

# Estagnado na carreira?

O que você precisa saber para se tornar um  
Desenvolvedor Sênior ou um Arquiteto de Software?



Eduardo Pires

Microsoft Regional Director – MVP

**desenvolvedor.io**



Canal .NET

O que veremos?



- Opções de carreira no desenvolvimento
- Desenvolvimento Web
- Níveis de conhecimento
- Erros comuns
- Caminho de conhecimento
- Como se tornar um arquiteto
- Dicas de leitura
- Vouchers \o/



Quais são as opções  
na carreira de  
desenvolvedor de  
software?

# Principais atuações no desenvolvimento de software



Desenvolvedor  
Desktop

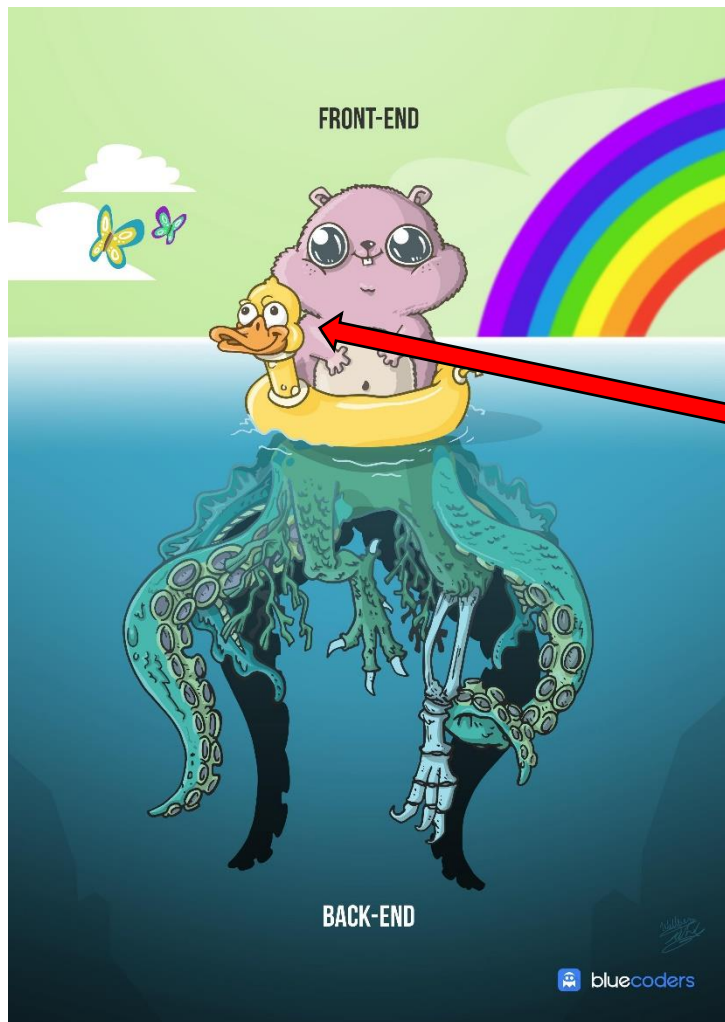


Desenvolvedor  
Web



Desenvolvedor  
Mobile

# Desenvolvimento Web



FullStack Developer



# Níveis de conhecimento no desenvolvimento de software



Iniciante  
Estudante  
Estagiário



Júnior



Pleno



Sênior



Especialista ou  
Arquiteto

# Erros comuns



# Iniciante Estudante Estagiário



- Está dando os primeiros passos no desenvolvimento.
- Pode ter algum conhecimento ou não saber absolutamente nada.
- Não entrega projetos.
- Pode ajudar em tarefas simples.
- Não pode ser deixado sozinho sem acompanhamento técnico.
- Deve investir todo seu tempo estudando



## Júnior



- Possui conhecimentos fundamentais sobre desenvolvimento.
- Geralmente escreve código baseado em tutoriais, códigos e projetos prontos.
- Possui muitas dúvidas.
- Consegue entregar projetos básicos / medianos.
- Comete erros clássicos, o código ser melhor.
- Dificuldade de entender um problema e propor uma solução.
- Deve ter contato/acompanhamento de pessoas mais avançadas tecnicamente.

### Desejável:

- Conhecer estrutura de dados e algoritmos
- Conhecer o básico sobre testes

### Pessoal

- Responsabilidade com as entregas
- Foco em entender o negócio

# Learning Path

Iniciante  
Estudante  
Estagiário



Júnior



Lógica de  
Programação

Noções em  
alguma linguagem  
e tecnologia

Estrutura  
de dados

Algoritmos

Banco de  
dados

Tecnologia de  
desenvolvimento

# Pleno



- Entende o fundamento (como as coisas funcionam por baixo dos panos).
- Sabe escrever código com boas práticas e padrões de design.
- Consegue entregar projetos do início ao fim.
- Consegue entender os requisitos do cliente, negociar e propor soluções com base em seu conhecimento.
- Capacidade de analisar, entender e resolver um problema técnico.
- Sabe testar, entrega código com diferentes tipos de testes (unidade/integração/automação).
- Tem certa autonomia na gestão do tempo e tarefas.
- Importante receber mentoria de um sênior.

## Desejável:

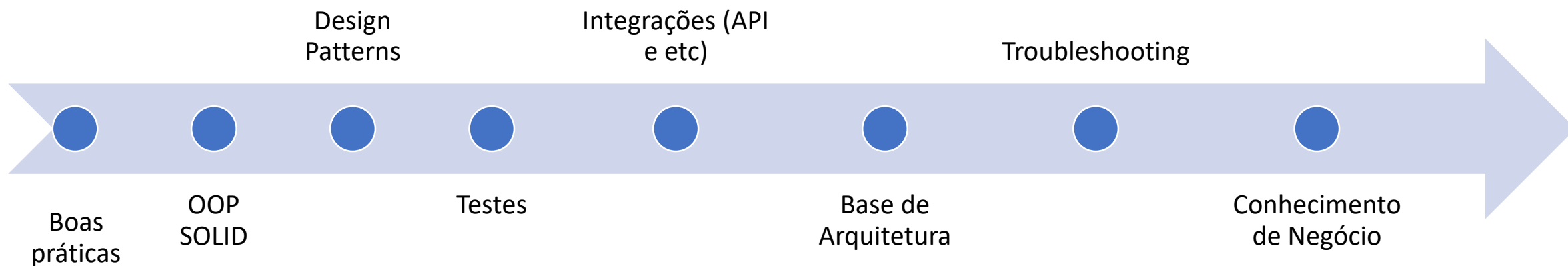
- Conhecimentos de arquitetura
- Conhecimentos sobre processos (DevOps, CI/CD, gestão ágil).

## Pessoal

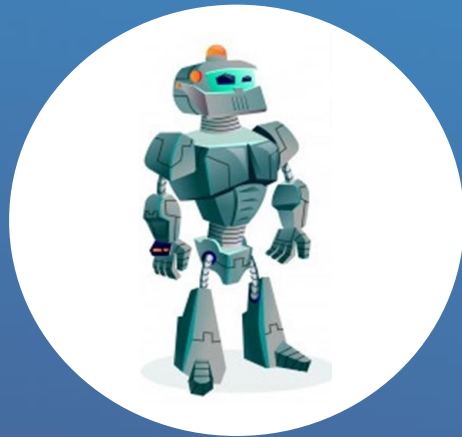
- Conhecimento de negócio
- Autonomia / Responsabilidade
- Capacidade de repassar conhecimento

# Learning Path

Pleno



# Sênior



- Domina por completo as tecnologias em sua área de atuação.
- Escreve código de qualidade e busca constante aperfeiçoamento.
- É capaz de entregar projetos complexos tanto tecnicamente quanto em questão de negócios.
- Conhece o negócio e sabe lidar com as demandas do cliente.
- Possui alta capacidade em análise e resolução de problemas utilizando conhecimento e ferramentas adequadas.
- Domina completamente as técnicas de testes e qualidade.
- Possui autonomia, consegue tocar projetos, negociar prazos, cobrar entregas.
- Conhece o processo de entrega de software por completo.

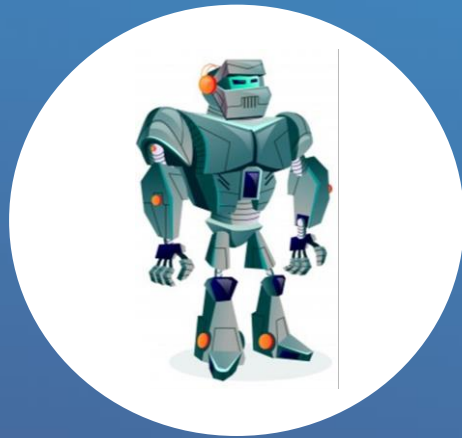
## Desejável:

- Conhecimentos de arquitetura
- Conhecimentos de SRE, DevOps
- Conhecimento em outras linguagens (ex: linguagem funcional)

## Pessoal

- Liderança (liderar pelo exemplo)
- Capacidade de repassar conhecimento
- Maturidade para lidar com conflitos (pessoas/times)

# Especialista / Arquiteto



- Possui todos os conhecimentos de um sênior, porém muito mais experiência profissional.
- Deve possuir uma série de hard e soft skills.
- Possui sólidos conhecimentos de arquitetura (padrões, estilos, etc).
- É responsável pelo planejamento, execução e qualidade dos projetos (assim como um dos responsáveis pelo fracasso dele).
- Autonomia plena, é um profissional auto-gerenciável, muitas vezes faz “par” com posições gerenciais (ou assume parte delas).

## Desejável:

- Conhecimentos de SRE, DevOps
- Conhecimento em outras linguagens (ex: linguagem funcional)

## Pessoal

- Liderança (liderar pelo exemplo)
- Capacidade de repassar conhecimento
- Maturidade para lidar com conflitos (pessoas/times)

# Erros comuns

```
ader+=(self[indexNum].getRow(0,hide)+) upper+=(self[indexNum].ge
image.open(image_file) /> width, height = image.size /> root.resiz
width=False, height=False) image_file = io.BytesIO(image) photoimage = Capa
n:10px n:10px
olid;} olid;}
ame"/> ame"/>
r-</st r-</st
1>0</h1> #font = ImageFont.truetype("3270.otf",size:18) # width_t
ext= font.getsize(text)# image_cover((text_x, text_y), text,fill=
hite" <h1>melhor</h1> font=font) width_text, height_text = setCo
ext((text_x, text_y), text, fill="white" #C2D3DC) photoimage=Capa
,image = photoimage<h1>investimento</h1> .place(x=0,y=0) entry_p
7/>
ge)
ze)
2/> Entry(root, background=#292A86).<h1>do</h1> place(x=text_x, y
xt + entry_pady)+ toplevel = Toplevel(root) + toplevel.resizable
h=False, height=False) image_file = io.BytesIO(image) photoimage2 = Imag
_IMAGE_BACKGROUND)) image = Image.open(image_file) /> photoimage2 = Imag
Image(image)/canvas =Canvas(toplevel,width=width, height=height) canvas.c
nvas.create_image((0,0), image=photoimage2, anchor="nw") canvas.c
ut> ELES TÊM 20 E POUCOS ANOS, ESCOLHEM ONDE VÃO TRABALHAR E COM
pe= QUAL SALÁRIO. PARA CONTRATÁ-LOS, EMPRESAS CHEGAM A COMPRAR
b"> STARTUPS INTEIRAS. ESCOLAS DE FORMAÇÃO NASCEM BRASIL AFORA
d;} JÁ COM FILA DE ESPERA. BEM-VINDO AO FANTÁSTICO MUNDO DOS
le> DESENVOLVEDORES DE SOFTWARE
ef> compile_MainMenu_Elements(self):<def>getBlankSpace(word,total)
wer+=(self[indexNum].getRow(2,hide)+) footer+=(self[indexNum].ge
sys_title(image) text_x = 55 + text_x = 45 text = "Exame capa">in
```

# EXAME

## A CORRIDA PELOS DEVS

ELES TÊM 20 E POUCOS ANOS, ESCOLHEM ONDE VÃO TRABALHAR E COM

QUAL SALÁRIO. PARA CONTRATÁ-LOS, EMPRESAS CHEGAM A COMPRAR

STARTUPS INTEIRAS. ESCOLAS DE FORMAÇÃO NASCEM BRASIL AFORA

JÁ COM FILA DE ESPERA. BEM-VINDO AO FANTÁSTICO MUNDO DOS

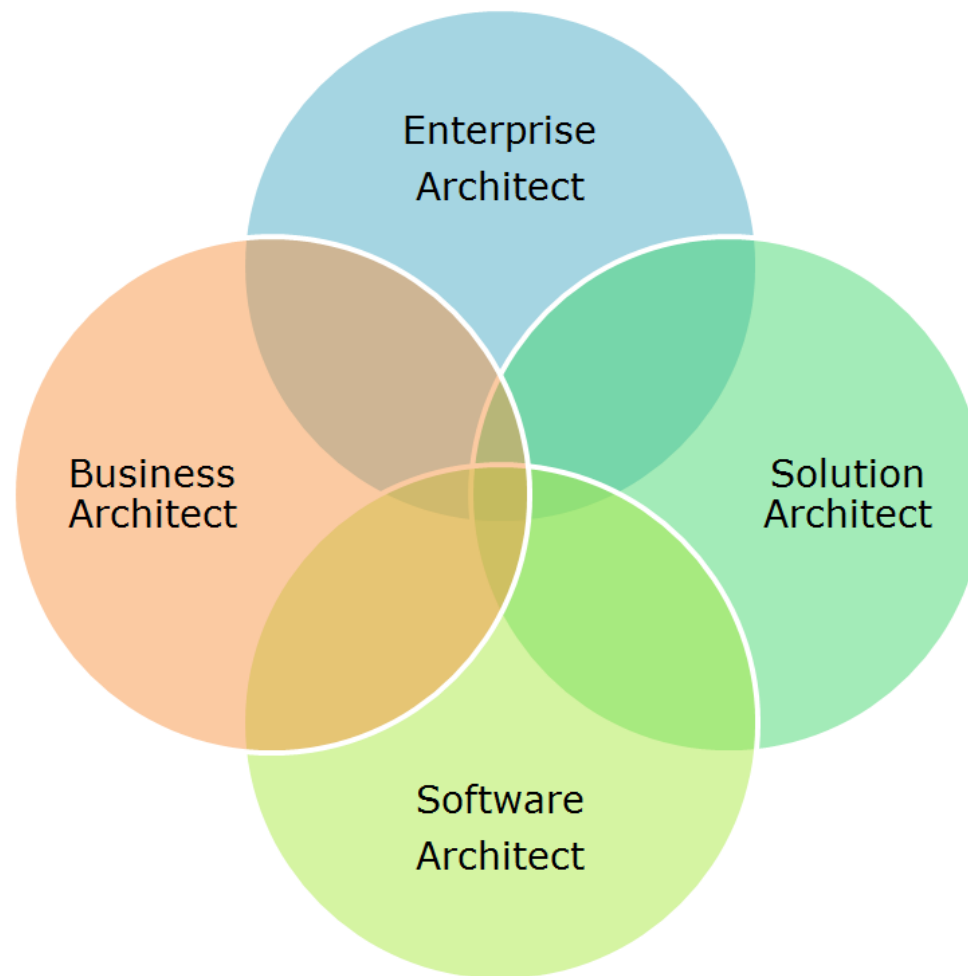
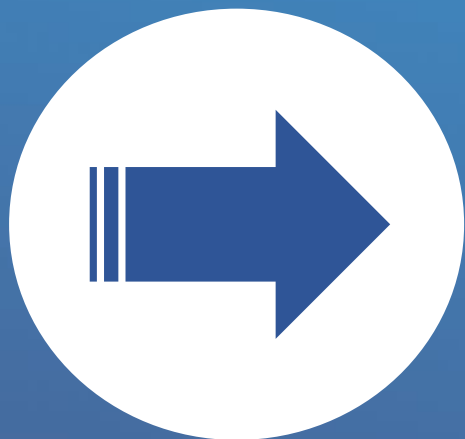
DESENVOLVEDORES DE SOFTWARE



Como faço para me  
tornar um  
arquiteto de software?



# Perfis de Arquiteto



# Responsabilidades



- ✓ O objetivo principal de um arquiteto de software é projetar uma solução compatível com os requisitos atuais da corporação, que tenha flexibilidade suficiente para comportar mudanças futuras e/ou novos requisitos resultantes de sua evolução ao longo do tempo.
- ✓ Ser responsável pela qualidade técnica do entregável, atuar no apoio técnico de toda equipe e ajudar nas decisões necessárias.
- ✓ Promover a comunicação do time, envolvendo e delegando as decisões de arquitetura com toda equipe.

# Requisitos Técnicos “Hard Skills”



- ✓ O principal requisito técnico de um arquiteto de software é o profundo conhecimento em programação e componentização. Conhecimentos adquiridos através de experiências sólidas em projetos.
- ✓ Dominar amplamente alguma plataforma de desenvolvimento, frameworks, ferramentas e modelagem técnica.
- ✓ Conhecer abordagens de arquitetura e padrões para soluções de problemas.
- ✓ Conhecer técnicas de design de código e qualidade para permitir uma fácil manutenção e testabilidade.

## Requisitos Pessoais “Soft Skills”



- ✓ Liderança
- ✓ Comunicação
- ✓ Proatividade
- ✓ Mindset empreendedor
- ✓ Humildade

# Requisitos Pessoais

## “Soft Skills”



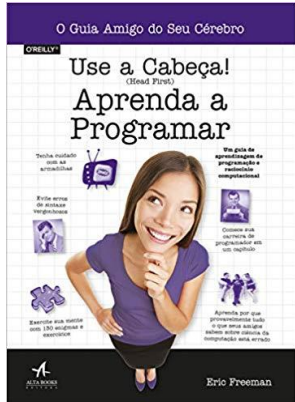
- ✓ Tomar boas decisões
- ✓ Estar aberto para críticas e sugestões
- ✓ Ser um profundo pesquisador
- ✓ Ser didático
- ✓ Saber delegar tarefas
- ✓ Saber questionar
- ✓ Saber conduzir discussões
- ✓ Saber conversar com diferentes níveis hierárquicos
- ✓ Estar sempre aberto às novas tecnologias
- ✓ Estar sempre atualizado com o mercado
- ✓ Ter maturidade e equilíbrio emocional
- ✓ Ter pensamento estratégico
- ✓ Reconhecer e assumir os próprios erros e de sua equipe

# Mitos sobre o Arquiteto de Software

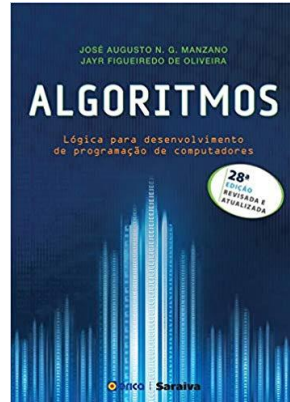


- ✓ O arquiteto não programa, apenas projeta, documenta e lidera.
- ✓ Um desenvolvedor sênior pode assumir o papel de arquiteto.
- ✓ O arquiteto é um desenvolvedor com muito tempo de experiência.
- ✓ Para ser arquiteto basta estudar técnicas de arquitetura de software.
- ✓ O arquiteto trabalha em ambiente isolado ou diferenciado da equipe.
- ✓ O arquiteto é a autoridade técnica e não deve ser questionado.
- ✓ O arquiteto é um gerente técnico.
- ✓ O arquiteto é um profissional com graduação em arquitetura de software.
- ✓ O arquiteto é o profissional que está na liderança de um projeto

# Leituras Recomendadas (iniciantes)



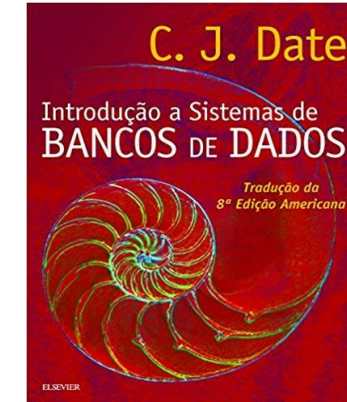
Use a Cabeça!  
Aprenda a  
Programar  
Eric Freeman



Algoritmos  
Jayr Figueiredo  
José Manzano



Lógica de  
Programação  
Andre Forbellone



Introdução a sistemas  
de bancos de dados  
C. J. Date

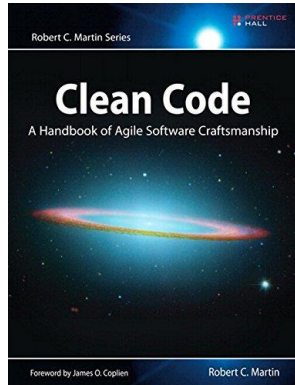
**Documentação em forma de tutorial para aprender C#**

<https://docs.microsoft.com/pt-br/dotnet/csharp/tutorials/>

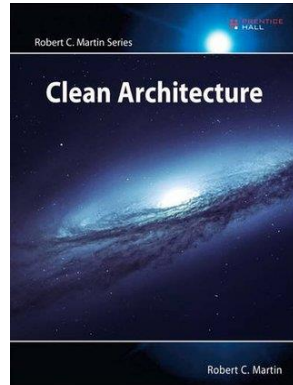
**Vídeo aulas sobre os primeiros passos em C#**

<https://dotnet.microsoft.com/learn/csharp>

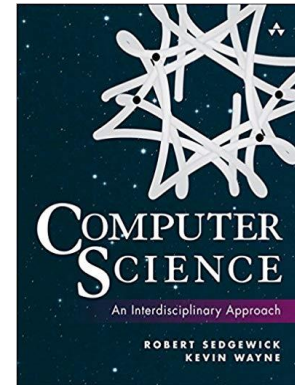
# Leituras Recomendadas



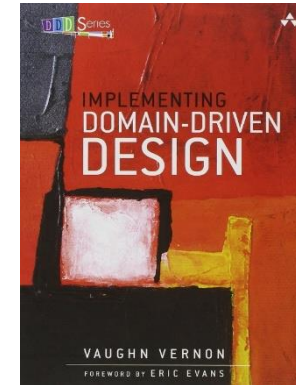
Clean Code  
Robert C. Martin



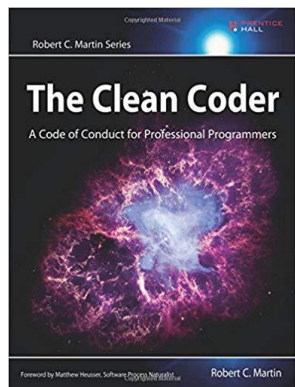
Clean Architecture  
Robert C. Martin



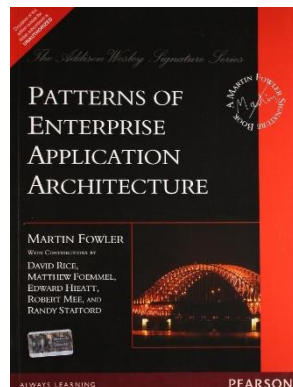
Computer Science  
Robert Sedgewick  
Kevin Wayne



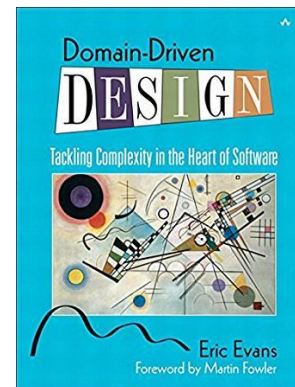
Implementing Domain  
Driven Design  
Vaughn Vernon



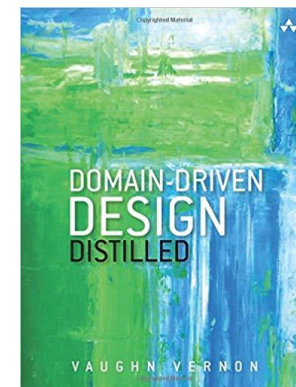
Clean Coder  
Robert C. Martin



Patterns of Enterprise  
Application  
Architecture  
Martin Fowler



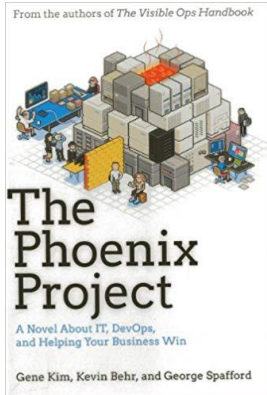
Domain-Driven  
Design  
Eric Evans



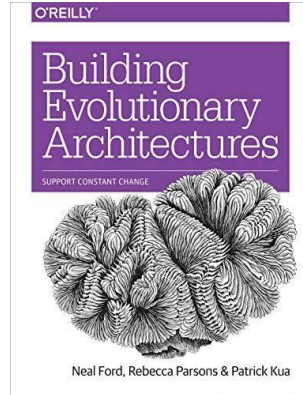
Domain-Driven Design  
Distilled  
Vaughn Vernon



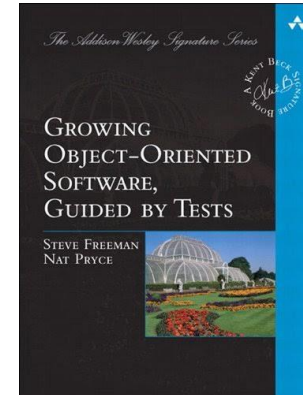
# Leituras Recomendadas



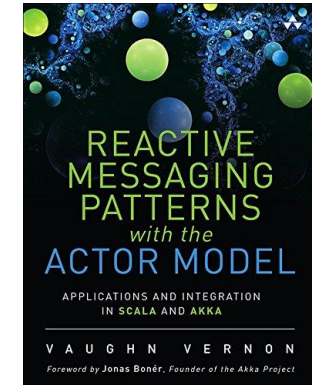
The Phoenix Project  
Gene Kim  
Kevin Behr



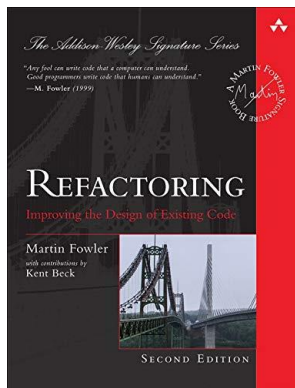
Building Evolutionary Architectures  
Neal Ford  
Rebecca Parsons



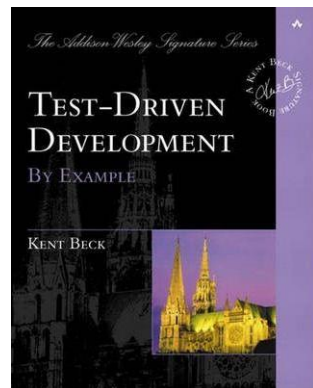
Growing Object-Oriented Software, Guided By Tests  
Steve Freeman  
Nat Pryce



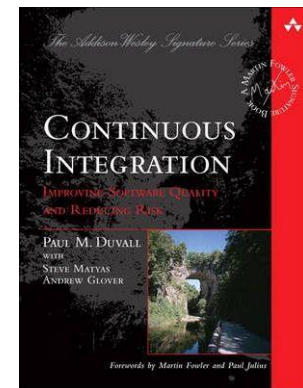
Reactive Messaging Patterns with the Actor Model  
Vaughn Vernon



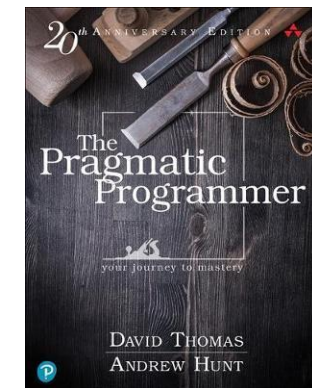
Refactoring  
Martin Fowler



Test Driven Development : By Example  
Kent Beck



Continuous Integration  
Paul M. Duvall



The Pragmatic Programmer  
David Thomas  
Andrew Hunt

# Cursos online de programação

Está na hora de decolar na sua carreira



10-OFF-CANAL-DOTNET