

**UNIVERSIDAD NACIONAL AUTONOMA DE NICARAGUA,
LEON -FACULTAD DE CIENCIA Y TECNOLOGIA**



UNAN - León

Carrera: Ingeniería en el énfasis marketing digital y pagina web..

Modalidad: Sabatino. **Año:** III año **Parcial:** I

Asignatura: Desarrollo Web III

Integrantes: Freddy Adolfo Miranda González. 22-00976-1

Profesor(a): Ervin Ismael Montes Tellez

Grupo: 1

Fecha: 03-07-2024

“A la libertad por la Universidad “

1. Investigar y escribir una breve descripción sobre lo que es un RDBMS.

Un RDBMS, o Sistema de Gestión de Bases de Datos Relacionales (por sus siglas en inglés Relational Database Management System), es un tipo de software que facilita la creación, el mantenimiento y el uso de bases de datos relacionales. Estas bases de datos organizan los datos en tablas que están estructuradas según un modelo relacional, donde cada tabla consiste en filas y columnas.

1. Bases de Datos Relacionales (RDBMS): Utilizan tablas para almacenar datos y relaciones entre ellos. Ejemplos: MySQL, PostgreSQL, SQLite.
2. SQL (Structured Query Language): Lenguaje de consulta utilizado para interactuar con bases de datos relacionales.
3. Esquema: Estructura que define las tablas, campos y relaciones en una base de datos relacional.

Las características clave de un RDBMS incluyen:

1. **Tablas:** Los datos se organizan en tablas que consisten en filas y columnas. Cada fila representa una entidad única, y cada columna representa un atributo de esa entidad.
2. **Relaciones:** Las tablas pueden estar relacionadas entre sí mediante claves primarias y claves foráneas. Esto permite establecer vínculos entre datos de diferentes tablas, facilitando consultas y análisis complejos.
3. **Integridad referencial:** Los RDBMS garantizan la integridad referencial mediante el uso de restricciones, lo que asegura que las relaciones entre tablas se mantengan válidas y coherentes.
4. **Lenguaje SQL:** Utilizan un lenguaje estándar llamado SQL (Structured Query Language) para realizar consultas y manipular datos en la base de datos.
5. **Transacciones:** Los RDBMS soportan transacciones ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad), lo que asegura que las operaciones en la base de datos sean fiables y se mantengan incluso en caso de errores o fallos del sistema.
6. **Escalabilidad y rendimiento:** Son diseñados para manejar grandes volúmenes de datos y ofrecer un rendimiento eficiente en la gestión de consultas y actualizaciones.

Estructura de los Datos

En un RDBMS, los datos se organizan en tablas. Cada tabla está compuesta por filas y columnas. Cada fila representa una entidad única (por ejemplo, un cliente, un producto, una transacción) y cada columna representa un atributo de esa entidad (por ejemplo, nombre, dirección, cantidad).

Relaciones entre Tablas

Una de las características fundamentales de los RDBMS es la capacidad de establecer relaciones entre diferentes tablas. Esto se logra mediante el uso de claves primarias y claves foráneas. Por ejemplo, en un sistema de ventas, podríamos tener una tabla de

clientes y una tabla de pedidos. La tabla de pedidos podría referenciar a la tabla de clientes utilizando la clave primaria del cliente como una clave foránea para establecer quién realizó cada pedido.

Lenguaje SQL

SQL (Structured Query Language) es el lenguaje estándar utilizado para interactuar con los RDBMS. Permite realizar consultas para recuperar datos, insertar nuevos registros, actualizar registros existentes y eliminar registros. Algunos ejemplos de operaciones SQL son:

- **Consulta básica:** `SELECT * FROM clientes WHERE Ciudad = 'Buenos Aires';`
- **Inserción de datos:** `INSERT INTO clientes (Nombre, Apellido, Edad, Ciudad) VALUES ('Ana', 'Martínez', 28, 'Madrid');`
- **Actualización de datos:** `UPDATE clientes SET Edad = 31 WHERE Nombre = 'Juan';`
- **Eliminación de datos:** `DELETE FROM clientes WHERE ID = 3;`

Integridad Referencial y Transacciones

Los RDBMS garantizan la integridad referencial, lo que significa que las relaciones entre tablas se mantienen válidas y coherentes. Esto se logra mediante el uso de restricciones como claves primarias y foráneas.

Las transacciones son operaciones que aseguran la consistencia de los datos. Una transacción es una secuencia de operaciones que se ejecutan como una unidad completa. Las propiedades ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad) garantizan que incluso en caso de errores o fallos del sistema, las transacciones se completen de manera segura.

Ejemplos de Sistemas RDBMS

Existen numerosos sistemas RDBMS populares, cada uno con sus propias características y aplicaciones específicas:

- **MySQL:** Un sistema de gestión de bases de datos de código abierto muy popular, ampliamente utilizado en aplicaciones web y de comercio electrónico.
- **PostgreSQL:** Otro sistema de gestión de bases de datos de código abierto conocido por su robustez, características avanzadas y soporte para datos geoespaciales.
- **Oracle Database:** Un sistema de gestión de bases de datos comercial ampliamente utilizado en grandes empresas y aplicaciones empresariales.
- **SQL Server:** Desarrollado por Microsoft, es conocido por su integración con otras tecnologías de Microsoft y es popular en entornos Windows.
- **SQLite:** Un motor de base de datos ligero y autónomo que se utiliza comúnmente en aplicaciones móviles y pequeñas aplicaciones locales.

1. MongoDB Atlas: Servicio de base de datos MongoDB en la nube. 2. MongoDB Compass: Herramienta GUI para interactuar con MongoDB. 3. mongoose: Biblioteca de Node.js para modelar datos en MongoDB.

2. Crear una tabla simple en una base de datos SQLite (puede utilizarse un archivo local para evitar configuraciones complejas).

```
views > usuarios.ejs > html > body > form.container > input#nombre
2  <html lang="en">
9  <body>
10  <h1>hola</h1>
11  <form action="/ingresar" method="post" class="container">
12
13    <label for="nombre">Nombre:</label>
14    <input type="text" name="nombre" id="nombre">
15
16    <label for="apellido">Apellido:</label>
17    <input type="text" name="apellido" id="apellido">
18
19    <label for="edad">Edad:</label>
20    <input type="number" name="edad" id="edad">
21
22    <button type="submit">Enviar</button>
23
24  </form>
```

```
app.post('/ingresar', (req, res) => {
  const datos = req.body
  console.log(datos)

  if(datos.nombre || datos.apellido || datos.edad){
    User.create({
      name: datos.nombre,
      lastName: datos.apellido,
      age: datos.edad
    })
  } else {
    console.log('error')
  }
  res.render('index')
});
```

Investigar y escribir una breve descripción sobre lo que es una base de datos NoSQL y MongoDB.

1. NoSQL: Base de datos no relacional, utiliza documentos, grafos, columnas, etc., para almacenar datos. Ejemplo: MongoDB.
2. MongoDB: Una base de datos NoSQL basada en documentos.

3. Documentos: Registros en MongoDB que se almacenan en formato JSON o BSON.

Características de las bases de datos NoSQL:

1. **Esquema flexible:** No requieren un esquema fijo como en las bases de datos relacionales. Los datos pueden ser almacenados en diferentes formatos, como documentos, grafos, columnas o pares clave-valor.
2. **Escalabilidad horizontal:** Son diseñadas para escalar horizontalmente de manera eficiente, distribuyendo los datos en múltiples servidores y soportando grandes volúmenes de datos y altas cargas de trabajo.
3. **Alta disponibilidad y rendimiento:** Generalmente están optimizadas para ofrecer un alto rendimiento y disponibilidad, permitiendo operaciones rápidas de lectura y escritura.
4. **Modelos de datos variados:** Pueden soportar diversos modelos de datos como documentos (document-oriented databases), grafos (graph databases), columnares (column-family stores) y bases de datos de pares clave-valor (key-value stores).
5. **No requieren lenguaje SQL:** Aunque algunos sistemas NoSQL tienen soporte para consultas similares a SQL, muchos utilizan interfaces y lenguajes de consulta propios.

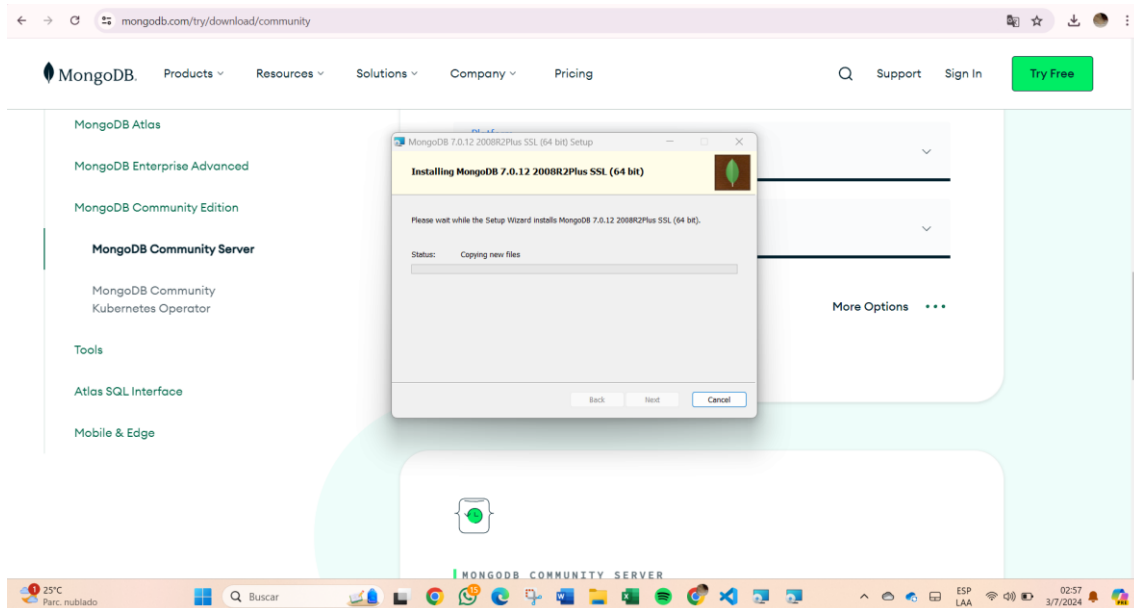
MongoDB:

MongoDB es uno de los sistemas de base de datos NoSQL más populares y ampliamente utilizados. Se clasifica como una base de datos orientada a documentos, lo que significa que almacena datos en documentos JSON (BSON en MongoDB, que es una extensión binaria de JSON). A continuación, algunas características destacadas de MongoDB:

- **Documentos:** Los datos se representan como documentos BSON, que son estructuras de datos flexibles y jerárquicas similares a JSON.
- **Escalabilidad:** MongoDB es altamente escalable y está diseñado para funcionar en clústeres distribuidos, lo que facilita la escalabilidad horizontal añadiendo más servidores.
- **Consultas y Agregaciones:** Utiliza un lenguaje de consulta flexible y poderoso llamado MongoDB Query Language (MQL), que admite consultas, índices y operaciones de agregación complejas.
- **Replicación y Alta Disponibilidad:** Ofrece replicación automática con failover para asegurar la alta disponibilidad y la tolerancia a fallos.
- **Indexación:** Soporta índices secundarios para optimizar las consultas y mejorar el rendimiento de las operaciones de lectura.

MongoDB se utiliza comúnmente en aplicaciones web modernas, plataformas de análisis de datos, aplicaciones móviles y otros casos de uso donde la flexibilidad en el esquema y la escalabilidad son prioritarias. Es una opción popular debido a su facilidad de uso, rendimiento escalable y comunidad activa de usuarios y desarrolladores.

2. Instalar MongoDB y la herramienta de línea de comandos `mongo`.



Parte 3: Conectarse a MongoDB desde Node.js

```
const app = express();

app.set('view engine', 'ejs');
app.set('views', './views');

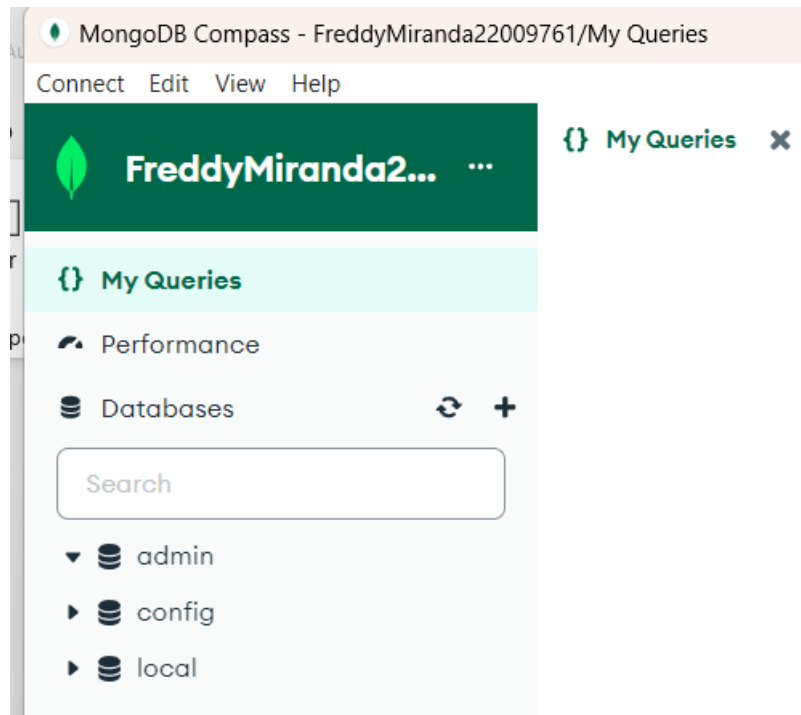
//middleware
app.use(express.static('public'));

app.use(express.json());
app.use(express.urlencoded({extended: true}));

//base de datos
mongoose.connect('mongodb://127.0.0.1:27017/Users');

//schema
```

1. Crear una cuenta en MongoDB Atlas y configurar un clúster



2. Instalar `mongoose` en su proyecto Node.js:

```
//base de datos
mongoose.connect('mongodb://127.0.0.1:27017/Users');

//schema
const userSchema = new Schema({
  name:String,
  lastName:String,
  age:Number,
});

//model
const User = mongoose.model('User',userSchema)

// Rutas
```

1. **MongoDB Atlas:** Servicio de base de datos MongoDB en la nube.
2. **MongoDB Compass:** Herramienta GUI para interactuar con MongoDB.
3. **mongoose:** Biblioteca de Node.js para modelar datos en MongoDB.
1. **Schema:** Define la estructura de un documento en una colección de MongoDB.
2. **Model:** Un "modelo" basado en un esquema para interactuar con documentos.

1. **Crear un esquema y un modelo para una colección de usuarios.**
2. **Insertar, buscar, actualizar y eliminar documentos.**

Se encuentra en el archivo RAR de github