

# Arquitetura de Software

## Trabalho Final

### Objetivo

Aplicar os conceitos estudados em sala projetando e implementando parcialmente um sistema real, documentado segundo o modelo arquitetural escolhido.

---

### Etapas do Trabalho

#### 1. Escolha do Modelo de Documentação

Cada grupo deverá escolher **um** dos modelos abaixo para guiar a documentação do sistema:

- **Modelo C4**  
(Contexto, Contêiner, Componentes e Código)
- **Modelo 4+1**  
(Visões: Lógica, de Desenvolvimento, de Processo, Física e de Casos de Uso)

#### 2. Escolha do domínio da aplicação

A aplicação escolhida deve possuir um grau de complexidade adequado para o modelo de documentação e o estilo arquitetural.

#### 3. Sorteio do Estilo Arquitetural

Cada grupo receberá **por sorteio** um dos seguintes estilos arquiteturais:

- Microkernel
- Hexagonal (Ports and Adapters)
- Clean Architecture
- Orientada a Serviços (SOA)
- Baseada em Eventos (Event-Driven)
- Serverless

O sistema deverá ser projetado **seguindo MAJORITARIAMENTE os princípios do estilo arquitetural sorteado**.

#### 4. Projeto Arquitetural

Com base no modelo e estilo definidos, o grupo deverá **documentar o sistema** através de:

- Diagramas e descrições para **cada visão** exigida pelo modelo escolhido.
- Identificação dos **principais componentes**, suas responsabilidades e interações.
- Definição clara dos **mecanismos de comunicação e fluxos de dados**.
- Indicação das **decisões arquiteturais** relevantes (com justificativas).

#### 5. Implementação Parcial

O grupo deve **implementar parcialmente** o sistema, de modo a demonstrar:

- A **estrutura arquitetural proposta**;
- O **funcionamento de pelo menos um fluxo principal** do sistema;
- A **adoção real do estilo arquitetural** sorteado.

---

### Entregáveis

1. Documento de arquitetura completo (PDF ou repositório com README estruturado).
  2. Diagramas (em formato legível, preferencialmente PlantUML ou Draw.IO).
  3. Código-fonte e diagramas hospedados no GitHub.
-

# Exemplos de Domínios

## 1. Sistema Bancário e PIX

Sistema voltado à simulação de transferências bancárias, gerenciamento de contas e chaves PIX. Deve permitir ao usuário consultar saldo, realizar transferências e receber notificações de operações concluídas.

**Adequado para:** Clean Architecture, Hexagonal, Event-Driven, Serverless

---

## 2. Histórico e Monitoramento de Saúde

Aplicativo para registrar consultas, exames e uso de medicamentos, com linha do tempo de saúde pessoal. Pode incluir lembretes de consultas e acompanhamento de indicadores (peso, pressão, glicemia etc.).

**Adequado para:** Clean Architecture, Microkernel

---

## 3. E-commerce / Loja Virtual

Sistema de catálogo e vendas de produtos com carrinho, pedidos e controle de estoque. Pode incluir integração com sistemas de pagamento e notificações de entrega.

**Adequado para:** Hexagonal, SOA, Clean Architecture

---

## 4. Plataforma de Ensino Online (E-learning)

Plataforma que permite cadastrar cursos, aulas e alunos, com progresso e feedback.

Pode ter módulos opcionais (como fórum, certificados ou ranking).

**Adequado para:** SOA, Microkernel

---

## 5. Sistema de Notificações e Mensageria

Central de envio de alertas por e-mail, SMS ou push, reagindo a eventos do sistema. Permite definir regras personalizadas e canais de entrega.

**Adequado para:** Event-Driven, Serverless

---

## 6. Gerenciamento de Frotas e Entregas

Sistema que acompanha motoristas, veículos e rotas, gerando alertas de manutenção ou atraso. Pode incluir mapa de rastreamento e relatórios de desempenho.

**Adequado para:** SOA, Event-Driven

---

## 7. Sistema de Gestão Escolar

Sistema para escolas ou universidades, com cadastros de alunos, turmas e notas. Permite acesso diferenciado para professores, alunos e administradores.

**Adequado para:** Clean Architecture, Hexagonal

---

## 8. Sistema de Monitoramento Climático / IoT

Sistema que coleta dados de sensores (temperatura, umidade, pressão) e dispara alertas em condições anormais.

**Adequado para:** Event-Driven, Serverless

---

## 9. Agendamento e Serviços (Consultas, Salão, etc.)

Sistema que permite a clientes agendar horários, confirmar presença e visualizar disponibilidade. Pode incluir notificações automáticas e cancelamentos.

**Adequado para:** Clean Architecture, SOA

---

## 10. Sistema de Reserva de Hotéis e Viagens

Permite ao usuário buscar hotéis, reservar quartos e gerenciar reservas, com integração de pagamentos.

**Adequado para:** SOA, Event-Driven, Clean Architecture

---

## 11. Marketplace de Freelancers

Plataforma que conecta profissionais a clientes, gerenciando projetos, propostas e pagamentos.

**Adequado para:** Microkernel, Hexagonal, Clean Architecture

---

## 12. Sistema de Biblioteca Digital

Gerencia acervo de livros digitais e físicos, empréstimos, reservas e histórico de leitura.

**Adequado para:** Hexagonal, Clean Architecture

---

## 13. Aplicativo de Transporte e Mobilidade Urbana

Permite criar rotas, agendar corridas ou compartilhar veículos, e gerenciar usuários e pagamentos.

**Adequado para:** SOA, Event-Driven

---

## 14. Sistema de Gerenciamento de Energia / Smart Grid

Coleta dados de consumo, gera alertas e permite otimização de energia em residências ou prédios inteligentes.

**Adequado para:** Event-Driven, Serverless, Clean Architecture

---