**Roteiro de Projeto Web/Mobile Extensionista**

**Avaliação POO JAVA PBL - Aplicação Web/Mobile**

**Instituição**: Wyden | UniRuy  
**Disciplina**: Programação Orientada a Objetos em Java  
**Data**: 17, setembro de 2025  
**Peso**: Projeto Web/Mobile **vale 06 pontos**

**1. Introdução**

Nesta seção, o aluno apresentará o projeto de forma geral, explicando o contexto da aplicação escolhida, o seu propósito e a motivação para o desenvolvimento. A introdução deve abordar a relevância da aplicação no mundo real e como ela pode solucionar um problema específico.

**Exemplo de Introdução**:

O projeto tem como objetivo desenvolver uma aplicação web/mobile para o controle de estoque de produtos em uma loja. A aplicação permitirá a gestão eficiente dos itens, controle de entradas e saídas, além de gerar relatórios de vendas. A escolha dessa aplicação visa facilitar a vida dos empresários e lojistas que buscam otimizar o controle de seu estoque de maneira simples e intuitiva.

**2. Objetivos da Aplicação**

Nesta seção descrever os objetivos principais da aplicação. O que ela será capaz de realizar? Quais são suas funcionalidades principais?

**Exemplo de Objetivos**:

* Gerenciar o estoque de produtos (entrada, saída e atualização de quantidade).
* Exibir relatórios de movimentação de produtos.
* Registrar informações dos produtos, como nome, categoria e preço.
* Permitir o controle de estoque em tempo real.
* Fornecer alertas de baixo estoque e expiração de produtos.

**3. Funções/Lista de Eventos (Funcionalidades) – RF / RNF**

Aqui o aluno deverá listar os **Requisitos Funcionais (RF)** e **Requisitos Não Funcionais (RNF)** da aplicação, se houver.

**Exemplo de Requisitos Funcionais (RF)**:

* **RF1**: O sistema deverá permitir o cadastro de produtos no estoque, com informações como nome, descrição, categoria e preço.
* **RF2**: O sistema deverá permitir o controle de entradas e saídas de produtos, atualizando automaticamente o estoque.
* **RF3**: O sistema deverá gerar relatórios mensais de movimentação de produtos, com gráficos e tabelas.

**Exemplo de Requisitos Não Funcionais (RNF)**:

* **RNF1**: A aplicação deverá ser responsiva, ou seja, funcionar tanto em dispositivos móveis quanto em desktops.
* **RNF2**: A aplicação deve ter um tempo de resposta inferior a 2 segundos para todas as ações realizadas pelo usuário.

**4. Especificação de Programas**

Nesta parte, serão detalhadas as especificações dos programas e as entidades envolvidas.

**a) Layout da Tela:**

Descreva os principais layouts das telas da aplicação, ou crie protótipos que demonstrem a interface gráfica do usuário (GUI). Por exemplo:

* Tela de login.
* Tela de cadastro de produto.
* Tela de relatório de movimentação.

**b) Regras de Negócio:**

Explique as principais regras que regem o funcionamento da aplicação. Isso inclui validações de entrada, cálculos ou processos lógicos que o sistema deverá realizar.

**Exemplo de Regras de Negócio**:

* O preço de um produto não pode ser negativo.
* A quantidade de um produto em estoque não pode ser inferior a zero após uma saída de estoque.

**c) Entidades Envolvidas (Classes):**

Apresente as classes principais da aplicação, com suas responsabilidades e os atributos que cada uma possui. As classes podem ser descritas ou diagramadas em UML.

**Exemplo de Classes**:

* **Produto**: Representa um produto do estoque, com atributos como nome, categoria, quantidade, preço, etc.
* **Estoque**: Responsável por controlar a entrada e saída de produtos, com métodos para adicionar e remover produtos.

**d) Tabelas:**

Liste e descreva as tabelas do banco de dados, incluindo os campos e tipos de dados.

**5. DER ou Diagrama de Classe**

**a) Artefato Gráfico:**

Crie um **Diagrama de Entidade-Relacionamento (DER)** ou um **Diagrama de Classe (UML)**, dependendo da abordagem adotada para o banco de dados e estrutura do sistema. Estes diagramas devem mostrar como as entidades/classes se relacionam entre si.

**b) Dicionário de Dados (DD):**

Apresente o dicionário de dados, explicando o significado de cada campo das tabelas (para banco de dados) ou atributos das classes.

**6. Aplicação Web/Mobile**

Aqui, você deve apresentar a aplicação web/mobile que desenvolvida, incluindo as funcionalidades principais.

**a) Menu/Submenu:**

O sistema deverá ter um menu principal com opções que direcionem para as principais funcionalidades da aplicação. Exemplo: Cadastro de produtos, Relatórios, Controle de Estoque etc.

**b) Telas Funcionais:**

Descreva as telas principais que o sistema terá, como a tela de login, tela de cadastro de produto, tela de controle de estoque, entre outras.

**c) Telas de Diálogo:**

Explique como serão as interações de diálogos entre o sistema e o usuário, como mensagens de erro, confirmação ou alertas.

**d) Layout Relatórios:**

Os relatórios devem ser apresentados de forma clara e intuitiva. Inclua exemplos de layout e gráficos que ajudem a apresentar as informações de forma acessível.

**7. Sugestões de Aplicações Web/Mobile**

A escolha do domínio da aplicação é livre, mas você pode escolher uma das sugestões fornecidas, como:

* **Aplicação de Controle de Estoque**
* **Aplicação Acadêmica**
* **Aplicação de Consórcio de Veículos**
* **Aplicação de Atendimento Médico**
* **Aplicação de Hóspedes de Hotel**
* **Aplicação de Cartão de Crédito**
* **Aplicação de Imóveis**
* **Aplicação Controle de Patrimônio**
* **Aplicação Agenda de Compromisso**
* **Aplicação de Contas a Pagar/Receber**

**Nota**: Ou sugerir outro domínio, desde que aprovado pelo professor.

**7. BAREMA – Avaliação, Critérios de Pontuação**

|  |  |  |  |
| --- | --- | --- | --- |
| **Barema Seminário: Projeto Web/Mobile: 06 pontos** | | | |
| **Projeto**: [0-4] | **PPT**: [0,5] | **Defesa**: [0,5] | **GitHub:** [0-1] |

**Considerações Finais**

1. **Equipe**: **Individual** ou **máximo** **02(dois) componentes**.
2. **Prazo de Entrega**: O prazo de entrega é uma semana antes da Data AV Institucional.
3. **Entrega do Trabalho**: Entregar tanto a aplicação web/mobile quanto a documentação do projeto.
4. **Não será aceita entrega com atraso**, a nota PROJETO WEB será ZERADA em caso de atraso.
5. **Criação de Repositório no GitHub**: O repositório deve ser público e o link postado na plataforma SAVA do aluno.
6. **Arquivo README**: Incluir no repositório do GitHub um arquivo README, contendo informações sobre os componentes da equipe e a descrição do projeto.
7. **Seminário**: Uma semana antes da Data AV Institucional, será realizado o seminário para apresentar a aplicação web/mobile desenvolvida.
8. **Tempo Apresentação:** 8min. a 10min – todos os componentes da equipe devem apresentar;
9. **Postar no SAVA: 01 semana antes do Seminário data da AV**, o link do GitHub.

**Nota:** No GitHub deve conter os nomes dos componentes, a parte escrita padrão ABNT, o código do projeto e o PPT da apresentação do seminário.