

INFORME BASE DE DATOS

Presentado por:

Emerson Esneyder Iles Sambony

Presentado a:

brayan Arcos

instituto tecnológico del putumayo

Area: desarrollo de datos

Mocoa putumayo Colombia

29/08/2024

Contenido

Resumen ejecutivo 3

Introducción..... 4

Metodología 5

Desarrollo del informe..... 6

Análisis y Discusión 9

Conclusiones 10

Referencias 11

Resumen ejecutivo

En este informe se presenta el diseño y la implementación de un sistema de gestión de cursos en una escuela utilizando bases de datos MySQL actualmente se han creado tablas para gestionar estudiantes, profesores y cursos, estableciendo las relaciones necesarias que nos permitirá una administración eficiente de nuestra base de datos, se abordan consultas SQL para la extracción de datos y se discuten los resultados, proporcionando un análisis sobre la eficacia del diseño propuesto.

Introducción

El desarrollo de este sistema de gestión de cursos es esencial para cualquier institución educativa gestiona la relación entre estudiantes, profesores y cursos ya que manualmente es ineficiente y se pueden cometer muchos más errores, este informe elaboró para implementar los conocimientos aprendidos durante el curso de desarrollo de base de datos

El informe cubre los aspectos fundamentales del diseño de bases de datos, incluyendo la creación de tablas, definición de relaciones y la ejecución de consultas SQL básicas y avanzadas. También se discuten prácticas de optimización y consideraciones de diseño, como la normalización.

Con este informe se pretende lograr:

- Reforzar los conocimientos aprendidos en clases.
- La Implementación de un sistema de gestión de cursos en MySQL.
- La Realización de consultas SQL eficientes para extraer información clave.
- -un diseño de base de datos escalable y eficiente.

Metodología

Herramientas Utilizadas

- MySQL como sistema de gestión de bases de datos relacionales.
- MySQL Workbench para la creación del esquema de la base de datos y la ejecución de consultas.
- Diagramas ER para la planificación del modelo de datos.

Procedimientos

1. Se diseño el esquema de la base de datos, identificando las tablas necesarias basándose en lo que queremos lograr cuando terminemos la base de datos.
2. Implementación del modelo de datos en MySQL.
3. Se ejecuto las consultas SQL necesarias para obtener la información sobre cursos, estudiantes y profesores.
4. Análisis de los resultados obtenidos y verificación de la correcta implementación del sistema.

Desarrollo del informe

Descripción de la Base de Datos

Esquema de la Base de Datos:

Se ha implementado una base de datos con las siguientes tablas:

- Cursos: Almacena el código, el nombre y la descripción de cada de los cursos.

```
CREATE TABLE cursos (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    codigo VARCHAR(10) NOT NULL,  
    nombre VARCHAR(100) NOT NULL,  
    descripcion TEXT  
);
```

- Estudiantes: Incluye el nombre, la dirección y el número de matrícula.

```
CREATE TABLE estudiantes (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100) NOT NULL,  
    direccion VARCHAR(255),  
    matricula VARCHAR(20) UNIQUE NOT NULL  
);
```

- Profesores: Contiene el nombre y la especialización de cada uno de los profesores que dictan alguno de los cursos.

```
CREATE TABLE profesores (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100) NOT NULL,  
    especializacion VARCHAR(100)  
);
```

- Curso-Profesor: Relaciona cursos con profesores, permitiendo la asignación de múltiples profesores a un curso.

```
CREATE TABLE curso_profesor (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    curso_id INT,  
    profesor_id INT,  
    FOREIGN KEY (curso_id) REFERENCES cursos(id),  
    FOREIGN KEY (profesor_id) REFERENCES profesores(id)  
);
```

- Curso-Estudiante: Relaciona estudiantes con cursos, gestionando la inscripción de estudiantes.

```
CREATE TABLE curso_estudiante (
    id INT AUTO_INCREMENT PRIMARY KEY,
    curso_id INT,
    estudiante_id INT,
    FOREIGN KEY (curso_id) REFERENCES cursos(id),
    FOREIGN KEY (estudiante_id) REFERENCES estudiantes(id)
);
```

Consultas SQL

1. Consultas Realizadas

- Consulta para un listado de todos los cursos.

```
SELECT codigo, nombre, descripcion
FROM cursos;
```

Estamos llamando de la tabla cursos los atributos código, nombre y descripción

Resultado de la consulta

	codigo	nombre	descripcion
▶	CURS001	Matemáticas Básicas	Introducción a las matemáticas para estudiante...
	CURS002	Ciencias Naturales	Estudio de la biología, física y química a nivel bá...
	CURS003	Historia Universal	Estudio de las civilizaciones antiguas y la evoluci...
	CURS004	Programación en Python	Curso introductorio de programación en Python.
	CURS005	Geografía	Curso sobre la geografía física y política del mun...
	CURS006	Inglés Intermedio	Desarrollo de habilidades en el idioma inglés a ni...
	CURS007	Educación Física	Actividades físicas y deportes.
	CURS008	Arte y Cultura	Estudio de las artes plásticas y la cultura mundial.
	CURS009	Desarrollo Web	Curso sobre HTML, CSS y desarrollo de sitios web.
	CURS010	Química Avanzada	Estudio de la química a nivel avanzado para est...

- Consulta para obtener los estudiantes inscritos en un curso específico.

```
SELECT e.nombre AS estudiante, c.nombre AS curso
FROM estudiantes e
JOIN curso_estudiante ce ON e.id = ce.estudiante_id
JOIN cursos c ON ce.curso_id = c.id
WHERE c.nombre = 'Ciencias Naturales';
```

Resultado de la consulta

	estudiante	curso
▶	María López	Ciencias Naturales

- Consulta para listado de los profesores que imparten un curso determinado.

```
SELECT p.nombre AS profesor, c.nombre AS curso
FROM profesores p
JOIN curso_profesor cp ON p.id = cp.profesor_id
JOIN cursos c ON cp.curso_id = c.id
WHERE c.codigo = 'CURS005';
```

Resultado de la consulta

	profesor	curso
▶	Javier Hernández	Geografía

- Consulta para ordenar los nombres en orden alfabético

```
SELECT e.nombre AS estudiante, c.nombre AS curso
FROM estudiantes e
JOIN curso_estudiante ce ON e.id = ce.estudiante_id
JOIN cursos c ON ce.curso_id = c.id
WHERE c.nombre = 'Programación en Python'
ORDER BY e.nombre ASC;
```

- Consulta para mostrar todos los cursos impartidos por un profesor específico.

```
SELECT c.nombre AS curso, p.nombre AS profesor
FROM cursos c
JOIN curso_profesor cp ON c.id = cp.curso_id
JOIN profesores p ON cp.profesor_id = p.id
WHERE p.nombre = 'Lucía Sánchez';
```


Análisis y Discusión

Los resultados obtenidos muestran que el sistema es capaz de gestionar múltiples estudiantes y profesores por curso, proporcionando una solución escalable y flexible las consultas SQL ejecutadas han permitido extraer información crítica para la toma de decisiones, como la cantidad de estudiantes en cada curso o la distribución de la carga académica entre los profesores.

Conclusiones

El diseño del sistema de gestión de cursos que se desarrolló en MySQL es eficiente y cumple con los objetivos establecidos. Las tablas y relaciones permiten una gestión estructurada de estudiantes, profesores y cursos, y las consultas SQL ofrecen la flexibilidad necesaria para obtener información relevante de la gestión de cursos.

Referencias

https://github.com/EmersonILES/My_sql_Emerson_ILes