**UNIVERSIDAD NACIONAL DE INGENIERÍA**

**FACULTAD DE INGENIERIA INDUSTRIAL   
Y DE SISTEMAS**



**ESCUELA DE INGENIERIA DE SISTEMAS**

**Curso:**

**Programación orientada a Objetos**

**Proyecto:**

**SISTEMA DE CONTROL DE REPARACIÓN DE COMPUTADORAS**

**Autores:**

Carhuas Bastidas, Jesús Arturo (20194090E)

Saavedra Balarezo, Gabriel Alessandro (20220087B)

Ugarte Ramirez, Rubén (20232041B)

**Docente:**

Mag. Coronel Castillo, Eric Gustavo (orcid.org/0000-0003-0494-5629)

**LIMA – PERÚ**

**2024**

# ÍNDICE GENERAL

Pág.

DEDICATORIA 2

ÍNDICE GENERAL 3

RESUMEN 5

ABSTRACT 6

INTRODUCCIÓN 7

OBJETIVOS 9

Objetivo General 9

Objetivos Específicos 9

REQUERIMIENTOS 10

Requerimientos Funcionales 10

Requerimientos NO Funcionales 13

Alcances 14

Limites 15

ANALISIS DE REQUERIMIENTOS 16

Identificación de Actores 16

Casos de Uso 17

Diagrama de Casos de Uso 18

ANALISIS DE REQUERIMIENTOS 19

Identificación de Actores 19

Casos de Uso 19

Diagrama de Casos de Uso 19

CU01 – NOMBRE DEL CASO DE USO 20

Especificación del caso de uso 20

Prototipo 20

Diagrama de secuencia 20

Modelo de base de dato 21

Criterios de aceptación 21

MODELO DE BASE DE DATOS 22

Modelo Conceptual 22

Modelo Relacional 22

Diccionario de Datos 24

Tabla: CLIENTE 24

Tabla: COMPUTADORA 25

Tabla: TECNICO 25

Tabla: REGISTRO 26

Tabla: PAGO 26

Tabla: ESTADO 27

Tabla: TIPOITEM 27

Tabla: ITEM 27

Tabla: USOITEMS 27

IMPLEMENTACIÓN DE LA BASE DE DATOS 28

Creación de la Base de Datos 28

Creación de los objetos de la base de datos 28

Cargar Datos de Prueba 31

CONCLUSIONES 34

LECCIONES APRENDIDAS 35

BIBLIOGRAFIA 36

# RESUMEN

Este proyecto tiene como objetivo el desarrollo de un sistema de control para una tienda de reparación de computadoras, con el fin de gestionar eficientemente los trabajos de reparación, costos, técnicos responsables y las piezas utilizadas en cada tarea. La aplicación permite el seguimiento de las reparaciones de computadoras, desde la recepción del cliente hasta la entrega final del equipo reparado, incluyendo la gestión de pagos y adelantos. El sistema registra la información del cliente, los detalles de la computadora y el diagnóstico realizado por el técnico, actualizando constantemente el estado de la reparación. Además, se realiza un cálculo de los costos involucrados y se genera un recibo final con el detalle de los pagos. Para este desarrollo, se utilizó una base de datos relacional que integra varias tablas y relaciones, garantizando la integridad y consistencia de la información. La implementación de este sistema mejorará la eficiencia operativa de la tienda y proporcionará un control completo sobre los procesos de reparación y facturación.

**Palabras clave:** Sistema de control, reparación de computadoras, gestión de pagos, base de datos relacional, administración de talleres.

# ABSTRACT

This project aims to develop a control system for a computer repair shop to efficiently manage repair tasks, costs, responsible technicians, and parts used in each job. The application tracks computer repairs from the client's initial reception to the final delivery of the repaired equipment, including payment and advance management. The system registers customer information, computer details, and the technician's diagnosis, constantly updating the repair status. Furthermore, it calculates the involved costs and generates a final receipt with payment details. A relational database was used for this development, integrating multiple tables and relationships to ensure data integrity and consistency. The implementation of this system will improve the shop's operational efficiency and provide complete control over the repair and billing processes.

**Keywords:** Control system, computer repair, payment management, relational database, workshop administration.

# INTRODUCCIÓN

En la actualidad, muchas tiendas de reparación de computadoras enfrentan dificultades significativas para gestionar la información relacionada con las reparaciones, lo que puede llevar a errores en el registro de pagos, seguimientos incorrectos de reparaciones y una comunicación ineficiente entre los clientes, los técnicos y los propietarios. Este problema se ve acentuado por la falta de un sistema centralizado que permita el manejo de los datos de manera eficiente y sin riesgo de omisiones o inconsistencias. Por ello, se hace necesario contar con una solución tecnológica que facilite el control de los trabajos de reparación, minimizando los errores y optimizando los procesos.

Con el fin de abordar estos desafíos, se ha desarrollado un sistema de control que permite gestionar de manera detallada cada trabajo de reparación, desde el registro inicial del cliente y su computadora hasta el cálculo de los costos de reparación y la emisión de los recibos finales. Este sistema no solo facilita la asignación de técnicos y el seguimiento de las reparaciones, sino que también automatiza el cálculo de los costos y el manejo de pagos, asegurando una mayor precisión y eficiencia en la gestión.

Este sistema, basado en una base de datos relacional, integra diferentes módulos que permiten llevar un registro preciso de las reparaciones realizadas, los adelantos y pagos efectuados, y el inventario de piezas utilizadas. Además, permite el monitoreo en tiempo real del estado de las reparaciones y la notificación al cliente cuando su equipo está listo para ser recogido. De esta manera, se optimizan los tiempos de respuesta y se mejora la comunicación con los clientes.

Al implementar esta solución, se espera que la tienda de reparación pueda mejorar considerablemente su eficiencia operativa, reducir los márgenes de error en los procesos administrativos y ofrecer un servicio más ágil y transparente a sus clientes. Sin embargo, el sistema desarrollado no contempla funcionalidades avanzadas como la integración con sistemas de inventario complejos o la generación de informes de análisis detallados sobre las reparaciones, limitándose a cubrir las necesidades más inmediatas y fundamentales del taller.

El presente informe describe el proceso de desarrollo del sistema, desde la conceptualización hasta la implementación, detallando las características técnicas y funcionales que permiten una administración más efectiva de las reparaciones y los pagos, y cómo este sistema puede transformar la manera en que se gestionan los trabajos de reparación en las pequeñas tiendas de computadoras.

# OBJETIVOS

## Objetivo General

Proveer a los clientes un servicio integral de reparación que permita gestionar, registrar, y dar seguimiento a las reparaciones de equipos o productos de manera eficiente, asegurando la correcta asignación de técnicos, la actualización oportuna de inventarios, y la correcta gestión de pagos, todo a través de un sistema intuitivo y accesible que garantice la satisfacción del cliente mediante un proceso ágil y transparente.

## Objetivos Específicos

1. Gestión de cliente: El cliente proporciona datos personales para crear un nuevo cliente en el sistema (nombre, contacto, etc.), y el sistema responde con la confirmación del registro exitoso.
2. Gestión de computadora: El cliente ingresa información sobre una computadora (marca, modelo, características) a su nombre para registrarla en el sistema, y el sistema devuelve la computadora registrada con todos los detalles proporcionados.
3. Gestión de pagos: El cliente registra un adelanto de pago inicial para una transacción o servicio, y realiza un pago final después la culminación de la reparación, y el sistema confirma la transacción con el saldo actualizado.
4. Consultar Datos Generales: El cliente consulta los datos almacenados en el sistema sobre una categoría específica (por ejemplo, registros de pruebas), y el sistema devuelve la lista solicitada.
5. Registrar una Operación/Transacción: El cliente ingresa datos de un nuevo técnico (nombre, especialidad, etc.) y el sistema registra el técnico para futuras asignaciones de tareas, devolviendo la información registrada.
6. Registrar un Técnico: El cliente ingresa datos de un nuevo técnico (nombre, especialidad, etc.) y el sistema registra el técnico para futuras asignaciones de tareas, devolviendo la información registrada.
7. Registrar Uso de Ítems: El cliente especifica los ítems que se usan en una operación (producto, material, equipo), y el sistema registra el uso y actualiza el inventario y el registro de la operación.
8. Consultar Tipos de Ítems: El cliente consulta todos los tipos de ítems disponibles en el sistema (materiales, equipos), y el sistema devuelve una lista de los tipos disponibles.
9. Consultar Ítems por Tipo: El cliente consulta ítems específicos de un tipo (por ejemplo, computadoras, materiales de oficina) y el sistema devuelve una lista de ítems disponibles según el tipo solicitado.

# REQUERIMIENTOS

## Requerimientos Funcionales

#### **Gestión de Reparaciones**

1. **Registrar Reparación de Computadoras**:  
   El sistema debe permitir registrar cada trabajo de reparación con los siguientes detalles:
   * Información del cliente (nombre, contacto, etc.).
   * Detalles de la computadora (marca, modelo, número de serie).
   * Descripción del problema reportado por el cliente.
   * Acciones a realizar y piezas/materiales necesarios.
   * Costo estimado de mano de obra y materiales (si se tiene un presupuesto inicial).
2. **Actualizar Estado de la Reparación**:  
   El sistema debe permitir actualizar el estado de la reparación en tiempo real. Los posibles estados son:
   * **En proceso**: Reparación en marcha.
   * **Reparado**: El equipo está listo, pero aún no ha sido entregado.
   * **Listo para entregar**: La reparación ha sido completada y está lista para que el cliente la recoja.
   * **Finalizado**: La reparación ha alcanzado su fase final después de que el cliente haya efectuado el pago final y recogido su computadora.
3. **Cálculo de Costos**:  
   El sistema debe calcular automáticamente:
   * **Costo de mano de obra**: Basado en el tiempo estimado de reparación por el técnico.
   * **Costo de los ítems**: Basado en las piezas y materiales necesarios para la reparación.
   * **Costo total**: La suma de los costos de mano de obra y de los ítems.
   * **Saldo pendiente**: El sistema debe restar el adelanto realizado por el cliente del costo total para calcular el saldo pendiente a pagar.
4. **Gestión de Pagos**:  
   El sistema debe registrar los pagos parciales (adelanto) y el saldo pendiente del cliente, para que el cliente pueda pagar la diferencia al recoger la computadora.
5. **Generación de Comprobantes**:  
   El sistema debe permitir generar y entregar un comprobante de recepción al cliente al momento de entregar el equipo para su reparación, que incluya:
   * El adelanto recibido.
   * La fecha estimada de entrega.
6. **Generación de Recibo Final**:  
   Al momento de la entrega final del equipo, el sistema debe generar y entregar un recibo detallado con:
   * Costo total de la reparación.
   * Adelanto recibido.
   * Pago final realizado por el cliente.

#### **Gestión de Técnicos**

1. **Asignación de Técnicos**:  
   El sistema debe permitir asignar un técnico responsable para cada reparación, y registrar el diagnóstico y las acciones realizadas por el técnico.
2. **Cálculo de Mano de Obra**:  
   El sistema debe calcular el costo de mano de obra basado en el tiempo estimado para la reparación y la tarifa del técnico asignado.

#### **Gestión de Inventario**

1. **Registro de Piezas y Materiales**:  
   El sistema debe permitir registrar las piezas y materiales utilizados en las reparaciones, incluyendo:
   * Nombre del material/pieza.
   * Cantidad utilizada.
   * Costo por unidad.
2. **Actualización del Inventario**:  
   El sistema debe actualizar en tiempo real el inventario de piezas y materiales utilizados, reduciendo las cantidades de stock según se utilicen en las reparaciones.

#### **Gestión de Clientes**

1. **Registro de Clientes**:  
   El sistema debe permitir registrar nuevos clientes o actualizar los existentes. La información que se debe registrar incluye:
   * Nombre, dirección, contacto.
   * Historial de reparaciones previas.
   * Información de pago (adelanto y saldo pendiente).
2. **Historial de Reparaciones**:  
   El sistema debe permitir consultar el historial de reparaciones previas de un cliente, incluyendo los detalles de la computadora, el diagnóstico realizado, y los costos de mano de obra y materiales.

#### **Generación de Reportes**

1. **Reportes de Reparaciones**:  
   El sistema debe generar reportes mensuales o semanales sobre:
   * Número total de reparaciones realizadas.
   * Costo total de reparaciones por tipo (mano de obra, piezas/materiales).
   * Número de reparaciones por técnico.
   * Ingresos generados por reparaciones.
2. **Reportes de Inventario**:  
   El sistema debe generar reportes de inventario mostrando:
   * Piezas/materiales disponibles.
   * Piezas/materiales agotados o con bajo stock.

## Requerimientos NO Funcionales

#### **Rendimiento**

1. El sistema debe ser capaz de manejar al menos 100 registros de reparaciones simultáneamente sin afectar el rendimiento.
2. El sistema debe tener tiempos de respuesta inferiores a 2 segundos al consultar o actualizar información en la base de datos.

#### **Seguridad**

1. El sistema debe garantizar la seguridad de la información sensible (datos del cliente, historial de reparaciones) mediante almacenamiento cifrado.
2. El acceso al sistema debe ser autenticado mediante usuario y contraseña.
3. El sistema debe permitir definir roles de usuario (administrador, técnico, cliente) con permisos específicos para cada uno.

#### **Usabilidad**

1. El sistema debe tener una interfaz fácil de usar, con acceso rápido a funciones clave como la gestión de reparaciones, clientes, inventario y reportes.
2. El sistema debe ser compatible con sistemas operativos Windows y Linux, y debe estar optimizado para dispositivos de escritorio.

#### **Confiabilidad**

1. El sistema debe estar disponible al menos el 99.9% del tiempo durante las horas laborales.
2. En caso de fallo, el sistema debe ser capaz de restaurar los datos de forma automática en menos de 5 minutos.

#### **Escalabilidad**

1. El sistema debe ser capaz de manejar un crecimiento en el número de clientes, reparaciones y técnicos sin comprometer el rendimiento.
2. Debe permitir la integración futura con otros sistemas (por ejemplo, sistemas de pago o gestión de ventas de repuestos) sin necesidad de una reestructuración mayor.

## Alcances

El alcance de este sistema de gestión para la reparación de computadoras abarca las siguientes funcionalidades clave para la optimización de la administración de los servicios y la información en el taller de reparaciones:

1. **Gestión de Reparaciones:**
   * El sistema permitirá registrar y hacer un seguimiento completo de cada trabajo de reparación realizado, incluyendo detalles como el tipo de reparación, la descripción del problema, los pasos seguidos y la fecha de inicio y finalización de cada reparación.
   * Se incluirán alertas y notificaciones para asegurar el cumplimiento de los tiempos estimados de reparación y garantizar que no se olvide ninguna reparación pendiente.
2. **Gestión de Costo de Mano de Obra:**
   * El sistema permitirá calcular y registrar el costo de mano de obra para cada reparación, basado en tarifas predeterminadas por hora o por tipo de servicio.
   * Se incluirán herramientas de cálculo automático para los costos asociados con el tiempo empleado en cada reparación.
3. **Seguimiento de Técnicos:**
   * Se llevará un registro detallado de los técnicos responsables de cada reparación, incluyendo su nombre, especialidad y el historial de trabajos realizados.
   * El sistema permitirá asignar reparaciones a técnicos según su disponibilidad y su nivel de especialización.
4. **Gestión de Costo Total de Reparación:**
   * El sistema calculará el costo total de cada reparación, incluyendo los ítems utilizados como piezas, componentes y materiales.
   * Se generarán informes detallados con los costos desglosados para facilitar el análisis financiero.
5. **Generación de Reportes y Estadísticas:**
   * El sistema ofrecerá la capacidad de generar reportes mensuales, semanales o por período específico sobre el número de reparaciones realizadas, ingresos generados y eficiencia de los técnicos.
   * Se podrán visualizar estadísticas sobre los costos de mano de obra, materiales y la rentabilidad de cada reparación.
6. **Gestión de Inventario:**

El sistema gestionará el inventario de piezas, componentes y materiales utilizados en las reparaciones, permitiendo registrar los ítems necesarios, las existencias disponibles y los reabastecimientos.

## Limites

Aunque el sistema tiene como objetivo cubrir múltiples aspectos de la gestión de reparaciones, existen algunas limitaciones inherentes que deben considerarse:

1. **Límites de Pagos:** El sistema establecerá límites máximos de dos pagos que seran solamente el pago de adelanto y el pago final.
2. **Límites Técnico:**
3. El sistema impondrá que si el técnico se encuentra realizando una reparación, se encontrara temporalmente ocupado.
4. **Límites de Intereses en Préstamos:**

El sistema establecerá límites en la tabla de registro. Enfocándose en un solo proceso, el usuario ni la computadora pueden ser derivados a otro técnico.

1. **Límites de días de duración estimada:**

El sistema establecerá límites de 15 días como máximo para que el técnico pueda ingresar la fecha real y el precio real.

1. **Límites de IGV.**

El sistema se basa en el IGV peruano así que está destinado a usar el 18% del Monto. Par así calcular su total.

# ANALISIS DE REQUERIMIENTOS

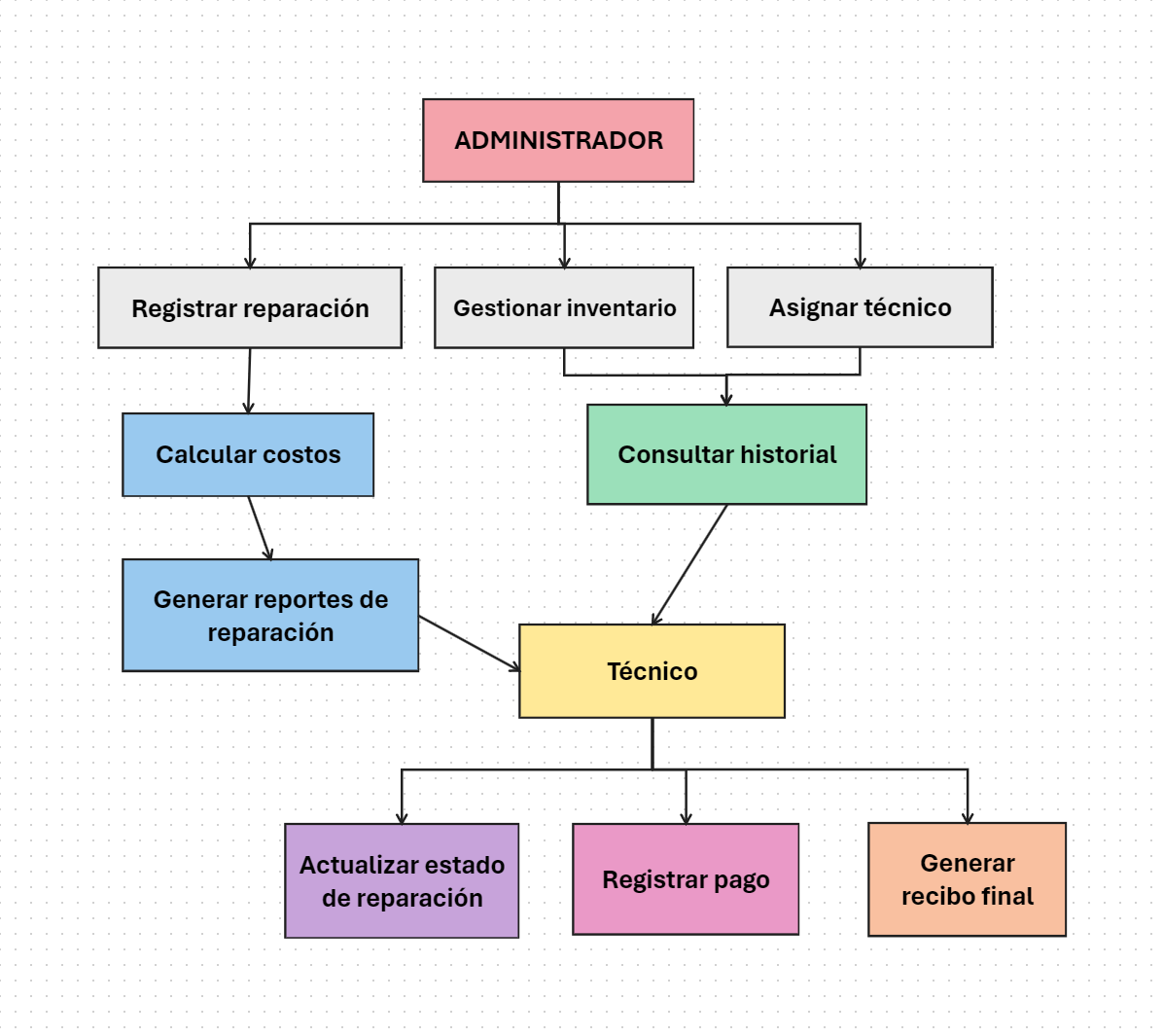
## Identificación de Actores

|  |  |  |
| --- | --- | --- |
| **CÓDIGO** | **NOMBRE** | **DESCRIPCIÓN** |
| **ACT01** | Cliente | Persona que solicita una reparación, realiza pagos y recibe el equipo reparado. |
| **ACT02** | Técnico | Persona que realiza las reparaciones de computadoras y registra información sobre las reparaciones realizadas. |
| **ACT03** | **Administrador** | Persona encargada de gestionar el sistema, roles de usuario, inventario, pagos, y generar reportes. |
| **ACT04** | **Sistema** | Entidad automatizada que gestiona el inventario, cálculos de costos, actualización de estados de reparación, y la generación de reportes. |

## Casos de Uso

|  |  |  |  |
| --- | --- | --- | --- |
| **CÓDIGO** | **ACTOR** | **NOMBRE** | **DESCRIPCIÓN** |
| **CU01** | |  | | --- | |  |  |  | | --- | | Administrador, Cliente | |  | El sistema permite registrar una nueva reparación, incluyendo información del cliente y detalles de la computadora. |
| **CU02** | Administrador, Técnico |  | El técnico o administrador actualiza el estado de la reparación (en proceso, reparado, listo para entregar, finalizado). |
| **CU03** | Sistema, Administrador |  | El sistema calcula los costos de mano de obra, piezas y materiales, y el saldo pendiente de pago. |
| **CU04** | Cliente, Administrador |  | El sistema permite registrar los pagos parciales y finales realizados por el cliente. |
| **CU05** | Sistema, Cliente |  | El sistema genera un comprobante con el adelanto recibido y la fecha estimada de entrega. |
| **CU06** | Sistema, Cliente |  | El sistema genera un recibo detallado con el costo total, adelanto recibido y saldo pagado. |
| **CU07** | Administrador, Técnico |  | El administrador asigna un técnico a cada reparación en función de la especialización y disponibilidad. |
| **CU08** | Administrador, Sistema |  | El sistema permite registrar y actualizar piezas y materiales, y actualizar el inventario en tiempo real. |
| **CU09** | Cliente, Administrador |  | El sistema permite consultar las reparaciones previas realizadas para un cliente. |
| **CU10** | Administrador |  | El sistema genera reportes sobre el número total de reparaciones, costos y técnicos involucrados. |
| **CU11** | Administrador |  | El sistema genera reportes sobre el estado del inventario y las piezas/materiales con bajo stock. |

## Diagrama de Casos de Uso



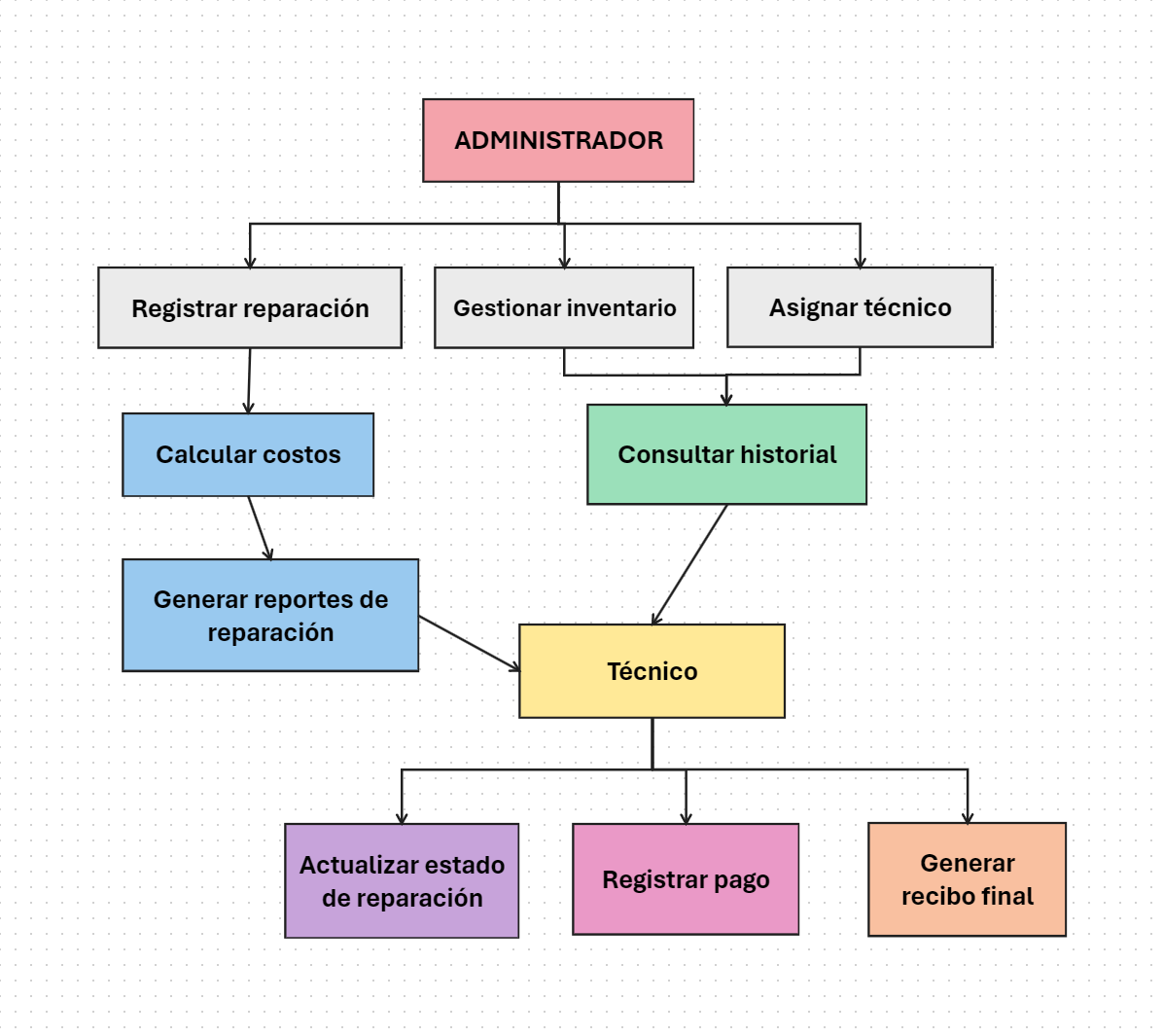
## Identificación de Actores

|  |  |  |
| --- | --- | --- |
| **CÓDIGO** | **NOMBRE** | **DESCRIPCIÓN** |
| **ACT01** | Cliente | Persona que solicita una reparación, realiza pagos y recibe el equipo reparado. |
| **ACT02** | Técnico | Persona que realiza las reparaciones de computadoras y registra información sobre las reparaciones realizadas. |
| **ACT03** | **Administrador** | Persona encargada de gestionar el sistema, roles de usuario, inventario, pagos, y generar reportes. |
| **ACT04** | **Sistema** | Entidad automatizada que gestiona el inventario, cálculos de costos, actualización de estados de reparación, y la generación de reportes. |

## Casos de Uso

|  |  |  |  |
| --- | --- | --- | --- |
| **CÓDIGO** | **ACTOR** | **NOMBRE** | **DESCRIPCIÓN** |
| **CU01** | |  | | --- | |  |  |  | | --- | | Administrador, Cliente | | **Registrar Reparación** | El sistema permite registrar una nueva reparación, incluyendo información del cliente y detalles de la computadora. |
| **CU02** | Administrador, Técnico | **Actualizar Estado de la Reparación** | El técnico o administrador actualiza el estado de la reparación (en proceso, reparado, listo para entregar, finalizado). |
| **CU03** | Sistema, Administrador | **Calcular Costos** | El sistema calcula los costos de mano de obra, piezas y materiales, y el saldo pendiente de pago. |
| **CU04** | Cliente, Administrador | **Registrar Pago** | El sistema permite registrar los pagos parciales y finales realizados por el cliente. |
| **CU05** | Sistema, Cliente | **Generar Comprobante de Recepción** | El sistema genera un comprobante con el adelanto recibido y la fecha estimada de entrega. |
| **CU06** | Sistema, Cliente | **Generar Recibo Final** | El sistema genera un recibo detallado con el costo total, adelanto recibido y saldo pagado. |
| **CU07** | Administrador, Técnico | **Asignar Técnico a Reparación** | El administrador asigna un técnico a cada reparación en función de la especialización y disponibilidad. |
| **CU08** | Administrador, Sistema | **Gestionar Inventario** | El sistema permite registrar y actualizar piezas y materiales, y actualizar el inventario en tiempo real. |
| **CU09** | Cliente, Administrador | **Consultar Historial de Reparaciones** | El sistema permite consultar las reparaciones previas realizadas para un cliente. |
| **CU10** | Administrador | **Generar Reportes de Reparaciones** | El sistema genera reportes sobre el número total de reparaciones, costos y técnicos involucrados. |
| **CU11** | Administrador | **Generar Reportes de Inventario** | El sistema genera reportes sobre el estado del inventario y las piezas/materiales con bajo stock. |

## Diagrama de Casos de Uso



# CU01 – NOMBRE DEL CASO DE USO

## Especificación del caso de uso

|  |  |  |
| --- | --- | --- |
| **Código** | **CU01** | |
| **Nombre** | **Registrar Reparación** | |
| **Descripción** | El caso de uso **Registrar Reparación** permite al administrador o cliente ingresar todos los detalles necesarios para registrar una nueva reparación en el sistema. Esto incluye la información del cliente, los detalles del equipo (computadora), el diagnóstico inicial y los costos estimados. El sistema también asigna un identificador único para cada reparación. | |
| **Actores** | **Administrador y Cliente** | |
| **Requerimiento asociado** | **Requerimiento Funcional 1**: El sistema debe permitir registrar cada trabajo de reparación con detalles como la información del cliente, la computadora, el diagnóstico y los costos estimados de mano de obra y materiales. | |
| **Casos de uso asociados** | **CU3: Calcular Costos**: Calcular el costo de la reparación basándose en los detalles ingresados.  **CU7: Asignar Técnico a Reparación**: Asignar un técnico para la reparación una vez registrada. | |
| **Precondición** | El cliente debe haber entregado la computadora para su reparación.  El administrador debe estar autenticado en el sistema para poder registrar la reparación. | |
| **Secuencia normal** | **Paso** | **Acción** |
| **1** | El cliente o administrador ingresa la información personal del cliente (nombre, contacto). |
| **2** | Se ingresan los detalles de la computadora (marca, modelo, número de serie). |
| **3** | El cliente describe el problema o motivo de la reparación. |
| **4** | Se ingresan las acciones a realizar y las piezas/materiales necesarios. |
| **5** | El sistema calcula el costo estimado de la mano de obra y materiales. |
| **6** | Se registra el costo total estimado de la reparación. |
| **7** | El sistema genera un identificador único para la reparación. |
| **8** | El sistema guarda todos los datos y notifica al administrador que la reparación ha sido registrada. |
| **Postcondición** | El sistema ha registrado correctamente la reparación, asignando un identificador único.  El cliente recibe la confirmación de la reparación registrada con los detalles ingresados. | |
| **Excepciones** | **Paso** | **Acción** |
| **1** | Si el cliente no proporciona información válida, el sistema muestra un mensaje de error indicando los campos faltantes. |
| **2** | Si el costo de las piezas o materiales excede el presupuesto estimado, el sistema solicita confirmación antes de registrar la reparación. |
| **3** | Si el técnico asignado no está disponible, el sistema muestra una advertencia para asignar otro técnico. |
| **Comentarios** | Este caso de uso está muy relacionado con otros, como el de la **Asignación de Técnico a Reparación** y el de **Calcular Costos**, los cuales deben ejecutarse en secuencia para asegurar la correcta gestión de la reparación.  En cuanto a la usabilidad, se recomienda que el formulario de registro sea claro y sencillo, con validación de datos en tiempo real para evitar errores al ingresar la información. | |

# MODELO DE BASE DE DATOS

## Modelo Conceptual

El modelo conceptual describe las entidades principales de la base de datos y sus relaciones. Las entidades clave para el sistema son:

* **CLIENTE**: Representa a los clientes que solicitan servicios de reparación.
* **COMPUTADORA**: Detalla las computadoras que serán reparadas.
* **TECNICO**: Almacena la información del personal encargado de realizar las reparaciones.
* **REGISTRO**: Guarda los detalles de las solicitudes de reparación.
* **ESTADO**: Representa el estado actual de una reparación.
* **TIPOITEM**: Define los tipos de items utilizados (e.g., Procesador, Servicio).
* **ITEM**: Contiene información sobre los componentes y servicios disponibles.
* **USOITEMS**: Registra los items utilizados en una reparación específica.
* **PAGO**: Contabiliza los pagos realizados por los clientes.

Las relaciones entre estas entidades son:

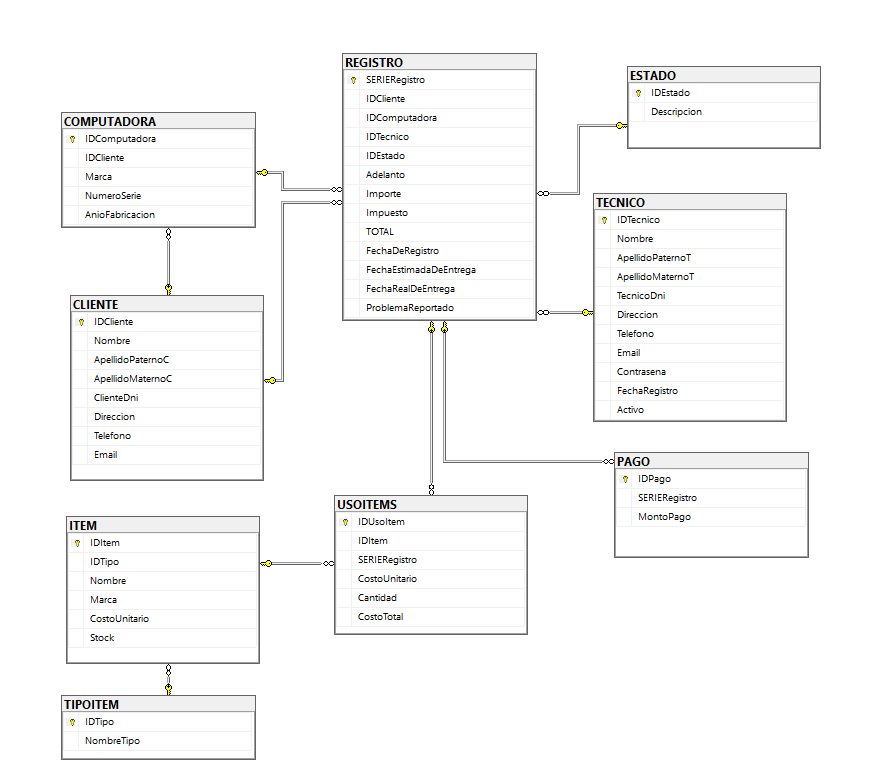
* **Un cliente** puede tener **múltiples computadoras**.
* **Una computadora** pertenece a un **único cliente**.
* **Un técnico** puede gestionar **múltiples registros de reparación**.
* **Cada registro de reparación** está asociado a **un cliente**, **una computadora** y **un técnico**.
* **Un registro** puede tener uno o varios **items utilizados**.
* **Un estado** indica la fase del proceso de reparación.

## Modelo Relacional

El modelo relacional describe las tablas y sus relaciones. Aquí tienes el diseño relacional para tu base de datos:

1. CLIENTE (IDCliente, Nombre, ApellidoPaternoC, ApellidoMaternoC, ClienteDni, Direccion, Telefono, Email)
2. COMPUTADORA (IDComputadora, IDCliente, Marca, NumeroSerie, AnioFabricacion)
   * FK: IDCliente referencia a CLIENTE(IDCliente)
3. TECNICO (IDTecnico, Nombre, ApellidoPaternoT, ApellidoMaternoT, TecnicoDni, Direccion, Telefono, Email, Contrasena, FechaRegistro, Activo)
4. ESTADO (IDEstado, Descripcion)
5. REGISTRO (SERIERegistro, IDCliente, IDComputadora, IDTecnico, IDEstado, Adelanto, Importe, Impuesto, TOTAL, FechaDeRegistro, FechaEstimadaDeEntrega, FechaRealDeEntrega, ProblemaReportado)
   * FK: IDCliente referencia a CLIENTE(IDCliente)
   * FK: IDComputadora referencia a COMPUTADORA(IDComputadora)
   * FK: IDTecnico referencia a TECNICO(IDTecnico)
   * FK: IDEstado referencia a ESTADO(IDEstado)
6. TIPOITEM (IDTipo, NombreTipo)
7. ITEM (IDItem, IDTipo, Nombre, Marca, CostoUnitario, Stock)
   * FK: IDTipo referencia a TIPOITEM(IDTipo)
8. USOITEMS (IDUsoItem, IDItem, SERIERegistro, CostoUnitario, Cantidad, CostoTotal)
   * FK: IDItem referencia a ITEM(IDItem)
   * FK: SERIERegistro referencia a REGISTRO(SERIERegistro)
9. PAGO (IDPago, SERIERegistro, MontoPago)
   * FK: SERIERegistro referencia a REGISTRO(SERIERegistro)

En el grafico se puede visualizar de manera más optima la relacion entre tablas



## Diccionario de Datos

### Tabla: CLIENTE

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| IDCliente | INT | Identificador del cliente (PRIMARY KEY) |
| Nombre | VARCHAR(100) | Nombre del cliente |
| ApellidoPaternoC | VARCHAR(25) | Apellido paterno del cliente |
| ApellidoMaternoC | VARCHAR(25) | Apellido materno del cliente |
| ClienteDni | CHAR(8) | DNI del cliente |
| Direccion | VARCHAR(50) | Dirección del cliente |
| Telefono | VARCHAR(20) | Teléfono del cliente |
| Email | VARCHAR(50) | Correo electrónico del cliente |

### Tabla: COMPUTADORA

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| IDComputadora | INT | Identificador de la computadora (PRIMARY KEY) |
| IDCliente | INT | Identificador del cliente (FOREIGN KEY) |
| Marca | VARCHAR(50) | Marca de la computadora |
| NumeroSerie | VARCHAR(50) | Número de serie de la computadora |
| AnioFabricacion | INT | Año de fabricación de la computadora |

### Tabla: TECNICO

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| IDTecnico | INT | Identificador del técnico (PRIMARY KEY) |
| Nombre | VARCHAR(100) | Nombre del técnico |
| ApellidoPaternoT | VARCHAR(25) | Apellido paterno del técnico |
| ApellidoMaternoT | VARCHAR(25) | Apellido materno del técnico |
| TecnicoDni | CHAR(8) | DNI del técnico |
| Direccion | VARCHAR(50) | Dirección del técnico |
| Telefono | VARCHAR(25) | Teléfono del técnico |
| Email | VARCHAR(100) | Correo electrónico del técnico |
| Contrasena | VARCHAR(255) | Contraseña del técnico |
| FechaRegistro | DATETIME | Fecha de registro del técnico |
| Activo | BIT | Estado activo del técnico |

### Tabla: REGISTRO

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| SERIERegistro | VARCHAR(10) | Identificador del registro (PRIMARY KEY) |
| IDCliente | INT | Identificador del cliente (FOREIGN KEY) |
| IDComputadora | INT | Identificador de la computadora (FOREIGN KEY) |
| IDTecnico | INT | Identificador del técnico (FOREIGN KEY) |
| IDEstado | INT | Identificador del estado (FOREIGN KEY) |
| Adelanto | NUMERIC(10,2) | Monto adelantado por el cliente |
| Importe | NUMERIC(10,2) | Importe total del registro |
| Impuesto | NUMERIC(10,2) | Monto del impuesto |
| TOTAL | NUMERIC(10,2) | Monto total a pagar |
| FechaDeRegistro | DATE | Fecha de registro |
| FechaEstimadaDeEntrega | DATE | Fecha estimada de entrega |
| FechaRealDeEntrega | DATE | Fecha real de entrega |
| ProblemaReportado | VARCHAR(200) | Descripción del problema reportado |

### Tabla: PAGO

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| IDPago | INT | Identificador del pago (PRIMARY KEY) |
| SERIERegistro | VARCHAR(10) | Identificador del registro (FOREIGN KEY) |
| MontoPago | NUMERIC(10,2) | Monto del pago realizado |

### Tabla: ESTADO

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| IDEstado | INT | Identificador del estado (PRIMARY KEY) |
| Descripcion | VARCHAR(20) | Descripción del estado |

### Tabla: TIPOITEM

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| IDTipo | INT | Identificador del tipo de item (PRIMARY KEY) |
| NombreTipo | VARCHAR(30) | Nombre del tipo de item |

### Tabla: ITEM

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| IDItem | INT | Identificador del item (PRIMARY KEY) |
| IDTipo | INT | Identificador del tipo de item (FOREIGN KEY) |
| Nombre | VARCHAR(100) | Nombre del item |
| Marca | VARCHAR(30) | Marca del item |
| CostoUnitario | NUMERIC(10,2) | Costo unitario del item |
| Stock | FLOAT | Cantidad disponible en stock |

### Tabla: USOITEMS

|  |  |  |
| --- | --- | --- |
| Campo | Tipo de Dato | Descripción |
| IDUsoItem | INT | Identificador del uso de item (PRIMARY KEY) |
| IDItem | INT | Identificador del item (FOREIGN KEY) |
| SERIERegistro | VARCHAR(10) | Identificador del registro (FOREIGN KEY) |
| CostoUnitario | NUMERIC(10,2) | Costo unitario del item utilizado |
| Cantidad | INT | Cantidad del item utilizado |
| CostoTotal | NUMERIC(10,2) | Costo total calculado (CostoUnitario \* Cantidad) |

# IMPLEMENTACIÓN DE LA BASE DE DATOS

## Creación de la Base de Datos

La base de datos se crea utilizando SQL Server. El script de creación elimina cualquier base de datos con el mismo nombre y luego procede a crear la base de datos REPARACIONES\_UNI.

USE master;

GO

IF (EXISTS (SELECT 1 FROM sys.sysdatabases WHERE name='REPARACIONES\_UNI'))

BEGIN

ALTER DATABASE REPARACIONES\_UNI SET SINGLE\_USER WITH ROLLBACK IMMEDIATE;

DROP DATABASE REPARACIONES\_UNI;

END;

GO

CREATE DATABASE REPARACIONES\_UNI;

GO

## Creación de los objetos de la base de datos

Se han creado las tablas con sus respectivas restricciones y claves foráneas. Un ejemplo de creación de tabla es:

USE master;

GO

-- Tabla CLIENTE

CREATE TABLE dbo.CLIENTE (

IDCliente INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

Nombre VARCHAR(100) NOT NULL,

ApellidoPaternoC VARCHAR(25) NOT NULL,

ApellidoMaternoC VARCHAR(25) NOT NULL,

ClienteDni CHAR(8) NOT NULL,

Direccion VARCHAR(50) NOT NULL,

Telefono VARCHAR(20) NOT NULL,

Email VARCHAR(50) NOT NULL

);

GO

-- Tabla COMPUTADORA

CREATE TABLE dbo.COMPUTADORA (

IDComputadora INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

IDCliente INT NOT NULL,

Marca VARCHAR(50) NOT NULL,

NumeroSerie VARCHAR(50) NOT NULL,

AnioFabricacion INT NOT NULL,

CONSTRAINT FK\_COMPUTADORA\_CLIENTE FOREIGN KEY (IDCliente) REFERENCES dbo.CLIENTE (IDCliente)

);

GO

-- Tabla TECNICO

CREATE TABLE dbo.TECNICO (

IDTecnico INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

Nombre VARCHAR(100) NOT NULL,

ApellidoPaternoT VARCHAR(25) NOT NULL,

ApellidoMaternoT VARCHAR(25) NOT NULL,

TecnicoDni CHAR(8) NOT NULL,

Direccion VARCHAR(50) NOT NULL,

Telefono VARCHAR(25) NOT NULL,

Email VARCHAR(100) NOT NULL,

Contrasena VARCHAR(255) NOT NULL,

FechaRegistro DATETIME DEFAULT GETDATE(),

Activo BIT NOT NULL

);

GO

-- Tabla ESTADO

CREATE TABLE dbo.ESTADO (

IDEstado INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

Descripcion VARCHAR(20) NOT NULL

);

GO

-- Tabla REGISTRO

CREATE TABLE dbo.REGISTRO (

SERIERegistro VARCHAR(10) NOT NULL PRIMARY KEY,

IDCliente INT NOT NULL,

IDComputadora INT NOT NULL,

IDTecnico INT NOT NULL,

IDEstado INT NOT NULL,

Adelanto NUMERIC(10,2) NOT NULL,

Importe NUMERIC(10,2) NOT NULL ,

Impuesto NUMERIC(10,2),

TOTAL NUMERIC(10,2),

FechaDeRegistro DATE NOT NULL,

FechaEstimadaDeEntrega DATE NOT NULL,

FechaRealDeEntrega DATE,

ProblemaReportado VARCHAR(200) NOT NULL,

CONSTRAINT FK\_REGISTRO\_COMPUTADORA FOREIGN KEY (IDComputadora) REFERENCES dbo.COMPUTADORA (IDComputadora),

CONSTRAINT FK\_REGISTRO\_CLIENTE FOREIGN KEY (IDCliente) REFERENCES dbo.CLIENTE (IDCliente),

CONSTRAINT FK\_REGISTRO\_TECNICO FOREIGN KEY (IDTecnico) REFERENCES dbo.TECNICO (IDTecnico),

CONSTRAINT FK\_REGISTRO\_ESTADO FOREIGN KEY (IDEstado) REFERENCES dbo.ESTADO (IDEstado)

);

GO

-- Tabla TIPOITEM

CREATE TABLE dbo.TIPOITEM (

IDTipo INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

NombreTipo VARCHAR(30) NOT NULL

);

GO

-- Tabla ITEM

CREATE TABLE dbo.ITEM (

IDItem INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

IDTipo INT NOT NULL,

Nombre VARCHAR(100) NOT NULL,

Marca VARCHAR(30) NOT NULL,

CostoUnitario NUMERIC(10,2) NOT NULL,

Stock FLOAT NOT NULL,

CONSTRAINT FK\_TIPO\_ITEM FOREIGN KEY (IDTipo) REFERENCES dbo.TIPOITEM (IDTipo)

);

GO

-- Tabla USOITEMS

CREATE TABLE dbo.USOITEMS (

IDUsoItem INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

IDItem INT NOT NULL,

SERIERegistro VARCHAR(10) NOT NULL,

CostoUnitario NUMERIC(10, 2) NOT NULL,

Cantidad INT NOT NULL,

CostoTotal AS (CostoUnitario \* Cantidad) PERSISTED,

CONSTRAINT FK\_USOITEMS\_ITEM FOREIGN KEY (IDItem) REFERENCES dbo.ITEM (IDItem),

CONSTRAINT FK\_REPARACION\_REGISTRO FOREIGN KEY (SERIERegistro) REFERENCES dbo.REGISTRO (SERIERegistro)

);

GO

-- Tabla PAGO

CREATE TABLE dbo.PAGO (

IDPago INT NOT NULL IDENTITY(1,1) PRIMARY KEY,

SERIERegistro VARCHAR(10) NOT NULL,

MontoPago NUMERIC(10,2) NOT NULL,

CONSTRAINT FK\_PAGO\_REGISTRO FOREIGN KEY (SERIERegistro) REFERENCES dbo.REGISTRO (SERIERegistro),

);

GO

## Cargar Datos de Prueba

En esta sección se cargan los datos de prueba respectivos.

--CLIENTE

INSERT INTO dbo.CLIENTE (Nombre, ApellidoPaternoC, ApellidoMaternoC, ClienteDni, Direccion, Telefono, Email)

VALUES

('Jesús', 'Carhuas', 'Ramírez', '12345678', 'Av. Los Álamos 123', '987654321', 'jesus.carhuas@mail.com'),

('Jesús', 'Carhuas', 'Ramírez', '12345678', 'Av. Los Álamos 123', '987654321', 'jesus.carhuas@mail.com');

GO

--COMPUTADORA

INSERT INTO dbo.COMPUTADORA (IDCliente, Marca, NumeroSerie, AnioFabricacion)

VALUES

(1, 'Dell', 'ABC12345', 2022),

(2, 'Dell', 'ABC12345', 2023);

GO

--TECNICO

INSERT INTO dbo.TECNICO (Nombre, ApellidoPaternoT, ApellidoMaternoT, TecnicoDni, Direccion, Telefono, Email, Contrasena, Activo)

VALUES

('Carlos', 'Ramirez', 'Lopez', '12345678', 'Av. Los Olivos 123', '999888777', 'carlos.ramirez@example.com', 'contrasenaSegura123', 0),

('Carlos', 'Ramirez', 'Lopez', '12355678', 'Av. Los Olivos 123', '999888777', 'carlos.ramirez@example.com', 'contrasenaSegura123', 0);

--ESTADOS

-- Tabla: ESTADO

INSERT INTO dbo.ESTADO (Descripcion) VALUES ('EN PROCESO');

INSERT INTO dbo.ESTADO (Descripcion) VALUES ('REPARADO');

INSERT INTO dbo.ESTADO (Descripcion) VALUES ('LISTO PARA ENTREGAR');

INSERT INTO dbo.ESTADO (Descripcion) VALUES ('FINALIZADO');

-- Tabla: TIPOITEM

INSERT INTO dbo.TIPOITEM (NombreTipo) VALUES ('Procesador');

INSERT INTO dbo.TIPOITEM (NombreTipo) VALUES ('Servicios');

---Tabla: ITEMS

-- Procesadores

INSERT INTO dbo.ITEM (IDTipo, Nombre, Marca, CostoUnitario, Stock) VALUES (1, 'Intel Core i9-13900K', 'Intel', 599.99, 50);

INSERT INTO dbo.ITEM (IDTipo, Nombre, Marca, CostoUnitario, Stock) VALUES (1, 'AMD Ryzen 9 7900X', 'AMD', 569.99, 30);

INSERT INTO dbo.ITEM (IDTipo, Nombre, Marca, CostoUnitario, Stock) VALUES (1, 'Intel Core i7-12700K', 'Intel', 399.99, 45);

INSERT INTO dbo.ITEM (IDTipo, Nombre, Marca, CostoUnitario, Stock) VALUES (1, 'AMD Ryzen 7 7700X', 'AMD', 349.99, 40);

INSERT INTO dbo.ITEM (IDTipo, Nombre, Marca, CostoUnitario, Stock) VALUES (1, 'Intel Core i5-13600K', 'Intel', 299.99, 60);

-- Servicios

INSERT INTO dbo.ITEM (IDTipo, Nombre, Marca, CostoUnitario, Stock) VALUES (2, 'Reemplazo de Procesador', 'Técnico', 100.00, 999999);

--Prueba Registro

INSERT INTO dbo.REGISTRO (

SERIERegistro, IDCliente, IDComputadora, IDTecnico, IDEstado,

Adelanto, Importe,Impuesto,TOTAL, FechaDeRegistro,

FechaEstimadaDeEntrega, ProblemaReportado

)

VALUES (

'R00002', -- SERIERegistro

1, -- IDCliente

1, -- IDComputadora

1, -- IDTecnico

1, -- IDEstado

100.00, -- Adelanto

500.00, -- importe

180.00, --igv

680.00, --total

GETDATE(), -- FechaDeRegistro

'2024-11-20', -- FechaEstimadaDeEntrega

'Problema de arranque del sistema operativo' -- ProblemaReportado

);

--TABLA USOITEMS

INSERT INTO dbo.USOITEMS (IDItem, SERIERegistro, CostoUnitario, Cantidad)

VALUES

(1, 'R00001', 600, 1),

(6, 'R00001', 100, 1)

INSERT INTO dbo.PAGO(SERIERegistro,MontoPago) VALUES

('R00001',100),

('R00001',726)

# CONCLUSIONES

* El trabajo busca la simpleza en la administración de inserción de datos, es por ello las pocas tablas que usamos en la base de datos. Para de esta forma mejorar en la parte de eficiencia y que se pueda adaptar en cualquier negocio de este rubro.
* Con funcionalidades como el seguimiento en tiempo real del estado de las reparaciones y la generación de comprobantes detallados, el sistema promueve una comunicación más transparente y confiable con los clientes. Esto no solo mejora la percepción del servicio, sino que también fomenta la fidelización de los usuarios.
* Al automatizar el cálculo de costos de mano de obra, materiales y totales, el sistema permite un manejo más preciso de los aspectos financieros del taller. Además, la capacidad de registrar adelantos y saldos pendientes facilita el seguimiento de pagos y asegura la claridad en las transacciones económicas.
* Este sistema representa un avance significativo para la gestión de talleres de reparación de computadoras, proporcionando una solución tecnológica robusta y centrada en las necesidades esenciales. Su implementación no solo mejora la eficiencia y la precisión, sino que también establece las bases para un crecimiento sostenible del negocio.

# LECCIONES APRENDIDAS

* SEMANA 9:

Aprender el entorno de Spring tool Suite y postman, Adicionalmente inyectar las dependencias.

* SEMANA 10:

Ingresar en la Base de datos de forma eficaz y correcta. En este caso se aprendió a definir manualmente la conexión al DataSource, lo que permite un control total sobre la forma en que la aplicación se comunica con la base de datos.

El aprendizaje de estas semanas sentó las bases para trabajar con Spring Boot, desde la configuración inicial hasta la manipulación directa de datos en la base de datos. Esto permite un desarrollo estructurado y controlado, clave para sistemas robustos y mantenibles.

* SEMANA 11:

Durante esta semana, aprendimos cómo implementar operaciones básicas de CRUD (Crear, Leer, Actualizar y Eliminar) en un proyecto Spring Boot utilizando el entorno de desarrollo Spring Tool Suite (STS). Esto incluyó el uso de herramientas y beneficios clave proporcionados por el Spring Framework.

* SEMANA 12

Introducción a transacciones y adicionalmente revisión de base de datos del docente.

# BIBLIOGRAFIA

* Java Guides. (2020, July 15). Spring Boot Tutorial for Beginners [Video]. YouTube. <https://www.youtube.com/watch?v=9SGDpanrc8U>