

mongodb

non sql database

stored as json format

MongoDB is a document database which is often referred to as a non-relational database

It's used in applications that require fast, efficient handling of large, unstructured, or semi-structured datasets and where schema flexibility and horizontal scaling are priorities.

ADV

1. Flexible Data Structure

MongoDB doesn't require a fixed structure for your data.

You can store different kinds of data together, and easily adjust the structure as your needs change, which is perfect for apps with evolving data.

2. High Speed for Writing Data

It's optimized for fast data writing, which makes it ideal for apps that need to save a lot of data quickly, like social media platforms or sensor data from IoT devices.

3. Document-Based Storage

MongoDB stores data in a format similar to JSON, which is easy to work with in web and mobile apps, especially if you're using JavaSc

DIS ADV

High Memory Usage

Limited Support for Large and Complex Transactions

Higher Costs with Large Datasets

What is JSON?

JSON stands for JavaScript Object Notation

JSON is a lightweight data interchange format

JSON is language independent

JSON is "self-describing" and easy to understand

\REACT

React is a JavaScript library for building user interfaces.

React is used to build single-page applications.

React allows us to create reusable UI components.

React is primarily used for building user interfaces, particularly for single-page applications

ADV

Component-Based Architecture:

rich tooling

flexibility

DISADV

Not a Complete Framework:

jsx syntax

NODE

use to connect the server http

running web applications outside the client's browser.

It is faster than other server-side technologies and ideal for building real-time applications.

Node.js is also an excellent choice for building microservices-based architecture

EXPRESS

Express is a node.js web application framework that provides broad features for building web and mobile applications.

It is used to build a single page, multipage, and hybrid web application.

It's a layer built on the top of the Node.js that helps manage servers and routes.

Express is a back-end web application framework of node.js, and with the help of express, we can create an API very easily.

MIDDLEWARE

Middleware is a software layer that connects the operating system to applications

Authentication and Authorization

Error Handling

Modifying Requests or Responses

other framework in js

angular

backbone

vue.js

FRONTEND

Frontend refers to the part of a web application or website that users interact with directly.

It encompasses everything that users see and experience in their web browsers, including the layout, design, graphics, and user interface (UI) elements.

FRONTEND tools

react
react router
bootstrap

BACKEND

Backend refers to the server-side part of a web application or website that handles the data processing, business logic, and server management. It is not directly visible to users, but it plays a crucial role in ensuring that the frontend operates smoothly and efficiently

BACKEND tools

mongodb
express

SELF INTRO

My name is Adarsh, and I recently completed my Bachelor's degree in Computer Applications (BCA) with 85% in sree narayana guru collage .

I also completed my Plus Two in Commerce with 65% and achieved 85% in my 10th grade.

after my graduation i completed a 6 month training program in mern at the time of my training i gained hand on experince in mongodb Express react and node

at the time of internship i done a project doctor booking application Developed a full-stack web application for booking doctor appointments using the MERN stack

Integrated user authentication with JWT for secure login and role based access control. Designed a responsive front-end UI using React and Ant Design, ensuring a seamless user experience across desktop and mobile devices.

version check

```
npm list express -express =v5  
npm list react -react= v 18.3  
npm --version or npm -v node =21.2  
mongo db 8.0  
mongoose =8.0
```

Definition:

Customer Relationship Management (CRM) is a strategy and set of processes that organizations use to manage and analyze interactions with customers, clients, and sales prospects.

The goal is to improve customer satisfaction, loyalty, and retention.

Key Components:

1. Customer Data Management: Collecting and storing customer information.
2. Sales Force Automation: Automating sales processes and tracking leads.
3. Marketing Automation: Automating marketing campaigns and lead generation.
4. Customer Service and Support: Providing timely and effective customer support.
5. Analytics and Reporting: Analyzing customer data to inform business decisions.

Benefits:

1. Enhanced customer satisfaction and loyalty.
2. Improved sales and revenue growth.
3. Increased efficiency and productivity.
4. Better decision-making through data analysis.
5. Competitive advantage through personalized customer experiences.

CRM Process:

1. Customer Identification: Identifying potential customers.
2. Customer Acquisition: Converting leads into customers.
3. Customer Retention: Maintaining customer relationships.
4. Customer Development: Upselling and cross-selling.
5. Customer Feedback: Collecting and acting on customer feedback.

CRM Tools and Technologies:

1. Salesforce
2. HubSpot
3. Zoho CRM
4. Microsoft Dynamics
5. Oracle CRM

CRM Best Practices:

1. Define clear customer segments and personas.
2. Develop a customer-centric culture.
3. Implement a CRM system that integrates with existing systems.
4. Provide ongoing training and support for CRM users.
5. Continuously monitor and analyze customer data.

Common CRM Interview Questions:

1. What is CRM, and how does it benefit organizations?
2. How would you implement a CRM system in our organization?
3. How do you ensure data quality and integrity in a CRM system?
4. Can you describe a successful CRM project you've managed?
5. How do you measure the effectiveness of a CRM strategy?

Example Answers:

1. "CRM is a strategic approach to managing customer interactions, providing a

360-degree view of customer relationships."

2. "To implement a CRM system, I would conduct stakeholder analysis, define requirements, select a vendor, and provide training and support."

3. "Data quality is ensured through data validation, data cleansing, and regular backups."

4. "In my previous role, I managed a CRM project that resulted in a 25% increase in sales revenue."

5. "CRM effectiveness is measured through metrics such as customer satisfaction, retention rates, and sales growth."

Tips for CRM Interviews:

1. Review the organization's CRM system and processes.
2. Emphasize customer-centricity and relationship-building.
3. Highlight data analysis and insights.
4. Showcase technical skills and knowledge of CRM tools.
5. Demonstrate strategic thinking and problem-solving.

Your introduction

Hello, I'm Adarsh, i complete my (BCA) graduate from Sree Narayana Guru College with 80%. Before my degree, I completed my Higher Secondary Education in Commerce with 60% and achieved 85% in my SSLC.

After graduation, I completed a 6-month MERN stack program, where I gained hands-on experience in MongoDB, Express.js, React, and Node.js. During this training, I developed a Doctor Appointment Booking application using the MERN stack. In this project, I built the front end with React, the back end with Node.js and Express, and used MongoDB as the database. I also have experience working with Git for version control and Postman for API testing.

During my college years, I developed a Motor Driving School Management System using VB.NET and MySQL. I'm excited to continue growing as a full-stack developer and contribute to meaningful projects.

MongoDB is a NoSQL database that stores data in a JSON-like format, which allows it to handle various types of unstructured data efficiently.

Instead of using traditional rows and tables, MongoDB organizes data in collections of documents, making it highly flexible and scalable for modern applications.

Express.js is a fast and lightweight server-side framework that operates on a Node.js server.

It simplifies the handling of HTTP requests and responses, allowing developers to easily map URLs to specific server-side operations.

Node.js uses JavaScript as its main programming language, allowing developers to use the same language for both front-end and server - side development.

It includes NPM, a large collection of open-source libraries and packages, making it easy for developers to manage dependencies and add third-party libraries to their applications.

React.js is an open-source JavaScript library developed by Facebook for building fast, interactive and user-friendly interfaces, especially for single-page applications.

It allows developers to create reusable UI components, making it easier to manage and update the user interface efficiently.

It is maintained by Facebook and a community of developers, and it is widely used for creating dynamic and interactive web applications.