

**UC de Modelagem de Software**

**UC de Programação e Soluções Computacionais**

Professora Edjane Mikaely

Professor Erivelton Lima

Professor José Padilha

Professor Alex Michael

Professor Wanfranklin Moreira

Professor Glauber Galvão

Esse projeto de avaliação A3 é um projeto transdisciplinar envolvendo as Unidades Curriculares identificadas acima e deverá ser construído para as duas Unidades Curriculares. Deverá ser apresentado conforme informado por cada professor identificado.

**Regras:**

**1. Os temas do desenvolvimento dos projetos serão sorteados e divulgados para os grupos.**

**2. O grupo deverá obrigatoriamente ser composto por 5 alunos, as exceções deverão ser aprovadas previamente pelos professores.**

**3. Os temas sorteados serão:**

- Sistema de Cadastro de Academia;
- Sistema de Reserva de Hotéis;
- Sistema de Gestão de Lanchonetes;
- Sistema de Biblioteca;
- Sistema de Cadastro de Alunos;
- Sistema de Gestão de Estoque.

**4. Etapas Comuns para Todos os Projetos**

**4.1. Planejamento (1 semana)**

- Definição dos requisitos da aplicação: Identificar e documentar os requisitos funcionais e não funcionais.
- Planejamento das tarefas e sprints: Dividir o projeto em tarefas menores e organizar sprints para gerenciar o desenvolvimento.

**4.2. Desenvolvimento (3 semanas)**

- Implementação das funcionalidades principais: Desenvolver as principais funcionalidades do sistema, como cadastro, consulta, atualização e exclusão de dados (CRUD).
- Criação de testes unitários, de integração e de sistema: Escrever testes para garantir que cada parte do sistema funcione corretamente.

**4.3. Integração (1 semana)**

- Configuração do repositório de controle de versão: Utilizar ferramentas como Git para versionamento do código.
- Execução automatizada de testes: Configurar scripts para execução automática dos testes.
- Configuração de pipelines de integração contínua: Implementar pipelines para integração contínua utilizando ferramentas como Jenkins ou GitHub Actions.

**4.4. Documentação (1 dias)**

Criação de documentação completa do projeto: Documentar todos os aspectos do projeto, incluindo requisitos, diagramas UML, diagramas de processos de negócios BPM e protótipos no Figma.

## 5. Desenvolvimento dos projetos de Modelagem:

### 5.1. Sistema de Cadastro de Academia

- Requisitos Funcionais: Cadastro de membros, gestão de planos de treino, registro de pagamentos, histórico de atividades.
- Requisitos Não Funcionais: Segurança dos dados, desempenho, usabilidade.
- UML: Diagrama de casos de uso, diagrama de classes, diagrama de sequência.
- BPM: Diagrama de processos de negócios para o fluxo de cadastro e gestão de membros.
- Prototipação: Interface de usuário no Figma para cadastro e gestão de membros.

### 5.2. Sistema de Reserva de Hotéis

- Requisitos Funcionais: Reserva de quartos, gestão de disponibilidade, check-in/check-out, pagamentos.
- Requisitos Não Funcionais: Escalabilidade, segurança, desempenho.
- UML: Diagrama de casos de uso, diagrama de classes, diagrama de sequência.
- BPM: Diagrama de processos de negócios para o fluxo de reserva e gestão de quartos.
- Prototipação: Interface de usuário no Figma para reserva e gestão de quartos.

### 5.3. Sistema de Gestão de Lanchonetes

- Requisitos Funcionais: Cadastro de produtos, gestão de pedidos, controle de estoque, pagamentos.
- Requisitos Não Funcionais: Usabilidade, segurança, desempenho.
- UML: Diagrama de casos de uso, diagrama de classes, diagrama de sequência.
- BPM: Diagrama de processos de negócios para o fluxo de pedidos e gestão de estoque.
- Prototipação: Interface de usuário no Figma para gestão de pedidos e estoque.

### 5.4. Sistema de Biblioteca

- Requisitos Funcionais: Cadastro de livros, gestão de membros, registro de empréstimos, histórico de devoluções.
- Requisitos Não Funcionais: Segurança dos dados, desempenho, usabilidade.
- UML: Diagrama de casos de uso, diagrama de classes, diagrama de sequência.
- BPM: Diagrama de processos de negócios para o fluxo de empréstimo e devolução de livros.
- Prototipação: Interface de usuário no Figma para gestão de empréstimos e cadastro de livros.

### 5.5. Sistema de Cadastro de Alunos

- Requisitos Funcionais: Cadastro de alunos, gestão de cursos, registro de notas, histórico acadêmico.
- Requisitos Não Funcionais: Segurança dos dados, desempenho, usabilidade.
- UML: Diagrama de casos de uso, diagrama de classes, diagrama de sequência.
- BPM: Diagrama de processos de negócios para o fluxo de cadastro e gestão de alunos.
- Prototipação: Interface de usuário no Figma para cadastro e gestão de alunos.

### 5.6. Sistema de Gestão de Estoque

- Requisitos Funcionais: Cadastro de produtos, controle de entradas e saídas, geração de relatórios de inventário.
- Requisitos Não Funcionais: Escalabilidade, segurança, desempenho.
- UML: Diagrama de casos de uso, diagrama de classes, diagrama de sequência.
- BPM: Diagrama de processos de negócios para o fluxo de gestão de estoque.
- Prototipação: Interface de usuário no Figma para gestão de estoque e cadastro de produtos.

## 6.0 Etapas do Desenvolvimento

### 6.1 Sistema de Cadastro de Academia

#### Atributos mínimos do Banco de Dados

1. **Membros:** ID, Nome, CPF, Telefone, Endereço, Data de Cadastro.
2. **Treinos:** ID, ID\_Membro, Tipo, Descrição, Duração, Data\_Início.
3. **Pagamentos:** ID, ID\_Membro, Valor, Data\_Pagamento, Status.
4. **Histórico de Atividades:** ID, ID\_Membro, Atividade, Data, Tempo\_Execução.
5. **Funcionários:** ID, Nome, Cargo, Login, Senha.

#### Diagrama Lógico

- **Relacionamentos:**
  - Membros com Treinos e Pagamentos.
  - Funcionários relacionados às atividades.

#### Telas

1. Tela de Cadastro de Membros (Formulário).
2. Tela de Gerenciamento de Treinos.
3. Tela de Registro de Pagamentos.
4. Tela de Histórico de Atividades.
5. Tela de Login de Funcionários.

#### CRUD

- **Create:** Cadastro de membros, treinos e pagamentos.
- **Read:** Consultar histórico de atividades.
- **Update:** Atualizar informações dos membros ou treinos.
- **Delete:** Excluir membros ou treinos.

### 6.2 Sistema de Reserva de Hotéis

#### Atributos mínimos do Banco de Dados

1. **Quartos:** ID, Número, Tipo, Status, Valor\_Diária.
2. **Reservas:** ID, ID\_Quarto, ID\_Cliente, Data\_CheckIn, Data\_CheckOut, Valor\_Total.
3. **Clientes:** ID, Nome, CPF, Telefone, Email.
4. **Pagamentos:** ID, ID\_Reserva, Valor, Data\_Pagamento, Método.
5. **Funcionários:** ID, Nome, Cargo, Login, Senha.

#### Diagrama Lógico

- **Relacionamentos:**
  - Reservas conectadas aos clientes e quartos.
  - Pagamentos relacionados às reservas.

#### Telas

1. Tela de Cadastro de Clientes.
2. Tela de Reservas (Formulário e Tabela de Consultas).
3. Tela de Check-in e Check-out.
4. Tela de Gestão de Pagamentos.
5. Tela de Login de Funcionários.

#### CRUD

- **Create:** Criar reservas e pagamentos.
- **Read:** Consultar disponibilidade de quartos.
- **Update:** Alterar datas de reserva ou cliente.
- **Delete:** Cancelar reservas.

## 6.3 Sistema de Gestão de Lanchonetes

### Atributos mínimos do Banco de Dados

1. **Produtos:** ID, Nome, Categoria, Preço, Estoque.
2. **Pedidos:** ID, ID\_Cliente, ID\_Produto, Quantidade, Valor\_Total, Status.
3. **Clientes:** ID, Nome, CPF, Telefone.
4. **Funcionários:** ID, Nome, Cargo, Login, Senha.
5. **Estoque:** ID, ID\_Produto, Quantidade\_Entrada, Quantidade\_Saída.

### Diagrama Lógico

- **Relacionamentos:**
  - Pedidos ligados a clientes e produtos.
  - Estoque integrado aos produtos.

### Telas

1. Tela de Cadastro de Produtos.
2. Tela de Gerenciamento de Pedidos.
3. Tela de Controle de Estoque.
4. Tela de Gestão de Clientes.
5. Tela de Relatórios.

### CRUD

- **Create:** Cadastro de produtos e pedidos.
- **Read:** Consultar estoque e pedidos.
- **Update:** Atualizar informações dos produtos ou pedidos.
- **Delete:** Excluir produtos ou cancelar pedidos.

## 6.4 Sistema de Biblioteca

### Atributos mínimos do Banco de Dados

1. **Livros:** ID, Título, Autor, Categoria, Disponibilidade.
2. **Membros:** ID, Nome, CPF, Telefone, Email.
3. **Empréstimos:** ID, ID\_Livro, ID\_Membro, Data\_Empréstimo, Data\_Devolução, Status.
4. **Funcionários:** ID, Nome, Cargo, Login, Senha.
5. **Histórico de Empréstimos:** ID, ID\_Membro, ID\_Livro, Data\_Ação.

### Diagrama Lógico

- **Relacionamentos:**
  - Livros conectados a empréstimos e histórico.
  - Membros com múltiplos empréstimos.

### Telas

1. Tela de Cadastro de Livros e Membros.
2. Tela de Registro de Empréstimos.
3. Tela de Gestão de Devoluções.
4. Tela de Histórico de Empréstimos.
5. Tela de Relatórios.

### CRUD

- **Create:** Cadastro de livros e membros.
- **Read:** Consultar disponibilidade de livros.
- **Update:** Atualizar status de empréstimos.
- **Delete:** Excluir registros de membros ou livros.

## 6.5 Sistema de Cadastro de Alunos

### Atributos mínimos do Banco de Dados

1. **Alunos:** ID, Nome, CPF, Email, Telefone, Endereço.

2. **Cursos:** ID, Nome, Categoria, Duração, Status.
3. **Notas:** ID, ID\_Aluno, ID\_Curso, Nota, Data\_Lançamento.
4. **Histórico Acadêmico:** ID, ID\_Aluno, ID\_Curso, Data, Status.
5. **Funcionários:** ID, Nome, Cargo, Login, Senha.

#### Diagrama Lógico

- **Relacionamentos:**
  - Alunos associados a cursos e notas.
  - Histórico ligado a alunos e cursos.

#### Telas

1. Tela de Cadastro de Alunos e Cursos.
2. Tela de Registro de Notas.
3. Tela de Consultas de Histórico.
4. Tela de Gerenciamento de Funcionários.
5. Tela de Relatórios Acadêmicos.

#### CRUD

- **Create:** Cadastro de alunos e cursos.
- **Read:** Consultar notas e histórico.
- **Update:** Atualizar dados dos alunos ou notas.
- **Delete:** Excluir registros de alunos ou cursos.

## 6.6 Sistema de Gestão de Estoque

#### Atributos mínimos do Banco de Dados

1. **Produtos:** ID, Nome, Categoria, Preço, Quantidade.
2. **Entradas:** ID, ID\_Produto, Quantidade, Data\_Entrada, Fornecedor.
3. **Saídas:** ID, ID\_Produto, Quantidade, Data\_Saída, Destinatário.
4. **Funcionários:** ID, Nome, Cargo, Login, Senha.
5. **Relatórios:** ID, Tipo, Data, Detalhes.

#### Diagrama Lógico

- **Relacionamentos:**
  - Produtos vinculados às entradas e saídas.
  - Relatórios relacionados às movimentações.

#### Telas

1. Tela de Cadastro de Produtos.
2. Tela de Controle de Entradas e Saídas.
3. Tela de Relatórios de Estoque.
4. Tela de Gestão de Funcionários.
5. Tela de Consultas.

#### CRUD

- **Create:** Cadastro de produtos e entradas.
- **Read:** Consultar relatórios e estoque.
- **Update:** Atualizar informações dos produtos ou entradas.
- **Delete**

## 7. As Linguagens de Programação

- Os projetos desenvolvidos nesse projeto **deverão ser criados usando a linguagem de programação JAVA**, exceção aos **alunos do professor Wanfranklin Moreira** que usarão o **Dot Net com C# conforme orientação do professor**.
- O banco de dados **deverá ser o MySQL**.

- Para a apresentação do código, deverá ser feita fazendo um **Git Clone do projeto depositado no GitHub** e, na **semana anterior a apresentação**, os grupos devem trazer o **script do MySQL** para que seja **criado o banco de dados na máquina do professor**.
- A máquina do laboratório **possui as configurações necessárias para gerar a apresentação**. É importante que um membro do grupo simule a operação de Git Clone para baixar o código para a máquina da apresentação para agilizar o processo de clonagem.
- A execução da aplicação deverá ser feita obrigatoriamente para os desenvolvimentos em Java no VSCode.
- É importante ter um notebook de backup para emergência configurado apresentação em caso de alguma emergência.

### Observações

- **A avaliação A3 é uma avaliação Colaborativa e Cooperativa**, logo, **a nota será atribuída ao grupo!** O Estudante que não souber responder as perguntas sobre o projeto, irá prejudicar a nota de toda a equipe. Afinal, **TODOS DEVEM** participar do desenvolvimento dos projetos e ter o conhecimento sobre tudo que foi criado. **Não existe “minha parte foi essa”!**
- A lista com o nome dos componentes do grupo deve ser entregue até uma semana antes da apresentação.
- Cursos da Oracle vale 25% da nota da A3 (10 pontos de 40) da A3.
- É **obrigatório a apresentação da Avaliação A3**, com a ausência, o aluno **terá toda a nota zerada** e será **desconsiderada a nota da Oracle**. A exceção será em caso de doença infectocontagiosa ou traumatismo incapacitante e deverá apresentar atestado médico com assinatura e carimbo do médico com o CRM, timbre da clínica/hospital/UPA, CID da avaliação médica e prazo do atestado para a falta, **até o dia da apresentação**. O mesmo fará uma avaliação oral individual do projeto em horário disponibilizado pelo professor pelo meio que o professor considerar melhor.
- A nota da apresentação corresponderá a 75% da nota (30 pontos de 40) da A3.
- A apresentação deverá ser **o sistema em funcionamento conforme os requisitos apresentados nesse projeto**. A utilização de mockups na apresentação será considerado que o grupo não cumpriu com os requisitos da avaliação e toda a nota será zerada inclusive a da Oracle.
- O grupo pode exceder os requisitos apresentados e entregar mais do que foi solicitado se desejar.
- É obrigatório subir (fazer commit) do código para o github **até o dia 8 de junho**. Após essa data o código não poderá ser alterado.
- O não cumprimento do prazo acarreta uma penalização de 12,5 % da nota (ou 5 pontos de 40) por dia atrasado.
- Todos os membros do projeto deverão estar preparados para defesa na hora da apresentação.
- As apresentações ocorrerão entre os dias 13 à 17 de junho.