VENTURUS 4TECH


It's rewind time

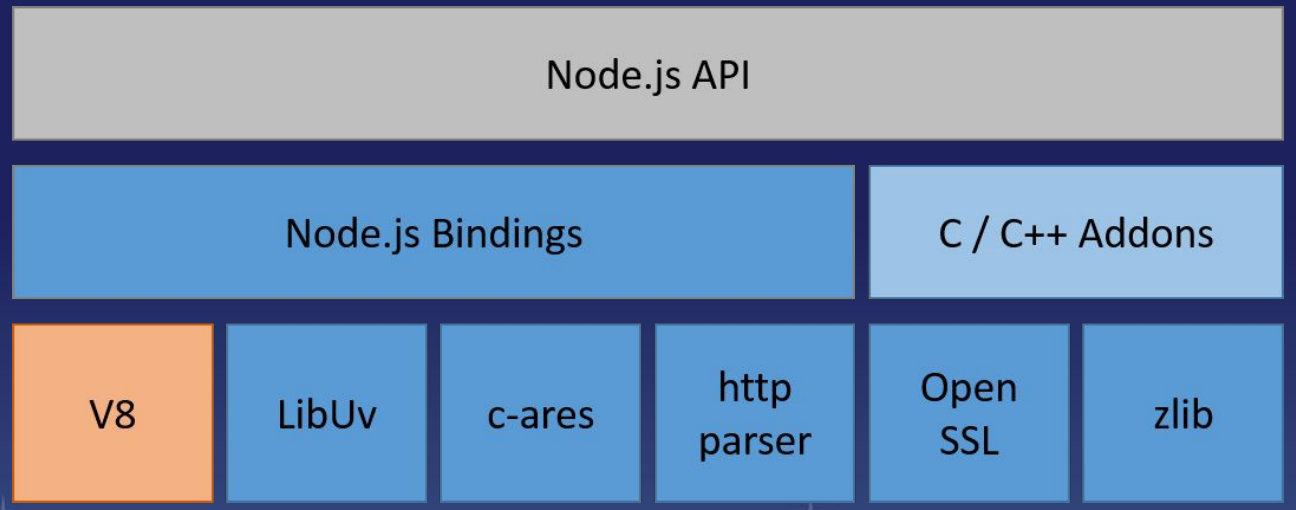
VENTURUS 4TECH

O que é Node.js?

- ➔ Plataforma para criação e execução de servidores para aplicações web
- ➔ Escrito em C++ (performance)
- ➔ Open-source (código aberto)
- ➔ Cross-platform (Linux, Mac, Windows)

Saiba mais: *Site oficial do Node.js*

➔ Por baixo dos panos o Node.js usa o interpretador de javascript **V8** criado pelo **Google** para o **Chrome**, além, claro, de outras funcionalidades:



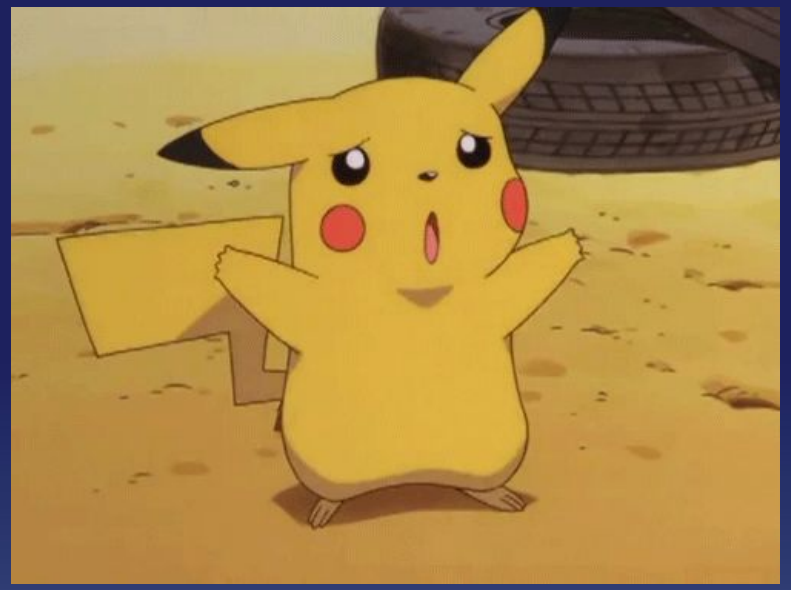
O que não é?

- Uma linguagem de programação
- Um framework
- Multi-thread
  - **Simplifica** muito a construção de uma aplicação.
  - Imperceptível na maioria dos casos por não ser **bloqueante**.

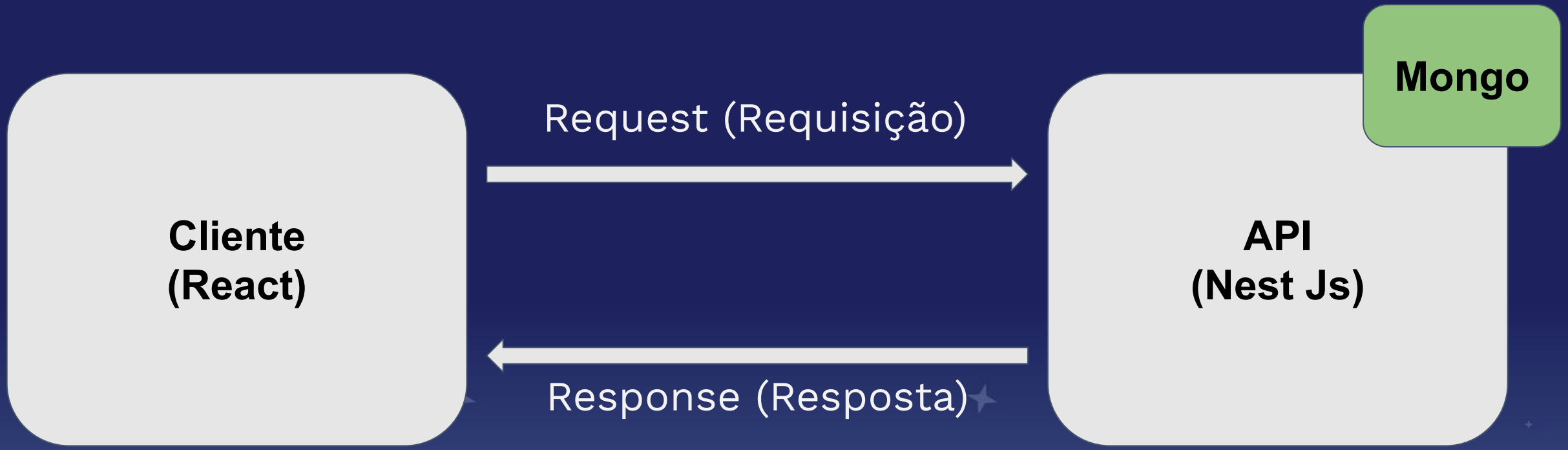
Como nossa aplicação vai funcionar?

Pede / Requisita / Ordena

Responde



REST API Design



Requisições HTTP(s)

Requisição

```
GET /HTTP/1.1
Host: www.google.com.br
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:29.0) Gecko/20100101 Firefox/29.0
Accept: text/html,application/xhtml+xml,application/xml
Accept-Language: pt-BR,pt,en-US,en
Accept-Encoding: gzip, deflate
Connection: keep-alive
```

Resposta

```
HTTP/1.1 200 OK
Date: Tue, 17 Jun 2014 01:20:13 GMT
Server: gws
Location: https://www.google.com.br/
Last-Modified: Tue, 17 Jun 2014 01:20:13 GMT
Content-Encoding: gzip
Content-Length: 234
Connection: closeContent-Type: text/html


<html>todo o html da página</html> *
```

Verbos:

GET, POST, PUT, DELETE

Saiba mais: *HTTP Status Codes*

VENTURUS 4TECH

Verbos HTTP(s)

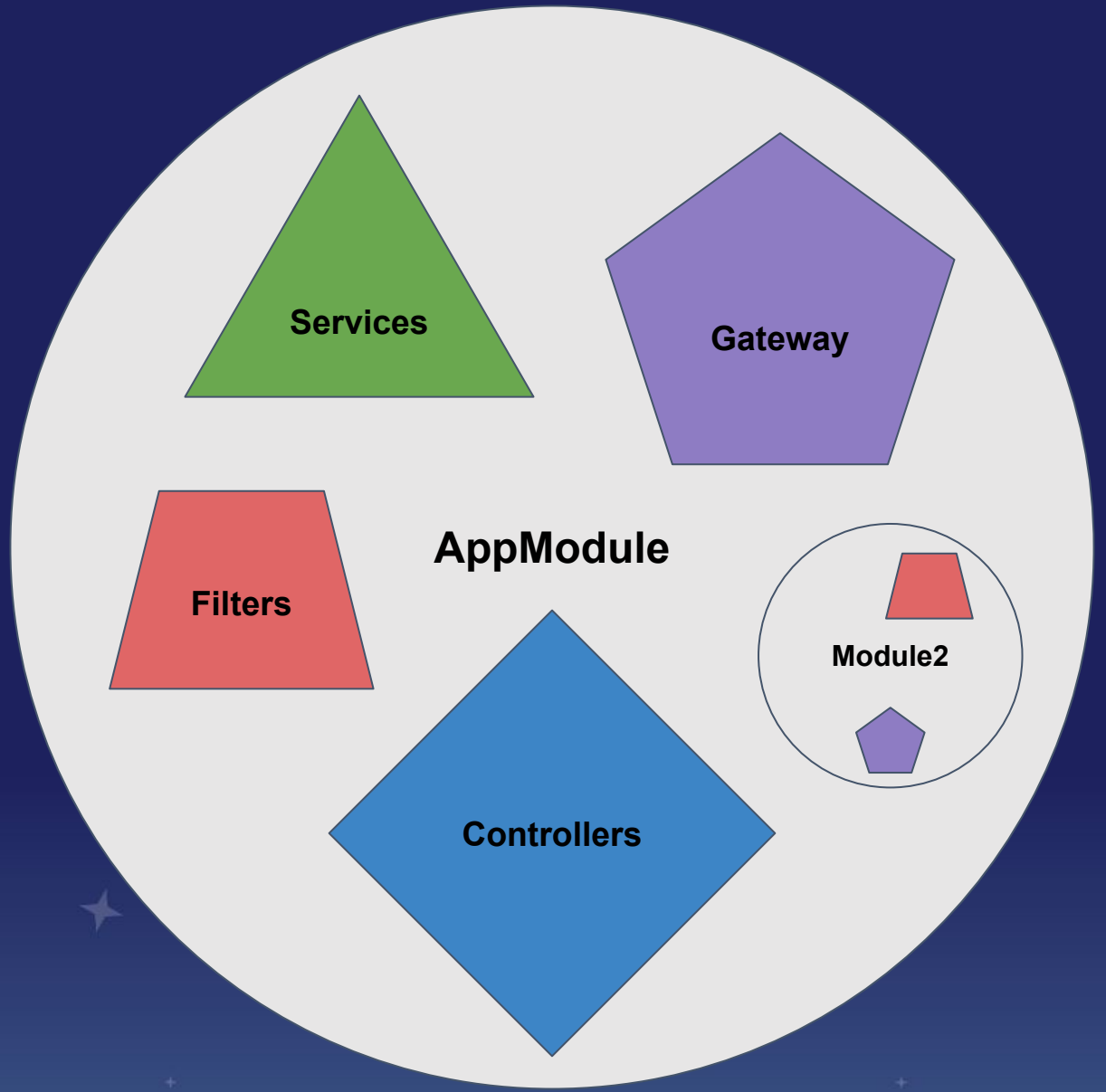
- GET
- HEAD
- POST
- PUT
- DELETE
- CONNECT
- OPTIONS
- TRACE
- PATCH

VENTURUS 4TECH

O que é o Nest.js?

- Framework para a construção eficiente e escalável de aplicações Node.Js no *server side.*
- Por baixo dos panos, faz uso de Frameworks Http robustos como *Express.Js* (por *Default*, mas também pode ser utilizado o *Fastify*).

Saiba mais: *Site oficial do Nest.Js*



Alguma dúvida?

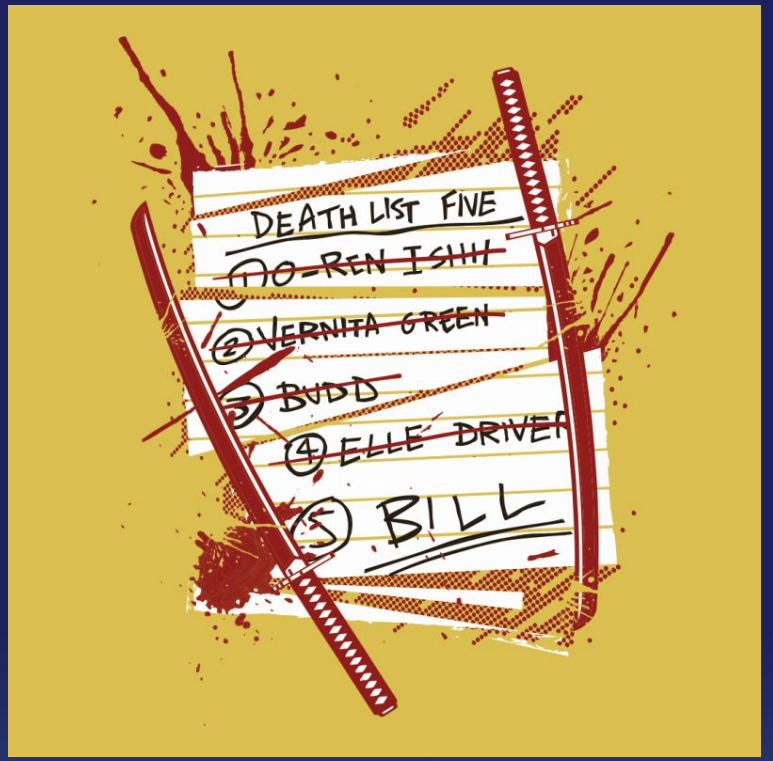
Mãos na massa

Criação do Projeto

nest new ‘nome do projeto’

Estruturação

- Faremos uso principalmente da arquitetura *Multi Layered* e alguns outros *Project Patterns*.
- Simplifica muito como fazemos a gestão de cada *Layer*.
- Se a aplicação exigir tráfego de rede mais rápido, maior confiabilidade e mais performance, então a arquitetura é uma boa escolha.



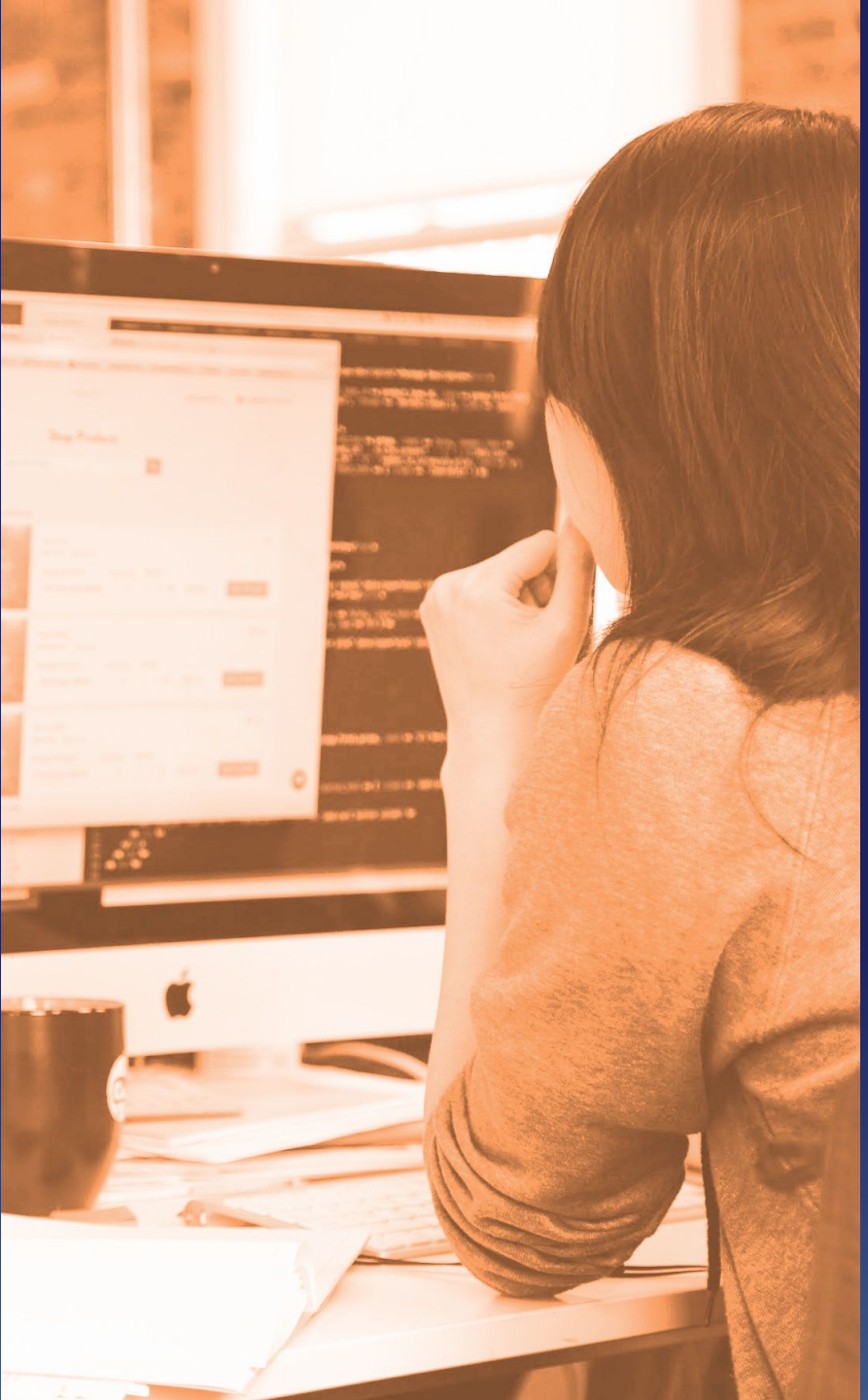
Saiba mais em: [Multi Layered Architecture]

Desafio

1 - Criar Endpoints para atualizar e remover um usuário utilizando os verbos HTTP ‘Put’ e ‘Delete’.

2- Criar um Endpoint para adicionar uma lista de usuários de uma vez só. Ou seja, o ‘Postman’ irá enviar uma lista de novos usuários, e todos devem ser cadastrados.

VENTURUS 4TECH



+Links e referências

- Se tiver dúvidas em Node ou quer estudar um pouco além:
  - What exactly is Node.js?
  - API/REST/RESTful
  - Nest.js Documentation

Valeu galera!
Até amanhã!

VENTURUS 4TECH

Missão #DIA3 concluída!