

SETT - Sistema de Controle e Estimativa de Tempo de Tarefas
Documento de Arquitetura de Software

Versão 1.0

Histórico da Revisão

Data	Versão	Descrição	Autor
02/NOV/18	1.0	Criação do documento inicial	Jonatas

Índice Analítico

1.	Introdução	4
1.1	Finalidade	4
1.2	Escopo	4
1.3	Definições, Acrônimos e Abreviações	4
1.4	Referências	4
1.5	Visão Geral	4
2.	Representação Arquitetural	4
3.	Metas e Restrições da Arquitetura	4
4.	Visão de Casos de Uso	5
4.1	Realizações de Casos de Uso	5
5.	Visão da Implementação	5
8.1	Visão Geral	5
8.2	Camadas	6
6.	Tamanho e Desempenho	6
7.	Qualidade	6

Documento de Arquitetura de Software

1. Introdução

1.1 Finalidade

Este documento oferece uma visão geral arquitetural abrangente do sistema, usando diversas visões arquiteturais para representar diferentes aspectos do sistema. O objetivo deste documento é capturar e comunicar as decisões arquiteturais significativas que foram tomadas em relação ao sistema.

1.2 Escopo

Este documento é útil na construção do software em questão definindo padrões do projeto, disposição e organização, além das relações entre cada componente do todo, para implementação das regras de negócios.

1.3 Definições, Acrônimos e Abreviações

SETT - Sistema de Controle e Estimativa de Tempo de Tarefas;

1.4 Referências

- CSU1 - Cadastrar Tarefa.pfd - 19/09/2018 - <https://github.com/jonatasgev/projetoSoftwareUFG/blob/master/Documentos/CSU1%20-%20Cadastrar%20Tarefa.pdf>
- CSU2 - Calcular a estimativa de tempo necessária para realizar uma tarefa.pfd - 21/09/2018 - <https://github.com/jonatasgev/projetoSoftwareUFG/blob/master/Documentos/CSU2.pdf>
- CSU3 - Compartilhar com os outros usuário do sistema o tempo média em realizar tarefas.pfd - 21/09/2018 - <https://github.com/jonatasgev/projetoSoftwareUFG/blob/master/Documentos/CSU3.pdf>
- Documento Visão do Sistema.pfd - 21/09/2018 - <https://github.com/jonatasgev/projetoSoftwareUFG/blob/master/Documentos/Documento%20Visao%20Sistema.pdf>
- Requisitos.pfd - 21/09/2018 - <https://github.com/jonatasgev/projetoSoftwareUFG/blob/master/Documentos/Requisitos.pdf>

1.5 Visão Geral

O documento está subdividido em seções numeradas que podem ser encontradas no Índice. Nele você encontrará a representação arquitetural, metas e restrições da arquitetura, casos de uso, tamanho e desempenho, e qualidade.

2. Representação Arquitetural

A arquitetura atual é baseada e representada no modelo MVC. Faz-se necessárias as seguintes visões:

1. **Visão de casos de uso:** contém casos de uso e cenários que abrangem comportamentos significativos em termos de arquitetura, classes ou riscos técnicos
2. **Visão de Implementação:** contém uma visão geral do modelo de implementação e sua organização em termos de módulos em pacotes e camadas. A alocação de pacotes e classes nos pacotes e módulos da Visão de Implementação também é descrita.

3. Metas e Restrições da Arquitetura

- RF3: O software deve ter a opção de utilizar o histórico de tarefas de outros usuários para realizar o cálculo descrito no RF2.
 - Este requisito envolve privacidade e reutilização. Para sua implementação, de acordo com as leis nacionais e internacionais, os dados só podem ser reutilizados com o consentimento do usuário. Faz-se também necessário a implantação de uma consulta personalizada e, dependendo da quantidade de dados, a análise e implementação de soluções alternativas mais eficientes.

- RNF2: O sistema deve ser executado em ambiente web e, deve ter o seu tempo de resposta reduzido em 5 segundos.
 - Este requisito pode ser influenciado por N fatores além da arquitetura.
- RNF3: Todas as informações sobre valores e, resultados das execuções de cada ação do sistema que gerem valores devem ser precisos. O sistema deve possuir sistema de recuperação a falhas.
 - Este requisito envolve a implantação de um mecanismo de validação dos dados e das suas funções. Para tal, a implementação de testes automatizados e a utilização de um banco relacional são diferenciais, se não fundamentais, para atender e garantir este requisito.
- RNF4: O sistema deve estar disponível ao usuário mesmo que ele não possua conexão com a internet durante seu uso. As alterações devem ser sincronizadas assim que a conexão for restabelecida
 - Para atender este requisito, faz-se necessário a utilização de tecnologias específicas que permitem a utilização deste recurso.

4. Visão de Casos de Uso

- CSU1
- CSU2
- CSU3

4.1 Realizações de Casos de Uso

- CSU1:
 - O software recebe os dados da tarefa, valida os dados, salva no banco e, retorna a tarefa;
 - O software recebe os dados da tarefa, valida e reprova os dados e, retorna uma mensagem de erro;
- CSU2:
 - O software recebe os dados da tarefa, busca no histórico do usuário tarefas semelhantes, calcula o tempo estimado e, retorna o resultado do cálculo;
 - O software recebe os dados da tarefa, busca no histórico do usuário tarefas semelhantes, não encontra, retorna -1;
 - O software recebe os dados da tarefa, busca no histórico do comunidade tarefas semelhantes, calcula o tempo estimado e, retorna o resultado do cálculo;
- CSU3:
 - O software recebe as tarefas compartilhadas pela comunidade, utiliza como base para calcular a estimativa de tempo e, retorna o resultado do cálculo para o usuário

5. Visão da Implementação

5.1 Visão Geral

O software será dividido em 2 camadas:

- Front-end (ou interface):
 - Subdividido em outras 3 camadas: Model, View e Control.
- Back-end (ou serviços):
 - Subdividido em outras 2 camadas: Regras de negócio e Banco de dados.

5.2 Camadas

6. Tamanho e Desempenho

7. Qualidade

A arquitetura contribui para facilitar a implementação, manutenção, divisão de responsabilidades, etc.