

Interfaces de Usuário em Python com PyQt5: Uma Abordagem Front-End

Gabriel Flauzino Jara¹, Lincoln Mateus Da Silva², Vanderlei Nunes Sanches³

¹Sistemas de Informação – Universidade Federal do Mato Grosso do Sul (UFMS)
Caixa Postal 549 – 79070-900 – Ponta Porã – MS – Brazil

gabriel.flauzino@ufms.com, lincolnmathsilva@gmail.com, v.sanches@ufms.br

Abstract. *This work details the development of a prototype user interface for a furniture delivery management system, using Python with PyQt5 and Qt Designer. Emphasizing educational aspects, the project integrates concepts from software engineering, such as Kanban-based project management and version control, with human-computer interaction principles guiding its interface design. Initially an academic exercise, it demonstrates potential for real-world application, showcasing the journey from theoretical learning to practical software development. This project exemplifies the continuous learning process in the technology field and the transition from academic concepts to real-world solutions.*

Resumo. *Este trabalho detalha o desenvolvimento de uma interface de usuário protótipo para um sistema de gerenciamento de entregas de móveis, utilizando Python com PyQt5 e Qt Designer. Enfatizando aspectos educacionais, o projeto integra conceitos de engenharia de software, como gerenciamento de projetos baseado em Kanban e controle de versão, com princípios de interação humano-computador guiando seu design de interface. Inicialmente um exercício acadêmico, ele demonstra potencial para aplicação no mundo real, mostrando a jornada do aprendizado teórico para o desenvolvimento prático de software. Este projeto exemplifica o processo contínuo de aprendizagem no campo da tecnologia e a transição de conceitos acadêmicos para soluções no mundo real.*

1. Introdução

Este artigo explora o desenvolvimento de interfaces de usuário com a biblioteca PyQt5 em Python, com foco no front-end para aplicações desktop. Por meio de um estudo de caso de um sistema de gerenciamento de pedidos, demonstramos como interfaces gráficas complexas e interativas podem ser construídas e manipuladas em Python. O estudo realça a experimentação e a aprendizagem na construção de interfaces voltadas ao ambiente desktop usando como base a linguagem de programação Python. Uma das características mais cativantes do Python é a sua capacidade de integração perfeita com a interface gráfica do usuário (GUI), neste contexto, o Qt Designer, uma ferramenta amplamente reconhecida e integrada ao ecossistema Python, permite que os desenvolvedores desenhem a interface do sistema de maneira intuitiva e visual. Esse paradigma de "desenhar primeiro, codificar depois" pode ser uma revolução no desenvolvimento de sistemas, acelerando a prototipagem e o desenvolvimento. Ao longo deste trabalho, exploraremos como o Python e o Qt Designer proporcionam uma abordagem mais eficaz para o desenvolvimento de sistemas, permitindo que os desenvolvedores comecem a construir sistemas com interfaces atraentes e funcionais antes mesmo de escrever uma única linha de código. Cabe ressaltar que muito tem-se

falado da utilização da linguagem Python, contudo no ambiente Web, por isso uma das contribuições deste trabalho é apresentar sua utilização no ambiente desktop. Essa abordagem inovadora tem o potencial de economizar tempo, melhorar a colaboração entre equipes e transformar a maneira como sistemas são concebidos e desenvolvidos.

O desenvolvimento de interfaces de usuário eficientes e intuitivas é fundamental no panorama atual do software, onde a experiência do usuário determina significativamente a aceitação e o sucesso de qualquer aplicação. Este artigo explora o uso do PyQt5, uma poderosa biblioteca de desenvolvimento de interfaces gráficas em Python, focando em sua aplicação no setor moveleiro, um domínio desafiador devido à sua natureza logística e operacional específica.

A escolha do Python e, em particular, do PyQt5 para o desenvolvimento de interfaces gráficas de usuário (GUI) não é aleatória. Python emergiu como uma das linguagens de programação mais populares e versáteis, muito devido à sua simplicidade e legibilidade. O PyQt5, por sua vez, combina a simplicidade do Python com o poder do Qt, uma das mais robustas e antigas bibliotecas de GUI. Este casamento entre linguagem e biblioteca oferece uma plataforma ideal para criar interfaces ricas, reativas e esteticamente agradáveis.

O setor moveleiro, objeto de nosso estudo, enfrenta desafios únicos. As empresas deste setor lidam com a venda e distribuição de produtos volumosos e muitas vezes personalizados, como móveis e utensílios domésticos. A complexidade aumenta com a necessidade de serviços de montagem em domicílio, que requerem uma logística precisa e eficiente. As interações cliente-empresa são frequentes e variadas, abrangendo consultas, pedidos, agendamentos de entrega e montagem, e serviços pós-venda. Um sistema que gerencie estas interações de maneira eficaz é, portanto, não apenas benéfico, mas essencial para a satisfação do cliente e a eficiência operacional.

Este artigo foca na construção de um sistema de gerenciamento de montagem e entrega para uma empresa fictícia de móveis, enfrentando desafios como agendamentos conflitantes, atrasos nas entregas, e a conseqüente insatisfação do cliente. A proposta visa explorar como o PyQt5 pode ser utilizado para criar uma solução que não apenas resolva esses desafios operacionais, mas também melhore a interação geral do usuário com o sistema. Este estudo de caso serve como um microcosmo para demonstrar a aplicabilidade e eficácia do PyQt5 em situações reais de negócios, particularmente em setores que não são tradicionalmente considerados na vanguarda tecnológica.

Ao longo deste artigo, discutiremos as especificidades do setor moveleiro e como elas influenciam os requisitos e design do sistema. Além disso, abordaremos como as características do PyQt5, como a facilidade de uso, a capacidade de criar designs responsivos e a integração com outras ferramentas e bibliotecas Python, tornam-no uma escolha ideal para este projeto. Ao final, pretendemos não apenas apresentar uma solução funcional para um problema específico, mas também destacar a versatilidade e o potencial do PyQt5 como uma ferramenta para desenvolvedores que buscam criar experiências de usuário ricas e envolventes em diversos domínios.

A interseção entre a tecnologia e o design de interfaces no contexto de aplicações empresariais representa uma área de crescente importância e inovação. Em particular, o

setor moveleiro, com suas exigências únicas, representa um campo fértil para a exploração e implementação de soluções tecnológicas inovadoras.

Este setor, muitas vezes visto como tradicional e resistente à mudança, na realidade, apresenta uma variedade de desafios operacionais e de atendimento ao cliente que podem ser efetivamente abordados com as soluções tecnológicas certas. A implementação de um sistema de gerenciamento de montagem e entrega de móveis baseado em PyQt5 ilustra perfeitamente como a inovação tecnológica pode ser aplicada para resolver problemas práticos em um setor tradicional.

A escolha de focar no desenvolvimento de interfaces gráficas usando Python e PyQt5 para este projeto não é apenas uma preferência tecnológica, mas também uma decisão estratégica. Python, com sua sintaxe clara e legibilidade, torna-se uma escolha lógica para projetos onde a clareza de código e a facilidade de manutenção são primordiais. Além disso, a extensa gama de bibliotecas e frameworks disponíveis em Python facilita a integração com sistemas existentes e a expansão para novas funcionalidades. O PyQt5, especificamente, oferece um conjunto robusto de ferramentas para criar interfaces de usuário que não são apenas funcionais, mas também esteticamente agradáveis e intuitivas. Este aspecto é crucial no setor moveleiro, onde a aparência e a apresentação têm um papel significativo.

Neste estudo de caso, abordamos os desafios práticos enfrentados por uma empresa de móveis fictícia. A empresa luta com problemas comuns no setor, como agendamentos conflitantes, atrasos nas entregas, e a dificuldade de gerenciar eficientemente a equipe de montagem. Esses desafios não são triviais; eles afetam diretamente a satisfação do cliente e, por extensão, o sucesso do negócio. A resposta a esses desafios é um sistema de gerenciamento que não apenas otimiza os processos operacionais, mas também melhora a experiência do usuário – tanto para os funcionários da empresa quanto para os clientes finais.

Além disso, este artigo visa lançar luz sobre a aplicabilidade prática e os benefícios de usar PyQt5 em um contexto de negócios real. Através deste estudo de caso, demonstraremos como o PyQt5 pode ser usado para criar interfaces de usuário que são não apenas funcionalmente ricas, mas também capazes de lidar com complexidades específicas do setor, como a logística de entrega e montagem de móveis e a comunicação entre diferentes partes interessadas. O estudo também explora como a integração e a sincronização com sistemas de back-end, essenciais para o sucesso de qualquer sistema de gerenciamento, podem ser alcançadas usando a combinação de Python e PyQt5.

Este artigo não se destina apenas a ser uma exposição técnica sobre o PyQt5 e suas capacidades. Pretendemos também fornecer insights sobre como a tecnologia pode ser aplicada de maneira eficaz para solucionar problemas reais do mundo empresarial, particularmente em setores que podem não ser imediatamente associados à inovação tecnológica.

Ao oferecer uma visão detalhada do processo de desenvolvimento, desde a concepção até a implementação, esperamos fornecer um modelo valioso para outros profissionais e entusiastas da tecnologia que buscam aplicar soluções inovadoras em seus próprios desafios de negócios.

2. Fundamentação Teórica

As subseções seguintes apresentam os conceitos que embasam o desenvolvimento deste trabalho.

2.1. Python

Python é uma linguagem de programação de alto nível, interpretada, com uma filosofia de design que enfatiza a legibilidade do código. Sua sintaxe simples e clara torna Python particularmente adequada para prototipagem rápida e desenvolvimento de aplicações complexas. Além disso, Python é uma linguagem versátil, suportando vários paradigmas de programação, como programação orientada a objetos, imperativa e funcional. Esta flexibilidade permite que desenvolvedores implementem soluções eficientes e mantenham o código facilmente compreensível e gerenciável.

A popularidade de Python também se deve à sua vasta comunidade e ao extenso ecossistema de bibliotecas e frameworks, que permitem a extensão de suas funcionalidades para diversas áreas, incluindo desenvolvimento Web, análise de dados, aprendizado de máquina, e claro, desenvolvimento de interfaces gráficas com ferramentas como PyQt5.

No contexto do desenvolvimento de interfaces gráficas, PyQt5 emerge como uma escolha predominante devido à sua integração profunda com o Qt, um dos frameworks de GUI mais poderosos e versáteis. O PyQt5 oferece aos desenvolvedores acesso a uma ampla gama de widgets e ferramentas para criar interfaces interativas e esteticamente atraentes. Além disso, a compatibilidade do PyQt5 com as ferramentas do Qt, como o Qt Designer, permite uma abordagem mais visual e menos codificada para o design de interface, o que é ideal para prototipagem e desenvolvimento rápido.

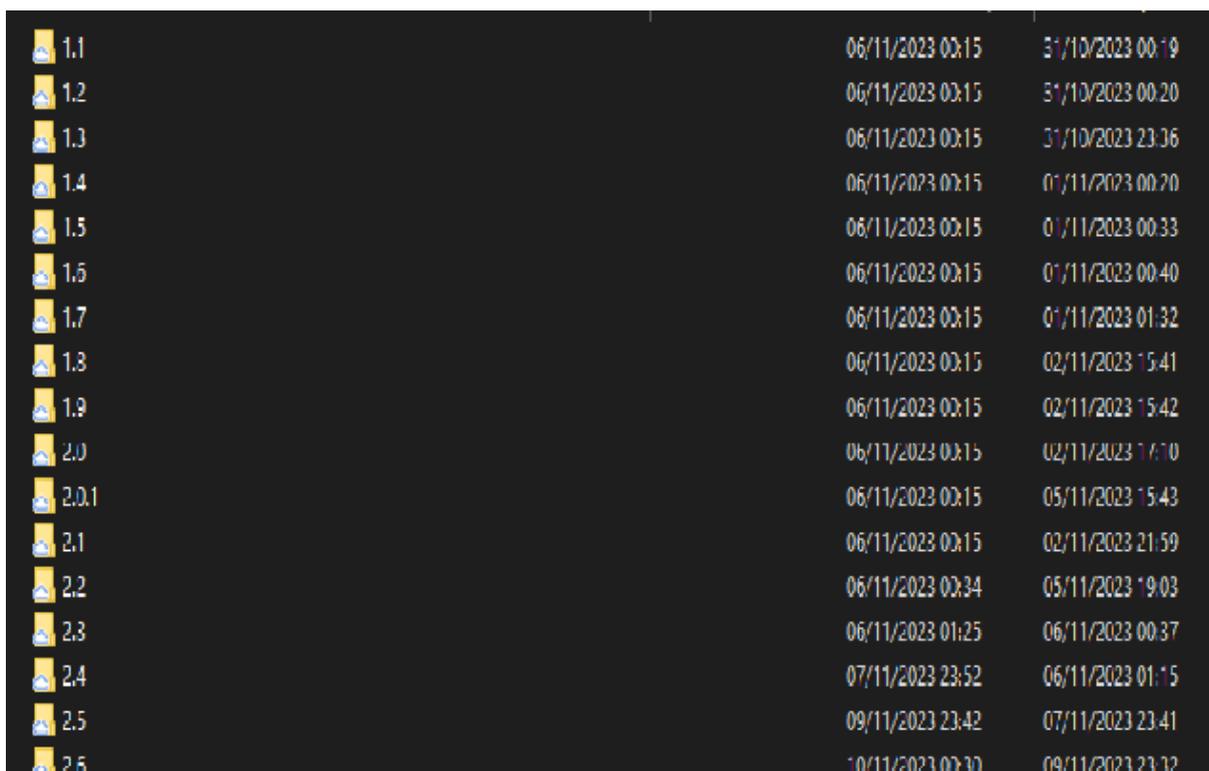
O Qt Designer é uma ferramenta fundamental neste processo. Ele permite a criação de designs de GUI de forma intuitiva e visual, sem a necessidade de escrever código para a interface. Isso não apenas acelera o processo de desenvolvimento, mas também facilita a colaboração entre designers e programadores, pois os designs podem ser facilmente compartilhados e revistos.

O ambiente de desenvolvimento para uma aplicação PyQt5 geralmente envolve um editor de código ou uma IDE (Integrated Development Environment) que suporte Python e ferramentas adicionais para controle de versão e debugging. IDEs populares como: PyCharm, Visual Studio Code, ou mesmo editores mais simples como Sublime Text ou Atom, podem ser configurados para trabalhar com PyQt5, proporcionando recursos como realce de sintaxe, autocompletar código e depuração integrada.

O desenvolvimento também pode ser apoiado por sistemas de controle de versão como Git, facilitando o gerenciamento de diferentes versões do código e a colaboração entre membros da equipe. Este aspecto é crucial em projetos de desenvolvimento de software, onde várias funcionalidades podem estar sendo desenvolvidas simultaneamente. Neste contexto, Sommerville (2011, p. 481) é preciso ao declarar: “O gerenciamento de versões (VM, do inglês version management) é o processo de acompanhamento de

diferentes versões de componentes de software ou itens de configuração e os sistemas em que esses componentes são usados. Ele também envolve a garantia de que as mudanças feitas por diferentes desenvolvedores para essas versões não interferem umas nas outras. Portanto, você pode pensar em gerenciamento de versões como o processo de gerenciamento de codelines e baselines.”

A Figura 1 mostra a uma imagem do Git contendo diferentes versões do desenvolvimento da aplicação desenvolvida.



1.1	06/11/2023 00:15	31/10/2023 00:19
1.2	06/11/2023 00:15	31/10/2023 00:20
1.3	06/11/2023 00:15	31/10/2023 23:36
1.4	06/11/2023 00:15	01/11/2023 00:20
1.5	06/11/2023 00:15	01/11/2023 00:33
1.5	06/11/2023 00:15	01/11/2023 00:40
1.7	06/11/2023 00:15	01/11/2023 01:32
1.8	06/11/2023 00:15	02/11/2023 15:41
1.9	06/11/2023 00:15	02/11/2023 15:42
2.0	06/11/2023 00:15	02/11/2023 16:10
2.0.1	06/11/2023 00:15	05/11/2023 15:43
2.1	06/11/2023 00:15	02/11/2023 21:59
2.2	06/11/2023 00:34	05/11/2023 19:03
2.3	06/11/2023 01:25	06/11/2023 00:37
2.4	07/11/2023 23:52	06/11/2023 01:15
2.5	09/11/2023 23:42	07/11/2023 23:41
2.5	10/11/2023 00:30	09/11/2023 23:32

Figura 1. versões do projeto organizadas por pastas.

A combinação de Python com PyQt5 e Qt Designer forma um ecossistema robusto para o desenvolvimento de interfaces de usuário. Python oferece a base programática com sua sintaxe clara e legibilidade, enquanto PyQt5 e Qt Designer ampliam essa base com poderosas ferramentas gráficas. O ambiente de desenvolvimento, complementado por uma IDE apropriada e sistemas de controle de versão, proporciona uma plataforma sólida para o desenvolvimento eficiente e colaborativo de aplicações de software.

2.2. Visão Geral do PyQt5

PyQt5 é um conjunto de ferramentas de desenvolvimento de software que permite a criação de aplicações GUI robustas em Python. Como um binding do famoso toolkit Qt, desenvolvido originalmente pela empresa norueguesa Trolltech (agora The Qt Company), o PyQt5 oferece acesso a uma ampla gama de funcionalidades de Qt em Python. O Qt é amplamente reconhecido por sua capacidade de desenvolver interfaces de usuário ricas e aplicativos cross-platform com um desempenho de alto nível. O

PyQt5 herda essa capacidade, permitindo aos desenvolvedores Python criar aplicativos que são esteticamente agradáveis, funcionalmente ricos e altamente responsivos.

PyQt5 se destaca por sua abrangência e versatilidade. Ele oferece recursos que vão desde a criação de simples janelas de diálogo até o desenvolvimento de complexas interfaces de usuário com gráficos avançados. Além disso, o PyQt5 suporta uma ampla gama de widgets padrão do Qt, permitindo uma personalização extensiva e a criação de interfaces intuitivas. Além de seus recursos de GUI, o PyQt5 também oferece módulos para acesso a banco de dados, threading, comunicação de rede e outros. Essa ampla gama de funcionalidades torna o PyQt5 uma escolha ideal para projetos de qualquer escala, desde pequenos aplicativos de desktop até sistemas empresariais complexos.

As interfaces gráficas de usuário são componentes cruciais de muitos softwares modernos. Uma GUI bem projetada melhora significativamente a experiência do usuário, tornando o software intuitivo, acessível e fácil de usar. Elementos fundamentais de uma GUI de sucesso incluem um layout claro, uma navegação intuitiva e feedback visual responsivo. Além disso, a consistência em todo o design da interface ajuda os usuários a aprenderem e a se adaptarem rapidamente ao software.

O desenvolvimento de uma GUI eficaz requer uma compreensão profunda dos usuários finais e de suas necessidades. Isto é especialmente verdadeiro em setores específicos, como o moveleiro, onde os requisitos e expectativas dos usuários podem ser bastante distintos. A GUI deve ser projetada não apenas para facilitar a realização de tarefas, mas também para aprimorar a experiência geral do usuário, engajando-o de maneira positiva com o produto ou serviço.

No setor moveleiro, a relevância do desenvolvimento front-end se manifesta em várias frentes. Primeiramente, um sistema de gerenciamento eficiente para agendamento de entregas e montagens de móveis necessita de uma interface que seja intuitiva para os funcionários e gestores. Esses usuários precisam acessar informações de forma rápida e eficiente, realizar agendamentos sem conflitos e comunicar-se eficazmente com os clientes. Uma interface bem projetada pode simplificar significativamente esses processos, reduzindo o risco de erros e melhorando a eficiência operacional.

Além disso, no setor de varejo de móveis, a experiência do cliente é um aspecto fundamental. Uma GUI atraente e fácil de usar pode ser um diferencial importante, especialmente em aplicações de e-commerce ou em sistemas de autoatendimento em lojas físicas. Uma interface que permite aos clientes explorar produtos, verificar disponibilidades, e agendar entregas e montagens de forma independente, não apenas melhora a experiência do cliente, mas também aumenta a eficiência ao reduzir a carga de trabalho do pessoal de vendas e suporte.

Finalmente, a flexibilidade do PyQt5 no desenvolvimento front-end permite a integração com outros sistemas back-end, como gerenciamento de estoque e logística, essenciais no setor moveleiro. Esta integração assegura que a informação apresentada na interface reflita os dados mais atualizados, melhorando a precisão e a confiabilidade do sistema na totalidade.

Em resumo, o PyQt5 emerge como uma ferramenta poderosa para enfrentar os desafios únicos do setor moveleiro. Sua capacidade de criar interfaces de usuário sofisticadas e

intuitivas, combinada com a flexibilidade e a robustez do Python, o torna uma escolha ideal para desenvolvedores que buscam criar soluções eficazes e centradas no usuário para este setor.

2.3. Utilização do Qt Designer e Conversão para Código Fonte em PyQt5

Qt Designer é uma ferramenta poderosa e versátil para o design visual de interfaces gráficas de usuário (GUI). Desenvolvida como parte do framework Qt, esta ferramenta permite aos desenvolvedores e designers criar interfaces de forma intuitiva, utilizando uma abordagem de arrastar e soltar para colocar widgets. O Qt Designer suporta a personalização detalhada de cada widget, incluindo propriedades como geometria, estilo e sinais/eventos, facilitando a criação de interfaces complexas e responsivas sem a necessidade de escrever código preliminar.

A principal vantagem do Qt Designer é a sua capacidade de acelerar significativamente o processo de desenvolvimento de interfaces. Ao fornecer uma representação visual da interface, o Qt Designer permite que desenvolvedores e designers:

Visualizem a disposição e aparência dos elementos da interface antes da implementação;

Experimentem diferentes layouts e estilos de forma rápida e conveniente;

Ajustem propriedades de widgets e elementos de interface de forma interativa;

Trabalhem colaborativamente, compartilhando e revisando designs de interface de forma eficaz;

Após a conclusão do design da interface no Qt Designer, o próximo passo é integrar essa interface com a lógica de programação em Python utilizando o PyQt5. Este processo é realizado através de uma ferramenta de conversão (como `pyuic5`), que transforma o arquivo de design (geralmente com a extensão `.ui`) em um módulo Python. Este módulo Python contém uma classe que replica a interface projetada no Qt Designer.

A classe gerada pelo processo de conversão pode ser integrada em qualquer aplicação PyQt5. No projeto em questão, a classe `Ui_MainWindow` foi gerada a partir de um design do Qt Designer, encapsulando todos os widgets e layouts definidos visualmente. Este método demonstra uma abordagem prática e eficiente para o desenvolvimento de GUI em PyQt5, unindo o poder do design visual com a flexibilidade da programação em Python.

A combinação do Qt Designer com a conversão para código PyQt5 representa uma metodologia poderosa no desenvolvimento de interfaces de usuário. Esta abordagem não só melhora a eficiência e a precisão no processo de desenvolvimento, mas também facilita a colaboração entre designers e programadores. Ao adotar esta metodologia, o projeto beneficia de uma interface de usuário que não só atende aos requisitos estéticos e funcionais, mas também é facilmente atualizável e manutenível.

2.4. Abordagem Adotada Para Construção de Interfaces de Usuário

A metodologia adotada para a construção de interfaces de usuário no projeto foca na combinação de princípios de design centrado no usuário com as funcionalidades

técnicas fornecidas pelo PyQt5. Inicialmente, foi realizada uma análise detalhada dos requisitos do usuário, identificando as necessidades específicas dos funcionários da empresa de móveis e de seus clientes. Essa análise envolveu workshops com partes interessadas, questionários e a revisão de processos de negócios existentes, para garantir uma compreensão abrangente dos desafios enfrentados.

Com base nessa análise, desenvolvemos protótipos de interfaces, que foram iterativamente refinados por meio de sessões de teste com usuários representativos. Esses testes ajudaram a identificar pontos de fricção e áreas de melhoria, garantindo que a interface final fosse tanto funcional quanto intuitiva. A abordagem iterativa também permitiu adaptar rapidamente a interface às mudanças nos requisitos do usuário ou nos processos de negócios.

Tecnicamente, a interface foi construída utilizando os recursos avançados do PyQt5, que permite a criação de layouts personalizados, a incorporação de widgets complexos e a implementação de funcionalidades interativas. A arquitetura do sistema foi projetada para ser modular, permitindo fácil manutenção e atualização. Além disso, grande ênfase foi colocada na integração da interface do usuário com os sistemas de back-end, garantindo que a interface reflita informações precisas e atualizadas.

O estudo de caso se concentra em uma empresa fictícia do setor moveleiro, enfrentando desafios relacionados à gestão de agendamentos de entregas e montagens de móveis. Os problemas identificados incluíam agendamentos conflitantes, atrasos nas entregas, dificuldades na comunicação entre equipe de montagem e clientes, e a necessidade de otimizar a alocação de recursos.

Para abordar esses desafios, o sistema de gerenciamento de montagem e entrega foi concebido com várias funcionalidades-chave. Primeiramente, o sistema permite aos funcionários gerenciar e agendar entregas e serviços de montagem, evitando conflito de horário e considerando a localização geográfica para otimizar as rotas. Além disso, o sistema oferece aos clientes a possibilidade de visualizar e alterar seus agendamentos de forma independente, por meio de uma interface de usuário amigável, conforme a Figura 2.



Figura 2. Tela Principal do projeto desenvolvida com Qt Designer.

Durante o desenvolvimento, a integração do estudo de caso no sistema foi realizada por meio de uma série de etapas iterativas, envolvendo a construção e o teste de protótipos da interface. Cada iteração foi acompanhada de feedback das partes interessadas, resultando em melhorias contínuas. A ênfase foi colocada em garantir que a interface fosse não apenas funcional, mas também agradável de usar, refletindo as melhores práticas de design de GUI.

Este processo metodológico garantiu que o sistema desenvolvido não apenas atendesse às necessidades operacionais da empresa fictícia, mas também proporcionasse uma experiência de usuário excelente, tanto para os funcionários quanto para os clientes. A abordagem cuidadosa e centrada no usuário adotada neste estudo de caso ilustra como a integração de práticas de design de interface de usuário com as capacidades técnicas do PyQt5 pode resultar em soluções de software eficazes e bem-sucedidas.

3. Estudo de Caso: Móveis/Controle de Entregas – Prototipagem de Interfaces

No contexto do setor moveleiro, o gerenciamento eficiente de entregas é crucial e este estudo de caso se concentra na prototipagem de uma interface para um sistema de controle de entregas em uma empresa de móveis fictícia.

O foco principal é desenvolver um protótipo de interface gráfica utilizando PyQt5, para um sistema desktop para agendamento de entregas e montagem de móveis, ao mesmo tempo, em que expõe e demonstra as capacidades de prototipagem de interfaces de usuário da ferramenta.

Assim o objetivo deste projeto é criar uma série de protótipos de interface que ilustram como tal sistema poderia funcionar e interagir com os usuários, ao mesmo tempo, em que permite avaliar o uso do Python e do PyQt5 como ferramentas de desenvolvimento.

O foco é:

- Demonstração das capacidades do PyQt5 para criar interfaces intuitivas e atraentes.
- Exploração de diferentes layouts e widgets para otimizar a experiência do usuário.
- Experimentação com funcionalidades interativas para gerenciamento de entregas e montagens.

Ainda que muito popular, muitos estudantes desconhecem a utilização da linguagem Python no contexto de aplicações desktop e, neste sentido, como objetivo secundário, objetiva-se contribuir para a disseminação de conhecimento entre a comunidade acadêmica.

Neste estudo de caso, abordamos a situação de uma empresa fictícia de móveis que enfrenta desafios significativos em seu sistema de gerenciamento de entregas. Na empresa, os vendedores anotam as entregas em um caderno, um método bastante

arcaico, que centraliza as informações apenas na loja matriz. Esse cenário cria uma dependência das filiais, que precisam constantemente entrar em contato com a matriz para organizar e confirmar suas entregas. Além disso, o chefe do depósito é forçado a um ciclo incessante de telefonemas para verificar as entregas agendadas, aumentando a complexidade e a possibilidade de erros.

Um dos principais problemas dessa abordagem é a falta de sincronização entre as filiais, o que leva a desencontros de informações e ineficiências operacionais. Outra questão crítica é a sobrecarga da equipe de expedição. Com os vendedores adiando frequentemente as entregas para o dia seguinte com o objetivo de maximizar as vendas, a equipe de expedição acaba sobrecarregada. Isso resulta em montagens apressadas e, conseqüentemente, em uma queda na qualidade do serviço oferecido. A inexistência de um sistema de gerenciamento centralizado também contribui para uma logística ineficiente e uma experiência ruim para o cliente.

Para superar esses desafios, propomos a prototipagem de um sistema unificado de gerenciamento de entregas. Esse sistema tem o objetivo de centralizar as informações de todas as filiais, permitindo que todos os envolvidos acessem os mesmos dados em tempo real. Um dos recursos mais impactantes do sistema é a capacidade do chefe do depósito de visualizar todas as entregas programadas e, se necessário, "bloquear" novos agendamentos para dias com alta demanda. Isso possibilita uma distribuição mais equilibrada das entregas, melhorando a logística e a qualidade do serviço.

A implementação desse sistema representa uma mudança significativa na maneira como a empresa gerencia suas entregas. A transição de um método antiquado para um processo digital e centralizado não apenas otimiza a logística, mas também resolve problemas de comunicação entre as filiais e a matriz. A interface do usuário, desenvolvida com base nos princípios de interação humano-computador, facilita a adaptação dos funcionários ao novo sistema, garantindo uma mudança suave e eficaz.

Em resumo, este estudo de caso ilustra como a prototipagem de interfaces para um sistema de controle de entregas pode resolver problemas reais em um ambiente empresarial fictício. Com a aplicação de tecnologia moderna, é possível superar as limitações de sistemas de gerenciamento tradicionais, trazendo eficiência, transparência e qualidade, e preparando a empresa para atender às demandas de um mercado em constante evolução.

4. Desenvolvimento: Requisitos do Projeto e Processo de Desenvolvimento

O desenvolvimento do projeto de interface de usuário para o sistema de controle de entregas de uma empresa de móveis fictícia começou com a definição clara dos requisitos. Estes requisitos foram categorizados em requisitos funcionais e não funcionais.

Requisitos Funcionais:

- Capacidade de visualizar e gerenciar agendamentos de entrega e montagem.

- Integração com um sistema de gerenciamento de estoque para verificar a disponibilidade dos produtos.
- Funcionalidades para adicionar, editar e excluir pedidos.
- Interface interativa para visualização de calendário e agendamentos.
- Opções para filtrar e buscar pedidos específicos.

Requisitos Não Funcionais:

- Usabilidade: Interfaces intuitivas e fáceis de usar.
- Desempenho: Respostas rápidas às interações do usuário.
- Escalabilidade: Capacidade de suportar um aumento no volume de pedidos.
- Estética: Design visual atraente, seguindo princípios de interação humano-computador.

Desenvolvimento de Protótipos

Utilizando a ferramenta Qt Designer, foram criados vários protótipos de interface, cada um apresentando diferentes aspectos do sistema de gerenciamento de entregas. Estes incluem:

Tela de Agendamento de Entregas:

Permite aos usuários programar e visualizar entregas, com um calendário interativo e opções para selecionar horários e datas.

Interface de Gerenciamento de Estoque:

Exibe informações sobre os móveis disponíveis para entrega, permitindo aos funcionários atualizar a situação do estoque.

Dashboard de Monitoramento de Entregas: Fornece uma visão geral das entregas agendadas, com funcionalidades para rastrear a situação em tempo real.

Cada protótipo foi projetado com foco na usabilidade, garantindo que as interfaces sejam tanto funcionais quanto esteticamente agradáveis. A intenção era criar uma representação visual de como o sistema de gerenciamento de entregas poderia ser operado, destacando a eficiência e a clareza na apresentação das informações.

Design Inspirado em Mural de Pedidos

Uma característica distintiva desta prototipagem é a representação visual dos pedidos como "cards" na interface, cuidadosamente projetados para assemelhar-se a um mural com post-its de entregas. Esta abordagem não é apenas esteticamente agradável, mas também funcional, refletindo uma prática comum em ambientes de logística e gerenciamento de projetos. Cada "card" representa um pedido, proporcionando informações essenciais de forma rápida e acessível, similar à experiência de visualizar

um mural físico de anotações. Este design permite que os usuários, com um simples olhar, obtenham um panorama dos agendamentos e situação das entregas.

Escolha Cuidadosa das Cores e Design

A seleção de cores para os "cards" e outros elementos da interface foi realizada com base em princípios de interação humano-computador, um tema estudado ao longo do curso. As cores foram escolhidas não só por sua harmonia e apelo estético, mas também por sua funcionalidade em termos de legibilidade e distinção visual. As cores suaves foram utilizadas para evitar a fadiga visual, enquanto as cores mais vibrantes marcavam elementos interativos importantes, como botões de ação. Rogers et al. (2013, p.8) definem design de interação como: “Projetar produtos interativos para apoiar o modo como as pessoas se comunicam e interagem em seus cotidianos, seja em casa ou no trabalho”.

Simplicidade e Clareza Visual

Na construção das interfaces, houve um esforço consciente para manter a simplicidade e evitar a poluição visual. Seguindo as melhores práticas de design visual, as interfaces foram projetadas para serem intuitivas e fáceis de usar, com um layout claro e uma hierarquia visual bem definida. Este aspecto foi crucial para garantir que os usuários pudessem navegar e utilizar o sistema eficientemente, sem sobrecarga de informações.

Feedback de Profissionais Reais

Embora o contexto do estudo de caso seja uma empresa fictícia, o feedback para o design da interface foi coletado de profissionais reais do setor. Consultas foram feitas com funcionários de uma loja real de móveis, incluindo o chefe do depósito, para garantir que as interfaces refletissem as necessidades e preferências de quem utilizaria o sistema no dia a dia. Essa abordagem baseada em feedback real foi fundamental para validar a eficácia do design da interface e garantir sua relevância prática.

Este capítulo reforça a importância de um design de interface bem pensado e baseado em feedback real. A utilização de métodos inspirados em práticas reais de gestão, a escolha cuidadosa de elementos visuais, e a consulta com profissionais do setor garantiram que os protótipos fossem não apenas teoricamente sólidos, mas também práticos e aplicáveis no mundo real. Esta abordagem exemplifica como a prototipagem de interfaces pode ser enriquecida por insights práticos, elevando o desenvolvimento de software a um nível que ressoa tanto com os usuários finais quanto com as necessidades do negócio.

Processo de Desenvolvimento

O desenvolvimento seguiu uma abordagem iterativa e incremental, dividida nas seguintes fases:

1. Concepção e Planejamento:

- Definição dos objetivos e escopo do projeto.
 - Análise inicial dos requisitos e planejamento das etapas de desenvolvimento.
 - Escolha das ferramentas: Python com PyQt5 e Qt Designer.
2. **Design da Interface:**
- Utilização do Qt Designer para criar protótipos visuais das interfaces.
 - Desenvolvimento de layouts iniciais, seleção de widgets e esquema de cores.
 - Revisão e validação dos designs com base em feedback de usuários potenciais.
3. **Implementação:**
- Conversão dos designs do Qt Designer em código Python utilizando ferramentas de conversão do PyQt5.
 - Desenvolvimento da lógica de negócios e integração com a interface de usuário.
 - Implementação de funcionalidades como visualização de calendário, gerenciamento de pedidos e interação com o sistema de estoque.
4. **Testes e Iterações:**
- Testes contínuos para identificar e corrigir problemas.
 - Avaliação da usabilidade com usuários reais e ajustes com base em seu feedback.
 - Iterações para refinar funcionalidades e interface de usuário.
5. **Avaliação e Melhoria Contínua:**
- Análise do desempenho e da escalabilidade do sistema.
 - Revisões regulares para melhorar a usabilidade e adicionar novas funcionalidades.

Interface Intuitiva para Gerenciamento de Agendamentos:

- O uso do QCalendarWidget permite aos usuários interagir com um calendário visual para selecionar datas, fundamental para o gerenciamento de agendamentos.
- Botões como pushButton_2 e pushButton_4 são projetados para ações específicas, como atualizar informações ou adicionar novos pedidos, simplificando a interação do usuário.

Visualização e Edição de Pedidos:

- A implementação de "cards" para representar pedidos, utilizando QDockWidget e layouts personalizados, oferece uma maneira clara e organizada de apresentar informações importantes.
- Funcionalidades para adicionar (adicionar_cartao), editar (editar_cartao) e excluir (remover_cartao) pedidos demonstram uma interface interativa e dinâmica.

Estética e Usabilidade:

- A estética cuidadosamente planejada do sistema é evidente na seleção de cores e no design dos widgets, que segue princípios de interação humano-computador.
- A disposição dos elementos na interface, como a separação visual de diferentes seções com QFrame e o uso de espaçamentos, contribui para uma experiência de usuário descomplicada e agradável.

Design Inicial e Prototipagem:

Utilizando o Qt Designer, os protótipos iniciais foram criados com foco em uma interface limpa e funcional. A estrutura da classe Ui_MainWindow reflete este design, organizando os widgets e layouts de maneira lógica e eficiente.

Codificação e Integração de Funcionalidades:

Cada função no código, como update_label_text e adicionar_cartao, foi cuidadosamente escrita para atender aos requisitos funcionais específicos. Isso incluiu a implementação de lógicas para interações do usuário, como a seleção de datas ou a manipulação de pedidos.

Testes Iterativos e Refinamento:

Testes contínuos foram realizados para garantir que cada elemento da interface funcionasse conforme o esperado. Os feedbacks dos usuários foram cruciais nessa fase, levando a ajustes iterativos no design e na funcionalidade.

Avaliação de Usabilidade e Desempenho:

O sistema foi avaliado não apenas em termos de funcionalidade, mas também de usabilidade. Isso garantiu que a interface não só cumprisse sua função, mas também fosse acessível e agradável de usar.

Aprimoramento Contínuo e Adaptação:

O projeto foi mantido flexível para acomodar mudanças e melhorias, um princípio refletido na modularidade e na clareza do código.

Através de um processo cuidadoso de design, prototipagem, codificação e testes iterativos, o projeto desenvolveu uma interface de usuário que não apenas cumpre os requisitos funcionais, mas também encanta os usuários com sua usabilidade e estética.

O código-fonte fornecido é um testemunho da abordagem metódica e centrada no usuário adotada, ilustrando como cada elemento da interface foi projetado e implementado com um propósito específico em mente.

Este projeto serve como um exemplo exemplar de como a teoria pode ser traduzida em prática no desenvolvimento de software, especialmente no contexto de interfaces de usuário interativas e atraentes.

Durante o desenvolvimento do sistema de controle de entregas, encontramos uma série de desafios que foram cruciais para o nosso crescimento e aprendizado. Alguns dos mais significativos incluíram:

- **Complexidade na Integração de Widgets:** A integração e o funcionamento harmonioso de múltiplos widgets do PyQt5, como QCalendarWidget e QDockWidget, apresentaram uma curva de aprendizado íngreme. Coordenar a interação entre diferentes componentes da interface e garantir a consistência dos dados foi um desafio técnico que exigiu uma compreensão profunda do PyQt5.

- **Design de Interface Intuitiva:** Manter a interface do usuário intuitiva, enquanto incorporava funcionalidades complexas, foi outro desafio. Encontrar o equilíbrio entre uma interface rica em recursos e uma experiência de usuário descomplicada exigiu várias iterações de design e feedback dos usuários.

- **Otimização de Desempenho:** À medida que o sistema se tornava mais complexo, o desempenho se tornou uma preocupação. Garantir que o sistema respondesse rapidamente às ações do usuário.

Apesar dos desafios, o projeto proporcionou aprendizados significativos, incluindo:

- **Proficiência no Uso do PyQt5 e Qt Designer:** Através da prática e experimentação, adquirimos uma compreensão aprofundada do PyQt5 e do Qt Designer. Aprendemos a explorar eficientemente os recursos dessas ferramentas para criar interfaces de usuário atrativas e funcionais.

- **Importância do Design Centrado no Usuário:** Compreendemos a importância de um design centrado no usuário, o que envolve não apenas a estética, mas também a usabilidade e a acessibilidade. O feedback dos usuários foi fundamental para refinar a interface.

- **Valor da Iteração e Flexibilidade:** A abordagem iterativa e flexível no desenvolvimento provou ser essencial. Estar aberto a mudanças e ajustes conforme recebíamos feedback nos permitiu melhorar continuamente a qualidade do sistema.

Reconhecemos que certos aspectos do projeto poderiam ter sido abordados de maneira mais eficaz:

- **Testes Mais Abrangentes:** Embora tenhamos realizado testes durante o desenvolvimento, testes mais rigorosos e abrangentes, especialmente em diferentes ambientes e condições de uso, poderiam ter nos ajudado a identificar e resolver problemas mais rapidamente.
- **Planejamento e Gerenciamento de Projeto:** Uma gestão de projeto mais estruturada desde o início teria ajudado a antecipar e mitigar desafios, como por exemplo, a modularização do código fonte em partes, contribuindo assim para futuras atualizações e manutenção do projeto.
- **Exploração de Novas Tecnologias:** A exploração de outras ferramentas e tecnologias complementares ao PyQt5 poderia ter enriquecido ainda mais o projeto, tanto em termos de funcionalidades quanto de desempenho.

5. Considerações Finais

Em suma, o desenvolvimento deste sistema foi uma jornada enriquecedora, cheia de desafios e aprendizados. Cada obstáculo enfrentado foi uma oportunidade de crescimento e aprimoramento de nossas habilidades. Este projeto não apenas fortaleceu nossa compreensão técnica do desenvolvimento de interfaces de usuário, mas também nos ensinou sobre a importância da flexibilidade, do design centrado no usuário e do planejamento cuidadoso no desenvolvimento de software. Os insights e experiências adquiridas neste projeto serão inestimáveis para nossos futuros empreendimentos no campo da tecnologia da informação.

Este projeto de desenvolvimento de uma interface para o sistema de controle de entregas de móveis não foi apenas um exercício prático, mas uma oportunidade para aplicar e vivenciar os conhecimentos adquiridos ao longo do nosso curso. Cada aspecto do projeto refletiu as lições aprendidas em diferentes disciplinas acadêmicas, integrando-as em um contexto prático e realista.

Utilizando as práticas aprendidas em Engenharia de Software, empregamos ferramentas de gerenciamento de projetos como o Trello para organizar e distribuir tarefas entre os membros da equipe. O método Kanban, em particular, provou ser essencial para acompanhar o progresso e manter todos os envolvidos alinhados com os objetivos do projeto. Esta abordagem estruturada permitiu uma gestão eficiente do projeto, facilitando a colaboração e a comunicação entre a equipe.

A disciplina de Engenharia de Software também nos introduziu ao gerenciamento de versões, uma habilidade crucial no desenvolvimento de software. A utilização de sistemas de controle de versão, como Git, foi fundamental para manter um histórico de todas as alterações realizadas, permitindo-nos rastrear e revisar o desenvolvimento do sistema eficazmente. Isso não só minimizou o risco de perder progressos importantes, mas também facilitou a experimentação e o teste de novas ideias sem comprometer a estabilidade do sistema.

A disciplina de Interação Humano-Computador foi particularmente relevante para o design da nossa interface. As lições aprendidas sobre usabilidade, acessibilidade e design centrado no usuário guiaram nossas decisões de design, ajudando-nos a criar uma interface que não apenas atendesse às necessidades funcionais, mas que também fosse intuitiva e agradável para os usuários. Este conhecimento foi crucial para transformar princípios teóricos de design de interface em uma aplicação prática e eficaz.

Nossos estudos em Algoritmos e Programação formaram a base do nosso entendimento técnico. Eles nos proporcionaram as habilidades fundamentais necessárias para navegar no mundo da programação e nos aventurar com confiança na utilização de Python e PyQt5. Esta base sólida foi essencial para abordar os desafios de programação do projeto e implementar as funcionalidades desejadas com eficiência.

Cada disciplina do nosso curso contribuiu de alguma forma para o projeto. Seja nos fornecendo o conhecimento teórico necessário, seja nos ensinando a aplicar esses conceitos em situações práticas, cada matéria desempenhou um papel crucial em nossa jornada de aprendizado e desenvolvimento.

Este projeto foi uma síntese de todo o conhecimento e habilidades adquiridos durante o nosso curso. Ele não apenas nos desafiou a aplicar teorias em práticas concretas, mas também nos proporcionou uma compreensão mais profunda e integrada do que significa ser um profissional na área de tecnologia da informação. Através deste projeto, vivenciamos o verdadeiro significado de transformar conhecimento acadêmico em soluções práticas e eficazes, preparando-nos para os desafios e oportunidades que encontraremos em nossas carreiras futuras.

Ao chegarmos ao final deste projeto, torna-se evidente que, na realidade, estamos no início de uma jornada muito mais ampla. Este projeto não representa o término, mas o começo de um contínuo processo de aprendizado e desenvolvimento profissional. Compreendemos agora que o mundo da tecnologia da informação e do desenvolvimento de software é dinâmico e em constante evolução, e permanecer atualizado e engajado com novos aprendizados é fundamental para o sucesso na área. Este TCC, mais do que um projeto acadêmico, foi uma porta de entrada para um mundo onde o aprendizado nunca termina.

Além de ser um projeto acadêmico, este sistema de controle de entregas de móveis tem potencial para se tornar um produto real e funcional. Durante o desenvolvimento, nos deparamos com desafios e oportunidades que são comuns no desenvolvimento de produtos de software no mundo real. Isso nos proporcionou uma experiência valiosa e uma visão sobre o que é necessário para transformar uma ideia em um produto comercializável.

Nós, membros da equipe deste projeto, estamos agora considerando seriamente a possibilidade de continuar a desenvolver e aprimorar este sistema além do ambiente acadêmico. Com o conhecimento e a experiência que adquirimos, vemos o potencial de transformar este TCC em um produto final que poderia ser implementado em uma empresa real. Esta perspectiva não apenas valida o esforço e a dedicação que colocamos no projeto, mas também nos motiva a continuar a explorar e inovar na área de desenvolvimento de software.

A ideia de levar o projeto para o mercado real é emocionante e desafiadora. Isso requer não apenas um aprofundamento técnico, mas também uma compreensão das demandas do mercado, estratégias de negócios e habilidades de empreendedorismo. Estamos prontos para explorar essas novas áreas, aplicar nossos conhecimentos e habilidades em um contexto mais amplo e ver até onde podemos levar esse projeto.

Este TCC marca não o fim, mas o início de um caminho promissor e enriquecedor. Através dele, não apenas alcançamos um marco acadêmico importante, mas também abrimos portas para futuras oportunidades profissionais. Estamos empolgados para continuar nossa jornada, levando os aprendizados deste projeto para novos horizontes e, potencialmente, transformando nosso trabalho acadêmico em um produto inovador e útil no mundo real.

Ao refletirmos sobre o percurso do nosso projeto, fica evidente que este TCC foi muito mais do que um simples requisito acadêmico; foi uma jornada de crescimento pessoal e profissional. Cada desafio que enfrentamos, cada problema que resolvemos e cada feedback que incorporamos, moldaram não apenas o nosso projeto, mas também nós mesmos como futuros profissionais da área de tecnologia. Aprendemos que o verdadeiro valor de um projeto acadêmico reside na sua capacidade de nos preparar para os desafios reais, ensinando-nos a pensar criticamente, agir com flexibilidade e inovar com propósito.

Este projeto transcende os limites de um trabalho de conclusão de curso. Ele simboliza o início de uma jornada promissora no vasto campo da tecnologia da informação. A possibilidade de transformar nosso projeto em um produto real para o mercado é não apenas um objetivo ambicioso, mas também um testemunho do nosso comprometimento e paixão pela área de desenvolvimento de software. Vemos este projeto como um trampolim para futuras inovações e uma oportunidade para causar um impacto real no setor de tecnologia.

À medida que avançamos, nosso compromisso com a inovação e a excelência permanece firme. Estamos determinados a continuar nosso aprendizado, aprimorar nossas habilidades e expandir nosso conhecimento. O projeto nos ensinou a importância de se adaptar às mudanças, de estar abertos a novas ideias e de buscar continuamente melhorias. Estamos prontos para levar essas lições adiante, aplicando-as não apenas no desenvolvimento deste sistema, mas em todos os nossos futuros empreendimentos tecnológicos.

Encaramos o futuro como uma oportunidade infinita. Com o término deste TCC, estamos ansiosos para explorar novas fronteiras, enfrentar novos desafios e deixar nossa marca no mundo da tecnologia. Seja continuando o desenvolvimento deste sistema ou embarcando em novos projetos, estamos confiantes de que as habilidades e experiências que adquirimos nos capacitarão a fazer contribuições significativas para a tecnologia e a sociedade. Em resumo, este TCC é apenas o começo. É o início de uma jornada emocionante em um campo em constante evolução, repleto de oportunidades para inovar, aprender e crescer. Com os olhos no futuro, estamos prontos para abraçar o desconhecido, alimentados pela paixão e pelo desejo incansável de criar, inovar e impactar positivamente o mundo ao nosso redor. Este projeto nos preparou não apenas como desenvolvedores de software, mas como pensadores criativos e solucionadores de

problemas, prontos para enfrentar os desafios de um mundo em constante transformação.

References

MENEZES, Nilo Ney Coutinho. Introdução à Programação com Python: Algoritmos e Lógica de Programação Para Iniciantes. 2019. v. 3ª Edição, TCC (Graduação) - Curso de Sistemas de Informação, Universidade Federal, Mato Grosso do Sul, 2024.

SOMMERVILLE, Ian. Engenharia de Software. 2011. v. 10ª Edição, TCC (Graduação) - Curso de Sistemas de Informação, Universidade Federal, Mato Grosso do Sul, 2024.

ROGERS, Yvonne; SHARP, Helen; PREECE, Jennifer. Design de Interação: Além da Interação Humano-Computador. 2013. v. 3ª Edição, TCC (Graduação) - Curso de Sistemas de Informação, Universidade Federal, Mato Grosso do Sul, 2024.