

Escopo do Sistema de Gerenciamento de Projetos

Definição do Escopo:

Este documento define o escopo do projeto para o desenvolvimento de um sistema de gerenciamento de projetos. O sistema visa oferecer uma plataforma completa e eficiente para que equipes possam colaborar, atribuir tarefas, acompanhar o progresso e gerenciar projetos de forma organizada.

Incluído no Projeto:

- **Funcionalidades de Criação e Gerenciamento de Projetos:**

- Permitir que gerentes criem novos projetos, definindo informações como título, descrição, data de início, data de término e status.
- Permitir que gerentes atribuam membros da equipe a projetos, com diferentes níveis de acesso e responsabilidades.
- Permitir que gerentes definam etapas, prazos e metas para cada projeto.
- Permitir que gerentes visualizem o progresso geral de cada projeto.

- **Funcionalidades de Tarefas:**

- Permitir que gerentes da equipe criem, editem, excluam e atribuam tarefas dentro de cada projeto.
- Permitir que gerentes da equipe definam prazos, descrição detalhada e status para cada tarefa.

- **Gerenciamento de Membros:**

- Permitir que gerentes adicionem e removam membros da equipe de um projeto.
- Permitir que membros da equipe atualizem seus perfis e informações de contato.
- Permitir que membros da equipe visualizem a lista de projetos que estão participando.

- **Relatórios:**

- Gerar relatórios sobre o progresso dos projetos, incluindo status das tarefas, tempo gasto, atrasos e outras métricas relevantes.
- Permitir que gerentes personalizem os relatórios de acordo com suas necessidades.

Tecnologias Utilizadas:

- **Laravel Framework:** Framework de desenvolvimento web para a construção da estrutura do sistema.
- **PostgreSQL:** Banco de dados relacional para armazenamento de informações sobre projetos, tarefas, membros e outras informações relevantes.
- **Pusher:** Ferramenta para notificações em tempo real para garantir a atualização instantânea de informações relevantes para os membros da equipe.

Revisão dos Objetivos

Objetivos SMART:

- **Específicos:**

- Criar um sistema que permita aos gerentes criar e gerenciar projetos de forma eficiente.
- Permitir que membros da equipe colaborem em tarefas, acompanhem o progresso e se comuniquem de forma organizada.

- **Mensuráveis:**

- O sistema deve atender a um mínimo de 100 projetos simultâneos e 200 usuários ativos.
- O sistema deve garantir a atualização em tempo real de informações para todos os membros da equipe.

- **Atingíveis:**

- Alocar uma equipe de desenvolvedores com experiência em Laravel e PostgreSQL, com marcos semanais para revisão de progresso e

integração dos módulos.

- **Relevantes:**

- Otimizar a gestão de projetos, reduzir o tempo de conclusão de tarefas e aumentar a produtividade das equipes.
- Melhorar a comunicação e colaboração entre membros da equipe.

- **Temporais:**

- Concluir a implementação e testes da versão beta do sistema até o final da 10ª semana do projeto.

Gestão de Riscos

3.1 Identificação de Riscos Potenciais:

- **Riscos de Comunicação e Colaboração:** Dificuldades em garantir uma comunicação eficiente entre membros da equipe, levando a atrasos e falhas na execução das tarefas.
- **Problemas com o Gerenciamento de Prazos:** Dificuldade em acompanhar os prazos das tarefas e projetos, levando a atrasos na entrega final.
- **Desafios na Geração de Relatórios Precisos:** Dificuldade em gerar relatórios completos e precisos sobre o progresso dos projetos.

3.2 Estratégias de Mitigação:

- **Riscos de Comunicação e Colaboração:** Implementar um sistema de chat integrado, fóruns de discussão e ferramentas de comunicação visual para facilitar a troca de informações entre membros da equipe.
- **Problemas com o Gerenciamento de Prazos:** Integrar um sistema de gestão de tempo e um calendário compartilhado para acompanhar os prazos das tarefas e projetos, com alertas e notificações para lembretes.
- **Desafios na Geração de Relatórios Precisos:** Desenvolver relatórios personalizados e flexíveis, com opções de filtros e visualização de dados para gerar relatórios precisos e relevantes.

Cronograma

Dia 1-2: Análise de Requisitos e Definição do Escopo, Arquitetura do Sistema e Design da Interface.

Dia 3-4: Implementação do CRUD para Projetos e Tarefas, incluindo criação, leitura, atualização e exclusão de projetos e tarefas.

Dia 5-6: Implementação do Sistema de Autenticação e Gerenciamento de Membros, incluindo login, registro e gerenciamento de perfis.

Dia 7-8: Implementação do Sistema de Notificações em tempo real utilizando Pusher, garantindo a atualização instantânea de informações relevantes para os membros da equipe.

dia 9-10: Implementação das funcionalidades de geração de relatórios, com opções de personalização e filtros.

Dia 11-12: Testes de performance, usabilidade e segurança, correção de bugs e documentação do sistema.

Diagramas

Diagrama de Classe:

- **Projetos:** Atributos (título, descrição, data de início, data de término, status, gerentes, membros, tarefas) e métodos (criar, editar, excluir, adicionar membros, adicionar tarefas, gerar relatórios).
- **Tarefas:** Atributos (título, descrição, prazo, status, membro responsável, arquivos anexados, comentários) e métodos (criar, editar, excluir, adicionar comentários, anexar arquivos).
- **Equipes:** Atributos (nome, membros) e métodos (adicionar membros, remover membros).
- **Membros:** Atributos (nome, email, senha, projetos) e métodos (acessar projetos, visualizar tarefas, atualizar status de tarefas, comentar em tarefas).

Diagrama de Uso:

- **Gerente:** Cria projetos, atribui tarefas, gerencia membros, monitora o progresso, gera relatórios.
- **Membro:** Acessa projetos, visualiza tarefas, atualiza status de tarefas, se comunica com outros membros, anexa arquivos.

Diagrama de Fluxo:

- Login → Painel do Projeto → Gestão de Tarefas → Relatórios de Progresso → Logout.