
SITE MINISTÉRIO DA TECNOLOGIA

SITE MINISTÉRIO DA TECNOLOGIA
Arquitetura de Software
Versão <1.0>

Histórico da Revisão

Data	Versão	Descrição	Autor
05/05/25	1.0	Elaboração do documento	Roberto Qiming Li

Sumário

1.	4	
1.1	4	
1.2	4	
1.3	5	
1.4	5	
2.	10	
3.	11	
4.	Erro! Indicador não definido.	
5.	12	
5.1	12	
5.2	Erro! Indicador não definido.	
6.	14	
7.	14	
8.	Tamanho	e Desempenho
	9
9.	Qualidade	
	9

1. Introdução

Este documento tem como objetivo descrever a proposta, os requisitos e a justificativa das tecnologias escolhidas para o desenvolvimento do site do Ministério da Tecnologia. O site servirá como um lobby digital para que pessoas possam apresentar seus projetos e receber investimentos estatais e privados.

1.1 Finalidade

Este documento oferece uma visão arquitetural geral do sistema, usando diversas visões arquiteturais para representar diferentes aspectos do mesmo. O objetivo deste documento é capturar e comunicar as decisões arquiteturais significativas que foram tomadas no decorrer do processo de desenvolvimento.

1.2 Escopo

Este documento auxilia os envolvidos no projeto a captar aspectos arquiteturais do sistema que são necessários para o desenvolvimento de uma solução que atenda às necessidades dos usuários finais. Além de auxiliar no entendimento do sistema por novos membros da equipe.

1.3 Descrição do Problema

Este documento auxilia os envolvidos no projeto a captar aspectos arquiteturais do sistema que são necessários para o desenvolvimento de uma solução que atenda às necessidades dos usuários finais. Além de auxiliar no entendimento do sistema por novos membros da equipe.

Problema	Dificuldades para divulgar projetos de forma acessível e atrativa a potenciais investidores, devido à ausência de uma plataforma centralizada e confiável.
Afeta	Empreendedores que buscam apoio financeiro e visibilidade, investidores que querem encontrar boas oportunidades, e órgãos públicos interessados em incentivar a inovação.
Impacto	A inovação nacional é prejudicada, bons projetos deixam de ser realizados por falta de visibilidade e investimento, e há perda de oportunidades de colaboração entre setores público e privado.
Solução	Desenvolver um site que funcione como um lobby digital, permitindo o cadastro, divulgação e avaliação de projetos, além de facilitar a comunicação entre empreendedores e investidores.

1.4 Definições, Acrônimos e Abreviações

MVC – Padrão de arquitetura de software onde M significa modelo sendo responsável pela parte de regras de negócio, V a visualização responsável pela parte de interfaces e C a parte de controle dos dados.

1.5 Visão Geral

São apresentados ainda neste documento diferentes visões arquiteturais de como o sistema deve se comportar em diferentes processos, como deve ser implantado é implementado e restrições de desempenho e qualidade.

1.5.1 Tabela de Funcionalidades

Funcionalidades
<ul style="list-style-type: none">• Cadastro e login de usuário• Cadastro, edição e exclusão de projetos com imagens e descrições• Busca e filtro de projetos por área, status, palavra-chave• Comentários e avaliações em projetos• Chat entre investidores e empreendedores• Recebimento de notificações sobre interações• Painel administrativo para gestão de usuários e projetos• Denúncia e moderação de conteúdo• Relatórios e métricas administrativas

2. Levantamento e Análise de Requisitos

2.1 Entendimento do Problema

Foi identificado que muitos cidadãos, pesquisadores e pequenas empresas possuem projetos inovadores, mas enfrentam dificuldades para divulgá-los, conectá-los com

investidores ou obter apoio governamental. Atualmente, não existe uma plataforma centralizada e acessível para facilitar esse tipo de interação com o Ministério da Tecnologia.

- Cenários de Uso Identificados:
 - **Cidadão inovador:** deseja cadastrar um projeto de tecnologia sustentável e busca visibilidade e apoio financeiro.
 - **Startup iniciante:** precisa de investimento para prototipar uma solução de impacto social e não conhece os canais disponíveis.
 - **Investidor privado:** procura projetos tecnológicos promissores com potencial de retorno e impacto positivo.
 - **Gestor público:** deseja acompanhar projetos submetidos para avaliar seu alinhamento com políticas públicas de inovação.
- Dores Principais dos Cidadãos:
 - Falta de um canal oficial e confiável para apresentar projetos.
 - Dificuldade em conseguir apoio financeiro e técnico.
 - Pouca visibilidade frente a investidores e instituições.
 - Burocracia e falta de transparência nos processos de seleção e apoio governamental.

2.2 Requisitos Funcionais e Não-Funcionais

- **Funcionais:**
 - Cadastro de usuários:
 - Permitir o registro e login de três perfis distintos: empreendedores, investidores e administradores, com permissões específicas para cada tipo.
 - Sistema de autenticação e recuperação de senha:
 - Garantir acesso seguro por meio de login com e-mail e senha, além de funcionalidade para redefinir senha em caso de esquecimento.
 - Validação de perfil institucional:
 - Oferecer a opção de validar perfis mediante envio de documentação, aumentando a confiança entre os usuários.
 - Cadastro e upload de projetos:
 - Empreendedores poderão cadastrar projetos com título, descrição, imagens e informações complementares.

- Sistema de busca e filtros:
 - Permitir a localização de projetos por nome, área, localidade, tipo de tecnologia, avaliação, entre outros.
- Sistema de favoritos:
 - Investidores e usuários autenticados poderão salvar projetos de interesse em uma lista pessoal.
- Avaliação e comentários em projetos:
 - Habilitar feedbacks construtivos e avaliações públicas nos projetos para promover interação e visibilidade.
- Notificações e comunicação interna:
 - Permitir troca de mensagens e envio de notificações entre investidores e empreendedores interessados.
- Histórico de interações:
 - Registrar atividades do usuário como projetos visualizados, favoritados, comentados ou avaliados.
- Sistema de denúncias e moderação:
 - Permitir a denúncia de conteúdos inadequados e encaminhar para análise do administrador.
- Dashboard administrativo:
 - Disponibilizar ferramentas para que administradores possam gerenciar usuários, projetos, denúncias e conteúdos.
- Relatórios e estatísticas:
 - Gerar dados sobre uso da plataforma, número de acessos, interações, tipos de projetos mais populares, etc.
- **Não-Funcionais:**
 - Interface responsiva e acessível:
 - O sistema deve adaptar-se a diferentes tamanhos de tela (computadores, tablets e smartphones) e seguir boas práticas de acessibilidade, garantindo uso por pessoas com deficiências visuais ou motoras.

- Segurança reforçada:
 - Implementar proteção contra ataques como CSRF, XSS e SQL Injection, além de armazenar senhas de forma criptografada (por exemplo, usando bcrypt ou Argon2).
- Banco de dados eficiente e escalável:
 - Utilizar um banco de dados estruturado e otimizado para lidar com grandes volumes de dados e permitir fácil expansão sem perda de desempenho.
- Baixo tempo de resposta das requisições:
 - O sistema deve responder rapidamente às interações do usuário, com tempo de carregamento reduzido e otimizações no back-end e front-end.
- Manutenibilidade e modularidade do código:
 - O código deve ser escrito de forma organizada e modular, facilitando futuras alterações, correções ou adição de funcionalidades por outros desenvolvedores.

2. Resumo dos Usuários

Nome	Responsabilidade	Descrição
Cliente	Explorar, interagir, avaliar e apoiar projetos.	Usuário interessado em encontrar projetos promissores para investimento, podendo comentar, avaliar e contatar empreendedores.
Empreendedor	Cadastrar e divulgar projetos para obter apoio.	Usuário que cria perfis de projetos com informações, imagens e metas, buscando visibilidade e investimento.
Administrador	Gerenciar o sistema e moderar conteúdos.	Responsável por manter o funcionamento da plataforma, controlar usuários e projetos, além de garantir a segurança e integridade dos dados.

3. Necessidades do Cliente

Identificador	Necessidade	Prioridade	Solução Atual
Cliente			

4. Representação Arquitetural

- Visão de caso de uso apresenta as funcionalidades arquiteturais importantes e os usuários do sistema.
- Visão lógica o Descreve as classes e sua organização e apresenta o padrão de arquitetura que deverá ser utilizado para o desenvolvimento do sistema.
- Visão de processos mostra o padrão de comportamento do sistema diante de diferentes ações do usuário.
- Visão de implantação o Descreve a estrutura do ambiente onde o software será instalado.

- Visão de implementação ilustra a distribuição do processamento em um conjunto de nós do sistema, incluindo a distribuição física dos processos e threads.

5. Metas e Restrições da Arquitetura

Existem algumas restrições de requisito e de sistema principais que têm uma relação significativa com a arquitetura, sendo elas:

- Utilização do paradigma Orientado a Objetos para o desenvolvimento;
- Estrutura MVC;
- Linguagem de programação Python;
- Framework *xxx*
- O sistema em questão deverá ser multiplataforma; • Bancos de dados *xxxx*.

6. Arquitetura do Sistema e Design de Software

6.1 Visão Geral da Arquitetura

- **Padrão:** Monolítico modular, migrável para micro-serviços na versão 2.
- **Tecnologias:**
 - **Back-end:**
 - **Flask ou Django:** Escolha baseada na menor quantidade de usuários para evitar sobrecarga de frameworks populares e garantir mais segurança e exclusividade.
 - **SQLite ou PostgreSQL:** SQLite para desenvolvimento e PostgreSQL para produção, garantindo robustez e escalabilidade.
 - **Front-end:**
 - **HTML, CSS e JavaScript:** Estrutura base da aplicação para garantir compatibilidade universal.
 - **Tailwind CSS:** Framework CSS que proporciona estilização rápida e otimizada.

- **React.js:** Biblioteca moderna para desenvolvimento dinâmico e reativo da interface.

- **Segurança:**

- **bcrypt ou passlib:** Criptografia segura para armazenamento de senhas.
- **CSRF Protection:** Proteção contra ataques de falsificação de solicitação entre sites.

6.2 Design de Software (Diagramas)

- **UML de Classes:** descreve entidades principais e relacionamentos.
- **Diagrama de Casos de uso:** mostra como os usuários interagem com o sistema.
- **Modelos Conceitual, Lógico do Banco de Dados**

Diagramas a baixo:

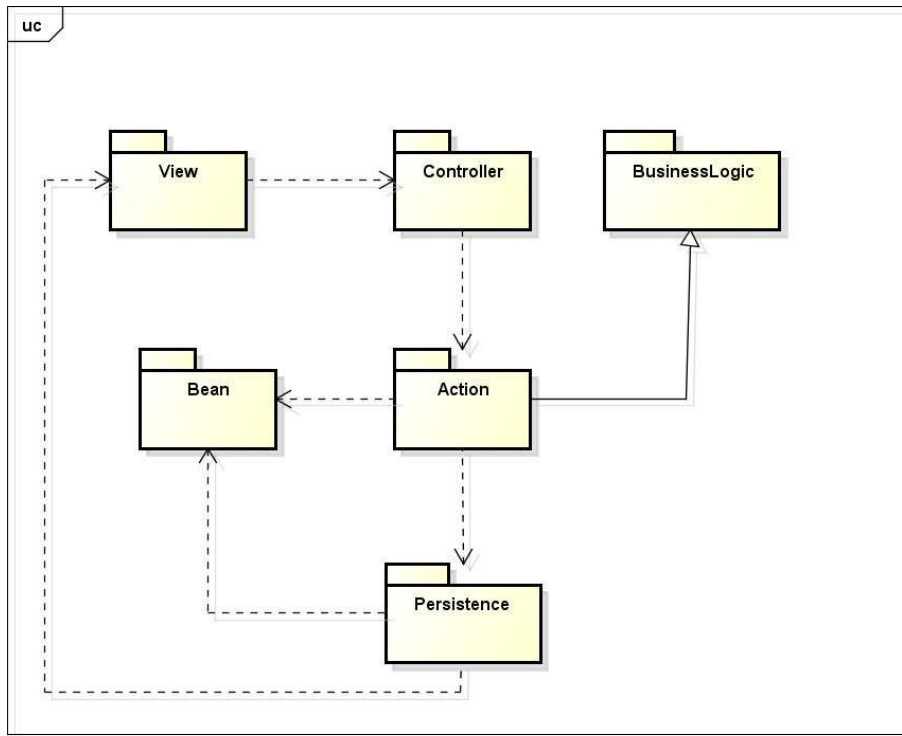
Diagrama: [Diagramas - Google Drive](#)

Comentado [A1]:

6. Visão Lógica

6.1 Visão Geral

A visão lógica define a estrutura da arquitetura. Abaixo será especificado o padrão utilizado para o desenvolvimento do sistema, no caso, MVC.



powered by astah®

- *View*: Componente que contém as visões do projeto, ou seja, as interfaces, formulários e etc.;
- *Controller*: Componente que recebe as informações e requisições do pacote *view* e os despacha para devida classe de controle, o *controller* se comunica com classes *BusinessLogic* que contém as regras de negócio do sistema;
- *Action - Model*: pacote que recebe as informações e requisições do pacote *controller* e os atribui às respectivas classes do pacote *bean* e do pacote *persistence*, utilizando dados de seus modelos;
- *Bean*: pacote que recebe e armazena as informações referentes a seus respectivos objetos;

- *Persistence*: pacote que recebe as requisições e realiza as operações relacionadas ao Banco de Dados, utilizando dados provindos do pacote *bean*. O pacote *persistence* também pode enviar dados para o pacote *view*.

7. Visão de Implantação

O sistema será implantado quando for validado entre os *stakeholders*.

8. Visão da Implementação

O sistema será implementado utilizando conceitos de Programação Orientada a Objetos através do framework *XXXX*, estrutura MVC, , *Web Container JBOSS*, Linguagem de Programação P e Bancos de Dados *xxxx*.

8. Tamanho e Desempenho

A aplicação deve rodar através de um site próprio na versão web para os *xxxx* e por meio de dispositivos móveis na versão mobile que será para o uso do usuário *xxxxx*.

9. Qualidade

O padrão de arquitetura adotado no projeto tem como finalidade garantir uma melhor organização do código-fonte, o que auxilia na manutenibilidade do software, bem como a portabilidade do mesmo.

10 Git hub ;

<https://github.com/Samucau1/Invest-Gov.git>