Nội dung trình bày



- ► Nodejs là gì?
- ► Tại sao lại dùng **nodejs**?
- ▶ Bắt đầu với **nodejs** như thế nào?
- ▶ Ứng dụng của **nodejs** trong thực tế.
- ▶ Q & A

NodeJS là gì?

- NodeJS là mã nguồn mở chạy trên môi trường V8

 JavaScript runtime (một trình thông dịch JavaScript chạy

 cực nhanh trên trình duyệt Chrome).
- Learn Once, Write Anywhere: NodeJS giúp các nhà phát triển xây dựng các ứng dụng một cách đơn giản và dễ dàng mở rộng.

Điểm mạnh của NodeJS

- IO hướng sự kiện bất đồng bộ giúp xử lý nhiều yêu cầu đồng thời.
- Đáp ứng được những yêu cầu về thời gian thực.
- Có tốc độ cực rất nhanh, đáp ứng được nhu cầu sử dụng của khách truy cập 'khổng lồ' trong thời gian ngắn.
- Dùng JavaScript **cho cả Client & Server**, một ngôn ngữ lập trình rất dễ học.
- Nhiều modules rất mạnh mẽ và vẫn đang tiếp tục phát triển rất nhanh.
- Có một cộng đồng lớn mạnh, có nhiều mã nguồn mở được chia sẻ rộng rãi.
- Tương thích với nhiều thiết bị, nhiều hệ điều hành như MacOS, Window, Linux,

Điểm yếu của NodeJS

- NodeJS không phù hợp với các tác vụ đòi hỏi nhiều CPU mà chỉ phù hợp với những I/O như máy chủ web.
- Nếu bạn có một web hosting dùng chung, sẽ rất khó khăn nếu bạn tải lên một ứng dụng NodeJS. VPS và Dedicated server là một sự lựa chọn tốt hơn nhiều.

NodeJS rất dễ học và ứng dụng thực tế

As an asynchronous event-driven JavaScript runtime, Node.js is designed to build scalable network applications. In the following "hello world" example, many connections can be handled concurrently. Upon each connection, the callback is fired, but if there is no work to be done, Node.js will sleep.

```
const http = require('http');
const hostname = '127.0.0.1';
const port = 3000;
const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World');
});
server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```



JavaScript vs TypeScript



TS

Great for beginners

For high salary

Loved by programmers

For web development

For enterprise projects





















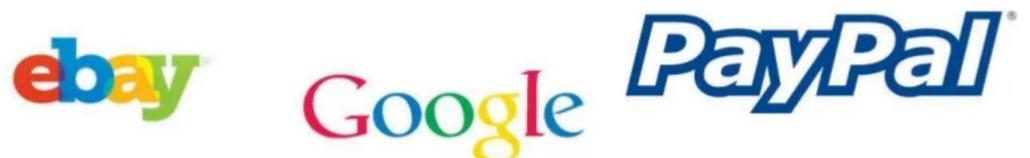
Ứng dụng của node.js







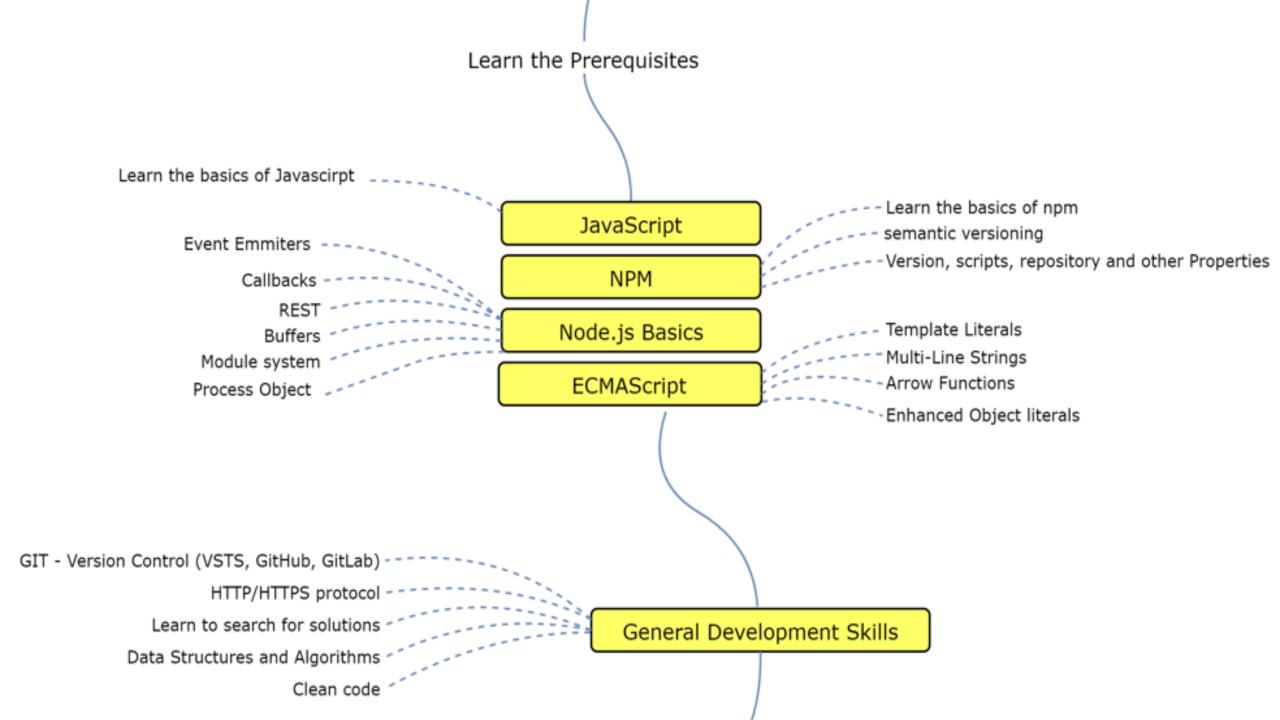


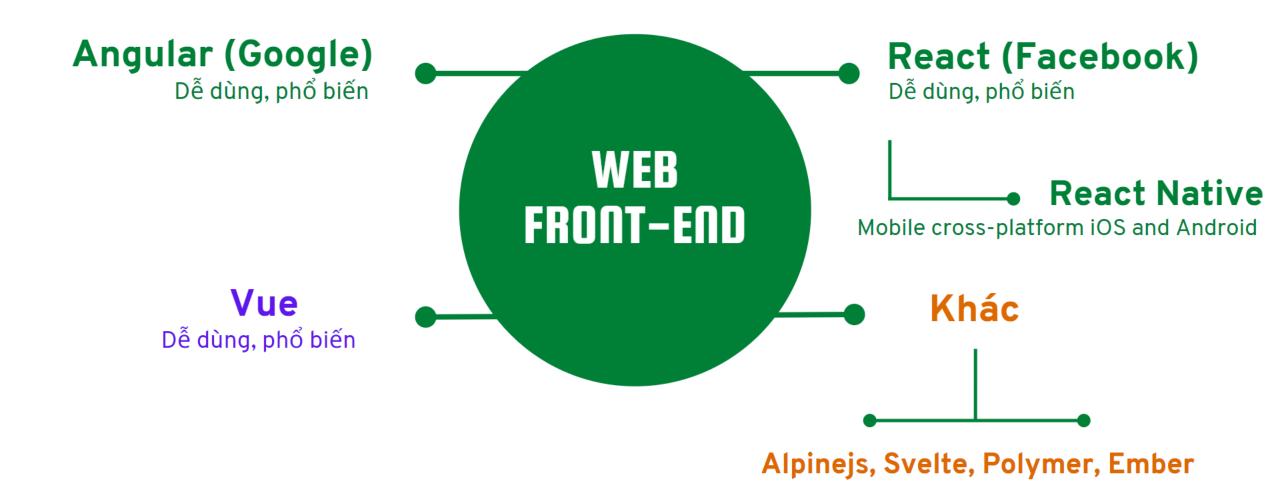




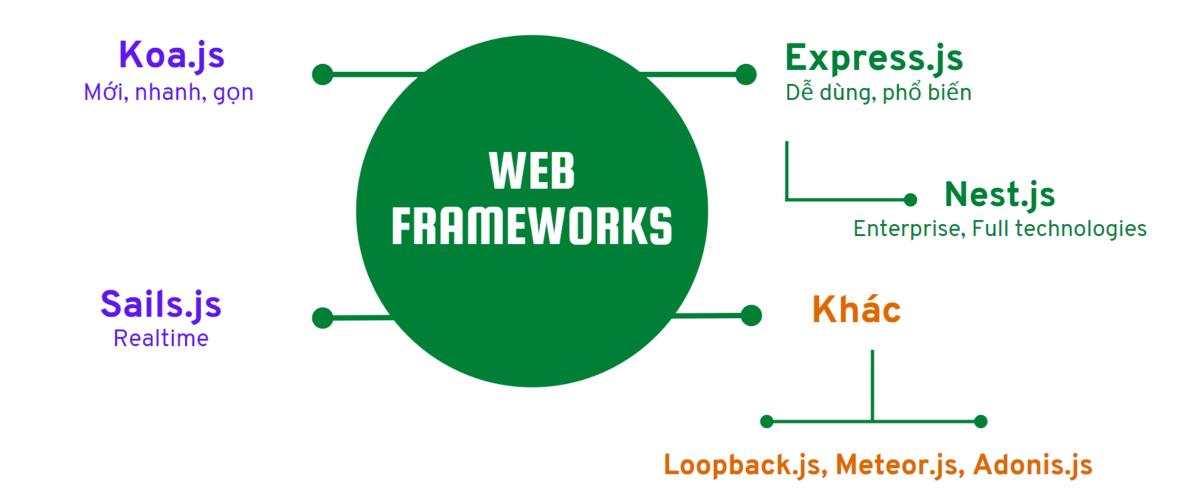


Bắt đầu với NodeJS như thế nào?

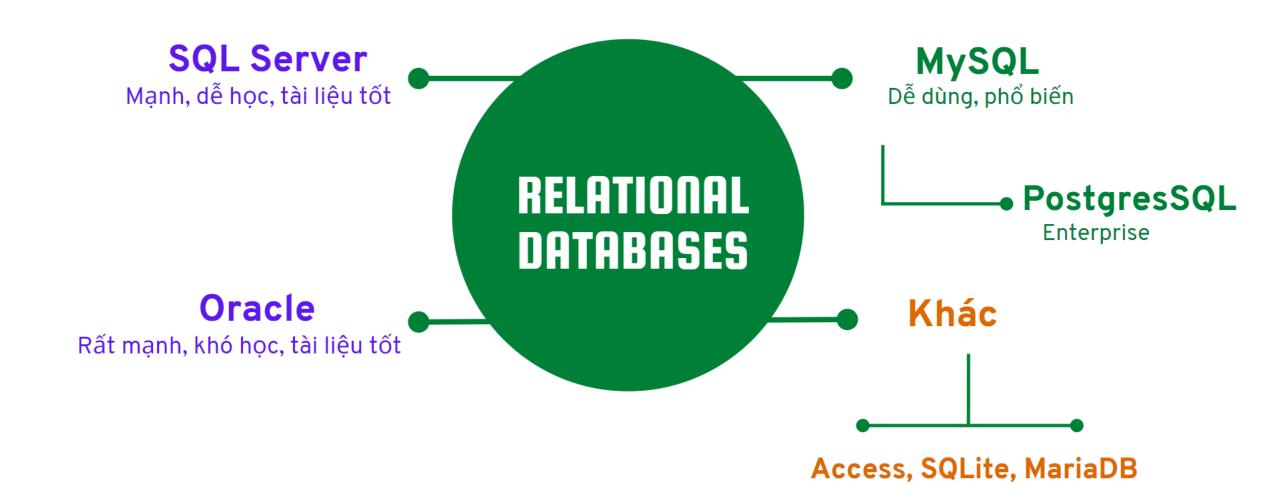




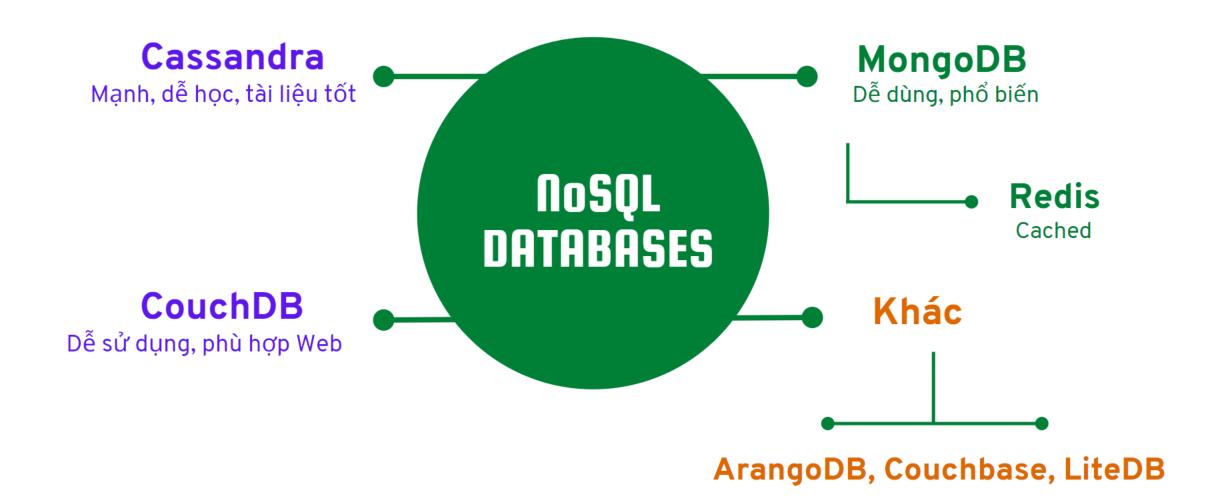
START WITH WEB FRONT-END



START WITH WEB FRAMWORKS



START WITH DATABASES

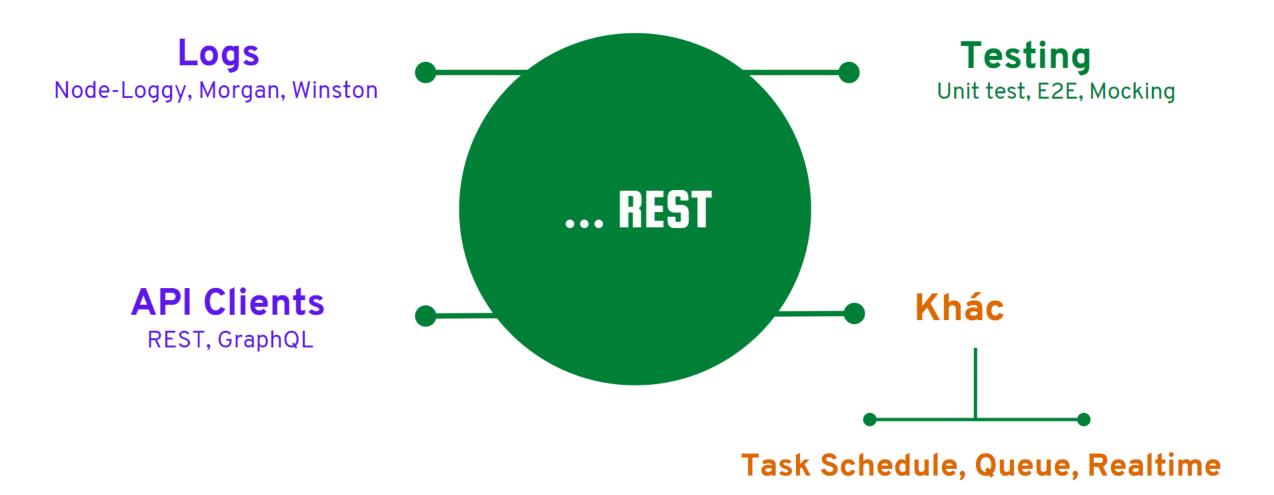


START WITH DATABASES



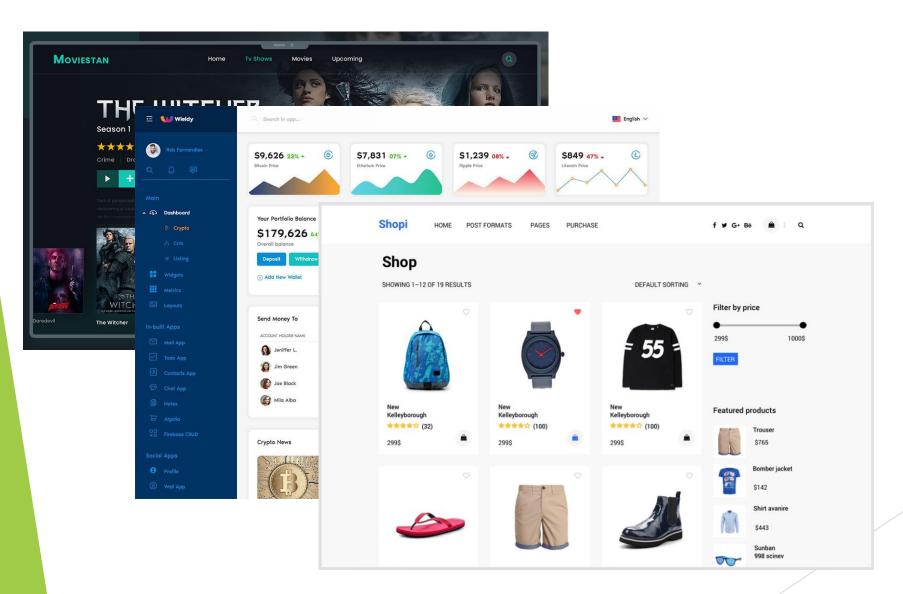
FaunaDB, RavenDB

START WITH DATABASES



... REST, BUT IMPORTANT

Nodejs với Web (Expressjs)



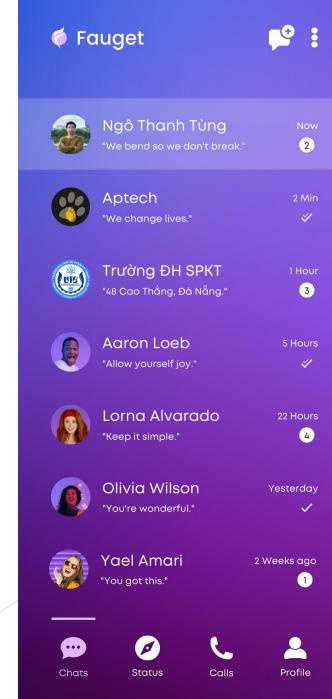
Node.js với Realtime



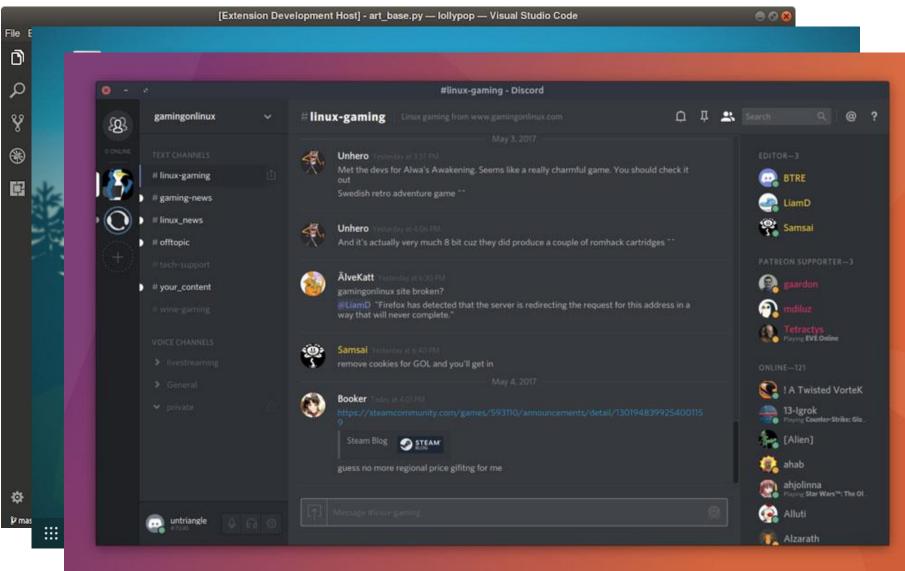








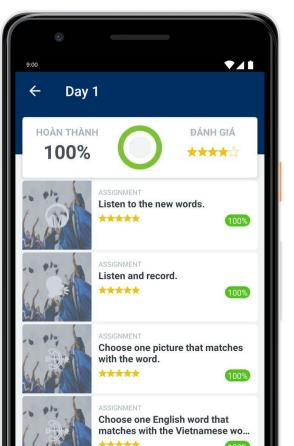
Node.js với Desktop Apps (Electron)

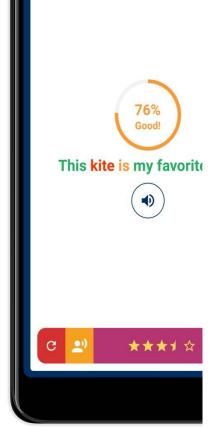


Node.js với Mobile (React Native)



Bài tập phong phú, đa dạng





Chấm điểm ph bằng A.I



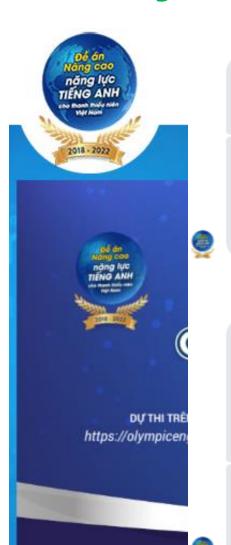
Mở rộng vốn từ vựng với Thư viện

Node.js với W



```
const { test, expect } = require('@playwright/test');
    test.describe('Basic tests', () \Rightarrow {
      test('Basic test 1', async (\{page\}) \Rightarrow \{
        await page.goto('https://aptech-danang.edu.vn/');
        await expect(page).toHaveTitle(/Softech1 Aptech2/);
      });
      test('Basic test 2', async (\{page\}) \Rightarrow \{
10
        await page.goto('https://aptech-danang.edu.vn');
        const title = page.locator('title').first();
11
        await expect(title).toContainText('APTECH');
13
      });
    });
14
15
```

Node.js với Chatbot



If these shoes don'

- A. shift
- B. transform
- C. exchange
- D. alter

Correct V +10

- → Điểm hiện tại: 1
- ⊹ Thời gian còn lạ

* Question 2. Whi youth" first implen

A. 2000

B. 2005

C. 2007

D. 2012

Send API Basics

All messages are sent by submitting a POST request to the Send API with your page access token appended to the URL query string:

```
https://graph.facebook.com/v14.0/me/messages?access_token=<PAGE_ACCESS_TOKEN>
```

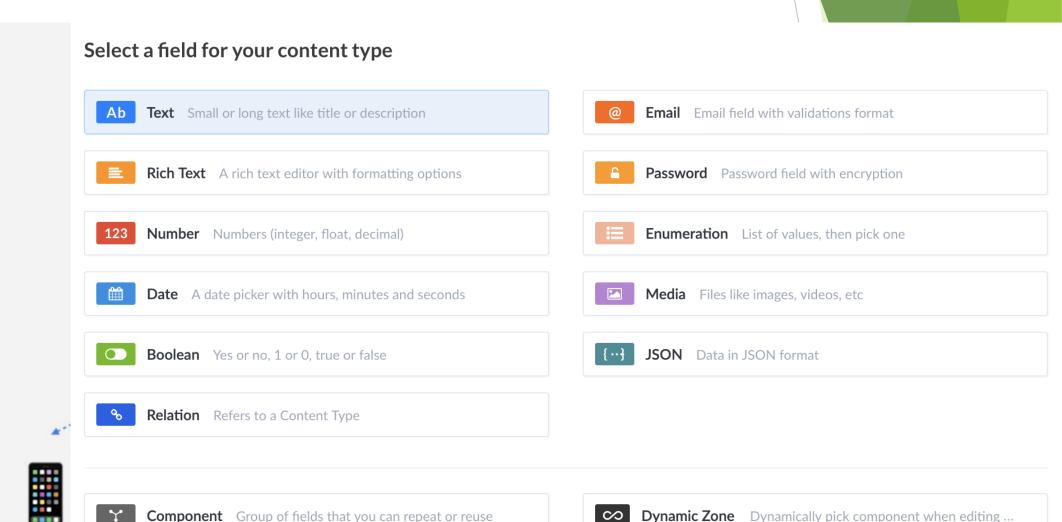
The body of the HTTP request is sent in JSON format and requires three properties:

- messaging type: Identifies the purpose of the message send.
- · recipient: Identifies the intended recipient of the message.
- message: Defines the message to be sent.

Here's a simple example of the body of a request to send a text message:

```
"messaging_type": "<MESSAGING_TYPE>",
"recipient":{
    "id":"<PSID>"
},
"message":{
    "text":"hello, world!"
}
```

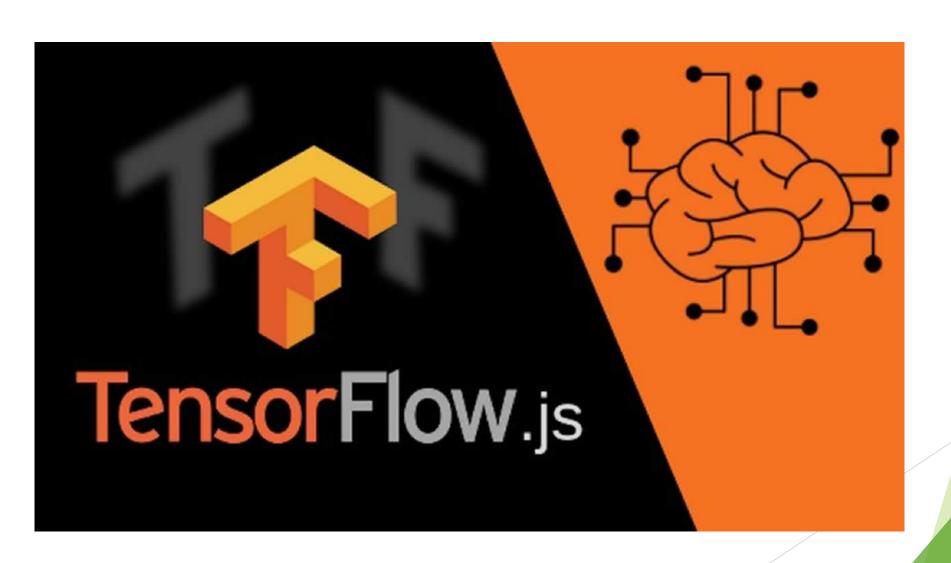
Node.js với CMS



Node.js với Cloud



Node.js với A.I / Machine Learning



Cảm ơn

