

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**ANDRÉ BARBOZA DA SILVA**

**DESENVOLVIMENTO DE UM SISTEMA WEB PARA O CONTROLE DE  
TREINAMENTOS REALIZADOS POR COLABORADORES DE UMA EMPRESA**

**GUARAPUAVA**

**2024**

**ANDRÉ BARBOZA DA SILVA**

**DESENVOLVIMENTO DE UM SISTEMA WEB PARA O CONTROLE DE  
TREINAMENTOS REALIZADOS POR COLABORADORES DE UMA EMPRESA**

**DEVELOPMENT OF A WEB SYSTEM FOR THE CONTROL OF TRAININGS  
CONDUCTED BY EMPLOYEES OF A COMPANY**

Projeto de Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Tecnólogo em Tecnologia em Sistemas para Internet do Curso Superior de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Diego Marczal

Coorientadora Prof<sup>a</sup>. Dr<sup>a</sup>. Renata Luiza Stange

**GUARAPUAVA**

**2024**



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

## LISTA DE FIGURAS

|                                                                                                |           |
|------------------------------------------------------------------------------------------------|-----------|
| <b>Figura 1 – Tela de dashboard do Training System com visualização por colaborador.</b>       | <b>5</b>  |
| <b>Figura 2 – TECNICON Sistemas Gerenciais Fonte: (TECNICON, 2022).</b>                        | <b>8</b>  |
| <b>Figura 3 – Estrutura do método MoSCoW. Fonte: (SEBRAE, 2022).</b>                           | <b>10</b> |
| <b>Figura 4 – Fluxo de trabalho do GitFlow. Fonte: (PX, 2023).</b>                             | <b>12</b> |
| <b>Figura 5 – Demonstração Mkdcos. Fonte: (MkDocs, 2024a).</b>                                 | <b>13</b> |
| <b>Figura 6 – Tela de Login: interface de autenticação do usuário. Fonte: Autoria própria.</b> | <b>23</b> |
| <b>Figura 7 – Listagem de Treinamentos: visualização dos treinamentos disponíveis.</b>         |           |
| Fonte: Autoria própria.                                                                        | 24        |
| <b>Figura 8 – Opções de Conta: configurações de perfil e ajustes pessoais. Fonte:</b>          |           |
| Autoria própria.                                                                               | 25        |
| <b>Figura 9 – Tela Inicial do Administrador: painel principal de acesso e navegação.</b>       |           |
| Fonte: Autoria própria.                                                                        | 26        |
| <b>Figura 10 – Cadastro de Treinamentos: formulário para inclusão de novos treina-</b>         |           |
| mentos. Fonte: Autoria própria.                                                                | 27        |
| <b>Figura 11 – Cadastro de Colaboradores: interface para inclusão de novos usuários.</b>       |           |
| Fonte: Autoria própria.                                                                        | 28        |
| <b>Figura 12 – Listagem de Colaboradores: visualização dos registros e gerencia-</b>           |           |
| mento dos usuários. Fonte: Autoria própria.                                                    | 29        |
| <b>Figura 13 – Vinculação de Funcionários: interface para associação de colaborado-</b>        |           |
| res aos treinamentos. Fonte: Autoria própria.                                                  | 30        |

## SUMÁRIO

|            |                                                                        |           |
|------------|------------------------------------------------------------------------|-----------|
| <b>1</b>   | <b>INTRODUÇÃO . . . . .</b>                                            | <b>3</b>  |
| <b>1.1</b> | <b>Considerações iniciais . . . . .</b>                                | <b>3</b>  |
| <b>1.2</b> | <b>Objetivo Geral . . . . .</b>                                        | <b>4</b>  |
| 1.2.1      | Objetivos específicos . . . . .                                        | 4         |
| <b>1.3</b> | <b>Justificativa . . . . .</b>                                         | <b>4</b>  |
| <b>2</b>   | <b>TRAINING SYSTEM . . . . .</b>                                       | <b>6</b>  |
| <b>3</b>   | <b>MATERIAIS E MÉTODOS . . . . .</b>                                   | <b>7</b>  |
| <b>3.1</b> | <b>Materiais . . . . .</b>                                             | <b>7</b>  |
| <b>3.2</b> | <b>Métodos . . . . .</b>                                               | <b>8</b>  |
| 3.2.1      | Metodologia Ágil Scrum . . . . .                                       | 8         |
| 3.2.2      | Priorização com o Método MoSCoW . . . . .                              | 9         |
| 3.2.3      | Metodologia de Versionamento no GitHub: GitFlow . . . . .              | 10        |
| 3.2.4      | Documentação com MkDocs . . . . .                                      | 11        |
| 3.2.5      | Etapas de Desenvolvimento . . . . .                                    | 12        |
| <b>4</b>   | <b>ANÁLISE E PROJETO DO SISTEMA . . . . .</b>                          | <b>14</b> |
| <b>4.1</b> | <b>Requisitos funcionais . . . . .</b>                                 | <b>14</b> |
| <b>4.2</b> | <b>Histórias de Usuário . . . . .</b>                                  | <b>16</b> |
| 4.2.1      | Usuário Administrador e Colaborador ( <i>Must e Should</i> ) . . . . . | 17        |
| <b>4.3</b> | <b>Protótipo das Telas . . . . .</b>                                   | <b>23</b> |
| 4.3.1      | Área do Colaborador . . . . .                                          | 23        |
| 4.3.2      | Área do Administrador . . . . .                                        | 25        |
| <b>5</b>   | <b>CONSIDERAÇÕES FINAIS . . . . .</b>                                  | <b>31</b> |
|            | <b>REFERÊNCIAS . . . . .</b>                                           | <b>32</b> |

## 1 INTRODUÇÃO

Este capítulo visa a apresentação do Training System, um sistema cujo objetivo é supervisionar os treinamentos e capacitações realizados por colaboradores de uma empresa.

### 1.1 Considerações iniciais

O investimento em treinamentos é essencial para garantir a segurança e a eficiência no ambiente de trabalho. Segundo a Associação Brasileira de Treinamento e Desenvolvimento, 94% das empresas brasileiras já investem em capacitação para seus colaboradores (AL1, 2024). Entretanto, apesar desse esforço, falhas na organização e no acompanhamento desses treinamentos ainda geram consequências graves, como acidentes de trabalho e baixa produtividade. Em alguns casos, embora os colaboradores estejam capacitados para realizar determinadas atividades, problemas relacionados ao controle inadequado dos treinamentos realizados pelos colaboradores geram efeitos negativos para as empresas.

Um caso real presenciado durante uma experiência de trabalho, ocorreu em uma empresa onde um colaborador sofreu um acidente que resultou na amputação de três dedos. Embora tivesse recebido treinamento, a ausência de um sistema eficiente de controle impossibilitou, inicialmente, a comprovação de sua capacitação. O processo de checagem foi lento e complicado devido ao armazenamento desestruturado de documentos físicos em um arquivo convencional. Além disso, a falta de controle sobre atualizações de treinamentos gerava inconsistências que comprometiam a segurança e as operações.

Essa situação reflete a necessidade de ferramentas adequadas para gerenciar as informações sobre os treinamentos realizados pelos colaboradores de uma empresa. Planilhas e documentos físicos são insuficientes para atender às demandas de um ambiente de trabalho moderno, onde a conformidade regulatória e a segurança dos processos são indispensáveis.

Neste sentido, o Training System foi idealizado para suprir essa lacuna. Trata-se de um sistema que organiza e automatiza a gestão de treinamentos realizados pelos colaboradores, permitindo às empresas: a) monitorar o histórico de capacitações, identificando rapidamente quem está apto para desempenhar atividades específicas; b) receber alertas automáticos sobre a necessidade de atualizações ou treinamentos; c) facilitar auditorias e consultas, com acesso rápido a informações organizadas digitalmente.

Além disso, o Training System foca nas necessidades das empresas, oferecendo ferramentas para garantir o cumprimento das normas de segurança e a conformidade com legislações, como a NR-12, que regula o trabalho com máquinas e equipamentos. Desta forma, além de auxiliar na redução dos riscos de acidentes, o sistema otimiza o processo de verificação dos treinamentos, melhora a organização documental e contribui para a tomada de decisões estratégicas. O resultado é uma gestão mais eficiente e segura, permitindo que as empresas se concentrem em seus objetivos principais sem comprometer a segurança no ambiente de traba-

lho. Assim, o Training System poderá proporcionar para as organizações um maior controle e eficiência na gestão de seus processos de capacitação.

## 1.2 Objetivo Geral

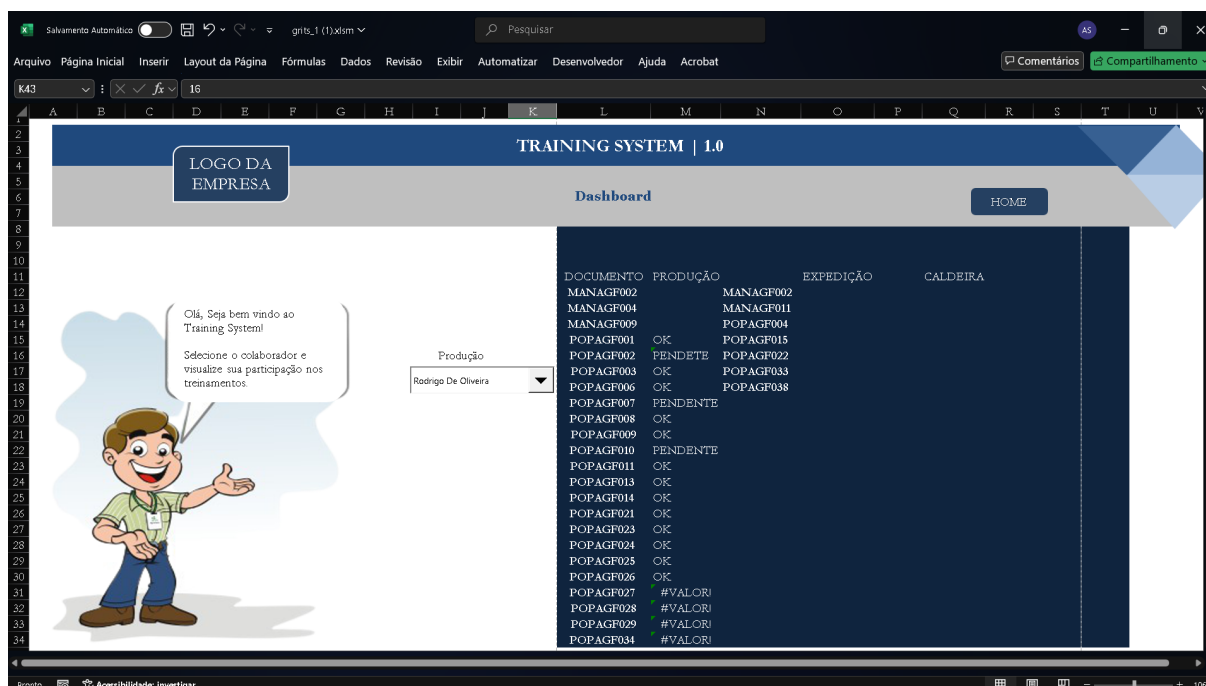
Desenvolver um sistema web para o controle dos treinamentos realizados por colaboradores de empresas, permitindo a identificação e acompanhamento de seu status.

### 1.2.1 Objetivos específicos

- Levantar e analisar os requisitos necessários para o desenvolvimento do sistema, considerando as necessidades do controle de treinamentos e as normativas aplicáveis.
- Projetar e desenvolver uma interface web que permita o acesso de administradores e colaboradores, com autenticação e funcionalidades adequadas para cada perfil.
- Implementar funcionalidades para cadastro, gerenciamento e vinculação de colaboradores e treinamentos, centralizando as informações em um único ambiente.
- Criar recursos para controle do status dos treinamentos, possibilitando o acompanhamento do histórico e o registro das ações realizadas no sistema.
- Desenvolver mecanismos para auxiliar em visualizações analíticas, além da documentação do projeto, facilitando a compreensão e manutenção do sistema.

## 1.3 Justificativa

A ideia para o desenvolvimento do Training System surgiu a partir de uma necessidade real observada no ambiente de trabalho, após um eventual acidente, e da dificuldade de acesso e organização dos registros de treinamentos dos colaboradores. A ausência de um sistema estruturado para armazenar e gerenciar essas informações pode resultar em atrasos na verificação dos treinamentos e impactar a tomada de decisões em situações que exigem agilidade. Além da organização dos registros, a proposta do sistema também considera a relevância da prevenção no ambiente corporativo. Um acompanhamento mais estruturado dos treinamentos pode contribuir para um controle mais eficiente da capacitação dos funcionários, auxiliando as empresas na adequação às normas de segurança e aos procedimentos internos. Inicialmente, a necessidade foi abordada por meio de planilhas para estruturar os dados e facilitar a gestão das informações. Agora, a evolução para um sistema web visa aprimorar essa abordagem, trazendo uma solução mais acessível e integrada. A Figura 1 exemplifica a tela de *Dashboard* da planilha, onde é possível visualizar a área de atuação do colaborador, consultar os treinamentos associados à ele e acompanhar o status atual de cada treinamento.



**Figura 1 – Tela de dashboard do Training System com visualização por colaborador.**

A partir da implementação dessa planilha, a empresa conseguiu maior agilidade na busca por informações em comparação com os documentos físicos. Contudo, o uso das planilhas também revelou novas limitações. Entre os principais desafios estavam: a) Falta de praticidade no gerenciamento completo dos dados; b) Impossibilidade de registrar acessos ou alterações feitas nos históricos; c) Dificuldade em associar diretamente os documentos aos treinamentos realizados e confirmar presenças. Esses problemas geraram insegurança durante as auditorias, pois não havia como garantir a rastreabilidade das alterações ou comprovar a validade das informações registradas.

A justificativa para o desenvolvimento de um sistema web de controle de treinamentos realizados por colaboradores de uma empresa está baseada na necessidade de otimizar e centralizar a gestão de capacitações e competências dos colaboradores. Esse tipo de sistema deve auxiliar na supervisão eficaz dos treinamentos, permitindo que a empresa mantenha um registro confiável das qualificações de seus colaboradores e identifique facilmente a falta de treinamentos e capacitações que precisam ser realizados para atender às demandas da empresa.

## 2 TRAINING SYSTEM

O Training System é proposto como sistema para controlar os treinamentos realizados por colaboradores das empresas, oferecendo controles centralizados, conformidades legais, redução de erros manuais, assistência em auditorias e relatórios. O sistema irá trabalhar seguindo um fluxo determinado, que se estabelece no momento do cadastro de colaboradores até o resultado das conclusões dos treinamentos, e com isso convém apresentar as seguintes funcionalidades do sistema.

- Controle centralizado: será possível armazenar todas as informações sobre os treinamentos realizados pelos colaboradores, incluindo datas, documentos utilizados, quem participou e o instrutor responsável por conduzir o conteúdo. Permitindo assim, que quando houver a necessidade de alguma informação, seja possível acessar o histórico e encontrar o que necessita de modo completo.
- Conformidade com as normas: o sistema terá funções de alertas, o qual verificará o prazo de validade dos documentos de treinamentos e disparará alertas quando esses estiverem próximos das datas de vencimento, permitindo que assim não possua documentos com datas vencidas e cumpra com as regulamentações definidas pelo Ministério do Trabalho e Emprego.
- Relatórios: recurso de relatórios onde será possível a gestão obter informações rápidas, como identificar áreas ou equipes que necessitam de mais treinamentos, auxiliando ainda nas tomadas de decisões.
- Auditorias: o sistema poderá ser utilizado como uma fonte para revisões internas pelos administradores. Além de ser um método para comprovação dos treinamentos, e conformidade com o fluxo de organização da empresa.

Ainda nas funcionalidades e benefícios do sistema, pode se destacar que ele poderá contribuir com uma redução a erros manuais, pois uma organização feita em planilhas, documentos físicos ou em softwares não apropriados, é possível que ocorra omissões de datas importantes. Por exemplo, se alguém não autorizado apagar informações ou ainda fazer alterações em dados relevantes, não é possível averiguar quem foi o responsável. Além disso, pode ocorrer perdas de informações e o principal, a falta de organização. Espera-se que com o sistema, esses erros sejam evitados, assegurando que os dados sejam registrados corretamente e que os prazos de treinamentos e requisitos para desempenhar cada função na empresa sejam atendidos.

### 3 MATERIAIS E MÉTODOS

Este capítulo descreve os materiais, ferramentas e metodologias que serão aplicadas durante o desenvolvimento do *Training System*, um sistema proposto para o controle de treinamentos realizados por colaboradores.

#### 3.1 Materiais

Os materiais incluem ferramentas, tecnologias e ambientes de desenvolvimento que serão utilizados para uma maior eficiência, organização e qualidade do projeto. A seguir, estão listados e descritos os principais recursos que serão empregados:

- **Ruby on Rails:** O Framework Ruby on Rails será escolhido como a principal tecnologia para o desenvolvimento do sistema web devido à sua agilidade e capacidade de simplificar a construção de aplicações robustas e escaláveis. Rails segue o padrão MVC (*Model-View-Controller*), que organiza o código em camadas, facilitando a manutenção e expansão do sistema (Ruby on Rails, 2024).
- **SQLite:** O banco de dados SQLite será selecionado para o projeto, pois atenderá às necessidades de armazenamento de forma leve e prática. Ele será ideal para aplicações de pequeno a médio porte, possibilitando o gerenciamento eficiente de informações de colaboradores, treinamentos e registros históricos (SQLite, 2024).
- **Docker:** O uso do Docker permitirá a criação de ambientes isolados e padronizados, eliminando problemas de dependências entre sistemas. A aplicação será containerizada com Docker, permitindo que funcione de maneira idêntica em diferentes ambientes de desenvolvimento e produção (Docker Inc., 2024).
- **Figma:** O Figma será utilizado para a prototipação da interface do sistema. A ferramenta possibilitará a criação de protótipos interativos, que serão validados antes da implementação, podendo oferecer uma experiência de usuário eficiente e alinhada às expectativas (Figma, 2024).
- **GitHub:** O repositório do projeto será hospedado no GitHub para controle de versão. A utilização do Git permitirá o rastreamento das alterações no código-fonte, facilitando o gerenciamento do desenvolvimento e mantendo a segurança do histórico do projeto (GitHub, 2024).
- **ClickUp:** O ClickUp será empregado para a gestão de tarefas e cronograma do projeto, utilizando práticas da metodologia ágil Scrum. As tarefas serão organizadas em Sprints, com a definição de prazos e prioridades, permitindo um acompanhamento eficiente do progresso (ClickUp, 2024).

- **MkDocs:** Para a documentação do projeto, será utilizado o MkDocs, uma ferramenta para a criação de documentação estruturada e de fácil navegação. O MkDocs permite a geração de páginas estáticas a partir de arquivos em Markdown, possibilitando uma organização clara e acessível da documentação técnica do sistema (MkDocs, 2024b).

## 3.2 Métodos

Os métodos a serem aplicados no desenvolvimento do *Training System* seguirão a metodologia ágil Scrum, que permitirá a entrega incremental e iterativa do sistema. O desenvolvimento será estruturado em etapas, conforme descrito a seguir.

### 3.2.1 Metodologia Ágil Scrum

Scrum é uma abordagem revolucionária para gerenciamento de projetos e formação de equipes, que tem ajudado a transformar desde empresas de software até o exército dos Estados Unidos e o setor de saúde em grandes hospitais (SUTHERLAND, 2014).

O Scrum é sustentado por três pilares fundamentais: transparência, inspeção e adaptação. A transparência assegura que todos os aspectos significativos do processo sejam visíveis para aqueles que são responsáveis pelo resultado, promovendo uma compreensão compartilhada. A inspeção envolve a avaliação contínua dos artefatos e do progresso em direção aos objetivos, permitindo a identificação precoce de desvios ou problemas. A adaptação refere-se à capacidade de ajustar processos e planos com base nas descobertas da inspeção, garantindo que a equipe possa responder eficazmente às mudanças e desafios que surgem durante o projeto (Atlassian, 2025).

A Figura 2 ilustra a estrutura do geral do Scrum.

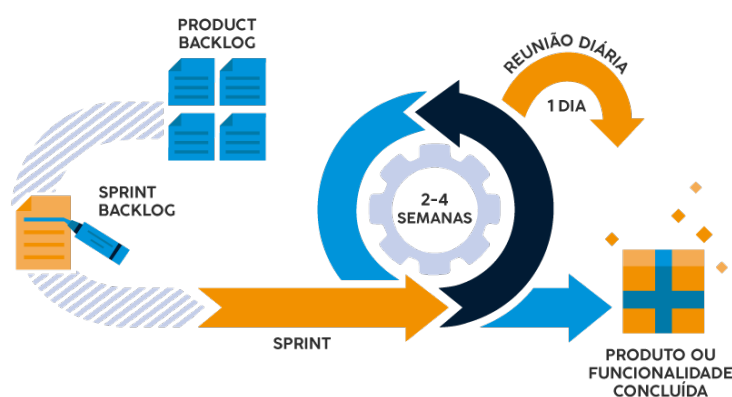


Figura 2 – TECNICON Sistemas Gerenciais Fonte: (TECNICON, 2022).

Baseado na eficácia dos projetos desenvolvidos com metodologias ágeis e na praticidade do *Scrum*, esta abordagem será utilizada para o controle das *Sprints* no desenvolvimento do *Training System*.

O desenvolvimento será estruturado por meio de ciclos iterativos chamados *Sprints*, com duração de duas semanas. No início de cada *Sprint*, será realizado um *Sprint Planning* para definir as tarefas prioritárias, baseadas no *Backlog do Produto*. Durante o ciclo, haverá um acompanhamento contínuo do progresso, podendo ser registrado por meio de anotações ou relatórios diários, substituindo as reuniões diárias (*Daily Scrum*).

Ao final de cada *Sprint*, será realizada uma *Sprint Review* para validação das funcionalidades desenvolvidas, seguida de uma *Sprint Retrospective*, na qual serão analisados pontos de melhoria no processo. As reuniões de abertura e fechamento das *Sprints* serão conduzidas com o *Product Owner*, que, neste contexto, será representado pelo orientador do projeto.

Além disso, será adotada uma *Definição de Pronto (DoD – Definition of Done)* para garantir que cada funcionalidade atenda aos critérios estabelecidos antes de ser considerada concluída. Como único desenvolvedor, o papel do *Scrum Master* será incorporado à rotina de trabalho, garantindo a organização e adaptação do processo conforme necessário.

### 3.2.2 Priorização com o Método MoSCoW

O método de priorização MoSCoW é utilizado para determinar a importância de requisitos no projeto. Foi desenvolvido por Dai Clegg, especialista em desenvolvimento de sistemas na Oracle, durante a década de 1990. Originalmente aplicado no desenvolvimento de softwares com prazos bem definidos, esse método expandiu-se para diversas áreas devido à sua flexibilidade e eficiência na tomada de decisões (SEBRAE, 2022). O nome MoSCoW é um acrônimo formado pelas iniciais das quatro categorias <sup>1</sup> que classificam os requisitos do projeto:

A Figura 3 ilustra a estrutura do método:

---

<sup>1</sup> A letra "o" presente no acrônimo não representa nenhuma categoria, sendo inserida apenas para facilitar a pronúncia da palavra.



Figura 3 – Estrutura do método MoSCoW. Fonte: (SEBRAE, 2022).

A aplicação do método MoSCoW é especialmente recomendada nas fases iniciais de um projeto, pois permite definir claramente o que deve ser priorizado. No entanto, sua utilização também pode ser estendida a fases posteriores, auxiliando em tomadas de decisão ao longo do desenvolvimento.

Para aplicar a técnica de forma eficiente, alguns passos são recomendados:

1. **Definir a lista de requisitos:** Identificar e documentar todos os itens que serão analisados e priorizados.
2. **Formar uma equipe multidisciplinar:** Reunir representantes de diferentes áreas para garantir uma visão abrangente do projeto.
3. **Estabelecer um mediador:** Designar uma pessoa responsável por conduzir discussões e resolver eventuais impasses durante o processo de priorização.
4. **Analisar os recursos disponíveis:** Apresentar as limitações orçamentárias, de tempo e tecnologia para orientar a definição de prioridades.
5. **Definir um prazo para a priorização:** Estabelecer um período para que a equipe finalize a análise e classificação dos requisitos.

A metodologia MoSCoW foi adotada para a priorização dos requisitos do *Training System* devido à sua abordagem objetiva e estruturada. Em combinação com o Framework Scrum, essa técnica auxilia a equipe de desenvolvimento na definição de funcionalidades essenciais, garantindo que os elementos mais críticos sejam implementados nas primeiras iterações.

### 3.2.3 Metodologia de Versionamento no GitHub: GitFlow

O Git Flow é uma estratégia de ramificação no Git que organiza o desenvolvimento de software em diferentes tipos de branches, cada uma com um propósito específico, facilitando

o controle de versões e a colaboração entre equipes. Segundo (DRIESSEN, Vincent, 2010), o Git Flow foi proposto por Vincent Driessen em 2010 e é especialmente útil para projetos que seguem um ciclo de lançamento estruturado.

A estrutura do Git Flow é composta por:

- Branches principais (permanentes):
  - **Main (ou Master)**: Contém o código de produção estável, representando o histórico oficial do projeto.
  - **Develop**: Serve como uma ramificação de integração para recursos, onde novas funcionalidades são incorporadas antes de serem lançadas.
- Branches de suporte (temporárias):
  - **Feature**: Derivadas de *develop*, são utilizadas para o desenvolvimento de novas funcionalidades. Após a conclusão, são mescladas de volta à *develop*.
  - **Release**: Criadas a partir de *develop* quando o projeto está próximo de um lançamento. Permitem ajustes finais e, após a finalização, são mescladas em *main* e *develop*.
  - **Hotfix**: Originadas de *main*, são destinadas a correções urgentes em produção. Após a correção, são mescladas tanto em *main* quanto em *develop*.

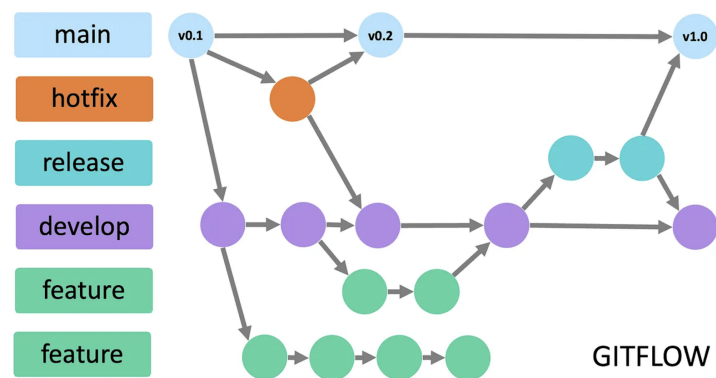
No desenvolvimento do *Training System*, o Git Flow será implementado para gerenciar o ciclo de vida do software. As *branches* **Main** e **Develop** serão mantidas permanentemente, enquanto as *branches* de suporte serão criadas conforme necessário para o desenvolvimento de novas funcionalidades, preparações para lançamentos e correções emergenciais. Essa abordagem estruturada garantirá um fluxo de trabalho organizado e eficiente, facilitando a gestão de versões e a colaboração entre os desenvolvedores.

Para auxiliar na implementação do Git Flow, existem ferramentas que automatizam a criação e o gerenciamento das *branches* conforme essa estratégia. Uma dessas ferramentas é a extensão de linha de comando `git-flow`, que simplifica o uso desse modelo no dia a dia do desenvolvimento (DRIESSEN, Vincent, 2010).

A Figura 4 ilustra a estrutura do fluxo de trabalho com *branches* no GitHub.

### 3.2.4 Documentação com MkDocs

Para a documentação do *Training System*, foi adotada a ferramenta MkDocs, uma ferramenta eficiente para criar documentações estáticas a partir de arquivos Markdown. Essa abordagem possibilita a geração rápida e organizada da documentação do projeto, facilitando a manutenção e acessibilidade das informações. O MkDocs foi desenvolvido inicialmente por Tom



**Figura 4 – Fluxo de trabalho do GitFlow. Fonte: (PX, 2023).**

Christie, Dougal Matthews, Waylan Limberg, Oleh Prypin e Ultrabug, com sua primeira versão lançada em 24 de janeiro de 2014 (MkDocs, 2024a).

A utilização do MkDocs no *Training System* permitirá documentar funcionalidades específicas e descrever métodos que executam ações complexas dentro do sistema. Além disso, será empregado para adicionar regras que garantam a integridade das operações, impedindo a execução de processos caso determinadas condições não sejam atendidas.

Outra vantagem do uso do MkDocs é a inclusão de soluções rápidas para o gerenciamento de erros, funcionando como uma base de conhecimento geral dentro do sistema. Isso possibilitará que desenvolvedores e usuários acessem informações detalhadas sobre falhas comuns e suas resoluções de forma ágil e eficiente.

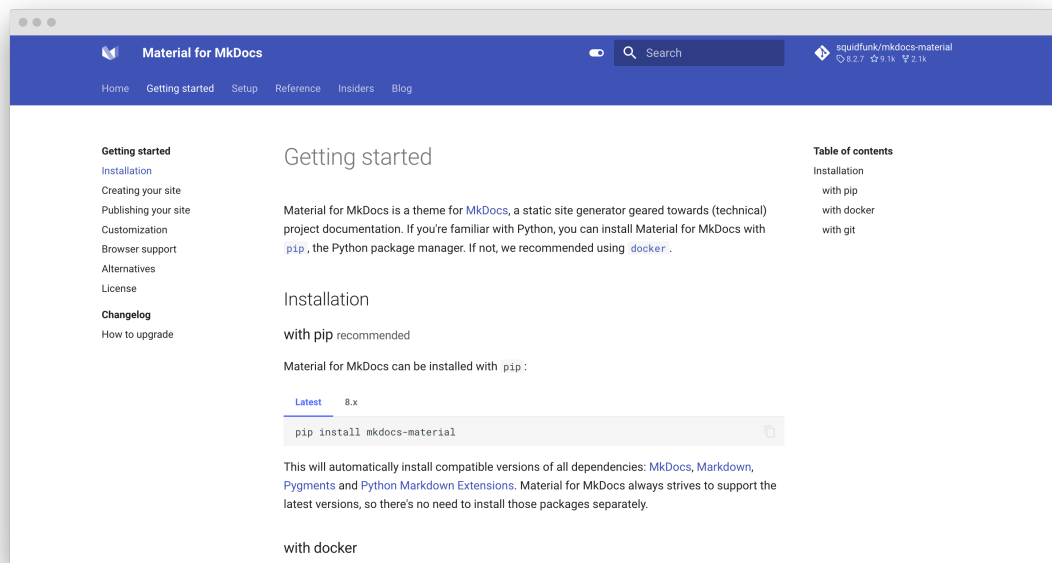
Embora o MkDocs seja uma ferramenta desenvolvida em Python, ele permite a atualização contínua da documentação por meio de um sistema em nuvem. Dessa forma, basta salvar novos arquivos e executar a ferramenta para que as alterações sejam refletidas automaticamente. Além disso, o MkDocs pode ser integrado ao GitHub, facilitando o versionamento e a colaboração no desenvolvimento da documentação.

A Figura 5 ilustra brevemente a tela da documentação do mkdocs *branches* no GitHub.

### 3.2.5 Etapas de Desenvolvimento

O desenvolvimento do sistema será dividido em etapas claras e estruturadas:

- **Levantamento de Requisitos:** A identificação das necessidades do sistema foi baseada em dificuldades previamente observadas no gerenciamento de treinamentos em empresas de casos reais vivenciados e no design das telas prototipadas. Essas referências serviram para definir funcionalidades iniciais e estruturar os fluxos do sistema.
- **Prototipação:** O design da interface será criado no Figma, com foco em proporcionar uma experiência de usuário intuitiva. Os protótipos serão validados antes da implementação, garantindo sua adequação às necessidades dos usuários.



**Figura 5 – Demonstração Mkdocs. Fonte: (MkDocs, 2024a).**

- **Configuração do Ambiente de Desenvolvimento:** A aplicação será desenvolvida em um ambiente containerizado utilizando Docker. As dependências do Ruby on Rails e do SQLite serão configuradas para assegurar a consistência entre os ambientes.
- **Implementação:** Com base nos requisitos levantados, será desenvolvida a estrutura do sistema, incluindo a lógica de funcionamento e a interface de usuário. Essa etapa envolve a implementação do backend e frontend, garantindo a comunicação entre os módulos e a aplicação das regras de negócio definidas.

## 4 ANÁLISE E PROJETO DO SISTEMA

Este capítulo apresenta os principais requisitos do desenvolvimento do sistema de gerenciamento de treinamentos. O objetivo é proporcionar uma compreensão clara das suas funcionalidades e adequação aos requisitos levantados.

### 4.1 Requisitos funcionais

Nesta seção, são detalhados os principais requisitos funcionais do sistema. Cada requisito foi classificado utilizando o método para priorização de requisitos MoSCoW.

As tabelas 1, 2 e 3 apresentam, respectivamente, os requisitos gerais (Usuário Administrador e Colaborador), os requisitos destinados especificamente ao Usuário Colaborador e aqueles destinados ao Usuário Administrador. Esses levantamentos estruturam as principais funcionalidades para cada perfil de usuário, classificadas pela técnica MoSCoW (Must, Should, Could e Won't).

**Tabela 1 – Requisitos gerais do sistema (Usuário Administrador e Colaborador).**

Fonte: Autoria própria.

| Requisito                       | Descrição detalhada                                                                                                                                              | MoSCoW |
|---------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| Tela Inicial                    | Exibir uma tela inicial contendo tutoriais e manuais de uso para facilitar a navegação dos usuários. Essa tela será acessível para todos os usuários do sistema. | Must   |
| Autenticação de usuário         | Validar o tipo de usuário (colaborador ou administrador) e direcionar o acesso às funcionalidades apropriadas.                                                   | Must   |
| Guarda de rotas                 | Restringir o acesso a áreas específicas do sistema, garantindo que apenas usuários autenticados possam acessá-las.                                               | Must   |
| Notificações e alertas          | Enviar lembretes automáticos sobre treinamentos pendentes, prazos de vencimento e novas inscrições.                                                              | Should |
| Integração com outros sistemas  | Permitir integração com sistemas de RH e ERPs para sincronização de dados.                                                                                       | Won't  |
| Mobile-friendly                 | Garantir que o sistema seja responsivo e funcione corretamente em dispositivos móveis.                                                                           | Won't  |
| Assinatura digital              | Implementar um sistema de assinatura digital para validar registros de conclusão de treinamento.                                                                 | Won't  |
| Reconhecimento facial           | Validar a presença dos colaboradores nos treinamentos por meio de reconhecimento facial.                                                                         | Won't  |
| Suporte multilíngue             | Disponibilizar o sistema em múltiplos idiomas para maior acessibilidade.                                                                                         | Won't  |
| Filtro avançado de treinamentos | Permitir busca detalhada por treinamentos aplicando filtros como colaborador, data, instrutor e status.                                                          | Won't  |

**Tabela 2 – Requisitos para colaboradores (Usuário Colaborador).****Fonte: Autoria própria.**

| <b>Requisito</b>                            | <b>Descrição detalhada</b>                                                                                                                     | <b>MoSCoW</b> |
|---------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|---------------|
| Listagem de treinamentos para colaboradores | Permitir que os colaboradores visualizem todos os treinamentos disponíveis, destacando o status (pendente, em andamento, concluído).           | Must          |
| Detalhes do treinamento                     | Exibir informações completas de um treinamento, incluindo descrição, datas, instrutor responsável e status. Disponível para todos os usuários. | Must          |
| Perfil do colaborador                       | Permitir que o colaborador visualize e edite suas informações básicas, como nome, setor e cargo.                                               | Must          |
| Vínculo de colaborador e treinamento        | Associar treinamentos a colaboradores e registrar os documentos obrigatórios utilizados no processo.                                           | Must          |
| Dashboard do colaborador                    | Exibir um painel com treinamentos pendentes, concluídos e próximos vencimentos, incluindo gráficos de progresso.                               | Won't         |
| Certificação automática                     | Gerar certificados automaticamente para colaboradores que concluírem um treinamento.                                                           | Won't         |

**Tabela 3 – Requisitos para administradores (Usuário Administrador).****Fonte: Autoria própria.**

| <b>Requisito</b>                              | <b>Descrição detalhada</b>                                                                                                                                                               | <b>MoSCoW</b> |
|-----------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|
| Registros de histórico                        | Registrar todas as ações realizadas no sistema, incluindo alterações de status e edições de cadastros, para fins de auditoria. Apenas administradores poderão visualizar esse histórico. | Must          |
| Alteração de status do treinamento            | Permitir que administradores modifiquem o status de um treinamento (pendente, em andamento, concluído).                                                                                  | Must          |
| Cadastro de colaboradores                     | Permitir que administradores registrem novos colaboradores no sistema.                                                                                                                   | Must          |
| Cadastro de treinamentos                      | Permitir que administradores registrem novos treinamentos, definindo datas, instrutores e critérios de participação.                                                                     | Must          |
| Listagem de colaboradores                     | Permitir que administradores visualizem a lista de colaboradores cadastrados, aplicando filtros e ordenação por nome, setor e status.                                                    | Must          |
| Listagem de treinamentos para administradores | Permitir que administradores consultem todos os treinamentos cadastrados, com filtros por nome, data e instrutor.                                                                        | Must          |
| Editar colaboradores                          | Permitir que administradores atualizem os dados cadastrais dos colaboradores, como nome, cargo e setor.                                                                                  | Must          |
| Editar treinamentos                           | Permitir que administradores editem as informações de um treinamento, incluindo datas e participantes.                                                                                   | Must          |
| Excluir treinamentos                          | Permitir que administradores removam treinamentos, garantindo que não haja exclusão indevida de dados críticos.                                                                          | Must          |
| Excluir colaboradores                         | Permitir que administradores removam colaboradores inativos, preservando o histórico de treinamentos realizados.                                                                         | Must          |
| Controle de acesso                            | Definir perfis de acesso para administradores e colaboradores, garantindo que cada usuário tenha permissões adequadas.                                                                   | Must          |
| Perfil do administrador                       | Permitir que administradores visualizem e editem suas informações e gerenciem contas de outros usuários.                                                                                 | Must          |
| Controle de versão dos treinamentos           | Registrar e permitir a atualização das versões dos treinamentos oferecidos, garantindo que colaboradores sempre tenham acesso às versões mais recentes.                                  | Should        |
| Padronização de documentos obrigatórios       | Automatizar a atribuição de treinamentos obrigatórios para novos colaboradores.                                                                                                          | Should        |
| Suporte a auditorias                          | Permitir que administradores exportem dados para revisões internas e conformidade com normas regulatórias.                                                                               | Could         |
| Exportação de relatórios                      | Gerar relatórios em formatos como PDF e Excel para facilitar a análise e compartilhamento.                                                                                               | Could         |
| Geração de relatórios                         | Criar relatórios detalhados sobre treinamentos realizados e pendentes, com filtros avançados.                                                                                            | Won't         |
| Dashboards interativos                        | Exibir indicadores de desempenho, como taxa de conclusão e pendências dos treinamentos.                                                                                                  | Won't         |

## 4.2 Histórias de Usuário

Além dos requisitos levantados, foram elaboradas histórias de usuário com base nos itens *Must* e *Should* de cada perfil (administrador e/ou colaborador), de acordo com a priorização MoSCoW. Cada história descreve o papel do usuário, a funcionalidade desejada e o valor que ela agrega, além de critérios de aceitação no formato *Dado – Quando – Então*.

#### 4.2.1 Usuário Administrador e Colaborador (*Must* e *Should*)

Essas histórias contemplam funcionalidades básicas compartilhadas pelos perfis de administrador e colaborador. O objetivo é garantir que ambos possam acessar a tela inicial com tutoriais, efetuar login de forma segura, ter acesso restrito a funcionalidades de acordo com seu nível de permissão e receber notificações pertinentes a treinamentos e prazos.

##### Feature: Tela Inicial

**Como** usuário (administrador ou colaborador), **quero** visualizar uma tela inicial contendo tutoriais e manuais de uso **para** facilitar a navegação e o aprendizado sobre o sistema.

**Dado** que estou acessando o sistema,

**Quando** entro na página principal,

**Então** devo ver tutoriais e manuais de uso disponíveis.

##### Feature: Autenticação de usuário

**Como** usuário (administrador ou colaborador), **quero** autenticar-me no sistema **para** acessar as funcionalidades apropriadas ao meu perfil.

**Dado** que possuo credenciais válidas,

**Quando** faço login informando usuário e senha,

**Então** sou redirecionado para as funcionalidades específicas do meu perfil.

##### Feature: Guarda de rotas

**Como** usuário autenticado, **quero** que o sistema restrinja o acesso a áreas específicas **para** que somente usuários autorizados possam acessá-las.

**Dado** que estou logado como usuário,

**Quando** tento acessar uma área restrita,

**Então** o sistema verifica minhas permissões e, se não forem adequadas, nega o acesso.

#### Feature: Notificações e alertas

**Como** usuário (administrador ou colaborador), **quero** receber notificações e alertas sobre treinamentos pendentes, prazos e novas inscrições **para** não perder prazos e me manter atualizado.

**Dado** que existem treinamentos ou eventos relevantes,

**Quando** o sistema identifica que há um prazo próximo ou uma nova inscrição disponível,

**Então** devo receber um alerta ou notificação no sistema (ou via e-mail, caso implementado).

#### Usuário Administrador (*Must e Should*)

A área do usuário administrador permitirá que usuários com permissões específicas auxiliem na administração da plataforma, oferecendo suporte em atividades operacionais, monitorando a execução dos treinamentos de cada colaborador, controlando o progresso e assegurando a conformidade com os requisitos estabelecidos.

A seguir, são apresentadas as definições das telas do administrador, descritas em formato de histórias de usuário.

#### Feature: Registros de histórico

**Como** administrador, **quero** que todas as ações realizadas no sistema sejam registradas **para** fins de auditoria e acompanhamento de alterações.

**Dado** que estou autenticado como administrador,

**Quando** realizo alterações de status ou edito cadastros,

**Então** essas ações devem ser registradas em um histórico acessível somente para administradores.

#### Feature: Alteração de status do treinamento

**Como** administrador, **quero** modificar o status de um treinamento (pendente, em andamento, concluído) **para** manter o controle atualizado de cada etapa do processo.

**Dado** que existe um treinamento cadastrado,

**Quando** acesso o gerenciamento de treinamentos e seleciono uma nova opção de status,

**Então** o sistema deve atualizar o status do treinamento e registrar essa mudança no histórico.

**Feature: Cadastro de colaboradores**

**Como** administrador, **quero** cadastrar novos colaboradores **para** que eles tenham acesso ao sistema e possam participar dos treinamentos.

**Dado** que sou um administrador autenticado,

**Quando** informo os dados do colaborador (nome, e-mail, cargo, etc.),

**Então** o colaborador deve ser registrado no sistema e receber acesso de acordo com seu perfil.

**Feature: Cadastro de treinamentos**

**Como** administrador, **quero** cadastrar novos treinamentos, definindo datas, instrutores e critérios de participação **para** organizar as ofertas de capacitação.

**Dado** que estou autenticado como administrador,

**Quando** insiro os dados necessários para um novo treinamento,

**Então** o sistema deve criar o registro do treinamento e disponibilizá-lo para os colaboradores.

**Feature: Listagem de colaboradores**

**Como** administrador, **quero** visualizar todos os colaboradores cadastrados **para** aplicar filtros e acompanhar o status de cada um.

**Dado** que sou um administrador autenticado,

**Quando** acesso a lista de colaboradores,

**Então** devo ver todos os colaboradores com a opção de filtrar e ordenar por nome, setor ou status.

**Feature: Listagem de treinamentos (Administrador)**

**Como** administrador, **quero** consultar todos os treinamentos cadastrados **para** acompanhar os detalhes, datas e instrutores.

**Dado** que existem treinamentos registrados,

**Quando** acesso a área de treinamentos como administrador,

**Então** devo poder visualizar a lista completa, com filtros por nome, data e instrutor.

**Feature: Edição de colaboradores**

**Como** administrador, **quero** atualizar os dados cadastrais de colaboradores **para** manter as informações corretas e atualizadas.

**Dado** que sou administrador,

**Quando** acesso o perfil de um colaborador e edito seus dados,

**Então** as alterações devem ser salvas e refletidas em todo o sistema.

**Feature: Edição de treinamentos**

**Como** administrador, **quero** editar as informações de um treinamento (datas, participantes, etc.) **para** ajustar detalhes conforme necessário.

**Dado** que já existe um treinamento cadastrado,

**Quando** acesso a opção de edição e altero os campos desejados,

**Então** o sistema deve atualizar os dados do treinamento e registrar essa mudança.

**Feature: Exclusão de treinamentos**

**Como** administrador, **quero** excluir treinamentos que não são mais relevantes **para** manter o sistema organizado e evitar sobrecarga de informações.

**Dado** que sou administrador,

**Quando** seleciono um treinamento para exclusão,

**Então** o sistema deve remover o registro, desde que não afete dados críticos, e registrar essa ação no histórico.

**Feature: Exclusão de colaboradores**

**Como** administrador, **quero** remover colaboradores inativos **para** manter o cadastro atualizado, preservando o histórico de treinamentos realizados.

**Dado** que um colaborador está inativo,

**Quando** seleciono a opção de exclusão,

**Então** o sistema deve desativar o colaborador, mantendo os registros históricos associados.

#### Feature: Controle de acesso

**Como** administrador, **quero** definir perfis de acesso **para** garantir que cada usuário tenha permissões adequadas às suas funções.

**Dado** que possuo privilégios de administrador,

**Quando** gerencio as configurações de perfis,

**Então** posso atribuir e remover permissões de acordo com as necessidades do sistema.

#### Feature: Perfil do administrador

**Como** administrador, **quero** visualizar e editar minhas próprias informações **para** manter meus dados atualizados e gerenciar contas de outros usuários.

**Dado** que estou autenticado como administrador,

**Quando** acesso a seção de perfil,

**Então** posso alterar meus dados pessoais e gerenciar as contas de outros usuários.

#### Feature: Controle de versão dos treinamentos

**Como** administrador, **quero** registrar e atualizar as versões dos treinamentos **para** garantir que os colaboradores sempre tenham acesso ao conteúdo mais recente.

**Dado** que possuo um treinamento já cadastrado,

**Quando** faço uma atualização de conteúdo,

**Então** o sistema deve criar uma nova versão e disponibilizá-la para os colaboradores, mantendo o histórico de versões anteriores.

#### Feature: Padronização de documentos obrigatórios

**Como** administrador, **quero** automatizar a atribuição de treinamentos obrigatórios a novos colaboradores **para** garantir que todos recebam o mesmo nível de formação inicial.

**Dado** que um novo colaborador é cadastrado,

**Quando** o sistema identifica que existem treinamentos obrigatórios,

**Então** esses treinamentos devem ser vinculados automaticamente ao colaborador.

### Usuário Colaborador (*Must e Should*)

Para o perfil de colaborador, as histórias de usuário demonstram as funcionalidades essenciais que possibilitam o acesso aos treinamentos, a visualização de informações detalhadas e a gestão dos dados pessoais.

**Feature: Listagem de treinamentos (Colaborador)**

**Como** colaborador, **quero** visualizar todos os treinamentos disponíveis **para** acompanhar meu progresso e identificar quais ainda estão pendentes, em andamento ou concluídos.

**Dado** que estou autenticado como colaborador,

**Quando** acesso a lista de treinamentos,

**Então** devo ver os treinamentos com status e detalhes relevantes.

**Feature: Detalhes do treinamento**

**Como** colaborador, **quero** visualizar informações completas de um treinamento **para** compreender sua descrição, datas, instrutor responsável e status.

**Dado** que existe um treinamento cadastrado,

**Quando** seleciono um treinamento específico,

**Então** devo ver todos os detalhes, incluindo a descrição, datas, instrutor e status.

**Feature: Perfil do colaborador**

**Como** colaborador, **quero** visualizar e editar minhas informações básicas **para** manter meus dados atualizados (nome, setor e cargo).

**Dado** que estou autenticado como colaborador,

**Quando** acesso meu perfil,

**Então** posso editar minhas informações e salvar as alterações.

**Feature: Vínculo de colaborador e treinamento**

**Como** colaborador, **quero** ser associado aos treinamentos obrigatórios **para** garantir que os documentos necessários sejam registrados e que eu possa participar das capacitações.

**Dado** que fui cadastrado no sistema,

**Quando** um treinamento obrigatório é definido,

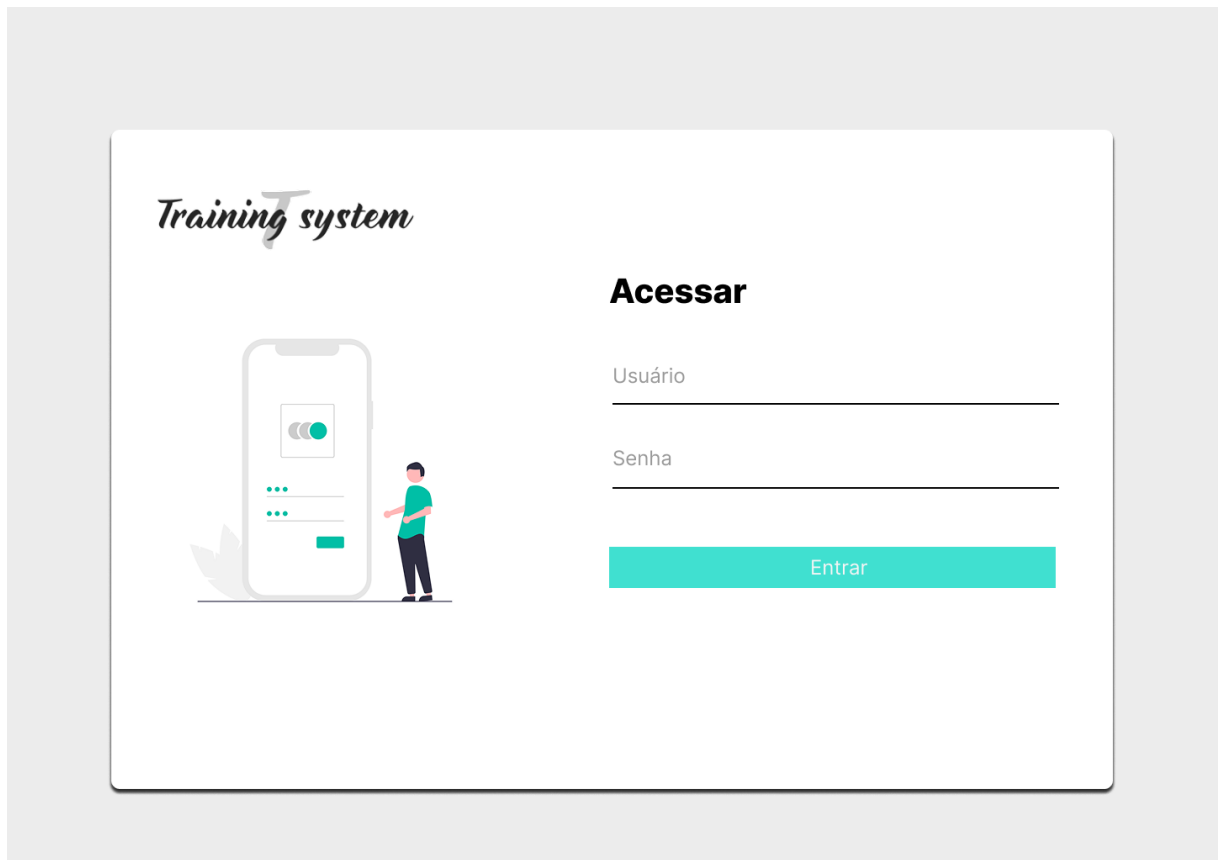
**Então** devo ser vinculado a esse treinamento e ter acesso aos documentos associados.

### 4.3 Protótipo das Telas

Nesta seção, são apresentados os protótipos das telas do sistema, acompanhados de descrições que elucidam suas funcionalidades e fluxos de navegação. As imagens e legendas serão atualizadas conforme o avanço do desenvolvimento.

#### 4.3.1 Área do Colaborador

Nesta subseção, apresenta-se a interface destinada aos colaboradores. O acesso inicial se dá por meio da tela de login, conforme exibido na Figura 6, onde o usuário informa suas credenciais para autenticação.



**Figura 6 – Tela de Login: interface de autenticação do usuário. Fonte: Autoria própria.**

A Figura 7 exibe a tela de listagem de treinamentos, onde são apresentados o nome, a categoria, o prazo de validade e as opções de edição ou remoção dos registros.





Olá, André!  
[Opções Conta >](#)

[Cadastro Treinamento](#)
[Listagem Treinamento](#)
[Cadastro Colaborador](#)
[Listagem Colaborador](#)
[Vincular](#)

### Listagem de Treinamentos

| Prazo de Validade | Treinamento                   | Categoria        |
|-------------------|-------------------------------|------------------|
| 12/03/2029        | Relacionamento Banco de Dados | Manutenção       |
| 12/03/2029        | Relacionamento Banco de Dados | Produção         |
| 12/03/2029        | Relacionamento Banco de Dados | Administrativo   |
| 12/03/2029        | Relacionamento Banco de Dados | Terceiros        |
| 12/03/2029        | Relacionamento Banco de Dados | Novo colaborador |
| 12/03/2029        | Relacionamento Banco de Dados | Infraestrutura   |
| 12/03/2029        | Relacionamento Banco de Dados | Expedição        |

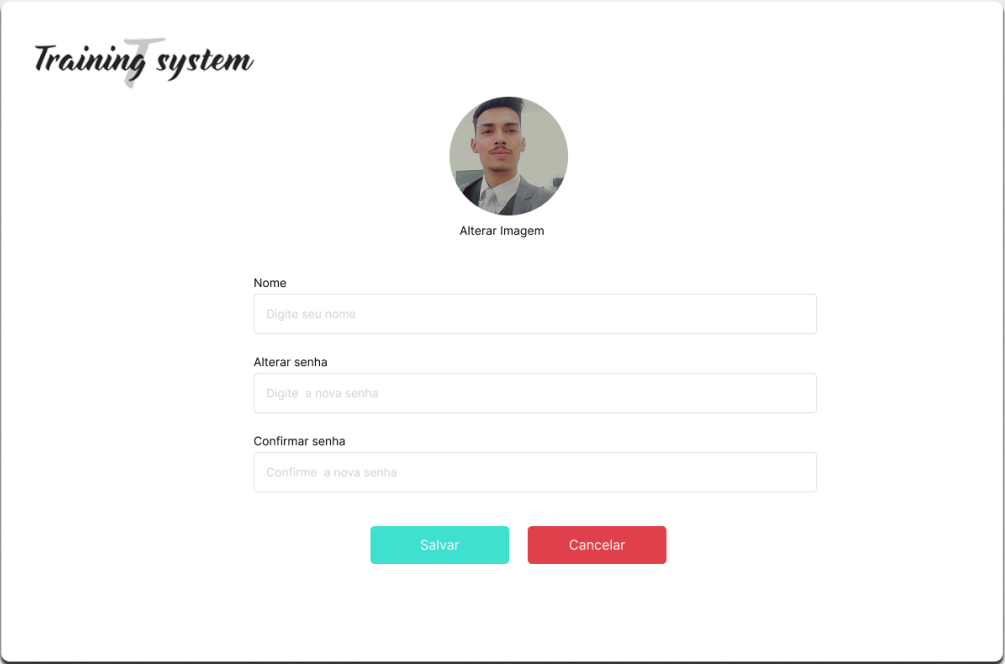
[Editar](#)
[Remover](#)

Versão 0.1  
 Data de Atualização 01/08/2023

(42)3623-5987   suporte@training.com

**Figura 7 – Listagem de Treinamentos: visualização dos treinamentos disponíveis. Fonte: Autoria própria.**

Em seguida, a Figura 8 apresenta a tela de ajustes de conta, que oferece opções de configuração de perfil acessíveis tanto para administradores quanto para colaboradores.



*Training system*



Alterar Imagem

Nome

Alterar senha

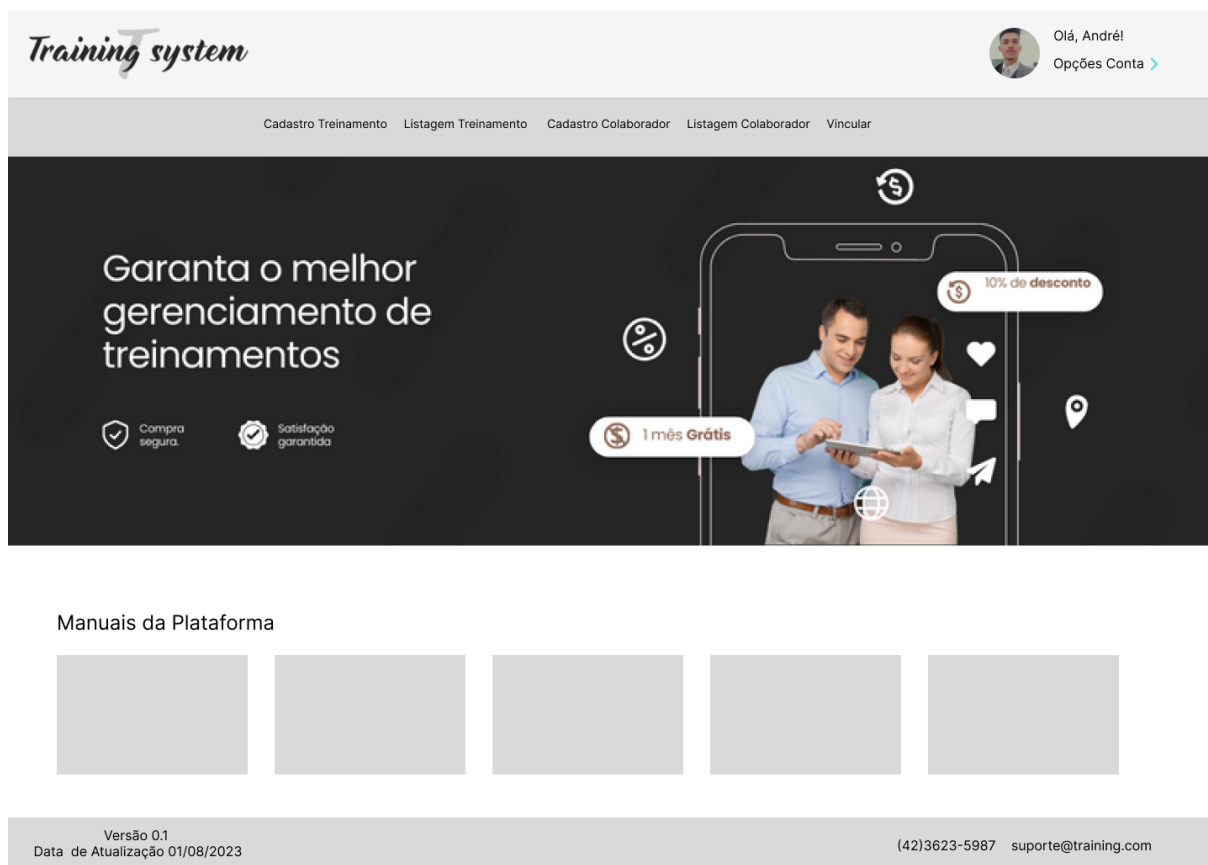
Confirmar senha

Salvar Cancelar

**Figura 8 – Opções de Conta: configurações de perfil e ajustes pessoais. Fonte: Autoria própria.**

#### 4.3.2 Área do Administrador

Nesta subseção, são detalhadas as interfaces destinadas ao administrador. Após realizar o login (inicialmente como colaborador), o administrador é redirecionado para a tela inicial do sistema, demonstrada na Figura 9.



**Figura 9 – Tela Inicial do Administrador: painel principal de acesso e navegação. Fonte: Autoria própria.**

A Figura 10 apresenta a tela de cadastro de treinamentos, que permite ao administrador registrar novos treinamentos no sistema.

**Training system**

 Olá, André!  
Opções Conta >

Cadastro Treinamento   Listagem Treinamento   Cadastro Colaborador   Listagem Colaborador   Vincular



### Cadastro Treinamento

Nome do Treinamento

Selecione a categoria

Instrutor

Prazo de Validade

SalvarCancelar

Versão 0.1  
Data de Atualização 01/08/2023

(42)3623-5987   suporte@training.com

**Figura 10 – Cadastro de Treinamentos: formulário para inclusão de novos treinamentos. Fonte: Autoria própria.**

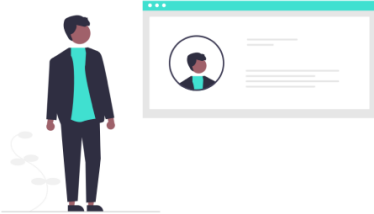
Em seguida, na Figura 11, exibe-se a tela de cadastro de colaboradores, destinada ao registro de novos usuários.

**Training system**

 Olá, André!  
Opções Conta >

Cadastro Treinamento   Listagem Treinamento   Cadastro Colaborador   Listagem Colaborador   Vincular

### Cadastro Colaborador



**Usuário de acesso**

**Nome do Colaborador**

**Senha**

**Confirmar Senha**

Salvar

Cancelar

Versão 0.1  
Data de Atualização 01/08/2023

(42)3623-5987   suporte@training.com

**Figura 11 – Cadastro de Colaboradores: interface para inclusão de novos usuários. Fonte: Autoria própria.**

A Figura 12 mostra a tela de listagem de colaboradores, que possibilita a visualização e o gerenciamento dos registros dos usuários cadastrados.





Olá, André!  
[Opções Conta >](#)

Cadastro Treinamento   Listagem Treinamento   Cadastro Colaborador   Listagem Colaborador   Vincular

**Listagem de Colaborador**

| Nome do Colaborador    |
|------------------------|
| André Barboza da Silva |
| André Barboza da Silva |
| André Barboza da Silva |
| André Barboza da Silva |
| André Barboza da Silva |
| André Barboza da Silva |
| André Barboza da Silva |

Editar

Remover

Versão 0.1  
Data de Atualização 01/08/2023

(42)3623-5987   suporte@training.com

**Figura 12 – Listagem de Colaboradores: visualização dos registros e gerenciamento dos usuários. Fonte: Autoria própria.**

Por fim, a Figura 13 ilustra a interface de vinculação de funcionários a treinamentos, permitindo a associação dos colaboradores aos treinamentos correspondentes.

**Training system**



Olá, André!  
[Opções Conta >](#)

Cadastro TreinamentoListagem TreinamentoCadastro ColaboradorListagem ColaboradorVincular

### Vincular colaborador

Nome do Colaborador

Selecione o colaborador

Nome do Treinamento

Selecione o treinamento

Vincular

|                        |                               |                                                                                     |          |                                                                                     |
|------------------------|-------------------------------|-------------------------------------------------------------------------------------|----------|-------------------------------------------------------------------------------------|
| André Barboza da Silva | Relacionamento Banco de Dados |  | Pendente |  |
| André Barboza da Silva | Relacionamento Banco de Dados |  | Pendente |  |
| André Barboza da Silva | Relacionamento Banco de Dados |  | Pendente |  |

Versão 0.1  
Data de Atualização 01/08/2023

(42)3623-5987   suporte@training.com

**Figura 13 – Vinculação de Funcionários: interface para associação de colaboradores aos treinamentos. Fonte: Autoria própria.**

## 5 CONSIDERAÇÕES FINAIS

O desenvolvimento do Training System consistem em oferecer uma solução estruturada para o controle de treinamentos de colaboradores em empresas. Durante a elaboração do projeto, foram definidas metodologias e ferramentas para orientar o processo de desenvolvimento, com o objetivo de criar um fluxo de trabalho organizado e adaptável às necessidades do sistema.

A metodologia ágil com Scrum foi escolhida para estruturar o desenvolvimento em ciclos iterativos, promovendo entregas incrementais e revisões constantes. A priorização dos requisitos foi realizada por meio do método MoSCoW, permitindo um direcionamento claro sobre quais funcionalidades são essenciais para atender às demandas do sistema. Além disso, a estratégia de versionamento baseada no GitFlow foi planejada visando um controle eficiente do código, enquanto a documentação com MkDocs contribuirá para a organização e acessibilidade das informações técnicas do projeto.

A proposta do Training System busca minimizar dificuldades relacionadas à organização e consulta de treinamentos, proporcionando um controle mais eficiente sobre a capacitação dos colaboradores. Com a centralização dos registros e a estruturação das informações, o sistema poderá auxiliar empresas no acompanhamento do histórico de treinamentos e no cumprimento de normas de segurança e regulamentações internas.

Dessa forma, este projeto estabelece as bases para o desenvolvimento de uma ferramenta que possa atender às necessidades das empresas no gerenciamento de treinamentos. Com isso, espera-se que o sistema possa contribuir para um melhor planejamento das empresas ao proporcionar uma visão mais rápida e detalhada desses dados.

## REFERÊNCIAS

- AL1. **Pesquisa mostra que 94% das empresas brasileiras investem no treinamento de funcionários.** 2024. Acesso em: 23 out. 2024. Disponível em: <https://al1.com.br/informacao/noticias/131133/pesquisa-mostra-que-94-das-empresas-brasileiras-investem-no-treinamento-de-funcionarios>.
- Atlassian. **Três pilares do Scrum: compreensão dos princípios fundamentais do Scrum.** 2025. Acesso em: 12 fev. 2025. Disponível em: <https://www.atlassian.com/br/agile/project-management/3-pillars-scrum>.
- ClickUp. **ClickUp: One app to replace them all.** 2024. Acesso em: 12 fev. 2025. Disponível em: <https://www.clickup.com/>.
- Docker Inc. **Docker: Accelerate How You Build, Share, and Run Applications.** 2024. Acesso em: 12 fev. 2025. Disponível em: <https://www.docker.com/>.
- DRIESSEN, Vincent. **A successful Git branching model.** 2010. Acesso em: 12 fev. 2025. Disponível em: <https://nvie.com/posts/a-successful-git-branching-model/>.
- Figma. **Figma: The Collaborative Interface Design Tool.** 2024. Acesso em: 12 fev. 2025. Disponível em: <https://www.figma.com/>.
- GitHub. **GitHub: Where the World Builds Software.** 2024. Acesso em: 12 fev. 2025. Disponível em: <https://github.com/>.
- MkDocs. **MkDocs: Project Documentation with Markdown.** 2024. Acesso em: 12 fev. 2025. Disponível em: <https://www.mkdocs.org/>.
- MkDocs. **MkDocs: Project documentation with Markdown.** 2024. Acesso em: 12 fev. 2025. Disponível em: <https://www.mkdocs.org/>.
- PX, T. **Git Flow Overview and Common Git Commands.** 2023. Acesso em: 11 fev. 2025. Disponível em: <https://dev.to/truongpx396/git-flow-overview-and-common-git-commands-1odh>.
- Ruby on Rails. **Ruby on Rails Guides.** 2024. Acesso em: 12 fev. 2025. Disponível em: <https://guides.rubyonrails.org/#main>.
- SEBRAE. **O método MoSCoW para definição de prioridades.** 2022. Acesso em: 12 fev. 2025. Disponível em: [https://sebrae.com.br/Sebrae/Portal%20Sebrae/Arquivos/ebook\\_sebrae\\_metodologia\\_moscow.pdf](https://sebrae.com.br/Sebrae/Portal%20Sebrae/Arquivos/ebook_sebrae_metodologia_moscow.pdf).
- SQLite. **SQLite: Database Engine.** 2024. Acesso em: 12 fev. 2025. Disponível em: <https://www.sqlite.org/>.
- SUTHERLAND, J. **Scrum: A Arte de Fazer o Dobro do Trabalho na Metade do Tempo.** Rio de Janeiro: Editora Sextante, 2014.
- TECNICON. **Metodologia Scrum para a gestão de processos ágeis na indústria.** 2022. Acesso em: 12 fev. 2025. Disponível em: [https://www.tecnicon.com.br/blog/411-Metodologia\\_Scrum\\_para\\_a\\_gestao\\_de\\_processos\\_ageis\\_na\\_industria](https://www.tecnicon.com.br/blog/411-Metodologia_Scrum_para_a_gestao_de_processos_ageis_na_industria).