



Bootcamp: Arquiteto(a) de Software

Plano de Ensino

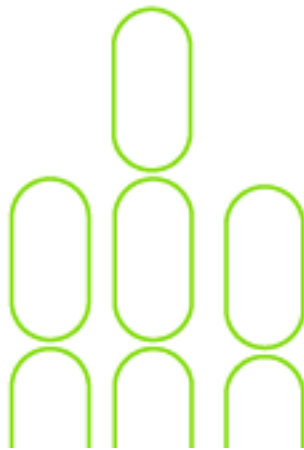
Módulo 3: Design Patterns, Estilos e Padrões Arquiteturais

Descrição do Módulo

Este módulo tem o objetivo de apresentar os principais princípios e boas práticas de desenvolvimento e arquitetura de software, de forma a explorar suas respectivas aplicabilidades e, dessa forma, contribuir para o enriquecimento do emprego das práticas de arquitetura de software dos alunos no dia a dia.

Objetivos de Ensino

Espera-se que o aluno consiga, ao final deste módulo:

1. Entender os principais padrões de desenho;
 2. Entender os principais estilos e padrões arquiteturais;
 3. Identificar e aplicar os diferentes padrões em um produto de software;
 4. Entender os principais padrões de integração de sistemas;
 5. Identificar e aplicar os diferentes padrões de integração de sistemas.
- 



Estrutura Analítica do Curso

1ª Parte do Módulo

Capítulo 0 - Visão Geral do Módulo **Videoaulas gravadas**

- 0.1. Apresentação do Professor
- 0.2. Apresentação do Módulo


Capítulo 1 – Arquitetura de Software e Qualidade de Código **Videoaulas gravadas**

- 1.1. Introdução
- 1.2. *Code Smells*
 - 1.2.1. *Code Smells – Bloaters*
 - 1.2.2. *Code Smells – Couplers*
- 1.3. Análise Estática de Código

Capítulo 2 – Catálogos de Padrões

- 2.1. Introdução aos Catálogos de Padrões  **EBOOK**
- 2.3. *Patterns of Enterprise Application Architecture*  **EBOOK**
- 2.3. *Design Patterns – GoF*  **Material Complementar**
- 2.4. *Anti Patterns*  **Material Complementar**

Capítulo 3 – GoF – Padrões de Criação **Videoaulas gravadas**

- 3.1. *Singleton*
 - 3.2. *Builder*
 - 3.3. *Factory Method*
 - 3.4. *Abstract Factory*
- 

Capítulo 4 – GoF – Padrões Estruturais 🎬 **Videoaulas gravadas**

- 4.1. *Adapter*
- 4.2. *Facade*
- 4.3. *Flyweight*
- 4.4. *Composite*
- 4.5. *Decorator*

Capítulo 5 – GoF – Padrões Comportamentais 🎬 **Videoaulas gravadas**

- 5.1. *Command*
- 5.2. *Template Method*
- 5.3. *State*
- 5.4. *Strategy*

Capítulo 6 – Princípios SOLID 🎬 **Primeira Aula Interativa**

- 6.1. *Simple Responsibility Principle*
- 6.2. *Open Closed Principle*
- 6.3. *Liskov Substitution Principle*
- 6.4. *Interface Segregation Principle*
- 6.5. *Dependency Inversion Principle*

Atividade de Fixação

2ª Parte do Módulo

Capítulo 7 – Estilos Arquiteturais 📖 **Ebook**

- 7.1. Estilos Arquiteturais
- 7.2. Estilos Arquiteturais – *Structure*
- 7.3. Estilos Arquiteturais – *Message Styles*
- 7.4. Estilos Arquiteturais – *Adaptive Systems*
- 7.5. Estilos Arquiteturais – *Distributed Systems*
- 7.6. Estilos Arquiteturais – *Deployment*



Capítulo 8 – Padrões Arquiteturais 🎬 Videoaulas gravadas

- 8.1. Padrões Arquiteturais
- 8.2. *Model View Controller* (MVC)
- 8.3. *Model View Presenter* (MVP)
- 8.4. *Model View ViewModel* (MVVM)

Capítulo 9 – Arquiteturas de Camadas 🎬 Videoaulas gravadas

- 9.1. *Hexagonal Architecture*
- 9.2. *Onion Architecture*
- 9.3. *Clean Architecture*

Atividade Modular


Capítulo 10 – Padrões para Sistemas Distribuídos 🎬 Videoaulas gravadas

- 10.1. Padrões para Sistemas Distribuídos
- 10.2. *Enterprise Application Patterns – Point to Point Channel*
- 10.3. *Enterprise Application Patterns – Publish Subscribe Channel*
- 10.4. *Enterprise Application Patterns – Concurrency*

Capítulo 11 – Arquiteturas de Sistemas Distribuídos 🎬 Videoaulas gravadas

- 11.1. *Enterprise Application Integration* (EAI)
- 11.2. *Service Oriented Architecture* (SOA)
- 11.3. *Enterprise Service BUS* (ESB)

Capítulo 12 – Aplicação de Padrões Arquiteturais 🎬 Segunda aula interativa

- 12.1. Aplicação de Padrões Arquiteturais – MVC
 - 12.2. Aplicação de Padrões Arquiteturais - MVP
- 



Referências

BROWN, William J. et al. **AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis**. 1. ed. Wiley, 2008. 336 p.

BUSCHMANN, Frank; SOMERLAD, Peter. **Pattern-Oriented Software Architecture, a System of Patterns: Volume 1**. 1. ed. Wiley, 1996. 476 p.

COULOURIS, George; DOLLIMORE, Jean; KINDBERG, Tim. **Sistemas Distribuídos: conceitos e projeto**. 4. ed. Porto Alegre: Bookman, 2007. ISBN: 9788560031498.

FOWLER, Martin. **Patterns of Enterprise Application Architecture**. 1. ed. Addison-Wesley Professional, 2002. 560 p.

FOWLER, Martin. **Refactoring: Improving the Design of Existing Code**. 1. ed. Addison-Wesley Professional, 2018. 448 p.

FREEMAN, Eric. **Head First - Design Patterns**. 1. ed. O'Reilly, 2004. 638 p.


GAMMA, Erich et al. **Design Patterns: Elements of Reusable Object-Oriented Software**. 1. ed. Addison-Wesley Professional, 1994. 416 p.

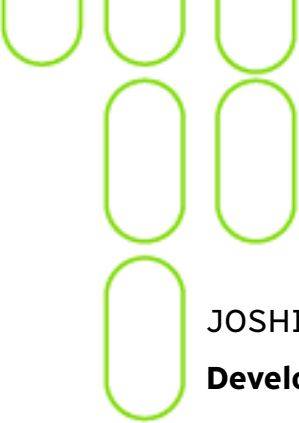
HALL, Gary M. **Adaptive Code via C#: Agile coding with design patterns and SOLID principles**. 1. ed. Microsoft Press, 2014. 401 p.

HOHPE, Gregor. **The Software Architect Elevator: Redefining the Architect's Role in the Digital Enterprise**. 1. ed. O'Reilly, 2020. 350 p.

HOHPE, Gregor; WOOLF, Bobby. **Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions**. 1. ed. Addison-Wesley Professional, 2003. 736 p.

INGENO, Joseph. **Software Architect's Handbook: Become a successful software architect by implementing effective architecture concepts**. 1. ed. Packt Publishing, 2018. 594 p.





JOSHI, Bipin. **Beginning SOLID Principles and Design Patterns for ASP.NET Developers**. 1. ed. Apress, 2016. 420 p.

KEELING, Michael. **Design It!: From Programmer to Software Architect**. 1. ed. O'Reilly, 2017. 354 p.

KRAFGIZ, Dirk; BANKE, Kalr; SLAMA, Dirk. **Enterprise SOA: Service-Oriented Architecture Best Practices**. 1. ed. Editora Prentice Hall, 2004. 408 p.

MALVEAU, Raphael C. et al. **AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis**. 1. ed. Wiley, 1998. 336 p.

MALVEAU, Raphael C.; MOWBRAY Thomas J. **Software Architect Bootcamp**. 1. ed. Prentice Hall, 2000. 352 p.

MARTIN, Robert C. **Clean Architecture: A Craftsman's Guide to Software Structure and Design**. 1. ed. Pearson, 2017. 432 p.

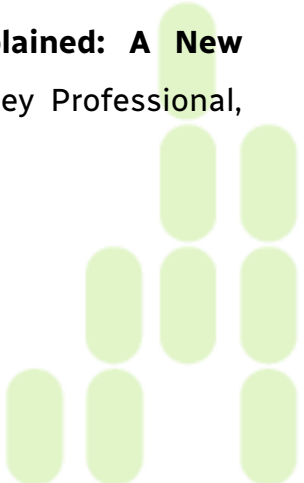
MIKOWSKI, Michael; POWELL, Josh. **Single Page Web Applications: JavaScript end-to-end**. 1. ed. Greenwich:Manning Publications, 2013. 432 p.

MILLET, Scott; TUNE, Nick. **Patterns, Principles and Practices of Domain-Driven Design**. 1. ed. Wrox, 2014. 800 p.

MONSON-HAEFEL, Richard. **97 Things Every Software Architect Should Know: Collective Wisdom from the Expert**. 1. ed. O'Reilly, 2009. 220 p.

PAI, Praseed; XAVIER, Shine. **.NET Design Patterns**. 1. ed. Packt Publishing, 2017. 314 p.

SHALLOWAY, Alan; TROTT, James R. **Design Patterns Explained: A New Perspective on Object Oriented Design**. 2. ed. Addison-Wesley Professional, 2014. 480 p.





SHYAM, Seshadri; BRAD, Green. **AngularJS: Up and Running: Enhanced Productivity with Structured Web Apps Paperback**. 1. ed. Sebastopol:O'Reilly Media, 2014. 275 p.

TANENBAUM, Andrew S.; STEEN, Maarten Van. **Sistemas Distribuídos: princípios e paradigmas**. 2. ed. Pearson/Prentice-Hall, 2007. ISBN: 9788576051428.

