión del uso de recursos y problemas de rendimiento.
- **Latencia y tiempos de respuesta:** Medición de tiempos de respuesta de las aplicaciones.

Alertas y notificaciones:

- **Umbral de recursos:** Configuración de alertas para cuando los recursos excedan ciertos umbrales.
- **Errores y excepciones:** Configuración de alertas para detectar y notificar sobre errores y excepciones en el sistema.

Logs y auditorías:

- **Registros de actividad:** Revisión de logs de actividad para rastrear acciones de usuarios y cambios en la configuración.
- **Eventos de seguridad:** Monitoreo de eventos de seguridad para detectar posibles amenazas o accesos no autorizados.

Microsoft Defender for Cloud

Evaluaciones de seguridad:

- **Análisis de vulnerabilidades:** Implementación de análisis periódicos para identificar y mitigar vulnerabilidades en la infraestructura y aplicaciones.
- **Recomendaciones de seguridad:** Seguimiento de las recomendaciones de Microsoft Defender para mejorar la postura de seguridad del sistema.

Monitoreo de amenazas:

- **Alertas de seguridad:** Configuración de alertas para ser notificado inmediatamente sobre incidentes de seguridad.

Cumplimiento y gobernanza:

- **Políticas de seguridad:** Monitorear el cumplimiento de políticas de seguridad y regulaciones relevantes.

Fase 3. Pruebas de seguridad

Análisis Estático de Código (SAST)

Objetivo de la prueba

El objetivo del Análisis Estático de Código (SAST) es identificar vulnerabilidades de seguridad en el código fuente antes de que la aplicación se ejecute. Este análisis ayuda a detectar errores de programación que pueden ser explotados por atacantes.

Metodología

- **Herramientas utilizadas:** Herramientas específicas como SonarCloud..
- **Ámbito del análisis:** Inclusión de todos los módulos y componentes del sistema Cloud ERP.
- **Ejecución del análisis:** Ejecución del análisis, incluyendo la configuración de la herramienta y la integración con el sistema de control de versiones GitHub.

Aspectos evaluados

- **Vulnerabilidades de seguridad:** Identificación de vulnerabilidades como inyecciones SQL, vulnerabilidades XSS (Cross-Site Scripting), CSRF (Cross-Site Request Forgery).

- **Errores de codificación:** Detección de errores comunes de programación que pueden llevar a problemas de seguridad.
- **Buenas prácticas de codificación:** Verificación del cumplimiento de las mejores prácticas y estándares de codificación.

Para la evaluación SAST se utiliza la herramienta SonarCloud, cuya configuración se observa en la Figura 24, y permite el análisis de diversas métricas para evaluar la calidad del código, incluyendo bugs, vulnerabilidades de seguridad, duplicaciones de código, code smells, cobertura de pruebas unitarias, complejidad ciclomática, documentación y mantenibilidad.

```

- name: Cache SonarCloud packages
  uses: actions/cache@v3
  with:
    path: ~\sonar\cache
    key: ${{ runner.os }}-sonar
    restore-keys: ${{ runner.os }}-sonar

- name: Cache SonarCloud scanner
  id: cache-sonar-scanner
  uses: actions/cache@v3
  with:
    path: .\sonar\scanner
    key: ${{ runner.os }}-sonar-scanner
    restore-keys: ${{ runner.os }}-sonar-scanner

- name: Install SonarCloud scanner
  if: steps.cache-sonar-scanner.outputs.cache-hit != 'true'
  shell: powershell
  run: |
    New-Item -Path .\sonar\scanner -ItemType Directory
    dotnet tool update dotnet-sonarscanner --tool-path .\sonar\scanner

- name: Build and analyze
  env:
    GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }} # Needed to get PR information, if any
    SONAR_TOKEN: ${{ secrets.SONAR_TOKEN }}
  shell: powershell
  run: |
    .\sonar\scanner\dotnet-sonarscanner begin /k:"JavierRevelo_CloudERP" /o:"javierrevelo" /d:sonar.token="${{ secrets.SONAR_TOKEN }}"
    dotnet build
    .\sonar\scanner\dotnet-sonarscanner end /d:sonar.token="${{ secrets.SONAR_TOKEN }}"

```

Figura 24. Configuración SonarCloud.

3 RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

3.1 Resultados de entrevistas Think Aloud

Tras examinar las transcripciones y extraer términos emocionales recurrentes, se identificaron patrones significativos en las respuestas de los usuarios. Este análisis revela no solo los aspectos que generaron mayor satisfacción y facilidad de uso, sino también aquellos que provocaron frustración y confusión. En la Tabla 10, se puede observar las palabras más recurrentes en cada una de las entrevistas realizadas a usuarios.

La contabilización de estas palabras proporciona una visión cuantitativa de las emociones y percepciones de los usuarios durante la interacción con el sistema. Por ejemplo, palabras positivas como "Bien" y "Cómodo/a" indican una experiencia satisfactoria y agradable, mientras que términos negativos como "Perdido/a" y "Confuso" sugieren áreas

problemáticas que requieren atención. Las palabras neutras como "Normal" y "Cambiar" reflejan aspectos de la interfaz que son aceptables, pero podrían mejorarse.

Al comprender las emociones y reacciones de los usuarios, se pueden hacer ajustes específicos para mejorar la experiencia del usuario y aumentar la satisfacción general con el sistema.

Tabla 10. Contabilización de palabras recurrentes.

Emoción	Palabra	Promedio	Entv1	Entv2	Entv3	Entv4	Entv5	Entv6
Positiva	Bien	10	12	10	8	9	11	10
	Claro	3	3	2	4	3	3	4
	Cómodo/a	4	4	5	3	4	4	5
	Fácil	4	6	4	2	6	4	4
	Entiendo	2	2	3	1	2	2	2
Negativa	Perdido/a	3	5	3	4	1	2	3
	Confuso	2	2	1	3	2	2	3
	No entiendo	4	6	4	2	4	4	6
	Difícil	2	3	2	2	2	1	2
	No sé	3	2	5	4	1	3	3
Neutro	Más o menos	3	4	5	3	1	2	3
	Normal	2	3	2	2	2	1	2
	Cambiar	4	6	4	2	4	4	4
	Mejorar	2	2	1	3	2	2	2

La Figura 25 muestra la distribución de términos recurrentes extraídos de las entrevistas Think Aloud. En este gráfico, se visualizan las palabras más mencionadas por los usuarios durante las entrevistas, categorizadas en emociones positivas, negativas y neutras. Este análisis gráfico complementa la Tabla anterior y proporciona una representación visual clara de las emociones y percepciones de los usuarios.

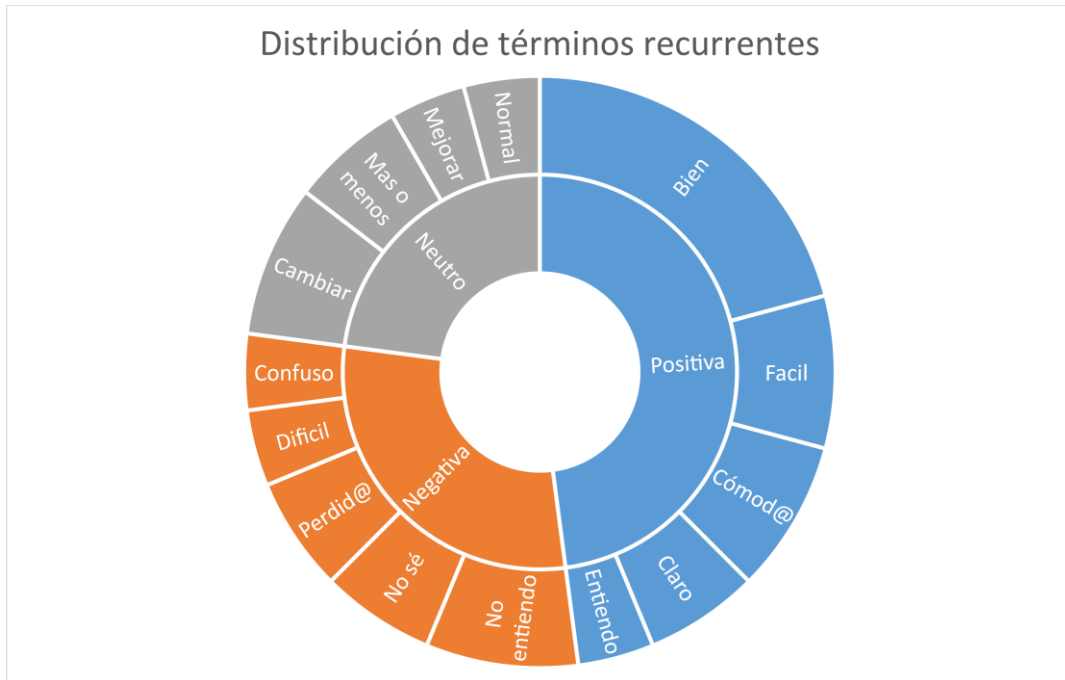


Figura 25. Distribución términos recurrentes.

3.2 Resultados encuesta SUS

La encuesta System Usability Scale (SUS) midió la usabilidad percibida del sistema Cloud ERP por los usuarios de FEPON a través de 10 enunciados, cuyos resultados se muestran en la Tabla 11.

Puntaje promedio por pregunta

■ Enunciados pares ■ Enunciados impares

Tabla 11. Promedio de puntaje por enunciado encuesta SUS.

Enunciados	Puntaje
Creo que me gustaría usar este sistema con frecuencia.	4.4
Encontré el sistema innecesariamente complejo.	1.6
Creo que el sistema es fácil de usar.	4.6
Creo que necesitaría la ayuda de una persona con conocimientos técnicos para poder usar este sistema.	1.6
Encontré que las diversas funciones del sistema estaban bien integradas.	4.2
Pensé que había demasiada inconsistencia en este sistema.	1.6
Imagino que la mayoría de las personas aprenderían a usar este sistema muy rápidamente.	4.6
Encontré el sistema muy engorroso de usar.	2.2
Me sentí muy seguro/a usando el sistema.	4.4
Necesité aprender muchas cosas antes de poder desenvolverme con el sistema.	2

La Figura 26 muestra el puntaje promedio obtenido por cada enunciado de la encuesta SUS, las preguntas impares se muestran en color anaranjado y las pares en color azul. Es importante notar que el resultado de las preguntas pares cuanto más alto sea el resultado es mejor, mientras que en las impares mientras más bajo es mejor. Los resultados revelan que los usuarios se sienten mayormente seguros usando el sistema y consideran que es fácil de usar, sin embargo, el enunciado “Necesité aprender muchas cosas antes de poder usar el sistema” sugiere que la curva de aprendizaje podría ser un obstáculo para algunos usuarios. En general, aunque el sistema es percibido como seguro y fácil de usar, hay oportunidades para mejorar la consistencia y reducir la complejidad percibida.

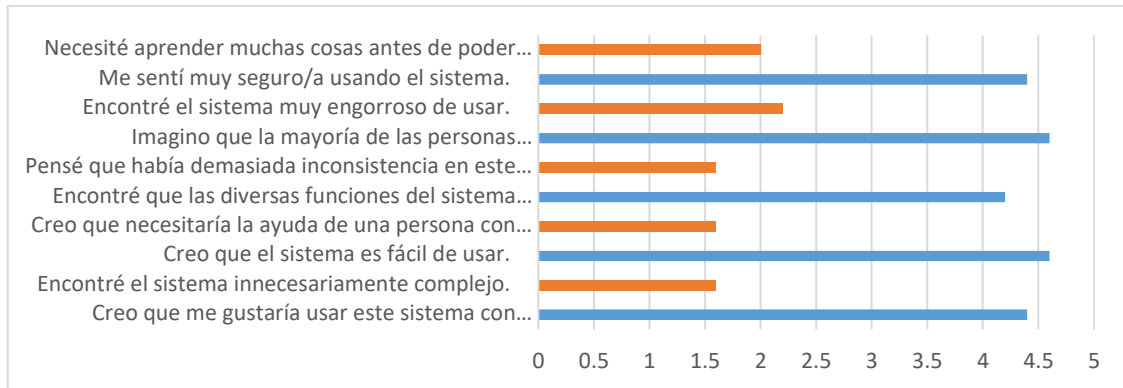


Figura 26. Puntaje promedio por enunciado encuesta SUS.

Puntaje SUS

$$SUS = [(Suma enunciados impares - 5) + (25 - Suma enunciados pares)] * 2.5$$

$$SUS = [(22.2 - 5) + (25 - 9)] * 2.5$$

$$SUS = 83$$

Rating SUS

Finalmente, en base al puntaje SUS obtenido, el módulo de inventario del sistema cloud ERP se encuentra en la categoría Excelente a nivel de usabilidad de acuerdo con la Tabla 12.

Tabla 12. Resultado rating SUS.

SUS Score	Rating
> 80.3	Excelente
68 – 80.2	Bueno
67	Aceptable
51 – 66	Pobre
< 51	Terrible

3.3 Resultados encuesta NPS

El Net Promoter Score (NPS) evaluó la lealtad y satisfacción de los usuarios con el sistema mediante la pregunta "En una escala del 0 al 10, ¿cuán probable es que recomiende este producto a un amigo o colega?".

En base a la respuesta del usuario, el índice NPS se clasifica según se observa en la Tabla 13.

Tabla 13. Resultado NPS.

Clasificación	Descripción	Puntuación	Cantidad
Promotor	Usuarios satisfechos y leales que probablemente recomendarán tu producto o servicio.	9-10	4
Pasivo	Usuarios satisfechos pero que no recomendarían el producto.	7-8	1
Detractor	Clientes insatisfechos que podrían afectar el producto a través de comentarios negativos.	0-6	0

En la Figura 27 se muestra un gráfico circular que representa la distribución de las opiniones de los usuarios: el 80% son promotores, el 20% son pasivos, y no hay detractores.

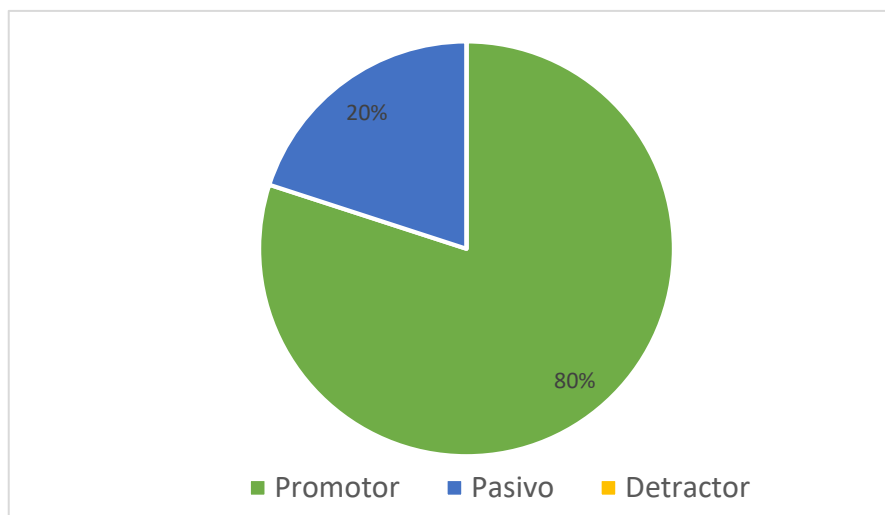


Figura 27. Gráfica de clasificación NPS.

Posteriormente se calcula el NPS utilizando la siguiente fórmula:

$$\text{NPS} = \% \text{promotores} - \% \text{detractores}$$

$$\text{NPS} = \left(\frac{4}{5} * 100\right) \% - \left(\frac{0}{5} * 100\right) \% = 80\% - 0\%$$

$$\text{NPS} = 80\%$$

Finalmente, en base al NPS obtenido, el módulo de inventario del sistema se encuentra en la clasificación Excelente como se observa en la Tabla 14.

Tabla 14 . Resultados NPS.

Clasificación	Descripción	Puntuación
Grave	El sistema tiene más detractores que promotores. Necesita mejoras urgentes para aumentar la satisfacción del usuario.	< 0
Insatisfactorio	Predominan los detractores sobre los promotores. Es necesario abordar problemas importantes para mejorar la experiencia del usuario.	0 – 27
Moderadamente positivo	Los usuarios están generalmente satisfechos, aunque hay oportunidades de mejora	28 – 54
Excelente	La mayoría de los usuarios son promotores y están muy satisfechos con el sistema. Existen muy pocas áreas de mejora.	55 - 100

3.4 Resultado encuesta de cumplimiento de heurísticas

La encuesta de cumplimiento de heurísticas evaluó el nivel en que el sistema cumple con las heurísticas de usabilidad definidas. Los usuarios calificaron en una escala del 1 al 5, donde 1 es "muy en desacuerdo" y 5 es "muy de acuerdo".

Los resultados por heurística se observan en la Figura 28.

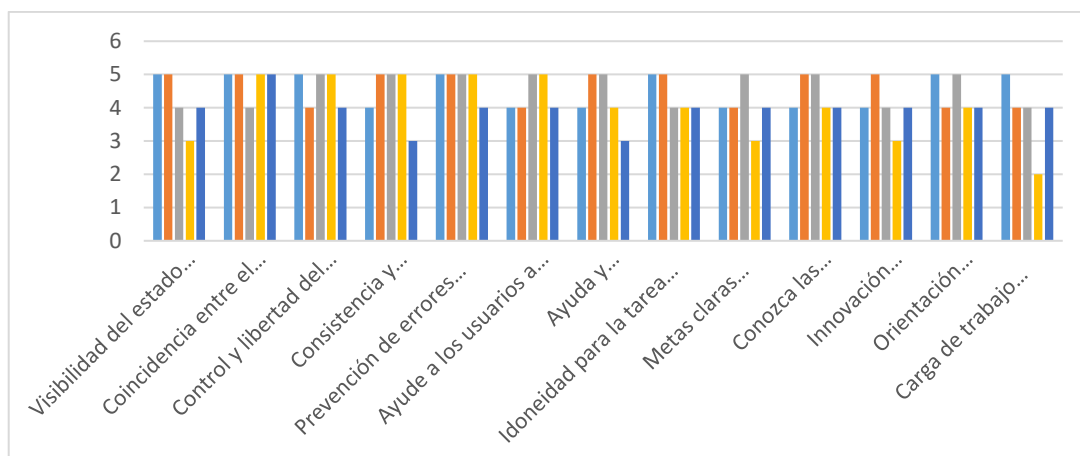


Figura 28. Puntaje promedio encuesta de cumplimiento de heurísticas.

Los resultados se muestran en la Tabla 15, donde se observa que el sistema en general cumple con la mayoría de las heurísticas con un rango de cumplimiento alto. Sin embargo, hay una oportunidad de mejora notable en el área de "Carga de trabajo", que obtuvo un puntaje promedio de 3.8 y un rango de cumplimiento moderado. Para mejorar esta heurística, se podría reducir la carga cognitiva del usuario simplificando tareas complejas, mejorando la automatización de procesos repetitivos y optimizando la interfaz para que sea más intuitiva y fácil de usar.

Tabla 15. Puntaje promedio y rango de cumplimiento de heurísticas.

Heurística	Promedio	Rango de Cumplimiento	Descripción
Visibilidad del estado del sistema	4.2	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Coincidencia entre el sistema y el mundo real	4.8	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Control y libertad del usuario	4.6	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Consistencia y estándares	4.4	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Prevención de errores	4.8	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.

Ayude a los usuarios a reconocer, diagnosticar y recuperarse de errores	4.4	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Ayuda y documentación	4.2	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Idoneidad para la tarea	4.4	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Metas claras	4	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Conozca las motivaciones de su usuario	4.4	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Innovación conservadora	4	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Orientación	4.4	Alto	El sistema cumple perfectamente con la heurística. No se requieren mejoras.
Carga de trabajo	3.8	Moderado	El sistema cumple bien con la heurística, existe poco margen de mejora.

3.5 Resultados evaluación con SortSite

El análisis de los resultados proporcionados por la herramienta SortSite, tal como se muestra en la Figura 29, indica que no se encontraron problemas de usabilidad en las páginas evaluadas de las interfaces del módulo de movimientos de inventario del sistema ERP. La evaluación abarcó un total de cuatro páginas y archivos, y los resultados demuestran que el 0% de las páginas presenta problemas, situándose por encima del promedio en comparación con otros benchmarks. Esto sugiere que la interfaz del módulo de movimientos de inventario cumple con los estándares de usabilidad establecidos,

proporcionando una experiencia de usuario óptima. El análisis de otras interfaces arrojó el mismo resultado, estos se pueden observar en el Anexo XI.

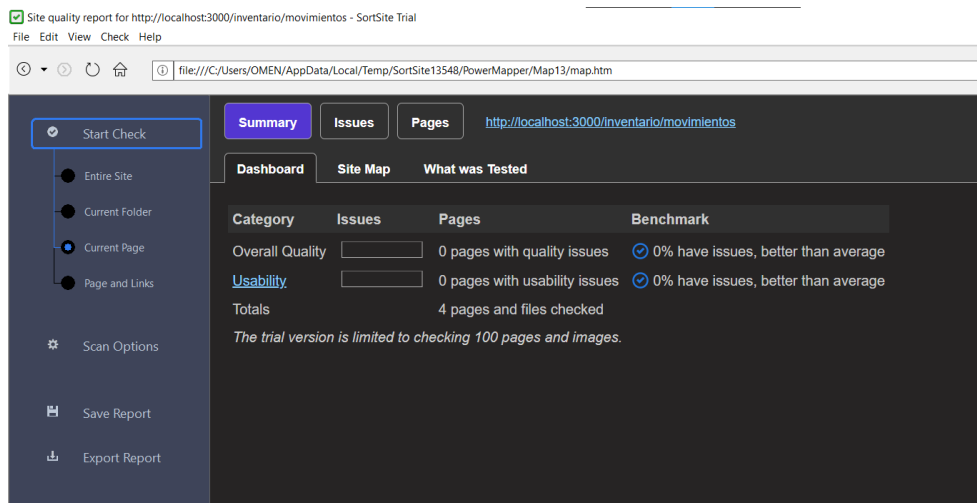


Figura 29. Resultado del análisis con SortSite para la interfaz de inventario.

3.6 Resultados de métricas servicios cloud

Servidor SQL

El análisis de las métricas de uso y rendimiento del servidor de la base de datos del sistema ERP Fepon revela un funcionamiento eficiente y estable. El uso promedio de la CPU es bajo, con un 0.18%, lo que indica que el servidor tiene suficiente capacidad de procesamiento disponible. El espacio máximo de datos utilizado se mantiene constante en 32.5 MB, lo que sugiere estabilidad en el almacenamiento de datos. Además, el servidor registra 128 conexiones exitosas, lo que refleja un acceso constante y sin problemas, indicando la fiabilidad y accesibilidad del sistema. En general, el servidor de la base de datos muestra un buen rendimiento, como se observa en la Figura 30.

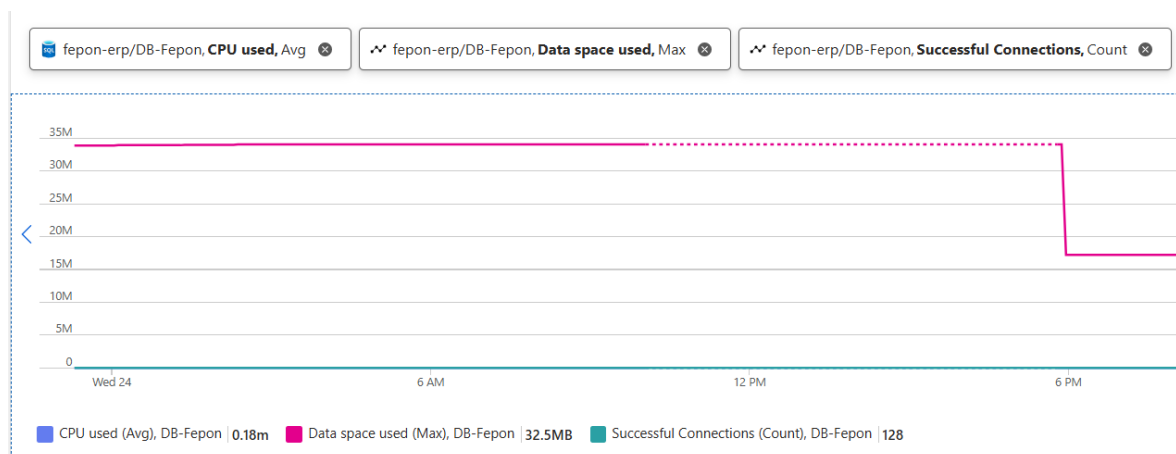


Figura 30. Métricas de servidor SQL.

App Management (APIs)

En la Figura 31, se muestra el análisis de las métricas de uso y rendimiento del servidor "backendfepon", un componente esencial del sistema Cloud ERP. Las métricas incluyen el número de solicitudes recibidas (Requests Count), el tiempo promedio de respuesta del servidor (Response Time Avg) y el uso promedio de memoria (Memory Working Set Avg). Durante las pruebas el servidor mantuvo un nivel constante de solicitudes, con un total de 33. El tiempo promedio de respuesta fue de 3.36 segundos, mientras que el uso promedio de memoria alcanzó los 117.6 MB. Este análisis permite identificar los momentos de mayor demanda y evaluar la eficiencia del servidor en términos de tiempo de respuesta y gestión de recursos, proporcionando una base para optimizar el rendimiento y la escalabilidad del sistema.

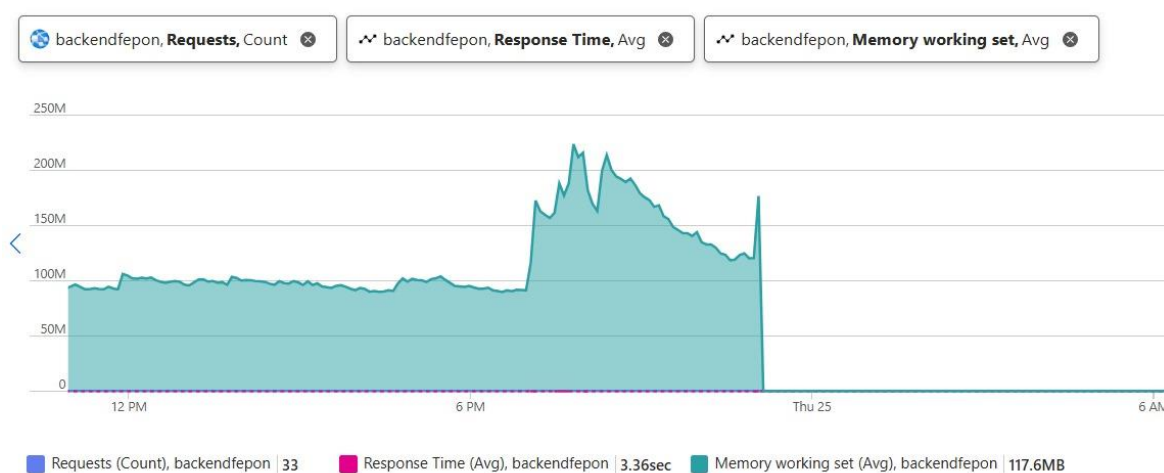


Figura 31. Métricas de API management.

3.7 Resultado de Microsoft Defender for Cloud

En la Figura 32, se muestra un análisis de la postura de seguridad proporcionada por Microsoft Defender for Cloud. Los resultados indican que la suscripción utilizada para la implementación de los servicios del sistema Cloud ERP, con un total de 11 recursos evaluados, no presenta recomendaciones críticas, caminos de ataque ni alertas de seguridad. En términos de riesgo ambiental y puntaje de seguridad, se identificaron 21 recomendaciones en total, categorizadas en niveles de riesgo: 0 críticas, 1 alta, 5 medias y 15 bajas. El puntaje total de seguridad es del 75%, lo cual refleja un buen nivel de seguridad general, aunque aún hay margen para mejoras. Este puntaje sugiere que, aunque no se han identificado amenazas críticas, existen oportunidades para fortalecer la postura de seguridad mediante la evaluación y aplicación de las recomendaciones dadas por Microsoft.

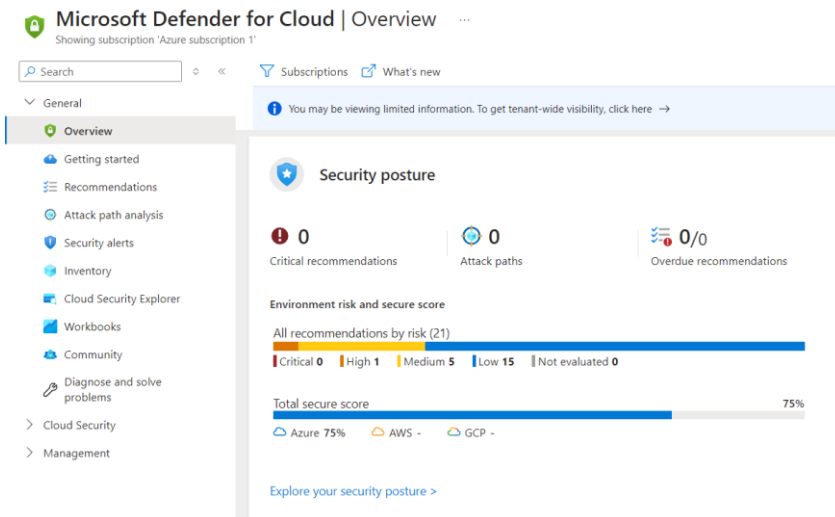


Figura 32. Resumen general de Microsoft Defender for Cloud.

En la Figura 33, se muestra que las recomendaciones de seguridad incluyen la resolución de vulnerabilidades en bases de datos SQL, la asignación de un administrador de Azure Active Directory en servidores SQL, y la desactivación del acceso a redes públicas en bases de datos SQL, entre otras. Estas recomendaciones se relacionan principalmente con la exposición a internet, indicando un riesgo potencial debido a la accesibilidad de estos recursos desde redes públicas. La implementación de estas recomendaciones mejoraría significativamente la postura de seguridad del sistema ERP en la nube.

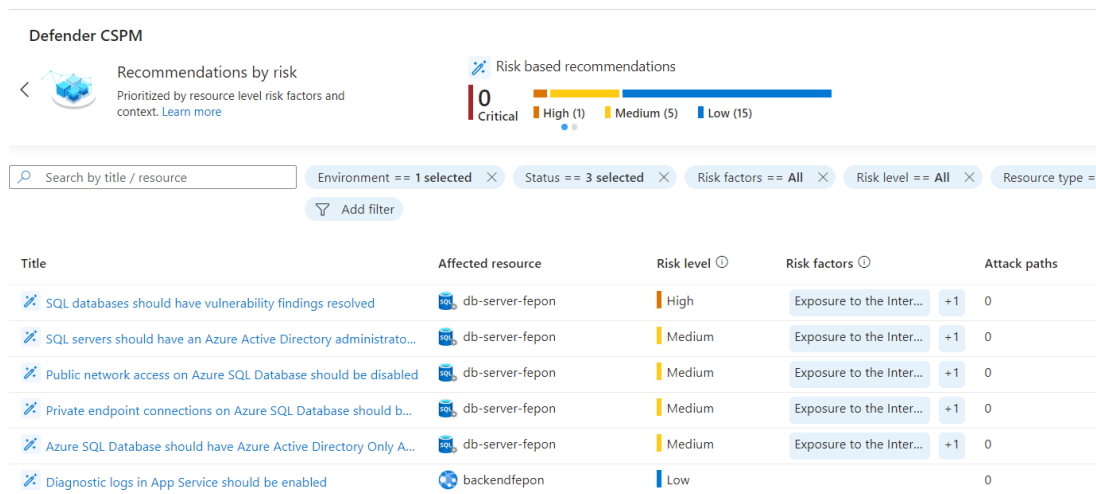


Figura 33. Resumen recomendaciones de Microsoft Defender for Cloud.

3.8 Resultados SAST

El análisis de código estático muestra resultados positivos en términos de seguridad general. El apartado de seguridad ha sido aprobado, con cero problemas abiertos en seguridad y fiabilidad, indicando un código seguro y robusto. Sin embargo, hay 324 problemas abiertos de severidad media en mantenibilidad que necesitan atención para mejorar la calidad a largo plazo. Adicional no se detectaron duplicaciones, lo que es positivo para el mantenimiento, como se observa en la Figura 34.

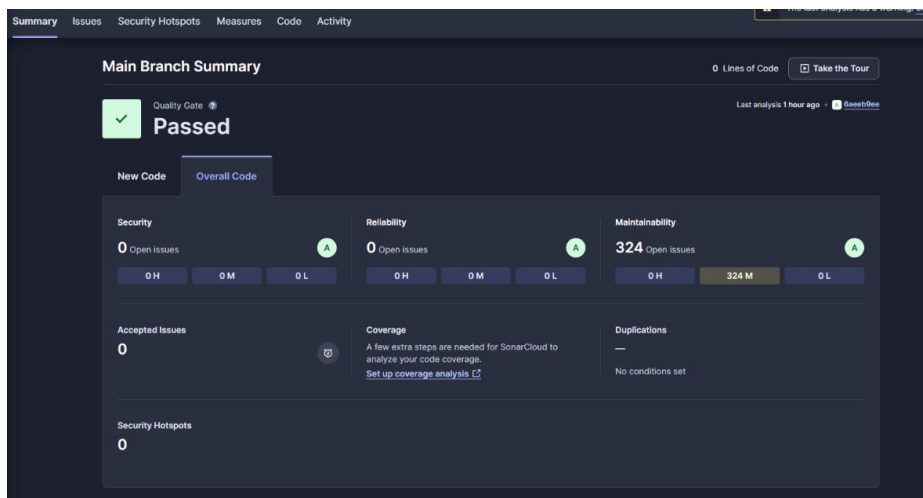


Figura 34. Resultados generales de análisis código estático.

En la Figura 35, se muestra a detalle el resultado del análisis de código estático por carpetas de código, destacando la ausencia de bugs, vulnerabilidades y puntos críticos de seguridad, lo cual es positivo. Se identificaron varios "code smells", especialmente en la carpeta backendfepon, indicando áreas que necesitan mejora. También se observa una falta de cobertura de pruebas en algunos archivos, sugiriendo mejorar la precisión de la configuración de cobertura del análisis.

	Lines of Code	Bugs	Vulnerabilities	Code Smells	Security Hotspots	Coverage	Duplications
<input type="checkbox"/> backendfepon	—	0	0	305	0	—	—
<input type="checkbox"/> Controllers	—	0	0	39	0	—	—
<input type="checkbox"/> Cypher	—	0	0	9	0	—	—
<input type="checkbox"/> Data	—	0	0	0	0	—	—
<input type="checkbox"/> DTOs	—	0	0	133	0	—	—
<input type="checkbox"/> ErrorsModels	—	0	0	4	0	—	—
<input type="checkbox"/> ModelConfigurations	—	0	0	0	0	—	—
<input type="checkbox"/> Models	—	0	0	119	0	—	—
<input type="checkbox"/> Properties	—	0	0	0	0	—	—
<input type="checkbox"/> Secrets	—	0	0	0	0	—	—
<input type="checkbox"/> Utils	—	0	0	0	0	—	—
<input checked="" type="checkbox"/> appsettings.Development.json	—	0	0	0	0	—	0.0%
<input checked="" type="checkbox"/> appsettings.json	—	0	0	0	0	—	0.0%
<input checked="" type="checkbox"/> Program.cs	—	0	0	1	0	—	0.0%

Figura 35. Resultados de análisis código estático.

3.9 Conclusiones

Se concluye que el enfoque de gestión visual del trabajo que ofrece Kanban permitió al equipo adaptarse rápidamente a los cambios en las prioridades de las tareas del módulo de inventario. Durante el desarrollo de este componente, se encontraron varias dificultades que, sin la metodología Kanban, podrían haber causado problemas significativos. Sin embargo, Kanban permitió al equipo ajustarse eficientemente a los cambios en las prioridades de los requisitos del proyecto, asegurando que el módulo de inventario llegara a tiempo y fuera completamente funcional.

Las heurísticas seleccionadas para la evaluación de usabilidad del módulo de inventario del sistema ERP ofrecen una metodología robusta y exhaustiva que aborda múltiples aspectos críticos de la experiencia del usuario. Al integrar estas heurísticas, se logró una evaluación completa que identificó y solucionó problemas de usabilidad, optimizando la eficiencia y efectividad de las tareas de gestión de inventario, y garantizando una experiencia de usuario coherente y satisfactoria.

Se concluye que la evaluación de usabilidad utilizando una herramienta automática no refleja completamente lo que en realidad siente el usuario. Aunque estas herramientas son útiles para identificar problemas técnicos y de diseño en las interfaces, no pueden capturar aspectos más subjetivos y emocionales de la experiencia del usuario. Los resultados de la herramienta pueden indicar que no hay problemas de calidad o usabilidad, pero esto no garantiza que los usuarios finales encuentren la interfaz intuitiva o agradable de usar. Por lo tanto, es fundamental complementar las evaluaciones automáticas con pruebas de usabilidad que involucren a usuarios reales, permitiendo obtener una visión más holística y precisa de la experiencia del usuario.

En base al análisis de la postura de seguridad proporcionada por Microsoft Defender for Cloud, el sistema Cloud ERP presenta un nivel de seguridad general del 75%, con 21 recomendaciones categorizadas en niveles de riesgo: 0 críticas, 1 alta, 5 medias y 15 bajas. Debido a la falta de créditos en Azure, no fue posible implementar las correcciones para estas recomendaciones en este momento, lo cual subraya la necesidad de ajustar el presupuesto para priorizar la seguridad y mejorar la postura del sistema ERP en la nube.

Los resultados del análisis estático de SonarCloud revelan varias oportunidades de mejora en la mantenibilidad del producto para futuras versiones del producto. No se detectaron errores críticos ni vulnerabilidades significativas, lo cual es positivo. Sin embargo, la identificación de malas prácticas en el código sugiere que hay áreas donde se puede optimizar el código para mejorar la calidad y mantenibilidad del sistema. Abordar estos

aspectos permitirá no solo un código más limpio y eficiente, sino también facilitará el mantenimiento y la escalabilidad del sistema en el futuro.

El componente permitió garantizar un seguimiento preciso y eficiente de los productos en el inventario, facilitando la identificación, clasificación y registro adecuado de los artículos, a través de las interfaces de movimientos de inventario, proveedores y productos, se logró agilizar y simplificar los procedimientos de ingreso de productos al inventario, asegurando la correcta actualización de la información en el sistema y proporcionando una vía efectiva para optimizar la gestión y operación del inventario de la federación.

3.10 Recomendaciones

Se recomienda solicitar permisos a las autoridades de la DGIP para establecer un ambiente de desarrollo al que los estudiantes puedan acceder, ya que, el uso del tenant de la Escuela Politécnica Nacional facilitaría la gestión de usuarios y la autenticación en la aplicación, permitiendo que esta sea accesible únicamente mediante el correo institucional. Esto no solo optimizaría la seguridad de la aplicación, sino que también simplificaría el proceso de acceso para los estudiantes.

Se recomienda considerar el límite de créditos disponibles para cada estudiante en el uso de Azure, con el fin de evitar problemas con la base de datos. Además, es importante tener en cuenta que, en caso de agotarse los créditos de las cuentas personales, se pueden utilizar las cuentas de compañeros de facultades que no hagan uso de este recurso, lo cual permite maximizar el uso de las herramientas disponibles.

Se recomienda añadir un módulo que permita visualizar el registro de inventario las asociaciones anteriores, ya que actualmente el sistema solo muestra el registro de la federación actual. Esto permitiría que todas las generaciones de la federación puedan tener conocimiento de las actividades de sus predecesores, fomentando la continuidad y el aprendizaje a lo largo del tiempo.

Se recomienda implementar un sistema de etiquetado eficiente para todos los productos en el inventario. Este sistema debería utilizar códigos de barras o etiquetas RFID para cada artículo, permitiendo una identificación rápida y precisa. Al escanear las etiquetas, se puede actualizar automáticamente el estado del inventario, reducir errores humanos y mejorar la eficiencia en los procesos de entrada, salida y conteo de productos. Además, este sistema facilitaría la trazabilidad de los productos, permitiendo un mejor control y gestión del inventario.

4 REFERENCIAS BIBLIOGRÁFICAS

- [1] P. Balbudhe, M. Meshram y R. Rasekar, "Enterprise Resource Planning (ERP) System for Educational Organization", *Int. J. Comput. Electron. Aspects Eng.*, vol. 3, n.º 1, p. 11, 2022.
- [2] S. Li *et al.*, "Understanding and addressing quality attributes of microservices architecture: A Systematic literature review", *Inf. Softw. Technol.*, vol. 131, p. 106449, marzo de 2021. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://doi.org/10.1016/j.infsof.2020.106449>
- [3] "The starting point for learning TypeScript". TypeScript: JavaScript With Syntax For Types. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://www.typescriptlang.org/docs/>
- [4] "React". React. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://react.dev/>
- [5] "OAuth 2.0 — OAuth". OAuth Community Site. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://oauth.net/2/>
- [6] "C# Guide - .NET managed language". Microsoft Learn: Build skills that open doors in your career. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://learn.microsoft.com/en-us/dotnet/csharp/>
- [7] "ASP.NET documentation". Microsoft Learn: Build skills that open doors in your career. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://learn.microsoft.com/en-us/aspnet/core/?view=aspnetcore-8.0>
- [8] <https://help.figma.com/hc/en-us>
- [9] F. Staiano, *Designing and Prototyping Interfaces with Figma: Learn essential UX/UI design principles by creating interactive prototypes for mobile, tablet, and desktop*. Packt Publ., 2022.
- [10] "Azure DevOps documentation". Microsoft Learn: Build skills that open doors in your career. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://learn.microsoft.com/en-us/azure/devops/?view=azure-devops>
- [11] "Azure Data Studio - Azure Data Studio". Microsoft Learn: Build skills that open doors in your career. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://learn.microsoft.com/en-us/azure-data-studio/>

- [12] Microsoft. "Visual Studio Code - Code Editing. Redefined". Visual Studio Code - Code Editing. Redefined. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://code.visualstudio.com>
- [13] "Azure App Service documentation - Azure App Service". Microsoft Learn: Build skills that open doors in your career. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://learn.microsoft.com/en-us/azure/app-service/>
- [14] "API Management documentation". Microsoft Learn: Build skills that open doors in your career. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://learn.microsoft.com/en-us/azure/api-management/>
- [15] "Microsoft Defender for Cloud documentation - Microsoft Defender for Cloud". Microsoft Learn: Build skills that open doors in your career. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://learn.microsoft.com/en-us/azure/defender-for-cloud/>
- [16] "Azure Monitor documentation - Azure Monitor". Microsoft Learn: Build skills that open doors in your career. Accedido el 26 de julio de 2024. [En línea]. Disponible: <https://learn.microsoft.com/en-us/azure/azure-monitor/>
- [17] R. S. Sandhu, "Role-based Access Control," *Advances in Computers*, vol. 46, pp. 237-286, 1998.
- [18] J. Nielsen, "10 Usability Heuristics for User Interface Design," *Nielsen Norman Group*, 2024. [Online]. Available: <https://www.nngroup.com/articles/ten-usability-heuristics/>. [Accessed: 2024]
- [19] "Colombo & Paschs Heuristics" *Heurio*, 2023. [Online]. Available: <https://www.heurio.co/colombo-and-paschs-heuristics>. [Accessed: 13-Apr-2024]
- [20] "ISO 9241 Part 110 Dialogue Principles," *Heurio*, 2023. [Online]. Available: <https://www.heurio.co/iso-9241-part-110-dialogue-principles>. [Accessed: 13-Apr-2024]
- [21] "Bastien & Scapin Evaluation of Human-Computer Interfaces" *Heurio*, 2023. [Online]. Available: <https://www.heurio.co/bastien-and-scapin-evaluation-of-human-computer-interfaces>. [Accessed: 13-Apr-2024]
- [22] "¿Qué es la norma ISO 27001 y para qué sirve?" GlobalSuite Solutions. Accedido el 13 de abril de 2024 [Online]. Available: <https://www.globalsuitesolutions.com/es/que-es-la-norma-iso-27001-y-para-que-sirve/> [Accessed: 13-Apr-2024].

[23] “Inicio - OWASP Top 10:2021”. OWASP Foundation, the Open Source Foundation for Application Security | OWASP Foundation [Online]. Available: <https://owasp.org/Top10/es/> [Accessed: 13-Apr-2024]

[24] “CWE - CWE Top 25 Most Dangerous Software Weaknesses”. CWE - Common Weakness Enumeration [Online]. Available at: <https://cwe.mitre.org/top25/> [Accessed: 13-Apr-2024].

5 ANEXOS

5.1 Anexo I. Tabla de heurísticas seleccionadas.

La tabla se encuentra [disponible aquí.](#)

5.2 Anexo II. Tabla de controles de seguridad seleccionados.

La tabla se encuentra [disponible aquí.](#)

5.3 Anexo III. Tabla de roles del sistema

La tabla se encuentra [disponible aquí.](#)

5.4 Anexo IV. Perfiles de usuario

La tabla se encuentra [disponible aquí.](#)

5.5 Anexo V. Desarrollo detallado del front-end y back-end

[Desarrollo del frontend: Disponible aquí](#)

[Desarrollo del backend: Disponible aquí](#)

5.6 Anexo VI: Ejemplos de pruebas de integración y unitarias

Ejemplo de prueba unitaria

```
[Fact]
public async Task PostInventoryMovement_ReturnsCreatedInventoryMovement ()
{
    // Arrange
    InitializeDatabase();

    var inventoryMovementDTO = new CreateUpdateInventoryMovementDTO
    {
        product_Name = "Product 1",
        inventory_Movement_Type_Name = "Type 1",
        quantity = 10,
        Date = DateTime.Now
    };

    var product = new Product { Product_Id = 1, Name = "Product 1", Description = "Description
1", Label = "Label 1", Quantity = 100 };
    var inventoryMovementType = new InventoryMovementType { Movement_Type_Id = 1,
Movement_Type_Name = "Type 1" };

    _context.Products.Add(product);
    _context.InventoryMovementTypes.Add(inventoryMovementType);
    _context.SaveChanges();

    try
    {
        // Act
        var result = await _controller.PostInventoryMovement(inventoryMovementDTO);

        // Assert
        var actionResult = Assert.IsType<ActionResult<InventoryMovementDTO>>(result);
        if (!(actionResult.Result is CreatedAtActionResult))
        {
            var objectResult = actionResult.Result as ObjectResult;
            var errorMessage = objectResult?.Value?.ToString() ?? "No error message";
            Assert.True(false, $"Unexpected result type: {actionResult.Result.GetType().Name},
Error: {errorMessage}");
        }
        Assert.IsType<CreatedAtActionResult>(actionResult.Result);

        var createdAtActionResult = (CreatedAtActionResult)actionResult.Result;
        var returnedMovement = Assert.IsType<InventoryMovementDTO>(createdAtActionResult.Value);

        Assert.Equal(inventoryMovementDTO.product_Name, returnedMovement.product);
        Assert.Equal(inventoryMovementDTO.inventory_Movement_Type_Name,
returnedMovement.movementType);
        Assert.Equal(inventoryMovementDTO.quantity, returnedMovement.quantity);
    }
    catch (Exception ex)
    {
        Assert.True(false, ex.Message);
    }
}
```

Ejemplo de prueba de integración

```
test('renders InventoryPage and shows edit modal', async () => {
  const user = { role: 'Presidente', email: 'user@example.com',
uid: 'mock-uid' };
  const token = 'mock-token';
  const loadingContext = false;

  render(
    <AuthContext.Provider value={{ user, token, loadingContext }}>
      <InventoryPage />
    </AuthContext.Provider>
  );

  // Verify initial render
  expect(screen.getByText(/Movimientos de
Inventario/)).toBeInTheDocument();
  expect(screen.getByText(/Antes de añadir un movimiento de
inventario/)).toBeInTheDocument();
  expect(screen.getByText(/Product A/)).toBeInTheDocument();

  // Click edit button and verify modal
  const editButton = screen.getByLabelText(/Edit Movement/);
  fireEvent.click(editButton);

  await waitFor(() => {
    expect(screen.getByText(/Editar
Inventario/)).toBeInTheDocument();
  });
});
```

5.7 Anexo VII. Repositorio que contiene el código fuente del proyecto

[Repositorio Backend: Disponible aquí.](#)

[Repositorio Frontend: Disponible aquí](#)

5.8 Anexo VIII. Tabla de objetivos a evaluar en cada heurística de usabilidad

[Tabla: Disponible aquí.](#)

5.9 Anexo IX. Formularios para la evaluación de usabilidad.

[Formulario encuesta SUS: Disponible Aquí.](#)

[Formulario encuesta NPS: Disponible Aquí.](#)

[Formulario evaluación de heurísticas: Disponible Aquí.](#)

5.10 Anexo X. Tabla de preguntas para evaluación de heurísticas.

[Tabla: Disponible aquí.](#)

5.11 Anexo XI. Resultado de evaluación de SortSite para las interfaces de productos y proveedores

[Resultados: Disponibles aquí.](#)