

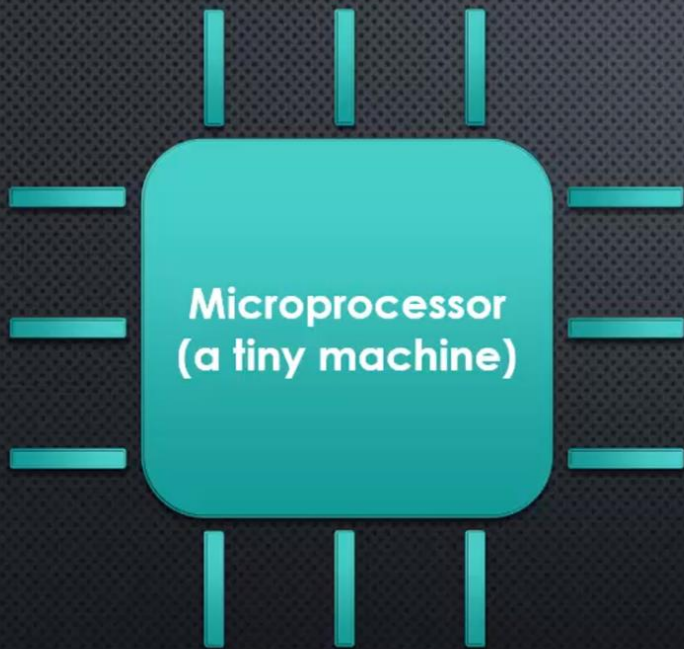
Tecnologias e Programação de Sistemas de Informação

Introdução

Desenvolvimento Web - Back-End | David Jardim

Cofinanciado por:





Instructions

IA-32

x86-64

ARM

MIPS

000018A45438100	0	55	push rbp
0000018A45438101	1	4889e5	REX.W movq rbp, rsp
0000018A45438104	4	56	push rsi
0000018A45438105	5	57	push rdi
0000018A45438106	6	41ff75a8	push [r13-0x58]
0000018A4543810A	10	56	push rsi
0000018A4543810B	11	49baf9552c7e8f010000	REX.W movq r10,
0000018F7E2C55F9	;;	object: 0000018F7E2C55F9	
0000018A45438115	21	4152	push r10
0000018A45438117	23	6a00	push 0x0
0000018A45438119	25	b803000000	movl rax, 00000000000000

Level of Abstraction

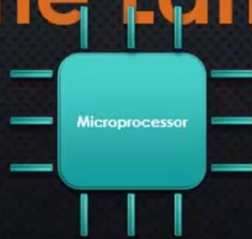


JavaScript

C/C++

Assembly Language

Machine Language





“Node.js® is a JavaScript runtime built on [Chrome's V8 JavaScript engine](#). Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js' package ecosystem, [npm](#), is the largest ecosystem of open source libraries in the world.”



Qual a linguagem de programação na qual o Node foi desenvolvido? E porquê?



“V8 is Google’s open source high-performance JavaScript engine, **written in C++**. It is used in Google Chrome, the open source browser from Google, and in Node.js, among others. It implements **ECMAScript** as specified in ECMA-262, and runs on Windows 7 or later, macOS 10.12+, and Linux systems that use IA-32, ARM, or MIPS processors. V8 can run standalone, or can be embedded into any C++ application.”

Javascript Code

My C++ Program

V8 (C++)

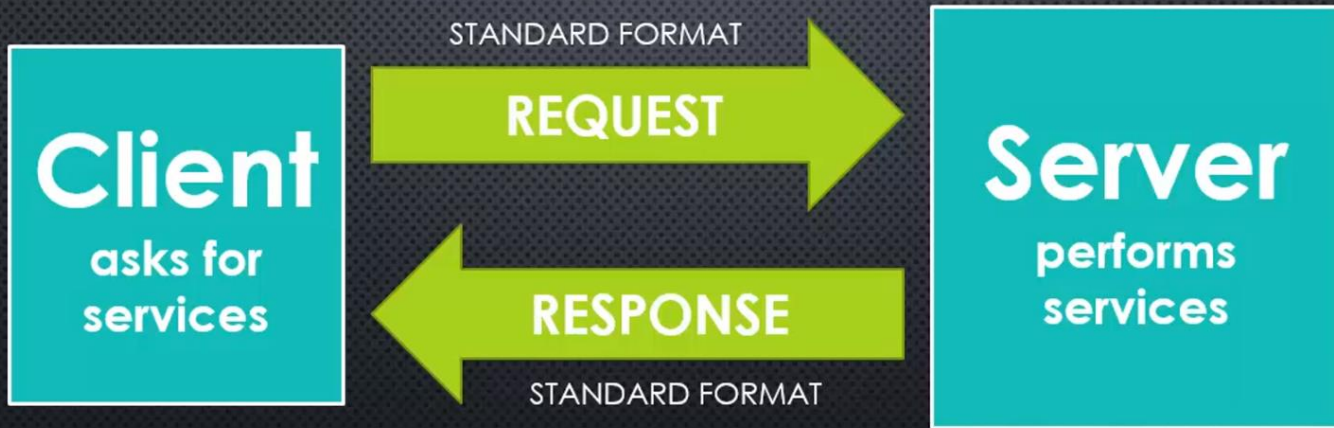
Machine Code

FRONT-END





BACK-END



Client-Server Model of Computing





O JS base não suporta as seguintes operações, o Node.js veio facilitar as seguintes funcionalidades:

- Organizar o código de forma modular/pacotes
- Melhores formas de lidar com ficheiros
- Melhores formas de lidar com bases de dados
- A capacidade de comunicar na Internet
- A capacidade de aceitar pedidos e enviar respostas

