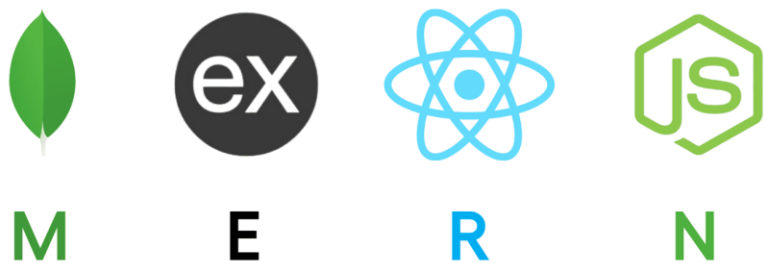




# Aula 01.1:

Introdução a disciplina

# O que é o MERN Stack?



O MERN Stack é uma coleção de tecnologias que permite construir **aplicações web completas**, desde o banco de dados até a interface do usuário.

Utiliza **JavaScript** em todas as camadas, facilitando o desenvolvimento e a manutenção.

As tecnologias principais são: **MongoDB**, **Express**, **React** e **Node.js**.



# MongoDB

**O que é?** MongoDB é um banco de dados **NoSQL** orientado a documentos, o que significa que ele armazena dados em formato semelhante ao JSON.

**Por que usar?** É altamente escalável e flexível, permitindo armazenar grandes volumes de dados de forma eficiente.

**Vantagens:** Facilidade de uso, alta performance, e compatibilidade com outras tecnologias da stack **MERN**.



# Express.js

**O que é?** Express é um framework web minimalista para **Node.js**, projetado para construir **APIs** e aplicações web robustas.

**Por que usar?** Simplifica o desenvolvimento de servidores e manipulação de rotas, com uma sintaxe clara e intuitiva.

**Vantagens:** Flexibilidade, extensibilidade com middlewares, e uma grande comunidade de suporte.

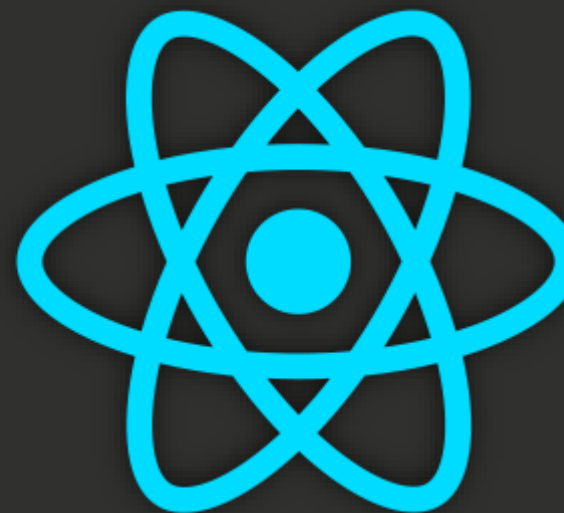


# React

**O que é?** React é uma biblioteca JavaScript desenvolvida pelo **Facebook** para construir interfaces de usuário (**UIs**) interativas e reutilizáveis.

**Por que usar?** Facilita a criação de **componentes** de UI que podem ser reutilizados em diferentes partes da aplicação, melhorando a manutenção do código.

**Vantagens:** Performance eficiente através do **Virtual DOM**, uma comunidade ativa, e um grande ecossistema de ferramentas e bibliotecas.



**React JS**



# Node.js

**O que é?** Node.js é um ambiente de execução JavaScript no servidor, baseado no motor V8 do Google Chrome.

**Por que usar?** Permite a construção de aplicações de alta performance e escaláveis, com um modelo de **I/O não bloqueante**.

**Vantagens:** Utilização de **JavaScript** em todo o **stack**, grande quantidade de módulos disponíveis através do **npm**, e adequação para aplicações em tempo real.



# Conteúdo programático



## Aula 01 – Desenvolvimento de API Rest com Node.js

- Criação da API
- Testes com Insomnia
- Conexão ao MongoDB

## Aula 02 – Desenvolvendo o CRUD da API

- Cadastro de dados na API (POST)
- Excluindo dados da API (DELETE)
- Alterando dados na API (PUT)
- Buscando registros únicos na API (GET)



# Conteúdo programático

## Aula 03 – Avançando com MongoDB

- Documentos aninhados no MongoDB
- Hospedagem do banco no MongoDB Atlas

## Aula 04 – Autenticação na API

- Login na API com JWT (Json Web Token)
- Middlewares de autenticação
- Documentação da API





# Conteúdo programático



## Aula 05 – Fundamentos do React com Next.js

- Rotas e componentes
- Props e children
- Hooks do React
- Formulários

## Aula 06 – Projeto front-end (base do site no React)

- Conteúdo do site
- Estilos globais e CSS Modules
- Responsividade para dispositivos móveis



# Conteúdo programático



## Aula 07 – Consumo da API no React

- Requisições GET, POST, PUT e DELETE com Axios
- Login e autenticação
- Context API

## Desenvolvimento dos projetos / Revisão de conteúdo

- Revisão para A.I
- Projeto Integrador (PI)
- Laboratórios de Práticas (LP)



# Métodos de avaliação

**1 – Atividades / Trabalhos**

**2 – Projeto Integrador (PI)**

**3 – Avaliação Integradora (AI)**

**4 – Maratona de programação (MRT)**

**5 – Laboratório de Práticas (LP)**





# Aula 01

Introdução a disciplina