

MeuScrum	
Architecture Notebook	Date: 04/10/2025

# MeuScrum

## Descrição de Arquitetura de Software

### 1. Propósito

Este documento descreve a filosofia, decisões, restrições, justificativas e mecanismos arquiteturais do sistema de software para **gestão de projetos ágeis com base no framework Scrum**, permitindo o gerenciamento de histórias de usuário, backlogs e planos de sprint, por meio de uma aplicação web acessível a partir de qualquer estação de trabalho disponibilizada de forma Web.

### 2. Filosofia e Metas Arquiteturais

O sistema é projetado para oferecer uma **plataforma colaborativa, intuitiva e robusta**, que suporte múltiplos projetos simultâneos, cada um com papéis e permissões específicas. O foco está na **escalabilidade, facilidade de manutenção e usabilidade** para usuários com diferentes níveis de experiência.

#### Metas arquiteturais:

- Ser **modular e manutenível** para facilitar futuras evoluções.
- **Alta disponibilidade e concorrência** para múltiplos usuários autenticados.
- **Segurança** na autenticação e controle de permissões.
- **Interface responsiva e intuitiva**, adaptável a desktop e dispositivos móveis.
- **Persistência confiável**, incluindo suporte a armazenamento de mídias (PDFs, vídeos, imagens).
- **Tolerância a falhas de conexão**, com salvamento local temporário (offline-first).
- **Aderência ao framework Scrum**, refletindo suas entidades e papéis principais.

### 3. Suposições e Dependências

#### Suposições:

- O sistema será utilizado em **ambientes com conexão à internet**, mas precisa lidar com instabilidades temporárias.
- Os usuários podem ter **baixo conhecimento técnico**, portanto a interface deve ser **autoexplicativa e assistida**.
- O acesso ocorrerá principalmente via **navegador web moderno** (Chrome, Edge, Firefox).

#### Dependências:

- **Framework web**: Django (python).
- **Banco de dados relacional**: PostgreSQL.
- **Serviço de autenticação e autorização**: JWT.
- **Serviços de armazenamento de arquivos**: File System local ou serviço em nuvem.

### 4. Requisitos Arquiteturalmente Significativos

[Insira uma referência ou link para os requisitos que devem ser implementados para realizar a arquitetura.]

MeuScrum	
Architecture Notebook	Date: 04/10/2025

## 5. Decisões, Restrições e Justificativas

### Adoção de arquitetura multicamadas (frontend, backend e banco de dados)

Facilita manutenção e evolução do sistema.

### Utilização de autenticação baseada em tokens (JWT)

Garante segurança e escalabilidade em ambiente web.

### Persistência em banco de dados relacional

Necessidade de manter relacionamentos entre usuários, projetos e papéis.

### Interface web responsiva (design adaptativo)

Permite acesso em diferentes dispositivos.

### Implementação de cache local e sincronização

Evita perda de dados em ambientes com conexão instável.

### Controle de acesso baseado em papéis

Garante integridade e segurança nas ações dos usuários.

### Adoção de arquitetura RESTful para a API

Padrão amplamente compatível e escalável para serviços web.

## 6. Mecanismos Arquiteturais

### Autenticação e Autorização (JWT)

Garante acesso seguro e controle de permissões baseado em papéis.

### Persistência de Dados (ORM)

Mapeia entidades do sistema para tabelas do banco de dados PostgreSQL.

### Gerenciamento de Sessão Offline

Armazena temporariamente dados em cache/localStorage e sincroniza quando a conexão é restabelecida.

### Interface Responsiva (HTML5 + CSS3 + JS)

Adapta o layout automaticamente conforme o dispositivo.

### Serviço de Upload de Arquivos

Permite armazenar PDFs, imagens e vídeos em nuvem.

### Notificações e Tooltips Interativos

Ajudam o usuário a compreender funcionalidades do sistema.

### Controle de Acesso (RBAC)

Define permissões de ação conforme o papel do usuário no projeto.

### API RESTful

Intermedia comunicação entre o front-end e o back-end.

### Logger e Monitoramento

Registra eventos e falhas para auditoria e manutenção.

## 7. Abstrações Principais

- **Usuário:** entidade que representa qualquer membro autenticado do sistema
- **Projeto:** contexto onde os artefatos e papéis são definidos
- **Papel (Role):** define as permissões do usuário (PO, SM, Developer, Admin)
- **História de Usuário:** requisito funcional descrito em linguagem natural
- **Backlog de Produto:** conjunto de histórias priorizadas
- **Backlog de Sprint:** conjunto de tarefas planejadas para uma iteração

MeuScrum	
Architecture Notebook	Date: 04/10/2025

- **Plano de Sprint:** contém objetivo, tarefas, equipe e incrementos
- **Artefato de Mídia:** arquivo anexo (vídeo, imagem, PDF).

## 8. Camadas ou Framework Arquitetural

Framework Arquitetural adotado: 3-Tier

- **Apresentação (Frontend)** — interface gráfica e interação com o usuário.
- **Lógica de Negócio (Backend)** — regras do Scrum, papéis e permissões.
- **Persistência (Database)** — armazenamento relacional e de mídia.

## 9. Visualização Arquitetural

Visualização adotada: Histórias de Usuário

- Mostra o que o usuário deseja para o sistema, principais funcionalidades, evitar alguns erros e prazos de responsividade do sistema.