**Documento de Projeto de Software**

**Projeto:** Agenda Pessoal de Tarefas

**Versão:** 1.0

**Responsáveis:** Jerffeson Gomes e Rodolfo Medeiros

1. **Introdução**

Este documento apresenta o documento de projeto (design) do sistema de Agenda Pessoal de Tarefas. Essa atividade foi conduzida em refinamentos sucessivos, começando pelo projeto da arquitetura do sistema, passando ao detalhamento dos componentes da arquitetura, até chegar ao projeto detalhado das classes. Este documento está organizado da seguinte forma: a seção 2 apresenta a plataforma de software a ser utilizada na implementação do sistema; a seção 3 discute aspectos do projeto da arquitetura do sistema; as seções 4 e finalmente, a seção 5 apresentam os modelos relativos aos subsistemas identificados;

1. **Plataforma de Implementação**

O sistema em questão trata-se de um Sistema de Informação e apresenta as seguintes características:

* Envolve grande quantidade de dados e a sua gerência deve ser feita usando um banco de dados;
* Há uma grande quantidade de interfaces com o usuário;
* O sistema deverá ser multiplataforma (Ou seja, acessível por celular, computador, tablet);

Levando-se em consideração essas características, decidiu-se implementar o sistema para a Agenda Pessoal de Tarefas em uma plataforma Web, usando a linguagem de programação Java e o banco de dados relacional MySQL.

1. **Arquitetura de Software**

Como se pode perceber pela especificação de requisitos para o sistema em questão, não há grandes restrições de desempenho e disponibilidade, ainda que algumas restrições tenham sido explicitamente apontadas. Assim, levando-se em consideração os requisitos para o sistema proposto, foram considerados como os principais atributos de qualidade a serem incorporados ao sistema os seguintes, apresentados juntamente com as táticas a serem aplicadas:

* Usabilidade:
  + A interface de ser simples e intuitiva
  + Prover ao usuário, sempre que possível, a entrada de informações por meio de seleção ao invés da digitação de informações.
  + A interface deve esta desvinculada a qualquer função de acesso a base de dados ou processamento de informações;
* Manutenibilidade:
  + Coerência semântica: a organização do sistema deve se dar de modo que as responsabilidades em um módulo trabalhem em conjunto sem depender excessivamente de outros módulos;
  + Uso de Padrões de projeto que organizem sua estrutura lógica;
* Segurança:
  + Autenticação de usuários por meio de login e senha;
  + Com usuário autenticado ele poderá ter acesso a todas as funcionalidades da plataforma;

Ainda que os demais atributos de qualidade não tenham sido considerados como sendo condutores da arquitetura, algumas táticas foram aplicadas visando garantir o nível de atendimento requerido. A seguir, as táticas consideradas são listadas:

Portabilidade: Será suportado pelos navegadores Google Chrome versão 53 ou superior e Firefox versão 48 ou superior.

Tomando por base as características do sistema discutidas na seção 2 e os atributos de qualidade e táticas selecionadas para tratá-los apresentados anteriormente, decidiu-se adotar um estilo combinando camadas e partições.

Inicialmente, duas partições principais foram definidas, procurando-se preservar a divisão em subsistemas realizada na fase de análise. Cada uma dessas partições, por sua vez, está organizada em três camadas, a saber: Camadas de Interface com o Usuário (ciu), Lógica de Negócio (cln) e Gerência de Dados (cgd).

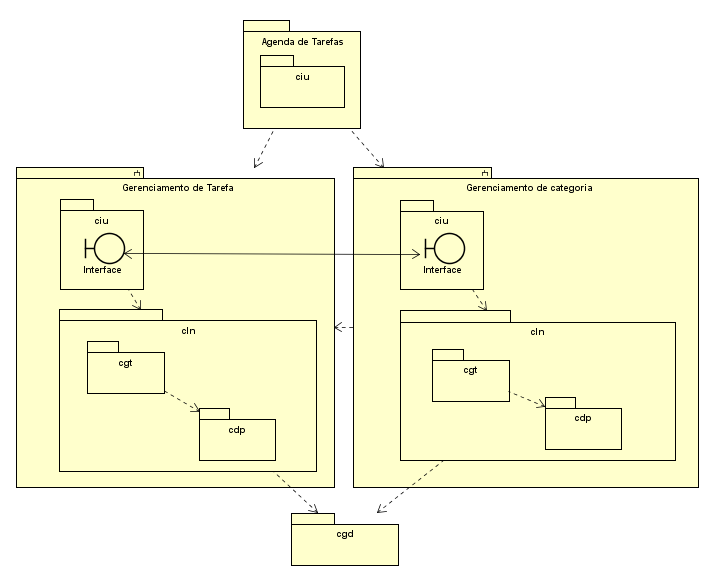


Figura 1 - Diagrama de Pacotes

1. **Subsistema Gerenciamento de Tarefas**

4.1 - Camada de Interface com o Usuário

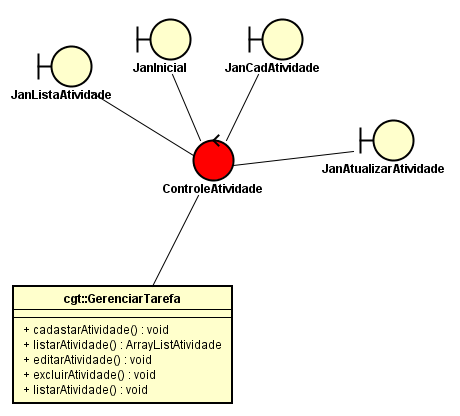


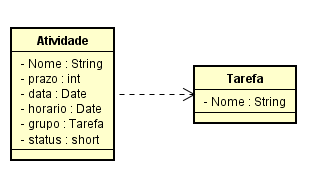
Figura 2 - Diagrama de Classes (View)

Para organizar construção da interface de usuário, será utilizado o padrão Model-View-Controler. Desse modo é possível dividir cada função da aplicação em níveis bem definidos, facilitando a construção e a manutenção.

4.2 - Camada Lógica de Controle

Gerencia toda lógica de negócio e domínio do problema. Essa camadas se subdivide em dois pacotes: Componente de Gerencia de Atividade (cgt) onde irá conter a classe de controle do projeto e o Componente de Domínio do Problema (cdp) que conterá as classes lógicas do sistema.

A figura 3 mostrar o diagrama de classes do pacote cdp:



1. **Subsistema Atendimento a Cliente**