

Sistema de Gerenciamento do Programa de Integridade

Dandara Gouveia Menezes Calumby

Definição dos Processos, Métodos e Ferramentas:

Para o desenvolvimento do projeto, será utilizada uma abordagem ágil baseada em Scrum, com sprints semanais. As ferramentas principais são:

- GitHub: versionamento de código e colaboração.
- Trello: gerenciamento de tarefas e organização dos sprints.
- Figma: prototipação e criação de interfaces.
- VS Code: ambiente de desenvolvimento.

Domínio do Software:

O sistema permitirá o gerenciamento de programas de integridade com base em pilares, ações estratégicas, prazos, alertas e dashboards informativos.

Artefatos gerados:

- Para desenvolvedores: Diagrama UML, código-fonte, documentação técnica, protótipos.
- Para usuários: Funcionalidades como cadastro de ações, alertas, relatórios e dashboards.

Requisitos Funcionais:

Cadastro de Pilares

- Permitir o cadastro de pilares que serão base para os Programas de Integridade.

Cadastro de Ações por Pilares

- Cadastro de ações estratégicas categorizadas por pilar.
- Associação de responsável pela ação (Analista, Coordenador ou Gestor).
- Cada ação deve conter:
 - Etapas com prazos.

- Método utilizado.
- Prazo para conclusão.
- Situação (Cadastrada, em avaliação, em andamento, concluída).
- Histórico de atualizações com data e autor.

Controle de Acesso e Permissões

- Analista:
 - Pode cadastrar e editar ações.
 - Não pode finalizar ações.
- Coordenador:
 - Pode cadastrar, aprovar, editar e finalizar ações.
- Gestor:
 - Pode consultar todas as ações.
 - Pode gerar relatórios e acessar dashboards.

Relatórios

- Relatórios consolidados por pilar, prazo e responsável.
- Exportação em PDF, Excel ou Word.

Alertas Automáticos

- Alerta de Prazo:
 - Notificar responsável 7 dias antes do vencimento da etapa ou ação.
- Alerta de Atraso:
 - Notificar responsável e coordenador se a etapa ou ação estiver vencida.

Dashboard Resumido

- Número de ações: concluídas, em andamento e atrasadas.
- Progresso por pilar.
- Prazos críticos.

Requisitos Não Funcionais

- Desempenho: O sistema deve ser responsivo e carregar telas principais.
- Segurança:
 - Autenticação por login e senha com perfis de acesso.

- Controle de permissões baseado no papel do usuário.
- Usabilidade: Interface intuitiva, com experiência consistente para desktop
 - Portabilidade: Web responsivo.
 - Auditabilidade: Histórico de alterações por usuário e data.
 - Disponibilidade: Sistema com uptime.

Diagrama de Classes UML

