



Facultad de Ciencias de la Administración

**Carrera de Ingeniería de Sistemas y
Telemática.**

**DESARROLLO DEL MÓDULO DE BITÁCORA
DEL PROYECTO UDA-ERP**

**Trabajo de titulación previo a la obtención del
grado de Ingeniero de Sistemas y Telemática**

Autor:

Jorge Luis Peralta Peñaloza.

Directora:

Ing. Catalina Verónica Astudillo Rodríguez.

Cuenca – Ecuador

2023

DEDICATORIA

A mis padres Diego y Celia y
a mis hermanos Cristina y Juan Diego,
sin ustedes nada de esto hubiera sido posible.

AGRADECIMIENTO

A mis padres y hermanos por su apoyo incondicional durante mi formación profesional y personal. Ustedes son lo más importante de mi vida.

A mi familia por motivarme a seguir adelante y ser un ejemplo de superación.

A mis amigos y amigas por compartir conmigo días de trabajo y también horas de ocio. Diego y Xavier, un agradecimiento especial para Ustedes.

A la Universidad del Azuay y a mi directora de tesis, Ing. Catalina Astudillo por dirigirme este proyecto con empeño y dedicación, siempre enseñándome con cariño la responsabilidad ética que tiene la profesión. Así también agradezco al Ing. Santiago Cedillo y al Ing. Luis Guerrero por su ayuda en cada etapa de este trabajo.

Índice de Contenidos

Dedicatoria	i
Agradecimiento.....	ii
Índice de Contenidos.....	iii
Índice de Figuras.....	iv
Índice de Tablas	vi
Índice de Anexos	vii
Resumen y Abstract	viii
1. INTRODUCCIÓN	1
1.1 Objetivos	1
1.2 Marco teórico	1
MIPYMES.....	1
Auditoría	2
Bitácora.....	2
Oracle APEX.....	2
Python	3
Framework Django	3
ORM	3
PostgreSQL	3
Syncfusion.....	4
2. ESTADO DEL ARTE.....	4
3. MÉTODOS.....	5
3.1 Análisis del problema.	5
3.2 Formulación del problema.	5
3.3 Revisión del estado del arte.	5
3.4 Solución candidata.....	6
3.4.1 Definición de la Pila del Producto.	6
3.4.2 Planificación de Sprint.....	7
3.4.3 Elaboración del Sprint.	8
Iteración 1. 152. Revisar la base de datos y definir un modelo ER para la Bitácora Nivel 2.....	8
Iteración 2. 153. Gestionar Tabla.....	11
Iteración 3. 154. Generar contenido bitácora nivel 2.....	16
Iteración 4. 155. Almacenar bitácora nivel 2.....	21
Iteración 5. 156. Visualizar bitácoras.....	25
Iteración 6. 157. Gestionar bitácora nivel 1.....	31
Iteración 7. 158. Gestionar bitácora nivel 2.....	33
3.5 Entrenamiento.	36
3.6 Validación inicial.....	36
3.7 Liberación de la solución.	36
4. RESULTADOS	36
4.1 Wireframe	36
4.2 Diseño visual.....	40
4.3 Software	40
4.4 Entrenamiento	43
5. CONCLUSIÓN.....	43
6. REFERENCIAS.....	45

Índice de Figuras

Figura 1 Metodología de la investigación.....	5
Figura 2 El proceso de Scrum.....	6
Figura 3 Modelo entidad relación.	9
Figura 4 Arquitectura de Python con el Framework Django y el PostgreSQL.	12
Figura 5 Diagrama de casos de uso de la gestión de la tabla.	13
Figura 6 Diagrama de secuencia de la gestión de la tabla.	14
Figura 7 Diagrama de casos de uso para generar contenido en la bitácora nivel 2.....	17
Figura 8 Diagrama de secuencia para generar contenido en la bitácora nivel 2.	18
Figura 9 Diagrama de casos de uso para almacenar la bitácora nivel 2.	21
Figura 10 Diagrama de secuencia para almacenar la bitácora nivel 2.....	22
Figura 11 Diagrama de casos de uso para visualizar bitácoras.	26
Figura 12 Diagrama de secuencia para visualizar bitácoras.....	27
Figura 13 Diagrama de casos de uso para gestionar bitácora nivel 1.	31
Figura 14 Diagrama de secuencia para gestionar bitácora nivel 1.	32
Figura 15 Diagrama de casos de uso para gestionar bitácora nivel 2.	33
Figura 16 Diagrama de secuencia para gestionar bitácora nivel 2.	34
Figura 17 Bitácoras en las tablas.	37
Figura 18 Seleccionar la bitácora nivel 1.	37
Figura 19 Seleccionar la bitácora nivel 2.	38
Figura 20 Listado de la tabla.....	38
Figura 21 Mantenimiento de la tabla.	39
Figura 22 Visualizar bitácora nivel 1 de todo el sistema.	39
Figura 23 Visualizar bitácora nivel 2 de todo el sistema.	40
Figura 24 Agregar un nuevo campo en el módulo de Rol.	41
Figura 25 Mantenimiento de la tabla de la versión 1.0.....	41
Figura 26 Mantenimiento de la tabla de la versión 2.0.....	42
Figura 27 Visualización de bitácora nivel 1.	42
Figura 28 Visualización de bitácora nivel 2.	43
Figura 29 Ingreso a la aplicación.....	49
Figura 30 Organización de elementos en pantalla de inicio.....	50
Figura 31 Cerrar la aplicación.....	50
Figura 32 Organización de elementos en pantalla del reporte interactivo.....	51
Figura 33 Visibilidad de la columna.....	51
Figura 34 Indicador del total de registros en pantalla.....	52
Figura 35 Organización de elementos en pantalla de datos de mantenimientos, opción Agregar.....	52
Figura 36 Organización de elementos en pantalla de datos de mantenimientos, opción Editar.....	53
Figura 37 Confirmación de datos guardados.....	53
Figura 38 Cuadro de diálogo para eliminar registro.....	53
Figura 39 Elemento de texto.....	54
Figura 40 Elemento para listas de selección.....	54
Figura 41 Elementos en lista de selección.....	54
Figura 42 Elemento para calendario.....	54
Figura 43 Calendario.....	54
Figura 44 Botón para seleccionar imagen.....	55
Figura 45 Explorador de Windows para seleccionar imagen.....	55
Figura 46 Nombre de imagen seleccionada.....	55
Figura 47 Gestionar sistema.....	56
Figura 48 Listado de registros de la tabla.	56
Figura 49 Campos del formulario para la creación y modificación del registro.	57
Figura 50 Visualización de la bitácora nivel 1 para el administrador del sistema.	58
Figura 51 Visualización de la bitácora nivel 2 para el administrador del sistema.	59
Figura 52 Gestionar administración de seguridad en la opción de rol.	60

Figura 53 <i>Listado de los roles.</i>	60
Figura 54 <i>Creación del campo en la tabla de rol.</i>	60

Índice de Tablas

Tabla 1 <i>Identificación y descripción de la pila del producto.</i>	7
Tabla 2 <i>Pila del producto: prioridad, responsable, sprint y tiempo de ejecución.</i>	7
Tabla 3 <i>Descripción de las tablas utilizadas.</i>	8
Tabla 4 <i>Diccionario de datos de la tabla aud_auditoria_n2.</i>	9
Tabla 5 <i>Diccionario de datos de la tabla sis.tabla.</i>	10
Tabla 6 <i>Diccionario de datos de la tabla adm.usuario.</i>	10
Tabla 7 <i>Diccionario de datos de la tabla adm.empresa.</i>	11
Tabla 8 <i>Historias de usuario de la gestión de la tabla.</i>	13
Tabla 9 <i>Caso de prueba de la gestión de la tabla.</i>	14
Tabla 10 <i>Historias de usuario para generar contenido en la bitácora nivel 2.</i>	17
Tabla 11 <i>Caso de prueba para generar contenido en la bitácora nivel 2.</i>	18
Tabla 12 <i>Historias de usuario para almacenar la bitácora nivel 2.</i>	21
Tabla 13 <i>Caso de prueba para almacenar la bitácora nivel 2.</i>	23
Tabla 14 <i>Historias de usuario para visualizar bitácoras.</i>	26
Tabla 15 <i>Caso de prueba para visualizar bitácoras.</i>	28
Tabla 16 <i>Historias de usuario para gestionar bitácora nivel 1.</i>	31
Tabla 17 <i>Caso de prueba para gestionar bitácora nivel 1.</i>	32
Tabla 18 <i>Historias de usuario para gestionar bitácora nivel 2.</i>	34
Tabla 19 <i>Caso de prueba para gestionar bitácora nivel 2.</i>	35

Índice de Anexos

Anexo 1 <i>PostgreSQL</i>	47
Anexo 2 <i>Manual de Usuario</i>	49



RESUMEN

La gestión y seguimiento de la información generada al interior de las empresas es de gran importancia, en este sentido, mantener un historial de todos los cambios que sufre esta información es un asunto de prioridad. Para apoyar en este propósito, el presente trabajo tiene como objetivo el desarrollo del módulo de bitácora para el sistema UDA-ERP desarrollado por la Universidad del Azuay. Se aplicó dos metodologías: modelo de transferencia tecnológica y el marco de trabajo SCRUM, el desarrollo de software se realizó con: Python, Django y PostgreSQL. Como resultado, se cuenta con el módulo de bitácora fácil de usar por los administradores, al tiempo que almacena la información detallada de las acciones realizadas por el usuario durante su sesión. Finalmente, la importancia de tener un módulo de bitácora recae en el seguimiento de revisiones del sistema, los cuales ayudan en los procesos de auditoría.

Palabras clave: auditoría, bitácora, django, erp, postgresql, python, uda-erp

ABSTRACT

The management and monitoring of the information generated within the companies are of great importance; in this sense, keeping a history of all the changes that this information undergoes is a matter of priority. To support this purpose, this work aims to develop the logbook module for the UDA-ERP system developed by Universidad del Azuay. Two methodologies were applied: the technology transfer model and the SCRUM framework; the software development was done using: Python, Django and PostgreSQL. As a result, there is a log module that is easy to use by administrators while storing detailed information on the actions performed by the user during his session. Finally, the importance of having a log module lies in tracking system revisions, which helps in the auditing process.

Keywords: audit, django, erp, log, postgresql, python, uda-erp



Este certificado se encuentra en el repositorio digital de la Universidad del Azuay, para verificar su autenticidad escanee el código QR

Este certificado consta de: 1 página

1. Introducción

Las MIPYMES desempeñan un rol importante en la generación de empleo a nivel latinoamericano. En la era de la globalización, el acceso a información oportuna para la gestión empresarial constituye un aspecto clave en la Administración de Operaciones. El proyecto UDA-ERP versión 1.0, iniciado en el año 2014, cambió el manejo de la información de las micro, pequeñas y medianas empresas (MIPYMES) ecuatorianas. Este trabajo de vinculación con la comunidad eliminó el uso de registros electrónicos por un Software ERP de tecnología Cloud, brindando enormes beneficios a la industria ecuatoriana (Astudillo Rodríguez et al. 2018)

En este sentido, el crecimiento de las plataformas web se ha vuelto indispensable para el desarrollo empresarial. Oracle APEX es una herramienta que incluye ventajas para el desarrollo de aplicaciones: seguridad, escalabilidad, rapidez en el desarrollo, entre otros. Considerando su utilidad, los creadores del proyecto UDA-ERP versión 1.0 optaron por esta tecnología. Sin embargo, los costos de licenciamiento de esta plataforma son monetariamente significativos. En este contexto, el equipo de desarrollo a resuelto la migración del software a una nueva versión denominada UDA-ERP versión 2.0 con software libre, para garantizar una gestión idónea de los datos y a la vez asegurar su factibilidad financiera, de manera que el proyecto continúe brindando su apoyo a las micro, pequeñas y medianas empresas (Astudillo Rodríguez et al. 2018)

En este proceso de migración, el equipo de desarrollo ha visto la necesidad de adicionar un módulo de bitácora al proyecto UDA-ERP versión 2.0. Si bien los sistemas de gestión de bases de datos (SGBD) guardan un expediente por cada transacción realizada, estos no brindan una amplia información de los procesos ejecutados. Esto constituye un problema en el ámbito empresarial, pues es necesario conocer los datos del usuario que realizó dicha transacción, la fecha en la que lo hizo, qué valores fueron afectados (conociendo su valor anterior y actual), qué tabla o tablas fueron alteradas y cuáles fueron los campos utilizados en cada operación (Gómez Baryolo et al. 2012). Así mismo, es necesario un módulo de bitácora que almacene los eventos sucedidos, ya sean registros, actualizaciones y/o eliminaciones relacionadas con cada transacción.

El presente trabajo se orienta en apoyar en la consecución del módulo de bitácora, desarrollado en el lenguaje de programación Python, con el Framework Django y la base de datos PostgreSQL.

1.1 Objetivos

Contribuir con un módulo de bitácora al sistema UDA-ERP versión 2.0 de la Universidad del Azuay, desarrollado en Python con Framework Django y la base de datos PostgreSQL. Como punto de partida, se presentan los aspectos conceptuales y teóricos, con esta base, se desarrolla el módulo de bitácora que almacena el historial de acciones por parte del usuario en las MIPYMES y proporciona información de auditoría, posterior, se validan las funcionalidades del módulo de bitácora y se documenta en el manual de usuario.

1.2 Marco teórico

Es importante conocer los conceptos básicos de las diferentes temáticas que forman parte importante del desarrollado de este trabajo.

MIPYMES

Las micro, pequeñas y medianas empresas (MIPYMES), son un grupo de organizaciones que se encuentran en toda Latinoamérica. Las grandes empresas tanto públicas como privadas reconocen que las MIPYMES son importantes en el ámbito local, ya que proporcionan muchos servicios o productos que las personas interesadas pueden adquirir. Estas iniciativas son las que proporcionan una mejor economía al país (Zevallos Vallejos 2003).

La capacidad de generar empleo de las micro, pequeñas y medianas empresas (MIPYME) motiva el desarrollo de proyectos de investigación que aseguren su supervivencia y competitividad en el mercado. Empero, sus debilidades de carácter estructural limitan su desarrollo y dificultan la toma de decisiones oportunas. La Fundación para el Análisis Estratégico y Desarrollo de la Pequeña y Mediana Empresa (FAEDPYME por sus siglas en inglés) reconoce tres grandes retos actuales: “el desarrollo tecnológico, la

globalización de los mercados y la incertidumbre económica”. En este escenario, la innovación tecnológica se convierte en un aspecto clave para alcanzar ventajas empresariales permanentes (Pérez de Lema et al. 2011).

Las tecnologías de información se han establecido como una herramienta esencial en la actividad administrativa de las MIPYNES, pues contribuye a superar sus principales desventajas. Un claro ejemplo es lo sucedido a raíz de la emergencia sanitaria a causa del coronavirus, que cambia la forma de comunicarse y relacionarse con la sociedad afectando directamente la economía y productividad de las MIPYMES. Este hecho refleja la urgente necesidad de reinventar la propuesta comercial desde la tecnología, para agilizar procesos y mejorar el desempeño organizacional (Hoyos Estrada y Sastoque Gómez 2020).

Auditoría

Una auditoría consiste en la evaluación de inconsistencias en las empresas. Las organizaciones públicas del país están regladas por la “Ley Orgánica de la Contraloría del Estado” (LOCGE por sus siglas en inglés) y su reglamento. El trabajo de una auditoría permite encontrar anomalías en la información, y realizar una serie de recomendaciones para mejorar el rendimiento de una empresa (Quispe Fernández et al. 2016). Así, la auditoría se constituye como un “método de revisión periódica” (Piñuel 1997), donde se identifica la causa del problema y se establece una acción correctiva (Muñoz Razo 2002).

En este contexto, para asegurar que la información sea exacta, confidencial y esté disponible, es preciso implementar controles internos informáticos. La necesidad de transparencia en la gestión de las empresas y la búsqueda de eficiencia convierten a la auditoría en una necesidad básica (Piattini Velthuis et al. 2008). Muñoz Razo (2002) considera como principales objetivos de la auditoría de sistemas:

- Evaluar con personal multidisciplinario para realizar un dictamen independiente sobre la razonabilidad del sistema.
- Hacer una evaluación sobre los recursos financieros utilizados en el aprovechamiento del sistema computacional.
- Analizar el uso y aprovechamiento de los equipos, así como los recursos técnicos y materiales usados en el procesamiento de información.
- Estudiar los sistemas de procesamiento, operativos, lenguajes, programas y paqueterías de aplicación y desarrollo.
- Valorar el cumplimiento de planes y proyectos que regulan las funciones de los sistemas computacionales.
- Evaluar las áreas, actividades y funciones de la empresa que cuentan con apoyo de sistemas computacionales, programas especiales de auditoría y de paquetería.

Bitácora

Una bitácora almacena información relevante del usuario como la fecha-hora, cambios realizados, nombre de la persona, entre otros; generando un registro de las acciones de una determinada organización. Éste constituye un respaldo de datos, el cual puede ser utilizado para identificar cualquier tipo de irregularidades en la empresa y sean corregidos de manera inmediata (Gómez Baryolo et al. 2012). Es preciso que las bitácoras sean validadas mediante expertos o el desarrollo de pilotajes previo a su implementación (Inalef et al. 2010)

Oracle APEX

Oracle APEX es una herramienta de desarrollo de Software fácil de utilizar, en la cual se pueden implementar sistemas completamente escalables, y lo más importante, seguros. La rapidez en el aprendizaje de esta aplicación permite dar soluciones oportunas a los problemas que existen en la actualidad, puesto que despliegan una serie de funciones que pueden ser utilizadas por los desarrolladores, para que su aplicación tenga un rendimiento adecuado y que pueda ser lanzada en el menor tiempo posible (Adelfio et al. 2012).

Adicionalmente este software, ayuda a los problemas de los usuarios ya sean grandes o pequeños, gracias a que gran parte de su elaboración no incurre a grandes cantidades de código, caso contrario una aplicación desarrollada en Oracle APEX contiene el menor código posible durante su desarrollo. Finalmente, contiene distintos modos de aprendizaje: documentación, tutoriales, videos, libros, entre otros. En los cuales un desarrollador se puede basar para elaborar un software que ayude a cumplir las necesidades de los

usuarios, a su vez las distintas licencias que ofrece garantiza; que dependiendo del tipo de programa que se elabora sea el mejor posible (Oracle 2022).

Python

Python es un lenguaje de programación de alto nivel creado por Guido Van Rossum, en el cual los desarrolladores pueden generar distintos algoritmos para la elaboración de los programas. Es un lenguaje que soporta múltiples paradigmas, es decir que se puede programar bajo diferentes modelos de programación como: Programación Orientada a Objetos, Imperativo, Estructural, Basado en Aspectos, entre otros. Proporciona diversas librerías para agilizar los procesos, se puede desarrollar para la web, para proyectos de Visualización de Datos, Análisis de Datos, Aprendizaje Profundo, etc. (Challenger Pérez et al. 2014).

Es un lenguaje sencillo en cuanto a aprendizaje, así también, para la elaboración de algoritmos, puesto que es similar a los programas que se realizan en pseudocódigo por ello es uno de los más utilizados. A su vez puede ser ejecutado de dos maneras por un intérprete o por una línea de comandos, también es soportado por múltiples sistemas operativos como Windows, Linux, Mac, entre otros. El manejo información de este lenguaje no existe, prácticamente no tienen una forma en la cual muestre que tipo de datos es ya sea entero, cadena, carácter, decimal, booleano, entre otro. En consecuencia, este tipo de datos se maneja de manera interpretada para que su programación y aprendizaje sea el más sencillo. Pero existen maneras de transformar un tipo de dato a otro para que tengan mayor libertad de utilizar las distintas variables que existen (González 2011).

Framework Django

Django es un Framework escrito con el lenguaje de programación Python. Con este se puede crear páginas o plataformas web de manera rápida y sin mucho código, puesto que contiene una serie de herramientas para facilitar el desarrollo de los sitios web. Su gratuidad asegura que pueda ser utilizado por todo tipo de desarrolladores (Molina Ríos et al. 2016). Cabe mencionar que este Framework utiliza el Modelo Vista Plantilla (MVT por sus siglas en inglés) el cual funciona y controla todas las peticiones y da una respuesta HTTP de esta forma aumenta la seguridad. A su vez, contiene una administración por defecto la cual permite controlar todos los registros de la base de datos y esta a su vez se crear un diseño personalizable.

El framework incluye funciones, herramientas, métodos, procedimientos, entre otros, las cuales pueden ser modificadas para que actúen de acuerdo a requerimientos específicos. Así mismo, permite editar la configuración por defecto y de esta manera puede comunicarse con diferentes controladores de base de datos, realizar un cambio de idiomas, aspectos de seguridad, la url predeterminada, login, logout, entre otros aspectos básicos (Holovaty y Kaplan-Moss 2008).

ORM

Django contiene una técnica para comunicarse con cualquier base de datos la cual se denomina ORM, y permite realizar consultas más seguras a la base de datos, ya que los sitios o plataformas web desarrollados con este sistema son infalibles a los ataques de inyecciones de bases de datos. Esta técnica fue diseñada para que sea similar al paradigma orientado a objetos (OOP por sus siglas en inglés), en consecuencia, la base de datos en la cual almacena todos los registros sería utilizada directamente con el ORM haciendo que la tabla, se relacione como el objeto en sí, y los atributos serían las columnas de esa tabla, por ello resulta que ORM sea fácil de entender y de utilizar durante el desarrollo de un programa (Duissebekova et al. 2021) (Salas-Zárate et al. 2015).

PostgreSQL

Es un gestor de base de datos de Open Source, esto quiere decir que las mejoras existentes son desarrolladas por un grupo de personas de manera independiente. Contiene procedimientos, funciones, integridad de los datos, y otras funcionalidades, así como múltiples tipos de datos para gestionar las columnas, finalmente puede soportar datos como arreglos, reglas, vistas, entre otras (Zea Ordoñez et al. 2017). PostgreSQL incorpora un diseño cliente-servidor; el cliente puede ser gestionado de dos formas: en un entorno gráfico (pgaccess) y en consola (psql); y el servidor postgres. Algunas de las características de este gestor es la alta concurrencia, es decir, que mientras uno escribe, otro puede leer de manera simultánea, este tipo de gestor de base de datos puede ser utilizado por distintos lenguajes de programación (Ginestà y Mora 2012).

Syncfusion

Es una plataforma de código abierto que facilita la creación de entornos automatizados. Su propósito central es suministrar componentes y marcos de software que faciliten la interacción del usuario. Entre sus principales funciones, destaca el enlace de datos, deslizamiento, clasificación, exportación de archivos y edición. Bajo este contexto, sus autores la describen como una solución de vanguardia que está en constante desarrollo (Syncfusion Inc. 2001).

2. Estado del Arte

Ahmad et al. (2018) señalan que las tablas de registros donde se almacenan datos críticos de los sistemas empresariales son un componente esencial al momento de auditar, debido a que detectan cambios y anomalías de forma rápida y sencilla. Por lo tanto, contar con un diseño adecuado de los registros que, a la vez, evite manipulaciones de todo tipo es clave para asegurar la confiabilidad del sistema. Con este propósito, los autores realizan un diseño en "BlockAudit", el cual garantiza que los registros no puedan ser modificados por terceras personas aportando seguridad, integridad y una mínima tolerancia a fallos. Este diseño tiene una particularidad, el ser desarrollado de manera escalable, lo que garantiza la seguridad en los registros y que sean confiables. A su vez el diseño fue evaluado en tres aspectos: latencia, red y carga. De este modo garantiza que los datos se mantengan intactos contra ataques de manipulaciones y minimiza las fallas.

En este sentido, los registros se utilizan para monitorear datos, actividades informáticas y redes. También los académicos de las Universidades de Québec à Chicoutimi y Laval de Canadá, recalcan su importancia en la investigación reportada, en donde describen que al analizar los datos es posible detener de manera rápida ataques cibernéticos o salidas de información para evitar daños relevantes en una organización. Además, realizan una revisión sistemática de los distintos tipos de registros de seguridad para proveer una mayor protección contra escapes de información. Adicionalmente este trabajo muestra la forma en la que se pueden utilizar los registros para detectar anomalías, e insisten que los problemas con las filtraciones de datos son muy comunes. Desde su perspectiva, hoy en día es indispensable encontrar nuevas maneras de mantener la información intacta contra problemas de vulnerabilidades de los datos (Ávila et al. 2021).

Así se reconoce que la creación de un módulo de bitácora en ERP es cada vez más utilizado y necesario para el manejo de empresas. Autores como Roumiana y Georgiev (2019) exponen que el rápido desarrollo de los ERP obliga a una actualización de las funcionalidades del sistema predecesor. En este sentido, tener un módulo adecuado de seguridad en el sistema es importante para asegurar confiabilidad en todos sus componentes. Cabe mencionar, que este módulo debe estar en constante cambio debido a los nuevos desafíos que implica la protección de información, asegurando la ejecución de auditorías y revisiones a los módulos para contar con un ERP de calidad.

Vale citar a los autores M. Hadidi y S. Hadidi (2020), quienes mencionan que los sistemas ERP desarrollados en Jordania son muy importantes en las empresas tanto públicas como privadas. Para ellos, los ERP necesitan servicios web debido a su vulnerabilidad frente a los ataques informáticos que han causado problemas en la seguridad de las organizaciones, ya que contienen módulos importantes con información delicada. Ellos establecen cuatro fundamentos necesarios para garantizar una buena defensa: "no repudio, autenticidad, confidencialidad y disponibilidad". Añaden que cualquier ERP debe cumplir los siguientes requisitos: "autenticación, autorización, integridad de datos y pistas de auditorías". De esta manera, el sistema de seguridad constituye un aspecto esencial en toda empresa y merece especial atención para evitar adulteraciones en la información.

En torno a la trascendencia del sistema de seguridad, es pertinente citar el trabajo realizado por Hrishev (2020), quien señala que los ERP son muy importantes en la gestión de la información porque contienen todos los procesos de negocio de las empresas: sin embargo, tener un sistema de seguridad de los datos es un punto sustancial en las organizaciones. Ante ello, los desarrolladores de los ERP han generado soluciones para garantizar seguridad en Software. Con los avances tecnológicos, los ERP son accesibles, flexibles y brindan mayor libertad para que los usuarios puedan registrar sus transacciones. Por lo tanto, buscar nuevas arquitecturas de la seguridad de información, políticas de autenticación y autorización, sistemas y dispositivos que mantengan los datos ocultos en la web es esencial. Y esto debe ir acompañado de la creación de nuevas leyes que garanticen una protección de datos más eficiente. De esta manera, se podrá tener una adecuada transparencia de los datos.

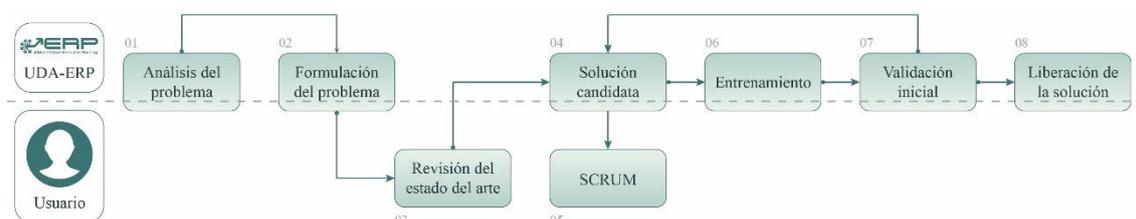
Finalmente, Rajavat et al. (2020) describe que en la era digital -en la que viven los empresarios- se deben tener los datos controlados, garantizando seguridad y privacidad. Actualmente, distintos usuarios exigen un sistema para controlar la privacidad de su información conforme a ciertos requisitos, de manera que ningún otro usuario pueda utilizar esos datos o servicios como él o ellos requieran. En consecuencia, mantener una seguridad de la información y servicios que estos proveen a sus usuarios es vital, debido que los problemas de seguridad son cada vez más habituales. Finalmente, estos autores mencionan dos ataques activos y pasivos con el único propósito de vulnerar los datos y valores que la empresa ofrece a sus usuarios finales.

Como se pudo evidenciar en los artículos, tener un registro de las acciones de los usuarios para detectar cualquier tipo de problemas en la organización es ampliamente necesario, de esta forma ayuda a la empresa a tener un mayor control de los módulos indispensables dentro de un ERP como el módulo de contabilidad, inventarios, usuarios, entre otros, en los cuales la seguridad de la información es de vital importancia. En tal sentido, tener este registro de eventos puede solucionar cualquier tipo de incongruencias de datos, filtraciones de información y cambios. Esta exposición contribuye a realzar la importancia que tendrá este módulo en el proyecto ERP para solucionar los problemas empresariales.

3. Métodos

Para el desarrollo de la propuesta relacionada con el módulo de bitácora del proyecto UDA-ERP versión 2.0, se unifican dos metodologías: la metodología del modelo de transferencia tecnológica propuesta por (Gorschek et al. 2006) y la metodología de desarrollo ágil (Scrum 2020) que se aplica en la fase 4, es decir en el desarrollo de la solución candidata. Esta propuesta metodológica facilita la implementación del módulo de bitácora al proyecto UDA-ERP versión 2.0 que apoye en la revisión de información en el sistema, como se muestra en la figura 1.

Figura 1
Metodología de la investigación.



Fuente: Basado en el modelo de transferencia tecnológica.

3.1 Análisis del problema.

Se diagnostica la situación a resolver, para esto, se realiza un análisis para identificar las necesidades del sistema, los posibles problemas que presenta y las oportunidades para el desarrollo del nuevo módulo de seguridad.

3.2 Formulación del problema.

Una vez realizado el análisis, se procederá a describir de manera clara y precisa el problema a abordar. Esto permite identificar los requisitos del módulo de bitácora, y definir las distintas herramientas que serán utilizadas para plantear una solución.

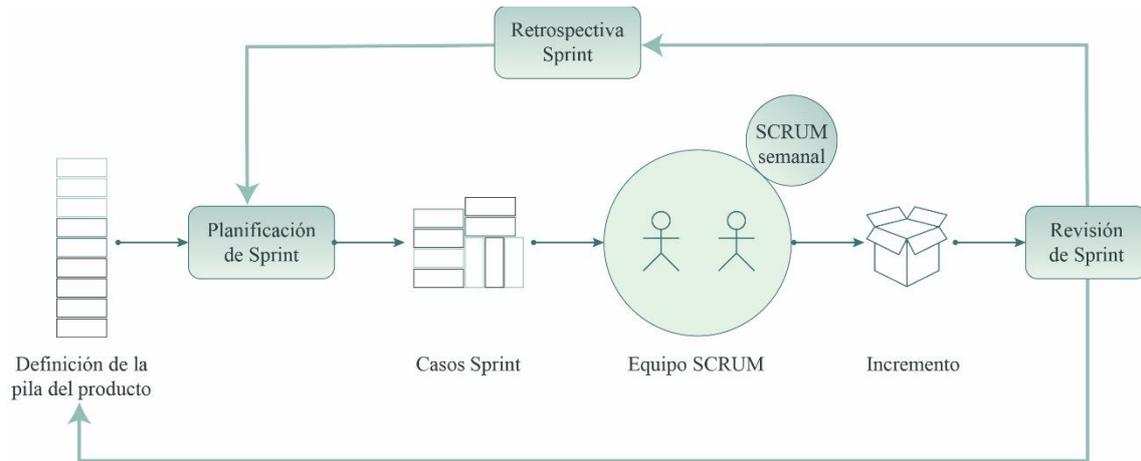
3.3 Revisión del estado del arte.

Se realiza una revisión de la literatura, a fin de recopilar investigaciones reportadas sobre problemas y soluciones en campos similares de software ERP, esto proporciona una visión general de otros estudios ya existentes y que se pueden aplicar al proyecto UDA-ERP versión 2.0.

3.4 Solución candidata.

Se desarrolla la propuesta del software para el módulo de bitácora; una vez que se cuente con la solución, se propone también, la ejecución de pruebas de funcionalidad. Para el seguimiento y control del desarrollo de la solución candidata se aplica SCRUM, el cual es un marco para el desarrollo ágil, es decir es un método utilizado para desarrollar proyectos en un corto tiempo con buenos resultados, esto debido a que, cada cierto tiempo se presentan avances que cubren las necesidades del usuario (Scrum 2020). Las fases de SCRUM se detallan en la Figura 2.

Figura 2
El proceso Scrum.



Fuente: Basado en Scrum.org. (2020).

3.4.1 Definición de la Pila del Producto.

Constituye un proceso de planeación, en el cual se especifican las necesidades y tareas prioritarias a realizarse en un tiempo determinado para el desarrollo del módulo de bitácora, pueden ser: implementación, corregir errores, añadir nueva funcionalidad, entre otros. Esta fase agiliza el desarrollo del sistema, permite al programador definir los principales requerimientos y asegura un proceso de validación efectivo. Una vez examinado los inconvenientes del usuario se analiza las oportunidades de solución para plantear una o varias propuestas.

Como paso previo a la elaboración del módulo de bitácora para el proyecto UDA-ERP versión 2.0 se analizan las necesidades del usuario para identificar y reconocer el o los problemas específicos a solucionar. Esta acción constituye un paso fundamental para detectar las dificultades al momento de comunicarse con el usuario y facilita la generación de propuestas adecuadas a los clientes. En este escenario, es clave establecer una comunicación efectiva entre los usuarios y los desarrolladores para configurar un acuerdo de trabajo en el marco de la corresponsabilidad y la participación colectiva.

La principal falencia del sistema UDA-ERP versión 2.0 que identifica el usuario se relaciona con las limitaciones en el acceso al registro de las acciones ejecutadas por el personal en el software. Esta situación impide analizar los inconvenientes que ocurriesen en el sistema, así como determinar aquellos procedimientos que puedan afectar a la MIPYME. Es por ello que, el registro constituye un aspecto esencial para la gestión de procesos de auditoría tanto interna como externa. Esto contribuye a que las tareas se planteen en coordinación con los requerimientos dados por el cliente para que la solución propuesta se adapte a la realidad actual y satisfaga todas las necesidades.

Conocidas las necesidades, se identifican las actividades fundamentales para el desarrollo del módulo de bitácora, el grado de prioridad y la persona responsable para garantizar una planificación coherente con el objetivo del proyecto. Considerando las historias de usuario y los trabajos de titulación paralelos que se realizan en el proyecto UDA-ERP versión 2.0, la numeración inicia en 152.

Tabla 1*Identificación y descripción de la pila del producto.*

ID	Descripción
152	Revisar la base de datos y definir un modelo ER para la Bitácora Nivel 2.
153	Gestionar Tabla
154	Generar contenido bitácora nivel 2
155	Almacenar bitácora nivel 2
156	Visualizar bitácoras
157	Gestionar bitácora nivel 1
158	Gestionar bitácora nivel 2

3.4.2 Planificación de Sprint.

Describe los requerimientos de las tareas que se tienen que completar en cada iteración. Las actividades identificadas se enlistan, indicando su grado de prioridad, el nombre del usuario responsable de su ejecución, el número de repetición en Sprint y el tiempo necesario para realizarla

Tabla 2*Pila del producto: prioridad, responsable, sprint y tiempo de ejecución.*

ID	Descripción	Prioridad	Responsable	Sprint	Semanas
152	Revisar la base de datos y definir un modelo ER para la Bitácora Nivel 2.	1	Jorge Peralta	1	1
153	Gestionar Tabla		Jorge Peralta	2	3
153.1	Recibir capacitación sobre el funcionamiento de Syncfusion.	2			0,25
153.2	Administrar las columnas para el mantenimiento de la tabla.	3			1
153.3	Dar acceso para generar contenido de bitácora nivel 2 en la tabla.	4			1
153.4	Reconocer las llaves foráneas que existen en la tabla.	5			0,25
153.5	Administrar procesos de contabilización.	6			0,5
154	Generar contenido bitácora nivel 2		Jorge Peralta	3	1,5
154.1	Crear registros mientras se realiza el mantenimiento de las tablas.	7			1,5
155	Almacenar bitácora nivel 2		Jorge Peralta	4	1,5
155.1	Guardar los registros generados.	8			0,5
155.2	Anular la opción de edición de registros en el sistema.	9			0,5
155.3	Crear el procedimiento "pr_respaldarauditoria()" para garantizar el respaldo de información en un archivo plano cada seis meses.	10			0,5
156	Visualizar bitácoras		Jorge Peralta	5	2,4
156.1	Adaptar íconos para mostrar las bitácoras nivel 1 y 2.	11			0,5
156.2	Generar pantallas emergentes para visualizar los contenidos de las bitácoras.	12			1
156.3	Administrar la visualización de datos según los permisos que tiene cada rol.	13			0,75
153.4	Filtrar por empresa.	14			0,15
157	Gestionar bitácora nivel 1		Jorge Peralta	6	1,2
157.1	Realizar una lista de todas las tablas que dispone el sistema.	15			0,5
157.2	Generar un botón de búsqueda.	16			0,4
157.3	Filtrar de manera ascendente y descendente para mejorar la visualización.	17			0,3

158	Gestionar bitácora nivel 2	Jorge Peralta	7	1,4
158.1	Realizar una lista de todas las tablas que dispone el sistema.	18		0,5
158.2	Generar un botón de búsqueda.	19		0,4
158.3	Filtrar de manera ascendente y descendente para mejorar la visualización.	20		0,3
158.4	Filtrar por opciones de registros en columnas.	21		0,2

3.4.3 Elaboración del Sprint.

Esta fase se estructura con los siguientes elementos, expuestos en el proceso SCRUM de la Figura 2:

- Casos Sprint. Se la conoce como ejecución o sprint en el cual se empieza a desarrollar el proyecto de acuerdo al plazo tiempo definido en las historias de usuario. En esta fase se detallan los objetivos de la pila de producto y se traza un plan para enfrentar el problema. Este proceso se compone de iteraciones sucesivas hasta encontrar la opción más adecuada que satisfaga la necesidad del cliente.
- Equipo SCRUM. Identifica al grupo responsable de la ejecución del SCRUM quienes mantienen reuniones semanales para garantizar agilidad y eficiencia en el trabajo. En este caso, el equipo está conformado por Jorge Peralta Peñaloza y Catalina Astudillo Rodríguez.
- Incremento. Hace referencia a los resultados del sprint en función a los objetivos del producto. En esta etapa, se agrupan los incrementos para ser ejecutados de manera sucesiva hasta que se encuentre la solución más adecuada que satisfaga la necesidad del cliente. Terminado el plazo se debe revisar el proceso, asegurar su funcionamiento y corregir oportunamente cualquier inconveniente.
- Revisión de Sprint. De conformidad a los objetivos de las tareas, se evalúa cada incremento para verificar los resultados, prever errores y poder avanzar con el sistema.
- Retrospectiva del Sprint. Finalmente, se realiza una regresión para identificar nuevas ideas que permitan garantizar un proceso más eficaz para el proyecto UDA-ERP versión 2.0.

A continuación, se realiza cada uno de las iteraciones definidas en la planificación del Sprint.

Iteración 1.

152. Revisar la base de datos y definir un modelo ER para la Bitácora Nivel 2.

La base de datos que utiliza el proyecto UDA-ERP versión 2.0 es PostgreSQL, el cual contiene toda la información del sistema, por ejemplo: sus funciones, procedimientos y triggers. El modelo de base de datos se realiza con Toad Data Modeler, por ser el modelador estándar del proyecto.

Modelo entidad relación

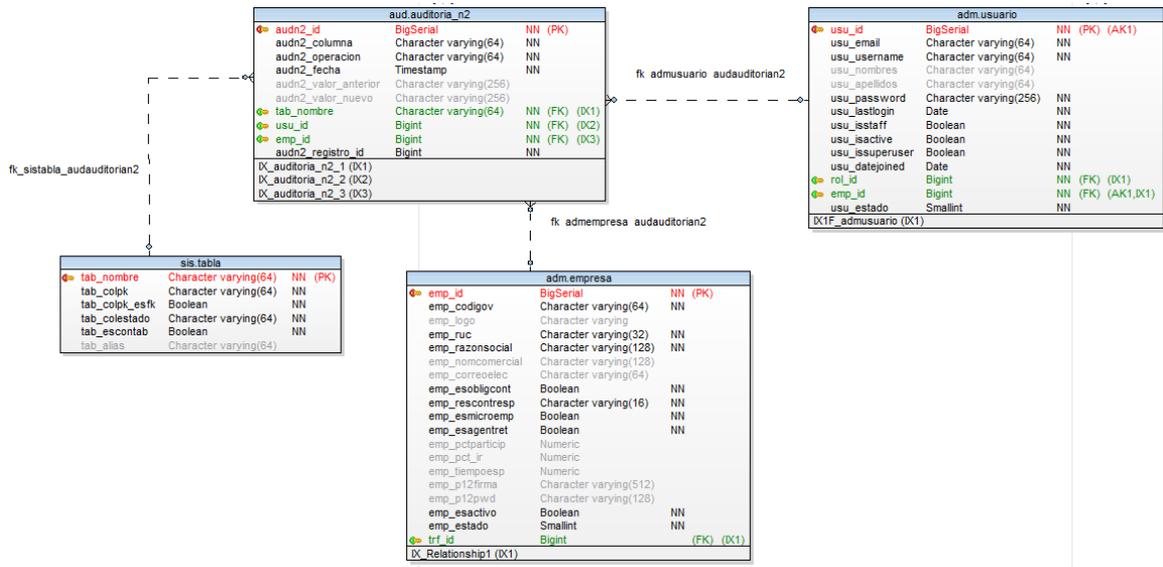
Para el diseño del modelo entidad relación se utiliza “Toad Data Modeler”, el cual cumple con los estándares establecidos en el proyecto UDA-ERP versión 2.0. Con este software de modelado se realiza la estructura de las tablas, la creación de esquemas, tipos de datos, referencias, descripción, etc. Como se puede observar en la Figura 3, el modelo incluye cuatro tablas, las cuales cumplen distintas funcionalidades.

Tabla 3

Descripción de las tablas utilizadas.

Tabla	Descripción
aud_auditoria_n2	Información de la bitácora nivel 2.
sis.tabla	Información de la tabla en la que se realiza la acción.
adm.usuario	Información del usuario que interactúa con la aplicación.
adm.empresa	Información de la empresa.

Figura 3
Modelo entidad relación.



Diccionario de datos

El diccionario de datos representa la forma técnica de como la información está estructurada en el sistema. En la tabla 4, se muestra los metadatos de la tabla aud_auditoria_n2 para el proceso de la bitácora nivel 2. Aquí, se indica el tipo de dato admitido para cualquier tipo de mala interpretación de información.

Tabla 4
Diccionario de datos de la tabla aud_auditoria_n2.

Columna	Valor por defecto	Valores nulos	Tipo de dato	Longitud	Descripción
audn2_id	Auto incrementable	No	BigSerial		Llave primaria
audn2_columna		No	Character varying	64	Columna de edición
audn2_operacion		No	Character varying	64	Acción que realiza
audn2_fecha	Fecha actual	No	Timestamp		Fecha actual
audn2_valor_anterior		Si	Character varying	256	Valor cambiante de las tablas
audn2_valor_actual		Si	Character varying	256	Valor actual de las tablas
tab_nombre		No	Character varying	64	Id de la tabla
usu_id		No	Bigint		Id del usuario
emp_id		No	Bigint		Id de la empresa
audn2_registro_id		No	Bigint		Id del registro de la tabla que se modifica

En las tablas 5, 6 y 7 están los metadatos de las tablas usadas para la elaboración de la tabla aud_auditoria_n2.

Tabla 5*Diccionario de datos de la tabla sis.tabla.*

Columna	Valor por defecto	Valores nulos	Tipo de dato	Longitud	Descripción
tab_nombre		No	Character varying	64	Llave primaria
tab_colpk		No	Character varying	64	Nombre del id de las tablas
tab_colpk_esfk	False	No	Boolean		La llave de la tabla es foránea de otra tabla
tab_colestado		No	Character varying	64	Nombre del estado de la tabla
tab_escontab	False	No	Boolean		Verifica si la tabla es contable
tab_alias		Si	Character varying	64	Descripción de la tabla
tab_aud_nivel	0	No	Small Int		Verificar la bitácora nivel 2

Tabla 6*Diccionario de datos de la tabla adm.usuario.*

Columna	Valor por defecto	Valores nulos	Tipo de dato	Longitud	Descripción
usu_id	Auto incrementable	No	BigInt		Llave primaria
usu_email		No	Character varying	64	Correo del usuario
usu_username		No	Character varying	64	Nombre para el inicio de sesión
usu_nombres		Si	Character varying	64	Nombres del usuario
usu_apellidos		Si	Character varying	64	Apellidos del usuario
usu_password		No	Character varying	256	Contraseña encriptada del usuario
usu_lastlogin		Si	Date		Fecha de inicio de sesión
usu_isstaff		No	Boolean		
usu_isactive	1	No	Boolean		Marca si el usuario está activo
usu_issuperuser		No	Boolean		Si es un usuario administrador
usu_datejoined		No	Date		Fecha en que se registra el usuario
rol_id		No	BigInt		Id de la tabla de rol
emp_id		No	BigInt		Id de la tabla empresa
usu_estado	1	No	SmallInt		Estado del usuario

Tabla 7*Diccionario de datos de la tabla adm.empresa.*

Columna	Valor por defecto	Valores nulos	Tipo de dato	Longitud	Descripción
emp_id	Auto incrementable	No	BigInt		Llave primaria
emp_codigov		No	Character varying	64	Identificador de la empresa
emp_logo		Si	Character varying		Logo de la empresa
emp_ruc		No	Character varying	32	Ruc de la empresa
emp_razonsocial		No	Character varying	128	Razón social de la empresa
emp_nomcomercial		Si	Character varying	128	Nombre comercial de la empresa
emp_correoelec		Si	Character varying	64	Correo electrónico de la empresa
emp_esobligcont		No	Boolean		Si la empresa es obligada a llevar contabilidad
emp_rescontresp		No	Character varying	16	Número de contribuyente especial
emp_esagentret		No	Boolean		Si la empresa es agente de retención
emp_pctparticip		Si	Numeric		Porcentaje de participación de los trabajadores
emp_pct_ir		Si	Numeric		Porcentaje impuesto a la renta.
emp_tiemposesp		Si	Numeric		Tiempo máximo de espera para respuesta de autorización en segundos
emp_p12firma		Si	Character varying	512	La firma electrónica de la empresa
emp_esactivo		No	Boolean		Si la empresa está activa o no
emp_estado		No	Small Int		El estado de la empresa

Iteración 2.**153. Gestionar Tabla.**

En esta iteración se diseña la arquitectura a utilizar para la plantear la solución, en este caso: el lenguaje de programación Python y el uso de su Framework Django. También se elabora un diagrama de casos de uso, diagrama de secuencia y casos prueba.

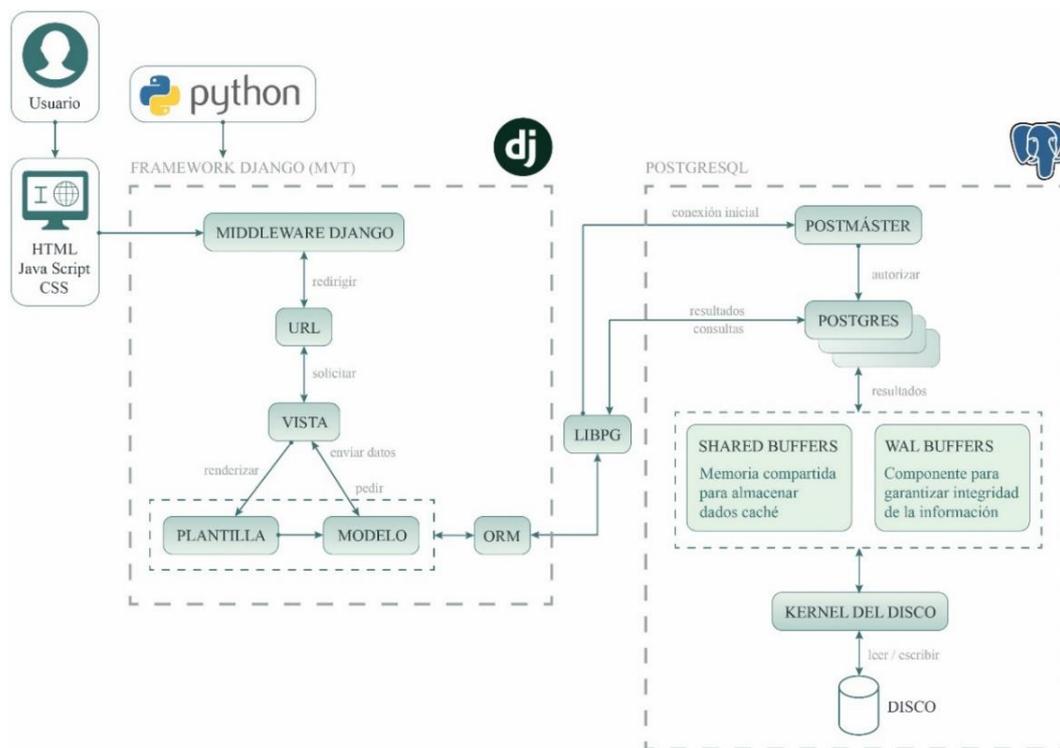
Arquitectura de software

El módulo de bitácora del Proyecto UDA-ERP versión 2.0 se desarrolla mediante la interrelación del Framework Django que sigue el lenguaje de programación Python y el sistema gestor de base de datos PostgreSQL. El primero facilita el desarrollo de sistemas de información web de manera fácil, ágil y segura, e implementa la arquitectura Modelo-Plantilla-Vista, conocido como MVT por sus siglas en inglés (Vidal-Silva et al. 2021). Este marco de trabajo emplea una capa Object Relational Model - ORM que permite interaccionar con bases de datos relacionales como PostgreSQL, el cual se sustenta en el modelo Cliente/Servidor basado en multiprocesos para garantizar la estabilidad del sistema. La principal ventaja de estas herramientas es contar con licencia libre para desarrollar aplicaciones de acuerdo a los diferentes requerimientos de los usuarios (Ginestà y Mora 2012).

Bajo estos fundamentos, la arquitectura de la plataforma UDA-ERP versión 2.0 se enfoca en un patrón de interacciones donde el Framework Django se convierte en el cliente y PostgreSQL actúa como servidor. Como ilustra la Figura 3, el Front end representa la conexión o interacción entre el usuario y la página web HTML, Java Script, CSS, entre otras. De este contacto, el Middleware de Django produce un marco de enlaces para facilitar el procesamiento de consultas y respuestas. A partir de este complemento, Django determina el módulo raíz y el patrón de URL coincidente con la petición del cliente. Cada uno de los argumentos generados se importan a la vista basada en clases para gestionar las solicitudes. Aquí se define un API estándar para cargar y renderizar plantillas en función de los datos de contexto. La información se estructura en diferentes modelos que contiene campos y comportamientos esenciales en tablas de bases de datos (Django 2022). El modelo actúa directamente con el ORM que media para la conexión con la base de datos PostgreSQL.

Para establecer la conexión entre Python y la base de datos PostgreSQL se utiliza la librería LIBPQ (Library Pooled Query), la cual contiene los detalles de la consulta realizada por el usuario y permite que éstos reciban los resultados de sus consultas. Así, se inicia con la configuración en Python mediante usuario y contraseña para ser reenviados a Postmaster, estos datos llegan directamente a Postmaster y, éste permite o no el paso al cliente para que pueda efectuar el diálogo con la base de datos. Una vez autorizado por el Postmaster, Postgres puede recibir múltiples consultas de manera simultánea. Así, ORM hace las consultas SQL necesarias y, con la ayuda de Shared Buffers y Wal Buffers se garantiza que los datos sean íntegros y sean almacenados momentáneamente si ocurriese un error. Finalmente, pasa al Kernel del Disco para realizar la lectura y escritura del o los registros en el Disco.

Figura 4
Arquitectura de Python con el Framework Django y el PostgreSQL.



Fuente: Basado en la documentación de Django Software Foundation y la Maqueta DCC PostgreSQL de la Universidad de Córdoba.

Diagrama de casos de uso.

En función de las diferentes opciones del sistema, se identifican tres tipos de usuario: Administrador del sistema, Administrador de la empresa y Usuario Normal. En esta iteración, únicamente se hará referencia al primero, el cual desempeña un rol fundamental en todo el proceso al ser el encargado de gestionar la bitácora. Este usuario puede acceder a todas las opciones disponibles en el sistema como se observa en la Figura 5.

Figura 5

Diagrama de casos de uso de la gestión de la tabla.



Historias de usuario

Las historias de usuario son descripciones cortas, en el cual detallan de manera informal las funciones que tiene el sistema. Generalmente son utilizadas como fichas o expedientes en el cual almacena lo esencial del Software. En este caso se utiliza para describir los módulos de gestión de la tabla, bitácora nivel 1 y bitácora nivel 2. La estructura en que se basan todas las historias de usuario presentes en este trabajo es en el formato realizado por los desarrolladores e investigadores del proyecto UDA-ERP versión 2.0 junto con su coordinadora.

Tabla 8

Historias de usuario de la gestión de la tabla.

153	Tabla				
COMO	Administrador				
QUIERO	Gestionar todas las tablas modeladas para el software UDA-ERP versión 2.0				
PARA	Administrar atributos asignados a cada tabla y gestionar la bitácora de nivel 2				
Estimación: (horas/días)	120 horas				
Prioridad:	1				
Criterios de aceptación	1. La aplicación podrá listar, crear, modificar las tablas que existen en la base de datos 2. La aplicación podrá activar/desactivar el historial de bitácora a nivel 2 3. La aplicación no permitirá la eliminación de un registro				
Acción de la aplicación	1. Aud_nivel: <table border="1" style="display: inline-table; vertical-align: top;"> <tr> <td>- Si:</td> <td>Activa la creación de registros de bitácora nivel 2</td> </tr> <tr> <td>- No:</td> <td>Desactiva la creación registros de bitácora nivel 2</td> </tr> </table>	- Si:	Activa la creación de registros de bitácora nivel 2	- No:	Desactiva la creación registros de bitácora nivel 2
- Si:	Activa la creación de registros de bitácora nivel 2				
- No:	Desactiva la creación registros de bitácora nivel 2				
Id del programa	sis_tabla: views: vw_sis_tabla • Sis_TablaForm (forms.py) • templates: sis_tabla: form.html, list.html • js:administracion:sis_tabla:sis_tabla.js (static) • models: models.py				
Tablas relacionadas	sis.tabla				
Procedimientos almacenados					
Funciones					
Componentes	jQuery, Syncfusion, EncryptDES				
Pluggins					
Responsable	Jorge Peralta				
Estado	Terminado				

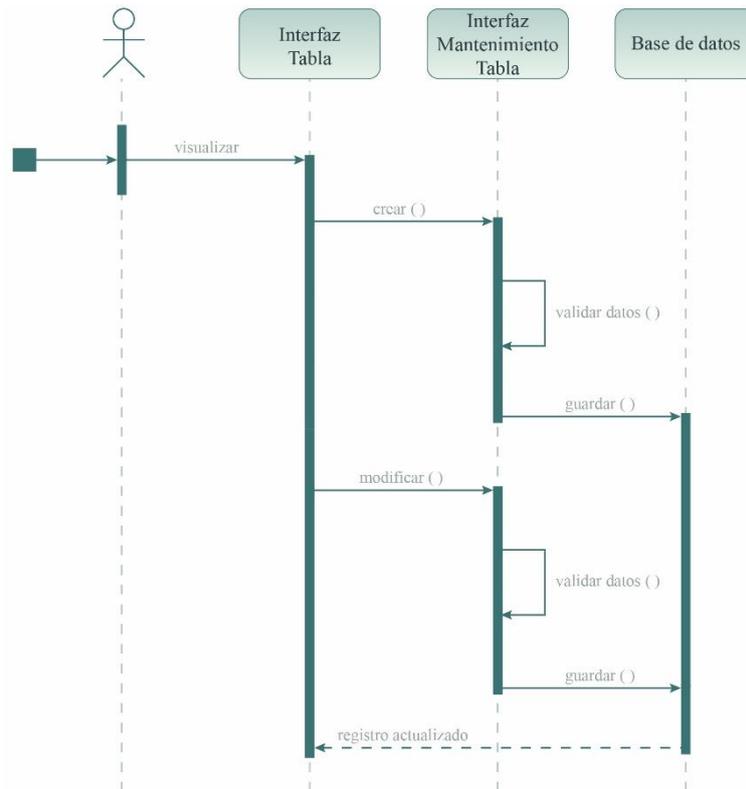
Diagrama de secuencia

El diagrama de secuencia, muestra cómo será la interacción entre el actor y los componentes, adicionalmente ilustra el comportamiento del sistema con sus respectivos mensajes y funcionalidades.

En este caso, el administrador del sistema tiene la capacidad de crear y editar registros en la tabla, generar nivel de auditoría, registrar datos y luego verificar la auditoría en la bitácora nivel 2. Entre sus funciones también se le permite consultar información en la tabla, como se puede evidenciar en la Figura 6.

Figura 6

Diagrama de secuencia de la gestión de la tabla.



Casos de prueba de la Iteración 2.

A continuación, se expone el caso de prueba 2 que permite crear y modificar los datos de la tabla para el proceso de bitácora nivel 2.

Tabla 9

Caso de prueba de la gestión de la tabla.

Prueba de usuario	
Identificador caso de prueba	CP-ADM-001-001
Caso de prueba	Gestionar el mantenimiento de la tabla
Función que se va a probar	Módulo: Sistema Código de requerimiento: RF-SIS-153 Caso de uso aplicado: HU-SIS-153 (Gestionar tipo de clasificación)
Autor	Jorge Peralta
Fecha de creación	2022-08-24
Objetivo de la prueba	Descripción: a) La aplicación permitirá agregar y modificar una tabla de la base de datos. b) La aplicación permitirá activar o desactivar la bitácora nivel 2. c) Constatar que los datos ingresados o modificados en cada campo correspondan al tipo de dato
Perfil del usuario	Usuario administrador con conocimientos avanzados en el sistema
Necesidades para el caso de prueba	Datos de prueba entregados por los desarrolladores
Criterios de éxito	La aplicación permite crear, modificar un registro de la tabla.
Criterios de falla	Al realizar la operación de Agregar/Modificar puede ser provocado por: • Campos incompletos (campos vacíos). • Campos incorrectos (error del tipo de dato). • Error al gestionar el registro en la base de datos. Como respuesta el sistema visualiza el error generado.
Precondiciones	Condiciones:

	<ul style="list-style-type: none"> • Ejecutar el sistema. • Usuario identificado en el sistema. • Usuario debe estar en la pantalla del reporte de Tabla
Responsable de la ejecución de la prueba	Santiago Cedillo
1. Agregar registro en la Tabla	
Secuencia normal	Datos: <ul style="list-style-type: none"> • Nombre: adm.rol • Alias: Rol • Llave primaria: rol_id • Columna estado: rol_estado • Es foránea: No • Nivel de auditoría: Si • En Contabilización: No
	Flujo: <ol style="list-style-type: none"> 1.1 El usuario presiona el botón agregar. 1.2 El sistema se dirige a la pantalla de mantenimiento. 1.3 El sistema visualiza los campos necesarios para crear el registro. 1.4 El usuario rellena los campos del mantenimiento. 1.5 El usuario presiona el botón guardar. 1.6 El sistema regresa a la opción del reporte y muestra un mensaje de datos guardados.
	Resultados esperados <ul style="list-style-type: none"> • El sistema regresa a la pantalla del reporte y visualiza el siguiente mensaje: “Datos guardados”. • El sistema crea y visualiza el registro en el reporte. • En la pantalla del reporte, a través de una búsqueda con la siguiente cadena “adm.rol”, en el campo buscar, se confirma que el registro ha sido creado.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) <ul style="list-style-type: none"> - El registro se crea exitosamente y la notificación es la correcta y esta permanece constante. - En la búsqueda el sistema encuentra exitosamente el registro creado.
Secuencia alternativa	Datos: <ul style="list-style-type: none"> • Nombre: adm.rol • Alias: Rol • Llave primaria: ----- • Columna estado: rol_estado • Es foránea: No • Nivel de auditoría: Si • En Contabilización: No
	Flujo: <ol style="list-style-type: none"> 1.5 El usuario presiona el botón guardar. Campos incompletos Campos incorrectos Error al crear el registro en la base de datos <ol style="list-style-type: none"> 1.5.1 El sistema alerta e indica el error en pantalla. 1.5.2 No se guardan los datos. 1.5.3 El usuario regresa al punto 1.4
	Resultados esperados: <p>El sistema muestra el siguiente aviso: Se han producido 3 errores.</p> <ul style="list-style-type: none"> • Campo obligatorio • Campo vacío • Valor incorrecto <p>El sistema indica los siguientes mensajes:</p> <ul style="list-style-type: none"> - Nombre: campo obligatorio - Alias: - Llave primaria: campo obligatorio - Columna estado: campo obligatorio - Es foránea: campo obligatorio - Nivel de auditoría: campo obligatorio - En Contabilización: campo obligatorio
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) <ul style="list-style-type: none"> - El registro no se crea y el sistema muestra el mensaje de “Correct errors before saving.” - El sistema pone de color rojo los campos incompletos y envía el mensaje “Completa este campo”.

2. Modificar registro de la Tabla	
Secuencia normal	Datos de Tabla: • Nombre: inv.transaccioncab • Alias: Transacción Cabecera • Llave primaria: cab_id • Columna estado: cab_estado • Es foránea: No • Nivel de auditoría: No • En Contabilización: No Datos de TablaCol: • Columna: cab_total • En Contabilización: Si
	Flujo: 2.1 El usuario puede buscar el registro deseado o pasar al punto 2.2 2.1.1 Ingresar los datos en el campo del buscador. 2.1.2 Presionar botón buscar. 2.2 El usuario presiona el enlace (icono lápiz) del registro identificado. 2.3 El sistema se dirige a la pantalla del mantenimiento. 2.4 El sistema carga y visualiza los datos del registro en los campos correspondientes. 2.5 El usuario modifica los campos deseados. 2.6 El usuario presiona el botón guardar. 2.7 El sistema regresa a la opción del reporte y muestra un mensaje de datos guardados.
	Resultados esperados: El sistema actualiza el registro y lo visualiza en el reporte.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El registro se actualiza correctamente y se visualiza en el reporte. - La notificación que envía el sistema es el correcto "Datos Guardados"
Secuencia alternativa	Datos de Tabla: • Nombre: Visible y sin opción a editar. • Alias: Campo vacío • Llave primaria: cab_id • Columna estado: ----- • Es foránea: No • Nivel de auditoría: No • En Contabilización: No Datos de TablaCol: • Columna: cab_total • En Contabilización: No
	Flujo: 2.6 El usuario presiona el botón guardar. a. Campos incompletos b. Campos incorrectos c. Error al crear el registro en la base de datos 2.6.1 El sistema alerta e indica el error en pantalla. 2.6.2 No se guardan los datos. 2.6.3 El usuario regresa al punto 2.4
	Resultados esperados: El registro no es modificado.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El registro no se modifica, ya que el sistema muestra que existe un error y que debe completar los campos en blanco.

Iteración 3.

154. Generar contenido bitácora nivel 2.

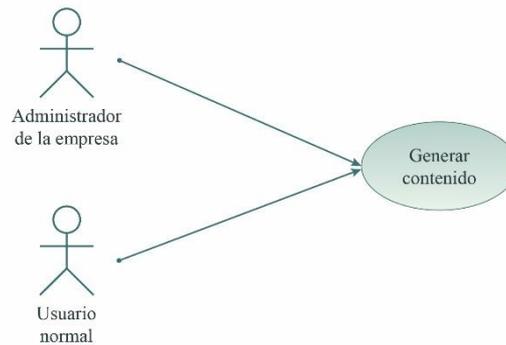
En la tercera iteración se genera información para la bitácora nivel 2, mediante acciones de agregación, modificación y eliminación de los campos de una tabla seleccionada que cumpla con los permisos correspondientes.

Diagrama de casos de uso.

En esta iteración, el administrador de la empresa y el usuario normal tienen la capacidad de generar contenido como se observa en la Figura 7.

Figura 7

Diagrama de casos de uso para generar contenido en la bitácora nivel 2.



Historias de usuario.

En la siguiente historia de usuario, se crea, modifica y elimina datos de las tablas seleccionadas para generar contenido de la bitácora nivel 2. Vale precisar que esta acción solo será posible cuando los modelos contengan ERPBaseModel.

Tabla 10

Historias de usuario para generar contenido en la bitácora nivel 2.

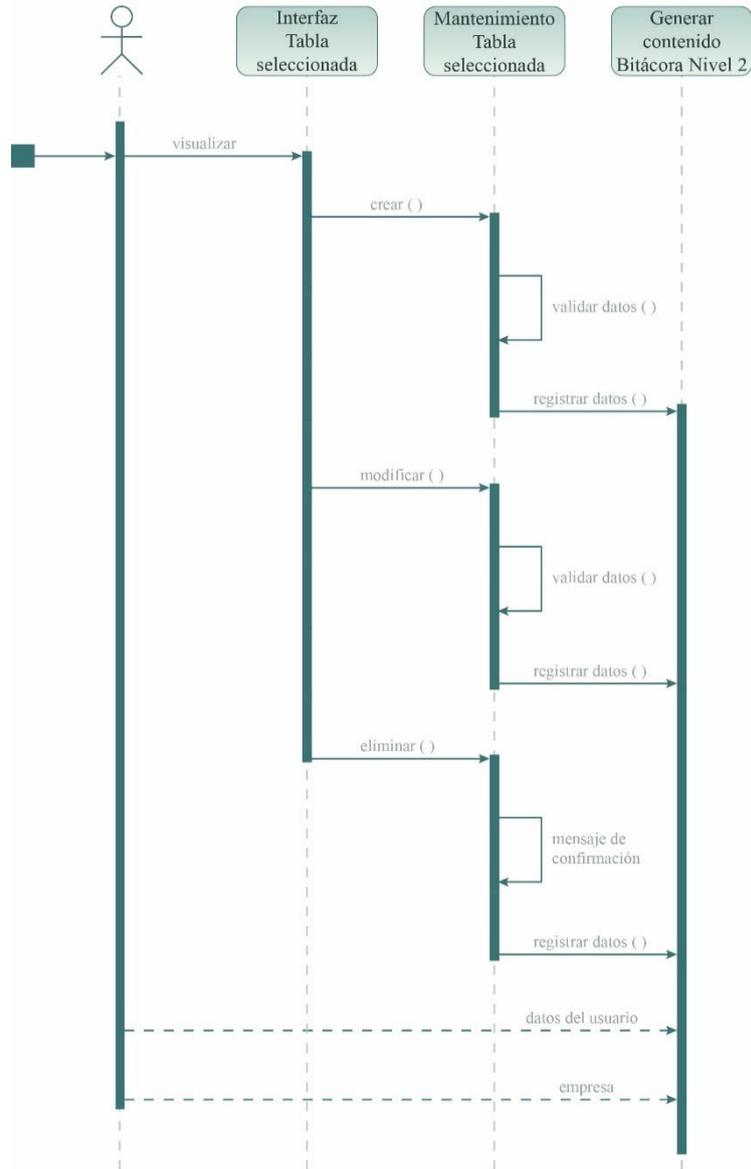
154	Generar contenido en la bitácora nivel 2
COMO	Administrador
QUIERO	Almacenar las actividades que se realiza durante la sesión activa de todos los usuarios
PARA	Tener información y poder generar revisiones
Estimación: (horas/días)	60 horas
Prioridad:	2
Criterios de aceptación	1. La aplicación no podrá generar contenido de bitácora nivel 2 si no se tiene los permisos necesarios en la tabla 2. La aplicación debe generar contenido cuando se crea, elimina, edita 3. La aplicación no permitirá la eliminación de un registro
Acción de la aplicación	
Id del programa	
Tablas relacionadas	tabla en la que el usuario se encuentre gestionando información
Procedimientos almacenados	
Funciones	callBinnacle(arg)
Componentes	
Pluggins	
Responsable	Jorge Peralta
Estado	Terminado

Diagrama de secuencia.

En este caso, el administrador de la empresa y el usuario normal únicamente puede acceder a la bitácora para generar contenido, se puede evidenciar en la Figura 8.

Figura 8

Diagrama de secuencia para generar contenido en la bitácora nivel 2.



Casos de prueba de la Iteración 3.

En las siguientes tablas se explica el caso de prueba 3, en donde se crea, modifica y elimina datos con el propósito de generar contenido.

Tabla 11

Caso de prueba para generar contenido en la bitácora nivel 2.

Prueba de usuario	
Identificador de prueba	CP-ADM-001-001
Caso de prueba	Gestionar el contenido de la bitácora nivel 2
Función que se va a probar	Módulo: Sistema Código de requerimiento: RF-SIS-154 Caso de uso aplicado: HU-SIS-154
Autor	Jorge Peralta
Fecha de creación	2022-08-24

Objetivo de la prueba	Descripción: a) La aplicación permitirá generar contenido, si contiene los permisos respectivos. b) Constatar que los datos ingresados, modificados y eliminados generen el contenido.
Perfil del usuario	Usuario administrador con conocimientos en el sistema
Necesidades para el caso de prueba	Datos de prueba entregado por el desarrollador
Criterios de éxito	La aplicación permite generar información en las tablas seleccionadas.
Criterios de falla	Al realizar la operación de Agregar/Modificar/Eliminar en las tablas: • No genera contenidos al crear. • No genera contenidos al editar. • No genera contenidos al eliminar. Como respuesta el sistema no genera registros en la tabla seleccionada.
Precondiciones	Condiciones: • Ejecutar el sistema. • Usuario identificado en el sistema • Usuario debe tener permisos de bitácora nivel 2 en el módulo de tabla • Usuario debe tener el modelo ERPBaseModel
Responsable de la ejecución de la prueba	Santiago Cedillo
1. Agregar registros en la tabla (rol)	
Secuencia normal	Datos: • Código: Automático, visible y sin opción a editar. • Nombre: Rol • Administrador: Si (Este campo aparece si tiene permisos)
	Flujo: 1.1 El usuario presiona el botón agregar. 1.2 El sistema se dirige a la pantalla de mantenimiento. 1.3 El sistema visualiza los campos necesarios para crear el registro. 1.4 El usuario rellena los campos del mantenimiento. 1.5 El usuario presiona el botón guardar. 1.6 El sistema regresa a la opción del reporte y muestra un mensaje de datos guardados.
	Resultados esperados • El sistema genera contenidos con cada campo creado en el reporte. • El usuario no sabe que se ha generado un registro al accionar el botón agregar en el módulo de bitácora nivel 1 y 2
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - Se genera varios registros con cada campo de la tabla creada.
Secuencia alternativa	Datos: • Código: Automático, visible y sin opción a editar. • Nombre: Rol • Administrador: Si (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	Flujo: 1.5 El usuario presiona el botón guardar. Campos incompletos Campos incorrectos Error al crear el registro en la base de datos 1.5.1 El sistema alerta e indica el error en pantalla. 1.5.2 No se guardan los datos. 1.5.3 El usuario regresa al punto 1.4
	Resultados esperados: El sistema muestra el siguiente aviso: Se han producido 3 errores. • Campo obligatorio • Campo vacío • Valor incorrecto El sistema indica los siguientes mensajes: • Nombre: Campo obligatorio • Administrador: (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2) El sistema no genera información de ese registro de la tabla.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - No genera registros con cada campo de la tabla creada.
2. Modificar registro en la tabla (rol)	
Secuencia normal	Datos: • Código: Automático, visible y sin opción a editar.

	<ul style="list-style-type: none"> • Nombre: Auditor • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>2.1 El usuario puede buscar el registro deseado o pasar al punto 2.2</p> <p>2.1.1 Ingresar los datos en el campo del buscador.</p> <p>2.1.2 Presionar botón buscar.</p> <p>2.2 El usuario presiona el enlace (icono lápiz) del registro identificado.</p> <p>2.3 El sistema se dirige a la pantalla del mantenimiento.</p> <p>2.4 El sistema carga y visualiza los datos del registro en los campos correspondientes.</p> <p>2.5 El usuario modifica los campos deseados.</p> <p>2.6 El usuario presiona el botón guardar.</p> <p>2.7 El sistema regresa a la opción del reporte y muestra un mensaje de datos guardados.</p>
	<p>Resultados esperados:</p> <p>El sistema genera registros con los campos modificados.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - El registro se actualiza correctamente y se visualiza en el reporte. - La notificación que envía el sistema es el correcto "Datos Guardados" - El sistema genera información correspondiente al o los campos modificados.
Secuencia alternativa	<p>Datos de Tabla:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: ---- • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>2.6 El usuario presiona el botón guardar.</p> <p>a. Campos incompletos</p> <p>b. Campos incorrectos</p> <p>c. Error al crear el registro en la base de datos</p> <p>2.6.1 El sistema alerta e indica el error en pantalla.</p> <p>2.6.2 No se guardan los datos.</p> <p>2.6.3 El usuario regresa al punto 2.4</p>
	<p>Resultados esperados:</p> <p>El registro no es modificado.</p> <p>El sistema no crea registros, hasta que se procese la modificación para la bitácora nivel 2.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - El registro no se modifica, ya que el sistema muestra que existe un error y que debe completar los campos en blanco, requeridos. - No crea registros con los campos modificados.
3. Eliminar registro (Rol)	
Secuencia normal	<p>Campos:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: Auditor • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>3.1 El usuario puede buscar el registro deseado o pasar al punto 3.2</p> <p>3.1.1 Ingresar los datos en el campo del buscador.</p> <p>3.1.2 Presionar botón buscar.</p> <p>3.2 El usuario presiona el enlace (icono lápiz) del registro identificado.</p> <p>3.3 El sistema se dirige a la pantalla del mantenimiento.</p> <p>3.4 El sistema carga y/o visualiza los datos del registro en los campos correspondientes.</p> <p>3.5 El usuario presiona el botón eliminar.</p> <p>3.6 El sistema emite un mensaje de confirmación de eliminar.</p> <p>3.7 El usuario presiona el botón aceptar para confirmar la operación.</p> <p>3.8 El sistema regresa a la opción del reporte y muestra un mensaje de datos eliminados.</p>
	<p>Resultados esperados:</p> <p>El sistema elimina lógicamente el registro.</p> <p>No lo visualiza en el reporte.</p> <p>El sistema genera registros con el registro eliminado.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - Da un mensaje de confirmación. - La notificación es la correcta y el registro sí se elimina del reporte. - Se crea información con el registro eliminado.
Secuencia alternativa	<p>Campos:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: Auditor • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)

	<p>Flujo: 3.5. Error al eliminar el registro en la base de datos. 3.5.1 El sistema alerta e indica el error en pantalla. 3.5.2 No se eliminan los datos. 3.5.3 El usuario regresa al punto 3.4</p>
	<p>Resultados esperados: El registro no es eliminado, el sistema alerta sobre el error. No se crea información, hasta que el registro sea eliminado.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El registro no se elimina. - No genera información con el registro.</p>

Iteración 4.

155. Almacenar bitácora nivel 2.

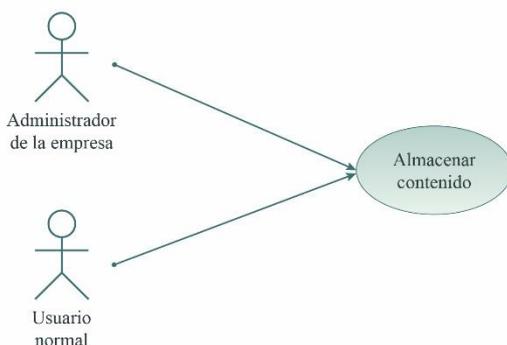
Almacena toda la información generada en la iteración anterior en el gestor de base de datos PostgreSQL. Con esta acción, se garantizan procesos de auditoría estables y seguros.

Diagrama de casos de uso.

Como se observa en la Figura 9, los datos e información generados por el administrador de la empresa y el usuario normal son almacenados en el sistema.

Figura 9

Diagrama de casos de uso para almacenar la bitácora nivel 2.



Historias de usuario.

Una vez generado el contenido en la iteración 3, se procede a guardar la información utilizando PostgreSQL.

Tabla 12

Historias de usuario para almacenar la bitácora nivel 2.

155	Almacenar bitácora nivel 2	
COMO	Administrador	
QUIERO	Generar un almacenamiento local cada seis meses de la bitácora nivel 2	
PARA	Liberar espacio de las tablas de la base de datos y respaldar la información	
Estimación: (horas/días)	20 horas	
Prioridad:	3	
Criterios de aceptación	1. La aplicación realizará un respaldo cada seis meses	
	2. La aplicación realizará el respaldo automáticamente en un archivo plano	
	3. La aplicación elimina la información almacenada en la tabla de bitácora nivel 2	
Acción de la aplicación		
Id del programa	NA	
Tablas relacionadas	Todas las tablas con auditoría nivel 2	

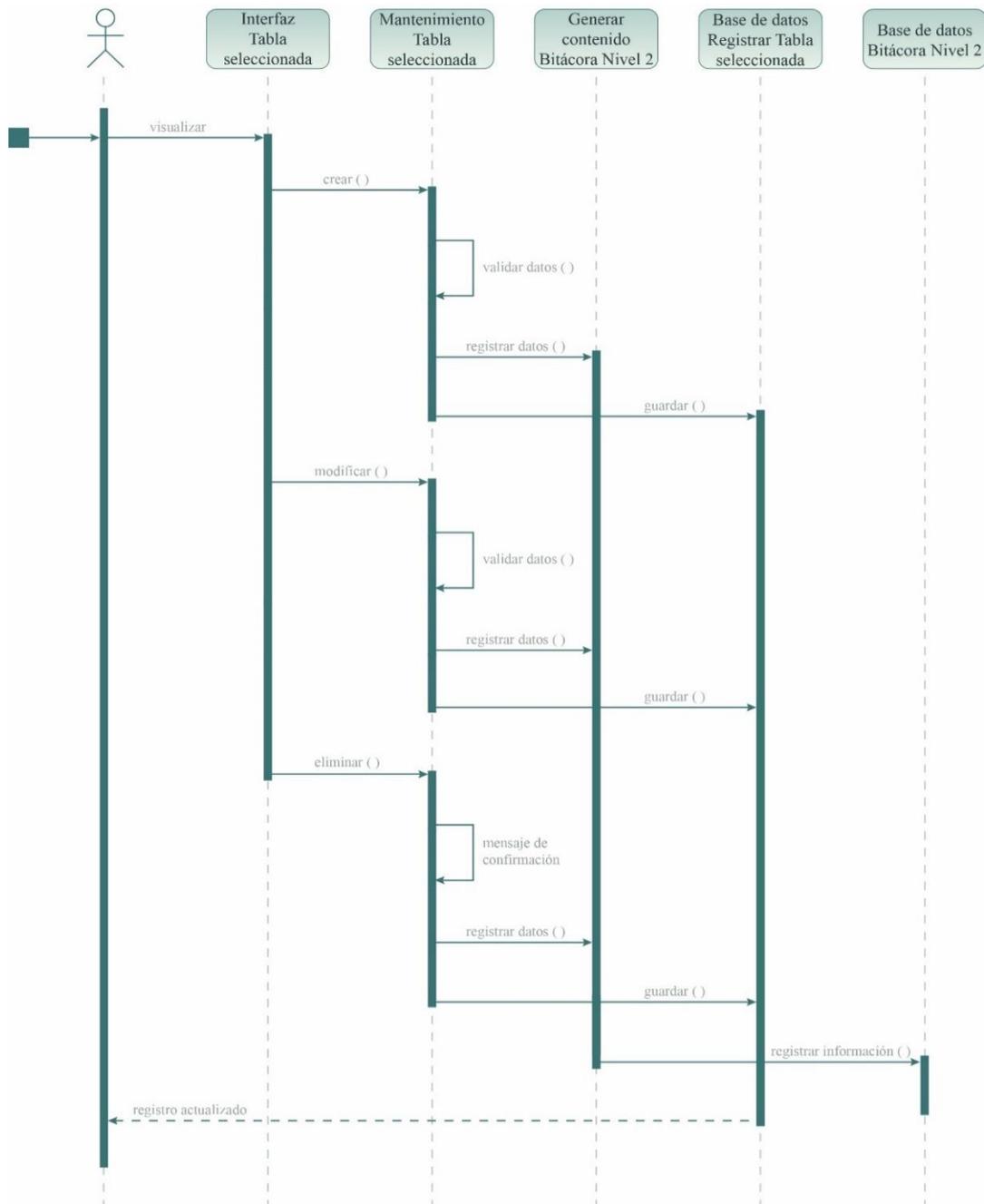
Procedimientos almacenados	adm.pr_respaldarauditoria()
Funciones	
Componentes	
Pluggins	schedule
Responsable	Jorge Peralta
Estado	Terminado

Diagrama de secuencia.

En la iteración 4, se almacena la información generada por el administrador de la empresa y el usuario normal, se evidencia en la Figura 10.

Figura 10

Diagrama de secuencia para almacenar la bitácora nivel 2.



Casos de prueba de la Iteración 4.

En esta sección, se detalla el caso de prueba 4 que almacena la información de cada una de las acciones realizadas en la iteración anterior.

Tabla 13

Caso de prueba para almacenar la bitácora nivel 2.

Prueba de usuario	
Identificador caso de prueba	CP-ADM-001-001
Caso de prueba	Almacenar el contenido de la bitácora nivel 2
Función que se va a probar	Módulo: Sistema Código de requerimiento: RF-SIS-155 Caso de uso aplicado: HU-SIS-155
Autor	Jorge Peralta
Fecha de creación	2022-08-24
Objetivo de la prueba	Descripción: a) La aplicación permitirá almacenar el contenido generado, si contiene los permisos respectivos. b) Constatar que los datos ingresados, modificados y eliminados puedan almacenar el contenido.
Perfil del usuario	Usuario administrador con conocimientos en el sistema
Necesidades para el caso de prueba	Datos de prueba entregado por el desarrollador
Criterios de éxito	La aplicación permite almacenar información en las tablas seleccionadas, en la tabla de bitácora nivel 2.
Criterios de falla	Al realizar la operación de Agregar/Modificar/Eliminar en las tablas: • No almacena registros al crear. • No almacena registros al editar. • No almacena registros al eliminar. Como respuesta el sistema no almacena registros en la tabla seleccionada.
Precondiciones	Condiciones: • Ejecutar el sistema. • Usuario identificado en el sistema • Usuario debe tener permisos de bitácora nivel 2 en el módulo de tabla • Usuario debe tener el modelo ERPBaseModel
Responsable de la ejecución de la prueba	Santiago Cedillo
1. Agregar registros en la tabla (rol)	
Secuencia normal	Datos: • Código: Automático, visible y sin opción a editar. • Nombre: Rol • Administrador: Si (Este campo aparece si tiene permisos)
	Flujo: 1.1 El usuario presiona el botón agregar. 1.2 El sistema se dirige a la pantalla de mantenimiento. 1.3 El sistema visualiza los campos necesarios para crear el registro. 1.4 El usuario rellena los campos del mantenimiento. 1.5 El usuario presiona el botón guardar. 1.6 El sistema regresa a la opción del reporte y muestra un mensaje de datos guardados.
	Resultados esperados • El sistema almacena los contenidos generados en los campos de la tabla seleccionada. • El usuario no sabe que se ha almacenado todos los campos de la tabla en dar clic al botón agregar. Esto ocurre en la bitácora nivel 1 y 2
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - Se almacena de manera correcta los registros con cada campo de la tabla creada.
Secuencia alternativa	Datos: • Código: Automático, visible y sin opción a editar. • Nombre: Rol • Administrador: Si (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)

	<p>Flujo:</p> <p>1.5 El usuario presiona el botón guardar.</p> <p>Campos incompletos</p> <p>Campos incorrectos</p> <p>Error al crear el registro en la base de datos</p> <p>1.5.1 El sistema alerta e indica el error en pantalla.</p> <p>1.5.2 No se guardan los datos.</p> <p>1.5.3 El usuario regresa al punto 1.4</p>
	<p>Resultados esperados:</p> <p>El sistema muestra el siguiente aviso:</p> <p>Se han producido 3 errores.</p> <ul style="list-style-type: none"> • Campo obligatorio • Campo vacío • Valor incorrecto <p>El sistema indica los siguientes mensajes:</p> <ul style="list-style-type: none"> • Nombre: Campo obligatorio • Administrador: (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2) <p>El sistema no almacena información de ese registro de la tabla.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <p>- No almacena registros de la tabla creada.</p>
2. Modificar registro en la tabla (rol)	
Secuencia normal	<p>Datos:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: Auditor • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>2.1 El usuario puede buscar el registro deseado o pasar al punto 2.2</p> <p>2.1.1 Ingresa los datos en el campo del buscador.</p> <p>2.1.2 Presiona botón buscar.</p> <p>2.2 El usuario presiona el enlace (icono lápiz) del registro identificado.</p> <p>2.3 El sistema se dirige a la pantalla del mantenimiento.</p> <p>2.4 El sistema carga y visualiza los datos del registro en los campos correspondientes.</p> <p>2.5 El usuario modifica los campos deseados.</p> <p>2.6 El usuario presiona el botón guardar.</p> <p>2.7 El sistema regresa a la opción del reporte y muestra un mensaje de datos guardados.</p>
	<p>Resultados esperados:</p> <p>El sistema almacena los registros con los campos modificados.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - El registro se actualiza correctamente y se visualiza en el reporte. - La notificación que envía el sistema es el correcto "Datos Guardados" - El sistema almacena información correspondiente al o los campos modificados.
Secuencia alternativa	<p>Datos de Tabla:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: ---- • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>2.6 El usuario presiona el botón guardar.</p> <p>a. Campos incompletos</p> <p>b. Campos incorrectos</p> <p>c. Error al crear el registro en la base de datos</p> <p>2.6.1 El sistema alerta e indica el error en pantalla.</p> <p>2.6.2 No se guardan los datos.</p> <p>2.6.3 El usuario regresa al punto 2.4</p>
	<p>Resultados esperados:</p> <p>El registro no es modificado.</p> <p>El sistema no almacena la información.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - El registro no se modifica, ya que el sistema muestra que existe un error y que debe completar los campos en blanco, requeridos. - La información no se almacena en la tabla de bitácora nivel 2.
3. Eliminar registro (Rol)	
Secuencia normal	<p>Campos:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar.

	<ul style="list-style-type: none"> • Nombre: Auditor • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>3.1 El usuario puede buscar el registro deseado o pasar al punto 3.2</p> <p>3.1.1 Ingresa los datos en el campo del buscador.</p> <p>3.1.2 Presiona botón buscar.</p> <p>3.2 El usuario presiona el enlace (icono lápiz) del registro identificado.</p> <p>3.3 El sistema se dirige a la pantalla del mantenimiento.</p> <p>3.4 El sistema carga y/o visualiza los datos del registro en los campos correspondientes.</p> <p>3.5 El usuario presiona el botón eliminar.</p> <p>3.6 El sistema emite un mensaje de confirmación de eliminar.</p> <p>3.7 El usuario presiona el botón aceptar para confirmar la operación.</p> <p>3.8 El sistema regresa a la opción del reporte y muestra un mensaje de datos eliminados.</p>
	<p>Resultados esperados:</p> <p>El sistema elimina lógicamente el registro.</p> <p>No lo visualiza en el reporte.</p> <p>El sistema almacena los registros y por defecto como valor nuevo deja en vacío.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - Da un mensaje de confirmación. - La notificación es la correcta y el registro sí se elimina del reporte. - Almacena la información de la bitácora nivel 2.
Secuencia alternativa	<p>Campos:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: Auditor • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>3.5. Error al eliminar el registro en la base de datos.</p> <p>3.5.1 El sistema alerta e indica el error en pantalla.</p> <p>3.5.2 No se eliminan los datos.</p> <p>3.5.3 El usuario regresa al punto 3.4</p>
	<p>Resultados esperados:</p> <p>El registro no es eliminado, el sistema alerta sobre el error.</p> <p>No se almacena información, hasta que el registro sea eliminado.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - El registro no se elimina. - No almacena información con el registro.

Iteración 5.

156. Visualizar bitácoras.

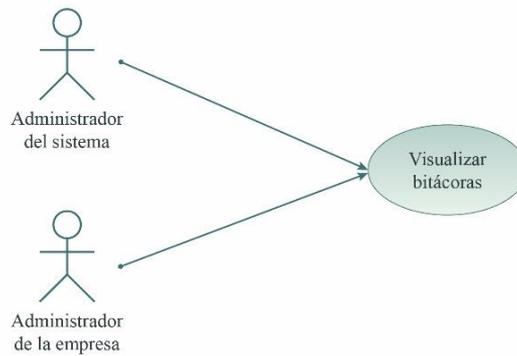
Aquí se visualiza la información de las bitácoras nivel 1 y 2. En este caso, la tabla tiene una columna extra con dos íconos que referencian cada bitácora. Para esta iteración es importante que el usuario que accede al sistema cuente con los permisos definidos.

Diagrama de casos de uso.

Para esta iteración, el administrador del sistema da permiso al administrador de la empresa para visualizar los datos de su MIPYNES. Éste último tiene la opción de dar permiso a uno o varios usuarios para la ejecución de procesos de auditoría o revisiones de datos del sistema. Así también, tiene la capacidad de cancelar los permisos cuando las actividades hayan sido concluidas, se expone en la Figura 11.

Figura 11

Diagrama de casos de uso para visualizar bitácoras.



Historias de usuario.

Se modifica la tabla de rol para generar los permisos de visualización en las bitácoras nivel 1 y 2. Todos los usuarios con acceso a esta función tienen una columna extra en la tabla con iconografía que distingue a cada nivel de auditoría. Es necesario precisar que al momento de seleccionar cualquiera de las imágenes se muestra información delicada de la empresa, por lo que es importante dar los permisos de acceso con responsabilidad ética.

Tabla 14

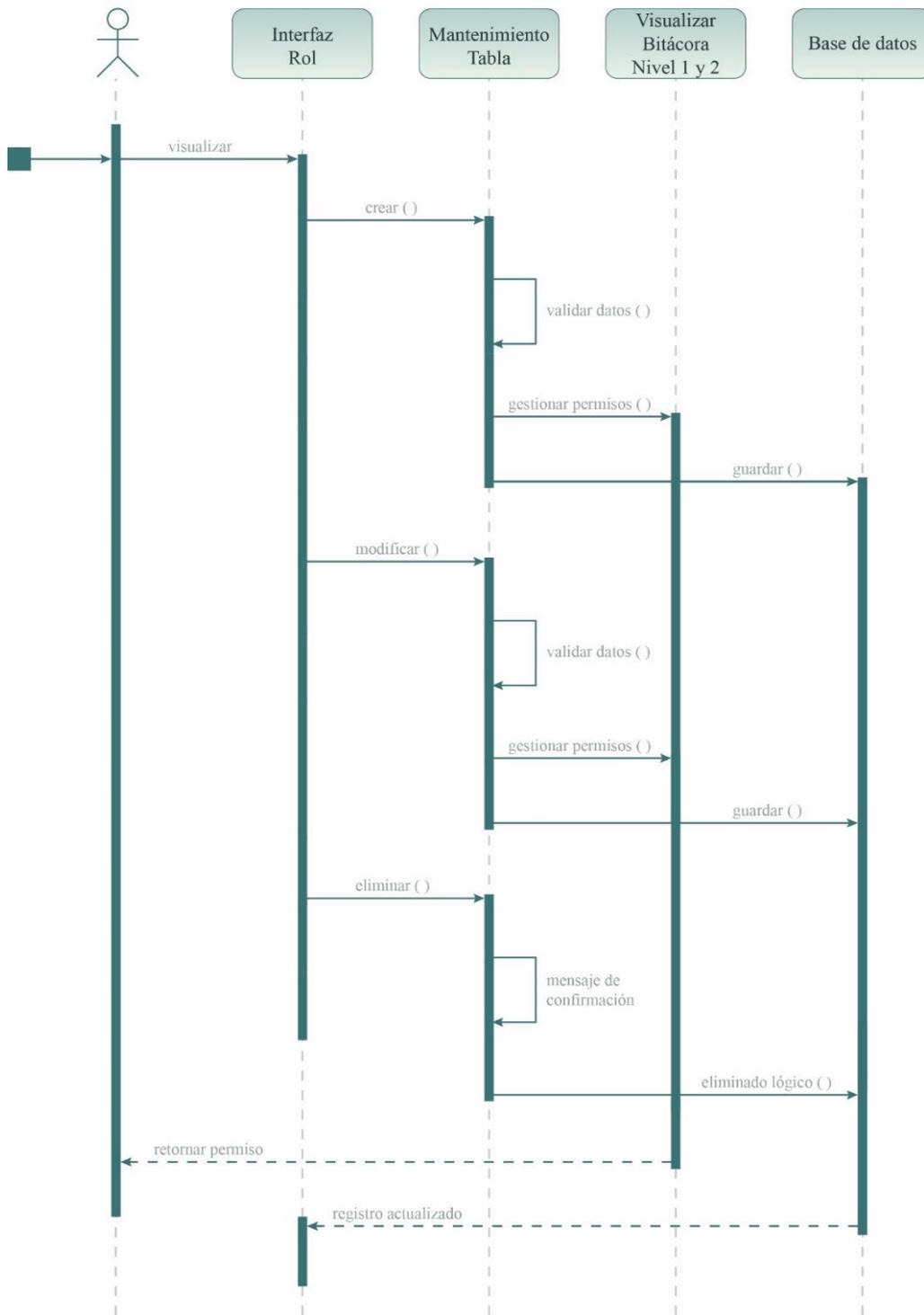
Historias de usuario para visualizar bitácoras.

156	Visualizar bitácoras
COMO	Administrador
QUIERO	Visualizar el historial de la bitácora nivel 1 y 2 en las tablas
PARA	Revisión de información
Estimación: (horas/días)	96 horas
Prioridad:	4
Criterios de aceptación	1. La aplicación podrá visualizar la bitácora nivel 1 y 2 si este tiene los permisos respectivos 2. La aplicación tendrá dos íconos para diferenciar la bitácora nivel 1 con respecto a la bitácora nivel 2 3. La aplicación podrá cambiar los permisos si solo es un usuario con privilegios 4. La aplicación podrá visualizar automáticamente tablas con información de cambios realizados en el sistema
Acción de la aplicación	
Id del programa	
Tablas relacionadas	tabla seleccionada
Procedimientos almacenados	
Funciones	
Componentes	Ajax
Pluggins	
Responsable	Jorge Peralta
Estado	Terminado

Diagrama de secuencia.

En esta Figura 12, se representa las posibilidades de acción que tienen los usuarios involucrados para la visualización de la información en las bitácoras nivel 1 y 2.

Figura 12
Diagrama de secuencia para visualizar bitácoras.



Casos de prueba de la Iteración 5.

Se detalla el caso de prueba 5 que visualizan la información de cada una de las acciones realizadas en la iteración anterior.

Tabla 15

Caso de prueba para visualizar bitácoras.

Prueba de usuario	
Identificador caso de prueba	CP-ADM-001-001
Caso de prueba	Visualizar bitácora tanto como el nivel 1 y 2
Función que se va a probar	Módulo: Sistema Código de requerimiento: RF-SIS-156 Caso de uso aplicado: HU-SIS-156
Autor	Jorge Peralta
Fecha de creación	2022-08-24
Objetivo de la prueba	Descripción: a) La aplicación permitirá visualizar el contenido generado en la prueba anterior, si contiene los permisos respectivos. b) Constatar que los datos ingresados, modificados y eliminados puedan mostrarse o quitarse los permisos de visualización. c) Contratar si una persona tiene los permisos necesarios, pueda dar generar la autorización de visualización de la bitácora nivel 1 y 2 a otra persona.
Perfil del usuario	Usuario administrador con conocimientos en el sistema
Necesidades para el caso de prueba	Datos de prueba entregado por el desarrollador
Criterios de éxito	La aplicación permite realizar revisiones del sistema dentro de la tabla seleccionada.
Criterios de falla	Al realizar la operación de Agregar/Modificar/Eliminar en las tablas: • No permite gestionar los permisos de visualización de bitácora nivel 1 y 2 al momento de crear. • No permite gestionar los permisos de visualización de bitácora nivel 1 y 2 al momento de modificar. • No permite eliminar el permiso de visualización. • No permite visualizar con los permisos. Como respuesta el sistema no visualiza las columnas de bitácora nivel 1 y 2
Precondiciones	Condiciones: • Ejecutar el sistema. • Usuario identificado en el sistema • Usuario debe tener permisos al mantenimiento de rol.
Responsable de la ejecución de la prueba	Santiago Cedillo
1. Agregar registros en la tabla (rol)	
Secuencia normal	Datos: • Código: Automático, visible y sin opción a editar. • Nombre: Auditar • Administrador: Si (Este campo aparece si tiene permisos) (Permiso de visualización)
	Flujo: 1.1 El usuario presiona el botón agregar. 1.2 El sistema se dirige a la pantalla de mantenimiento. 1.3 El sistema visualiza los campos necesarios para crear el registro. 1.4 El usuario rellena los campos del mantenimiento. 1.5 El usuario presiona el botón guardar. 1.6 El sistema regresa a la opción del reporte y muestra un mensaje de datos guardados.
	Resultados esperados • El sistema almacena los permisos, y el nombre del rol. • El usuario puede visualizar la bitácora nivel 1 y 2 en todas las tablas que tiene acceso.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - Se almacena de manera correcta el permiso de visualización, así como el nombre del rol creado.
Secuencia alternativa	Datos: • Código: Automático, visible y sin opción a editar. • Nombre: ----- • Administrador: Si (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	Flujo: 1.5 El usuario presiona el botón guardar.

	<p>Campos incompletos Campos incorrectos Error al crear el registro en la base de datos 1.5.1 El sistema alerta e indica el error en pantalla. 1.5.2 No se guardan los datos. 1.5.3 El usuario regresa al punto 1.4</p> <p>Resultados esperados: El sistema muestra el siguiente aviso: Se han producido 3 errores.</p> <ul style="list-style-type: none"> • Campo obligatorio • Campo vacío • Valor incorrecto <p>El sistema indica los siguientes mensajes:</p> <ul style="list-style-type: none"> • Nombre: Campo obligatorio • Administrador: (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2) <p>El sistema genera el rol con el permiso de visualización.</p> <p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - No genera el rol con los permisos de visualización.</p>
2. Modificar registro en la tabla (rol)	
Secuencia normal	<p>Datos:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: Auditor • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2) <p>Flujo:</p> <p>2.1 El usuario puede buscar el registro deseado o pasar al punto 2.2 2.1.1 Ingresa los datos en el campo del buscador. 2.1.2 Presiona botón buscar. 2.2 El usuario presiona el enlace (icono lápiz) del registro identificado. 2.3 El sistema se dirige a la pantalla del mantenimiento. 2.4 El sistema carga y visualiza los datos del registro en los campos correspondientes. 2.5 El usuario modifica los campos deseados. 2.6 El usuario presiona el botón guardar. 2.7 El sistema regresa a la opción del reporte y muestra un mensaje de datos guardados.</p> <p>Resultados esperados: El sistema almacena los registros con los campos modificados, pero en este caso el usuario no va a tener permisos de visualización de las bitácoras nivel 1 y 2.</p> <p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El registro se actualiza correctamente y se visualiza en el reporte. - La notificación que envía el sistema es el correcto "Datos Guardados" - El sistema permite visualizar la bitácora nivel 1 y 2 dentro del sistema.</p>
Secuencia alternativa	<p>Datos de Tabla:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: ---- • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2) <p>Flujo:</p> <p>2.6 El usuario presiona el botón guardar. a. Campos incompletos b. Campos incorrectos c. Error al crear el registro en la base de datos 2.6.1 El sistema alerta e indica el error en pantalla. 2.6.2 No se guardan los datos. 2.6.3 El usuario regresa al punto 2.4</p> <p>Resultados esperados: El registro no es modificado. El sistema visualiza la bitácora nivel 2, y se quedan los cambios anteriores.</p> <p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El registro no se modifica, ya que el sistema muestra que existe un error y que debe completar los campos en blanco, requeridos. - No se visualiza la bitácora nivel 1 y 2</p>
3. Eliminar registro (Rol)	
Secuencia normal	<p>Campos:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: Auditor

	<ul style="list-style-type: none"> • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>3.1 El usuario puede buscar el registro deseado o pasar al punto 3.2</p> <p>3.1.1 Ingresar los datos en el campo del buscador.</p> <p>3.1.2 Presionar botón buscar.</p> <p>3.2 El usuario presiona el enlace (icono lápiz) del registro identificado.</p> <p>3.3 El sistema se dirige a la pantalla del mantenimiento.</p> <p>3.4 El sistema carga y/o visualiza los datos del registro en los campos correspondientes.</p> <p>3.5 El usuario presiona el botón eliminar.</p> <p>3.6 El sistema emite un mensaje de confirmación de eliminar.</p> <p>3.7 El usuario presiona el botón aceptar para confirmar la operación.</p> <p>3.8 El sistema regresa a la opción del reporte y muestra un mensaje de datos eliminados.</p>
	<p>Resultados esperados:</p> <p>El sistema elimina lógicamente el registro.</p> <p>No lo visualiza en el reporte.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - Da un mensaje de confirmación. - La notificación es la correcta y el registro sí se elimina del reporte. - Se elimina con el rol con su respectivo permiso.
Secuencia alternativa	<p>Campos:</p> <ul style="list-style-type: none"> • Código: Automático, visible y sin opción a editar. • Nombre: Auditor • Administrador: No (Este campo aparece si tiene permisos a visualizar la bitácora nivel 2)
	<p>Flujo:</p> <p>3.5. Error al eliminar el registro en la base de datos.</p> <p>3.5.1 El sistema alerta e indica el error en pantalla.</p> <p>3.5.2 No se eliminan los datos.</p> <p>3.5.3 El usuario regresa al punto 3.4</p>
	<p>Resultados esperados:</p> <p>El registro no es eliminado, el sistema alerta sobre el error.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - El registro no se elimina. - El rol mantiene el permiso.
4. Visualización Bitácora Nivel 1 y 2 (Tabla Seleccionada)	
Secuencia normal	<p>Campos:</p> <ul style="list-style-type: none"> • Columna Bitácora Nivel 1: Botón con imagen. • Columna Bitácora Nivel 2: Botón con imagen.
	<p>Flujo:</p> <p>4.1 El usuario presiona el botón de Bitácora Nivel 1 o Bitácora Nivel 2 en el registro.</p> <p>4.2 El sistema muestra una pantalla emergente.</p> <p>4.3 El sistema visualiza los campos de la Bitácora Nivel 1 o Nivel 2 en base al registro seleccionado.</p> <p>4.4 El usuario presiona el botón de cerrar.</p> <p>4.5 El sistema cierra la pantalla emergente y se localiza el usuario en el reporte.</p>
	<p>Resultados esperados:</p> <p>El usuario puede visualizar la bitácora nivel 1 y nivel 2.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - El usuario puede visualizar de manera correcta la bitácora nivel 1 y bitácora nivel 2.
Secuencia alternativa	<p>Campos:</p> <ul style="list-style-type: none"> • Columna Bitácora Nivel 1: Botón con imagen. • Columna Bitácora Nivel 2: Botón con imagen.
	<p>Flujo:</p> <p>4.2. El sistema no permite mostrar la pantalla emergente.</p> <p>4.2.1 El sistema no visualiza los datos de la bitácora nivel 1 y nivel 2.</p>
	<p>Resultados esperados:</p> <p>El registro no visualiza los datos de la pantalla emergente.</p>
	<p>Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso)</p> <ul style="list-style-type: none"> - El sistema no muestra la pantalla emergente.

Iteración 6.

157. Gestionar bitácora nivel 1.

En esta iteración, el administrador del sistema tiene la capacidad de gestionar la visualización de los datos de las tablas de todas las empresas que pertenecen al proyecto UDA-ERP versión 2.0.

Diagrama de casos de uso.

La Figura 13 expone que únicamente el administrador del sistema puede participar en esta iteración, al ser el gestor del proyecto. De esta manera, se garantiza una correcta gestión de los datos y es posible solucionar inconvenientes con mayor agilidad.

Figura 13

Diagrama de casos de uso para gestionar bitácora nivel 1.



Historias de usuario.

En este caso, el administrador no puede modificar, crear o eliminar algún registro almacenado en la bitácora nivel 1. Su función se limita a la visualización de datos en caso que alguna empresa lo requiera. Para facilitar estas tareas, se creó la opción de filtrado ascendente y descendente en cada uno de los encabezados de las columnas, y se añadió un icono que permite buscar registros.

Tabla 16

Historias de usuario para gestionar bitácora nivel 1.

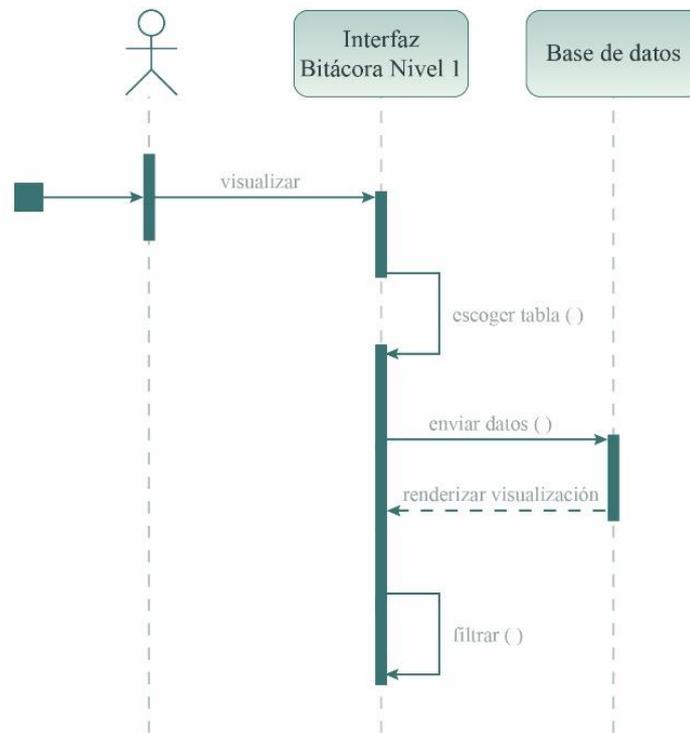
157	Gestionar bitácora nivel 1
COMO	Administrador
QUIERO	Almacenar el historial de la bitácora nivel 1 de todas las tablas
PARA	Revisión posterior del estado de los registros existentes en la base de datos
Estimación: (horas/días)	48 horas
Prioridad:	5
Criterios de aceptación	1. La aplicación podrá seleccionar que tabla desea visualizar 2. La aplicación podrá realizar una búsqueda dentro de la tabla seleccionada 3. La aplicación podrá realizar un filtrado ascendente y descendente
Acción de la aplicación	
Id del programa	adm_auditoriaN1: views: vw_adm_audN1 • templates: adm_auditoriaN1: list.html
Tablas relacionadas	tabla seleccionada
Procedimientos almacenados	
Funciones	
Componentes	Ajax, Syncfusion
Pluggins	
Responsable	Jorge Peralta
Estado	Terminado

Diagrama de secuencia.

En la Figura 14, representa las posibilidades de acción que tienen los usuarios involucrados para la visualización de la información en la bitácora nivel 1.

Figura 14

Diagrama de secuencia para gestionar bitácora nivel 1.



Casos de prueba de la Iteración 6.

En las siguientes tablas, se especifica el caso de prueba 6 que facilita visualizar la información de la bitácora nivel 1.

Tabla 17

Caso de prueba para gestionar bitácora nivel 1.

Prueba de usuario	
Identificador caso de prueba	CP-ADM-001-001
Caso de prueba	Visualizar bitácora nivel 1
Función que se va a probar	Módulo: Sistema Código de requerimiento: RF-SIS-157 Caso de uso aplicado: HU-SIS-157
Autor	Jorge Peralta
Fecha de creación	2022-08-24
Objetivo de la prueba	Descripción: a) La aplicación permitirá visualizar el contenido de todo el sistema por los administradores del proyecto UDA-ERP versión 2.0
Perfil del usuario	Usuario administrador con conocimientos en el sistema
Necesidades para el caso de prueba	Datos de prueba entregado por el desarrollador
Criterios de éxito	La aplicación permite realizar revisiones del sistema dentro de la tabla seleccionada.
Criterios de falla	Al realizar la operación de selección en las tablas: • No permite visualizar ninguna tabla del sistema.
Precondiciones	Condiciones: • Ejecutar el sistema. • Usuario identificado en el sistema • Usuario debe tener permisos de sistema.
Responsable de la ejecución de la prueba	Santiago Cedillo

1. Visualización Bitácora Nivel 1	
Secuencia normal	Campos: <ul style="list-style-type: none"> • ID: Muestra el identificador del registro de la tabla seleccionada. • Elemento: Código V segundo identificador de la tabla seleccionada. • Fecha de Creación: Fecha en la que se crea el registro. • Usuario Creador: Persona quién creó el registro. • Modificado el: Fecha en la que se realiza alguna modificación del registro. • Usuario Modificador: El último usuario que modifica el registro. • Última acción: Acción que realiza en el sistema entre ellos se encuentra, creación, modificación y eliminación. • Empresa: El nombre de la empresa a la que pertenece ese registro.
	Flujo: 1.1 El usuario selecciona la tabla que desea revisar. 1.2 El sistema muestra en una tabla los campos de bitácora nivel 1 de la tabla seleccionada. 1.3 El sistema genera un filtrado con los datos.
	Resultados esperados: El usuario puede visualizar la bitácora nivel 1 de todas las empresas.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El usuario puede visualizar de manera correcta la bitácora nivel 1 y puede generar un filtrado para mejorar su revisión
Secuencia alternativa	Campos: <ul style="list-style-type: none"> • ID: Muestra el identificador del registro de la tabla seleccionada. • Elemento: Código V segundo identificador de la tabla seleccionada. • Fecha de Creación: Fecha en la que se crea el registro. • Usuario Creador: Persona quién creó el registro. • Modificado el: Fecha en la que se realiza alguna modificación del registro. • Usuario Modificador: El último usuario que modifica el registro. • Última acción: Acción que realiza en el sistema entre ellos se encuentra, creación, modificación y eliminación. • Empresa: El nombre de la empresa a la que pertenece ese registro.
	Flujo: 1.2. El sistema no muestra los datos de la tabla seleccionada. 1.2.1 El sistema no visualiza información de esa tabla.
	Resultados esperados: El sistema no visualiza los datos.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El sistema no muestra la tabla.

Iteración 7.

158. Gestionar bitácora nivel 2.

Al igual que la iteración anterior, en este caso script el administrador del sistema visualiza los datos de todas las MIPYNES inscritas en el proyecto.

Diagrama de casos de uso.

El administrador del sistema, como ente gestor y consultor de la propuesta tiene la capacidad de visualizar la información almacenada por cada empresa. Esto asegura una buena administración de la información y un constante monitoreo del correcto funcionamiento del sistema, se evidencia en la Figura 15.

Figura 15

Diagrama de casos de uso para gestionar bitácora nivel 2.



Historias de usuario.

El administrador no puede editar los datos del sistema y su actividad se limita a la visualización de dicha información. Se ha creado dos opciones de filtrado: ascendente y descendente, y por columna. Asimismo, se implementó la opción de búsqueda para facilitar el manejo de los registros de cada tabla.

Tabla 18

Historias de usuario para gestionar bitácora nivel 2.

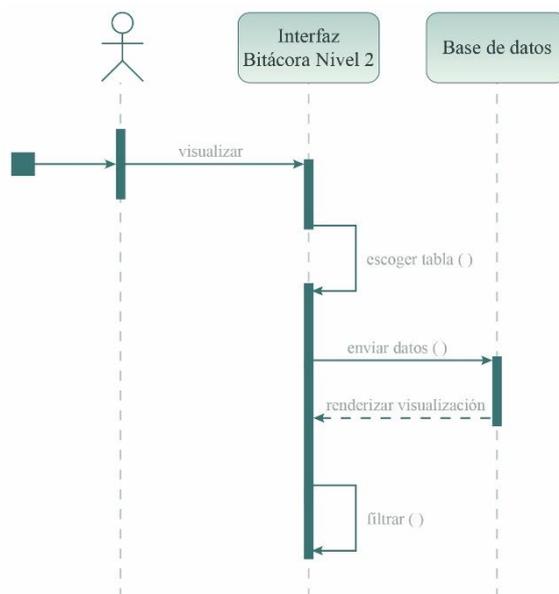
158	Gestionar bitácora nivel 2
COMO	Administrador
QUIERO	Generar una lista con los datos extraídos de la bitácora nivel 2
PARA	Revisión detallada de cambios en la información
Estimación: (horas/días)	56 horas
Prioridad:	6
Criterios de aceptación	1. La aplicación podrá una caja de selección para elegir la tabla a visualizar
	2. La aplicación podrá realizar una búsqueda dentro de la tabla
	3. La aplicación podrá realizar un filtrado por cada columna
	4. La aplicación podrá realizar un filtrado ascendente y descendente
Acción de la aplicación	
Id del programa	adm auditoriaN2: views: vw_adm_audN2 • templates: adm_auditoriaN2: list.html
Tablas relacionadas	Tabla seleccionada
Procedimientos almacenados	
Funciones	
Componentes	Ajax, Syncfusion
Pluggins	
Responsable	Jorge Peralta
Estado	Terminado
Estado	Terminado

Diagrama de secuencia.

En la Figura 16, representa las posibilidades de acción que tienen los usuarios involucrados para la visualización de la información en la bitácora nivel 2.

Figura 16

Diagrama de secuencia para gestionar bitácora nivel 2.



Casos de prueba de la Iteración 7.

Finalmente se representa el caso de prueba 7, que permite gestionar los datos de la bitácora nivel 2.

Tabla 19

Caso de prueba para gestionar bitácora nivel 2.

Prueba de usuario	
Identificador caso de prueba	CP-ADM-001-001
Caso de prueba	Visualizar bitácora nivel 2
Función que se va a probar	Módulo: Sistema Código de requerimiento: RF-SIS-158 Caso de uso aplicado: HU-SIS-158
Autor	Jorge Peralta
Fecha de creación	2022-08-24
Objetivo de la prueba	Descripción: a) La aplicación permitirá visualizar el contenido de todo el sistema por los administradores del proyecto UDA-ERP versión 2.0
Perfil del usuario	Usuario administrador con conocimientos en el sistema
Necesidades para el caso de prueba	Datos de prueba entregado por el desarrollador
Criterios de éxito	La aplicación permite realizar revisiones del sistema dentro de la tabla seleccionada.
Criterios de falla	Al realizar la operación de selección en las tablas: • No permite visualizar ninguna tabla del sistema.
Precondiciones	Condiciones: • Ejecutar el sistema. • Usuario identificado en el sistema • Usuario debe tener permisos de sistema.
Responsable de la ejecución de la prueba	Santiago Cedillo
1. Visualización Bitácora Nivel 2	
Secuencia normal	Campos: • Columna: Muestra en que atributo se realiza la operación. • Fecha: Fecha en la que se realiza alguna acción por parte del usuario. • Operación: Acción que realiza en el sistema entre ellos se encuentra, creación, modificación y eliminación. • Valor Anterior: Dato anterior de una operación. • Valor Nuevo: Dato nuevo de una operación. • Empresa: El nombre de la empresa a la que pertenece ese registro. • Usuario: El nombre del usuario que realiza una operación.
	Flujo: 1.1 El usuario selecciona la tabla que desea revisar. 1.2 El sistema muestra en una tabla los campos de bitácora nivel 2 de la tabla seleccionada. 1.3 El sistema genera un filtrado con los datos.
	Resultados esperados: El usuario puede visualizar la bitácora nivel 2 de todas las empresas.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El usuario puede visualizar de manera correcta la bitácora nivel 2 y puede generar un filtrado para mejorar su revisión
Secuencia alternativa	Campos: • Columna: Muestra en que atributo se realiza la operación. • Fecha: Fecha en la que se realiza alguna acción por parte del usuario. • Operación: Acción que realiza en el sistema entre ellos se encuentra, creación, modificación y eliminación. • Valor Anterior: Dato anterior de una operación. • Valor Nuevo: Dato nuevo de una operación. • Empresa: El nombre de la empresa a la que pertenece ese registro. • Usuario: El nombre del usuario que realiza una operación.
	Flujo: 1.2. El sistema no muestra los datos de la tabla seleccionada. 1.2.1 El sistema no visualiza información de esa tabla.

	Resultados esperados: El sistema no visualiza los datos. El sistema manda un mensaje del error.
	Resultados obtenidos (hallazgos o recomendaciones identificadas en el proceso) - El sistema no muestra la tabla. - El sistema muestra un mensaje que no existen datos a visualizar.

3.5 Entrenamiento.

Esta es una etapa incremental, se procederá como primer punto a capacitar a los desarrolladores del área, para que tengan una visión general de las herramientas que pueden ser utilizadas. Este paso es fundamental para garantizar que las posibles soluciones al problema respondan a las necesidades del usuario.

3.6 Validación inicial.

El profesional procederá a realizar evaluaciones en un entorno de laboratorio para analizar si las soluciones preliminares contribuyen a solucionar el problema.

3.7 Liberación de la solución.

Una vez encontrada la mejor ruta, se procederá a implementar en el software UDA-ERP versión 2.0

4. Resultados

En este apartado se muestran todos los resultados obtenidos durante la elaboración del módulo de bitácora nivel 2 del software UDA-ERP versión 2.0. En esta sección se encuentra los wireframes es decir los diseños a blanco y negro; diseño visual como se muestra el color, imágenes, texto, entre otros; software una explicación del módulo creado; entrenamiento el cual muestra el manual de usuario.

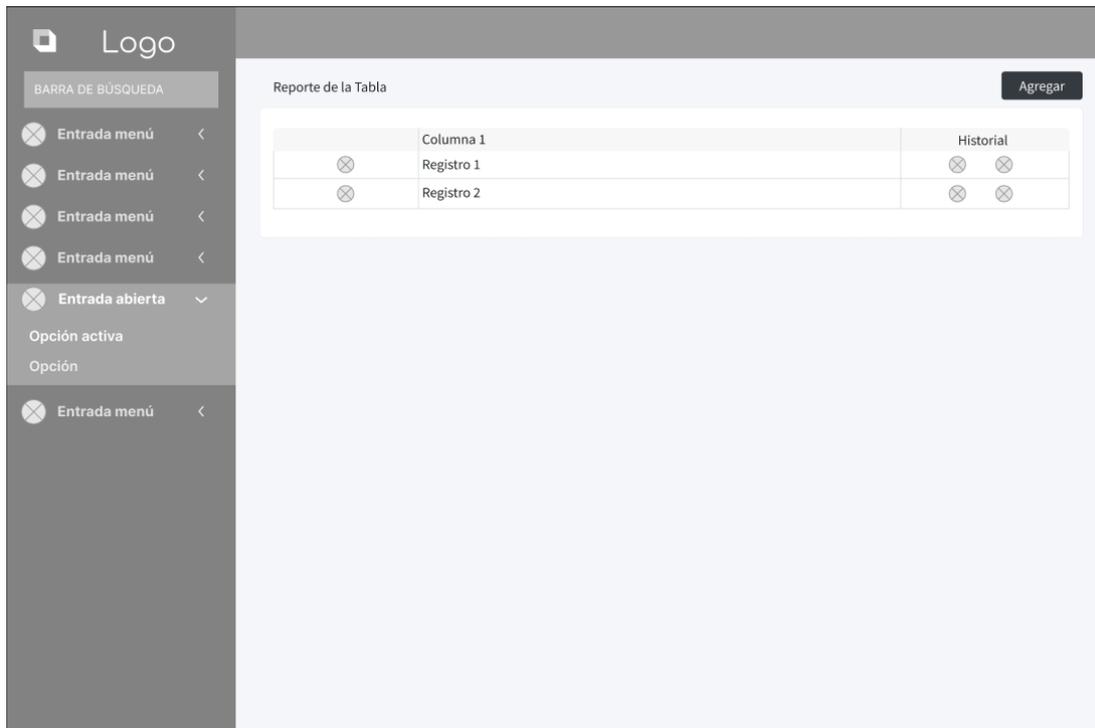
4.1 Wireframe

En las primeras etapas de desarrollo de sitios o plataformas web, los wireframes son una herramienta importante para describir la representación del sistema, facilitando su estructuración y visualización de los objetos; es decir, se encarga de diseñar todos los componentes que van a interactuar con el usuario final (Romensky y Rotkov 2020). En este caso se utiliza para mostrar cómo está estructurado el módulo de bitácora en una escala de grises. Se inicia con la definición del contenido en función de 3 interrogantes clave: ¿qué debo presentar? ¿cómo y dónde? ¿por qué? Cada una de estas preguntas contribuyen en la definición de las pestañas y los ítems apropiados para el funcionamiento de la bitácora y facilita la creación de un esqueleto visual.

Para esta representación se utiliza el Software Libre Figma Asnal et al. (2022) el cual permite realizar prototipos de manera vectorial. Desde la figura 17 a la figura 23 se expone la representación gráfica del módulo de sistema que contiene la bitácora nivel 1, bitácora de nivel 2, reportes de las bitácoras y el mantenimiento de la tabla al momento de realizar la gestión de la bitácora nivel 2. Estos wireframe se fundamenta en las jerarquías y esquemas predeterminados en el proyecto UDA-ERP versión 2.0.

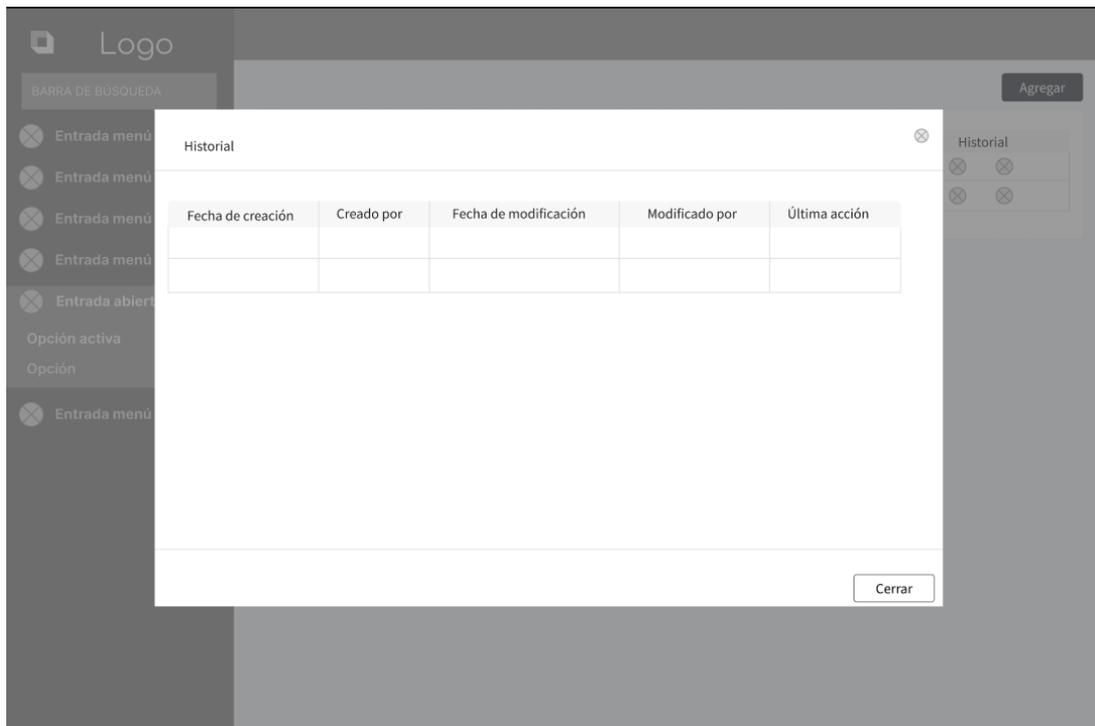
En la Figura 17 se muestra el diseño de las bitácoras tanto nivel 1 como nivel 2, en cada registro de una tabla seleccionada.

Figura 17
Bitácoras en las tablas.



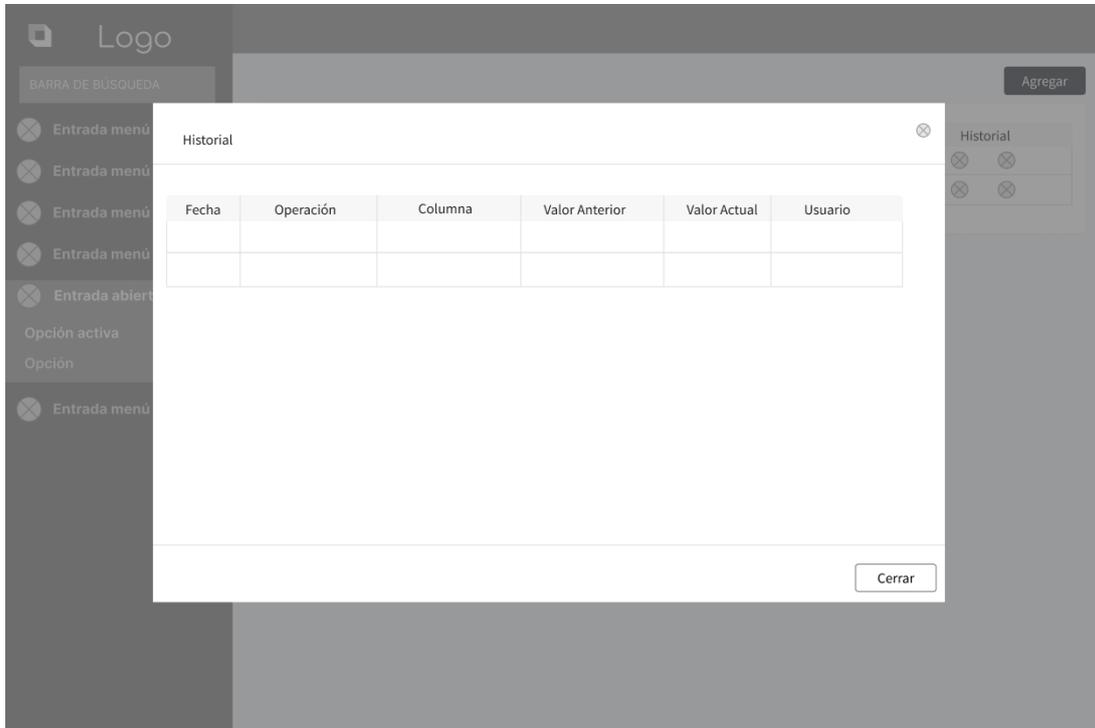
En la Figura 18 se grafica una ventana emergente de la bitácora nivel 1, para los usuarios que tengan permiso de visualización.

Figura 18
Seleccionar la bitácora nivel 1.



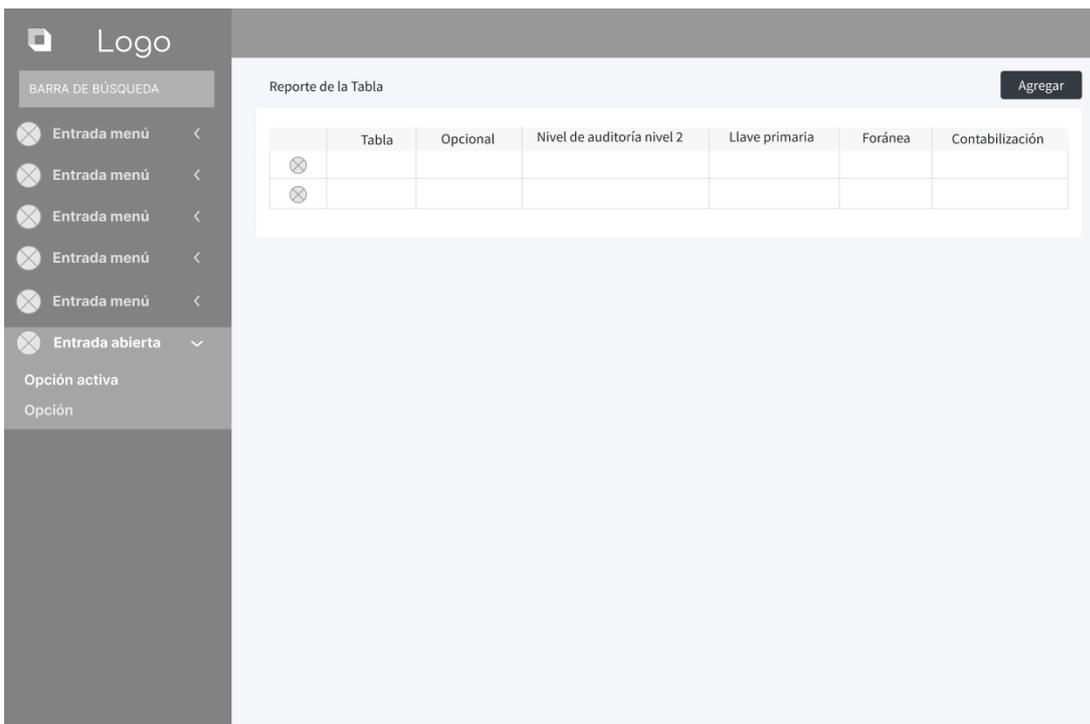
En la Figura 19 se representa el diseño de la ventana emergente para la bitácora nivel 2, según los permisos de usuario y las opciones que tiene disponibles.

Figura 19
Seleccionar la bitácora nivel 2.



En la Figura 20 se presentan las tablas que tiene el sistema con sus atributos.

Figura 20
Listado de la tabla.



La Figura 21 indica un prototipo de formulario para la creación y modificación de una tabla del sistema.

Figura 21
Mantenimiento de la tabla.

Logo

BARRA DE BÚSQUEDA

- Entrada menú <
- Entrada abierta ▾
- Opción activa
- Opción

Tabla Cancelar Guardar

Tabla: NOMBRE DE LA TABLA

Opcional: DESCRIPCIÓN DE LA TABLA

Llave primaria: NOMBRE DE LA LLAVE PRIMARIA

Columna estado: NOMBRE DEL ESTADO

Foránea: Si No

Auditoría nivel 2: Si No

Contabilización: Si No

Columna	Tipo	Descripción	Contabilización

Tabla relacionada	Llave foránea

A continuación, se representa la Figura 22 en la que se expone un listado de todas las tablas que existen en el sistema para facilitar su selección y visualización en la bitácora nivel 1.

Figura 22
Visualizar bitácora nivel 1 de todo el sistema.

Logo

BARRA DE BÚSQUEDA

- Entrada menú <
- Entrada abierta ▾
- Opción activa
- Opción

Reporte de la Tabla

--Seleccionar tabla--

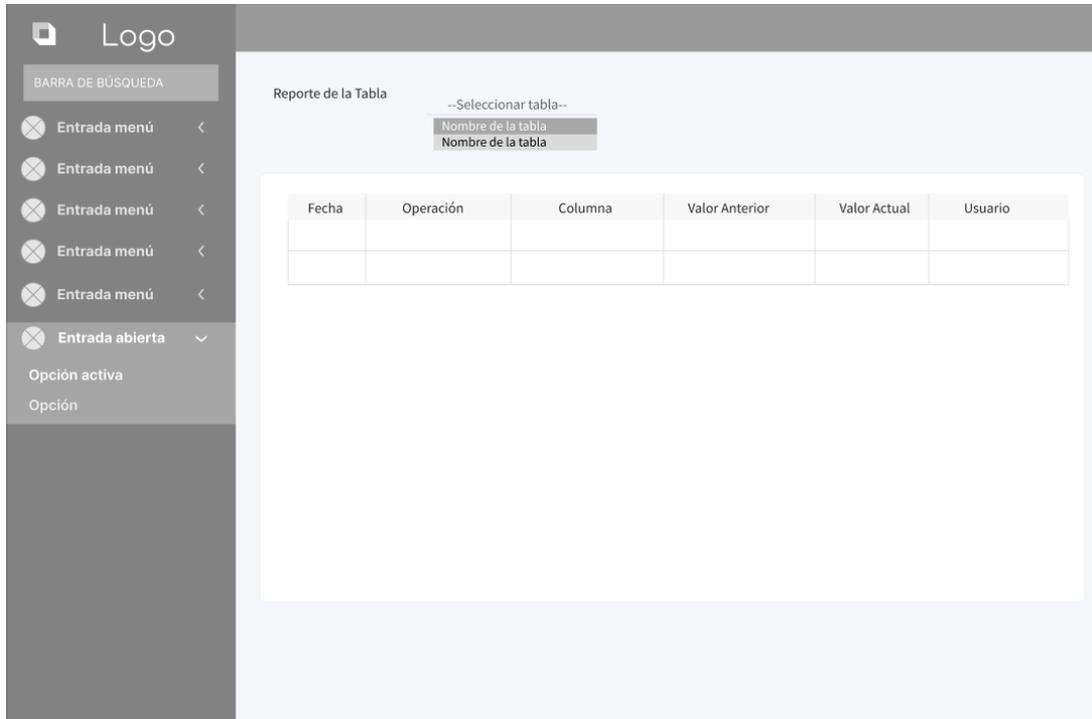
- Nombre de la tabla
- Nombre de la tabla

Fecha de creación	Creado por	Fecha de modificación	Modificado por	Última acción

La Figura 23 expone el listado de tablas para los administradores del sistema en la bitácora nivel 2.

Figura 23

Visualizar bitácora nivel 2 de todo el sistema.



4.2 Diseño visual

El diseño gráfico constituye un aspecto muy importante para asegurar una adecuada interacción entre los usuarios y la plataforma (Hassan et al. 2002). Así, la disposición de los elementos, la gama de color, la tipografía, los contrastes y las imágenes deben priorizar una lectura fácil, atractiva, comprensible y clara en coordinación con las necesidades del usuario y los objetivos de la bitácora (Anchundia et al. 2020). Para este propósito se adopta la imagen visual predefinida en el proyecto UDA-ERP versión 2.0 que cumple con las características de usabilidad y accesibilidad requeridas. A partir de estas premisas y el wireframe realizado en el paso anterior, se inicia un proceso de maquetación que se fundamenta en la creatividad para asignar una estética visual a la plataforma. A continuación, se exponen los tres aspectos considerados:

- Colores. Se adoptan los colores corporativos del proyecto para garantizar armonía en el diseño. Así, se utiliza el código de color RGB 58, 58, 64 para el encabezado y la caja de herramientas, y el color RGB 245, 246, 250 para el espacio de trabajo. Estos colores se combinan con tonos medios para la tipografía y las imágenes.

- Tipografía. Con el afán de agilizar la lectura de los contenidos del módulo de bitácora se utiliza un solo tipo de fuente: *Khang Medium* creada por *Indian Type Foundry*, con variaciones en negrita para los títulos. Esta tipografía garantiza legibilidad y equilibrio.

- Imágenes. Para asegurar dinamismo en el diseño se emplean íconos de fácil entendimiento que expliquen cada una de las opciones disponibles. Además, en el encabezado se incluye el logotipo del proyecto UDA-ERP versión 2.0, así como imágenes que describan al usuario y las notificaciones de la plataforma.

4.3 Software

El desarrollo del módulo de bitácora está elaborado con el lenguaje de programación Python con ayuda del Framework Django, a su vez el almacenamiento de toda la información está generada por el gestor de base de datos PostgreSQL. En la creación de este módulo participan diferentes tablas dentro de la base las cuales corresponden a: tabla y rol. Estas tablas son utilizadas para la generación y visualización del contenido de

la bitácora nivel 2. Adicionalmente, se vio la necesidad de añadir una tabla al sistema la cual lleva todo el almacenamiento de la bitácora nivel 2 y es denominada: “auditoria_n2”. En este sistema intervienen todas las tablas para que se pueda generar registros el cual permita almacenar información relevante al módulo, ya sea por la columna en que el usuario crea, modifica o elimina; cabe mencionar que esta creación de registros se realiza únicamente si tiene los permisos adecuados, caso contrario el sistema no puede almacenar o generar información con esa tabla.

Finalmente, el sistema se acomoda para que el usuario administrador, administrador de la empresa y el usuario con privilegios de visualizar las bitácoras, tengan un control interno y externo de la información delicada de las MYPIMES, para que de esta forma generar procesos de revisión o auditorías si en caso de que algún siniestro ocurriese durante la ejecución del ERP.

Gestión de Rol

Se realiza un alcance en el módulo de Rol, para agregar un nuevo campo que ayude a visualizar tanto la bitácora nivel 1 y 2 en la versión 2.0 del software UDA-ERP. El cambio se observa en la Figura 24.

Figura 24
Agregar un nuevo campo en el módulo de Rol.



Gestión de Tabla

En la versión 1.0 del software UDA-ERP existe la opción de crear bitácora nivel 1. En la versión 2.0, representada en la Figura 25 y Figura 26, todas las tablas van a tener bitácora nivel 1 y únicamente en las tablas que se requieran mayor información tendrán la bitácora nivel 2.

Figura 25
Mantenimiento de la tabla de la versión 1.0.

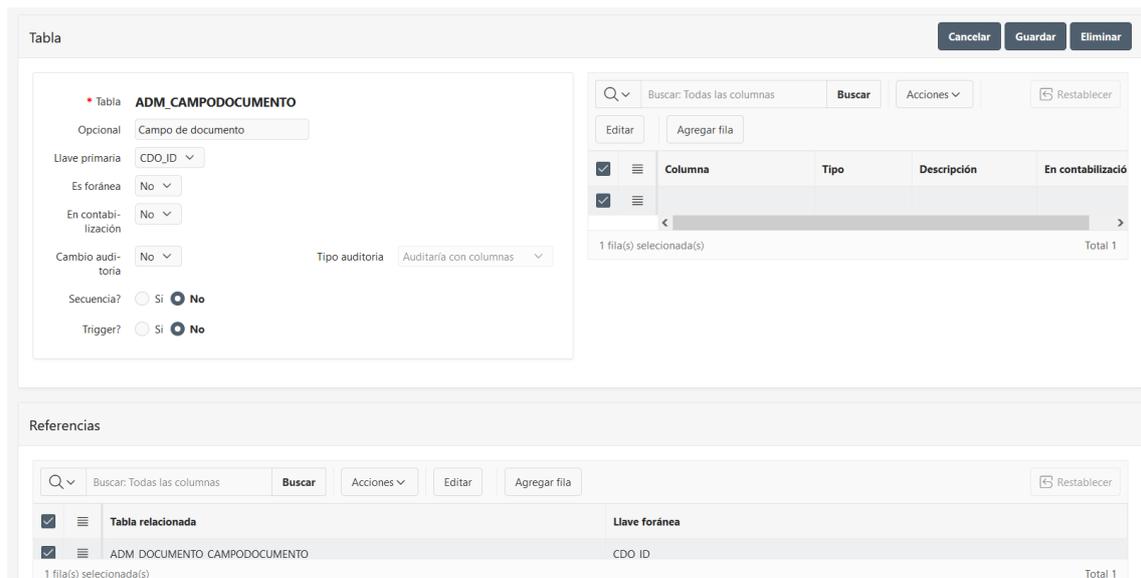


Figura 26
Mantenimiento de la tabla de la versión 2.0.

✕ Cancelar
✔ Guardar

Nombre: * con.transaccioncab

Alias: Movimientos Cuentas

Llave primaria: * cab_id

Columna estado: * cab_estado

Es foránea: * Sí No

Nivel auditoría 2: * Sí No

En Contabilización: * Sí No

+ Add
✎ Edit
🔄 Update
✕ Cancel

Columna	En contabilización
No records to display	

Tabla relacionada	Llave foránea
con.transacciondet	cab_id

Visualización de Bitácora Nivel 1

En la versión 2.0 se agrega una nueva opción en el menú del sistema el cual consiste que el administrador del sistema va a visualizar la bitácora nivel 1 de todas las empresas de una tabla seleccionada, como se muestra en la Figura 27.

Figura 27
Visualización de bitácora nivel 1.

Reporte Auditoría Nivel 1

ID	Elemento	Fecha de crea...	Usuario creador	Modificado el	Usuario modif...	Última acción	Empresa
14	GABI	6/1/2022, 13:53:12	Luis Guerrero	27/5/2022, 8:25:44	Gabriela Chica	Modificación	GABI
3	erpuda	8/1/2022, 17:33:42	Luis Guerrero	17/1/2022, 16:33:09	Jorge Peralta	Modificación	erpuda
9	Testing	1/2/2022, 15:33:22	Luis Guerrero	9/2/2022, 18:23:21	Desconocido	Creación	Testing
10	dummy	4/2/2022, 14:12:42	Luis Guerrero	13/6/2022, 19:23:12	Desconocido	Creación	dummy

10 Items per page
1 of 1 pages (4 items)

Visualización de Bitácora Nivel 2

En la versión 2.0 se agrega una nueva opción en el menú del sistema el cual consiste que el administrador del sistema va a visualizar la bitácora nivel 2 de todas las empresas de una tabla seleccionada, como se visualiza en la Figura 28.

Figura 28
Visualización de bitácora nivel 2.

Reporte Auditoría Nivel 2

adm.rol

Columna	Fecha	Operación	Valor anterior	Valor nuevo	Empresa	Usuario
rol_nombre	5/8/2022, 0:00:00	EDITAR	ADMINISTRADOR	ADMINISTRADORES	erpuda	Luis Guerrero
emp	5/8/2022, 0:00:00	EDITAR	GABI	14	erpuda	Luis Guerrero
rol_nombre	5/8/2022, 0:00:00	EDITAR	ADMINISTRADORES	Contadores	erpuda	Luis Guerrero
emp	5/8/2022, 0:00:00	EDITAR	GABI	14	erpuda	Luis Guerrero
rol_id	5/8/2022, 0:00:00	CREAR		4	erpuda	Luis Guerrero
rol_nombre	5/8/2022, 0:00:00	CREAR		Prueba1	erpuda	Luis Guerrero
emp	5/8/2022, 0:00:00	CREAR		3	erpuda	Luis Guerrero
rol_estado	5/8/2022, 0:00:00	CREAR		1	erpuda	Luis Guerrero
rol_nombre	5/8/2022, 0:00:00	EDITAR	Prueba1	Prueba4	erpuda	Luis Guerrero
emp	5/8/2022, 0:00:00	EDITAR	erpuda	3	erpuda	Luis Guerrero

10 Items per page 1 of 7 pages (63 items)

4.4 Entrenamiento

Para facilitar el uso por parte de los usuarios al momento de utilizar el módulo de bitácora nivel 2, se desarrolla el manual de usuario. Este manual describe las funcionalidades que tiene el sistema, a su vez es generado con el mismo formato del proyecto UDA-ERP versión 2.0, este mismo se halla en el anexo 2 (Astudillo Rodríguez et al. 2022). En este caso el módulo tiene requisitos previos antes de utilizar el software los cuales son: el usuario debe estar registrado en el sistema, el usuario debe haber ingresado en el sistema y el usuario debe estar en la opción del sistema.

5. Conclusión

Dentro de las investigaciones realizadas en la revisión de la literatura se han presentado distintas formas de manejar el comportamiento de los usuarios dentro de los sistemas. Esta información es relevante puesto que, generó un aprendizaje de qué datos pueden ser manejados dentro de un módulo de bitácora. De esta manera se genera los atributos necesarios para la creación del módulo en los destacados son los siguientes: la identificación del usuario quién realiza una acción dentro del sistema, fecha actual, columna afectada, registro afectado, tabla afectada, entre otros.

En el caso del software UDA-ERP versión 2.0 tiene la expectativa que este módulo de bitácora nivel 2, contribuya con los procesos de auditoría interna como externa o con revisiones de información dentro de las MYPIMES. Un punto importante dentro de este módulo provee la trazabilidad de la ejecución del usuario de una empresa de esta manera se puede generar mejoras del sistema, corregir errores, gestionar las acciones del usuario, verificar comportamientos inadecuados, prevenir fugas de información, detectar amenazas, entre otros.

Como se evidencia el módulo funciona como una grabación secuencial de las acciones del usuario dentro del sistema, de esta manera el software genera pistas que ayuden al auditor a tener un mayor control. De igual manera brinda seguridad al administrador del sistema y al administrador de la empresa, puesto que cualquier información errónea pueda ser detectada rápidamente y de esta forma no generar mayores inconvenientes a la empresa.

Al momento de desarrollar software es importante realizar las validaciones del sistema y a su vez dar un manejo funcional para el usuario. Por ello, las validaciones del módulo de bitácora permitieron garantizar que este módulo se acople a la infraestructura del sistema en desarrollo y asegurar la mejor calidad posible.

A su vez, realizar una explicación del funcionamiento del sistema para que el cliente sepa que funcionalidad puede realizar con el software y de esta forma generar un mayor entendimiento.

Finalmente, como se observa el nuevo módulo de bitácora nivel 2 para el software UDA-ERP versión 2.0 es eficiente, puesto que permite a las empresas llevar un mayor control de las transaccionalidades durante la gestión de un usuario para minimizar errores o cualquier evento que ocurriese durante la gestión. A su vez, ayuda a los auditores, tanto internos como externos, en los procesos de auditoría con toda la información almacenada durante el dentro del sistema para validar de manera eficaz los datos de la empresa y describir cualquier problema ocurrido durante el tiempo que requiera esa auditoría.

6. Referencias

- Adelfio, C. M. y Cho, C. y Farrell, H. y Hichwa, M. y Jones, C. y Kennedy, S. y Peake, D. y Rayner, A. y Sewtz, M. y Spadafore, S. y Straub, J. (2012). *Oracle® Application Express Application Builder 4.1 User's Guide*. April.
- Ahmad, A. y Saad, M. y Bassiouni, M. y Mohaisen, A. (2018). Towards blockchain-driven, secure and transparent audit logs. *ACM International Conference Proceeding Series*, 443–448. <https://doi.org/10.1145/3286978.3286985>
- Anchundia, C. L. y Valero, M. P. y Pages, W. (2020). *El diseño gráfico y su relación con el desarrollo de páginas web*. 2(1), 29–37.
- Asnal, H. y Junadhi y Jamaris, M. y Mardainis y Irawan, Y. (2022). Workshop UI/UX Design dan Prototyping dengan Figma di SMK Taruna Masmur Pekanbaru. *J-PEMAS - Jurnal Pengabdian Masyarakat*, 3(1), 18–25. <https://doi.org/10.33372/j-pemas.v3i1.800>
- Astudillo Rodríguez, C. y Crespo Martínez, E. y Dueñas Andrade, I. (2018). UDA-ERP: Emprendimiento y Gestión de recursos empresariales. La llave para la vinculación empresarial. En *MEMORIAS XIV FORO Internacional del Emprendedor* (pp. 92–106). Casa Editora de la Universidad del Azuay.
- Astudillo Rodríguez, C. y Maldonado Matute, J. y Crespo Martínez, E. y Erazo Garzón, L. (2022). *UDA-ERP para PYMES Módulos inventarios y manufactura*. Universidad del Azuay.
- Ávila, R. y Khoury, R. y Khoury, R. y Petrillo, F. (2021). Use of Security Logs for Data Leak Detection: A Systematic Literature Review. *Security and Communication Networks*, 2021(4). <https://doi.org/10.1155/2021/6615899>
- Challenger Pérez, I. y Díaz Ricardo, Y. y Becerra García, R. (2014). El lenguaje de programación Python/The programming language Python. *Revista Ciencias Holguín*, 20, 1–13. <http://www.linuxjournal.com/article/2959>
- Django. (2022). *The web framework for perfectionists with deadlines | Django*. <https://www.djangoproject.com/>
- Duisebekova, K. y Khabirov, R. y Zholzhan, A. (2021). DJANGO AS SECURE WEB-FRAMEWORK IN PRACTICE. *The Bulletin of KazATC Вестник KazATK*, 1(116), 275–281. <https://doi.org/10.52167/1609-1817-2020-116-1-275-281>
- Ginestà, M. G. y Mora, O. P. (2012). *Bases de datos en PostgreSQL* (pp. 1–78).
- Gómez Baryolo, O. y Estrada Sentí, V. y Bauta Camejo, R. R. y García Rodríguez, I. (2012). Modelo de gestión de log para la auditoría de información de apoyo a la toma de decisiones en las organizaciones. *Acimed*, 23(2), 187–200.
- González, R. (2011). *Python PARA TODOS*.
- Gorschek, T. y Garre, P. y Larsson, S. y Wohlin, C. (2006). A model for technology transfer in practice. *IEEE Software*, 23(6), 88–95. <https://doi.org/10.1109/MS.2006.147>
- Hadidi, M. y Hadidi, S. (2020). ERP Security Based on Web Services. *Journal of Computer Science and Technology*, 20(2). <https://www.researchgate.net/publication/343417638>
- Hassan, Y. y J, F. y Fernández, M. y Iazza, G. (2002). *Diseño Web Centrado en el Usuario: Usabilidad y Arquitectura de la Información*. 2, 461.
- Holovaty, A. y Kaplan-Moss, J. (2008). *El libro de Django* (J. Dunck (ed.)).
- Hoyos Estrada, S. y Sastoque Gómez, J. D. (2020). *Marketing Digital como oportunidad de digitalización de las PYMES en Colombia en tiempo del Covid – 19*. 3(1), 39–46.
- Hrischev, R. (2020). ERP systems and data security. *IOP Conference Series: Materials Science and Engineering*, 878(1). <https://doi.org/10.1088/1757-899X/878/1/012009>
- Inalef, K. y Guerra, J. y Aros, C. y Scheihing, E. (2010). DESARROLLO DE UNA BITÁCORA VIRTUAL PARA EL REGISTRO DE INTERACCIONES EN CLASES PRESENCIALES. *Congreso Iberoamericano de Informática Educativa*, 595–602. <http://www.tise.cl/volumen6/TISE2010/Documento84.pdf>
- Molina Ríos, J. R. y Loja Mora, N. M. y Zea Ordóñez, M. P. y Loaiza Sojos, E. L. (2016). Evaluación de los Frameworks en el Desarrollo de Aplicaciones Web con Python. *Revista Latinoamericana de Ingeniería de Software*, 4(4), 201. <https://doi.org/10.18294/relais.2016.201-207>
- Muñoz Razo, C. (2002). *Auditoría en sistemas computacionales*. PEARSON Prentice Hall.

- https://books.google.com.ec/books?hl=es&lr=&id=3hVDQuxTvxc&oi=fnd&pg=PR11&dq=auditorias+sistemas&ots=3gPqjjBXog&sig=kTom5rBn6lpVrea5VAiW6Lpr44&redir_esc=y#v=onepage&q=auditorias sistemas&f=false
- Oracle. (2022). *APEX*. <https://apex.oracle.com/es/>
- Pérez de Lema, D. G. y Francisco Javier, G. M. y Aragón Sánchez, A. (2011). *Análisis Estratégico para el Desarrollo de la MPYME en Iberoamérica*.
- Piattini Velthuis, M. y Del Peso Navarro, E. y Del Peso Ruiz, M. (2008). *Auditoría de Tecnologías y Sistemas de Información*. RA-MA Editorial. https://books.google.com.ec/books?hl=es&lr=&id=6o2fDwAAQBAJ&oi=fnd&pg=PR7&dq=auditorias+sistemas&ots=AOV5xYctw5&sig=CvM_78-S1NIYTcc1YrZ89CzCrBI&redir_esc=y#v=onepage&q=auditorias sistemas&f=false
- Piñuel, J. (1997). *Teoría de la comunicación y gestión de las organizaciones* (p. 127).
- Quispe Fernández, G. M. y Arellano Cepeda, O. E. y Ayaviri Nina, D. (2016). Aplicación de la Auditoría en las MyPEs del Ecuador: Un estudio de la demanda. *Revista de Investigaciones Altoandinas - Journal of High Andean Research*, 18(4), 483. <https://doi.org/10.18271/ria.2016.241>
- Rajavat, Y. S. y Bansal, A. y Dubey, S. (2020). Analysis of Efficient Secure & Privacy Preserving Technique for Cloud ERP. En *Mukt Shabd Journal: Vol. IX* (Número X, pp. 518–526).
- Romensky, S. A. y Rotkov, S. I. (2020). *Construction of 3D wireframe model in problem of converting paper drawing and design documentation into an electronic model of an object*. 67–82. <https://doi.org/https://doi.org/10.51130/graphicon-2020-1-67-82>
- Roumiana, A. y Georgiev, G. (2019). *ERP Security, Audit and Process Improvement*. Springer International Publishing. <https://doi.org/10.1007/978-3-030-01659-3>
- Salas-Zárate, M. D. P. y Alor-Hernández, G. y Valencia-García, R. y Rodríguez-Mazahua, L. y Rodríguez-González, A. y López Cuadrado, J. L. (2015). Analyzing best practices on Web development frameworks: The lift approach. *Science of Computer Programming*, 102, 1–19. <https://doi.org/10.1016/j.scico.2014.12.004>
- Scrum. (2020). *The Home of Scrum*.
- Syncfusion Inc. (2001). *Syncfusion Documentation*. <https://help.syncfusion.com/>
- Vidal-Silva, C. L. y Sánchez-Ortiz, A. y Serrano, J. y Rubio, J. (2021). Experiencia académica en desarrollo rápido de sistemas de información web con Python y Django. *Formacion Universitaria*, 14(5), 85–94. <https://doi.org/http://dx.doi.org/10.4067/S0718-50062021000500085>
- Zea Ordoñez, M. P. y Molina Ríos, J. y Redrován Castillo, F. (2017). *Administración de base de datos con PostgreSQL*.
- Zevallos Vallejos, E. G. (2003). Micro, pequeñas y medianas empresas en América Latina. *Revista de la CEPAL*, 79, 53--70.

ANEXOS

Anexo 1 PostgreSQL

Creación de la auditoría nivel 1 dentro de tabla

```
call sis."pr_agregarTablaAud"('sis.tabla');
```

Creación de tabla Bitácora (Auditoría nivel 2)

```
CREATE TABLE "adm"."auditoria_n2"  
(  
  "audn2_id" BigSerial NOT NULL,  
  "audn2_columna" Character varying(64) NOT NULL,  
  "audn2_operacion" Character varying(64) NOT NULL,  
  "audn2_fecha" Timestamp NOT NULL,  
  "audn2_valor_anterior" Character varying(256),  
  "audn2_valor_nuevo" Character varying(256),  
  "tab_nombre" Character varying(64) NOT NULL,  
  "usu_id" Bigint NOT NULL,  
  "emp_id" Bigint NOT NULL,  
  "audn2_registro_id" Bigint NOT NULL  
)  
WITH (  
  autovacuum_enabled=true)  
;  
  
CREATE INDEX "IX_auditoria_n2_1" ON "adm"."auditoria_n2" ("tab_nombre")  
;  
  
CREATE INDEX "IX_auditoria_n2_2" ON "adm"."auditoria_n2" ("usu_id")  
;  
  
CREATE INDEX "IX_auditoria_n2_3" ON "adm"."auditoria_n2" ("emp_id")  
;  
  
ALTER TABLE "adm"."auditoria_n2" ADD CONSTRAINT "PK_auditoria_n2" PRIMARY KEY  
("audn2_id")  
;  
  
-- Create foreign keys (relationships) section -----  
  
ALTER TABLE "adm"."auditoria_n2"  
ADD CONSTRAINT "fk_sistabla_audauditorian2"  
FOREIGN KEY ("tab_nombre")  
REFERENCES "sis"."tabla" ("tab_nombre")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION  
;  
  
ALTER TABLE "adm"."auditoria_n2"  
ADD CONSTRAINT "fk_admusuario_audauditorian2"  
FOREIGN KEY ("usu_id")  
REFERENCES "adm"."usuario" ("usu_id")  
ON DELETE NO ACTION  
ON UPDATE NO ACTION  
;  
  
ALTER TABLE "adm"."auditoria_n2"  
ADD CONSTRAINT "fk_admpresidencia_audauditorian2"  
FOREIGN KEY ("emp_id")  
REFERENCES "adm"."empresa" ("emp_id")
```

```

ON DELETE NO ACTION
ON UPDATE NO ACTION
;

```

Procedimiento almacenado que guarda a un csv en una ruta especificada los datos y los borra de la tabla de la base de datos

```

-- PROCEDURE: adm.pr_respaldarauditoria()

-- DROP PROCEDURE IF EXISTS adm.pr_respaldarauditoria();

CREATE OR REPLACE PROCEDURE adm.pr_respaldarauditoria(
)
LANGUAGE 'sql'
AS $BODY$
DO
$$DECLARE
datestr text := to_char(current_timestamp, 'YYYY-MM-DD');
BEGIN
EXECUTE format(
'COPY '
'(SELECT * FROM adm.auditoria_n2) '
'TO %L WITH DELIMITER ";" CSV HEADER',
'C:\csv\respaldo_auditoria_' || datestr || '.csv'
);
TRUNCATE TABLE adm.auditoria_n2
RESTART IDENTITY;
END;$$;
$BODY$;
ALTER PROCEDURE adm.pr_respaldarauditoria()
OWNER TO postgres;

```

Creación de la auditoría nivel 1 y 2 dentro de la tabla opción

```

INSERT INTO adm.opcion(
opc_id, opc_id_padre, opc_nombre, opc_descripcion, opc_url, opc_icon, opc_menu_tipo, opc_orden,
opc_menu_sis, opc_inc_agregar, opc_inc_modificar, opc_inc_eliminar, opc_inc_imprimir, mod_id,
tab_nombre, opc_estado, opc_codigo)
VALUES
(94, 91,'Auditoría N1','Auditoría nivel
1','/555/auditorian1/list/',NULL,'O',20,NULL,1,1,1,1,NULL,NULL,1,'AUD_1');

```

```

INSERT INTO adm.opcion(
opc_id, opc_id_padre, opc_nombre, opc_descripcion, opc_url, opc_icon, opc_menu_tipo, opc_orden,
opc_menu_sis, opc_inc_agregar, opc_inc_modificar, opc_inc_eliminar, opc_inc_imprimir, mod_id,
tab_nombre, opc_estado, opc_codigo)
VALUES
(95, 91,'Auditoría N2','Auditoría nivel
2','/555/auditorian2/list/',NULL,'O',30,NULL,1,1,1,1,NULL,NULL,1,'AUD_2');

```

Creación de una columna más para el proceso de auditoría en el rol

```

ALTER TABLE adm.rol
ADD rol_esadmin Boolean;

```

Anexo 2 Manual de Usuario.

Administración

Se induce al usuario en la especificación de aspectos básicos y de configuración de componentes del software, como son: i) el ingreso, ii) la organización de la aplicación vista de un modo general, en donde, se describe la pantalla de inicio, reportes interactivos y mantenimientos.

Ingreso a la aplicación

El acceso a la aplicación UDA-ERP se lo efectúa mediante el navegador web. Para un correcto funcionamiento se recomienda Google Chrome o Mozilla Firefox.

En la barra de direcciones del navegador digite:

erp.uazuay.edu.ec

Terminada la carga de la página web, en el navegador se visualiza la pantalla de inicio de sesión. Se solicita al usuario el ingreso de la información que se visualiza en la siguiente figura:

Figura 29

Ingreso a la aplicación



La imagen muestra la interfaz de inicio de sesión de UDA ERP. El encabezado contiene el logo de UDA ERP y el texto "UDA Enterprise Resource Planning". Debajo del encabezado, hay una serie de campos de entrada y un botón. Los campos son: "Empresa" (con un ícono de casa), "-- Sucursal --" (lista desplegable), "-- Período --" (lista desplegable), "Usuario" (con un ícono de persona) y "Contraseña" (con un ícono de llave). El botón "Iniciar sesión" es azul y está ubicado al final. Seis flechas rojas numeradas del 1 al 6 apuntan a cada uno de estos elementos de la interfaz.

1. Digite el código asignado a su empresa por parte del administrador del sistema.
2. Seleccione de la lista desplegable la sucursal a la que desea ingresar.
3. Seleccione de la lista desplegable el periodo al que desea ingresar.
4. Digite el nombre de usuario asignado previamente por el administrador.
5. Digite la contraseña.
6. Clic en el botón Iniciar Sesión para ingresar al sistema.

El sistema verifica los datos ingresados por el usuario, en caso de haber datos incorrectos o incompletos, el sistema notifica en pantalla lo sucedido.

Organización general de la aplicación

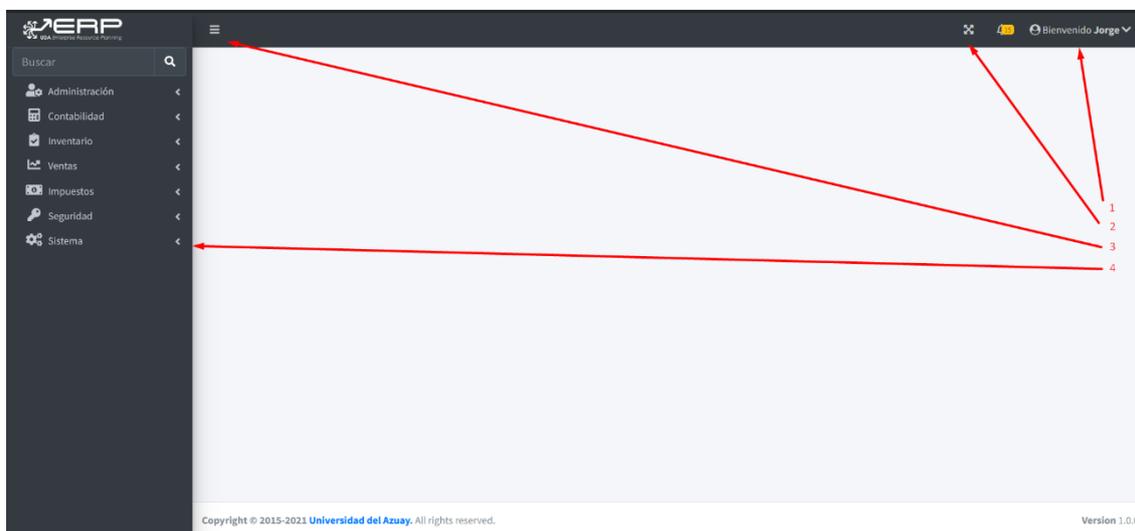
Una vez que se ingresa a la aplicación, es importante identificar elementos básicos en su organización, como son: elementos en la pantalla de inicio, cerrar la aplicación, organización de los elementos en la pantalla del reporte interactivo, organización de los elementos en la pantalla de datos de mantenimientos y objetos en pantalla

Organización de los elementos en la pantalla de inicio

La aplicación en su pantalla de inicio cuenta con diferentes elementos visualizados a lo largo de área de trabajo, como se detalla a continuación:

Figura 30

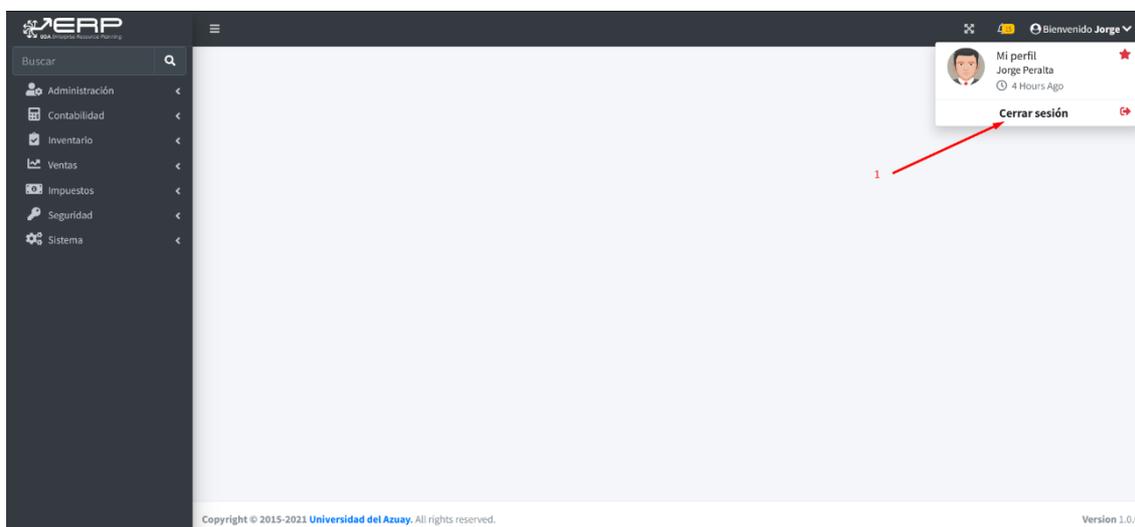
Organización de elementos en pantalla de inicio



1. Nombre de usuario que ingresó al sistema, al hacer clic sobre este elemento, se despliega la opción de **Cerrar sesión**.

Figura 31

Cerrar la aplicación



- a. **Cerrar la aplicación**, esta opción permite al usuario terminar su sesión de trabajo actual o ingresar en un periodo o sucursal diferente.
 1. Clic sobre el nombre de usuario que ingresó al sistema, se despliega la opción para **Cerrar sesión**.
 2. Clic sobre **Cerrar sesión**, el sistema redirige al usuario a la pantalla de ingreso a la aplicación.
2. Botón para activar el modo de Pantalla completa.
3. Botón para mostrar/ocultar menú lateral.
4. Opciones de menú principal.

Organización de los elementos en la pantalla del reporte interactivo

La aplicación cuenta con un formato estandarizado para la pantalla de reporte interactivo, el mismo que se visualiza desde varias opciones en la aplicación y permite al usuario alterar el diseño de los datos enlistados. Cuenta con diferentes elementos en el área de trabajo, como se detalla a continuación:

Figura 32

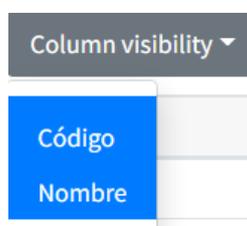
Organización de elementos en pantalla del reporte interactivo



1. El botón **Agregar** guía hasta el formulario de datos de mantenimientos relacionado con la opción elegida en el menú principal de la pantalla de inicio, y podrá agregar un nuevo registro de información. Algunas opciones de la aplicación no permiten esta acción, si es el caso, no se visualiza este botón.
2. Barra de búsqueda, ubicada en la parte superior, permite explorar entre los diferentes elementos enlistados en pantalla. Esta sección se compone de:
 - a. Área de texto, es el espacio de entrada para indicar la cadena que se va a buscar. Si previamente selecciona uno de los campos enlistados en la lupa, su nombre se muestra como fondo en este espacio y dicha búsqueda se realiza solo en la columna a la que hace referencia, en caso contrario buscará una coincidencia con cualquiera de los datos existentes.
3. El botón **Imprimir** otorga al usuario una vista previa de impresión para, finalmente, imprimir el listado de datos en formato de tabla.
4. El botón **PDF** descarga el conjunto de resultados en formato PDF.
5. El botón **Excel** descarga el conjunto de resultados en formato de Libro de Excel.
6. El botón **Copiar** copia al portapapeles del dispositivo el conjunto de resultados.
7. **Visibilidad de la columna**, permite crear un grupo en una o varias columnas y los muestra como un registro maestro.

Figura 33

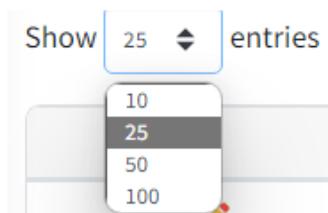
Visibilidad de la columna



8. Indicador del total de registros en pantalla, permite agregar o reducir la cantidad de registros a mostrar en pantalla en base al valor seleccionado.

Figura 34

Indicador del total de registros en pantalla



9. Encabezado de columnas, corresponde con el nombre de cada uno de los campos visualizados en el listado actual, relacionados con la opción elegida en el menú principal de la pantalla de inicio.

Al hacer clic en cualquier encabezado de columna se visualiza un menú que incluye:

- a. Icono de orden ascendente, ordena el informe por la columna en orden ascendente.
 - b. Icono de orden descendente, ordena el informe por la columna en orden descendente.
10. Botón para **Editar** la información del registro guía hasta el formulario de datos de mantenimientos relacionado con la opción elegida en el menú principal de la pantalla de inicio, y podrá consultar el detalle del registro seleccionado o modificar la información actual.
 11. **Detalle** del listado previamente ingresado y almacenado en el sistema, y relacionado con la opción elegida en el menú principal de la pantalla de inicio.
 12. **Elementos de paginación** que indican el número de registros visualizados en pantalla y cuatro botones para avanzar o retroceder entre las diferentes páginas, ir al principio o al final.

Organización de los elementos en la pantalla de datos de mantenimientos

La aplicación cuenta con un formato estandarizado para la pantalla de datos de mantenimientos, la misma que se visualiza desde varias opciones en la aplicación y permite al usuario agregar, consultar, guardar o cancelar cambios sobre información específica de un registro, relacionado con el origen desde cual fue redirigido.

Si en la pantalla de reporte interactivo presiona el botón Agregar (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla del reporte interactivo), el área de trabajo se visualiza como se detalla a continuación:

Figura 35

Organización de elementos en pantalla de datos de mantenimientos, opción Agregar

A screenshot of a web form. At the top left, there is a title 'Mantenimiento operación 1' enclosed in a red box labeled '1'. At the top right, there are two buttons: 'Cancelar' with a close icon and 'Guardar' with a checkmark icon, both enclosed in a red box labeled '2'. Below the title, there is a text input field with the label 'Nombre*' and an asterisk, enclosed in a red box labeled '3'. The form has a light gray background and a white border.

1. **Título** de la opción en la que se encuentra.
2. Botones:
 - a. **Cancelar**, descarta el ingreso de datos y no los guarda en la aplicación. Redirige al usuario a la pantalla del reporte interactivo asociado.
 - b. **Guardar**, almacena el ingreso de datos en la aplicación, redirige al usuario a la pantalla del reporte interactivo asociado, en el que se visualiza el nuevo registro de datos.
3. **Campos de información** a ingresar, relacionados con la opción seleccionada. El signo * junto a la etiqueta indica que el campo es obligatorio.

Si en la pantalla de reporte interactivo presiona el botón Editar (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla del reporte interactivo), el área de trabajo se visualiza como se detalla a continuación:

Figura 36

Organización de elementos en pantalla de datos de mantenimientos, opción Editar

Mantenimiento operación 1

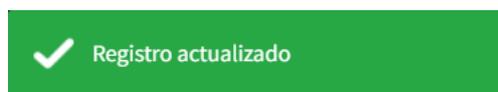
2 Cancelar Eliminar Guardar

3 Nombre*

1. Título de la opción en la que se encuentra.
2. Botones:
 - a. **Cancelar**, descarta la actualización de datos y no los guarda en la aplicación. Redirige al usuario a la pantalla del reporte interactivo asociado.
 - b. **Guardar**, almacena los datos actualizados en la aplicación, redirige al usuario a la pantalla del reporte interactivo asociado, en el que se visualiza el nuevo registro de datos.

Figura 37

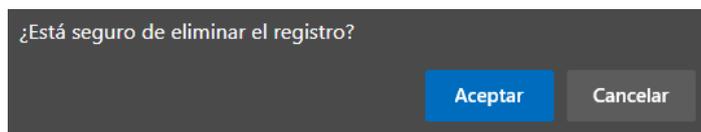
Confirmación de datos guardados



- c. **Eliminar**, muestra un cuadro de diálogo en el que el usuario confirma o cancela la eliminación, como se observa en la siguiente figura. Si el usuario confirma el eliminar, el registro es eliminado por la aplicación y redirige al usuario a la pantalla del reporte interactivo asociado.

Figura 38

Cuadro de diálogo para eliminar registro



3. **Campos de información** relacionados con la opción seleccionada, se muestran activos para editar y modificar únicamente los que la opción así lo permite. El signo * junto a la etiqueta indica que el campo es obligatorio.

Objetos en pantalla

En las diferentes opciones presentes en la aplicación se visualizan elementos comunes que se explican a continuación:

Texto

Un cuadro de texto permite ingresar la información relacionada con la opción de la aplicación seleccionada y a su etiqueta correspondiente, se visualiza con la siguiente apariencia:

Figura 39
Elemento de texto

Nombre* _____

Listas

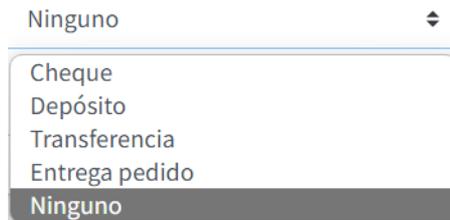
Las listas se presentan cuando se puede seleccionar entre varias opciones la información solicitada, estos elementos se visualizan con la siguiente apariencia:

Figura 40
Elemento para listas de selección



Clic sobre el símbolo marcado en el recuadro y esto abrirá una ventana emergente con una lista de información relacionada con el tema tratado en ese momento. Se visualiza como se detalla a continuación:

Figura 41
Elementos en lista de selección



Calendario

El calendario se presenta cuando se requiere del ingreso de una fecha específica. Este elemento se visualiza con la siguiente apariencia:

Figura 42
Elemento para calendario



Clic sobre el símbolo marcado en el recuadro y esto abrirá una ventana emergente con un calendario detallado del mes actual, puede retroceder o avanzar en los diferentes meses del año y seleccionar la fecha deseada con un clic. Se visualiza como se detalla a continuación:

Figura 43
Calendario



Seleccionar imagen

En las opciones de aplicación en las que se encuentre la posibilidad de incluir imágenes, siga los siguientes pasos:

1. En el botón Examinar hacer clic.

Figura 44

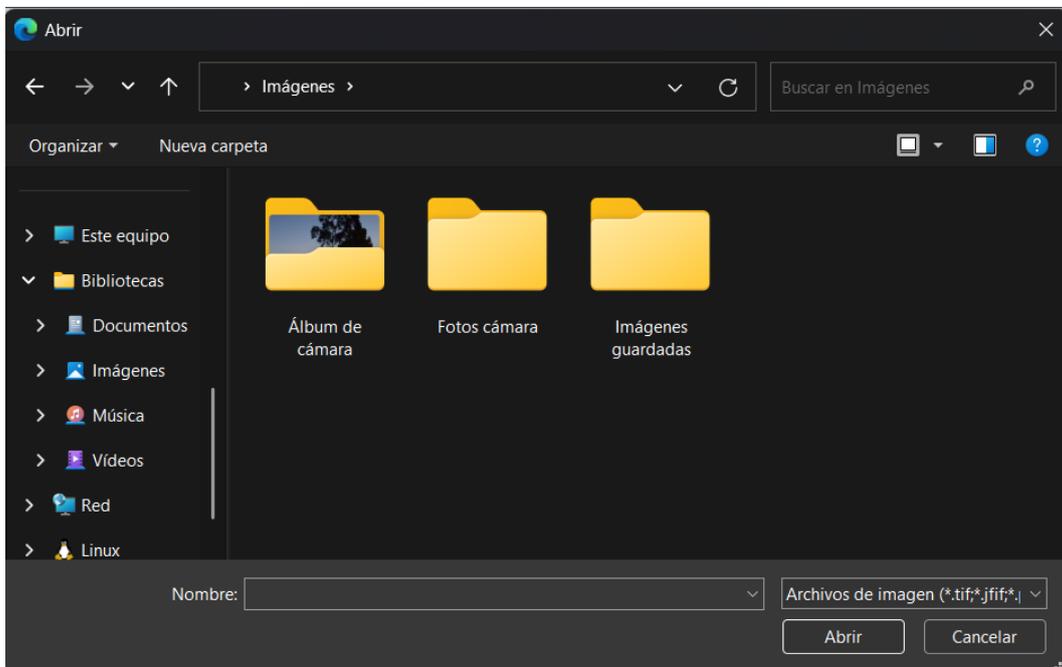
Botón para seleccionar imagen



2. Se muestra la ventana del explorador de Windows, localice la imagen, y haga clic en **Abrir**.

Figura 45

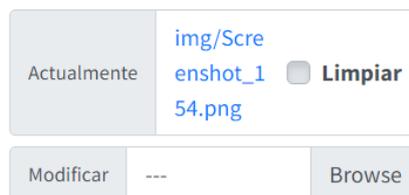
Explorador de Windows para seleccionar imagen



3. En la aplicación se mostrará el nombre de la imagen seleccionada.

Figura 46

Nombre de imagen seleccionada



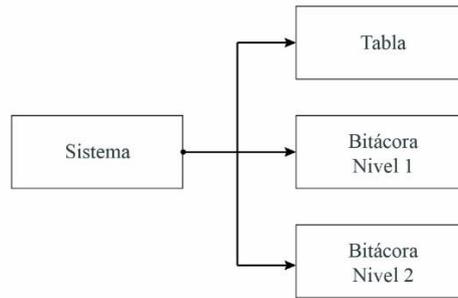
Módulo de gestión de Sistema

La gestión de tabla es un aspecto importante para el proceso de desarrollo del módulo de bitácora nivel 2 propuesto, puesto que genera los permisos necesarios para ejecutar la bitácora nivel 2 en las tablas

seleccionadas por el administrador del sistema. Aquí, se visualiza las llaves de la base de datos que permiten identificar la tabla que está siendo utilizada por un usuario específico. El manejo de los procesos de contabilización para reconocer la columna de la tabla facilita este proceso. También, este documento aspira a contribuir a la solución de problemas en caso de que alguna actividad realizada en la opción de tabla tanto de la bitácora nivel 1 o bitácora nivel 2.

Con el objetivo de guiar al usuario en el uso correcto de la aplicación, este manual está organizado por procesos; cada proceso detalla a su vez, las actividades requeridas.

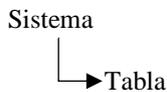
Figura 47
Gestionar sistema.



Gestionar tabla

Lista ordenada, codificada y agrupada de tabla que requiera la organización para el control de acceso basado en roles.

Para acceder a esta opción elija en el menú lateral:



Se enlista todas las tablas previamente almacenadas, en un reporte interactivo (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla del reporte interactivo).

La información que almacena la aplicación referente a gestión de tabla se muestra en la siguiente figura:

Figura 48
Listado de registros de la tabla.

Nombre	Alias	Nivel auditoría 2:	Llave primaria	Es foránea	En Contabilización	Bitácora
adm.documento	Documento	Si	doc_id	No	No	[icon]
adm.documento_periodo	Documento Periodo	No	dop_id	No	No	[icon]
adm.empresa	Empresa	Si	emp_id	No	No	[icon]
adm.mescerrado	Mes Cerrado	No	mce_id	No	No	[icon]
adm.opcion	Opción	No	opc_id	No	No	[icon]
adm.periodo	Periodo	Si	per_id	No	No	[icon]
adm.rol	Roles de los usuarios	Si	rol_id	No	No	[icon]
adm.sucursal	Sucursal	Si	suc_id	No	No	[icon]
adm.usuario	Usuarios del sistema	Si	usu_id	No	No	[icon]
adm.usuario_sucursal	Usuario sucursal	No	usc_id	No	No	[icon]
con.cambiopatrim_estruc	Cambio de patrimonio	No	cpe_id	No	No	[icon]
con.cuenta	Cuentas	No	cue_id	No	No	[icon]

El botón **Agregar** guía hasta el formulario de datos de mantenimientos, y podrá agregar una nueva tabla (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla de datos de mantenimientos).

El botón **Editar** del registro guía hasta el formulario de datos de la tabla, y podrá consultar o modificar el detalle de la información actual de la tabla seleccionada (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla de datos de mantenimientos).

La información que almacena la aplicación referente a una tabla se muestra en la siguiente figura:

Figura 49

Campos del formulario para la creación y modificación del registro.

Tabla relacionada	Llave foránea
con.transacciondet	cab_id

1. Campos del formulario (disponibles al crear y editar).
 - a. **Nombre**, selecciona de las opciones disponibles, la tabla que no es utilizada en el sistema.
 - b. **Alias**, frase corta que describe a la tabla.
 - c. **Llave primaria**, combinación de símbolos, números y/o letras que identifican a la tabla.
 - d. **Columna estado**, combinación de letras que permite identificar el estado que tiene cada tabla en el sistema.
 - e. **Es foránea**, define que columna contiene toda la información de otra tabla. Selección entre sí o no.
 - f. **Nivel de auditoría 2**, define los permisos para generar contenido y almacenamiento en las tablas con esta autorización. Selección entre sí o no.
 - g. **En Contabilización**, define si la tabla genera procesos de contabilidad. Selección entre sí o no.
2. Campos de la vista (disponibles al editar).
 - a. **Columna**, selecciona de la tabla, a la columna que es utilizada en el proceso de contabilización.
 - b. **En Contabilización**, verifica si la columna seleccionada es de contabilización.
3. Campos del proceso de contabilización (disponibles al editar).
 - a. **Tabla relacionada**, define en una lista todas las tablas que están relacionadas con una tabla específica.
 - b. **Llave foránea**, funciona como la llave principal de la tabla.
4. Botones del formulario (disponibles al crear y editar).
 - a. **Cancelar**, descarta el proceso de cambio de estado en movimientos. Redirige a la página de inicio.
 - b. **Guardar**, almacena la información del registro dentro de la base de datos. Redirige a la página de inicio y muestra un mensaje de "Registro creado" o "Registro editado".

Visualización de Bitácora Nivel 1

Gestiona en un listado la bitácora nivel 1 de las tablas que se encuentran en funcionamiento del sistema, con sus respectivas columnas adicionales para entender de manera general la información que maneja esa tabla.

Para acceder a esta opción elija en el menú lateral:



De esta manera se listará los datos de la bitácora nivel 1 previamente almacenados, estos datos se mostrarán en un reporte interactivo. (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla del reporte interactivo).

Figura 50

Visualización de la bitácora nivel 1 para el administrador del sistema.

Reporte Auditoría Nivel 1

adm.empresa

ID	Elemento	Fecha de crea...	Usuario creador	Modificado el	Usuario modif...	Última acción	Empresa
14	GABI	6/1/2022, 13:53:12	Luis Guerrero	27/5/2022, 8:25:44	Gabriela Chica	Modificación	GABI
3	erpuda	8/1/2022, 17:33:42	Luis Guerrero	17/1/2022, 16:33:09	Jorge Peralta	Modificación	erpuda
9	Testing	1/2/2022, 15:33:22	Luis Guerrero	9/2/2022, 18:23:21	Desconocido	Creación	Testing
10	dummy	4/2/2022, 14:12:42	Luis Guerrero	13/6/2022, 19:23:12	Desconocido	Creación	dummy

1 of 1 pages (4 items)

1. Listado de tablas disponibles en el sistema.
 - a. **Lista.** Seleccionar una de las tablas disponibles.
2. Campos de la vista
 - a) **ID**, combinación de símbolos, números y/o letras que identifican a la tabla.
 - b) **Elemento**, combinación de símbolos, números y/o letras que distingue al registro.
 - c) **Fecha de creación**, fecha de creación del registro.
 - d) **Usuario creador**, nombre del usuario que crea el registro.
 - e) **Modificado el**, fecha en que modifica el registro.
 - f) **Usuario modificador**, nombre del usuario que modifica el registro.
 - g) **Última acción**, combinación de letras el cual describe la acción realizada, puede ser: creación, modificación o eliminación.
 - h) **Empresa**, combinación de letras y números que nombran a una empresa.

Visualización de Bitácora Nivel 2

Gestiona en un listado la bitácora nivel 2, cabe mencionar que este registro está diseñado en un apartado de las tablas preexistentes, para tener un mayor control de las tablas que se encuentran en funcionamiento del sistema, adicionalmente se selecciona columnas adicionales para generar una lista con mayor información de los registros añadidos, modificados y eliminados de los mantenimientos de las tablas realizadas con sus respectivos permisos.

Para acceder a esta opción elija en el menú lateral:



De esta manera se listará los datos de la bitácora nivel 2 previamente almacenados, para visualizar esta información en un reporte interactivo. (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla del reporte interactivo).

Figura 51

Visualización de la bitácora nivel 2 para el administrador del sistema.

The screenshot shows a web interface with a dropdown menu at the top left containing 'adm.rol' and a search bar at the top right. Below these is a table with 10 columns: Columna, Fecha, Operación, Valor anterior, Valor nuevo, Empresa, and Usuario. The table contains 10 rows of log entries. At the bottom, there is a pagination control showing '1' selected, 'Items per page' set to 10, and '1 of 10 pages (95 items)'.

Columna	Fecha	Operación	Valor anterior	Valor nuevo	Empresa	Usuario
rol_nombre	5/8/2022, 0:00:00	EDITAR	ADMINISTRADOR	ADMINISTRADORES	erpuda	Luis Guerrero
emp	5/8/2022, 0:00:00	EDITAR	GABI	14	erpuda	Luis Guerrero
rol_nombre	5/8/2022, 0:00:00	EDITAR	ADMINISTRADORES	Contadores	erpuda	Luis Guerrero
emp	5/8/2022, 0:00:00	EDITAR	GABI	14	erpuda	Luis Guerrero
rol_id	5/8/2022, 0:00:00	CREAR		4	erpuda	Luis Guerrero
rol_nombre	5/8/2022, 0:00:00	CREAR		Prueba1	erpuda	Luis Guerrero
emp	5/8/2022, 0:00:00	CREAR		3	erpuda	Luis Guerrero
rol_estado	5/8/2022, 0:00:00	CREAR		1	erpuda	Luis Guerrero
rol_nombre	5/8/2022, 0:00:00	EDITAR	Prueba1	Prueba4	erpuda	Luis Guerrero
emp	5/8/2022, 0:00:00	EDITAR	erpuda	3	erpuda	Luis Guerrero

1. Listado de tablas disponibles en el sistema.

- Lista. Seleccionar una de las tablas disponibles.

2. Campos de la vista

- Columna. Combinación de símbolos, números y/o letras que identifican el atributo de la tabla.
- Fecha. Fecha de creación del registro.
- Operación. Combinación de letras que describe la acción realizada, puede ser: crear, editar o eliminar.
- Valor anterior. Combinación de símbolos, números y/o letras que detallan la información previa antes de su modificación.
- Valor nuevo. Combinación de símbolos, números y/o letras que detallan la información actual del sistema.
- Empresa. Combinación de letras y números que nombran a una empresa.
- Usuario: Combinación de símbolos, números y/o letras que describen al usuario que realizó algún tipo de cambio en el sistema.

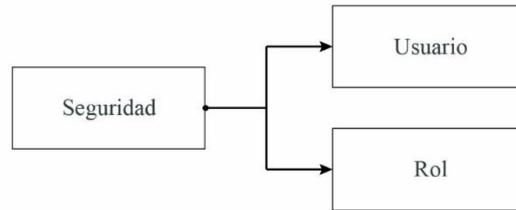
Módulo de gestión de Administración

La administración de la seguridad del sistema es importante, puesto que se genera permisos tanto para los usuarios y los roles que maneja en la empresa. En este caso el rol será modificado para que el usuario tenga los permisos de visualización y que estos permisos a su vez se les pueda otorgar a otros usuarios dentro de la empresa y sistema. De esta forma el objetivo es presentar una guía para que el usuario que esté gestionando la parte de las visualizaciones de las bitácoras pueda generar un mejor proceso.

De esta manera en los siguientes pasos se van a describir las partes importantes del módulo de visualización de bitácoras para que cada proceso sea entendible, que el uso sea lo más intuitivo posible y un correcto uso del módulo y las ventajas que tiene el administrador al poder realizar revisiones en el sistema. También este documento quiere contribuir con soluciones que pueden existir durante el proceso y que el usuario quien use este sistema tenga idea de porque puede suceder cualquier error y de esta manera generar un proceso sólido durante su uso

Figura 52

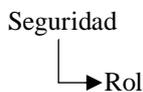
Gestionar administración de seguridad en la opción de rol.



Gestión de Rol

Es necesario que en el perfil del rol de usuario se especifique si corresponde con un Administrador de la aplicación, esto otorga privilegios como el de agregar o eliminar permisos a un rol en específico de usuarios, así también, el visualizar la bitácora de nivel 1 y/o nivel 2.

Para acceder a esta opción elija en el menú lateral:



Se enlista todos los roles previamente almacenados, en un reporte interactivo (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla del reporte interactivo).

Figura 53

Listado de los roles.

	Código	Nombre	Bitácora
	1	Administrador	
	2	Recursos Humanos	
	10	Gerente	

El botón **Agregar** guía hasta el formulario de datos de roles, y podrá agregar un nuevo módulo (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla de datos de mantenimientos).

El botón **Editar** del registro guía hasta el formulario de datos de mantenimientos, y podrá consultar o modificar el detalle de la información actual del rol seleccionado (consultar: Administración, Organización general de la aplicación, Organización de los elementos en la pantalla de datos de mantenimientos).

La información que almacena la aplicación referente a una opción se muestra en la siguiente figura:

Figura 54

Creación del campo en la tabla de rol.

Form fields:

- Nombre* Administrador
- Administrador

Buttons: Cancelar, Eliminar, Guardar

1. Campos del formulario.

- a. **Nombre**, combinación de símbolos, números y/o letras que determina el nombre del rol en una empresa.
- b. **Administrador**, define los permisos para visualizar la bitácora nivel 1 y 2. Este campo solo aparecerá si el administrador de la empresa les otorga el privilegio de visualizar. Este es un campo de selección entre verdadero o falso.

2. Botones del formulario.

- a. **Cancelar**, cancela todos los cambios que se han realizado durante el mantenimiento y no se proceden a guardar. Redirecciona a la lista de los roles.
- b. **Eliminar**, permite la eliminación de un rol. Este campo da un mensaje de confirmación.
- c. **Guardar**, permite almacenar información al crear o modificar un rol.