

# OneTouch Metas

## Gerenciador de tarefas para gestores

Kevin Bisso Kunde<sup>1</sup>, Ângelo Gonçalves da Luz<sup>1</sup>

<sup>1</sup>Curso de Análise e Desenvolvimento de Sistemas – Faculdade de Tecnologia SENAC Pelotas  
Rua Gonçalves Chaves 602 – 96015560 – Pelotas – RS – Brazil

kevinkunde96@gmail.com, ageluz@senacrs.edu.br

**Abstract.** *This article describes the functioning of a task management system, its development and its respective technologies, aiming at facilitate productivity control for managers. It also shows how the system facilitates the day-to-day of your users. It also brings a study on the growing need to organize day-to-day tasks and monitor*

**Keywords:** *OneTouch, Tasks, Managers, Productivity, Team.*

**Resumo.** *Este artigo descreve o funcionamento de um sistema de gerenciamento de tarefas, seu desenvolvimento e suas respectivas tecnologias, visando facilitar o controle de produtividade para gestores. Também mostra como o sistema facilita o dia-a-dia dos seus usuários. Também traz um estudo sobre o crescimento da necessidade de organização das tarefas do dia-a-dia e monitoramento.*

**Palavras-Chave:** *OneTouch, Tarefas, Gestores, Produtividade, Equipe.*

### 1. Introdução

No mundo atual, onde o ritmo de vida é cada vez mais acelerado, estabelecer metas claras e acompanhar o progresso das tarefas se tornou uma necessidade urgente para alcançar objetivos de maneira eficiente. Diante desse contexto, este artigo apresenta o desenvolvimento de um sistema de gerenciamento de metas e tarefas, que visa auxiliar indivíduos e organizações a planejar, monitorar e alcançar suas metas de forma sistemática e eficaz.

O gerenciamento adequado de metas e tarefas é um fator fundamental para o sucesso e a realização pessoal. No entanto, frequentemente nos deparamos com dificuldades para manter o foco, estabelecer prioridades e acompanhar o progresso de nossas atividades diárias. Nesse sentido, o sistema proposto neste artigo surge como uma solução tecnológica capaz de simplificar e aprimorar o processo de gerenciamento. Com recursos intuitivos e personalizáveis, o sistema permite definir metas específicas, estabelecer prazos realistas, atribuir tarefas relacionadas a cada objetivo e rastrear o progresso.

Este projeto se justifica pela necessidade de suprir a lacuna existente no gerenciamento eficiente de metas e tarefas, proporcionando uma solução prática e integrada para indivíduos e organizações. Ao longo deste artigo, serão explorados os desafios enfrentados durante o desenvolvimento do sistema, a metodologia adotada, as funcionalidades-chave e estudos de caso para validar sua eficácia. O objetivo geral deste projeto é desenvolver um sistema de gerenciamento de metas e tarefas que seja intuitivo, personalizável e eficiente. Como objetivos específicos, buscaremos implementar uma interface amigável,

permitir a definição de metas personalizadas, estabelecer prazos e atribuir tarefas relacionadas, além de fornecer recursos de acompanhamento do progresso.

Através deste trabalho, espera-se proporcionar aos usuários uma ferramenta confiável e eficaz para o gerenciamento de metas e tarefas, auxiliando-os a otimizar seu tempo, aumentar a produtividade e alcançar seus objetivos de forma eficiente. Ao final deste artigo, serão apresentadas conclusões e considerações finais sobre a importância desse sistema na vida pessoal e profissional, destacando seu potencial para transformar a maneira como lidamos com o planejamento e a realização de metas.

## **2. Referencial Teórico**

A gestão de tarefas desempenha um papel fundamental na organização das atividades necessárias para alcançar os objetivos e metas de uma equipe. Esse processo inclui a identificação e levantamento das tarefas, a definição de prioridades, a delegação adequada e o acompanhamento da execução dentro dos prazos estabelecidos. No entanto, segundo [Stefano Pacini, Rodolpho Tobler, Viviane Seda Bittencourt 2023] com o início da pandemia em 2020 e a crescente adoção do trabalho remoto, a responsabilidade de gerenciar o tempo das atividades tornou-se ainda mais desafiadora.

Durante esse período, foi possível observar de perto as dificuldades enfrentadas pelos gestores da empresa em que o autor deste trabalho atua, no que diz respeito ao gerenciamento eficiente das tarefas dos funcionários e ao acompanhamento de seu progresso. O contexto de trabalho remoto acrescentou uma camada adicional de complexidade, exigindo a adaptação das práticas de gestão existentes.

Ao longo dessa experiência, foi notável o aumento da carga de responsabilidade dos gestores para garantir que as tarefas fossem realizadas de forma eficiente e dentro dos prazos estipulados. A falta de um sistema adequado para o gerenciamento de tarefas contribuiu para a sobrecarga dos gestores, comprometendo a produtividade e a efetividade das equipes.

Diante desse cenário, este trabalho tem como objetivo desenvolver uma solução para a gestão de tarefas, visando simplificar e otimizar o processo de acompanhamento das atividades. Por meio de um sistema intuitivo e personalizável, pretende-se fornecer aos gestores uma ferramenta eficaz para definir, atribuir, acompanhar e avaliar o progresso das tarefas.

Ao implementar esse sistema, espera-se que os gestores possam gerenciar as tarefas de forma mais eficiente, reduzindo o esforço manual e aumentando a produtividade da equipe. Além disso, a solução proposta visa melhorar a comunicação entre gestores e membros da equipe, proporcionando uma visão clara e compartilhada do progresso das atividades.

Ao longo deste artigo, serão abordados os desafios enfrentados durante o desenvolvimento da solução, a metodologia adotada, as funcionalidades-chave do sistema e estudos de caso para validar sua eficácia. Acredita-se que essa solução trará benefícios tangíveis para as organizações, permitindo uma gestão de tarefas mais eficiente, uma melhor distribuição de responsabilidades e um aumento na produtividade geral da equipe.

Posteriormente, será descrito o processo de desenvolvimento da solução, incluindo a arquitetura do sistema, as tecnologias utilizadas e as etapas de implementação.

## **2.1. Trabalhos Relacionados**

As ferramentas disponíveis na internet, como o Jira e o Trello, foram identificadas como complexas e demandando tempo considerável para a organização das tarefas. Além disso, constatou-se que essas ferramentas não estavam especificamente direcionadas para atender às necessidades dos gestores, pois não forneciam relatórios ou painéis de controle que permitissem uma visualização clara do progresso.

Diante desse contexto, o objetivo deste trabalho é o desenvolvimento de um sistema web que possibilite aos supervisores cadastrar e gerenciar os funcionários sob sua supervisão. Os usuários terão acesso às suas tarefas individuais e poderão marcá-las como concluídas. Os supervisores, por sua vez, terão a capacidade de criar, delegar e excluir tarefas, além de contar com um painel de controle (dashboard) que apresentará o progresso dos funcionários em base diária, semanal ou mensal. A proposta central é fornecer aos gestores uma visão clara e objetiva do progresso de suas equipes, facilitando a tomada de decisões e o acompanhamento do desempenho dos funcionários.

Para atingir esse objetivo, serão utilizadas tecnologias web como ReactJs, Node, Prisma, PostgreSQL. O sistema será desenvolvido com ênfase na facilidade de uso e na interface intuitiva, de modo a proporcionar uma experiência agradável aos usuários. Além disso, serão aplicadas práticas de design responsivo, garantindo que o sistema seja acessível e adaptável a diferentes dispositivos e tamanhos de tela.

## **3. Projeto de Sistema**

Nesta sessão serão apresentados os requisitos funcionais e não funcionais do sistema, a apresentação do diagrama de classes do sistema, dos casos de uso, as tecnologias utilizadas no sistema e uma breve descrição do funcionamento do mesmo.

### **3.1. Levantamento de Requisitos**

De maneira breve, levantamento de requisitos trata-se do processo de compreensão e identificação das necessidades que o cliente espera ser solucionado pelo sistema desenvolvido, definindo o que o software irá desempenhar.

É a etapa número um no ciclo de desenvolvimento de software, onde se definem as funcionalidades e o escopo do projeto. Existem dois tipos de requisitos que compõem um sistema, os Requisitos Funcionais (RF) e os Requisitos não-funcionais (RNF).

Os requisitos funcionais tratam das funcionalidades que o sistema deve ter e ser capaz de executar. Sendo assim, segue a lista de requisitos funcionais:

[RF 01] Cadastrar novos usuários

RN: Somente o usuário que é gestor poderá cadastrar novos usuários, informando nome, cargo (se é manager ou usuário normal), gestor imediato, e-mail e senha do usuário.

[RF 02] Editar usuários

RN: Apenas gestores poderão editar usuários existentes.

[RF 03] Excluir usuários

RN: Apenas gestores poderão excluir usuários existentes.

[RF 04] Listar usuários cadastrados

RN: Apenas gestores poderão listar os usuários cadastrados.

[RF 05] Cadastro de tarefas

RN: Somente os gestores poderão criar uma tarefa nova, informando o título, o corpo da tarefa, o usuário responsável pela tarefa, e uma data esperada para conclusão da mesma.

[RF 06] Edição de tarefas

RN: Usuários que sejam gerentes do usuário responsável pela tarefa poderão fazer a edição da mesma.

[RF 07] Exclusão de tarefas

RN: Usuários que sejam gerentes do usuário responsável pela tarefa poderão fazer a exclusão da mesma.

[RF 08] Concluir tarefas

RN: O responsável pela tarefa poderá concluir uma tarefa.

[RF 09] Listar tarefas

RN: O usuário responsável pela tarefa poderá ver apenas as suas tarefas, já os gestores poderão ver as tarefas de seus supervisionados.

[RF 10] Visualizar dashboard

RN: Gestores poderão visualizar a parte de dashboard, onde irão ter informações sobre o rendimento dos seus supervisionados.

[RF 11] Filtrar por tarefas diárias

RN: Mostrará apenas tarefas criadas naquele dia

[RF 12] Filtrar por tarefas criadas na semana

RN: Mostrará apenas tarefas criadas naquela semana

[RF 13] Filtrar por tarefas criadas no mês

RN: Mostrará apenas tarefas criadas naquele mês

[RF 14] Filtrar por tarefas criadas em determinado período

RN: Mostrará apenas tarefas criadas em determinada data

Os requisitos não funcionais são recursos que não são funcionalidades, mas características do sistema, como restrições, segurança, confiabilidade, velocidade, validações, entre outros.. Sendo assim, segue a lista de requisitos funcionais:

[RNF 01] O sistema deverá rodar em todos os navegadores.

[RNF 02] O sistema utilizará criptografia para segurança de dados.

[RNF 03] O banco de dados será relacional.

[RNF 04] Será utilizada a linguagem Javascript no front-end

[RNF 05] Será utilizada a linguagem Node.js no back-end

[RNF 06] Será utilizado o Tailwind para estilização dos componentes

[RNF 07] Será responsivo

### 3.2. UML

Segundo [DevMidia 2012] "UML é um acrônimo para a expressão Unified Modeling Language. Pela definição de seu nome, vemos que a UML é uma linguagem que define uma série de artefatos que nos ajuda na tarefa de modelar e documentar os sistemas orientados a objetos que desenvolvemos. Ela possui nove tipos de diagramas que são usados para documentar e modelar diversos aspectos dos sistemas".

### 3.3. Casos de Uso

O diagrama de casos de uso documenta o que o sistema faz do ponto de vista do usuário. Ele descreve as funcionalidades e interação delas com os usuários desse sistema. Nesse diagrama não se aprofunda em detalhes técnicos que dizem como o sistema faz.

Conforme a Figura 1, o usuário é representado por um boneco palito e ligando ele às tarefas que o mesmo poderá realizar.

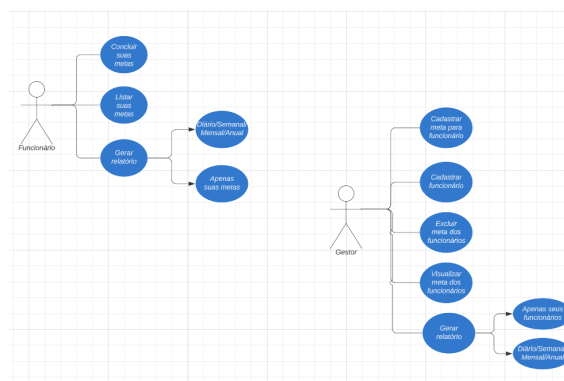


Figura 1. Diagrama de casos de uso do sistema.

### 3.4. Diagrama Entidade Relacionamento

De acordo com [DevMidia 2014], o Modelo Entidade Relacionamento (ER) é um modelo conceitual amplamente utilizado na Engenharia de Software para descrever os objetos (entidades) envolvidos em um determinado domínio de negócio, juntamente com seus atributos e relacionamentos entre si.

Conforme a Figura 2, o diagrama ER do sistema é composto por três tabelas distintas, as quais serão brevemente detalhadas abaixo:

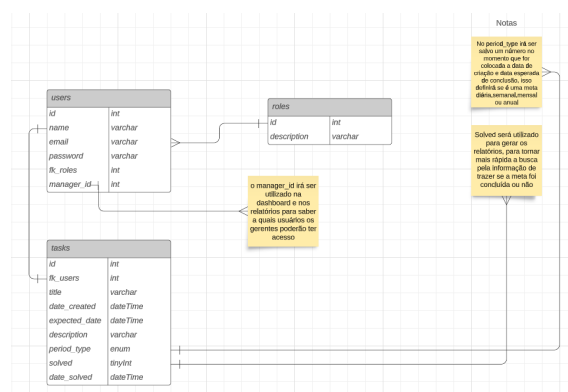
A tabela "user" armazena informações de todos os usuários do sistema, incluindo campos como senha, e-mail, cargo e a referência ao supervisor imediato (identificado pelo ID de outro usuário com a função de gerente).

A tabela "role" representa os diferentes cargos disponíveis no sistema, atualmente definidos como "manager"(gerente) e "user"(usuário). Essa estrutura permite a adição de

novos cargos no futuro, proporcionando maior flexibilidade para a atribuição de diferentes níveis de acesso.

Por fim, temos a tabela "task"(tarefa), responsável pelo registro das atividades a serem realizadas. Essa tabela contém os seguintes campos: ID do usuário responsável pela tarefa, título, descrição, data estimada de conclusão, status de conclusão (solucionada ou não) e data de solução (caso aplicável).

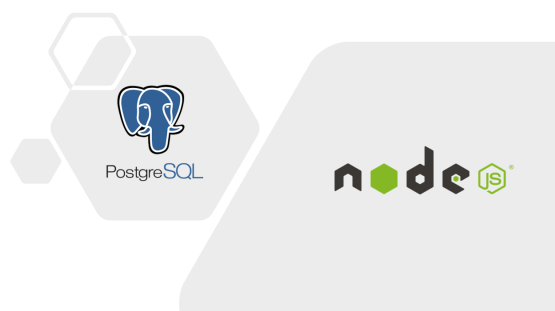
Essa estrutura de tabelas, conforme descrita, permite o armazenamento adequado e organizado dos dados relacionados aos usuários, cargos e tarefas no sistema. Ao seguir essa modelagem ER, o sistema ganha em eficiência e clareza na representação dos relacionamentos entre os diferentes elementos, facilitando a manipulação e consulta das informações.



**Figura 2. Diagrama de Entidade Relacionamento do sistema.**

### 3.5. Tecnologias do Projeto

Para o desenvolvimento do Back-End, conforme ilustrado na Figura 3, optou-se por utilizar o [Node 2009] como tecnologia principal, uma escolha motivada pelo desejo de se desafiar e aprender uma nova ferramenta amplamente utilizada no mercado de trabalho devido à sua eficiência. Para facilitar a integração entre as diferentes partes do sistema, foi empregado o [Prisma 2018], uma popular ORM (Object-Relational Mapping) para Node.js.



**Figura 3. Tecnologias utilizadas no Back-End do projeto.**

No que diz respeito ao banco de dados, a escolha recaiu sobre o [Postgres 1996], uma das plataformas de banco de dados mais amplamente adotadas globalmente, optou-se

pelo [Supabase 2023] para hospedar o banco de dados. Essa decisão foi tomada considerando a reputação do Postgres por sua confiabilidade, robustez e capacidade de lidar com um volume substancial de dados.

A utilização do Node.js como base do Back-End proporciona benefícios significativos, como um ambiente de execução assíncrono, permitindo uma melhor utilização dos recursos do sistema e a construção de aplicações escaláveis e de alto desempenho. A adoção do Prisma como ORM facilita a interação com o banco de dados, simplificando a escrita de consultas e a manipulação dos dados.

Por meio dessa combinação de tecnologias, espera-se obter um Back-End eficiente, confiável e compatível com os requisitos do sistema, fornecendo uma base sólida para a implementação das funcionalidades desejadas. A escolha do Postgres como plataforma de banco de dados garante uma estrutura robusta e estável para armazenar e recuperar os dados relevantes para o sistema.



**Figura 4. Tecnologias utilizadas no Front-End do projeto.**

No desenvolvimento do Front-End, conforme ilustrado na Figura 4, foi escolhido utilizar o [ReactJs 2013] como framework principal, uma decisão fundamentada na agilidade e produtividade que essa tecnologia oferece. O React é amplamente utilizado em conjunto com a linguagem de programação JavaScript, proporcionando uma abordagem eficiente para a criação de interfaces de usuário interativas e responsivas. Além disso, foram empregadas as tecnologias HTML5 e CSS3 (alinhado ao [TailWind 2023]) para estruturar e estilizar as páginas do sistema.

A escolha do React como framework para o Front-End traz diversos benefícios, como a modularidade e reutilização de componentes, facilitando o desenvolvimento e a manutenção do código. A abordagem baseada em componentes também possibilita uma melhor organização e escalabilidade do sistema, tornando-o mais flexível para futuras atualizações e modificações.

Ao utilizar HTML5 e CSS3, foram exploradas as capacidades avançadas dessas tecnologias para criar uma experiência de usuário agradável e moderna. O HTML5 permite a estruturação semântica do conteúdo, facilitando a acessibilidade e a indexação por motores de busca. Por sua vez, o Tailwind proporciona a estilização e o design das páginas, permitindo a criação de interfaces visualmente atraentes e responsivas.

Essa combinação de tecnologias no Front-End proporciona uma base sólida para o desenvolvimento de uma interface de usuário eficiente, intuitiva e de alto desempenho. A escolha do React, em conjunto com Tailwind, demonstra um compromisso em oferecer

uma experiência de usuário aprimorada e alinhada com as tendências atuais de desenvolvimento web.

#### 4. OneTouch-Metas

Conforme mencionado anteriormente, o sistema apresenta duas formas de visualização distintas: uma para os supervisores e outra para os usuários supervisionados, de acordo com o login efetuado.

##### 4.1. Tarefas

A Tela de Tarefas é exclusiva para os gestores, conforme ilustrado na Figura 5, permitindo o cadastro simplificado de tarefas. As informações solicitadas incluem título, descrição, data esperada para conclusão e usuário responsável pela execução.

A Tela de Tarefas traz benefícios significativos para a organização. Simplifica o processo de cadastro, economizando tempo e esforço dos gestores. Também facilita a distribuição equitativa de responsabilidades e identificação de gargalos. A transparência promove um ambiente de trabalho colaborativo e engajado, melhorando a comunicação entre gestores e colaboradores.

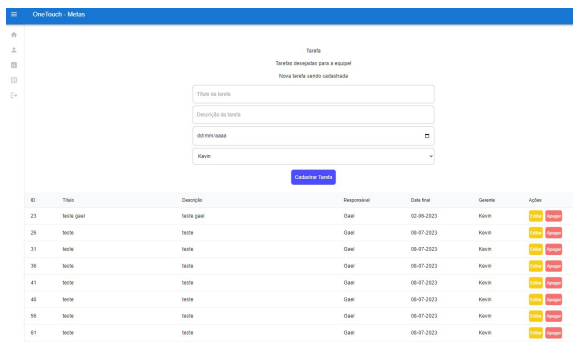


Figura 5. Tela de tarefas visualizada pelo gestor.

##### 4.2. Usuários

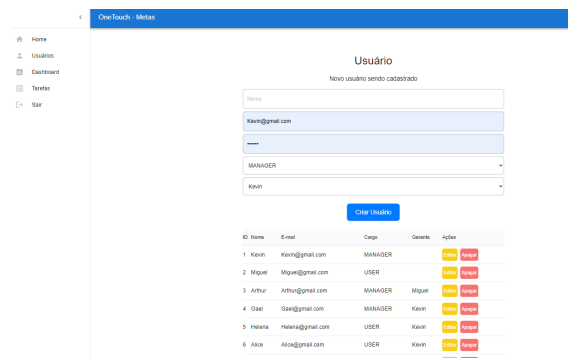
A Tela de Gerenciamento de Usuários, conforme ilustrado na Figura 6, de acesso exclusivo aos gestores, desempenha um papel fundamental na administração dos usuários dentro do sistema. Essa interface oferece recursos abrangentes, permitindo a inclusão de novos usuários, bem como a edição e exclusão dos mesmos conforme necessário.

Através dessa tela, os gestores têm a capacidade de cadastrar novos usuários de forma intuitiva e simplificada. Os campos essenciais incluem nome, e-mail, senha e cargo, que pode ser atribuído como gestor ou usuário. A inclusão dessas informações é fundamental para o correto registro e identificação de cada usuário dentro do sistema.

##### 4.3. Dashboard

A Tela de Dashboard é destinada aos gestores, conforme ilustrado na Figura 7, fornecendo informações valiosas por meio de três gráficos principais. O primeiro gráfico, localizado à esquerda, exibe a quantidade de tarefas realizadas por cada usuário sob a responsabilidade do gestor logado. Essa visualização permite uma análise imediata do desempenho individual de cada membro da equipe.





**Figura 6. Tela de usuários visualizada pelo gestor.**

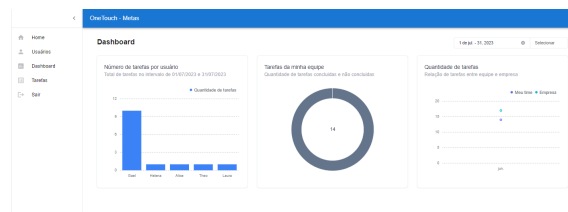
O segundo gráfico, localizado no centro do dashboard, mostra o total de tarefas registradas para os usuários sob a supervisão do gestor logado, bem como o número de tarefas concluídas. Essa representação visual oferece uma visão geral do progresso das tarefas atribuídas à equipe e ajuda o gestor a identificar áreas que requerem atenção ou suporte adicional.

O terceiro gráfico, localizado à direita, permite uma comparação entre os chamados registrados em toda a empresa (para todos os gestores) e aqueles direcionados especificamente à equipe do gestor logado. Essa comparação oferece uma perspectiva valiosa sobre o volume de trabalho enfrentado pela equipe em relação ao restante da organização.

#### Filtro e suas Funcionalidades

A Tela de Dashboard apresenta um filtro localizado no canto superior direito, que permite ao gestor personalizar as informações exibidas de acordo com suas preferências. Por meio desse filtro, o gestor pode determinar um período de datas específico para filtrar os dados exibidos ou selecionar opções pré-definidas, como "hoje", "ontem", "últimos 30 dias" ou "último ano".

É importante ressaltar que o filtro afeta apenas os dois primeiros gráficos, permitindo que o gestor obtenha uma visão mais precisa e focada das atividades recentes e do progresso da equipe. No entanto, o terceiro gráfico é influenciado apenas quando a opção "último ano" é selecionada, fornecendo uma análise comparativa mais ampla.



**Figura 7. Tela de dashboard visualizada pelo gestor.**

## 4.4. Home Gestor

A home destinada aos gestores é um espaço exclusivo, projetado especificamente para atender às necessidades de supervisão e acompanhamento de suas equipes. Nessa área, os gestores têm acesso privilegiado à dashboard, bem como às informações relevantes sobre as tarefas atribuídas aos usuários que estão sob sua supervisão.

Por meio da home do gestor, é possível visualizar de forma clara e organizada os detalhes essenciais de cada tarefa. Isso inclui o título da tarefa, sua descrição, o responsável pela sua execução e a data estabelecida para sua conclusão. Essas informações fornecem uma visão abrangente e precisa do status e dos prazos associados às tarefas em andamento.

Em resumo, a home exclusiva para os gestores oferece uma visão aprofundada e detalhada das tarefas atribuídas aos usuários sob sua supervisão. Com informações como título, descrição, responsável e data de conclusão, os gestores têm acesso a uma visão completa das tarefas em andamento. Essa interface intuitiva e eficiente auxilia na supervisão e no gerenciamento adequado das equipes, contribuindo para um fluxo de trabalho mais eficaz e resultados satisfatórios.

#### **4.5. Home Usuário**

A home destinada aos usuários não gestores é um espaço exclusivo, dedicado a fornecer uma visão clara e organizada de suas tarefas atribuídas. Nessa área, os usuários têm acesso privilegiado às informações essenciais de suas tarefas, incluindo título, descrição, responsável pela execução e data de conclusão.

A interface da home do usuário foi cuidadosamente projetada para permitir uma visualização fácil e rápida das tarefas designadas. Os usuários podem monitorar seus compromissos e responsabilidades, mantendo-se atualizados sobre os prazos e metas a serem cumpridos. Além disso, a home do usuário oferece a funcionalidade de conclusão de tarefas, permitindo que o usuário sinalize quando uma tarefa for concluída.

Ao permitir a conclusão das tarefas diretamente na home do usuário, o sistema proporciona praticidade e agilidade no registro do progresso e na atualização do status das atividades. Isso facilita a comunicação entre os membros da equipe e os gestores, permitindo que todos estejam cientes do andamento das tarefas.

Em suma, a home exclusiva para os usuários não gestores prioriza a visualização clara e acessível de suas tarefas atribuídas. Com detalhes como título, descrição, responsável e data de conclusão, os usuários podem se manter organizados e cumprir suas responsabilidades de maneira eficaz. A possibilidade de conclusão de tarefas diretamente na home do usuário agiliza o registro do progresso e mantém todos os envolvidos informados sobre o status das atividades. Essa abordagem contribui para uma gestão eficiente das tarefas e promove uma colaboração fluida e produtiva entre os membros da equipe.

### **5. Considerações Finais**

O trabalho proporcionou um aprimoramento substancial dos conhecimentos tecnológicos, particularmente no campo do desenvolvimento back-end. Anteriormente, o domínio era do PHP no ambiente profissional. No entanto, durante o projeto, se teve a oportunidade de explorar o mundo do JavaScript através do Node.js. Essa experiência ampliou significativamente a expertise e permitiu adquirir habilidades valiosas nessa tecnologia emergente.

O aprendizado e a aplicação prática do Node.js contribuíram para a formação profissional e capacitaram a lidar com uma variedade mais ampla de tecnologias e ferramentas. Essa expansão de conhecimentos certamente terá um impacto positivo em na carreira futura, permitindo abordar projetos mais desafiadores e diversificados.

Em suma, o presente trabalho trouxe melhorias na organização da equipe e nas práticas de gerenciamento de tarefas, além de proporcionar um aprimoramento significativo dos conhecimentos tecnológicos, especialmente no campo do desenvolvimento back-end. Esses resultados evidenciam o impacto positivo e as contribuições significativas deste trabalho para o desenvolvimento profissional e o avanço tecnológico.

### **5.1. Trabalhos Futuros**

Avaliando o sistema desenvolvido, é possível afirmar que ele se encontra funcional e disponível para uso por qualquer interessado, oferecendo a possibilidade de aprimorar a gestão de equipes. No entanto, reconhece-se que existem oportunidades para investir mais tempo no aperfeiçoamento do layout e na melhoria da experiência de visualização e utilização do sistema.

Durante o desenvolvimento do projeto, o foco principal sempre foi a praticidade e agilidade de uso, buscando oferecer aos usuários uma solução eficiente e funcional. No entanto, o aspecto do layout foi considerado como uma área a ser aprimorada no futuro.

Embora o sistema atual cumpra seu propósito de melhorar a gestão da equipe, é importante reconhecer que uma interface atraente e intuitiva pode desempenhar um papel fundamental na experiência do usuário e na adoção efetiva do sistema. Portanto, é recomendável direcionar esforços futuros para melhorar o layout e tornar a experiência de utilização ainda mais agradável e eficiente.

Aperfeiçoar o visual e a usabilidade do sistema pode resultar em uma maior adesão por parte dos usuários, além de facilitar a compreensão e o uso de suas funcionalidades. Isso contribuirá para uma melhor integração do sistema na rotina dos gestores, proporcionando um ambiente de trabalho mais eficiente e produtivo.

Em suma, embora o sistema desenvolvido esteja funcional e possa ser utilizado para melhorar a gestão da equipe, há espaço para aprimoramento no aspecto do layout e na experiência de visualização e utilização. O aperfeiçoamento desses aspectos oferecerá uma experiência mais agradável aos usuários e aumentará a eficácia e a aceitação do sistema. Portanto, é recomendável considerar essas melhorias como uma oportunidade para desenvolvimentos futuros.

### **5.2. Dificuldades Encontradas**

Durante o desenvolvimento do projeto, a equipe enfrentou diversas dificuldades, tornando a jornada um verdadeiro desafio. Além das complexidades e obstáculos encontrados na implementação do front-end, como a responsividade do sistema e a integração da dashboard, também houve a necessidade de definir permissões de acesso de acordo com o cargo de cada usuário. Além disso, aspectos além do código também demandaram atenção, como a organização do tempo em meio às demandas diárias de trabalho, a definição de metas e a busca por uma gestão eficiente de horários.

Cada uma dessas dificuldades encontradas ao longo do caminho foi um valioso aprendizado, não apenas para os membros da equipe como programadores, mas também como pessoas. A superação desses desafios contribuiu para o desenvolvimento profissional, permitindo que a equipe aprimorasse suas habilidades técnicas e explorasse soluções criativas. Além disso, essas experiências reforçaram a importância de estar sempre aberto a aprender coisas novas e enfrentar desafios com determinação.

O processo de superação de dificuldades também proporcionou uma valiosa reflexão sobre a importância da organização pessoal e do estabelecimento de metas. Lidar com o tempo limitado e enfrentar múltiplas demandas exigiu um planejamento cuidadoso e a capacidade de estabelecer prioridades. Essa habilidade de gerenciamento de tempo e organização pessoal será benéfica ao longo da trajetória profissional da equipe, ajudando-os a enfrentar desafios futuros com confiança e eficiência.

## **Referências**

- DevMidia (2012). O que é UML e Diagramas de Caso de Uso introdução prática à uml. Disponível em: <https://www.devmedia.com.br/o-que-e-uml-e-diagramas-de-caso-de-uso-introducao-pratica-a-uml/23408>. Acessado em: 03/06/2023.
- DevMidia (2014). MER e DER modelagem de bancos de dados. Disponível em: <https://www.devmedia.com.br/mer-e-der-modelagem-de-bancos-de-dados/14332>. Acessado em: 03/06/2023.
- Node (2009). Disponível em: <https://nodejs.org/en>. Acessado em: 10/07/2023.
- Postgres (1996). Disponível em: <https://www.postgresql.org/>. Acessado em: 03/07/2023.
- Prisma (2018). Disponível em: <https://www.prisma.io/>. Acessado em: 03/07/2023.
- ReactJs (2013). Disponível em: <https://pt-br.legacy.reactjs.org/docs/getting-started.html>. Acessado em: 05/06/2023.
- Stefano Pacini, Rodolpho Tobler, Viviane Seda Bittencourt (2023). Tendências do home office no brasil. Disponível em: <https://blogdoibre.fgv.br/posts/tendencias-do-home-office-no-brasil>. Acessado em: 14/07/2023.
- Supabase (2023). Disponível em: <https://supabase.com/>. Acessado em: 07/04/2023.
- TailWind (2023). Disponível em: <https://tailwindcss.com/docs/installation>. Acessado em: 03/07/2023.