

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

NESTOR HUGO FERREIRA CARDOZO

**REDESIGN DA INTERFACE GRÁFICA DO SISTEMA DE GESTÃO DE
TRABALHO DE CONCLUSÃO DE CURSO**

GUARAPUAVA

2025

NESTOR HUGO FERREIRA CARDOZO

**REDESIGN DA INTERFACE GRÁFICA DO SISTEMA DE GESTÃO DE
TRABALHO DE CONCLUSÃO DE CURSO**

Redesign of the Graphical Interface of the Final Project Management System

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

Coorientadores: Prof^a. Dra. Renata Luiza Stange, Prof. Me. Dênis Lucas Silva

GUARAPUAVA

2025



[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

Figura 1 – Recorte da documentação visual Material Design, desenvolvido pela Google	5
Figura 2 – Representação visual dos níveis do Atomic Design	10
Figura 3 – Exemplo de refatoração visual com base nos princípios do Refactoring UI.	11
Figura 4 – Etapas metodológicas adotadas no desenvolvimento do projeto	15
Figura 5 – Matriz de esforço e impacto	18
Figura 6 – Organização da paleta de cores do Design System do Sistema de Gestão de Trabalho de Conclusão de Curso (SGTCC), estruturada por categorias semânticas.	21
Figura 7 – Hierarquia tipográfica do Design System do SGTCC, com estilos definidos e documentados no Figma.	22
Figura 8 – Fragmento de ícones, parte do conjunto organizado no Figma para o Design System do SGTCC.	23
Figura 9 – Variações de botões organizadas como moléculas no Figma, configuradas com as cores e tipografia do Design System do SGTCC.	24

LISTA DE ABREVIATURAS E SIGLAS

Siglas

SGTCC	Sistema de Gestão de Trabalho de Conclusão de Curso
TCC	Trabalho de Conclusão de Curso
TSI	Tecnologia em Sistemas para Internet
UTFPR	Universidade Tecnológica Federal do Paraná
UX	User Experience Design

SUMÁRIO

1	INTRODUÇÃO	4
1.1	Contextualização e Levantamento do Problema	5
1.2	Objetivo Geral	6
1.3	Objetivos Específicos	6
1.4	Justificativa	7
2	REVISÃO DA LITERATURA	9
2.1	Design System	9
2.2	Atomic Design	9
2.3	Refactoring UI	10
2.4	Teste de Usabilidade	12
3	MATERIAIS E MÉTODOS	13
3.1	Materiais	13
3.2	Métodos	14
3.2.1	Identificação e Organização dos Componentes	15
3.2.2	Análise e Organização de Funcionalidades	15
3.2.3	Identificação de Problemas de Interface	16
3.2.4	Refatoração Visual e Propostas de Melhorias	17
3.2.5	Validação com Usuários	17
3.2.6	Implementação Parcial no Código	18
4	RESULTADOS PRELIMINARES	20
4.1	Átomos	20
4.1.1	Paleta de Cores	20
4.1.2	Tipografia	20
4.1.3	Ícones	22
4.2	Moléculas	23
4.3	Próximos Passos	24
5	CONSIDERAÇÕES FINAIS	25
	REFERÊNCIAS	26

1 INTRODUÇÃO

O Trabalho de Conclusão de Curso (TCC) é uma etapa fundamental na formação dos alunos dos cursos superiores de tecnologia, representando uma síntese do conhecimento adquirido ao longo da graduação. No curso de Tecnologia em Sistemas para Internet (TSI) da Universidade Tecnológica Federal do Paraná (UTFPR), essa atividade assume papel importante ao integrar teoria e prática em projetos que envolvem pesquisa, desenvolvimento e aplicação de soluções reais. Dada sua relevância acadêmica, a gestão dos TCCs, como observada na prática do curso de TSI da UTFPR, exige controle rigoroso de prazos, documentos, orientações, avaliações e registros.

Para atender a essa demanda, foi desenvolvido, em 2015, o SGTCC, uma plataforma web criada para organizar e centralizar as etapas envolvidas no acompanhamento dos TCCs do curso. Desde então, o sistema tem sido continuamente aprimorado por meio de colaborações técnicas de alunos e egressos, recebendo atualizações em funcionalidades, organização de interfaces e estrutura tecnológica. Atualmente, o SGTCC é utilizado por dezenas de alunos, professores e membros externos a cada semestre, desempenhando papel essencial na formalização dos processos relacionados ao TCC.

Apesar desses avanços, o sistema apresenta limitações no que se refere à experiência de uso e à organização visual de suas interfaces. A plataforma enfrenta dificuldades de navegação — especialmente para usuários com múltiplos papéis — e restrições na responsividade para dispositivos móveis, como a exibição de tabelas complexas. Além disso, o sistema carece de documentação visual formal, como um Design System. Por documentação visual entende-se um conjunto organizado de componentes gráficos, padrões de interface e diretrizes de uso que orientam o desenvolvimento visual de sistemas. Um exemplo amplamente conhecido é o Material Design, desenvolvido pela Google (2023), que serve como referência visual e interativa para diversas aplicações web. A ausência desse tipo de documentação dificulta a implementação de novas funcionalidades e compromete a escalabilidade do projeto (FROST, 2016), impactando diretamente a consistência visual entre as telas e o tempo de adaptação de novos desenvolvedores. A Figura 1 apresenta um recorte da documentação visual do Material Design (GOOGLE, 2023), que serve como referência para o uso padronizado de tipografia, cores e elevação em interfaces digitais.

Diante desse cenário, este trabalho tem como objetivo repensar o design das interfaces internas do SGTCC, com foco na criação de uma documentação visual por meio de ferramentas de design de interfaces. A proposta inclui a construção de um Design System com componentes reutilizáveis e protótipos interativos, elaborados considerando a experiência e as necessidades dos diferentes tipos de usuários, como alunos, professores e membros externos. Parte das alterações será aplicada no código do sistema, buscando trazer melhorias concretas à sua usabilidade. Espera-se, com isso, facilitar futuras atualizações na interface, apoiar a tomada de decisões relacionadas ao design e oferecer uma visão geral da plataforma sem a

necessidade de navegar manualmente por cada tela com diferentes perfis de usuários. Além disso, a documentação visual também visa contribuir para o processo de integração de novos desenvolvedores no projeto, proporcionando uma base que agilize sua adaptação e integração à equipe.

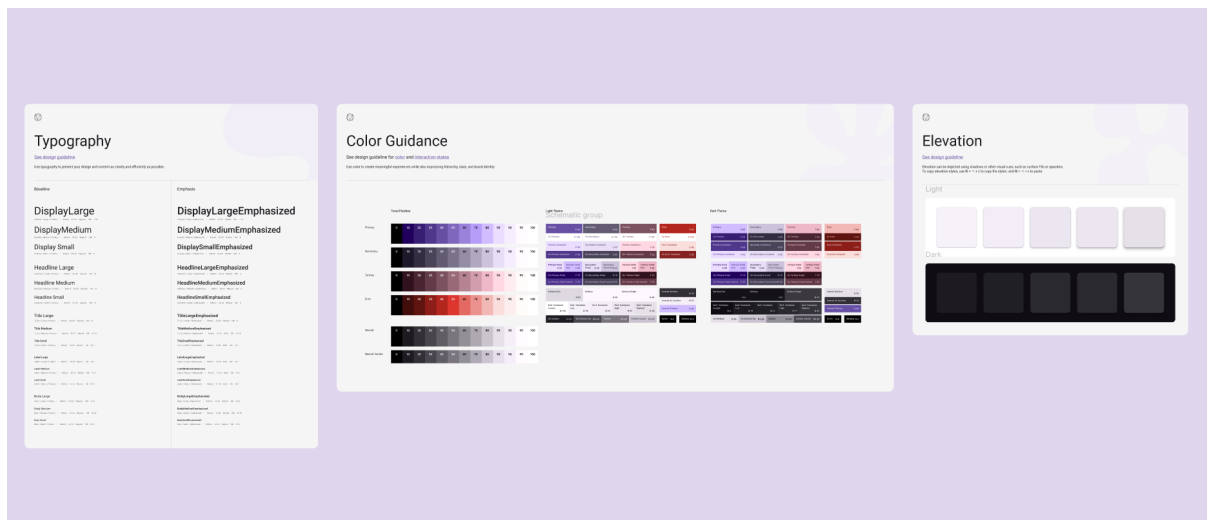


Figura 1 – Recorte da documentação visual Material Design, desenvolvido pela Google

1.1 Contextualização e Levantamento do Problema

A usabilidade de sistemas web influencia diretamente na forma como usuários interagem com funcionalidades essenciais, afetando não apenas a qualidade da experiência, mas também a produtividade e o engajamento (NIELSEN, 1993). No contexto do curso de TSI da UTFPR, o SGTCC foi desenvolvido em 2015 com o objetivo de centralizar e modernizar o processo de gerenciamento dos TCCs. Desde então, o sistema tem sido continuamente atualizado, tanto em funcionalidades (SILVA, 2019) quanto na experiência do usuário (LIMA, 2023) e na estrutura técnica (LUZ, 2025), consolidando-se como uma ferramenta amplamente utilizada por alunos, orientadores, professores e membros externos envolvidos no processo.

Atualmente, o SGTCC é uma ferramenta central no acompanhamento dos TCCs, sendo utilizado de forma recorrente por dezenas de usuários a cada semestre. É fundamental para a gestão de atividades como submissão de propostas, acompanhamento de orientações, marcação de bancas e registro de avaliações. A evolução do sistema é marcada por contribuições significativas de alunos e egressos. Silva (2019) digitalizou processos anteriormente manuais, integrou relatórios administrativos e iniciou a documentação da plataforma. Lima (2023) focou na reorganização de menus e melhorias pontuais de usabilidade, aplicando princípios de User Experience Design (UX). Mais recentemente, Luz (2025) modernizou o ambiente técnico, atualizando o Framework Ruby on Rails e suas bibliotecas, garantindo a continuidade e segurança do sistema.

Apesar da robustez funcional e das melhorias já realizadas, a interface do SGTCC ainda apresenta pontos que demandam aprimoramentos. Entre as dificuldades mais evidentes estão a redundância de elementos para usuários com múltiplos papéis, problemas de visualização em dispositivos móveis e a falta de uma base comum para decisões de design, o que compromete a consistência da interface e dificulta a manutenção contínua do sistema, especialmente por novos colaboradores.

Esses aspectos não apenas afetam a experiência do usuário, mas também dificultam a evolução técnica do projeto. Embora o sistema já utilize um *template*¹ padronizado no *front-end*², não há uma documentação visual formal que organize e explicita o uso desses elementos de maneira sistemática. Essa limitação torna-se ainda mais evidente diante da robustez do sistema, que atualmente conta com 174 telas distintas voltadas para diferentes tipos de usuários e fluxos de interação.

As melhorias realizadas até o momento focaram em resolver problemas pontuais, mas ainda não consolidaram uma base visual documentada que facilite a evolução do sistema. Entre as principais lacunas a serem abordadas, destacam-se:

- A ausência de um Design System documentado visualmente, baseado no template atual;
- A falta de protótipos navegáveis que sirvam como referência para novos desenvolvedores e colaboradores;
- A necessidade de resolver problemas recorrentes de usabilidade e melhorar a experiência em dispositivos móveis.

1.2 Objetivo Geral

Desenvolver um Design System para o SGTCC que organize diretrizes visuais e melhore a experiência de uso para todos os diferentes tipos de usuários do sistema.

1.3 Objetivos Específicos

- Mapear os componentes visuais atualmente utilizados na interface do SGTCC, identificando padrões e inconsistências de usabilidade relevantes;
- Desenvolver um conjunto de componentes visuais reutilizáveis com diretrizes claras de aplicação, visando à padronização da interface do sistema;

¹ Template: estrutura visual básica utilizada como ponto de partida para o desenvolvimento de interfaces.

² Front-end: camada de um sistema responsável pela interface com o usuário, onde ocorrem as interações visuais e operacionais.

- Elaborar protótipos navegáveis representando telas redesenhadas com base nas diretrizes propostas, com foco na experiência de uso por diferentes tipos de usuários do sistema.

1.4 Justificativa

A implementação de um Design System utilizando ferramentas de design de interfaces e a aplicação parcial desse sistema na interface do SGTCC buscam preencher as lacunas identificadas, oferecendo uma solução documentada, reutilizável e alinhada com práticas modernas de desenvolvimento visual (FROST, 2016).

Do ponto de vista da comunidade acadêmica, este trabalho contribui diretamente para a melhoria do SGTCC, criando um ambiente mais coeso, acessível e fácil de manter. Além disso, ao estabelecer uma base de design e documentação, este projeto oferece oportunidades para a continuidade de melhorias por meio de novos TCCs, como estudos sobre acessibilidade ou integração do Design System com o código-fonte de maneira dinâmica.

A adoção de Design System tem se consolidado em projetos profissionais, como no caso do Material Design, do Google, que serve como base de referência visual e interativa para o desenvolvimento de softwares (GOOGLE, 2023). A ausência desse tipo de documentação afeta diretamente o processo de desenvolvimento e manutenção da plataforma, especialmente para novos colaboradores, que deixam de ter acesso a uma base comum de referência desde sua entrada no projeto. Como aponta Frost (2016), integrar o Design System ao processo de *onboarding*³ é uma prática essencial para garantir que novos membros compreendam os princípios de modularidade e reutilização que sustentam o sistema. Sem uma referência visual clara, decisões sobre interface tornam-se inconsistentes, o que aumenta o tempo de desenvolvimento e dificulta a padronização entre módulos.

Do ponto de vista da viabilidade, o projeto é executável dentro do contexto do curso: o sistema SGTCC é de código aberto, possui histórico de colaboração por parte de alunos em trabalhos anteriores e adota tecnologias acessíveis a novos colaboradores. O autor já possui familiaridade com ferramentas de *front-end*, como o Bootstrap, e vem se aprofundando nos conceitos e práticas relacionados ao Framework Ruby on Rails com o objetivo de viabilizar a aplicação das melhorias propostas. O acesso ao código e a abertura institucional para contribuições garantem que o trabalho possa ser implementado, testado e eventualmente expandido por futuros projetos.

No contexto educacional, a qualidade das ferramentas digitais influencia diretamente a fluidez dos processos acadêmicos (BARROSO; ANTUNES, 2020). Um sistema bem estruturado e intuitivo facilita a gestão dos TCCs, contribui para a organização do curso e melhora a experiência de todos os usuários. Dessa forma, compreende-se a importância de estabelecer

³ Onboarding: processo de introdução e adaptação de novos usuários ou colaboradores a um sistema ou projeto.

uma base visual estruturada por meio de um Design System, capaz de sustentar a evolução do SGTCC de maneira alinhada aos princípios de usabilidade e manutenção colaborativa observados em projetos similares.

A escolha do tema também se justifica pelo interesse do autor na área de Interação Humano-Computador, especialmente em temas como design centrado no usuário, usabilidade e documentação visual. Esse interesse tem se refletido no estudo de ferramentas de prototipação como o Figma e na compreensão de como decisões de design e usabilidade influenciam diretamente a forma como sistemas são percebidos e utilizados pelos usuários.

2 REVISÃO DA LITERATURA

Este capítulo apresenta os principais conceitos que embasam o desenvolvimento do projeto.

2.1 Design System

Design System é um conjunto estruturado de elementos visuais e regras de uso que orientam a construção de interfaces digitais. Segundo Frost (2016), um Design System é uma abordagem modular e organizada que facilita o desenvolvimento de interfaces escaláveis. Ele permite que equipes de design e desenvolvimento compartilhem uma base comum de elementos, como botões, tipografia e cores, promovendo consistência e agilidade no processo de criação.

Kholmatova (2017) complementa essa definição ao destacar que um Design System não se limita a componentes visuais, mas também inclui diretrizes claras sobre como esses elementos devem ser aplicados em diferentes contextos, ajudando a criar uma apresentação visual uniforme em todo o produto.

Um Design System pode ser construído e mantido com o auxílio de ferramentas específicas. É comum utilizar softwares de prototipação de interfaces para organizar visualmente os componentes e documentar seus usos. Esses ambientes permitem criar bibliotecas reutilizáveis, versões navegáveis das telas e descrições detalhadas de cada elemento, o que contribui para que o sistema seja acessado e compreendido por toda a equipe durante o desenvolvimento do produto.

2.2 Atomic Design

Atomic Design é uma metodologia proposta por Frost (2016) para a construção e organização de interfaces de forma modular. A abordagem é baseada na ideia de que sistemas de interface podem ser compostos por unidades menores que se combinam progressivamente para formar estruturas mais complexas, seguindo uma lógica inspirada na composição da matéria na química.

A metodologia é dividida em cinco níveis:

- **Átomos:** são os elementos mais básicos da interface, como botões, ícones, campos de texto ou rótulos;
- **Moléculas:** são agrupamentos simples desses átomos, que formam componentes com funções específicas, como um campo de busca com botão;

- **Organismos:** são estruturas maiores que combinam moléculas e desempenham um papel mais definido na interface, como cabeçalhos ou cartões informativos;
- **Templates:** organizam esses organismos em um *layout*¹ de página, representando a estrutura de telas sem dados nem contextos reais da aplicação;
- **Páginas:** são versões reais dessas telas, com conteúdo final, utilizadas para simular cenários de uso.

O objetivo do Atomic Design é oferecer uma forma sistemática de construir e manter interfaces, promovendo consistência visual, reutilização de componentes e clareza na hierarquia dos elementos. Ao dividir a interface em partes menores, a metodologia facilita o entendimento da composição visual do sistema e torna mais acessível o processo de manutenção e atualização de seus elementos ao longo do tempo.

A Figura 2 ilustra visualmente os níveis propostos pelo Atomic Design, reforçando sua abordagem modular e progressiva para a construção de interfaces.

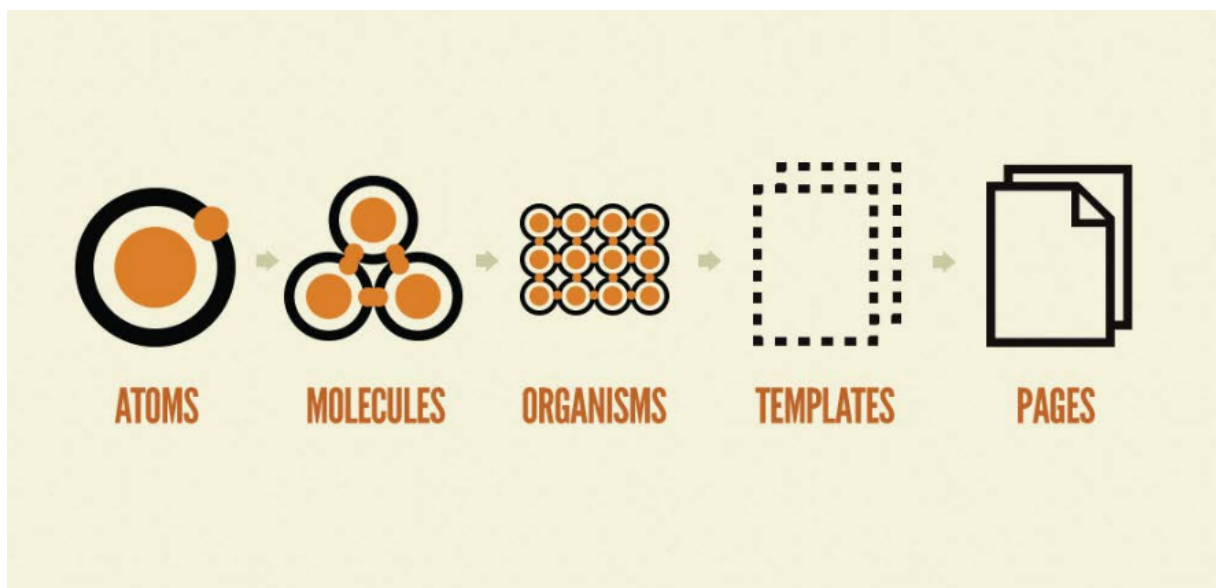


Figura 2 – Representação visual dos níveis do Atomic Design

2.3 Refactoring UI

Refactoring UI é uma abordagem apresentada por Wathan e Schoger (2019) que propõe a melhoria de interfaces gráficas por meio de ajustes visuais orientados por princípios de design aplicados de forma prática. A proposta não se baseia em modelos visuais prontos ou *layouts* preexistentes, mas sim na observação e organização das funcionalidades do sistema.

Os autores defendem que, ao reformular uma interface, o foco deve estar nas ações que o usuário precisa realizar em cada etapa do sistema, sem se basear necessariamente em telas

¹ Layout: organização visual dos elementos em uma interface, definindo a disposição dos conteúdos.

antigas. A ideia central é reorganizar os elementos com base na relevância funcional, permitindo que o *layout* final seja construído a partir de decisões práticas ligadas ao uso real da interface.

Um dos princípios centrais da abordagem é o uso da hierarquia visual. Elementos como contraste, tamanho, espaçamento e agrupamento são utilizados para orientar o olhar do usuário, indicando o que é mais relevante em cada área da tela. Essa hierarquia busca tornar a leitura da interface mais clara e compreensível, independentemente da complexidade do sistema.

A Figura 3 ilustra uma aplicação prática dos princípios do Refactoring UI, destacando como ajustes visuais — como contraste, espaçamento, hierarquia de informações e organização funcional — podem transformar uma interface confusa em uma versão mais clara, acessível e centrada nas necessidades do usuário.

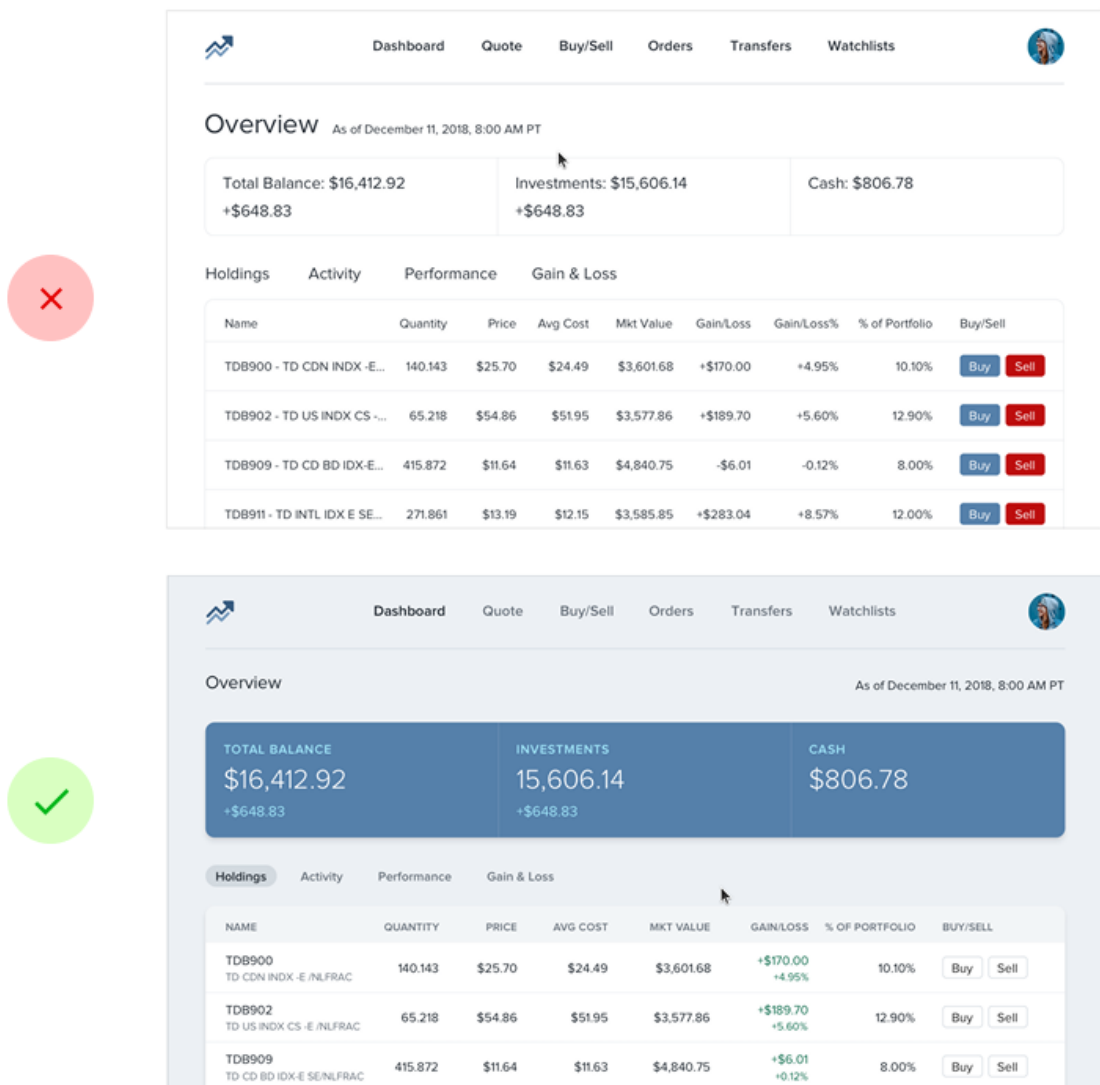


Figura 3 – Exemplo de refatoração visual com base nos princípios do Refactoring UI.

2.4 Teste de Usabilidade

O teste de usabilidade é uma técnica consolidada no design centrado no usuário, com o objetivo de identificar barreiras de compreensão, problemas de navegação e oportunidades de melhoria na interação com interfaces digitais (NIELSEN, 1993). Essa prática consiste em observar usuários reais enquanto realizam tarefas representativas, permitindo validar fluxos, componentes visuais e a clareza funcional antes da implementação definitiva (RUBIN; CHISNELL, 2008).

De acordo com a International Organization for Standardization (2018), a usabilidade de um sistema deve ser avaliada considerando a eficácia, a eficiência e a satisfação do usuário em contextos reais de uso, que podem ser medidas das seguintes maneiras:

- **Eficácia:** A eficácia é medida observando o sucesso nas tarefas realizadas pelos usuários. Durante o teste, os usuários são solicitados a completar tarefas específicas, e é verificado se conseguem atingir seus objetivos sem cometer erros significativos;
- **Eficiência:** A eficiência é avaliada com base no tempo gasto para completar cada tarefa e no número de tentativas necessárias. A análise inclui o número de erros cometidos, o tempo total para a execução e o tempo gasto em pontos críticos da interface;
- **Satisfação:** A satisfação pode ser medida por meio de questionários pós-teste, entrevistas ou observações subjetivas. Os participantes fornecem retornos sobre sua experiência com a interface, avaliando aspectos como facilidade de uso, intuitividade e a sensação geral de conforto durante a interação.

No contexto deste trabalho, os testes de usabilidade realizados com protótipos navegáveis terão como objetivo não apenas garantir que o Design System atenda às necessidades dos usuários, mas também validar as propostas de redesign da interface com base em retornos de usuários.

3 MATERIAIS E MÉTODOS

Este capítulo apresenta os materiais previstos para a realização do projeto, bem como os métodos que serão adotados para a criação e validação de um Design System para o SGTCC da UTFPR e a aplicação de certas melhorias propostas no código do projeto.

3.1 Materiais

Para acompanhar o andamento do projeto, será utilizada uma ferramenta de gerenciamento de tarefas que contribui para organizar as etapas de desenvolvimento e monitorar o progresso das atividades.

- **ClickUp:** Plataforma baseada na nuvem que oferece recursos como visualização de tarefas em listas ou quadros, categorização das mesmas além de anotações e acompanhamento de prazos (CLICKUP, 2025). No contexto deste trabalho, será utilizada para estruturar o fluxo de atividades e garantir o alinhamento com o planejamento previsto.

A construção do Design System e do redesign das interfaces exige uma ferramenta de design colaborativo voltada à criação e organização visual dos elementos que compõem o sistema.

- **Figma:** Ferramenta de design e prototipação de interfaces baseada na nuvem, que permite a criação de *layouts*, componentes reutilizáveis e protótipos interativos (FIGMA, 2025). A escolha do Figma se justifica por sua ampla adoção no mercado e pelas funcionalidades que oferece para a construção e organização de componentes visuais, recursos estes que são essenciais para o desenvolvimento deste projeto. Além disso, o autor já possui domínio prévio da ferramenta, o que permite maior autonomia na criação dos protótipos e na organização dos componentes visuais.
- **Font Awesome 6:** Biblioteca de ícones vetoriais amplamente utilizada em projetos web, oferecendo um conjunto extenso de símbolos com diferentes estilos (marcas, ícones vazados e ícones sólidos) (AWESOME, 2025). No contexto deste trabalho, a biblioteca foi utilizada como fonte principal para compor o conjunto de ícones do Design System, garantindo variedade, consistência visual e compatibilidade com o código do projeto.

Para viabilizar a aplicação prática do redesign no sistema, foi necessário utilizar ferramentas compatíveis com a base tecnológica já adotada no SGTCC. Isso inclui recursos para desenvolvimento web, versionamento de código e execução do ambiente local de forma controlada.

- **Ruby on Rails:** Framework web completo e robusto que acelera o desenvolvimento de aplicações web utilizando a linguagem Ruby. Ele facilita a criação de código estruturado, escalável e de fácil manutenção, seguindo convenções que reduzem a necessidade de configuração e promovem boas práticas de desenvolvimento (RAILS, 2025). Neste projeto, sua utilização se justifica pelo fato de o sistema SGTCC já estar desenvolvido com esse framework, permitindo que as melhorias propostas sejam aplicadas diretamente na base existente;
- **Bootstrap:** Framework *front-end* baseado em HTML, CSS e JavaScript, desenvolvido para facilitar o desenvolvimento de interfaces responsivas e padronizadas, a partir de um conjunto de componentes reutilizáveis e estilos pré-definidos (BOOTSTRAP, 2025). Neste projeto, o Bootstrap é utilizado porque já integra a base visual do SGTCC, servindo como suporte para a construção e organização de elementos da interface de forma compatível com o que já está implementado no sistema.
- **Docker:** Plataforma de virtualização leve que utiliza contêineres para empacotar aplicações com suas dependências, garantindo consistência na execução em diferentes ambientes (DOCKER, 2025). No projeto, o uso do Docker se dá porque o SGTCC já adota essa tecnologia para facilitar a instalação e execução do sistema localmente. Isso permite rodar o ambiente completo sem configurações manuais, mantendo compatibilidade com a estrutura original do projeto;
- **Git:** Sistema de controle de versão distribuído, gratuito e de código aberto, projetado para lidar com projetos de diferentes escalas, permitindo registrar e gerenciar alterações com rapidez e confiabilidade (GIT, 2025). No contexto deste trabalho, o Git é utilizado para versionamento do código-fonte e controle de branches, auxiliando na organização das alterações realizadas durante o desenvolvimento e facilitando a retomada de versões anteriores sempre que necessário;
- **Github:** Plataforma online de hospedagem de código-fonte que utiliza o Git como sistema de versionamento, oferecendo recursos para colaboração, revisão de código e gerenciamento de projetos (GITHUB, 2025). Neste trabalho, o GitHub é utilizado como repositório remoto para armazenamento do código, possibilitando o acompanhamento do histórico de desenvolvimento e a organização colaborativa das versões do projeto.

3.2 Métodos

A metodologia deste trabalho foi dividida nas seguintes etapas: identificação e organização dos componentes visuais existentes, análise das funcionalidades do sistema e refatoração das interfaces. Essas etapas foram estruturadas para orientar o processo de melhoria da interface do SGTCC, com foco na clareza visual, padronização e na experiência de uso. A Figura 4

ilustra esse fluxo metodológico, destacando as atividades desenvolvidas em cada fase e sua relação com os objetivos do projeto.

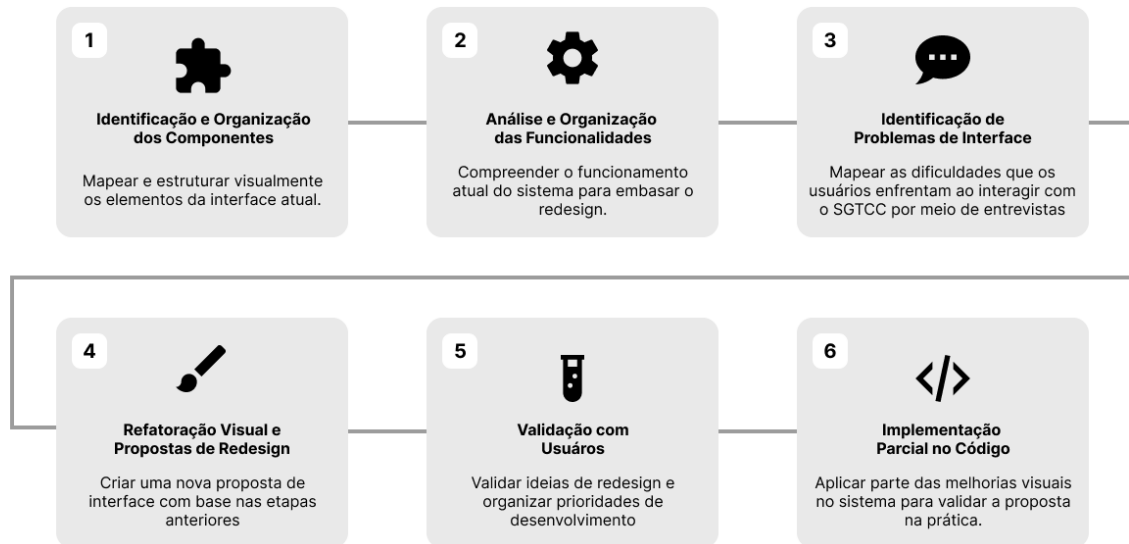


Figura 4 – Etapas metodológicas adotadas no desenvolvimento do projeto

3.2.1 Identificação e Organização dos Componentes

Nesta fase, será realizado o mapeamento dos elementos visuais presentes na interface atual do SGTCC. A estruturação seguirá a metodologia proposta por Frost (2016) em Atomic Design, organizando os elementos em níveis progressivos: de componentes básicos até a formação de páginas completas. Esse processo será conduzido diretamente no Figma, ferramenta adotada para a prototipação e documentação visual do sistema. O objetivo é estabelecer uma base clara e reutilizável que dê suporte às próximas etapas do projeto.

3.2.2 Análise e Organização de Funcionalidades

Antes da proposta de redesign das interfaces do SGTCC, é fundamental compreender a estrutura funcional do sistema. Para isso, será realizada uma etapa de análise de funcionalidades, com o objetivo de identificar, descrever e classificar os recursos atualmente disponíveis no sistema.

Essa etapa será conduzida por meio de uma exploração da aplicação, navegando por todas as seções disponíveis para os diferentes tipos de usuários. Cada funcionalidade será registrada com informações como nome, descrição, localização dentro do sistema, frequência de uso e observações quanto à sua clareza, relevância ou redundância.

Este mapeamento fornecerá uma visão abrangente da estrutura funcional do SGTCC, sendo essencial para embasar decisões sobre o que reestruturar durante a refatoração. Além disso, espera-se identificar eventuais inconsistências ou funcionalidades obsoletas, o que contribuirá para um redesenho mais centrado nas reais necessidades dos usuários.

3.2.3 Identificação de Problemas de Interface

A fase de identificação de problemas de interface visa mapear as dificuldades que os usuários enfrentam ao interagir com o SGTCC. O objetivo é coletar informações sobre as barreiras de usabilidade, os pontos de frustração e as áreas em que a navegação, a clareza ou a responsividade do sistema podem ser aprimoradas.

Para identificar os problemas de interface que afetam a experiência do usuário, serão realizadas entrevistas com alunos e professores que atualmente utilizam o sistema. Essas entrevistas abordarão as seguintes questões:

- **Dificuldades na navegação:** Como o usuário percebe a organização do sistema? Existem funcionalidades difíceis de acessar?
- **Interações confusas:** Existem áreas onde o usuário tem dificuldade em compreender a função de determinados botões, campos ou ícones?
- **Usabilidade em dispositivos móveis:** O usuário encontra dificuldades em acessar ou visualizar o sistema em telefones celulares, especialmente em páginas que envolvem tabelas ou formulários longos?

Durante as entrevistas, será incentivado que os participantes compartilhem experiências concretas, dificuldades específicas e sugestões para melhorias.

As respostas das entrevistas serão analisadas de forma a identificar padrões e temas comuns nas dificuldades relatadas pelos usuários. Esse processo vai envolver a organização das informações, agrupando as falas que tratam de questões semelhantes, como dificuldades de navegação ou problemas de visualização em dispositivos móveis. A ideia é entender quais são os problemas mais recorrentes e mais relevantes para os usuários, de modo a identificar com clareza os pontos que precisam de melhorias na interface.

As propostas de melhoria na interface serão desenvolvidas com base nos problemas e necessidades relatados pelos usuários durante as entrevistas, garantindo que as soluções atendam diretamente às questões mais críticas e que impactam a experiência dos usuários no SGTCC.

3.2.4 Refatoração Visual e Propostas de Melhorias

A etapa consiste na construção das novas propostas de interface com base nas informações coletadas nas entrevistas realizadas com os usuários, conforme descrito na fase anterior. A reformulação será orientada por uma abordagem prática de redesign, conforme discutido por Wathan e Schoger (2019) no Refactoring UI, que foca na clareza funcional das telas e na hierarquia visual dos elementos. Inicialmente, serão desenvolvidos protótipos de baixa fidelidade para representar os fluxos principais do sistema, sem preocupação com detalhes estéticos. Esses protótipos servirão como base para ajustes iterativos, permitindo testar e validar a disposição dos elementos conforme o uso previsto. Com as validações realizadas, será iniciado o refinamento visual, incorporando aspectos como espaçamento, contraste, tipografia e consistência entre componentes.

3.2.5 Validação com Usuários

Após o desenvolvimento dos protótipos navegáveis no Figma, será conduzido um teste de usabilidade com usuários reais do SGTCC, abrangendo participantes com diferentes perfis de acesso, como alunos, professores e membros externos. O objetivo desse teste é validar as propostas de redesign da interface, focando nas melhorias que foram feitas para resolver os problemas de usabilidade identificados nas entrevistas.

Durante os testes, os participantes serão solicitados a realizar tarefas específicas dentro dos protótipos, que correspondem às áreas da interface que foram alteradas ou aprimoradas com base no retorno obtido nas entrevistas anteriores. As ações que serão solicitadas aos usuários estarão diretamente relacionadas aos problemas identificados, como a navegação entre as telas ou a visualização de conteúdos em dispositivos móveis, por exemplo.

A ideia é que, ao realizar essas tarefas, o participante possa testar as mudanças feitas, como a reorganização de elementos, a melhoria da clareza dos menus ou a otimização da responsividade para dispositivos móveis. Durante a execução das tarefas, o participante será incentivado a relatar suas impressões, destacando eventuais dificuldades ou confusões, para que o autor consiga entender a experiência do usuário com as alterações propostas.

Os dados coletados durante a execução das tarefas, como dificuldade de navegação, áreas de confusão e sugestões espontâneas, serão analisados para identificar a eficácia das melhorias feitas. Esse retorno será fundamental para validar se as alterações atenderam às expectativas dos usuários e resolveram os problemas identificados anteriormente.

O desempenho do usuário será avaliado não apenas pelo sucesso em completar as tarefas, mas também pela facilidade com que ele as realiza. Se os usuários conseguirem completar as tarefas de forma fluida, sem dificuldades significativas, isso indicará que as melhorias atendem aos problemas de usabilidade identificados nas entrevistas. Caso contrário, se os usuários encontrarem dificuldades ou relatarem problemas durante a execução, isso sugere que a pro-

posta de redesign não foi suficientemente eficaz para resolver as questões de usabilidade e navegação.

Para priorizar quais melhorias devem ser implementadas de forma definitiva no sistema, será utilizada uma matriz de Esforço x Impacto. Nessa matriz, cada proposta de alteração será classificada de acordo com o benefício esperado para a experiência do usuário e o nível de esforço necessário para sua implementação. A Figura 5 ilustra essa matriz, destacando a priorização de soluções que oferecem alto impacto e baixo esforço de implementação.

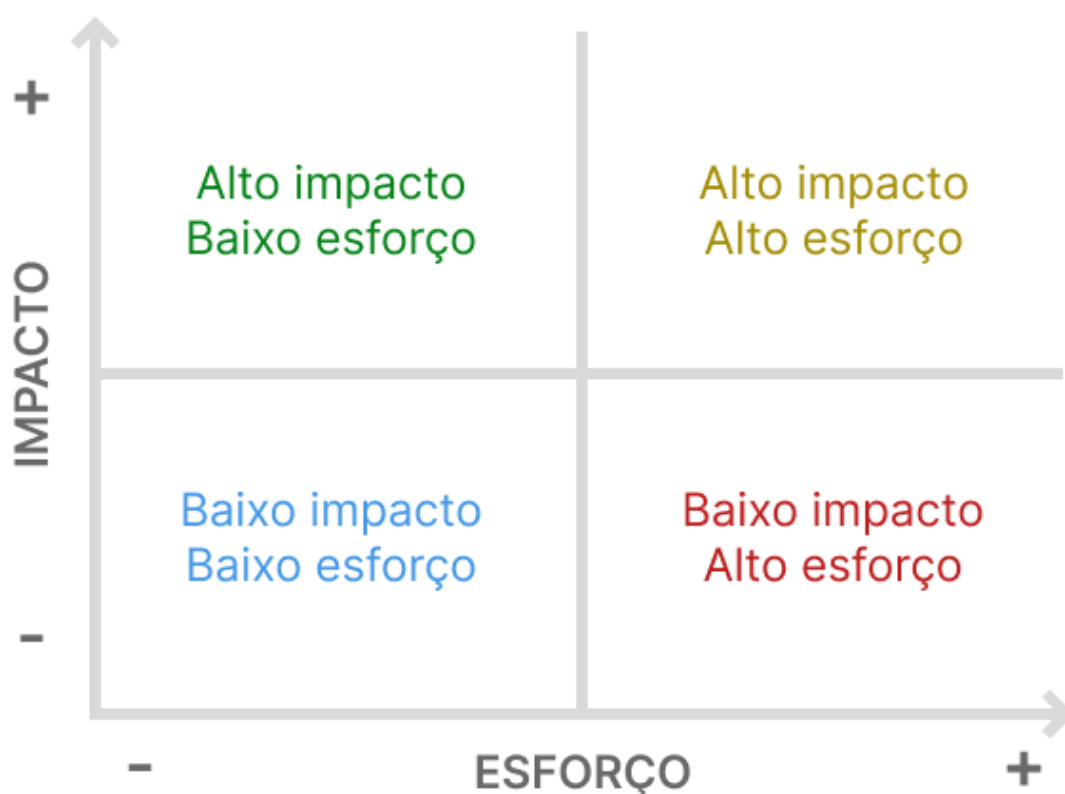


Figura 5 – Matriz de esforço e impacto

3.2.6 Implementação Parcial no Código

Após os testes de usabilidade e análise dos retornos dos usuários, as melhorias visuais com maior impacto e menor esforço serão aplicadas ao sistema SGTCC. O objetivo é integrar essas alterações na estrutura existente do sistema.

A priorização das propostas de redesign a serem implementadas será guiada pela matriz de Esforço x Impacto, levando em consideração o tempo disponível e a complexidade técnica das alterações. A proposta selecionada será aquela com maior impacto na experiência do usuário, mas que também exija um esforço técnico compatível com o prazo estipulado. Isso é relevante pois mudanças simples no design podem, em alguns casos, exigir ajustes mais com-

plexos, como a modificação de buscas no banco de dados ou a alteração de lógica no lado do servidor.

A implementação será realizada utilizando as tecnologias já empregadas no projeto, como o Framework Ruby on Rails, o Bootstrap e o Docker. Todas as alterações serão versionadas com o uso do Git e mantidas em repositório no GitHub, garantindo o controle do progresso, a rastreabilidade das mudanças realizadas e a continuidade para futuras evoluções do sistema.

4 RESULTADOS PRELIMINARES

A etapa inicial da metodologia foi concluída, com a identificação e organização dos principais componentes visuais do sistema. Foram definidos os átomos (paleta de cores, tipografia e ícones) e estruturadas as primeiras moléculas, tomando como base os padrões técnicos já utilizados no projeto. Esses elementos formam a base do Design System do SGTCC e servirão de apoio para as próximas fases de prototipação, testes de usabilidade e aplicação no código.

4.1 Átomos

No contexto do Design System elaborado para o SGTCC, os átomos representam os elementos mais básicos da interface. Esta seção descreve os principais átomos definidos até o momento, apresentando sua organização e as diretrizes que orientam sua aplicação nos componentes do sistema.

4.1.1 Paleta de Cores

A paleta de cores foi definida a partir de um levantamento dos estilos existentes no sistema, seguido por uma reorganização semântica que dissocia os valores de cor de suas funções visuais na interface. Essa estruturação garante consistência e facilita alterações centralizadas, propagando ajustes de forma automática em todos os elementos relacionados.

As cores foram agrupadas em quatro categorias principais:

- **Marca:** cores que reforçam a identidade do sistema, aplicadas em elementos de destaque e interação;
- **Texto:** tons dedicados a conteúdos textuais, planejados para assegurar contraste e legibilidade;
- **Tela:** tons de fundo e superfícies, que estruturam hierarquia de camadas;
- **Semânticas:** cores que sinalizam estados de uso e retornos ao usuário, como confirmações, alertas e erros.

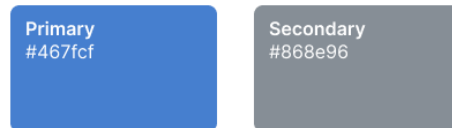
A Figura 6 apresenta essa organização, já parametrizada com variáveis no Figma, o que contribui para a manutenção e evolução do Design System.

4.1.2 Tipografia

A estrutura tipográfica foi definida a partir da análise dos tamanhos, pesos e hierarquias já utilizados na interface original. Observou-se que o sistema não contava com uma família tipo-

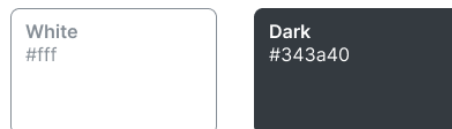
Marca

Cores usadas principalmente em elementos de destaque, como botões principais, links importantes, ícones de ação e componentes que comunicam a presença da marca na interface.



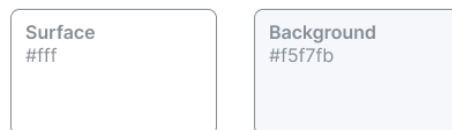
Neutras

Conjunto de cores básicas utilizadas como suporte



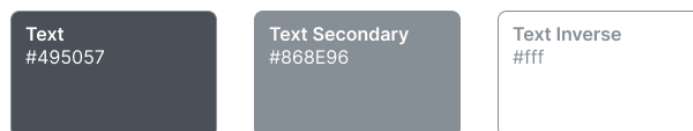
Tela

Conjunto de cores aplicadas como base visual da interface, englobando o fundo geral da tela e as superfícies de elementos sobrepostos. Define a hierarquia de camadas e contraste adequado.



Texto

Agrupar todas as cores destinadas a elementos textuais. Garante contraste, legibilidade e coerência tipográfica em diferentes fundos e contextos.



Semânticas

Cores que comunicam estados ou feedbacks específicos ao usuário, como sucesso, aviso, informação e erro. Ajudam a sinalizar ações, alertas e mensagens importantes, reforçando a experiência de uso.

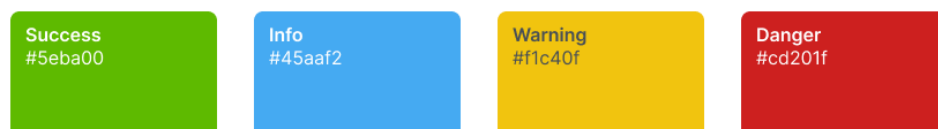


Figura 6 – Organização da paleta de cores do Design System do SGTCC, estruturada por categorias semânticas.

Elemento	Descrição	Tamanho (px)	Peso
<h1> Título Principal	Usado para títulos de páginas ou seções principais	32px	Semibold
<h2> Subtítulo	Usado para subtítulos diretos do Título Principal	28px	Semibold
<h3> Título de Seção	Para dividir seções dentro de uma página ou card	24px	Semibold
<h4> Título Secundário	Subdivisão de uma seção; ajuda a quebrar tópicos	18px	Semibold
<h5> Título de Bloco	Títulos de pequenos blocos de conteúdo	16px	Semibold
<h6> Título Auxiliar	Título menos importante; detalhes ou labels	14px	Semibold
<p> Parágrafo	Texto corrido padrão do conteúdo	16px	Regular

Figura 7 – Hierarquia tipográfica do Design System do SGTCC, com estilos definidos e documentados no Figma.

gráfica única, mas sim com uma lista de fontes comuns de sistemas operacionais, organizada em ordem de prioridade no arquivo de estilo.

Para garantir uniformidade e controle visual, definiu-se a família **Inter** como fonte principal do Design System, considerando sua legibilidade e compatibilidade em interfaces digitais.

A hierarquia tipográfica foi organizada em estilos semânticos, que variam de títulos principais a parágrafos padrão. Esses estilos foram cadastrados como variáveis de texto no Figma, garantindo consistência na aplicação, clareza na comunicação visual e facilidade de ajustes futuros.

A estrutura contempla:

- **Títulos principais e subtítulos:** com variações de tamanho para diferentes níveis hierárquicos, utilizando peso semi-negrito para reforçar a ênfase visual.
- **Parágrafo padrão:** definido em peso regular, adequado para textos corridos e conteúdos explicativos.

A Figura 7 apresenta a hierarquia tipográfica implementada no Figma, evidenciando tamanhos, pesos e descrições de uso, servindo como guia para a aplicação consistente dos estilos em toda a interface.

4.1.3 Ícones

O conjunto de ícones do Design System baseia-se na biblioteca **Font Awesome 6**, já utilizada no sistema do SGTCC. Para integrá-la ao projeto visual no Figma, foi incorporado

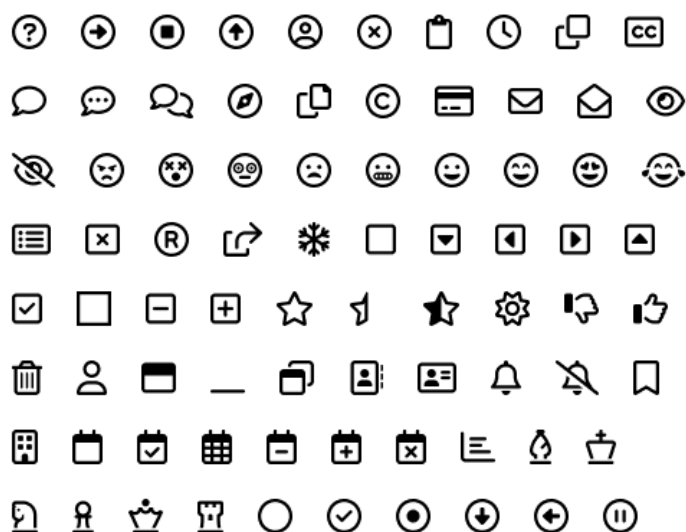


Figura 8 – Fragmento de ícones, parte do conjunto organizado no Figma para o Design System do SGTCC.

um arquivo disponível gratuitamente na Figma Community, contendo os ícones da biblioteca organizados como componentes reutilizáveis. Essa versão foi criada por terceiros e importada integralmente, sem modificações estruturais por parte do autor.

Os ícones estão agrupados em três categorias: marcas, ícones vazados e ícones sólidos, totalizando 1.734 elementos prontos para uso. A estrutura como componentes facilita substituições, garante uniformidade visual e simplifica futuras atualizações no sistema.

A Figura 8 mostra um fragmento dessa organização, conforme importado da Figma Community.

4.2 Moléculas

A etapa de organização das moléculas foi desenvolvida tomando como referência os componentes do Bootstrap 4, framework já adotado na base de código do SGTCC. Para garantir aderência entre o Design System e a implementação técnica, foram importados os principais componentes do Bootstrap a partir de bibliotecas gratuitas disponibilizadas na Figma Community.

O processo não se limitou à importação: cada componente foi revisado e ajustado para utilizar as variáveis de cores e tipografia definidas previamente, assegurando consistência visual e alinhamento com a identidade construída para o sistema. Essa padronização facilita futuras manutenções e garante que os protótipos reflitam com precisão o ambiente de produção.

A Figura 9 exemplifica um fragmento dessa organização, apresentando as variações de botões disponíveis.



Figura 9 – Variações de botões organizadas como moléculas no Figma, configuradas com as cores e tipografia do Design System do SGTCC.

4.3 Próximos Passos

Com a organização dos átomos e moléculas concluída, o próximo passo será realizar um mapeamento detalhado das funcionalidades do SGTCC, o qual guiará a criação de componentes compostos (organismos) e o desenvolvimento de protótipos navegáveis. Esses protótipos refletirão fluxos reais de uso, com base nos retornos obtidos nas entrevistas com usuários.

Esses protótipos passarão por testes de usabilidade com usuários do sistema, permitindo validar ajustes antes da aplicação no código. Uma interface validada e priorizada será implementada, integrando o Design System ao ambiente de produção.

Essa evolução servirá de base para a monografia, que reunirá o registro do processo, os resultados alcançados e as reflexões sobre as melhorias propostas.

5 CONSIDERAÇÕES FINAIS

A construção do Design System para o SGTCC representa um passo importante na padronização e documentação da interface do sistema. Ao longo deste trabalho, foi possível avançar em diversos pontos, que servirão de base para as etapas seguintes. Primeiramente, o estudo aprofundado das abordagens e metodologias que embasarão o desenvolvimento do projeto e as decisões a serem tomadas ao longo do trabalho garantiu que cada etapa do processo seja fundamentada em práticas consolidadas, como o Atomic Design de Frost (2016). Esse estudo assegura que todas as decisões de redesign estarão sempre orientadas por metodologias bem estabelecidas e referências reconhecidas.

Além disso, foi desenvolvido um processo metodológico que orientará as próximas fases do projeto, promovendo uma abordagem clara e consistente para a criação do Design System. Esse processo metodológico tem como objetivo a evolução contínua do sistema, garantindo que as decisões de redesign sejam tomadas com base em dados reais e na experiência dos usuários.

Outro avanço importante foi a definição e organização dos átomos e moléculas no Figma, utilizando os princípios do Atomic Design. Esse trabalho estabelece uma base sólida para a criação e reutilização de componentes visuais no sistema, garantindo que o design seja mais consistente e fácil de manter ao longo do tempo.

Esses avanços são fundamentais para a continuidade do trabalho, permitindo que as decisões de redesign, quando realizadas, sejam sempre fundamentadas em uma base sólida de metodologias e evidências práticas. Com isso, espera-se que o Design System continue evoluindo, atendendo às necessidades do SGTCC e proporcionando uma interface cada vez mais intuitiva.

REFERÊNCIAS

- AWESOME, F. **Icons for your projects. Easy. Done.** 2025. Disponível em: <https://fontawesome.com/>. Acesso em: 15 jun. 2025.
- BARROSO, F.; ANTUNES, M. Tecnologia na educação: ferramentas digitais facilitadoras da prática docente. **Pesquisa e Debate em Educação**, v. 5, n. 1, p. 124–131, jul. 2020. Disponível em: <https://periodicos.ufjf.br/index.php/RPDE/article/view/31969>.
- BOOTSTRAP. **Build fast, responsive sites with Bootstrap.** 2025. Disponível em: <https://getbootstrap.com/>. Acesso em: 9 jun. 2025.
- CLICKUP. **One app for projects, knowledge, conversations and more.** 2025. Disponível em: <https://clickup.com/>. Acesso em: 8 jun. 2025.
- DOCKER. **Accelerated Container Application Development.** 2025. Disponível em: <https://www.docker.com/>. Acesso em: 9 jun. 2025.
- FIGMA. **Figma helps design and development teams build great products, together.** 2025. Disponível em: <https://figma.com/>. Acesso em: 9 jun. 2025.
- FROST, B. **Atomic Design.** [S.l.]: Brad Frost, 2016. ISBN 0998296600, 9780998296609.
- GIT. **Distributed version control system.** 2025. Disponível em: <https://git-scm.com/>. Acesso em: 9 jun. 2025.
- GITHUB. **Build and ship software on a single, collaborative platform.** 2025. Disponível em: <https://github.com/>. Acesso em: 9 jun. 2025.
- GOOGLE. **Material Design Guidelines.** [S.l.], 2023. Disponível em: <https://m3.material.io/>. Acesso em: 4 maio 2025.
- International Organization for Standardization. **ISO 9241-11: Ergonomics of human-system interaction.** Rio de Janeiro, 2018. 29 p.
- KHOLMATOVA, A. **Design Systems: A practical guide to creating design languages for digital products.** [S.l.]: Smashing Media AG, 2017.
- LIMA, A. C. d. **Projeto e implementação de interface baseada na experiência do usuário para um sistema de gerenciamento de trabalho de conclusão de curso.** Universidade Tecnológica Federal do Paraná, Guarapuava, PR: [s.n.], 2023.
- LUZ, G. S. d. **Atualização do framework Rails para garantia da evolução do Sistema de Gestão de TCC.** Universidade Tecnológica Federal do Paraná, Guarapuava, PR: [s.n.], 2025.
- NIELSEN, J. **Usability Engineering.** [S.l.]: Academic Press, Inc., 1993. ISBN 9780125184069.
- RAILS, R. on. **Compress the complexity of modern web apps.** 2025. Disponível em: <https://rubyonrails.org/>. Acesso em: 9 jun. 2025.
- RUBIN, J.; CHISNELL, D. **Handbook of Usability Testing: How to plan, design, and conduct effective tests.** [S.l.]: Wiley, 2008.
- SILVA, R. G. A. **Aperfeiçoamento do Sistema de Gestão de Processos de Trabalho de Conclusão de Curso de Tecnologia em Sistemas para Internet da UTFPR Câmpus Guarapuava.** Universidade Tecnológica Federal do Paraná, Guarapuava, PR: [s.n.], 2019.

WATHAN, A.; SCHOGER, S. **Refactoring UI**: Practical design tips for developers to build beautiful interfaces. Self-published, 2019. Disponível em: <https://refactoringui.com/book/>.