

**FACULDADE DE TECNOLOGIA DA ZONA LESTE**  
**CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**GUSTAVO HENRIQUE DA SILVA**  
**JULIA VICTORIA DA SILVA FERNANDES**  
**NICOLAS DOMINGOS DA SILVA**

**GOODOOIT: SISTEMA DE GERENCIAMENTO DE TAREFAS**

**SÃO PAULO**

**2025**

**FACULDADE DE TECNOLOGIA DA ZONA LESTE**  
**CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**GUSTAVO HENRIQUE DA SILVA**

**JULIA VICTORIA DA SILVA FERNANDES**

**NICOLAS DOMINGOS DA SILVA**

**GOODOOIT: SISTEMA DE GERENCIAMENTO DE TAREFAS**

Documentação do sistema de gestão de tarefas  
GooDooIt desenvolvido como trabalho  
semestral para a conclusão da disciplina de  
Banco de Dados, sob orientação do Prof.  
M.Sc. Leandro Colevati

**SÃO PAULO**

**2025**

## SUMÁRIO

<b>1. INTRODUÇÃO</b>	<b>4</b>
<b>2. OBJETIVOS DO SISTEMA</b>	<b>5</b>
3.1. DESENVOLVIMENTO CRIATIVO	6
3.2. MASCOTE	6
<b>4. REQUISITOS DO SISTEMA</b>	<b>7</b>
4.1. REQUISITOS FUNCIONAIS	7
4.2. REQUISITOS NÃO FUNCIONAIS	9
<b>5. REGRAS DE NEGÓCIO</b>	<b>9</b>

## 1. INTRODUÇÃO

A organização de tarefas e o gerenciamento de atividades são desafios recorrentes no cotidiano de estudantes, profissionais e equipes que trabalham de forma colaborativa. Em muitos casos, a ausência de uma ferramenta simples e acessível dificulta o planejamento e o acompanhamento das responsabilidades, comprometendo prazos, metas e a produtividade como um todo.

Foi com essa preocupação em mente que surgiu a proposta do GooDooIt: um sistema de gerenciamento de tarefas que alia simplicidade, praticidade e colaboração em um único ambiente. A ideia nasceu da vivência com sistemas mais robustos e, muitas vezes, excessivamente complexos para demandas mais objetivas — como o gerenciamento de atividades acadêmicas, por exemplo.

O GooDooIt foi desenvolvido com base em uma estrutura visual clara e direta, utilizando alguns dos princípios do modelo Kanban. O sistema organiza as tarefas em três quadros fixos — *A fazer*, *Em andamento* e *Concluídas* — promovendo uma visualização rápida do andamento de um projeto. Além disso, oferece recursos como convites para participação em projetos, interações em tarefas, notificações automatizadas e controle de permissões, com o foco em uma experiência intuitiva.

Pensado para funcionar de forma local, sem a exigência de conexão com a internet, o sistema foi construído utilizando a linguagem Java, com interface gráfica em JavaFX e banco de dados SQL Server. Essas escolhas técnicas visam garantir estabilidade, desempenho e facilidade de uso, especialmente em ambientes acadêmicos.

Este documento apresenta todas as etapas do desenvolvimento do sistema, desde a definição dos requisitos até os modelos de dados, diagramas de classes, casos de uso e demais aspectos que estruturam a solução. O GooDooIt é, acima de tudo, uma tentativa sincera de facilitar a organização pessoal e em grupo, com leveza, eficiência e propósito — e, com um pouco de sorte (e bastante esforço), também uma boa tentativa de garantir uma nota positiva na disciplina de Banco de Dados.

## **2. OBJETIVOS DO SISTEMA**

O principal objetivo do sistema GooDooIt é oferecer uma ferramenta para o gerenciamento de tarefas em projetos individuais ou colaborativos. A proposta é permitir que os usuários organizem suas atividades de forma visual e eficiente, utilizando uma estrutura que permite registrar cada tarefa durante o desenvolvimento de um projeto. Além da organização pessoal, o sistema também busca incentivar o trabalho em equipe, oferecendo recursos para convites, atribuições de responsabilidade e controle de permissões

Do ponto de vista técnico, o GooDooIt tem como objetivo aplicar os conhecimentos adquiridos em disciplinas como Banco de Dados, Programação Orientada a Objetos e Engenharia de Software, explorando conceitos de modelagem, camadas de arquitetura e persistência de dados.

## **3. JUSTIFICATIVAS DO SISTEMA**

O desenvolvimento do sistema GooDooIt surgiu a partir de uma necessidade observada entre os integrantes do grupo no contexto universitário: a dificuldade de organizar tarefas acadêmicas de forma colaborativa usando ferramentas acessíveis. Muitas das aplicações disponíveis atualmente exigem conexão com a internet, possuem funcionalidades importantes bloqueadas por recursos pagos — como a atribuição de datas às tarefas — ou apresentam uma série de recursos que atendem às necessidades avançadas, mas pecam no simples.

Diante disso, o grupo propôs a criação de uma solução leve, funcional e gratuita, que permita organizar projetos de maneira visual, com suporte a prazos e colaboração em equipe, funcionando de forma totalmente local, para atender às necessidades encontradas. O projeto também se apresenta como uma oportunidade de aplicar na prática os conhecimentos adquiridos até aqui.

### 3.1. DESENVOLVIMENTO CRIATIVO

Para além da funcionalidade, o sistema GooDooIt também foi pensado com um toque de identidade. O nome surgiu como uma brincadeira entre a expressão “*Go do it!*” — que significa “vá e faça!” — e o som de um fantasma dizendo “boo”.

A ideia do mascote surgiu de uma suave obsessão por coisas assombradas com uma pitada de humor. Em vez de criar mais um sistema genérico de tarefas, quisemos dar personalidade ao projeto. O fantasma, Goo, é uma criatura simpática que não assusta pessoas, mas sim a bagunça e a desorganização.

O GooDooIt, portanto, não é apenas uma ferramenta: é quase um companheiro na missão de colocar a vida em ordem. Ele convida o usuário a se organizar de forma leve, sem pressão, mas com consistência. Incorporar essa identidade visual e simbólica ao sistema foi uma forma de tornar a experiência mais próxima, mais humana — e, quem sabe, até mais divertida.

### 3.2. MASCOTE



O mascote do GooDooIt é, naturalmente, um fantasma. Sua aparência mistura fofura com um certo ar de urgência, como quem diz: “vamos lá, você consegue!”. A escolha por um fantasma também faz parte do espírito leve do sistema, fugindo do óbvio e reforçando a ideia de que se organizar pode — e deve — ser algo mais divertido do que assustador.

Visualmente, o mascote aparece como um elemento de apoio à interface do sistema, podendo estar presente em telas de boas-vindas, alertas ou até mesmo como um aviso em momentos importantes como trocar a senha. Ele é o “rosto” do GooDooIt, dando personalidade ao projeto e tornando a experiência mais marcante e acolhedora.

#### 4. REQUISITOS DO SISTEMA

A engenharia de requisitos do sistema foi desenvolvida com o objetivo de identificar e organizar de forma clara todas as funcionalidades necessárias para o funcionamento do GooDooIt, bem como as restrições e expectativas relacionadas à sua operação, visando atender às necessidades dos nossos principais usuários

##### 4.1. REQUISITOS FUNCIONAIS

Código	Descrição
RF01	O sistema deve permitir o cadastro de um novo usuário com nome de usuário e senha, desde que o usuário seja único.
RF02	O sistema deve validar o login do usuário utilizando nome de usuário e senha cadastrados.
RF03	O sistema deve manter a sessão autenticada do usuário enquanto a aplicação estiver aberta.
RF04	O sistema deve permitir que o usuário altere seu nome, e-mail e senha, mas não o nome de usuário.
RF05	O sistema deve permitir que o usuário crie um novo projeto informando nome e descrição.
RF06	O sistema deve listar todos os projetos criados pelo usuário autenticado.
RF07	O sistema deve permitir que o criador de um projeto convide outros usuários para colaborar no projeto.
RF08	O sistema deve notificar o usuário convidado e permitir que ele aceite ou recuse o convite para participar do projeto.

RF09	O sistema deve permitir que somente o criador de um projeto edite as informações sobre ele ou exclua o projeto.
RF10	O sistema deve permitir que o criador remova usuários da equipe do projeto.
RF11	O sistema deve permitir que o usuário saia de um projeto do qual participa.
RF12	O sistema deve permitir que o usuário visualize todas as equipes das quais faz parte.
RF13	O sistema deve permitir ao criador da tarefa e ao dono do projeto editar os dados da tarefa, incluindo título, descrição, prazo e prioridade.
RF14	O sistema deve permitir que qualquer usuário do projeto possa atribuir uma tarefa a si mesmo e a outros usuários do projeto.
RF15	O sistema deve permitir reatribuir uma tarefa a outro usuário.
RF16	O sistema deve permitir que somente o dono do projeto possa excluir uma tarefa.



## 4.2. REQUISITOS NÃO FUNCIONAIS

Código	Descrição
RNF01	O sistema deve ser executado localmente, sem necessidade de conexão com a internet.
RNF02	O sistema deve utilizar banco de dados relacional (SQL Server) para persistência das informações.
RNF03	A interface gráfica do sistema deve ser desenvolvida com JavaFX.
RNF04	O sistema deve ser compatível com os sistemas operacionais Windows 10 e superiores.
RNF05	O sistema deve utilizar uma arquitetura em camadas, separando lógica de apresentação, aplicação, domínio e infraestrutura.
RNF06	Toda interação com o sistema deve ser feita por meio de interface gráfica amigável, sem dependência de terminal ou comandos manuais.
RNF07	O sistema deve manter um histórico mínimo de ações (comentários, notificações), com integridade dos registros.
RNF08	A interface deve evitar o uso de fontes serifadas, utilizando tipografia limpa e de fácil leitura.
RNF09	O sistema deve apresentar mensagens amigáveis em caso de erro, evitando termos técnicos como “NullPointerException”.

## 5. REGRAS DE NEGÓCIO

O sistema possui 11 regras de negócio que definem as restrições e comportamentos esperados. Essas regras foram elaboradas com base nos casos de uso identificados, visando manter a consistência das permissões, papéis dos usuários e estrutura dos projetos.

#### **RN01**

Somente o criador de um projeto pode editar suas informações, excluí-las e remover membros da equipe. Esse mesmo criador tem controle total sobre a estrutura do projeto, incluindo listas, tarefas e permissões de exclusão de conteúdo.

#### **RN02**

O nome de usuário cadastrado no sistema deve ser único. Após o cadastro, o nome de usuário não pode ser alterado. O usuário pode modificar seu nome pessoal, e-mail e senha, mas nunca o identificador de login.

#### **RN03**

Um usuário só pode visualizar e interagir com projetos dos quais faça parte. A entrada em um projeto ocorre apenas após o aceite de um convite enviado pelo criador do projeto.

#### **RN04**

Qualquer membro da equipe de um projeto pode criar tarefas, mas apenas o criador da tarefa ou proprietário do projeto pode editá-las ou excluí-las. As tarefas podem ser movidas livremente entre os três quadros fixos do projeto.

#### **RN05**

Uma tarefa pode ser atribuída a um único usuário por vez, desde que ele já seja integrante do projeto. Essa atribuição pode ser modificada posteriormente.

#### **RN06**

O sistema deve gerar notificações automáticas sempre que um usuário for adicionado ou convidado a um projeto, permitindo o aceite ou a recusa, de modo que se torne um colaborador quando aceitar.