

Sistemas Corporativos

Documento de Arquitetura de Software

Versão 1.0

Sumário

1. Introdução ao documento de Arquitetura de Software.....	3
1.1 Propósito	3
1.2 Escopo	3
1.3 Visão Geral do Documento de Arquitetura	3
1.4 Metodologias de Desenvolvimento.....	4
1.5 Representação Arquitetural do Documento de Arquitetura de Software	4
1.5.1 Visão de Casos de Uso	4
1.5.2 Visão Lógica.....	4
1.5.3 Visão de Processos.....	4
1.5.4 Visão da Implantação.....	4
3. Metas e Restrições da Arquitetura	5
4. Visão de caso de uso.....	5
4.1 – Módulo Admin	5
4.1.1 – Verificar grupo de funções	5
4.1.2 – Movimentações e alterações.....	6
4.2 – Módulo Inventário.....	6
4.2.1 – Requisição de pedido de peças	6
4.3 – Módulo Vendas	6
4.3.1 – Adicionar item ao carrinho de compras	6
4.3.2 – Realizar compra	6
5. Visão Lógica	6

Documento de Arquitetura de Software

1. INTRODUÇÃO AO DOCUMENTO DE ARQUITETURA DE SOFTWARE

Por meio da introdução será apresentada uma visão geral do Documento de Arquitetura de software, onde será definido o propósito, seu escopo vinculado ao software que está em desenvolvimento, metodologia de desenvolvimento entre outros.

1.1 Propósito

O Documento de arquitetura fornece um molde de desenvolvimento que deve guiar as atividades do projeto e a implementação do software. Por representar as decisões significativas do projeto, o documento estabelece qual a estrutura do sistema, ele define um modelo de construção padrão para direcionar no desenvolvimento de todas as partes do sistema.

1.2 Escopo

O sistema trata-se de um e-commerce, divididos em 9 módulos diferentes que contêm ligações entre os mesmos, cada módulo é independente com suas respectivas responsabilidades separando o sistema em “pedaços”, ou seja, cada módulo será um sistema diferente que irá dar origem a um sistema maior de e-commerce.

1.3 Visão Geral do Documento de Arquitetura

Neste documento serão demonstrados os detalhes significativos da construção arquitetural do projeto. Na seleção 1.6 “Representação Arquitetural”, serão apresentadas as visões arquiteturais, e a forma como as visões se integram fornecendo uma completa perspectiva do sistema. Teremos uma breve descrição dos requisitos de software, que possuem um grande impacto representativo para estrutura arquitetural do sistema.

Além de conter requisitos não funcionais que possam ter influência significativa nas soluções arquiteturais no sistema, como performance e desempenho, e considerações sobre a estrutura de implementação do sistema. Teremos também documentado como o sistema estará sendo desenvolvido para consultas futuras de possíveis stakeholders.

1.4 Metodologias de Desenvolvimento

No desenvolvimento separamos o projeto em vários módulos cada módulo com seus próprios métodos de desenvolvimento, linguagem e tecnologias individuais com um porém todos devem seguir um padrão para se comunicar entre os mesmos. Ou seja cada módulo será um aplicação independente que no final fará parte do nosso projeto

1.5 Representação Arquitetural do Documento de Arquitetura de Software

1.5.1 Visão de Casos de Uso

Temos os analistas, gerentes de projeto e outros interessados nas regras de negócio do sistema, os casos de uso amostrados são de maior criticidade arquitetural, chamados de casos de uso arquiteturalmente significativo, esses casos de uso representam elementos do projeto que contribuem para o insucesso da implementação devido aos riscos técnicos, são considerados casos de uso arquiteturalmente significativos.

1.5.2 Visão Lógica

Na visão lógica, os implementadores poderão entender a decomposição do sistema em pacotes, camadas e classes, responsabilidades, mecanismos e dependências, ela permite ao sistema uma maior flexibilidade, extensibilidade que agrega uma qualidade maior ao sistema.

1.5.3 Visão de Processos

Esta visão exhibe os principais processos e as principais threads do sistema, mostrando a concorrência de tarefas em tempo de execução do sistema, devido a separação por módulos do nosso sistema, não estaremos adotando métodos para visualizar esta visão de processos pois na maioria dos casos cada módulo estará rodando em máquinas separadas.

1.5.4 Visão da Implantação

Utilizaremos um serviço chamado jenkins para fazer a validação das versões, teste e fazer o deploy no servidor que vai ser definido de acordo com a linguagem de implementação do módulo e o banco que o mesmo está utilizando, pois cada módulo tem conexão com seu próprio banco de dados definido pela equipe do

módulo, já o front-end, será desenvolvido de acordo com a linguagem que os integrantes do módulo definirem, porém, seguindo um padrão estipulado pelos arquitetos, para manter a integridade continuidade das conexões com o servidor central.

2. METAS E RESTRIÇÕES DA ARQUITETURA

Alguns recursos possuem restrições e recursos com necessidades específicas e que influenciam de maneira significativa na arquitetura da solução. São eles:

- Todos os módulos funcionam de forma individual e muitos deles possuem informações estritamente sigilosas, precisando de criptografia para dados de acesso e de uma conexão com comunicação segura;
- Garantia de funcionamento do servidor de integração;
- Todos os serviços devem ter coesão máxima, todos prevendo o máximo de erros que podem ocorrer e oferecendo opções para usuário caso ocorra uma falha;
- Um módulo não pode impossibilitar outros módulos de funcionar corretamente caso haja qualquer problema de comunicação;
- Em caso de falha de comunicação, são necessárias notificações ao administrador da solução para que sejam tomadas as ações corretivas necessárias.
- Por possuir diversas tabelas transacionais, é extremamente importante, pela integridade do sistema, que em caso de falha, toda a transação seja anulada.

3. VISÃO DE CASO DE USO

Esta parte do documento foi desenvolvida tendo como visão, salientar parte dos módulos os quais mais implicam na arquitetura e desenvolvimento do sistema pois representam as funcionalidades essências e significantes dentro do sistema.

3.1 – Módulo Admin

3.1.1 – Verificar grupo de funções

O caso de uso “Verificar grupo de funções” se encontra na parte de autenticação do usuário. Ele é responsável por verificar quais as funções cadastradas para aquele usuário, para que o mesmo tenha acesso somente ao que lhe foi concedido e não as demais funções do sistema.

3.1.2 – Movimentações e alterações

O caso de uso “Movimentações e alterações” é responsável por gravar todo e qualquer tipo de alteração em qualquer um dos módulos, para manter assim um controle interno no sistema gerando logs para essas alterações que o sistema sofrer.

3.2 – Módulo Inventário

3.2.1 – Requisição de pedido de peças

O caso de uso “Requisição de pedido de peças” fica responsável por fazer os pedidos das peças que estão faltando, verificando sua disponibilidade e dando autoria para ser aceito ou não. Ele não fica atrelado somente ao inventário como também a parte da manufatura da empresa, pois é la que se fazem o uso e requisições dessas peças, essas requisições que são passadas pelo modulo de inventário.

3.3 – Módulo Vendas

3.3.1 – Adicionar item ao carrinho de compras

O caso de uso “Adicionar item ao carrinho de compras ” é responsável por adicionar um produto ao cliente que está visitando o site. Essa é uma parte que exige grande integridade entre os módulos pois nele será verificado se o produto tem em estoque, descrição, fotos e tudo relacionado ao produto, inclusive posteriormente a parte da segurança em si.

3.3.2 – Realizar compra

O caso de uso “Realizar compra” conclui a compra do cliente abatendo no estoque o a quantidade total dos produtos que foram pegos pelo mesmo. Também há muitas integrações envolvidas nessa etapa do projeto.

4. VISÃO LÓGICA

Visão Geral

A visualização lógica da arquitetura nos dá uma visão geral de como será a arquitetura da aplicação e sua subdivisão em módulos. Descreve as classes mais importantes, suas organizações nos módulos e integração entre serviços e subsistemas. Descreve também os casos de uso mais importantes como os

principais aspectos da arquitetura. Os diagramas de classe podem ser incluídos para ilustrar os relacionamentos entre as classes, subsistemas, pacotes e camadas arquitetonicamente significantes.

A visualização lógica do Sistema Corporativo é composta por 8 módulos: Compras, Inventário, Produtos, Vendas, Produção, Administrativo, Pessoas e Recursos Humanos. Cada módulo é decomposto nos seguintes pacotes:

Apresentação

A camada de apresentação deve interagir via API com seu módulo utilizando tecnologia JSON para troca de informações entre o backend e o frontend, assim como para comunicação com outros módulos.

Controller

Contém classes para processamento das funcionalidades dentro do sistema, cada módulo tem suas próprias regras de negócio.

Service

Cada módulo tem suas tabelas que são representados por classes modelos e interagem com tabelas de módulos que também são representados por modelos. Cada classe de modelo terá seus getters e setters para acessar os atributos do banco de dados, além das validações de dados necessários para evitar duplicação de dados e erros de digitação.

Repositório

Contém classes para acesso e persistência dos objetos específicos na base de dados do sistema. Como cada módulo é independente, cada um pode ter um banco de dados diferente, e conseqüentemente é nessa camada que se fará a conexão como banco de dados.

Banco de Dados

Cada módulo terá seu próprio banco de dados e que deverá ser do tipo relacional, mas que deverá ter seu acesso restrito somente ao seu próprio módulo.