

Ana Teixeira

- Bacharela em Ciência da Computação - PUC Minas
- Especialista em Engenharia de Software - PUC Minas
- Especialização em Linguagem e Tecnologia - UFMG (em andamento)

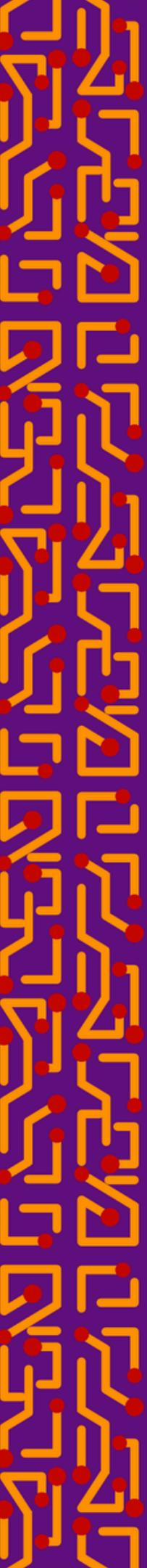
Trabalhei com empresas diversas como Raízen, CI&T, Unisys, Faculdade Descomplica, Gama Academy, Digital House e algumas outras.

Sou mineira, nascida em Belo Horizonte, tenho 31 anos e minha experiência é em tecnologia: curiosa e entusiasta desde sempre.

Vamos nos conectar?

acrispteixeira@gmail.com | anateixeira.tech





Java com Spring Boot: Criando Aplicações Robustas com Abordagem Prática

Ana Teixeira



Agenda

1. Introdução: Java e Spring
 2. Arquitetura e Principais Recursos
 3. Projeto na prática
 4. Dicas e Tendências do Mercado
 5. Próximos passos
- 



1. Introdução: Java e Spring

- O que é esse tal do **Java**?
- E esse tal do **Spring**?





Por que usar Spring?

- Reduz a complexidade do setup
- Configuração baseada em convenção
- Ampla documentação e suporte da comunidade.

Detalhes técnicos específicos:

- Java Versão 17
- Spring 3.4.3
- Windows 11 Home



2. Arquitetura e Principais Recursos

- Arquitetura: convenção sobre configuração via módulos
- Recursos para reduzir dependências manuais
 - Spring Boot Starter
 - Spring Boot AutoConfiguration
 - Spring Boot Actuator

Exemplo de estrutura de projeto



```
meu-projeto-springboot
├── src
│   ├── main
│   │   ├── java/com/exemplo
│   │   │   └── MeuProjetoApplication.java # Classe principal (@SpringBootApplication)
│   │   ├── controller
│   │   │   └── MeuController.java # Controladores REST ou serviços
│   │   ├── service
│   │   │   └── MeuService.java # Regras de negócio
│   │   ├── repository
│   │   │   └── MeuRepository.java # Interface para acesso a dados
│   │   ├── model
│   │   │   └── MeuModelo.java # Representação de entidades
│   │   └── resources
│   │       ├── application.properties # Configurações do projeto
│   │       ├── static/ # Arquivos estáticos (se aplicável)
│   │       └── templates/ # Templates para renderização (se aplicável)
│   └── test
│       └── java/com/exemplo
│           └── MeuProjetoApplicationTests.java # Testes unitários
└── pom.xml # Arquivo de dependências (Maven)
```

MeuProjetoApplication.java # Classe principal (@SpringBootApplication)

MeuController.java # Controladores REST ou serviços

MeuService.java # Regras de negócio

MeuRepository.java # Interface para acesso a dados

MeuModelo.java # Representação de entidades

application.properties # Configurações do projeto

static/ # Arquivos estáticos (se aplicável)

templates/ # Templates para renderização (se aplicável)

MeuProjetoApplicationTests.java # Testes unitários

pom.xml # Arquivo de dependências (Maven)

README.md # Documentação do projeto



3. Projeto na Prática

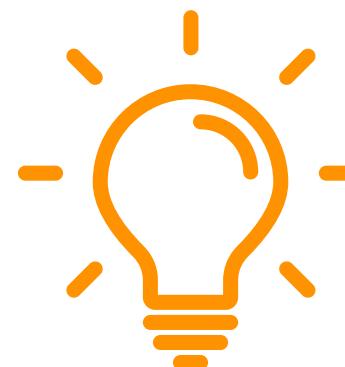
1. Configurar o ambiente
2. Criar novo projeto Spring Boot
 - Se possível, colocando as dependências
3. Estruturar o projeto
4. Configurar o projeto (pom.xml)
5. Criar classe principal

Detalhes técnicos específicos:

- Projeto completo no link do Github
- Uso do Spring Batch
- Sistema de Processamento de Arquivos

4. Dicas e Tendências do Mercado

- Desenvolvimento de microserviços e sistemas distribuídos
- Aplicações de processamento de dados, principalmente movidos a eventos
- Integração com sistemas legados



4. Dicas e Tendências do Mercado

- Integração com containerização
- Observabilidade e monitoramento
- Uso de arquiteturas Serverless e Edge Computing com Java





5. Próximos Passos

- Revisar conceitos
- Estudar projetos
- Praticar com o código
- Conectar com a comunidade Java



Obrigada!

Me encontre em

acrispteixeira@gmail.com | anateixeira.tech

