

## Trabalho Engenharia de Software

### Especificação de requisitos não funcionais

#### 1. Introdução

Este documento detalha os Requisitos Não Funcionais (também chamados de Requisitos de Suporte ou System-Wide) para o sistema de gerenciamento Scrum UnBScrum. Ele define os atributos de qualidade, interfaces e restrições que guiarão o projeto e o desenvolvimento do sistema.

#### 2. System-Wide Functional Requirements

1. **Autenticação:** O sistema deve prover um mecanismo de criação de conta e autenticação (login) para todos os usuários.
2. **Autorização Baseada em Papel (RBAC):** O acesso aos serviços do sistema será determinado pelo papel do usuário (Product Owner, Scrum Master, Developer) dentro de um projeto específico.

#### 3. System Qualities

##### 1. Usability

- **Facilidade de Aprendizado:** A interface deve ser intuitiva. Um usuário com conhecimento básico do framework Scrum deve ser capaz de realizar suas tarefas principais (ex: criar história, iniciar sprint) sem a necessidade de um manual.
- **Consistência Terminológica:** O sistema deve usar estritamente a terminologia do Scrum (ex: "Product Backlog", "Sprint Backlog", "História de Usuário").
- **Feedback ao Usuário:** O sistema deve fornecer feedback claro e imediato para as ações do usuário (ex: "História salva com sucesso", "Erro: O campo 'Nome' é obrigatório").

##### 2. Reliability

- **Disponibilidade:** O sistema (protótipo) deve estar disponível para acesso durante o período de avaliação.
- **Recuperabilidade:** Em caso de falha de conexão com o banco de dados, o sistema deve apresentar uma página de erro amigável, sem expor detalhes técnicos da falha ao usuário.
- **Validação de Dados:** Todas as entradas de dados em formulários devem ser validadas no lado do servidor para prevenir a entrada de dados corrompidos.

##### 3. Performance

- **Tempo de Resposta (Login):** A autenticação do usuário não deve exceder 2 segundos.
- **Tempo de Resposta (Carregamento de Dados):** O carregamento das páginas de "Product Backlog" e "Sprint Backlog", contendo até 100 histórias, deve ocorrer em menos de 4 segundos.
- **Tempo de Resposta (Transação):** As ações de criar, atualizar ou excluir um artefato Scrum (História, Sprint) devem ser concluídas em menos de 1,5 segundos.

##### 4. Manutenibilidade

- **Compatibilidade de Navegador:** A interface gráfica deve ser compatível com as duas últimas versões dos navegadores Google Chrome, Edge, Safari e Mozilla Firefox.
- **Versionamento:** Todo o código-fonte do protótipo e os artefatos de documentação devem ser armazenados em um sistema de controle de versão (GitHub).
- **Escalabilidade:** A arquitetura deve permitir a adição futura de novos papéis ou artefatos Scrum sem a necessidade de uma reestruturação completa.

## 4. System Interfaces

### 1. User Interfaces

- **Tipo de Interface:** O sistema implementa uma Interface Gráfica de Usuário (GUI) baseada na web.
- **Look & Feel:** A interface deve ser limpa, profissional e focada na funcionalidade, evitando desordem visual. O design seguirá os wireframes definidos no Artefato 6.
- **Consistência:** Elementos de interface comuns (ex: botões "Salvar", "Cancelar", "Excluir") devem ter aparência, posição e comportamento consistentes em todas as telas do sistema.
- **Acessibilidade:** O sistema deve ser acessível a partir das estações de trabalho dos membros do projeto (desktops e notebooks).

### 2. Interfaces com Sistemas Externos ou Dispositivos

Não há requisitos de interface com sistemas externos ou dispositivos para esta versão do protótipo.

## 5. Regras de Negócio

As regras de negócio a seguir definem a lógica central de acesso e operação do sistema, baseada nos papéis do Scrum:

- **Criação de Conta:** Um usuário pode criar uma conta, mas só pode interagir com projetos após ser autenticado.
- **Acesso ao Projeto:** Um usuário só pode ver ou modificar artefatos de projetos aos quais ele está associado.
- **Papel Product Owner:** Apenas usuários com o papel Product Owner podem criar, editar, excluir e priorizar Histórias de Usuário no Product Backlog.
- **Papel Scrum Master:** Apenas usuários com o papel Scrum Master podem criar Sprints, definir o Plano de Sprint e mover Histórias do Product Backlog para o Sprint Backlog.
- **Papel Developer:** Usuários com o papel Developer podem visualizar o Sprint Backlog e associar tarefas a ele.

## 6. Restrições do Sistema

Estas são as restrições de design e implementação definidas pela equipe:

- **Tecnologia Backend:** O backend do sistema será desenvolvido usando Python e FastAPI.
- **Tecnologia Frontend:** O frontend será desenvolvido usando HTML.
- **Banco de Dados:** O sistema utilizará um banco de dados relacional MySQL.
- **Plataforma de Versionamento:** O GitHub será usado para controle de versão.
- **Hospedagem do Protótipo:** O protótipo funcional (Artefato 8) será implantado localmente.

## 7. System Compliance

### 1. Licensing Requirements

- **Licença do Software:** O código-fonte produzido para este projeto será disponibilizado sob a licença MIT License.
- **Licenças de Componentes:** Todas as bibliotecas e frameworks de terceiros utilizados devem ter licenças compatíveis (MIT, Apache 2.0) que permitam o uso acadêmico e em protótipos.

### 2. Legal, Copyright, and Other Notices

- Não é aplicável para este protótipo acadêmico.

### 3. Applicable Standards

- **LGPD (Lei Geral de Proteção de Dados):** O sistema deve estar em conformidade com a LGPD no que tange ao armazenamento e tratamento de dados pessoais dos usuários (nome, email), garantindo que senhas sejam armazenadas com hash e salt.
- **Guia Scrum:** A terminologia e o fluxo de trabalho implementados devem ser aderentes aos conceitos definidos no Guia Scrum oficial.

## 8. System Documentation

- **Documentação de Entrega:** Além deste e dos outros artefatos textuais (em PDF) , será produzido um vídeo de demonstração.
- **Documentação Interna:** O código-fonte deve conter comentários claros para facilitar a manutenção e avaliação.