

# ARQUITETURA DE SOFTWARE

João Choma Neto

[joao.choma@unicesumar.edu.br](mailto:joao.choma@unicesumar.edu.br)

<https://github.com/JoaoChoma/arquitetura-software2025>

Unicesumar – Maringá

## 2 BIMESTRE

- Arquitetura monolítica
- Arquitetura em camadas

# ARQUITETURA MONOLÍTICA

# MONOLÍTICA

- A arquitetura monolítica é um padrão de arquitetura de software em que uma aplicação é construída como uma única unidade de implementação e implantação.

# COMPOSIÇÃO

- **Interface do Usuário (UI):**
  - A camada de interface do usuário na arquitetura monolítica é responsável por interagir com os usuários finais.
- **Lógica de Negócios:**
  - A camada de lógica de negócios contém as regras e processos de negócios da aplicação. Essa camada é responsável por manipular as solicitações do usuário.
- **Acesso a Dados:**
  - A camada de acesso a dados é responsável por interagir com o banco de dados ou outros sistemas de armazenamento de dados.

## MONOLÍTICO

---

A arquitetura monolítica refere-se à forma como uma aplicação é estruturada e implantada.

---

Em uma arquitetura monolítica, todos os componentes da aplicação são agrupados e implantados juntos como uma única unidade.

---

Uma aplicação monolítica pode ou não seguir o padrão MVC

## MVC

- O padrão MVC é uma forma de estruturar o código dentro de uma aplicação
- O padrão MVC é independente da forma como a aplicação é implantada.
- O MVC pode ser aplicado tanto em arquiteturas monolíticas quanto em arquiteturas distribuídas, como microserviços

# EXEMPLOS

- **Sistemas de E-commerce**
- **Ferramentas de Gestão de Projetos**
- **Aplicações Financeiras**
- **Sistemas de Reservas e Bilheteira**
- **Plataformas de Ensino à Distância**
  - Moodle

# BLOG

- Um exemplo comum de aplicação implementada com arquitetura monolítica é um blog
- Um blog monolítico é uma aplicação web que permite aos usuários criar, editar, publicar e visualizar postagens de blog.
- No caso de um blog monolítico, todas as funcionalidades são implementadas como parte de uma única aplicação. Isso inclui o frontend (interface de usuário), backend (lógica de negócios) e acesso a dados (gerenciamento de banco de dados). Todas essas partes são desenvolvidas, testadas e implantadas juntas como uma unidade coesa.



# ARQUITETURA EM CAMADAS

# CAMADAS

- A arquitetura em camadas é um padrão de design de software que organiza um sistema em camadas distintas, onde cada camada tem uma responsabilidade específica e se comunica apenas com camadas adjacentes.
- Essa abordagem visa separar as preocupações e promover a modularidade, flexibilidade e reusabilidade do código.

# CAMADAS

- A arquitetura em camadas geralmente segue a seguinte estrutura:
- 1. Camada de Apresentação (UI - User Interface):** Responsável por interagir com o usuário, renderizar a interface gráfica e capturar entradas.
  - 2. Camada de Aplicação (Application Layer):** Contém a lógica da aplicação e orchestra as operações entre a interface e a camada de negócios.
  - 3. Camada de Negócio (Business Layer):** Implementa as regras de negócio e lógica principal do sistema.
  - 4. Camada de Persistência (Data Access Layer - DAL):** Responsável pela comunicação com o banco de dados.
  - 5. Camada de Banco de Dados (Database Layer):** Onde os dados são armazenados e gerenciados.

# BENEFÍCIOS

- **Separação de Responsabilidades:** Cada camada tem uma função bem definida, facilitando a manutenção e evolução do sistema.
- **Facilidade de Testes:** Como cada camada é modular, testes unitários podem ser aplicados a cada uma separadamente.

# BENEFÍCIOS

- **Reutilização de Código:** Camadas podem ser reaproveitadas em diferentes projetos ou módulos do sistema.
- **Escalabilidade:** Facilita a distribuição da aplicação em diferentes servidores.
- **Flexibilidade e Substituição de Tecnologias:** Possibilita a troca de tecnologias em camadas específicas sem impactar outras partes do sistema.

# LIMITAÇÕES

- **Desempenho:** O tráfego entre camadas pode impactar a performance, especialmente em sistemas de grande porte.
- **Rigidez:** Seguir estritamente o modelo pode dificultar a evolução do software.
- **Complexidade Adicional:** Implementar a separação de camadas pode exigir um esforço inicial maior.

# CAMADAS

- A arquitetura em camadas está intimamente relacionada a outros padrões de design de software, como MVC (Model-View-Controller), MVVM (Model-View-ViewModel)

# REDE DE COMUNICAÇÃO

- Um dos modelos mais conhecidos que utiliza a arquitetura em camadas é o Modelo de Referência OSI (Open Systems Interconnection) e o Modelo TCP/IP.