**INDICE**

[**Documento Técnico: Arquitetura do Sistema de Gestão de Tarefas** 2](#_Toc188800563)

[**1. Sumário Executivo (Introdução)** 3](#_Toc188800564)

[**2. Arquitetura e Requisitos** 4](#_Toc188800565)

[2.1. Componentes da Arquitetura 4](#_Toc188800566)

[2.2. Interações entre Componentes 5](#_Toc188800567)

[2.3. Benefícios da Arquitetura 5](#_Toc188800568)

[2.4. Funcionalidades e Considerações 5](#_Toc188800569)

[**3. Metodologia e Abordagem** 6](#_Toc188800570)

[3.1. Constituição da Equipe 6](#_Toc188800571)

[**4. Plano do Projeto** 6](#_Toc188800572)

[**5. Orçamento e Estimativa de Custo** 7](#_Toc188800573)

[5.1. Estimativa de Tempo 7](#_Toc188800574)

[**6. Suposições e Dependências** 7](#_Toc188800575)

[**7. Conclusão** 8](#_Toc188800576)

**Documento Técnico: Arquitetura do Sistema de Gestão de Tarefas**

****

**Trabalho Elaborado por:**

Nicholas e Edna

1. Sumário Executivo (Introdução)

A crescente necessidade de ferramentas digitais para organização e produtividade motivou o desenvolvimento deste Sistema de Gestão de Tarefas. O objetivo é oferecer uma plataforma robusta, intuitiva e segura para a criação, organização e gerenciamento de tarefas, atendendo às necessidades de indivíduos e equipes. Este documento apresenta a arquitetura do sistema, as tecnologias empregadas e os detalhes do projeto, garantindo uma visão clara e humanizada sobre o funcionamento e o planejamento.

1. Arquitetura e Requisitos
   1. Componentes da Arquitetura

**Front-end:**

* + Desenvolvido com **React.js** para criar uma interface responsiva e amigável.
  + Uso de bibliotecas modernas para estilização (como Tailwind CSS).
  + Compatível com diferentes dispositivos (desktop, tablet e smartphones).

**Back-end:**

* + Construído com **Node.js** e **Express.js**, garantindo escalabilidade e rapidez.
  + Fornece a base para a lógica do sistema e interações seguras com o banco de dados.

**Banco de Dados:**

* + Utiliza **PostgreSQL** por sua robustez e facilidade em lidar com dados estruturados.
  + Armazena informações de tarefas, usuários e registros de atividades.

**API de Tarefas:**

* + Endpoints RESTful para criação, atualização, consulta e exclusão de tarefas.
  + Integrada com autenticação para garantir segurança nos acessos.

**Autenticação:**

* + Implementada com **JWT (JSON Web Tokens)**, garantindo sessões seguras.
  + Proteção por HTTPS para comunicações seguras entre cliente e servidor.
  1. Interações entre Componentes
  + O **front-end** consome os endpoints fornecidos pelo **back-end** para exibir e atualizar dados.
  + O **back-end** se comunica diretamente com o **banco de dados** para armazenar ou recuperar informações.
  + A **API de tarefas** serve como intermediária entre o usuário e a lógica do sistema.
  1. Benefícios da Arquitetura
  + **Escalabilidade:** Capacidade de adicionar novos recursos sem comprometer a performance.
  + **Manutenção Simplificada:** Uso de tecnologias modernas e bem documentadas.
  + **Segurança:** Proteção dos dados dos usuários através de autenticação e criptografia.
  1. Funcionalidades e Considerações
  + Criação de tarefas com campos como título, descrição, prazo e prioridade.
  + Sistema de notificações em tempo real para alertas de prazos.
  + Níveis de acesso (administrador e usuário comum) para garantir gestão eficaz.
  + Interface responsiva adaptada para diferentes tamanhos de tela.

1. Metodologia e Abordagem
   1. Constituição da Equipe
   * **Gerente de Projeto:** Responsável pelo planejamento e monitoramento do progresso.
   * **Desenvolvedores Front-end:** Criação da interface do usuário.
   * **Desenvolvedores Back-end:** Desenvolvimento da lógica do sistema e APIs.
   * **Especialista em Banco de Dados:** Estruturação e otimização do banco de dados.
   * **Designer de UX/UI:** Garantir que a interface seja intuitiva e atraente.
   * **Especialista em Testes:** Realiza testes para identificar e corrigir erros.
2. Plano do Projeto
   * **Planejamento:** Definição de objetivos e requisitos (1 mês).
   * **Desenvolvimento Inicial:** Criação da interface e APIs básicas (2 meses).
   * **Integração e Testes:** Conexão de todos os componentes e validação de funcionalidades (1 mês).
   * **Lançamento:** Publicação do sistema e suporte inicial (1 mês).
3. Orçamento e Estimativa de Custo
   * **Recursos Humanos:** Salários da equipe.
   * **Infraestrutura:** Hospedagem em servidores na nuvem.
   * **Tecnologias:** Licenças e ferramentas adicionais.
   1. Estimativa de Tempo
   * Total: 4 a 5 meses.
4. Suposições e Dependências
   * **Suposições:** A equipe terá experiência com as tecnologias selecionadas. Os requisitos iniciais não mudarão significativamente.
   * **Dependências:** Acesso à internet, ferramentas de desenvolvimento e recursos financeiros para hospedagem.
5. Conclusão

Este documento apresenta uma visão clara e abrangente da arquitetura e planejamento do Sistema de Gestão de Tarefas. Com uma equipe dedicada e uso de tecnologias modernas, o projeto busca oferecer soluções ágeis, seguras e intuitivas para os usuários. Este sistema não é apenas uma ferramenta, mas um aliado para aumentar a produtividade e melhorar a colaboração em diferentes contextos.