

Diseño e implementación de bases de datos para Venta de Guitarras.

Ingeniería en Sistemas de la Información, Universidad Nacional de Costa Rica.

Diseño e implementación de bases de datos - EIF 211.

Roberto Campos Calvo 118260676

Sebastián Campos Calvo 119120829

David Morales Gatica 208270603

Ing. Francisco Coulon Ollivier

12 de junio del 2024.

Evaluación %	Exposición
15 pts	Dominio del tema y fluidez (debe demostrar que domina el contexto de la exposición).
10 pts	Presentación Personal (vestimenta Formal)
15 pts	Desenvolvimiento, dominio del escenario, dicción y elocuencia
10 pts	Uso adecuado de los medios audiovisuales (multimedia).
50 pts	Cumplimiento del proyecto
100 PTOS	TOTAL:
5%	Porcentaje Total
	Parte de desarrollo operativo o programada
25 pts	Diagrama Contextual
15 pts	Diagrama relacional
15 pts	Normalización de cada tabla diseñada
43 pts	Implementación base datos, tablas, inserción de datos, consulta y procedimiento
2 pts	Diccionario
100 pts	TOTAL:
20%	Porcentaje Total:
	Parte Escrita
2 pts	Portada y hoja de evaluación
2 pts	Tabla de contenido
4 pts	Introducción
2 pts	Planteamiento del problema:
4 pts	Justificación
10 pts	Objetivos (General y Específicos)
13 pts	Resultados y su discusión
10 pts	Conclusiones
13 pts	Recomendaciones
3 pts	Referencias (Normas APA)
10 pts	Anexos
7 pts	Bitácora
	<i>Aspectos de Estilo, Forma y Fondo</i>
5 Ptos	Redacción
5 pts	Ortografía
10 pts	Encadenamiento entre: Objetivos - Metodología - Resultados y Conclusiones
100 pts	TOTAL:
5 %	Porcentaje Total: 30%

Introducción.....	4
Planteamiento del problema.....	4
Justificación.....	5
Objetivo General.....	6
Objetivos Específicos:.....	6
Descripción de su empresa.....	6
Recomendaciones.....	7
Modelo entidad relación.....	9
Modelo relacional.....	10
Estrategia de auditoría.....	11
Tecnologías Utilizadas.....	11
Draw.io.....	11
¿Aplicaciones en nuestro proyecto para el diseño de bases de datos?.....	11
Visual Studio Code.....	12
SQL Server.....	12
DBever.....	12
SQL Server Management Studio (SSMS) 21.....	12
Google Docs.....	12
WhatsApp, Discord y Google Meet.....	12
GitHub.....	13
Gmail.....	13
Conclusiones.....	13
Referencias.....	14
Anexos.....	14
Consideraciones para siguientes pasos.....	14
Bitácora de trabajo.....	15
Sesión #1.....	15
Sesión #2.....	17
Sesión #3.....	19
Sesión #4.....	20

Introducción

El presente proyecto consiste en el diseño e implementación de una base de datos relacional para una empresa local dedicada a la venta de guitarras, repuestos y la oferta de clases de guitarra. El objetivo principal es proporcionar una solución concisa que permita gestionar de manera eficiente tanto el inventario y las ventas de productos, como la administración de clases y profesores, asegurando la trazabilidad y el control de todas las operaciones relevantes del negocio.

La base de datos ha sido desarrollada en SQL Server, siguiendo principios de normalización y buenas prácticas de modelado de datos. El sistema contempla la gestión de usuarios (clientes, profesores, estudiantes), productos (guitarras y repuestos), inventario, pedidos, ofertas, historial de precios, clases, matrículas y auditoría de operaciones. Además, se han implementado procedimientos almacenados, triggers y vistas para automatizar procesos clave y facilitar la obtención de reportes y consultas frecuentes.

Este documento describe la estructura del modelo de datos, las tecnologías utilizadas, los retos identificados para una versión final, la estrategia de auditoría implementada y las principales funcionalidades desarrolladas, cumpliendo con los requerimientos establecidos para la prueba de concepto.

Planteamiento del problema

En la actualidad, la gestión manual de ventas, inventario y clases en pequeños negocios musicales representa un reto significativo. Un grupo de amigos apasionados por la música y las guitarras ha decidido emprender un negocio local enfocado en la venta de guitarras, repuestos y la oferta de clases de guitarra. Sin embargo, la ausencia de un sistema centralizado y automatizado para el manejo de la información genera diversos problemas:

1. **Dificultad para controlar el inventario:** La falta de un registro actualizado de productos y repuestos puede llevar a errores en el stock, ventas de productos agotados o pérdidas por falta de reposición oportuna.
2. **Gestión ineficiente de ventas y pedidos:** El registro manual de ventas y pedidos dificulta el seguimiento de las transacciones, la generación de reportes y la identificación de los productos más vendidos.
3. **Falta de trazabilidad en los cambios de precios y ofertas:** Los precios de los productos pueden variar con el tiempo y la gestión manual de estos cambios puede

generar confusión, errores y falta de transparencia para los clientes y administradores.

4. **Administración limitada de clases y profesores:** Sin un sistema adecuado, la programación de clases, la asignación de profesores y el seguimiento del historial de alumnos y docentes se vuelve compleja y propensa a errores.

5. **Ausencia de auditoría y control:** No contar con un registro de las operaciones importantes impide detectar errores, fraudes o cambios no autorizados en la información crítica del negocio.

Justificación

La implementación de una base de datos relacional para la gestión de una empresa dedicada a la venta de guitarras, repuestos y la oferta de clases de guitarra es crucial para garantizar la eficiencia, el control y la escalabilidad del negocio. Un sistema centralizado permite facilitar procesos clave, reducir errores humanos y mejorar el uso tanto de los clientes como de los administradores.

En primera instancia, la sistematización de la gestión de inventario, ventas, clases y ofertas optimiza los procesos internos, permitiendo ahorrar tiempo y minimizar errores. Esto facilita el seguimiento de las operaciones diarias y libera recursos que pueden ser destinados a tareas de mayor valor para el negocio. Por otro lado, la trazabilidad y transparencia que ofrece el registro histórico de precios, ventas, clases y cambios importantes en la base de datos es esencial para la toma de decisiones informadas. Además, facilita la auditoría y la detección de posibles irregularidades, lo que contribuye a la confianza y seguridad en la administración del negocio. Asimismo, la estructura relacional y normalizada de la base de datos facilita la incorporación de nuevos productos, categorías, promociones y funcionalidades. Esto asegura que el sistema pueda adaptarse y crecer junto con las necesidades del negocio, evitando limitaciones tecnológicas a futuro.

Finalmente, la disponibilidad de vistas y consultas especializadas permite obtener información relevante sobre ventas, productos más populares, historial de clases y precios. Esto apoya la gestión estratégica y la toma de decisiones basadas en datos, elementos indispensables para la profesionalización y competitividad de la empresa.

En otras palabras, la base de datos propuesta no solo resuelve los problemas actuales de gestión, sino que también sienta las bases para el crecimiento y la profesionalización de la empresa.

Objetivo General

Diseñar e implementar una base de datos relacional en SQL Server que permita gestionar de manera eficiente y centralizada la venta de guitarras, repuestos y la administración de clases de guitarra, facilitando el control de inventario, el rastreo de operaciones, la sistematización de procesos clave y la creación de reportes, con el fin de optimizar la gestión del negocio y mejorar la experiencia de clientes, profesores y administradores.

Objetivos Específicos:

1. Desarrollar un modelo entidad-relación y un modelo relacional que representen de manera precisa las entidades y procesos clave del negocio, implementando la base de datos en SQL Server con todas las tablas, relaciones, procedimientos almacenados, triggers y vistas.
2. Establecer mecanismos de auditoría y rastreo que aseguren el control de las operaciones, así como facilitar la generación de reportes y consultas especializadas que apoyen la toma de decisiones, como el análisis de productos más vendidos, historial de precios, historial de clases y búsqueda avanzada de productos.
3. Sistematizar la gestión de inventario, ventas, ofertas y clases, permitiendo el registro, actualización y consulta eficiente de productos, pedidos, promociones y actividades académicas.
4. Proveer scripts de llenado y pruebas para validar el correcto funcionamiento de la base de datos y documentar la solución desarrollada, incluyendo la justificación técnica, la estrategia de auditoría, los retos identificados y las recomendaciones para futuros casos.

Descripción de su empresa

Riff Music es una tienda especializada en la venta de guitarras y accesorios musicales, fundada con la visión de ser el destino principal para músicos y entusiastas de la guitarra. Nuestra tienda se distingue por ofrecer una variedad de selección de instrumentos de las marcas más prestigiosas del mercado, asegurando que cada cliente encuentre el instrumento perfecto para sus necesidades.

Además de nuestra amplia gama de productos, Riff Music se enorgullece de ofrecer un programa educativo completo de clases de guitarra. Nuestro equipo de profesores, calificados y con años de experiencia, está comprometido con la enseñanza personalizada.

Ofrecemos clases para todos los niveles, desde principiantes hasta músicos avanzados, adaptando nuestros métodos de enseñanza a las necesidades y objetivos de cada estudiante.

En Riff Music, no solo vendemos instrumentos; creamos una comunidad de músicos apasionados y ofrecemos un espacio donde la música cobra vida.

Análisis del proyecto

Desde Riff Music, vemos este proyecto como una herramienta fundamental para nuestro crecimiento y eficiencia operativa. El sistema que se está desarrollando nos permitirá gestionar de manera eficaz varios aspectos de nuestro negocio, desde el control de inventario hasta la gestión de clases de guitarra.

Uno de los aspectos más valiosos del proyecto es su capacidad para manejar nuestro catálogo de productos. El sistema nos permitirá mantener un registro detallado de cada guitarra y accesorio, incluyendo información específica como el tipo de guitarra (acústica, eléctrica o clásica), marca, modelo, y precio.

En el ámbito educativo, el proyecto nos brinda una plataforma robusta para gestionar nuestras clases de guitarra. Podemos registrar a nuestros profesores, sus especialidades y experiencia, así como mantener un historial detallado de las clases impartidas y el progreso de nuestros estudiantes. Esta funcionalidad es crucial para mantener la calidad de nuestra oferta educativa y para personalizar la experiencia de aprendizaje de cada estudiante.

El sistema también incluye características avanzadas como el control de ofertas y descuentos, lo que nos permitirá implementar estrategias de marketing más efectivas y mantener a nuestros clientes informados sobre las mejores oportunidades.

Recomendaciones

En el área de gestión de inventario, notamos que el sistema actual no incluye alertas automáticas para niveles bajos de stock. Recomendamos implementar un sistema que notifique automáticamente cuando el inventario de guitarras populares caiga por debajo de un umbral predefinido.

En el módulo educativo, el sistema actual maneja la información básica de clases y profesores, pero carece de funcionalidades específicas que mejorarían la experiencia de enseñanza como un calendario integrado que permita a los profesores programar clases de manera eficiente.

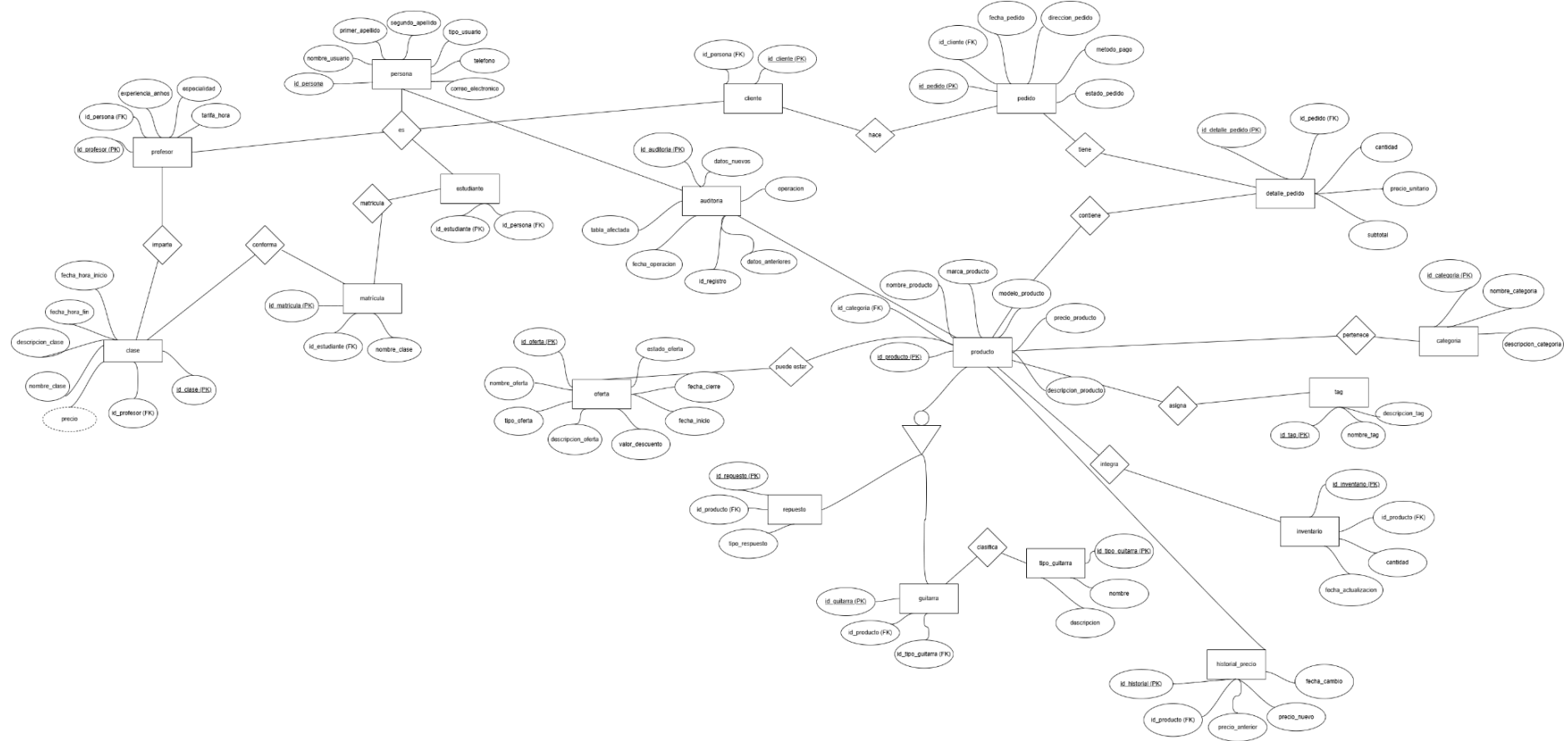
La experiencia del cliente es un área que requiere atención inmediata. El sistema actual debido a que es totalmente local no incluye un portal web para que los clientes vean el

inventario en tiempo real. Recomendamos desarrollar una interfaz web que permita a los clientes buscar guitarras por tipo, marca, precio y disponibilidad.

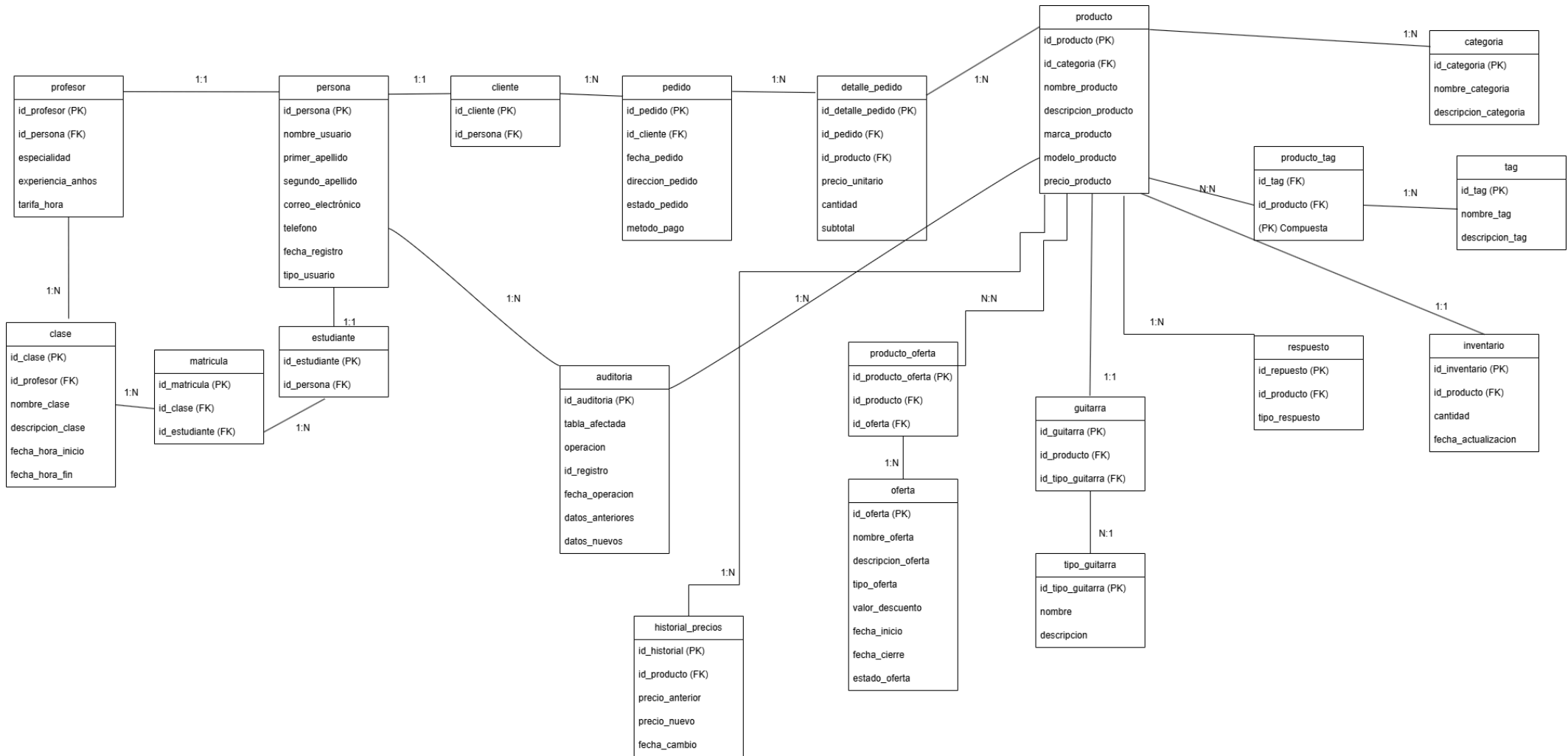
A pesar de que el sistema provee una solución de base de datos robusta y que cumple con los requerimientos establecidos en un inicio en cuanto a las tablas que lo componen, procedimientos, trigger y vistas. Siempre cabe la posibilidad de mejora y se pueden considerar la implementación de scripts adicionales basados en las otras funcionalidades que puedan ser requeridas.

Modelo entidad relación

Link board de Draw.io con los modelos <https://drive.google.com/file/d/1LzBLlrZMhol9GsxYq264x6gQQ-1Y72Kv/view?usp=sharing>



Modelo relacional



Estrategia de auditoría

La estrategia de auditoría implementada se centra en una tabla auditoría que registra automáticamente todos los cambios importantes en la base de datos. Esta tabla almacena información esencial como la tabla afectada, tipo de operación, registro modificado, fecha del cambio y los datos antes y después de la modificación.

Los triggers son el mecanismo clave que automatiza este proceso. Por ejemplo, el trigger `trg_auditoria_producto` se activa después de cada operación INSERT o UPDATE en la tabla producto, capturando automáticamente los cambios realizados. Para inserciones, registra los nuevos datos, mientras que para actualizaciones, guarda tanto el estado anterior como el nuevo estado del registro.

Esta implementación proporciona trazabilidad completa de los cambios, facilita la recuperación de datos en caso de errores y cumple con requisitos de auditoría. Un ejemplo práctico es el seguimiento de cambios en precios de productos, donde el sistema registra automáticamente el precio anterior, el nuevo precio y el momento exacto del cambio.

Tecnologías Utilizadas

Draw.io

Draw.io es una herramienta online y aplicación de escritorio gratuita y de código abierto diseñada para crear y editar una amplia gama de diagramas. Fue particularmente útil para el diseño de bases de datos, permitiéndonos crear y visualizar fácilmente los modelos MER (Modelo Entidad-Relación) y Modelo Relacional.

¿Aplicaciones en nuestro proyecto para el diseño de bases de datos?

- **Modelado MER:** Para la creación de diagramas Entidad-Relación de manera intuitiva. Ya que cuenta con una biblioteca de formas específica para entidades, atributos, relaciones, y cardinalidades, facilitando la representación de la estructura conceptual de la base de datos.
- **Modelado Relacional:** Una vez definido el MER, se implementó para construir el modelo relacional visualizar las tablas, sus columnas (atributos), claves primarias y foráneas, y las relaciones entre ellas, listo para la implementación de la base de datos.

Visual Studio Code

Editor de código fuente utilizado como herramienta principal de desarrollo durante el proyecto. Facilitó la escritura de scripts SQL mediante funcionalidades de resaltado de sintaxis, detección automática de errores de sintaxis y autocompletado de comandos SQL. Su interfaz intuitiva y extensiones especializadas para bases de datos mejoraron significativamente la productividad del equipo en la creación de consultas, triggers y procedimientos almacenados.

SQL Server

Motor gestor de base de datos sobre el cual se desarrolló completamente el proyecto. Proporcionó el entorno robusto necesario para implementar todas las funcionalidades del sistema, incluyendo la creación de tablas, relaciones, índices, triggers, vistas y procedimientos almacenados. Su capacidad de manejo de transacciones y herramientas de administración fueron fundamentales para el desarrollo y testing del sistema.

DBeaver

Cliente universal de base de datos utilizado para la gestión y manipulación de SQL Server. Proporcionó una interfaz gráfica intuitiva para la exploración de esquemas, ejecución de consultas, gestión de datos y tareas administrativas, siendo una herramienta complementaria para el desarrollo y el testing en diferentes entornos de base de datos.

SQL Server Management Studio (SSMS) 21

Entorno de desarrollo integrado (IDE) para gestionar cualquier infraestructura de SQL Server. Utilizado específicamente para la administración de la base de datos SQL Server, la ejecución de scripts, el monitoreo del rendimiento y ver tablas y vistas s. Su robustez fue clave para las tareas de administración avanzadas.

Google Docs

Plataforma colaborativa utilizada para la generación y edición conjunta de la documentación del proyecto. Permitió que los miembros del equipo trabajaran simultáneamente en la redacción de informes, especificaciones técnicas y manuales, facilitando la revisión en tiempo real y el control de versiones de los documentos.

WhatsApp, Discord y Google Meet

Herramientas de comunicación que facilitaron la coordinación del trabajo en equipo y la realización de sesiones de discusión técnica. WhatsApp se utilizó para comunicación rápida y coordinación diaria, Discord y Meet para sesiones de trabajo colaborativo y compartir pantalla durante las sesiones de trabajo.

GitHub

Repositorio remoto utilizado para el control de versiones y almacenamiento centralizado del código SQL y documentación del proyecto. Permitió el trabajo colaborativo seguro, el seguimiento de cambios y avances del proyecto, ayudó a la implementación de un flujo de trabajo organizado para las contribuciones de cada miembro del equipo.

Gmail

Se utilizó como plataforma de correo electrónico para la comunicación formal con el profesor del curso, para la resolución de consultas técnicas.

Conclusiones

En conclusión, la construcción de este sistema de base de datos para la gestión integral de la tienda de guitarras y clases representa un logro para el equipo, completando los objetivos establecidos para el proyecto. Además de proporcionar una solución tecnológica funcional, capaz de gestionar ventas, el control de inventario y la administración de matrículas a clases, este proyecto ha constituido una experiencia de aprendizaje para todo el equipo de desarrollo.

La base de datos provee un control centralizado de la información, evitando inconsistencias y los errores relacionados a los procesos manuales que se implementaban anteriormente. La funcionalidad de auditoría, garantiza la trazabilidad necesaria para mantener controles internos efectivos y salvaguardar la integridad de los datos en todo momento.

El proceso estuvo lleno de desafíos técnicos, los cuales pusieron a prueba nuestros conocimientos y expandieron nuestra comprensión en el diseño y la implementación de bases de datos. Desde la fase inicial de ingeniería, crucial para comprender la lógica de negocio y diseñar los primeros diagramas relacionales, hasta la implementación de elementos avanzados como los triggers, y las views, especialmente aquellas que involucran consultas multitable complejas, múltiples JOINS y subconsultas anidadas.

Este proyecto ha sido una práctica inmersiva en el diseño e implementación de sistemas de bases de datos dentro de un contexto de lógica de negocio real, permitiendo la aplicación directa y concreta de los conceptos abordados durante el curso. Cada tema estudiado en el aula fue aplicado en la práctica en diversos aspectos del proyecto, desde la concepción inicial del esquema hasta la implementación de código sql.

De esta manera, este sistema de base de datos no solo resuelve las problemáticas operativas definidas por los requerimientos, sino que también es la culminación de nuestro proceso de aprendizaje continuo, adaptación y superación de desafíos técnicos

Referencias

Aragón Barreda, Y. L., González Herrera, C. Y., Hernández Socarrás, O. F., & Hernández Luque, E. (2018). Herramienta para el aprendizaje de bases de datos relacionales. *Revista Cubana de Ciencias Informáticas*, 12(3), 163-176. <http://rcci.uci.cu>

Microsoft. (n.d.). *SQL Server*. Microsoft Learn. Recuperado de <https://learn.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver17>

Anexos

Link del repositorio de github con los scripts, documentación del proyecto, Modelo Entidad Relación y Modelo Relacional.

<https://github.com/RobertoUNA/venta-de-guitarras.git>

Consideraciones para siguientes pasos

El sistema desarrollado establece una base sólida para futuras expansiones y mejoras. La arquitectura implementada permite la integración de nuevas funcionalidades, el desarrollo de interfaces de usuario más sofisticadas y la posible implementación de aplicaciones web o móviles que interactúen con la base de datos. Además se debe aplicar una lógica más robusta a la hora de realizar validaciones, aplicar descuentos, alertas de stock, estandarización de categorías, tipos de repuestos, guitarras, productos, manejar casos especiales entre otros.

Bitácora de trabajo

Participantes:

David Morales Gatica	David.Morales.Gatica@est.una.ac.cr
Roberto Campos Calvo	roberto.campos.calvo@est.una.ac.cr
Sebastian Campos Calvo	sebastia.campos.calvo@est.una.ac.cr

Registro de Reuniones y Actividades

Sesión #1

Fecha: Martes 3 de junio, 2025 Hora: 10:00 AM

Modalidad: Virtual (Google Meet)

Objetivos de la reunión:

- Extraer y definir la problemática del problema planteado
- Definir los límites y el propósito del proyecto
- Iniciar el modelo relacional

Actividades realizadas:

1. Consulta al docente vía correo electrónico:

Pregunta 1: ¿Se trata de una tienda online tipo e-commerce o de una aplicación para uso del personal de la tienda tipo punto de venta?

Pregunta 2: ¿Las clases son lecciones independientes por hora o son tipo cursos con una duración de clases estimada?

Respuesta del docente: "Todas esas preguntas ustedes deciden cómo enfocarlo"

2. Diseño inicial del modelo relacional: La herramienta utilizada fue Draw.io
3. Se desarrolló en la documentación la introducción, planteamiento del problema y justificación.

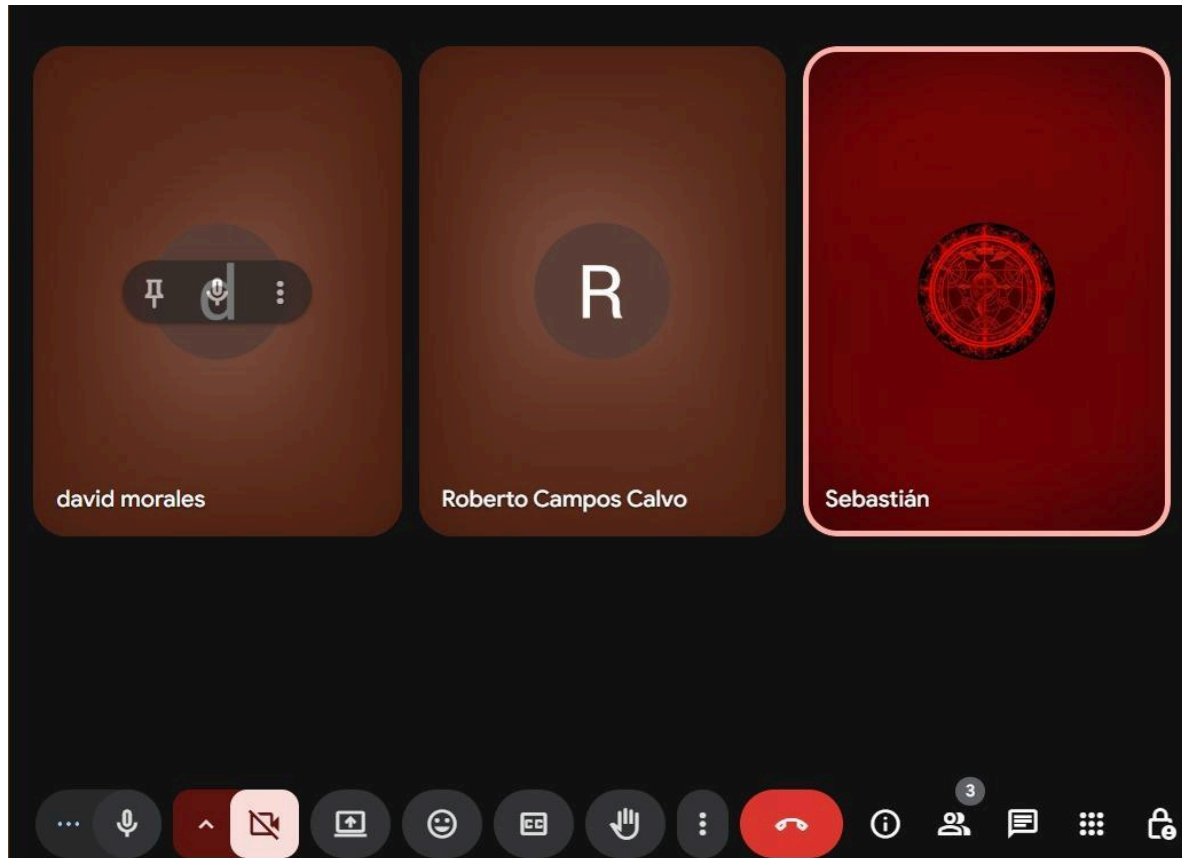
Decisiones tomadas:

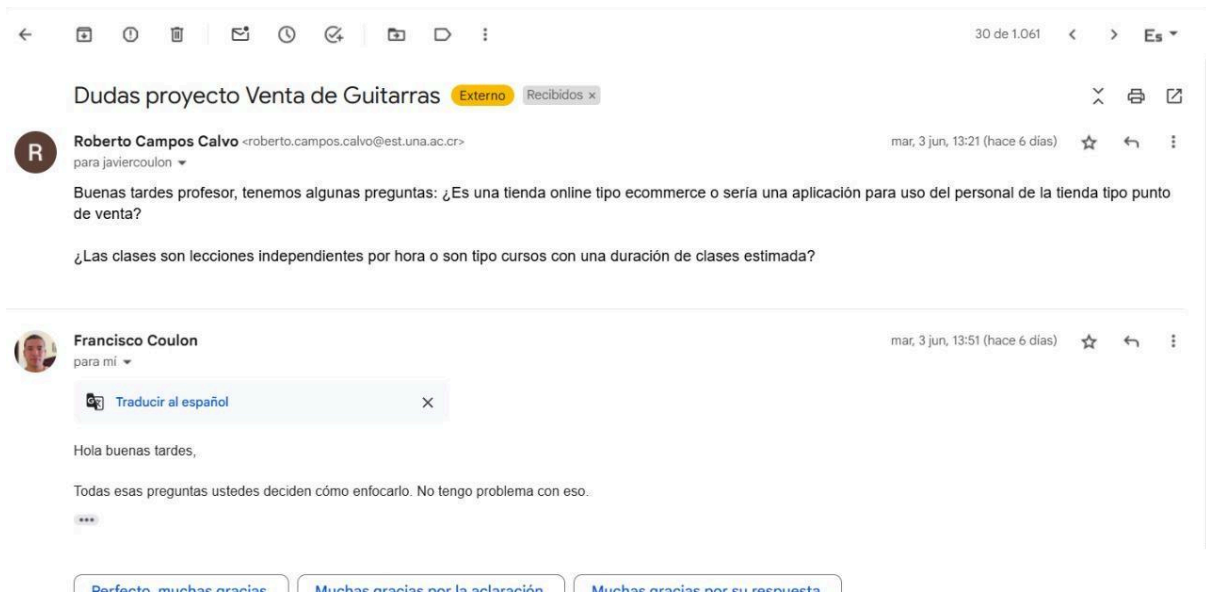
- Se definió que será un sistema punto de venta
- Establecimiento de clases no lecciones independientes

Pendientes:

- Revisar y refinar el modelo relacional inicial en la próxima reunión
- Continuar con la documentación del proyecto
- Crear el repositorio de git

Notas adicionales:





Observaciones importantes:

- El docente otorgó libertad creativa para definir el enfoque del proyecto

Sesión #2

Fecha: Lunes 9 de junio, 2025 Hora: 11:35 AM

Modalidad: Virtual (Discord)

Objetivos de la reunión:

- Refinar y finalizar el modelo relacional
- Diseñar el modelo entidad-relación
- Continuar el desarrollo de la documentación
- Primeras versiones de los scripts de la base de datos

Actividades realizadas:

1. Consulta al docente vía correo electrónico:

Se le preguntó al docente sobre puntos presentes en las instrucciones generales, específicamente sobre la carpeta que debe estar presente en el repositorio llamada Backup y acerca de la implementación del formato kebab-case, si sería solamente para el repositorio o si también debería ser implementado en las tablas y campos dentro de la base de datos.

Respuesta del docente: "El archivo backup es básicamente un archivo con todas las instrucciones SQL en un solo archivo. El formato kebab-case es solo para el repositorio"

2. Diseño del modelo entidad-relación:

Se creó el diagrama entidad-relación basado en el modelo relacional refinado.

3. Desarrollo de documentación: Se continuó con la elaboración de la documentación del proyecto
4. Se desarrollaron las primeras versiones de los scripts de la base de datos

Decisiones tomadas:

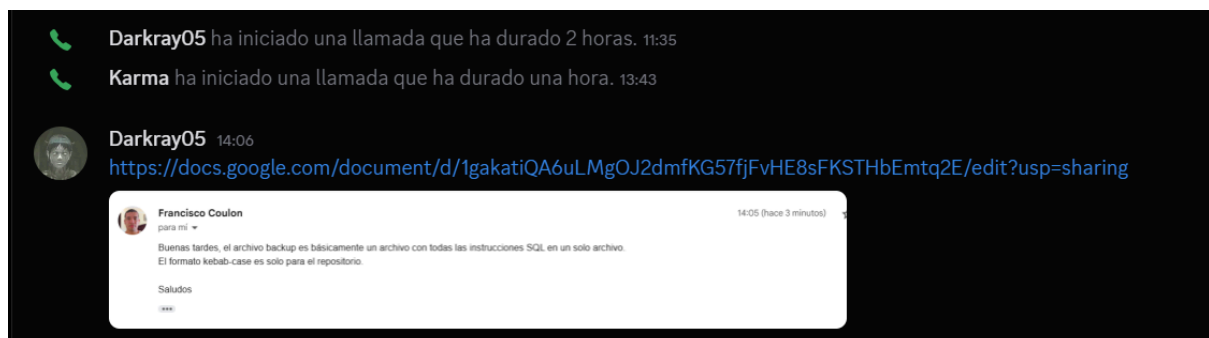
- Implementar DBeaver como herramienta principal para la gestión de base de datos y facilidad para realizar testing.

Pendientes:

- Revisar y refinar el modelo relacional inicial en la próxima reunión

Notas adicionales:

Se creó el repositorio de git el día anterior a la reunión.



Duda Proyecto
Externo
Recibidos x


Roberto Campos Calvo <roberto.campos.calvo@est.una.ac.cr>
para javiercoulon
13:55 (hace 3 horas)

Buenas tardes profesor, tenemos una duda respecto al proyecto de Diseño e implementación de bases de datos.

En las instrucciones generales entre las carpetas que debe tener el repositorio está una llamada Backup pero luego no se hace más mención, y no recordamos como vimos eso en clases, a que se refiere exactamente con ese punto es a realizar un backup o es lo de transacciones y rollbacks?

Y con lo del formato kebab-case sería solo para el repositorio o también en las tablas y campos dentro de la base de datos? Es que vimos que tanto en las presentaciones que usted nos brindó como en ejemplos el estándar es snake_case.


Francisco Coulon
para mí
14:05 (hace 3 horas)

Buenas tardes, el archivo backup es básicamente un archivo con todas las instrucciones SQL en un solo archivo. El formato kebab-case es solo para el repositorio.

Saludos

Observaciones importantes:

- Se logró avance significativo en la documentación del proyecto

Sesión #3

Fecha: Martes 10 de junio, 2025 Hora: 11:00 AM

Modalidad: Trabajo asíncrono

Objetivos de la sesión:

- Revisar Modelos ER y Relacional
- Realizar testing
- Continuar el desarrollo de la documentación

Actividades realizadas:

1. Se realizaron testings
2. Se realizaron ajustes en los modelos ER y Relacional.
3. Se realizaron ajustes en las tablas, se eliminan redundancias en producto (stock, nombre, descripción) en matrícula (id_profesor) y se simplificó la relación categoría - producto.

Decisiones tomadas:

- Realizar los cambios necesarios en la bases de datos.
- Subir todos los archivos trabajados al github.
- Se seleccionó la plantilla de canvas a utilizar.
- Utilizar también SQL Server Management Studio 21.

Pendientes:

- Revisar y refinar modelos y lógica de negocio la próxima reunión.
 - Iniciar con la presentación en canvas.
-

Notas adicionales:

Sesión #4

Fecha: Martes 11 de junio, 2025 Hora: 11:00 AM

Modalidad: Trabajo asíncrono

Objetivos de la sesión:

- Refinar y finalizar el modelo relacional
- Realizar testing
- Continuar el desarrollo de la documentación

Actividades realizadas:

1. Se realizaron testings y se identificaron ciertos comportamientos y casos en la base de datos
2. Se realizaron ajustes en los modelos ER y Relacional.
3. Se subieron todos los archivos depurados que se venían trabajando de manera local al repositorio en github.
4. Iniciar con el archivo de presentación en canvas.
5. Se realizaron las correcciones indicadas por el profesor en el repositorio de github.

Decisiones tomadas:

- Realizar los cambios necesarios en bases de datos y los modelos relacionales.

Notas adicionales:

