



Rafael Ribeiro
Head de Engenharia

Introdução ao NodeJS

Módulo X



Desenvolvedores NodeJS

São desenvolvedores **Javascript**, pois NodeJS não é uma linguagem de programação



NodeJS não depende do browser

Utilizado para criação de páginas dinâmicas, conexão com banco de dados, acesso ao sistema operacional, criação de CLIs (utilitários de linha de comando) etc



Afinal, o que é NodeJS?

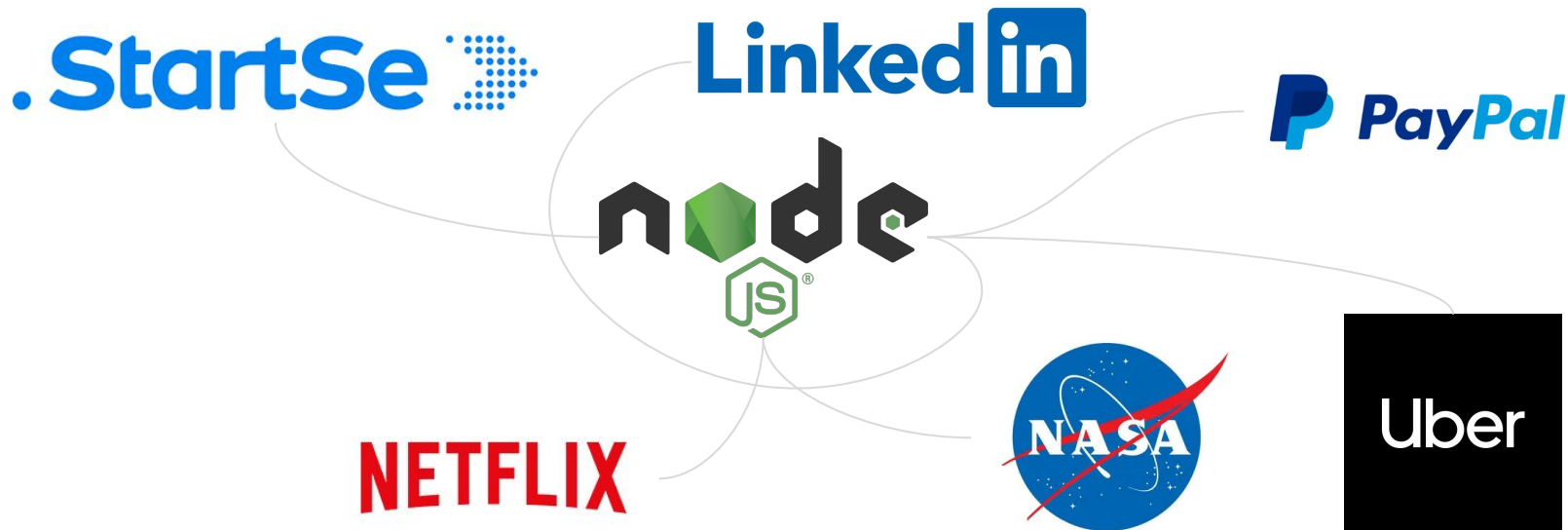
É uma plataforma voltada para o desenvolvimento de aplicações utilizando javascript



NodeJS

Criado por Ryan Dahl (2009), possibilitou rodarmos o javascript independente do browser. Logo podemos trabalhar com front-end e back-end utilizando uma linguagem

Algumas Empresas que Trabalham com NodeJS





Por que grandes empresas utilizam **NodeJS** ?



Benefícios em utilizar o **NodeJS**

Ter um time de engenharia focado em uma linguagem utilizada para front-end e back-end



Benefícios em utilizar o NodeJS


Aplicações leves e escaláveis. Veja um exemplo de resultado com NodeJS no PayPal:

- O aplicativo foi desenvolvido quase **duas vezes mais rápido** com **menos pessoas**
- Com **33% menos linhas de código** e **40% menos arquivos** (em comparação com o aplicativo anterior baseado em Java).
- **Dobrou** o número de *requests* atendidas por segundo e ao mesmo tempo diminuiu o tempo médio de resposta em 35%.



Javascript

Possui uma arquitetura não bloqueante de thread única (*single thread*), baseada em eventos



Arquitetura do Javascript

Call Stack

Responsável por empilhar as chamadas de funções

Callback Queue

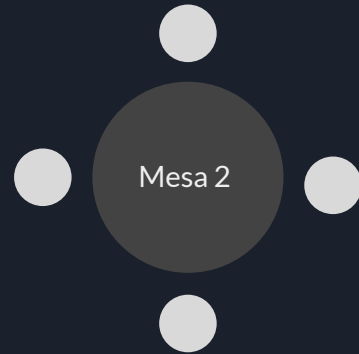
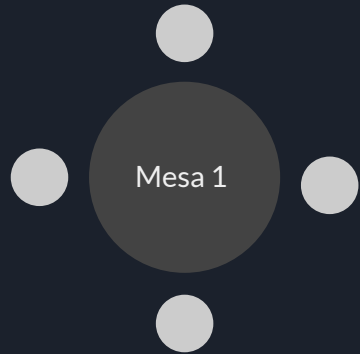
Responsável por empilhar os callbacks (funções enviadas como parâmetro para funções assíncronas)

Event Loop

Responsável por checar continuamente se algum evento assíncrono foi disparado para posteriormente executar o callback.

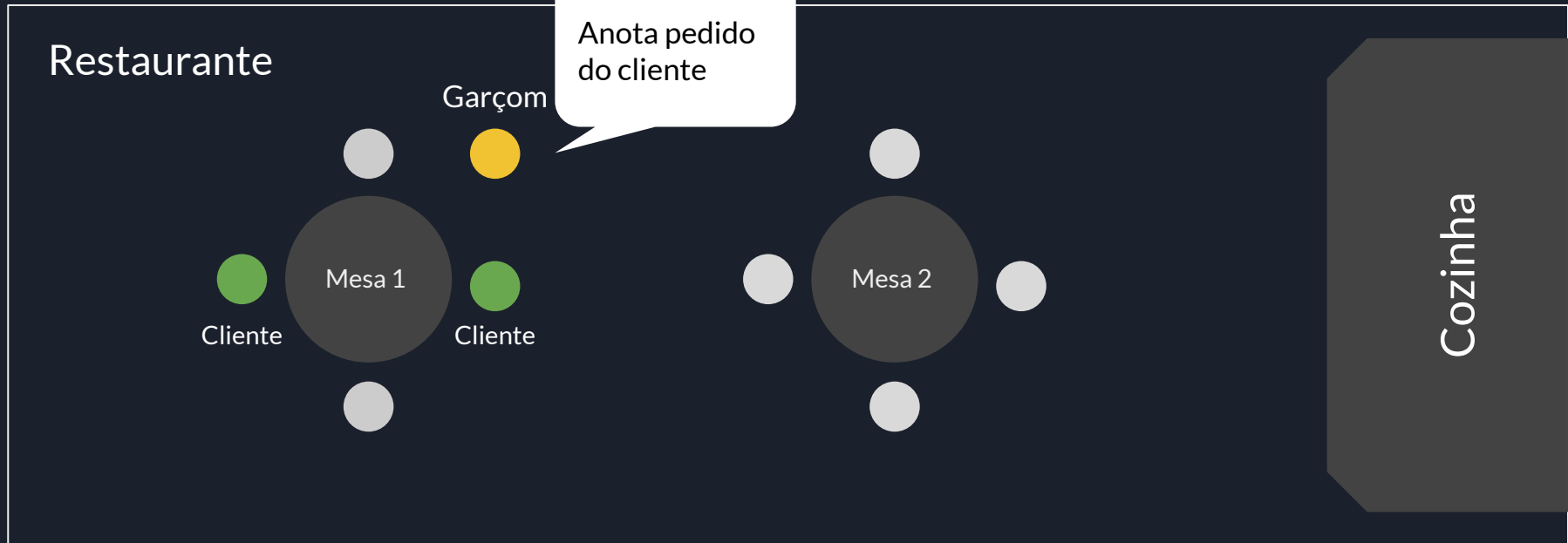
Entendendo a arquitetura single-thread não bloqueante

Restaurante



Cozinha

Entendendo a arquitetura single-thread não bloqueante



Entendendo a arquitetura single-thread não bloqueante



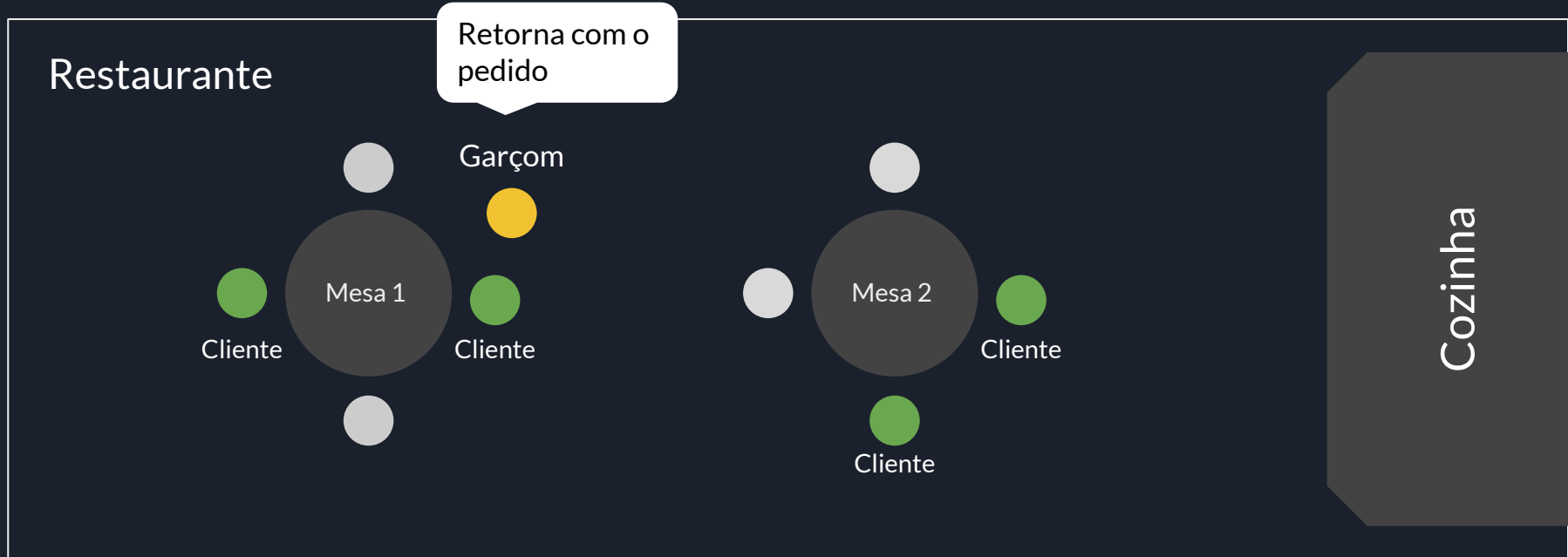
Entendendo a arquitetura single-thread não bloqueante



Entendendo a arquitetura single-thread não bloqueante

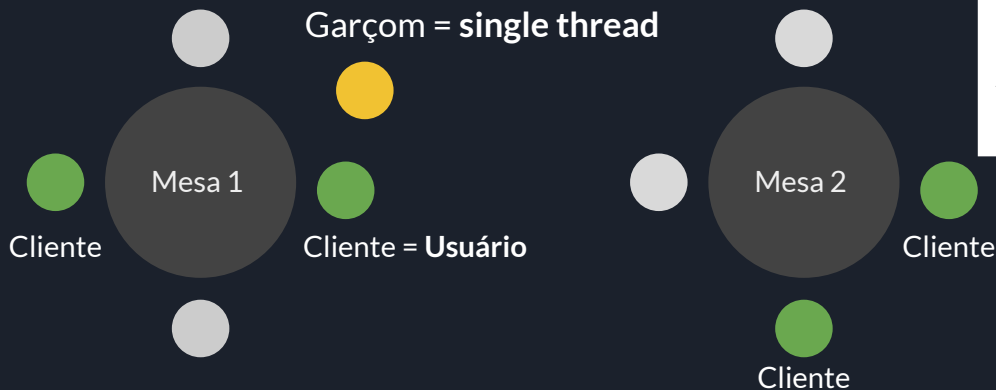


Entendendo a arquitetura single-thread não bloqueante



Entendendo a arquitetura single-thread não bloqueante

Restaurante = **Aplicação Web**



Processos de regras de negócio. Ex. Acesso ao banco de dados

Cozinha



```
1 console.log('Node.js');  
2 setTimeout(function callback() { console.log('é') }, 0);  
3 console.log('sensacional!');
```



Node.js
sensacional!
é



NodeJS

Não é a solução para todos os problemas



NodeJS

Ideal para aplicações com um grande volume de i/o, tais como chats, streaming, servidores web e comunicação de rede



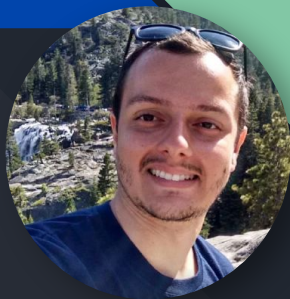
NodeJS

Não é recomendado por aplicações que utilizam alto uso do CPU, como por exemplo, manipulação de vídeos e imagens



<> Bora para cima!!!! </>

#techacademy #mecomprometo



Rafael Ribeiro
Head de Engenharia

Introdução ao NodeJS

Módulo X