

Especificação Trabalho Prático

Detalhamento da Entrega 2 - Projeto Arquitetural

Entregáveis	1
1) Plataforma de Implementação:	1
2) Definir Táticas para tratar Atributos de qualidade:	1
3) Elaborar Arquitetura lógica do Sistema:	1

Entregáveis

1) Plataforma de Implementação:

- A. Definir a plataforma de hardware e software a ser usada: SGBD, Servidor Web, dispositivos, máquinas (um para o sistema inteiro);
- B. Definir tecnologias a serem usadas: frameworks de persistência, interface com o usuário, etc;
- C. Justificar a escolha tecnológica apresentando o contexto. Para guiar essa justificativa, tome como base as perguntas abaixo:

a. Enviar o cenário do trabalho respondendo (no formato de minimundo):

b. Qual localização geográfica dos usuários?

c. Qual o ambiente de hardware e software? Há restrições?

d. Qual a frequência de uso do sistema ou funcionalidade? E tempo de resposta esperada?

e. Qual o volume de dados esperado? (inicial e futuramente)

f. Há restrições de confiabilidade? (tempo entre falhas) Tarefas críticas, que envolvam risco ou de alto impacto no negócio?

g. Há restrições de segurança? dados confidenciais, por exemplo?

h. Quais os perfis dos usuários? são leigos, experts, usuários frequentes, esporádicos? possuem alguma dificuldade ou limitação?

i. Há alguma previsão de mudanças futuras? há alterações no negócio (tipo de produtos, etc)?

2) Definir Táticas para tratar Atributos de qualidade:

- A. Agrupar Requisitos Não Funcionais em categorias e definir táticas para atendê-los, conforme o exemplo abaixo:

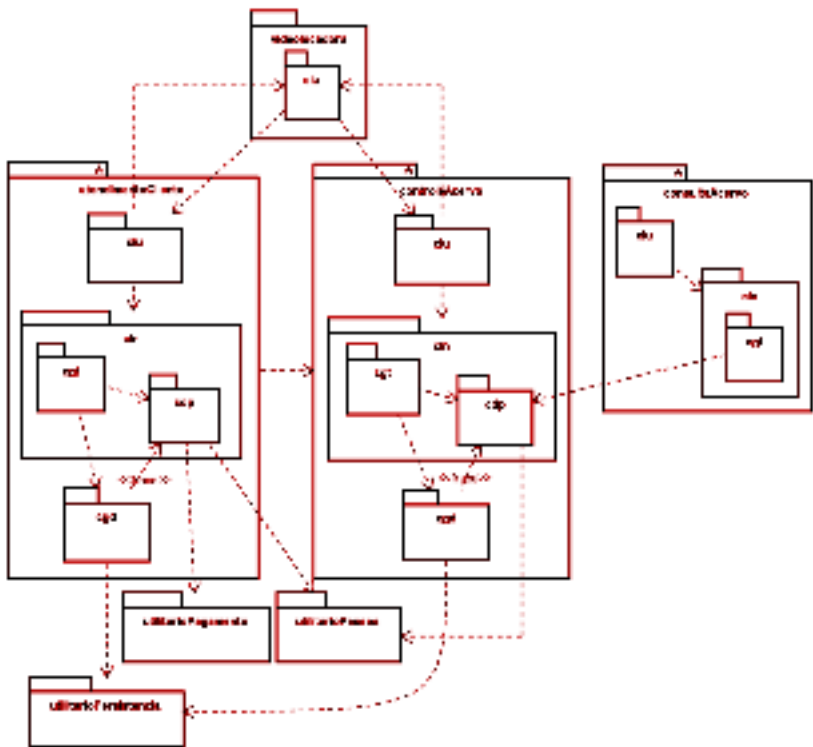
Exemplo:

Categoria	Requisitos Não Funcionais Considerados	Condutor da Arquitetura	Tática
Facilidade de Operação	RNF03, RNF08	Sim	<ul style="list-style-type: none">Prover ao usuário a capacidade de entrar com comandos que permitam operar o sistema de modo mais eficiente. Para tal, as interfaces do sistema devem permitir, sempre que possível, a entrada por meio de seleção ou leitura de código de barras ao invés da digitação de campos.
Segurança de Acesso	RNF01	Sim	<ul style="list-style-type: none">Identificar usuários usando login e autenticá-los por meio de senha.Autorizar usuários, criando os seguintes grupos: (i) Gerente de Acervo –acesso às funcionalidades do controle de acervo; (ii) Atendente – acesso às funcionalidades de atendimento a clientes; (iii) Administrador – acesso geral a todas as funcionalidades do sistema, incluindo o cadastro de usuários.Limitar a exposição, disponibilizando pela Internet somente funcionalidades de consulta ao acervo.
Manutenibilidade	RNF07, RNF09	Sim	<ul style="list-style-type: none">Uso de um intermediário para isolar o mecanismo de persistência de dados.Separar a interface do restante da aplicação.Uso de um intermediário para tratar as requisições da interface.

Consulte as táticas descritas na **Seção 3.7 - Tática para Tratar Atributos de Qualidade**, da apostila do Ricardo Falbo

3) Elaborar Arquitetura lógica do Sistema:

- A. Elaborar arquitetura lógica por meio do diagrama de classes da UML, conforme exemplo abaixo. Represente camadas e componentes como sendo pacotes. Estabeleça a relação de dependência entre eles.



B. Justifique a arquitetura, a divisão em camadas, as partições (ou outros estilos arquiteturais). Exemplo:

A arquitetura de software do sistema baseia-se em uma combinação dos estilos em partições e camadas. Inicialmente, duas partições principais foram definidas, procurando-se preservar a divisão em subsistemas realizada na fase de análise: atendimentoCliente e controleAcervo. Cada uma dessas partições, por sua vez, está organizada em três camadas, a saber: Camada de Interface com o Usuário (ciu), Camada de Lógica de Negócio (cln) e Camada de Gerência de Dados (cgd). A Figura 1 mostra a arquitetura proposta. No projeto da Camada de Lógica de Negócio, optou-se por usar o padrão Camada de Serviço. Assim, essa camada foi subdividida em dois pacotes: Componente de Domínio do Problema (cdp) e Componente de Gerência de Tarefas (cgt). Além disso, visando ao desenvolvimento para e com reúso, foi reutilizado o Utilitário Pessoa (utilitarioPessoa) e criado o framework Pagamento (utilitarioPagamento)¹.