

Documento de Visão e Escopo

Projeto de Engenharia de Software

Preparado por: [Arthur Luiz Lima de Araújo]

Versão: 2.0
16 de novembro de 2025

Sumário

1 Controle do Documento	3
2 Introdução e Propósito	3
3 Declaração do Problema e Oportunidade	3
3.1 Problema a ser Resolvido (C1)	3
3.2 Oportunidade de Negócio	4
4 Visão e Posicionamento do Produto	4
4.1 Declaração de Visão	4
4.2 Posicionamento de Mercado (C2)	5
5 Stakeholders, Perfis de Usuários e Ambiente	5
5.1 Perfil: Product Owner (PO)	5
5.2 Perfil: Scrum Master (SM)	6
5.3 Perfil: Developer (Membro da Equipe)	6
6 Escopo do Produto	6
6.1 Necessidades Atendidas (C5)	6
6.2 Funcionalidades Principais (Features) (C6)	7
6.3 Fora do Escopo (Limites)	7
7 Requisitos Não Funcionais e Restrições	8
7.1 Requisitos Não Funcionais Resumidos (C7)	8
7.2 Restrições (Constraints)	9
8 Elementos da Solução Proposta (C8)	9
9 Premissas e Dependências	10
9.1 Premissas	10
9.2 Dependências	10

1. Controle do Documento

Nome do Projeto:	Sistema de Gestão de Artefatos Scrum (SGAS)
Patrocinador (Product Owner):	Arthur Luiz Lima de Araújo
Preparado por (Equipe):	Arthur Luiz Lima de Araújo, Arthur da Silva Pereira Bispo, João Carlos Goncalves de Oliveira Filho, Ana Luisa de Souza Paraguassu, Tauã Valentin de A. M. Frade
Data de Criação:	16 de novembro de 2025

Histórico de Versões

Versão	Data	Autor	Descrição da Mudança
1.0	16 de novembro de 2025	Arthur Luiz Lima de Araújo	Versão inicial do documento de visão e escopo.
2.0	16 de novembro de 2025	Arthur Luiz Lima de Araújo	Realizado alterações para melhor conformidade.

2. Introdução e Propósito

Este documento define a visão de alto nível e o escopo do produto **Sistema de Gestão de Artefatos Scrum**. Ele serve como a fonte de verdade para alinhar a equipe de desenvolvimento e os stakeholders internos sobre o problema de mercado que estamos resolvendo, os objetivos de negócio e os limites do produto a ser desenvolvido.

O propósito central do **Sistema de Gestão de Artefatos Scrum** é prover uma ferramenta de software leve, acessível e focada para equipes de desenvolvimento que adotam o framework Scrum. O sistema visa simplificar a gestão de artefatos ágeis, permitindo que as equipes gerenciem seus requisitos (histórias de usuário) e seus ritos (sprints, backlogs) de forma eficiente e sem a sobrecarga de ferramentas mais complexas.

3. Declaração do Problema e Oportunidade

3.1. Problema a ser Resolvido (C1)

Esta seção descreve o problema que o projeto visa solucionar.

- O problema de:** Equipes de desenvolvimento, especialmente as de pequeno e médio porte ou aquelas em transição para métodos ágeis, enfrentam grande dificuldade para gerenciar seus projetos Scrum de forma eficaz. As ferramentas de gestão de projetos líderes de mercado (como Jira ou Azure DevOps) são frequentemente percebidas como excessivamente complexas, caras e com uma curva de aprendizado íngreme, não muito

úteis para integração rápida, dado que o custo e tempo de adaptação aos processos organizacionais é considerável.

- **Que afeta:**
 - Equipes Scrum iniciantes que se sentem "sobrecarregadas" com funcionalidades que não utilizam.
 - Pequenas empresas e startups que não possuem orçamento para licenças caras.
 - Product Owners e Scrum Masters que gastam mais tempo configurando a ferramenta do que gerenciando o produto e o processo.
- **O impacto do qual é:**
 - **Baixa Adoção:** As equipes abandonam a ferramenta e voltam para planilhas ou quadros físicos, perdendo rastreabilidade e métricas.
 - **Custo Elevado:** Desperdício de orçamento em licenças de software com recursos subutilizados.
 - **Ineficiência:** Perda de tempo em configuração e gerenciamento da ferramenta, em vez de focar na entrega de valor.
- **Uma solução bem-sucedida seria:** Uma plataforma de software leve, intuitiva e acessível, focada exclusivamente nos artefatos e ritos essenciais do Scrum, permitindo que as equipes se organizem rapidamente e gerenciem seus projetos com o mínimo de sobrecarga (overhead). A ferramenta deve focar apenas no desenvolvimento Scrum, sendo especializada no mesmo, sem providenciar suporte para funcionalidades a serem usadas por outros processos de desenvolvimento.

3.2. Oportunidade de Negócio

Existe uma oportunidade de mercado clara para um produto "de prateleira" que atenda ao nicho de equipes que buscam uma solução "Scrum-essencial". Ao focar na simplicidade e no fluxo de trabalho central do Scrum (gestão de Histórias de Usuário, Backlogs e Sprints), o **SGAS** pode capturar uma parte do mercado de pequenas e médias empresas, bem como de equipes acadêmicas, que atualmente são mal servidas pelas soluções empresariais dominantes.

4. Visão e Posicionamento do Produto

4.1. Declaração de Visão

Esta é a "Declaração de Visão" (Elevator Pitch) do produto.

- **Para:** Equipes de desenvolvimento de software (pequenas, médias, startups e acadêmicas).
- **Que (necessidade/oportunidade):** Necessitam gerenciar seus projetos ágeis de forma eficiente, mas consideram as ferramentas de mercado atuais muito complexas, caras e com alta curva de aprendizado.

- **O (nome do produto) é um(a):** Plataforma de software (web) para gestão de projetos.
- **Que (principal benefício/valor):** Foca exclusivamente nos artefatos e ritos essenciais do framework Scrum (Histórias de Usuário, Backlogs, Sprints), permitindo uma adoção rápida e gerenciamento sem sobrecarga.
- **Diferente de (concorrentes/alternativas):** Ferramentas "enterprise" complexas (como Jira, Azure DevOps) e ferramentas genéricas de quadros (como Trello ou planilhas).
- **Nosso produto (diferencial chave):** Oferece uma experiência de usuário "pura" de Scrum. Em vez de ser uma ferramenta genérica que *pode* ser configurada para o Scrum, o SGAS é uma ferramenta *especialista*, projetada apenas para o Scrum, garantindo simplicidade, baixo custo e foco total no processo.

4.2. Posicionamento de Mercado (C2)

O SGAS se posicionará no mercado como a ferramenta "**Scrum Essencialista**".

Nosso nicho de mercado são as equipes que valorizam a simplicidade e a adesão estrita ao framework Scrum, em detrimento de um conjunto exaustivo de funcionalidades personalizáveis.

Nosso posicionamento em relação às alternativas é:

- **Contra Ferramentas Enterprise (Jira, etc.):** O SGAS é a alternativa leve, acessível e de adoção instantânea. Onde eles oferecem complexidade e personalização total (que geram alto custo de setup e licença), nós oferecemos simplicidade e foco.
- **Contra Ferramentas Genéricas (Trello, Planilhas):** O SGAS é a alternativa estruturada. Onde eles oferecem flexibilidade total (mas sem suporte nativo a Sprints, Backlogs ou Histórias de Usuário), nós oferecemos a estrutura fundamental que o Scrum exige, garantindo que o processo seja seguido corretamente e sem desvios.

O SGAS pretende ser a solução preferencial para equipes que estão começando com Scrum ou para equipes experientes que desejam uma ferramenta "limpa" que apenas facilite o processo, sem ditar ou complicar o fluxo de trabalho.

5. Stakeholders, Perfis de Usuários e Ambiente

Esta seção descreve as partes interessadas (stakeholders), suas responsabilidades no sistema (Critério C3) e o ambiente de trabalho onde o utilizarão (Critério C4).

Conforme os requisitos (Item 2), os usuários devem criar uma conta, ser autenticados e, uma vez autenticados, assumem um dos três papéis em um projeto: Product Owner, Scrum Master ou Developer.

5.1. Perfil: Product Owner (PO)

- **Descrição (Stakeholder):** A parte interessada que representa a visão do negócio/cliente e é responsável por maximizar o valor do produto.

- **Responsabilidades (C3):** Foco total no "O Quê?". Utilizar o SGAS para sua responsabilidade primária: **criar, ler, atualizar e excluir (CRUD)** as Histórias de Usuário e gerenciar (**criar, ler, atualizar e excluir (CRUD)** e priorizar tarefas) o **Product Backlog**. O PO também possui acesso de **leitura (Read-Only)** ao **Sprint Backlog** (quadro Kanban) para acompanhar o progresso e aceitar itens concluídos, mas não pode gerenciar o projeto ou seus membros.
- **Ambiente e Contexto de Uso (C4):** Utiliza o sistema em sua estação de trabalho (notebook/desktop), primariamente durante as reuniões diárias (Daily Scrum) e Planejamento de Sprint (Sprint Planning) para gerenciar o backlog com a equipe.

5.2. Perfil: Scrum Master (SM)

- **Descrição (Stakeholder):** A parte interessada que atua como facilitador do processo Scrum, removendo impedimentos e garantindo que os ritos sejam seguidos.
- **Responsabilidades (C3):** Foco total no "Contexto"e "Processo". Utilizar o SGAS como administrador do ambiente: **criar projetos** e gerenciar o **CRUD de membros** (convidar, definir papéis, remover). É o único responsável por gerenciar o ciclo de vida das Sprints (**criar, iniciar e encerrar**). Possui acesso de **leitura (Read-Only)** aos backlogs e histórias de usuários para facilitar os ritos.
- **Ambiente e Contexto de Uso (C4):** Acessa o sistema em sua estação de trabalho, frequentemente projetando a ferramenta em telas compartilhadas durante os ritos (Planning, Daily, Review) para prover visibilidade do progresso e do fluxo de trabalho.

5.3. Perfil: Developer (Membro da Equipe)

- **Descrição (Stakeholder):** A parte interessada responsável por realizar o trabalho técnico necessário para entregar o incremento do produto.
- **Responsabilidades (C3):** Foco total no "Como?"(a execução). Durante o Planejamento, são os donos do plano: selecionam itens do Product Backlog e são responsáveis pelo **CRUD do Sprint Backlog** e do **Plano de Sprint** (as tarefas técnicas). Diariamente, gerenciam o quadro Kanban, **movendo tarefas** (A Fazer, Fazendo, Concluído) e atualizando o status do trabalho. Não podem iniciar ou encerrar o ciclo da Sprint. Apenas tem acesso a leitura das histórias de usuário e do backlog do produto.
- **Ambiente e Contexto de Uso (C4):** Acessa o sistema a partir de sua estação de trabalho de desenvolvimento. O uso é diário, principalmente para consulta durante a Daily Scrum e para atualização de tarefas. A interface deve ser rápida para não interromper o fluxo de desenvolvimento.

6. Escopo do Produto

6.1. Necessidades Atendidas (C5)

Com base nos problemas e usuários descritos, o sistema **SGAS** deve atender às seguintes necessidades de alto nível:

- A necessidade de **identificação e segurança**, permitindo que cada usuário crie uma conta e seja autenticado.
- A necessidade de **controle de acesso**, garantindo que usuários autenticados tenham acesso aos serviços de acordo com seu papel (PO, SM, Dev).
- A necessidade de **organização de múltiplos projetos**, permitindo que um usuário integre uma ou mais equipes/projetos.
- A necessidade de **gerenciamento do Product Backlog**, permitindo criar, ler, atualizar e excluir (CRUD) Histórias de Usuário.
- A necessidade de **gerenciamento de Sprints**, permitindo o CRUD do Backlog de Sprint e do Plano de Sprint.
- A necessidade de **detalhamento e transparência do Sprint**, permitindo que a equipe acesse as informações essenciais do Plano de Sprint: objetivo (what), trabalho (how), equipe (who), tarefas (backlog) e saída (increment).

6.2. Funcionalidades Principais (Features) (C6)

Para atender às necessidades listadas, as seguintes funcionalidades de alto nível (features) serão implementadas.

- **Feature 1: Gestão de Contas e Autenticação**
 - Descrição resumida: Permitir que usuários criem uma nova conta de acesso. Prover um sistema de login para autenticação segura.
- **Feature 2: Gestão de Projetos e Permissões**
 - Descrição resumida: Permitir a criação de novos projetos. Permitir que um usuário Scrum Master atribua usuários a um ou mais projetos, definindo seus papéis (Product Owner ou Developer) em cada um.
- **Feature 3: Gerenciamento do Product Backlog**
 - Descrição resumida: Implementar o CRUD (Criar, Ler, Atualizar, Excluir) para o **Product Backlog** e seus itens, as **Histórias de Usuário**.
- **Feature 4: Gerenciamento de Sprints**
 - Descrição resumida: Implementar o CRUD para o **Backlog de Sprint** (o subconjunto de Histórias de Usuário selecionadas) e para o **Plano de Sprint**.
- **Feature 5: Detalhamento do Plano de Sprint**
 - Descrição resumida: Prover uma interface visual que exiba as informações detalhadas do Plano de Sprint, incluindo: o **objetivo** (what), o **trabalho** detalhado (how), a **equipe** envolvida (who), as **tarefas** do backlog (backlog) e o **incremento** esperado (output).

6.3. Fora do Escopo (Limites)

Para garantir o foco na simplicidade e no "Scrum Essencial", os seguintes itens estão explicitamente **fora do escopo** desta versão do produto:

- **Métricas e Relatórios Avançados:** Não serão implementados gráficos de Burndown, Burnup, Velocity ou Cumulative Flow Diagrams (CFD).
- **Gestão Financeira e de Tempo:** O sistema não incluirá módulos de "timesheet" (planilha de horas), gestão de custos ou alocação de recursos financeiros.
- **Integrações Externas:** Não haverá integração nativa com repositórios de código (GitHub, GitLab), ferramentas de CI/CD ou plataformas de comunicação (Slack, Teams).

7. Requisitos Não Funcionais e Restrições

7.1. Requisitos Não Funcionais Resumidos (C7)

Além das funcionalidades, o sistema **SGAS** deve aderir às seguintes características de qualidade mensuráveis e realistas.

- **Segurança:**

- **Autenticação:** As senhas dos usuários **não devem** ser armazenadas em texto plano. Elas devem ser armazenadas utilizando um algoritmo de hash com salt (ex: bcrypt).
- **Autorização:** O acesso às operações de CRUD deve ser validado no backend. Um usuário com papel 'Developer' não pode ter permissão para executar uma ação de 'Product Owner' (ex: excluir uma História de Usuário do Product Backlog) via API.

- **Usabilidade (Consistência):**

- **Consistência de Ações:** Todas as ações primárias (Ex: "Salvar", "Criar", "Confirmar") devem ser representadas pelo mesmo componente visual (ex: botão verde, sólido). Todas as ações secundárias ou destrutivas (Ex: "Cancelar", "Excluir") devem ser representadas por seus próprios componentes padronizados (ex: botão cinza, botão vermelho).
- **Navegação:** A navegação principal (entre Projetos, Backlog e Sprints) deve estar sempre presente e na mesma posição em todas as telas principais.

- **Compatibilidade:**

- **GUI (Web):** O sistema deve ser funcionalmente compatível com as duas últimas versões estáveis dos navegadores Google Chrome e Mozilla Firefox.

- **Desempenho (Tempo de Resposta):**

- O tempo de carregamento inicial do sistema (após o login) não deve exceder **3 segundos** em uma conexão de banda larga padrão.
- As operações de CRUD nos artefatos (criar uma História, salvar uma alteração no Plano de Sprint) devem ter o feedback visual (confirmação ou erro) na interface em menos de **1.5 segundos**.

- **Confiabilidade (Integridade de Dados):**

- O sistema deve garantir a integridade referencial dos dados. Não deve ser possível excluir um 'Projeto' se ele ainda contiver 'Histórias de Usuário' associadas (a menos que seja uma exclusão em cascata confirmada).

- **Conformidade Legal:**

- O sistema deve estar em conformidade com a LGPD no que tange ao consentimento do usuário para a criação da conta e armazenamento de seus dados pessoais (nome, e-mail).

7.2. Restrições (Constraints)

O projeto deve operar dentro das seguintes restrições (limitações ou decisões já tomadas) que impactam o design e a implementação:

- **Restrição de Foco:** O sistema é **exclusivamente** para o framework Scrum. Esta é uma restrição de design que proíbe a adição de funcionalidades de outros métodos (como XP, etc.).
- **Restrição de Tecnologia (Negócio):** Para viabilizar a proposta de valor de "baixo custo" e "acessibilidade", a solução deve ser construída utilizando um stack de tecnologia **open-source**.
- **Restrição de Interface:** O sistema não é "mobile-first". O design deve priorizar o uso em estações de trabalho (desktops/notebooks), conforme requisito.

8. Elementos da Solução Proposta (C8)

Para atender aos requisitos funcionais e não funcionais dentro das restrições estabelecidas, a equipe de desenvolvimento propõe a seguinte solução técnica de alto nível.

A proposta é baseada na Restrição de Tecnologia (Seção 7.2) de utilizar um stack **100% open-source**.

- **Arquitetura (Padrão):** Será adotada uma arquitetura **N-Tier (3-Camadas) baseada em serviços**.
 - **Camada de Apresentação (Frontend):** Uma aplicação web moderna, do tipo **Single Page Application (SPA)**. Esta camada será responsável por toda a renderização da interface (GUI) e pela comunicação com o backend.
 - **Camada de Aplicação (Backend):** Uma **API RESTful** centralizada. Esta API será a única responsável por aplicar as regras de negócio, realizar a autenticação/autorização (NFR de Segurança) e se comunicar com o banco de dados.
 - **Camada de Persistência (Database):** Um banco de dados relacional (SGBDR) único.
- **Tecnologias Propostas (Stack):**
 - **Frontend (GUI):** **React.js** (ou **Vue.js**). Será uma SPA que consome a API Django.
 - **Backend (API):** **Python** com o framework **Django** (especificamente **Django Rest Framework - DRF**). Esta escolha é estratégica, pois o Django já fornece:

- * Um sistema de autenticação robusto (atendendo ao NFR de Segurança).
 - * Um ORM poderoso que garante a integridade dos dados (atendendo ao NFR de Confiabilidade).
 - * Um painel de Admin nativo (facilitando o papel do "Administrador"implícito).
- **Database: PostgreSQL.** É a escolha recomendada pela comunidade Django para produção e é mais robusto que o SQLite, garantindo a integridade referencial.
- **Ambiente de Desenvolvimento e Implantação:**
 - **Consistência do Ambiente:** Para garantir um ambiente de desenvolvimento consistente, a equipe utilizará gerenciadores de pacotes e ambientes virtuais:
 - * **Backend (Python/Django):** Ambientes virtuais (`venv`) e gerenciamento de pacotes via `pip` com um arquivo `requirements.txt`.
 - * **Frontend (React):** Gerenciamento de pacotes via `npm` com o arquivo `package-lock.json` para travar as versões das dependências.

9. Premissas e Dependências

9.1. Premissas

O sucesso deste projeto e o cumprimento dos requisitos de escopo dentro do prazo estipulado (16/11/2025) baseiam-se nas seguintes suposições (premissas):

- **Disponibilidade da Equipe:** Assume-se que todos os 5 membros da equipe terão disponibilidade substancial de tempo para dedicar ao projeto durante o curto período de 14 dias restantes para sua entrega.
- **Trabalho Paralelo Eficaz:** Assume-se que os 9 artefatos solicitados poderão ser desenvolvidos em paralelo pela equipe. (Ex: Enquanto parte da equipe foca no Documento de Visão e Requisitos (Artefatos 2, 3, 4), outra parte já inicia o design da Arquitetura, BD e UI (Artefatos 5, 6, 7)).
- **Proficiência Técnica:** Assume-se que a equipe possui a experiência necessária (conforme definido na Seção 8) com o stack (Python, Django, React, PostgreSQL) para desenvolver o protótipo (Artefato 8) sem a necessidade de um longo período de aprendizado de novas ferramentas.
- **Escopo Congelado:** Assume-se que o escopo definido neste documento (particularmente o "Fora do Escopo"na Seção 6.3) é final e não sofrerá alterações.

9.2. Dependências

O projeto depende dos seguintes fatores externos ou entregas de terceiros:

- **Plataforma de Versionamento (Externa):** O projeto depende de uma plataforma de controle de versões (conforme Roteiro, Item 4.2), da qual será adotada o GitHub, para funcionar como o repositório central dos artefatos e do código-fonte.

- **Infraestrutura de Colaboração (Externa):** Dependemos de ferramentas de comunicação (ex: Discord, Slack) e gestão de tarefas (o próprio Kanban do projeto, no Trello ou similar) para coordenar o trabalho paralelo da equipe. As escolhas foram o Discord e Whatsapp para comunicação e o Jira para armazenamento do quadro Kanban.
- **Bibliotecas Open-Source (Terceiros):** A Solução Proposta (Seção 8) depende da disponibilidade, licenciamento permissivo (ex: MIT, BSD) e estabilidade das bibliotecas de terceiros (Django, DRF, React, etc.).

Aprovações

Eu, o Patrocinador/Product Owner deste projeto, li, entendi e concordo com a Visão e o Escopo definidos neste documento. A aprovação deste documento formaliza o escopo e autoriza a equipe de desenvolvimento a iniciar os trabalhos de design e implementação.

Aprovado por:

Papel: Product Owner (PO)

Data: __/__/_____