



Sistema de inventario para el refaccionamiento de peaje y de sistemas inteligentes de tráfico (ITS)

Inventory system for toll retrofitting and intelligent traffic systems (ITS)

Martha Ruth Camacho Vázquez

Universidad Tecnológica de Nayarit, Xalisco, México
martha.camacho@utnay.edu.mx
ORCID: 0009-0006-3358-3718

Jazmín Pérez Méndez

Universidad Tecnológica de Nayarit, Xalisco, México
jazmin@utnay.edu.mx
ORCID: 0000-0001-9334-9858

Janitzín Cárdenas Castellanos

Universidad Tecnológica de Nayarit, Xalisco, México
janitzin.cardenas@utnay.edu.mx
ORCID: 0000-0002-9893-6653

Nadia Teresa Adaile Benítez

Universidad Tecnológica de Nayarit, Xalisco, México
nadia.adaile@utnay.edu.mx
ORCID: 0000-0002-1816-4981

doi: <https://doi.org/10.36825/RITI.12.28.003>

Recibido: Junio 14, 2024
Aceptado: Agosto 25, 2024

Resumen: En esta investigación se desarrolla la propuesta para la creación del Sistema de Inventario para el refaccionamiento de Peaje y de Sistemas Inteligentes de Tráfico (ITS). La problemática que originó este proyecto se resume como la deficiencia en el proceso del control del inventario de refaccionamiento en la caseta de cobro Amado Nervo, ubicada en la Autopista Jala-Compostela en el kilómetro 31+620. Lo anterior impedía llevar un seguimiento adecuado de las piezas y componentes utilizados en el mantenimiento de los equipos. La metodología que se siguió para el desarrollo del sistema web fue la incremental. Como resultado, se expone el desarrollo del citado sistema web con un diseño intuitivo y un alto grado de usabilidad, mejorando la eficiencia en la gestión del inventario de piezas para el mantenimiento. La conclusión del estudio es que la creación de este sistema representa un beneficio inmediato tanto para los usuarios de la autopista como los trabajadores de la caseta al automatizar mediante la plataforma la gestión del inventario para el refaccionamiento.

Palabras clave: *Sistema de Inventario, Sistemas Inteligentes de Tráfico, Refaccionamiento de Peaje, Sistema Web.*

Abstract: This research develops the proposal for the creation of the Inventory System for the repair of Tolls and Intelligent Traffic Systems (ITS). The problem that originated this project is summarized as the deficiency in the process of controlling the repair inventory at the Amado Nervo toll booth, located on the Jala-Compostela Highway at kilometer 31+620. This prevented proper monitoring of the parts and components used in the maintenance of the equipment. The methodology followed for the development of the web system was incremental. As a result, the development of the aforementioned web system is presented with an intuitive design and a high degree of usability, improving the efficiency in the management of the parts inventory for maintenance. The conclusion of the study is that the creation of this system represents an immediate benefit for both highway users and toll booth workers by automating the management of the repair inventory through the platform.

Keywords: *Inventory System, Intelligent Traffic Systems, Toll Retrofitting, Web System.*

1. Introducción

La Sociedad Ibérica de Construcciones Eléctricas, S.A. (SICE) se erige como una empresa multinacional líder en la integración de tecnologías, abarcando diversos sectores como Tráfico y Transporte, Medioambiente y Energía, Telecomunicaciones y procesos industriales. Su enfoque se centra en la prestación de servicios de valor añadido, mediante la integración de tecnologías y sistemas propios y de terceros, ofreciendo soluciones personalizadas que destacan por su capacidad tecnológica, experiencia, integraciones de sistemas y sistemas abiertos [1].

Adicionalmente, como complemento a su actividad industrial, SICE destaca en el desarrollo de aplicaciones y sistemas dentro de las Tecnologías de la Información, contribuyendo a la gestión, explotación y control de equipamientos e instalaciones implementados por la empresa, entre los servicios que ofrecen son los de Sistemas Inteligentes de Transporte (ITS) [2], es dentro de este rubro de servicios que se ha desarrollado el proyecto que se presenta.

Es en este contexto de trabajo con la empresa ya descrita, que se identificó una problemática en la caseta de cobro Amado Nervo, ubicada en la Autopista Jala-Compostela en el kilómetro 31+620, cerca de San Pedro Lagunillas, Nayarit, México, que forma parte de la autopista de cuota que conecta Guadalajara con Puerto Vallarta. En el establecimiento se encontró una deficiencia en el proceso del control del inventario de refaccionamiento (dicho inventario se integra por los componentes disponibles para mantener la continuidad operativa del tramo carretero). Esta situación surgió a raíz de que el control de dicho inventario se realizaba de manera manual, mediante anotaciones escuetas en un cuaderno y en ciertos formatos que a menudo eran extraviados, maltratados y desorganizados. Lo anterior impedía llevar un seguimiento adecuado de las piezas y componentes utilizados en el mantenimiento de los equipos de peaje. Se observó que factores como la falta de una base de datos y la dependencia de los métodos manuales contribuyeron a la generación de esta problemática.

La afectación de esta carencia se reflejaba en retrasos en la atención de los mantenimientos, dificultaba la identificación y reposición oportuna de las piezas agotadas, y en última instancia, impactaba en el funcionamiento de los equipos de peaje y sistemas inteligentes de tráfico. Lo anterior afectó directamente al personal encargado de la gestión de inventario, así como a los usuarios de los equipos, quienes experimentaron interrupciones no planificadas en el servicio.

Para abordar la falta de control del inventario de refaccionamiento, en el ámbito de gestión de equipos de peaje y de Sistemas Inteligentes de Tráfico (ITS), se presentó la propuesta que consistió en implementar una solución integral que facilite el acceso y la administración de una base de datos para almacenar los registros de las piezas y componentes utilizados en el mantenimiento de los equipos, para realizar un seguimiento preciso de los movimientos de inventario.

Debido a lo anterior, el objetivo del proyecto fue desarrollar un sistema web para el control del inventario de refaccionamiento de los equipos de peaje e ITS, para optimizar la gestión de piezas, componentes y servicios. El sitio web intuitivo con un control de accesos y seguridad, con interfaz amigable permitirá buscar y gestionar el inventario desde diferentes dispositivos, mejorando la eficiencia en la gestión de inventario, reduciendo los retrasos en la atención de mantenimientos, facilitando la reposición de piezas agotadas y beneficiando tanto al personal responsable como a los usuarios de los equipos.

2. Estado del arte

Desde mucho tiempo atrás los datos y la información se consideran recursos valiosos para las organizaciones tanto para sus procesos internos y su interrelación con el entorno. Con la tecnología actual de recopilación y almacenamiento de datos, la capacidad de absorber la información útil oculta detrás de los datos se hace cada vez más importante [3]. Derivado de lo anterior la calidad de los datos y el procesamiento de información será la base para la toma de decisiones científicas y el desarrollo de proyectos de intercambio de información, evitando problemas para la gerencia empresarial [4].

Se estima que el 80% de todos los datos que se producen en la actualidad no están estructurados, lo que indica la importancia de la generación de conocimiento mediante criterios de integridad y precisión [5]. Para consolidar la comprensión y medición de la calidad, las empresas incorporan diversas técnicas y herramientas tecnológicas que garantizan la calidad y estabilidad de los datos a través de un sistema de gestión de información computarizado [6].

La medición de la calidad de los datos y la información tiene diversas dimensiones. Por ejemplo, Panahy *et al.* [7] señalan la relevancia de tener los datos correctos, en el formato correcto, en el lugar correcto, en el momento correcto. La medición de la calidad permite hallar diferencias entre diversos grupos de datos, a partir de procedimientos y prácticas del marco utilizado. Con lo anterior, se concluye que el resultado de la calidad de los datos y la información afecta los beneficios o costos dentro de cualquier organización [8].

En otro orden de ideas, según Feinberg y Kraemer [9] el inventario es una parte indispensable de las actividades diarias de la empresa. Los productos en venta, los productos en producción y los materiales consumidos en la producción de materiales pertenecen a la categoría de inventario. Este inventario representa uno de los activos circulantes importantes. Su existencia representa cierta cantidad de liquidez, su utilización y la forma de su gestión representa la relación entre la eficiencia de la operación de los activos de la empresa y el nivel de consumo de capital [10] [11].

Correa-Parra y Olea-Miranda [12] coinciden con lo anterior y afirman que el inventario representa el mayor activo en las firmas, los mismos autores agregan que la eficiencia en su control y gestión son un punto clave para el desempeño y estabilidad de una organización. “Controlar permite abastecer de manera oportuna la demanda de los clientes en tiempo y forma; mientras que gestionar, asigna y distribuye los productos en áreas cercanas y de fácil acceso, las cuales habiliten un mejor flujo de materiales dentro de almacén y faciliten el monitoreo físico” [12].

La gestión de inventarios forma parte importante de la gestión de la cadena de suministro, impactando significativamente en la eficiencia operativa y el control de costos de las empresas, este último aspecto es relevante en la gestión financiera empresarial, que está directamente relacionada con la rentabilidad y la competitividad a largo plazo de las empresas [13]. “El manejo del inventario garantizará el buen funcionamiento y el control de toda organización por lo tanto toda aquella que se realice una mala gestión en los mismos obtendrá información equivocada sobre su estado de económico y en consecuencia no podrá tomar decisiones eficientes” [14].

3. Materiales y métodos

Los materiales y métodos deben describirse con detalles suficientes para que otros puedan replicar y desarrollar los resultados publicados. Tenga en cuenta que la publicación de su manuscrito implica que debe poner a disposición de los lectores todos los materiales, datos, códigos de computadora y protocolos asociados con la publicación (actualmente todas las revistas y congresos de investigación son flexibles en este rubro). Por favor, revele en la etapa de envío cualquier restricción sobre la disponibilidad de materiales o información. Los nuevos métodos y protocolos deben describirse en detalle, mientras que los métodos bien establecidos pueden describirse brevemente y citarse adecuadamente.

Los estudios de intervención que involucran animales o seres humanos, y otros estudios que requieren aprobación ética, deben incluir la autoridad que otorgó la aprobación y el código de aprobación ético correspondiente.

Para el diseño y codificación del sistema propuesto se implementó una metodología de desarrollo incremental, que incluye cinco actividades estructurales genéricas: comunicación, planeación, modelado, construcción y despliegue. Éstas se ejecutan a través de cierto número de repeticiones del proyecto. Cada iteración produce un incremento del software que da a los participantes un subconjunto de características y funcionalidad generales del software. Conforme se produce cada incremento, el software se hace más y más completo [15].

3.1. Fases de la metodología incremental

Siguiendo con Pressman [15], enseguida se presenta el desarrollo de cada una de las fases efectuadas dentro de la metodología incremental:

3.1.1. Comunicación

Antes de comenzar cualquier trabajo técnico, tiene importancia crítica comunicarse y colaborar con el cliente (y con otros participantes). Se busca entender los objetivos y necesidades de los participantes respecto del proyecto, y reunir los requerimientos que ayuden a definir las características y funciones del software. En el proceso incremental el primer paso que se ejecutó fueron una serie de entrevistas en donde se logró una comunicación efectiva con todas las partes interesadas, especialmente con los usuarios finales. Esta etapa inicial implicó la recopilación de los requisitos del proyecto, que sirvieron como base para el desarrollo posterior del software.

3.1.2. Planeación

La actividad de planeación permite crear un “mapa” que guía al equipo mientras viaja. El mapa —llamado plan del proyecto de software— define el trabajo de ingeniería de software al describir las tareas técnicas por realizar, los riesgos probables, los recursos que se requieren, los productos del trabajo que se obtendrán y una programación de las actividades.

Una vez que se establecieron los requisitos iniciales, se procedió a planificar las actividades técnicas necesarias para alcanzar los objetivos del proyecto. Esta fase implicó la identificación de riesgos potenciales, la asignación de recursos adecuados y la programación de las actividades en un calendario estructurado. Como se muestra en la imagen de planeación rápida de la Tabla 1.

Tabla 1. Planeación rápida.

Actividad	Fecha
Adecuación de equipo y herramientas técnicas para el desarrollo	25/01/2024
Selección de herramientas técnicas	30/01/2024
Diseño del sistema (interfaces y base de datos)	05/02/2024
Codificación y creación de base de datos	09/02/2024
Diseño visual con CSS	27/02/2024
Desarrollo de documentos PHP para la gestión de la base de datos	05/03/2024
Pruebas	13/03/2024
Entrega previa al usuario final	18/03/2024
Modificaciones y correcciones de acuerdo con las observaciones del cliente	25/03/2024
Entrega final	29/03/2024

Fuente: Creación propia.

3.1.3. Modelado

En esta fase se crea un “bosquejo” del objeto, a fin de entender el panorama general —cómo se verá arquitectónicamente, cómo ajustan entre sí las partes constituyentes y muchas características más—. Si se requiere, refina el bosquejo con más y más detalles en un esfuerzo por comprender mejor el problema y cómo resolverlo. Un ingeniero de software hace lo mismo al crear modelos a fin de entender mejor los requerimientos del software y el diseño que los satisfará.

Durante esta fase, se desarrollaron los modelos que sirvieron como representaciones visuales de las funcionalidades del sistema, esto proporcionó una guía clara para el desarrollo posterior, se realizó un análisis profundo para obtener a grandes rasgos los datos a utilizar. La Figura 1 y Figura 2 presentan algunos de los modelos creados para cada pantalla.

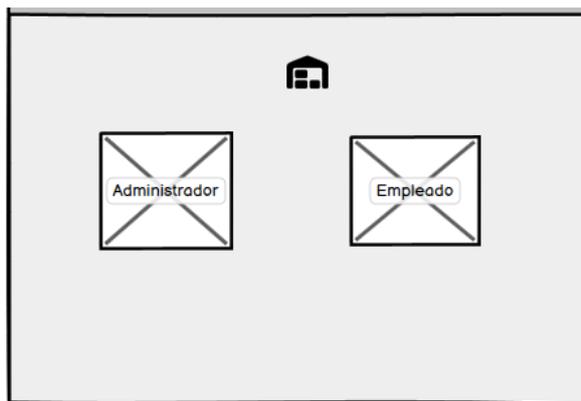


Figura 1. Mockup de pantalla de inicio. Fuente: Creación propia.

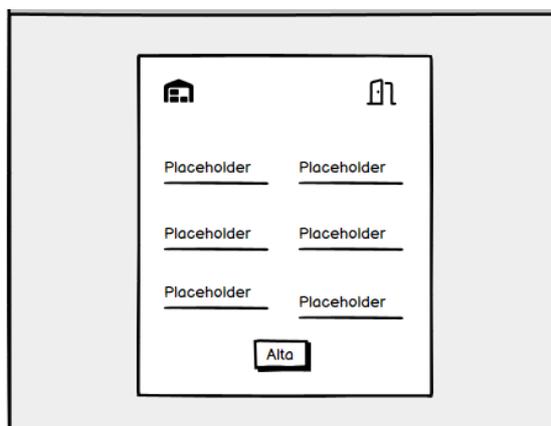


Figura 2. Mockup de pantalla para alta. Fuente: Creación propia.

3.1.4. Modelado de la base de datos

Se llevó a cabo el proceso de transformar (o mapear) el esquema conceptual del dominio de aplicación en el esquema para el modelo de datos subyacente a un Gestor de Base de Datos particular, así como el modelo de datos relacional u orientado a objetos. Este mapeo puede entenderse como el resultado de lograr dos distintos objetivos: (1) preservar la capacidad de capturar y distinguir todos los estados válidos del esquema conceptual; (2) abordar cuestiones relacionadas con la facilidad y el costo de consultar el esquema lógico, así como los costos de almacenamiento y mantenimiento de restricciones. Esta entrada se centró en el mapeo de diagramas entidad-relación (EER) (extendidos) a bases de datos relacionales. Se muestra en la Figura 3 el análisis de la información de la base de datos.

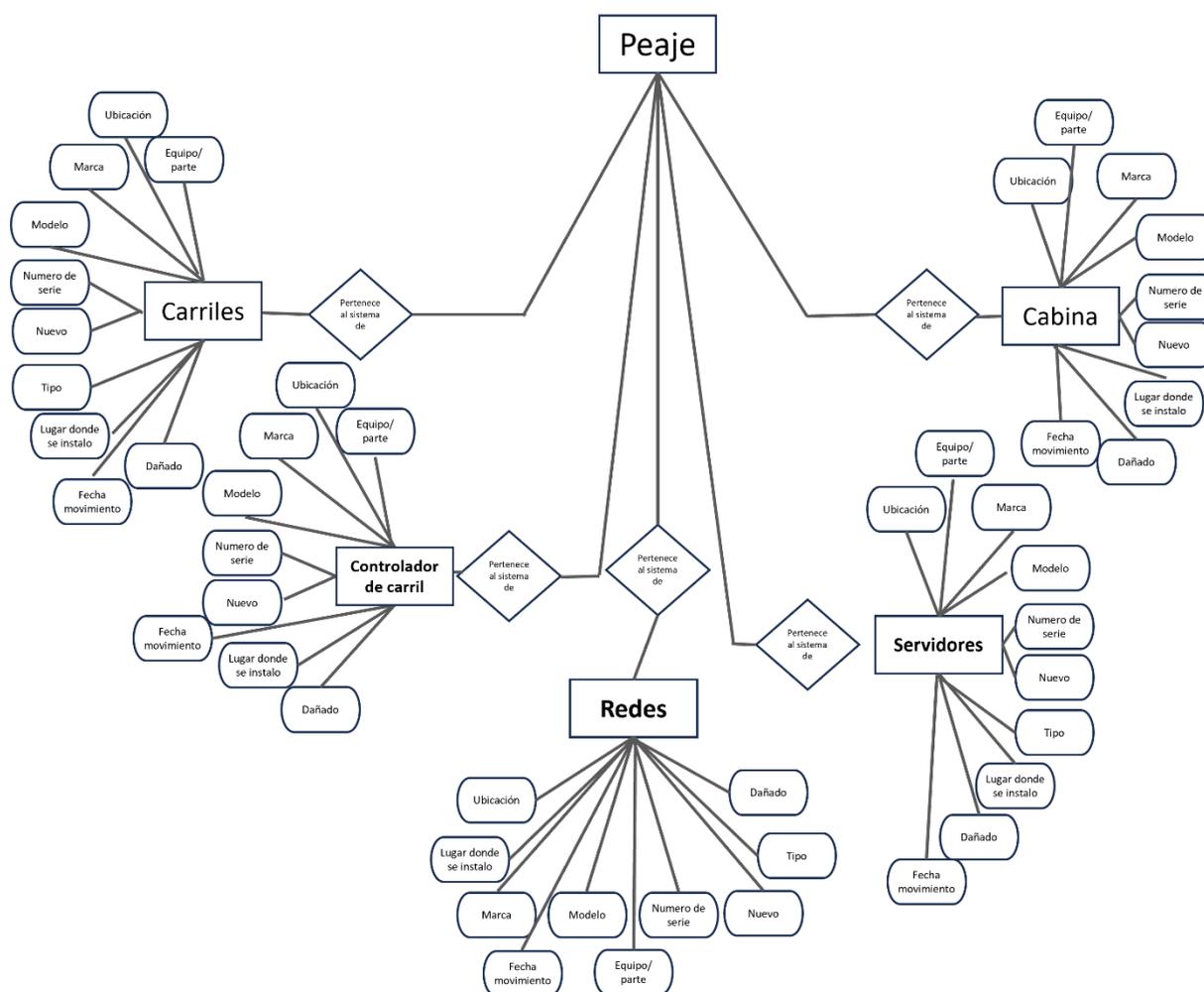


Figura 3. Diagrama de la base de datos. Fuente: Creación propia.

3.1.5. Construcción

Esta actividad combina la generación de código (ya sea manual o automatizada) y las pruebas que se requieren para descubrir errores en este. En esta fase se inició el desarrollo del software, donde se generó el código y se llevaron a cabo pruebas para garantizar la calidad y la funcionalidad del producto. Esta etapa implica un enfoque iterativo, donde se implementaron incrementos del software a medida que se completaba.

3.2. Código

Se configuró el editor de texto, y se empleó MySQLworbench para el diseño de las entidades relacionales en la creación de la base de datos, la cual se muestra en la Figura 4.

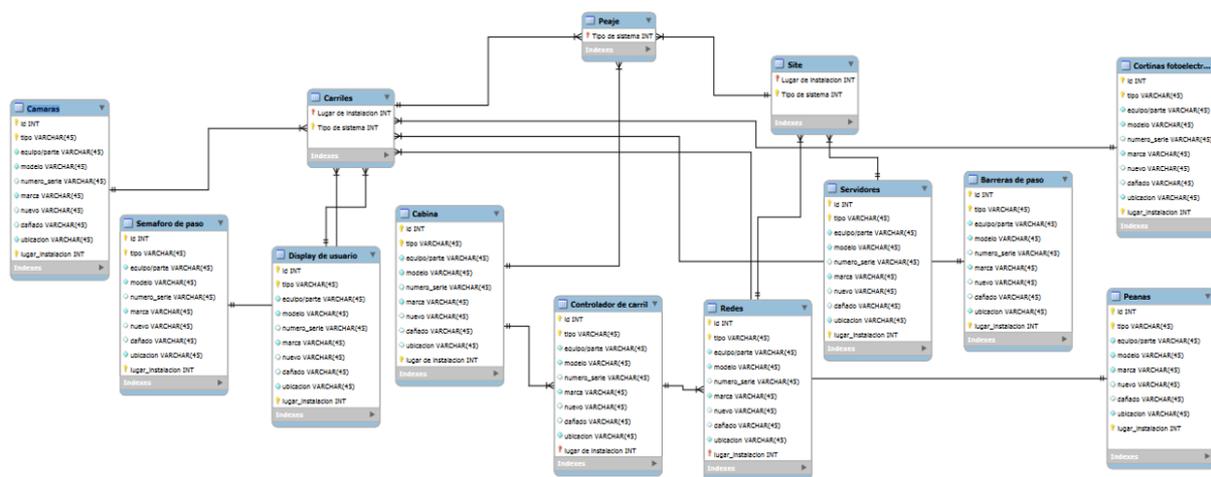


Figura 4. Modelo Entidad - relación para la base de datos. Fuente: Creación propia.

Se ejecutó el *script* en el gestor de Mysql worbench para la creación de la base de datos, sus tablas y los atributos que se definieron con anterioridad. Cabe añadir que se utilizó este manejador de Base de Datos por la facilidad de uso y también su gratuidad. Se muestra en la Figura 5 la base de datos junto con sus tablas en Mysql worbench.

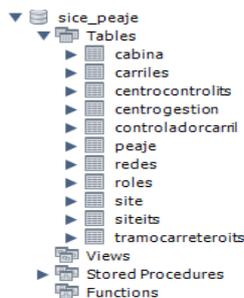


Figura 5. Base de datos con sus tablas en Mysql Worbench. Fuente: Creación propia.

Se usó el editor de texto visual Studio Code para la creación de carpetas dentro del mismo proyecto con todos los archivos a utilizar, para tener una mejor perspectiva y organización en el proceso de creación de este. Se incluyen algunas muestras de los documentos con las carpetas y archivos en las Figuras 6 y 7.

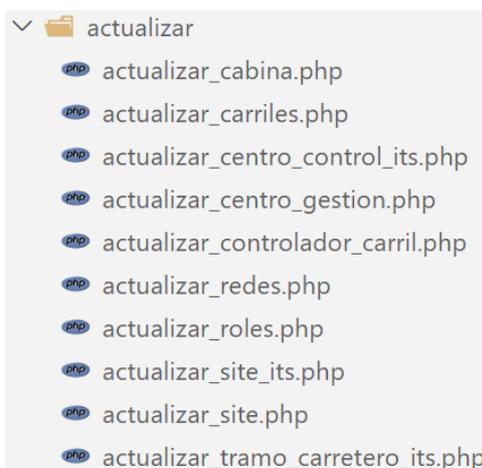


Figura 6. Carpeta de actualizar. Fuente: Creación propia.

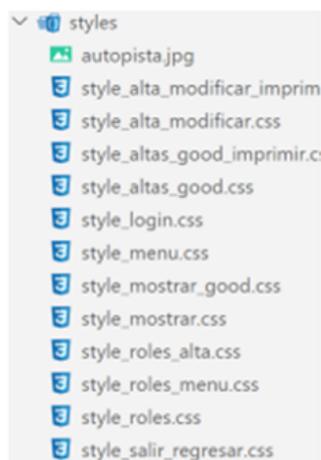


Figura 7. Carpeta de estilos CSS. Fuente: Creación propia.

Empleando diferentes hojas de estilos CSS se dio formato a las páginas, de acuerdo con el diseño de los prototipos, para contar con una interfaz agradable y con buen grado de usabilidad. En cuanto a las opciones que se programaron en el sistema, en la opción de actualizar se pueden observar los datos en tiempo real de cada una de las tablas que se han llenado previamente. En la parte de altas se registran todos los campos que se requiera llenar para cada una de las tablas, ya que al momento de dar alta el archivo, la petición se llevará a cabo. El apartado de eliminar permite borrar el campo que se eligió para que no aparezca en las tablas. En mostrar se realizó una estructura para visualizar el esqueleto de todas las tablas, se agregó la interfaz y con la ayuda de load se extrae el contenido de la base de datos.

En la opción de Menú se muestra el mismo, para establecer el mecanismo de navegación entre cada una de las páginas que conforman el sitio. De igual manera se programó la opción para imprimir la hoja con los datos de mayor importancia para el control del inventario, posibilitando tener un mejor control en el manejo de las piezas. Adicionalmente, con el lenguaje JavaScript se obtuvieron datos más legibles, más seguridad en cuanto a la inactividad y en general la automatización del manejo del inventario. De igual manera mediante la opción de editar los productos en tiempo real, se pueden realizar cambios en el estatus y datos de cada uno de los productos registrados en el inventario.

3.3. Despliegue

El software (como entidad completa o como un incremento parcialmente terminado) se entrega al consumidor que lo evalúa y ofrece una retroalimentación, misma que se basa en dicha evaluación [15].

El software desarrollado se entregó al cliente para su evaluación y retroalimentación. Este proceso de despliegue puede ser continuo a lo largo del proyecto, permitiendo ajustes y mejoras basados en las sugerencias del cliente. Se les presentó funcionando a los trabajadores. Como primera observación, solicitaron que todo lo que se diera de alta se tenía que convertir a un formato de hoja de cálculo, (se corrigió en ese instante). La segunda observación que se sugirió fue que existiera la forma de darle actualizaciones en el futuro, para posibles modificaciones del sistema. Se muestran en las Figuras 8, 9, 10 y 11 algunas de las adecuaciones realizadas durante esta fase del desarrollo.

El CSS se utilizó para darle formato a las páginas, de acuerdo con el diseño previo. En la parte de altas se creó el código para registrar correctamente todos los campos que se requirieron para cada una de las tablas (Figura 8). En la Figura 9 se observa el código generado para la opción de actualizar registro, en el caso del archivo cabina. En la Figura 10 se observa el código creado para la gestión de la información de los carriles, utilizando PHP. Finalmente, en la Figura 11 se observa el código del esqueleto con diseño incluyendo los campos correspondientes al alta de una cabina, que durante la opción de insertar procede al uso de altas y luego manda llamar a las opciones codificadas previamente: mostrar y load.

```

body{
background: linear-gradient(■rgb(0, 0, 0, 0.5), ■rgba(0, 0, 0, 0.2)), url(img_peaje/autopista.jpg);
background-size: cover;
background-attachment: fixed;
background-position: center;
height: 500px;
width: auto;
}
.abc{
display: flex;
justify-content: center;
margin-top: 70px;
}
.container{
position: relative;
max-width: 500px;
width: 90%;
background: ■#fcedda;
padding: 25px;
border-radius: 8px;
box-shadow: 0 0 15px ■rgb(0, 0, 0, 0.1);
}
.container header{
font-size: 1.2rem;
color: ■black;
font-weight: 600;
text-align: center;
}
.container .form{
margin-top: 15px;
}
.form .input-box{
width: 100%;
margin-top: 10px;
}
.form :where(.input-box input, .select-box){
position: relative;
height: 35px;
width: 93%;
outline: none;
font-size: 1rem;
color: ■#000000;
margin-top: 5px;
border: 1px solid ■#ee4e34;
border-radius: 6px;
padding: 0 15px;
background: ■#fcedda;
}
    
```

Figura 8. CSS del archivo alta_modificar.css. Fuente: Creación propia.

```

include('../conexion.php');
// Verificar si se han enviado datos por el formulario
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
// Obtener los datos del formulario
$equipo_parte = $_POST['equipo_parte'];
$Marca = $_POST['Marca'];
$Modelo = $_POST['Modelo'];
$NO_Serie = $_POST['NO_Serie'];
$Tipo = $_POST['Tipo'];
$Ubicacion = $_POST['Ubicacion'];
$Instalacion = $_POST['Instalacion'];
$Fecha_Movimiento = $_POST['Fecha_Movimiento'];
$Nuevo = $_POST['Nuevo'];
$Danado = $_POST['danado'];
$ID = $_POST['ID'];
$Peaje_tipo_de_sistema = $_POST['Peaje_tipo_de_sistema'];
$usuario = $_POST['usuarioupdate'];
$hora = $_POST['horaupdate'];
// Consulta SQL para actualizar la información en la base de datos
$sql = "UPDATE cabina SET
    equipo_parte = '$equipo_parte',
    Marca = '$Marca',
    Modelo = '$Modelo',
    Tipo = '$Tipo',
    Ubicacion = '$Ubicacion',
    Instalacion = '$Instalacion',
    Fecha_Movimiento = '$Fecha_Movimiento',
    Nuevo = '$Nuevo',
    danado = '$Danado',
    ID = '$ID',
    Peaje_tipo_de_sistema = '$Peaje_tipo_de_sistema',
    usuarioupdate = '$usuario',
    horaupdate = '$hora'
    WHERE NO_Serie = '$NO_Serie'";
// Ejecutar la consulta
if ($conn->query($sql) === TRUE) {
    echo "Registro exitoso";
    echo "<script>alert('Actualización exitosa.');

```

Figura 9. Codificación de la opción actualizar para el archivo cabina. Fuente: Creación propia.

```

include('../conexion.php');

// Verifica si el formulario ha sido enviado
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    // Obtiene los datos del formulario
    $equipo_parte = $_POST['equipo_parte'];
    $marca = $_POST['marca'];
    $modelo = $_POST['modelo'];
    $no_serie = $_POST['no_serie'];
    $tipo = $_POST['tipo'];
    $ubicacion = $_POST['ubicacion'];
    $instalacion = $_POST['instalacion'];
    $fecha_movimiento = $_POST['fecha_movimiento'];
    $nuevo = $_POST['nuevo'];
    $danado = $_POST['danado'];
    $peaje_tipo_de_sistema = $_POST['peaje_tipo_de_sistema'];
    $id = $_POST['id'];
    $usuario = $_POST['usuario'];

    // Nuevos campos
    $usuarioupdate = isset($_POST['usuarioupdate']) ? $_POST['usuarioupdate'] : null;
    $horaupdate = isset($_POST['horaupdate']) ? $_POST['horaupdate'] : null;

    // Prepara la consulta SQL
    $consulta = "INSERT INTO carriles (equipo_parte, marca, modelo, no_serie, tipo, ubicacion, instalacion, fecha_movimiento, nuevo, danado, peaje_tipo_de_sistema, usuarioupdate, horaupdate) VALUES ('" . $equipo_parte . "', '" . $marca . "', '" . $modelo . "', '" . $no_serie . "', '" . $tipo . "', '" . $ubicacion . "', '" . $instalacion . "', '" . $fecha_movimiento . "', '" . $nuevo . "', '" . $danado . "', '" . $peaje_tipo_de_sistema . "', '" . $usuarioupdate . "', '" . $horaupdate . "')";

    // Ejecuta la consulta
    $resultado = mysqli_query($conn, $consulta) or die("Error de registro: " . mysqli_error($conn));

    // Verifica si la consulta fue exitosa
    if ($resultado) {
        echo "<script>
            alert('Registro exitoso');
            window.location.href = '../mostrar_carriles.php';
        </script>";
        exit();
    } else {
        echo "<script>
            alert('Error en el registro');
        </script>";
    }
}

```

Figura 10. Archivo Carriles.php. Fuente: Creación propia.

```

<body>
    <div class="abc">
        <section class="container">
            <h1 style="text-align: center;">Alta Cabina</h1>
            <form class="form" action="../alta_procesar/alta_procesar_cabina.php" method="post" autocomplete="off">
                <div class="column1">
                    <div class="input-box">
                        <label>Equipo/Parte</label>
                        <input type="text" placeholder="Ingrese los datos" name="equipo_parte">
                    </div>
                    <div class="input-box">
                        <label>Marca</label>
                        <input type="text" placeholder="Ingrese los datos" name="marca">
                    </div>
                    <div class="input-box">
                        <label>Modelo</label>
                        <input type="text" placeholder="Ingrese los datos" name="modelo">
                    </div>
                    <div class="input-box">
                        <label>Numero de Serie</label>
                        <input required type="text" placeholder="Ingrese los datos" name="no_serie">
                    </div>
                    <div class="column1">
                        <div class="small-field">
                            <div class="input-box">
                                <label>Tipo</label>
                                <input type="text" placeholder="Ingrese los datos" name="tipo">
                            </div>
                            <div class="select-section">
                                <div class="input-box">
                                    <label>Ubicación</label>
                                    <div class="select-box">
                                        <select name="ubicacion">
                                            <option>Bodega</option>
                                            <option>Plaza Amado Nervó</option>
                                            <option>Plaza Compostela</option>
                                        </select>
                                    </div>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </form>
        </div>
    </div>

```

Figura 11. Archivo Alta cabina.html modificado. Fuente: Creación propia.

4. Resultados

A continuación, se muestran como resultado las principales interfaces que componen al sistema web para el control de inventario para el Refaccionamiento de peaje y de Sistemas Inteligentes de Tráfico (ITS). En primera instancia se incluye la interfaz de acceso al sistema en donde se evidencian los perfiles de usuario con los que se ingresa al sistema; ya sea como empleado o como administrador. Esto en correspondencia a los roles existentes en la empresa. Como se muestra en la Figura 12.



Figura 12. Perfiles para ingresar al sistema. Fuente: Creación propia.

Una vez elegido alguno de estos dos roles, se desplegará el menú correspondiente en la siguiente pantalla, así mismo al ingreso se realiza la validación de contraseña para evitar accesos no autorizados. Lo anterior se muestra en la Figura 13.

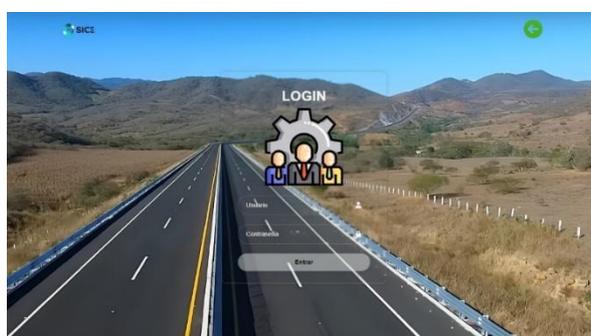


Figura 13. Acceso al sistema. Fuente: Creación propia.

El menú desplegable de empleados permite elegir alguna de las opciones siguientes: en Peaje son las de carriles, controlador de carril, redes, SITE, cabina y centro de gestión y en ITS son las de Tramo carretero, Centro de Control y SITE. Si la tabla está vacía se inicia con el primer registro y si ya tiene datos permite agregar nuevos y muestra los existentes. En la Figura 14 se aprecia el menú de empleados y su contenido correspondiente.



Figura 14. Menús empleados y la tabla Carriles. Fuente: Creación propia.

Para completar la inserción de un nuevo registro, luego de llenar los datos correspondientes será necesario volver a ingresar los datos del *login* del usuario en cuestión, con el propósito de conservar el registro del usuario que haya agregado estos datos para una posterior consulta en caso de que exista un error o una modificación. Como se muestra en la Figura 15 sobre la página de alta del producto en Carriles.



Formulario de Alta Carriles con los siguientes campos:

- Equipo/Parte: Computadora
- Marca: ASUS
- Modelo: TUF FX506LH HN004T
- Numero de Serie: xxxxxxxxxxxxxxxxx
- Tipo: Uso Rudo
- Ubicación: Plaza Amado Nervo
- Lugar Donde se Instalo: Se instalo en la caseta de amado nervo
- Fecha de Registro: 17/03/2024
- Nuevo: SI
- Dañado: NO
- Pertenece al sistema de peaje: Carriles
- ID: 1
- Usuario que dio de alta: Meza

Botón: Dar Alta

Figura 15. Alta producto. Fuente: Creación propia.

Al realizar un alta en la tabla, aparecerá un mensaje de registro exitoso. Esto se muestra en la Figura 16, alerta del registro.

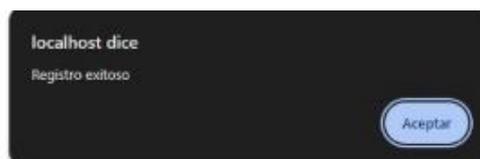


Figura 16. Alerta alta. Fuente: Creación propia.

Una vez llenando el registro, la página se redirecciona a la tabla que se editó con anterioridad, mostrando los datos previos y los nuevos. Esto se observa en la Figura 17 en donde el registro se efectuó correctamente.



Tabla de Carriles con los siguientes datos:

#	Equipo_Parte	Marca	Modelo	Numero_de_Serie	Tipo	Ubicación	Instalación	Nuevo	Dañado	ID	Tipo_de_sistema	Usuario_Alta	Fecha_de_Registro	Usuario_M
1	Computadora	ASUS	TUF FX506LH HN004T	xxxxxxxxxxxxxx	Uso Rudo	Plaza Amado Nervo	Se instalo en la caseta de amado nervo	SI	NO	1	Carriles	Meza	17/03/2024	

Mostrando 1 de 1 registros

Figura 17. Registro Exitoso. Fuente: Creación propia.

Para la edición de cualquier dato de una tabla será necesario utilizar la opción de editar. Este proceso se muestra en las Figuras 18 y 19.



Figura 18. Datos vacíos en actualización. Fuente: Creación propia.

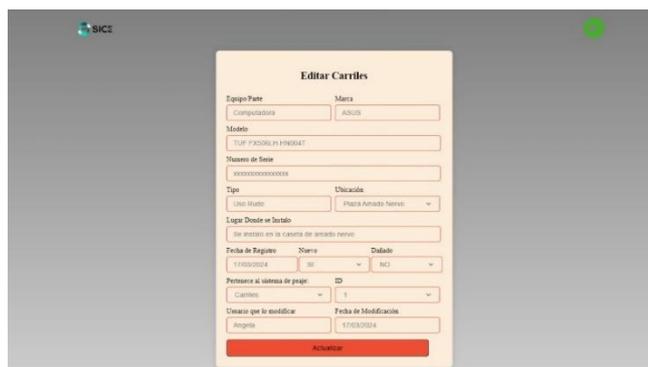


Figura 19. Editar campo. Fuente: Creación propia.

Si se desea buscar algún registro en específico, basta con escribir en el apartado de buscar alguna palabra que se asocie con dicho registro, ya que con ello la página mostrará los resultados obtenidos. Como se ve en la Figura 20.



Figura 20. Búsqueda de registro. Fuente: Creación propia.

Una opción muy valiosa para los usuarios es la de imprimir. Derivado de esta necesidad al final de cada tabla se agregó un botón con tal finalidad, el cual redirecciona a un nuevo formulario en el que se solicitan los parámetros para la impresión. Como se aprecia en las Figuras 21 y 22.



Figura 21. Página para imprimir. Fuente: Creación propia.

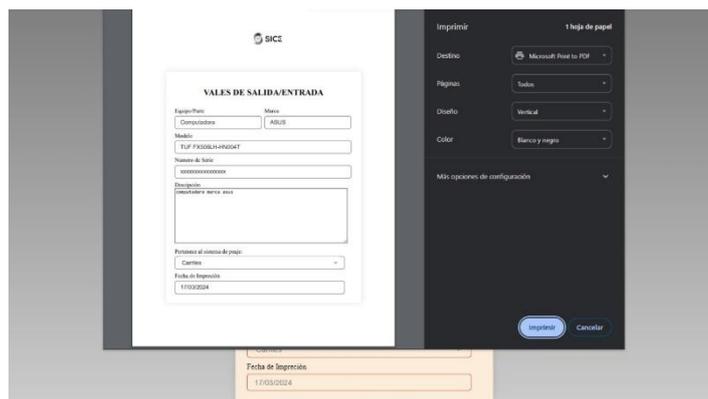


Figura 22. Configuración de opciones de impresión. Fuente: Creación propia.

De igual manera, otra opción para administrar los datos del inventario es simplemente descargar el archivo, donde previamente se modificaron las opciones de guardado. Lo anterior se observa en la Figura 23.

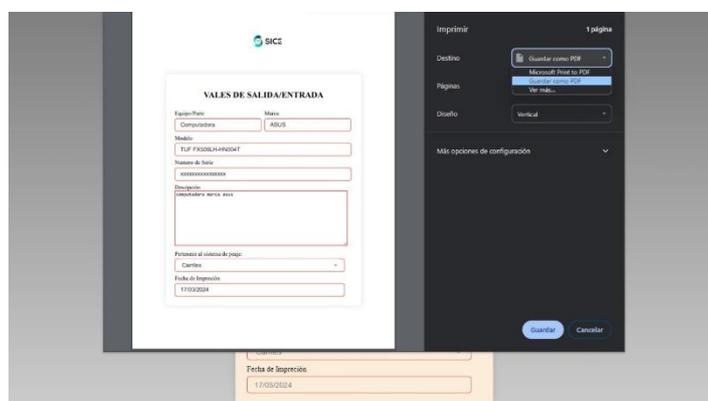


Figura 23. Página para descargar archivo correspondiente. Fuente: Creación propia.

Cada vez que se realice una actualización en el sistema (ya sea inserción, eliminación, o incluso modificación) dicho proceso se estará reflejando en tiempo real en un archivo con formato de Excel, recordando que esta fue una petición particular de los usuarios, con la finalidad de facilitarles la manipulación de los datos. Como se identifica en la Figura 24, actualización en tiempo real en Excel.

equipo parte	Marca	Modelo	NO_Serie	Tipo	Ubicación	Instalación	Fecha_Movimiento	Nuevo	danado	ID	Pea
Laptops	HP	Pavilion 15	xxxx00	Laptops	Tramos		17/03/2024	SI	NO	1	
Computadora	ASUS	TUF FX506LH-HN004T	xxxxxxxxxxxxxxxx	Uso Rudo	Plaza Amado Nervo	Se instalo en la caseta de amado nervo	17/03/2024	SI	NO	1	

Figura 24. Actualización en tiempo real de los datos en Excel. Fuente: Creación propia.

5. Conclusiones

En la actualidad, las plataformas web han revolucionado la manera en la que se interactúa con los datos, proporcionando medios poderosos para la optimización de recursos, la personalización de experiencias y la generación de conocimiento. En un mundo cada vez más digitalizado, aprovechar al máximo las potencialidades de esas plataformas se ha convertido en una necesidad imperativa para mantener la competitividad en cualquier campo, sin lugar a duda en el trabajo presentado en este documento se da cuenta de ello.

Al presentar los resultados del proyecto se puede concluir que se logró satisfactoriamente el objetivo definido para el mismo, ya que se realizó la creación de un sistema web para el control del inventario de refaccionamiento de los equipos de peaje e ITS. Al realizar las pruebas en el ciclo de desarrollo se observó un funcionamiento correcto, además de un alto grado de aceptación por parte de los usuarios. De igual manera el proceso de alta, baja, modificación y consulta de la información fue verificado para garantizar la integridad de los datos y la adecuada gestión de la Base de Datos.

Lo anterior permitirá a los usuarios de dicho sistema optimizar la gestión de piezas, componentes y servicios. El sitio web creado tiene un diseño intuitivo con un alto grado de usabilidad, con interfaces simples que posibilitan su manejo sin contratiempos. El acceso al sistema desde la web asegura la gestión del inventario mediante diferentes dispositivos en línea, mejorando la eficiencia y reduciendo los retrasos en la atención de mantenimientos, así mismo facilitando la reposición de piezas agotadas.

El beneficio para los usuarios finales (quienes transitan por ese tramo de la autopista) y para el personal responsable del uso y operación del sistema es evidente, coincidiendo con lo mencionado en la indagación teórica acerca de la importancia del adecuado manejo del registro del inventario para toda organización. Cabe añadir que la accesibilidad a los datos que se añadió con la sincronía hacia una hoja de Excel es una peculiar característica que facilita en gran medida la manipulación y consulta de los registros para todo el personal. Es preciso decir que, a lo largo de la ejecución de este proyecto, se utilizaron diversas herramientas digitales que la empresa ya tenía en uso, teniendo en cuenta también los tiempos en los que éstos eran empleados y los procesos de los que formaban parte.

7. Referencias

- [1] SICE. (2024)). *Sobre SICE*. <https://www.sice.com/sobre-sice/>
- [2] SICE Seguridad. (2024). *Grupo Empresarial*. <https://www.siceseguridad.com/grupo-empresarial/>
- [3] Pan, X., Zhang, M., Chen, X. (2018). *A method of quality improvement based on big quality warranty data analysis*. IEEE International Conference on Software Quality, Reliability and Security Companion. Lisbon, Portugal. <https://doi.org/10.1109/QRS-C.2018.00115>
- [4] Chen, B., Wang, B., Zheng, C., Hu, X. (2009). *Research and implementation of information quality improvement bing*. 4th International Conference on Cooperation and Promotion of Information Resources in Science and Technology. Beijing, China. <https://doi.org/10.1109/COINFO.2009.17>
- [5] Taleb, I., Serhani, M. A., Dssouli, R. (2019). *Big data quality assessment model for unstructured data*. 13th International Conference on Innovations in Information Technology (IIT). Al Ain, United Arab Emirates. <https://doi.org/10.1109/INNOVATIONS.2018.8605945>
- [6] Mandke, V. V., Nayar, M. K. (2004). Beyond quality: The information integrity imperative. *Total Quality Management and Business Excellence*, 15 (5-6), 645-654. <https://doi.org/10.1080/14783360410001680134>
- [7] Panahy, P. H. S., Sidi, F., Affendey, L. S., Jabar, M. A. (2014). *The impact of data quality dimensions on business process improvement*. 4th World Congress on Information and Communication Technologies, (WICT). Melaka, Malaysia. <https://doi.org/10.1109/WICT.2014.7077304>
- [8] Bowo, W. A., Suhanto, A., Naisuty, M., Ma'mun, S., Hidayanto, A. N., Habsari, I. C. (2019). *Data quality assessment: A case study of PT JAS using TDQM Framework*. 4th International Conference on Informatics and Computing (ICIC). Semarang, Indonesia. <https://doi.org/10.1109/ICIC47613.2019.8985896>
- [9] Feinberg, E., Kraemer, D. (2021). *Continuity of discounted values and the structure of optimal policies for periodic-review inventory systems with setup costs*. Naval Research Logistics (NRL).
- [10] Arian, M., Demir, S., Erkoç, M. (2023). Inventory management with advance supply contracts across multiple replenishment periods. *Asia-pacific journal of operational research*, 40 (3). <https://doi.org/10.1142/S0217595922500312>

- [11]Halkos, G. E., Kevork, I. S., Tziourtzioumis, C. N. (2018). On the existence of unique minimum for the cost function in a (q,r) inventory model with backorders. *Ima Journal of Management Mathematics*, 29 (1), 1-21. <https://doi.org/10.1093/imaman/dpw010>
- [12]Corella-Parra, L. M., Olea-Miranda, J. (2023). Desarrollo de un sistema de control de inventario para una empresa comercializadora de sistemas de riego. *Ingeniería Investigación y Tecnología*, 24 (1), 1-10. <https://doi.org/10.22201/ii.25940732e.2023.24.1.006>
- [13]Gutiérrez-Alcoba, A., Rossi, R., Martín-Barragan, B., Hendrix, E. M. T. (2017). A simple heuristic for perishable item inventory control under non-stationary stochastic demand. *International Journal of Production Research*, 55 (7-8), 1885-1897. <https://doi.org/10.1080/00207543.2016.1193248>
- [14]De la Torre Marín, D. (2022). *El Control de Inventarios en una Empresa Manufacturera y Comercializadora de Alimentos* [Tesis de Grado]. Universidad Peruana Unión, Lima, Perú. <https://repositorio.upeu.edu.pe/handle/20.500.12840/5651>
- [15]Pressman, R. S. (2010). *Ingeniería del software. Un Enfoque Práctico* (7ma. Ed.). McGraw Hill.