

# Como o *Open-Source* me torna um desenvolvedor melhor

## Estudos de caso sobre contribuições *open-source*

Paulo Valente

Tech Leader Backend @ Stone Pagamentos

ElugCE - Abril/2021

# Introdução

## Estudos de Caso

- › Tesla
  - › <https://github.com/teamon/tesla>
- › ElixirLS
  - › <https://github.com/elixir-lsp/elixir-ls>
- › Nx - Numerical Elixir
  - › <https://github.com/elixir-nx/nx>

# Estudo de Caso - Tesla

## Melhorias advindas da necessidade

### Retry Middleware

- › Não entendi o uso de um parâmetro
- › Estudei o código-fonte para entendê-lo
- › Descobri um pequeno bug na estratégia de tentativas

### Aprendizados

- › Fix foi bem curto, mas carregava muita informação.
- › Precisei prestar atenção para colocar comentários relevantes no código.

# Estudo de Caso - Tesla

## Melhorias advindas da necessidade

### KeepRequest Middleware

- › Precisei utilizar uma combinação de middlewares que não interagiu bem
- › Propus um fix em um dos middlewares
- › Review fez com que eu desfizesse as mudanças e atacasse por outro caminho

### Aprendizados

- › Nem todo review negativo significa que voltamos à estaca zero
- › Iteração final criou sinergia entre os 3 middlewares envolvidos
- › Precisei documentar bem - nem tudo é claro para quem não escreveu o código 4/10

# Estudo de Caso - ElixirLS

## Comando nativo do LS para manipular pipes

Várias iterações:

1. TypeScript em uma extensão pessoal do VSCode
2. Mix Escripts e ElixirSense
  - › Precisei tirar dúvidas com um dos maintainers do ElixirLS, Jason Axelson
  - › Sugeriu incorporarmos ao Language Server
3. Incorporada e em fase de testes!
  - › Vai estar disponível para todos os editores de texto que usam o ElixirLS

# Estudo de Caso - ElixirLS

## Comando nativo do LS para manipular pipes

### Aprendizados

- › Ajudou a praticar a habilidade de mexer em bases de código desconhecidas
- › Apreendi vários detalhes de como o Language Server Protocol funciona
- › Precisei lidar com casos de borda que só foram pegos em review

# Estudo de Caso - Nx

- › Descobri o projeto com a primeira divulgação do José Valim
- › Primeiras issues: funções de álgebra linear
  - › Relembrar AlgLin foi o primeiro desafio
- › O caso da SVD
  - › Livros didáticos não costumam ensinar algoritmos para calcular
  - › Precisei estudar artigos e código fonte do TensorFlow para finalizar
- › Argsort: primeira funcionalidade implementada no EXLA
  - › Dificuldades para coordenar as diferentes partes do sistema (incluindo NIFs em C++!)

# Estudo de Caso - Nx

## Aprendizados

- › Relembrar conceitos matemáticos para contribuir
  - › Certas bibliotecas tem seu domínio específico
- › Base de código já surgiu bem extensa, mas a documentação ajudou a me situar
- › Praticamente tudo é testado com doctests
- › Parte do desafio é pensar sobre a usabilidade e manutenibilidade da API



# Conclusão

- › Open-source pode ser a porta de entrada para muitas oportunidades
- › É uma forma de contribuir com a sociedade
- › Aprendizados não apenas em software
  - › Ajuda a praticar habilidades sociais também

**Obrigado!**

**Perguntas?**