|  |  |  |
| --- | --- | --- |
|  | **UNIVERSIDADE FEDERAL DA BAHIA**   1. **ESCOLA POLITÉCNICA** 2. **COLEGIADO DO CURSO DE ENG. ELÉTRICA** |  |

1. **João Pedro Caires Ferreira**
2. **SIMULOAD: Simulador de Curvas de Carga**
3. **Orientador: Prof. Dr. André Luiz de Carvalho Valente**
4. **Salvador-Ba - Brasil**
5. **30 de Junho de 2023**
6. **João Pedro Caires Ferreira**
7. **SIMULOAD: Simulador de Curvas de Carga**

Projeto apresentado ao Curso de Graduação em Engenharia Elétrica da Universidade Federal da Bahia como parte dos requisitos para a obtenção do grau de Engenheiro(a) Eletricista.

1. **Orientador: Prof. Dr. André Luiz de Carvalho Valente**
2. **Salvador-Ba - Brasil**
3. **14 de Julho de 2023**
4. **João Pedro Caires Ferreira**
5. **SIMULOAD: Simulador de Curvas de Carga**

Projeto apresentado ao Curso de Graduação em Engenharia Elétrica da Universidade Federal da Bahia como parte dos requisitos para a obtenção do grau de Engenheiro(a) Eletricista.

Trabalho aprovado. Salvador-Ba – Brasil, 14 de Julho de 2023:

**-------------------------------------------------------**

**Orientador(a): Prof. Dr. André Luiz de Carvalho Valente**

**-------------------------------------------------------**

**Prof. Dr. W**

**------------------------------------------------------**

**Prof. Dr. X**

1. **Salvador-Ba – Brasil**
2. **14 de Julho de 2023**
3. *Este trabalho é dedicado...*
4. **Agradecimentos**

Agradeço por ter saúde...

“Liberdade não é fazer o que se quer, mas querer o que se faz.”

(Jean-Paul Sartre)

1. **Resumo**

O resumo deve ressaltar o objetivo, o método, os resultados e as conclusões do documento.

Video provides a powerful way to help you prove your point. When you click Online Video, you can paste in the embed code for the video you want to add. You can also type a keyword to search online for the video that best fits your document.

To make your document look professionally produced, Word provides header, footer, cover page, and text box designs that complement each other. For example, you can add a matching cover page, header, and sidebar. Click Insert and then choose the elements you want from the different galleries.

**Palavras-chave**: Word, CCEE, Politécnica.

1. **Abstract**

This is the english abstract.

**Keywords**: Word, CCEE, Politécnica.

**Lista de Figuras**

[*Fig. 1 Sistema de controle.* 9](#Fig.!0|sequence)

[*Fig. 2 Sistema de controle.* 10](#Fig.!1|sequence)

[*Fig. 3 Sistema de controle.* 11](#Fig.!2|sequence)

[*Fig. 4 Sistema de controle.* 12](#Fig.!3|sequence)

[*Fig. 5 Sistema de controle.* 13](#Fig.!4|sequence)

[*Fig. 6 Sistema de controle.* 17](#Fig.!5|sequence)

[*Fig. 7 Sistema de controle.* 17](#Fig.!6|sequence)

[*Fig. 8 Sistema de controle.* 18](#Fig.!7|sequence)

[*Fig. 9 Sistema de controle.* 18](#Fig.!8|sequence)

[*Fig. 10 Sistema de controle.* 19](#Fig.!9|sequence)

[*Fig. 11 Sistema de controle.* 20](#Fig.!10|sequence)

[*Fig. 12 Sistema de controle.* 21](#Fig.!11|sequence)

[*Fig. 13 Sistema de controle.* 22](#Fig.!12|sequence)

[*Fig. 14 Sistema de controle.* 23](#Fig.!13|sequence)

[*Fig. 15 Sistema de controle.* 24](#Fig.!14|sequence)

[*Fig. 16 Sistema de controle.* 25](#Fig.!15|sequence)

[*Fig. 17 Sistema de controle.* 26](#Fig.!16|sequence)

[*Fig. 18 Sistema de controle.* 26](#Fig.!17|sequence)

[*Fig. 19 Sistema de controle.* 28](#Fig.!18|sequence)

[*Fig. 20 Sistema de controle.* 29](#Fig.!19|sequence)

[*Fig. 21 Sistema de controle.* 30](#Fig.!20|sequence)

1. **Lista de Tabelas**

**Lista de Abreviaturas e Siglas**

**Lista de Símbolos**

Sumário

[Capítulo 1: Introdução 1](#__RefHeading___Toc783_3772085703)

[1.1 Simuload 2](#__RefHeading___Toc785_3772085703)

[Capítulo 2: Objetivos 3](#__RefHeading___Toc789_3772085703)

[2.1 Objetivo Geral 3](#__RefHeading___Toc791_3772085703)

[2.2 Objetivos Específicos 4](#__RefHeading___Toc793_3772085703)

[2.2.1 Requisitos Funcionais. 5](#__RefHeading___Toc795_3772085703)

[Capítulo 3: Metodologia 6](#__RefHeading___Toc797_3772085703)

[3.1 Desenvolvimento do Software 6](#__RefHeading___Toc799_3772085703)

[3.2 Validação 7](#__RefHeading___Toc550_3461207297)

[Capítulo 4: Especificações 8](#__RefHeading___Toc803_3772085703)

[4.1 Módulos e Ferramentas 8](#__RefHeading___Toc805_3772085703)

[4.1.1 Ferramenta de Interface Visual 8](#__RefHeading___Toc807_3772085703)

[4.1.2 Ferramentas Matemáticas 9](#__RefHeading___Toc552_3461207297)

[4.1.3 Banco de Dados e Armazenamento 9](#__RefHeading___Toc554_3461207297)

[4.2 Mapeamento de Dados 12](#__RefHeading___Toc556_3461207297)

[4.3 Algoritmo de Aleatoriedade 13](#__RefHeading___Toc558_3461207297)

[Capítulo 5: Estudo de Caso 16](#__RefHeading___Toc809_3772085703)

[5.1 Simulando com outros Softwares 16](#__RefHeading___Toc811_3772085703)

[5.2 Aplicação do Simuload 19](#__RefHeading___Toc560_3461207297)

[5.3 Resultados e comparações 29](#__RefHeading___Toc562_3461207297)

[Capítulo 6: Conclusão 31](#__RefHeading___Toc815_3772085703)

[6.1.1 Título da primeira seção do sexto capítulo 31](#__RefHeading___Toc817_3772085703)

[6.1.2 Título da subseção do sexto capítulo. 32](#__RefHeading___Toc819_3772085703)

[Referências 33](#__RefHeading___Toc821_3772085703)

[Apêndice A – Código do algoritmo de aleatoriedade 34](#__RefHeading___Toc823_3772085703)

# **Introdução**

Sabe-se que no atual cenário brasileiro no que diz respeito ao consumo de energia elétrica, há uma grande necessidade de estudar formas efetivas de aproveitamento dos recursos e suas distribuições. A população brasileira, de uma forma geral, não faz uso consciente da energia elétrica, pois presume que a capacidade e o potencial energético do país é abundante, sendo seu único freio a fator financeiro. As formas de coletar, mensurar, analisar e utilizar os dados referentes ao consumo de energia são de extrema importância para o país e portanto devem ser abordadas de forma crítica, sobretudo dentro da universidade no curso de engenharia elétrica.

Em uma perspectiva mais geral, o Programa Nacional de Conservação de Energia Elétrica - Procel, foi criado pelo governo com intuito de promover o uso eficiente da energia elétrica e combater o seu desperdício. O programa possui ações que auxiliam na eficiência dos bens e serviços, bem como possibilitam a postergação de investimentos no setor elétrico, reduzindo assim também os possíveis impactos ambientais. Tais ações são fundamentais para a manutenção de um sistema de distribuição eficiente e que vise maior economia e uso consciente dos nossos recursos.

Nessa conjuntura é importante salientar que a simulação de curvas de carga é uma atividade fundamental no planejamento da distribuição de energia elétrica, permitindo que as empresas do setor controlem a demanda de energia de forma eficiente e segura. Com a simulação de curvas de carga, é possível determinar o perfil de consumo de uma determinada rede de distribuição, identificar pontos críticos que podem gerar sobrecargas e avaliar o desempenho da rede em situações de pico de demanda. Ou seja, é parte intrínseca na construção de um planejamento energético eficiente, independente da sua dimensão ou âmbito.

Este é um campo de atuação dentro da engenharia elétrica que permite uma infinidade de possibilidades para o profissional da área. Portanto, é necessário que o estudante, ainda na universidade, tenha contato com ferramentas que contribuam de forma eficiente na sua formação, fazendo com que ele tenha domínio tanto na execução quanto na análise dos dados coletados. A simulação de curvas de carga é uma atividade cujos seus resultados e análises podem ser exploradas tanto no setor público, quanto privado, o que permite ampliar de maneira ainda mais significativa as possibilidades de atuação dentro do mercado de trabalho.

## Simuload

O software Simuload, produto desenvolvido como objeto de estudo deste trabalho, foi pensado e elaborado para ser uma ferramenta importante no aprendizado da simulação de curvas de carga em redes de distribuição de baixa tensão, pois oferece uma maneira eficiente de simular curvas de carga, evitando excesso de trabalho manual e consequentemente conectando o estudante com a parte prática. Durante o estudo de caso feito com o Simuload, foi observada a importância do software no desenvolvimento das habilidades técnicas dos estudantes, que são de extrema importância no campo de atuação da Engenharia Elétrica.

O objetivo deste trabalho de conclusão de curso é apresentar o software Simuload, sua metodologia de desenvolvimento, sua eficácia como ferramenta didática para simulação de curvas de carga em redes de distribuição de baixa tensão, bem como apresentar os resultados obtidos através do experimento. Para isso, o software foi utilizado por uma dupla de alunos em um estudo de caso real, que realizou uma análise dos resultados obtidos através do uso do Simuload na simulação de curvas de carga em uma rede de distribuição de energia elétrica, comparando-os com outros métodos mais tradicionais utilizados para simulação de curvas de carga.

O trabalho está estruturado da seguinte forma: após esta introdução, será apresentada uma revisão bibliográfica sobre a simulação de curvas de carga em redes de distribuição de energia elétrica, seguida pela descrição detalhada do software Simuload e sua metodologia de desenvolvimento. Com esta pesquisa, espera-se contribuir para o desenvolvimento de ferramentas didáticas eficientes para o ensino de simulação de curvas de carga em redes de distribuição de baixa tensão, promovendo uma formação mais completa e qualificada para os estudantes de Engenharia Elétrica.

# **Objetivos**

O capítulo de objetivos tem como propósito definir de forma clara e concisa os resultados almejados com o desenvolvimento e utilização do software Simuload. Neste capítulo, serão apresentados os objetivos gerais e específicos do trabalho, proporcionando uma visão abrangente das metas a serem alcançadas. Os objetivos estabelecidos serão fundamentais para direcionar o desenvolvimento do software e guiar a análise e avaliação dos resultados obtidos.

## **Objetivo Geral**

O objetivo geral deste trabalho é apresentar o software Simuload e sua metodologia de desenvolvimento, bem como avaliar sua eficácia como ferramenta didática para a simulação de curvas de carga em redes de distribuição de baixa tensão. Para isso, foram realizadas simulações de curvas de carga em uma rede de distribuição de energia elétrica, utilizando o Simuload como ferramenta principal, a fim de demonstrar sua capacidade de simulação de uma forma didática.

A criação do software Simuload foi motivada pela necessidade da existência de uma ferramenta didática que pudesse auxiliar na simulação de curvas de carga em redes de distribuição de baixa tensão, a ser utilizada sobretudo por estudantes de Engenharia Elétrica. Sua interface foi desenvolvida de maneira que se apresentasse de forma intuitiva e de fácil utilização para os usuários no momento da simulação das curvas de carga, permitindo ao usuário otimizar o trabalho e consequentemente concluir a tarefa de forma mais rápida e eficiente.

Por meio da utilização do Simuload, espera-se que seja possível contribuir para o desenvolvimento de habilidades técnicas importantes para a atuação na área de Engenharia Elétrica. Dentre essas habilidades pode-se destacar: a capacidade de simular e analisar curvas de carga em redes de distribuição de energia elétrica e a capacidade de compreender e aplicar conceitos relacionados à gestão de demanda de energia elétrica, o que é de extrema importância no campo prático de atuação, pois através dos dados obtidos com a simulação o profissional desta área deve ser capaz de planejar maneiras eficientes de lidar com as demandas de consumo.

Com isso, espera-se também que este trabalho possa contribuir para a disseminação do conhecimento sobre a simulação de curvas de carga em redes de distribuição de energia elétrica, bem como para o aprimoramento da formação de estudantes e profissionais da área de Engenharia Elétrica.

Por meio da utilização do Simuload, espera-se que seja possível contribuir com o desenvolvimento de habilidades técnicas importantes para a atuação dos profissionais na área de Engenharia Elétrica.

Por conseguinte, espera-se que este trabalho possa contribuir para a disseminação do conhecimento sobre a simulação de curvas de carga em redes de distribuição de energia elétrica, bem como sua importância para o aprimoramento da formação de estudantes e profissionais da área de Engenharia Elétrica, sobretudo no que diz respeito ao campo prático de atuação.

O objetivo geral é, portanto, avaliar a eficiência e usabilidade do software Simuload para facilitar o processo de estudo e aprendizagem de curvas de carga, e os objetivos específicos visam a identificação de limitações e proposição de melhorias para aprimorar o software.

## **Objetivos Específicos**

Os objetivos específicos deste trabalho incluem:

* Avaliação da eficiência e usabilidade do software Simuload para a simulação de curvas de carga;
* Identificação de possíveis limitações ou dificuldades no uso do software;
* Proposição de melhorias e aprimoramentos para torná-lo ainda mais didático e útil para fins de aprendizagem.

Para tanto, serão realizadas simulações de curvas de carga utilizando o software Simuload em diferentes cenários, e os resultados obtidos serão utilizados para avaliar a eficiência do software em facilitar o processo de elaboração e estudo de curvas de carga. Além disso, serão realizadas análises para identificar possíveis limitações ou dificuldades no uso do software, de forma a propor melhorias e aprimoramentos que possam torná-lo ainda mais didático e útil para fins de aprendizagem.

# **Metodologia**

Neste capítulo será descrito como o software Simuload foi desenvolvido, incluindo linguagem de programação, ferramentas utilizadas e mapeamento do banco de dados. Além disso, será apresentado o processo de validação da ferramenta, que consiste em avaliar a sua eficácia em simular curvas de carga e atender às necessidades dos estudantes. O Simuload é uma ferramenta desenvolvida para facilitar o processo de elaboração, estudo e aprendizagem de curvas de carga, e a sua metodologia de desenvolvimento é fundamental para garantir a qualidade e eficácia da ferramenta.

## Desenvolvimento do Software

Durante o processo de desenvolvimento do Simuload foram estabelecidos alguns requisitos funcionais com o objetivo de tornar a ferramenta prática e completa para o uso dos estudantes. Esses requisitos foram:

1. Desenvolver botões, telas e gráficos de curva totalmente funcionais, permitindo a criação de simulações de curva de carga de redes de distribuição de baixa tensão de forma simples e intuitiva;
2. Criação de um banco de dados funcional com integridade dos dados das entidades, garantindo a correta representação dos dados utilizados nas simulações de curva de carga;
3. Assegurar a integridade dos dados nas curvas de carga simuladas, verificando a precisão das simulações e a consistência dos resultados obtidos;
4. Incluir modelos de exemplo padrões mapeados do PPH 2019, relatório do Procel, que permitam ao usuário realizar simulações com dados reais e comparar os resultados obtidos com os dados já consolidados;
5. Calcular a faixa de carregamento do transformador, possibilitando a verificação das limitações de carga em um determinado transformador;
6. Permitir a exportação das simulações para um arquivo .CSV, para que os resultados possam ser facilmente analisados e compartilhados.

Esses seis itens foram pensados levando em consideração a experiência vivida nas matérias Laboratório Integrado VI e Transmissão e Distribuição de Energia Elétrica. Ambas fazem parte da grade curricular do sétimo semestre do curso de Engenharia Elétrica da UFBA e são de extrema importância na formação profissional. Visando maior aproveitamento e crescimento dos estudantes nessa fase crucial do curso, todo o conteúdo e trocas vivenciadas nesses dois componentes curriculares serviram de insumos para o processo de desenvolvimento do Simuload.

## **Validação**

Após a etapa de desenvolvimento e criação, se fez necessária a validação da efetividade da ferramenta. Para tanto, foi realizada uma avaliação qualitativa do seu uso por uma dupla de estudantes no contexto da disciplina Laboratório Integrado VI. Os critérios avaliados incluíram a facilidade de uso, a clareza dos resultados gerados, a compreensão da interface e a adequação dos modelos de exemplo. A avaliação teve como objetivo identificar pontos fortes e fracos da ferramenta e fornecer subsídios para possíveis melhorias. Os resultados desta avaliação serão discutidos de maneira mais abrangente no capítulo cinco, no qual será abordado o estudo de caso de maneira mais detalhada.

Foi estabelecido um diálogo constante com a dupla de estudantes que foi escolhida para utilizar o software em simulações de curvas de carga, possibilitando maior compreensão e aproximação das reais necessidades dos estudantes durante todo o processo de desenvolvimento. Dessa maneira, houve uma aproximação das funcionalidades do software com as principais necessidades dos estudantes, o que tornou o Simuload uma ferramenta bastante didática e eficaz no que se propõe. Essa abordagem colaborativa foi de extrema importância na criação do software, sobretudo por contribuir com maior efetividade no alcance dos objetivos específicos e consequentemente na no objetivo geral do trabalho.

# **Especificações**

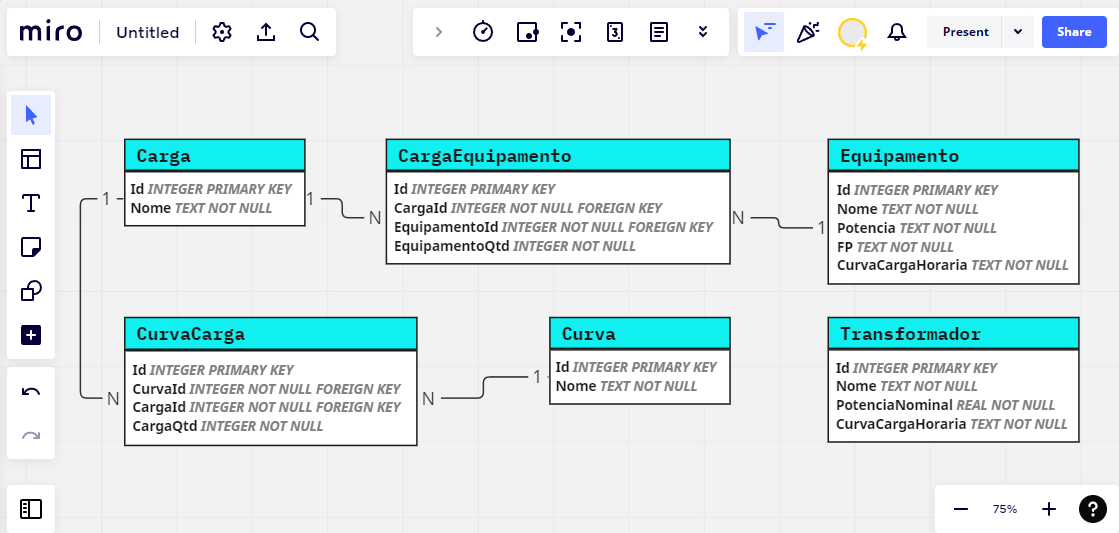
Nesta seção, são apresentadas as especificações técnicas do software Simuload, abrangendo desde os módulos e ferramentas utilizados até o mapeamento dos dados e o algoritmo de aleatoriedade implementado. Serão discutidas as decisões tomadas em relação às funcionalidades, os requisitos matemáticos e estatísticos, bem como a estrutura do banco de dados utilizado. A seção de especificação é fundamental para compreender as escolhas técnicas feitas ao longo do desenvolvimento do Simuload, garantindo a transparência e a compreensão das decisões tomadas para construir uma ferramenta robusta e eficiente.

## **Banco de Dados**

O Simuload utiliza um banco de dados relacional que inclui quatro entidades: Equipamento, Carga, Curva e Transformador. A entidade Equipamento representa os equipamentos que compõem uma carga em Watts [W]. Já a entidade Carga representa um estabelecimento consumidor em Watts [W], podendo ser uma casa, um galpão, um mercado, iluminação pública, entre outros. A entidade Curva representa a distribuição temporal do consumo de várias cargas e, consequentemente, o consumo de energia elétrica numa região atendida por um transformador. Por fim, a entidade Transformador representa o fornecimento na curva característica de um transformador que atende uma região.

As entidades no banco de dados são relacionadas entre si. A entidade Equipamento tem uma relação de muitos para muitos com a entidade Carga, e para isso existe uma tabela associativa CargaEquipamento que relaciona quais Equipamentos fazem parte de quais Cargas. A entidade Carga tem uma relação de muitos para muitos com a entidade Curva, e para isso também existe uma tabela associativa CurvaCarga que relaciona quais Curvas possuem quais Cargas. A entidade Transformador é independente e possui uma estrutura de demanda distribuída ao longo do tempo, assim como o equipamento. Essas relações permitem que os dados sejam organizados e recuperados de forma eficiente no Simuload e estão representadas no gráfico UML (Unified Modeling Language) abaixo, desenhado através da ferramenta Miro.

*Fig. 1 Gráfico UML do Banco de Dados desenhado na ferramenta Miro.*



## **Módulos e Ferramentas**

Na fase de planejamento e desenvolvimento do Simuload, foram tomadas decisões estratégicas em relação aos módulos e ferramentas a serem utilizados. Para garantir a eficiência e o funcionamento adequado do sistema, uma pesquisa detalhada foi realizada para identificar os pacotes e ferramentas mais adequados às necessidades do projeto.

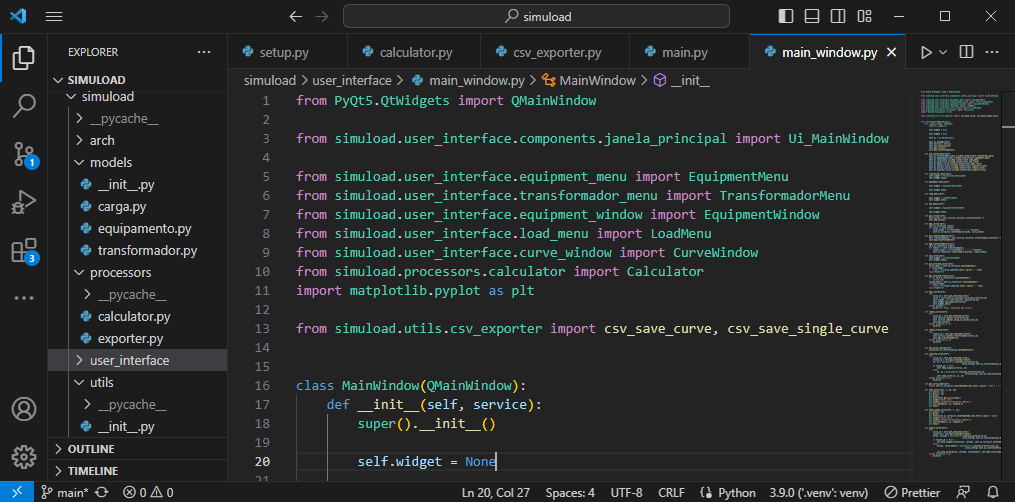
Durante essa pesquisa, diversos critérios foram considerados, como as funcionalidades oferecidas pelos módulos, a facilidade de uso, a popularidade no meio de desenvolvimento Python e a disponibilidade de material de apoio, como documentação e comunidades de suporte. Esses critérios foram cruciais para selecionar as melhores opções que atendessem às demandas específicas do Simuload.

A escolha criteriosa dos módulos e ferramentas utilizados é fundamental para o sucesso do projeto, pois eles fornecem recursos e funcionalidades essenciais para a implementação das diversas funcionalidades do software. Através dessa abordagem, foi possível garantir a qualidade, a robustez e a efetividade do Simuload, proporcionando uma experiência aprimorada ao usuário e facilitando o desenvolvimento das atividades propostas.

### *Ferramenta de Edição de Código.*

Dentre as várias ferramentas disponíveis para a edição de código, optou-se por utilizar o Visual Studio Code (VSCode) como o principal editor de código para o desenvolvimento do Simuload. O VSCode foi selecionado devido à sua ampla popularidade, reconhecido como um dos principais editores de código para desenvolvedores Python, oferecendo uma experiência de edição altamente produtiva.

*Fig. 2 Sistema de controle.*

A escolha do VSCode baseou-se em diversos fatores. Em primeiro lugar, sua interface intuitiva e amigável proporciona uma curva de aprendizado suave, permitindo aos desenvolvedores se familiarizarem rapidamente com o editor. Além disso, o VSCode possui suporte para a linguagem Python, o que facilita a escrita, depuração e execução de código Python diretamente no editor.

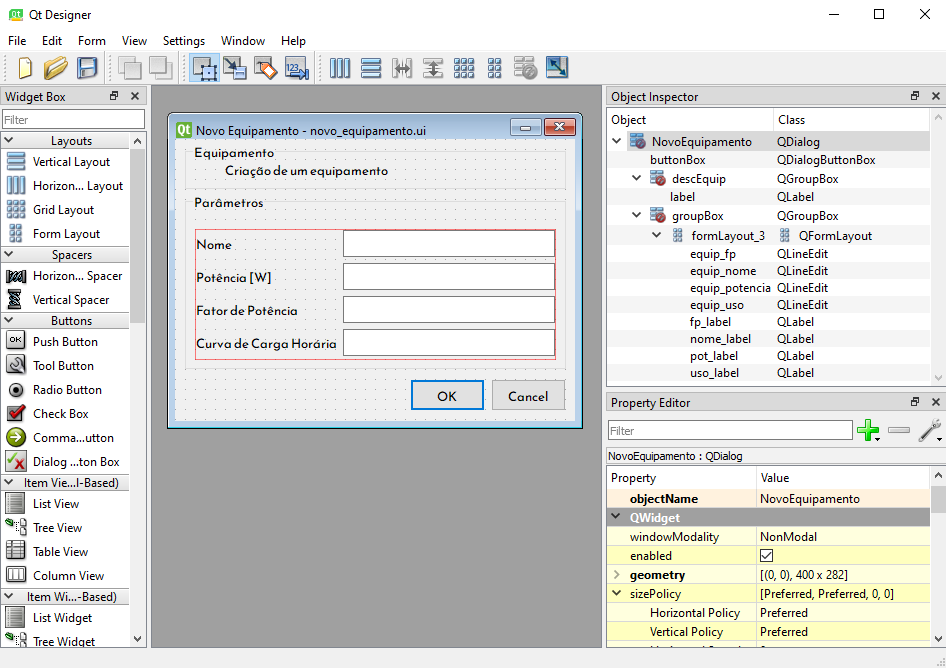
Um dos principais pontos fortes do VSCode é a sua extensibilidade. Através da instalação de extensões, é possível expandir ainda mais as funcionalidades do editor, personalizando-o de acordo com as necessidades do Simuload. No caso do desenvolvimento em Python, existem diversas extensões úteis disponíveis, como integração com sistemas de controle de versão, suporte a testes automatizados, formatação de código, linting e autocompletar, que contribuem para um fluxo de trabalho eficiente e eficaz.

Portanto, a escolha do VSCode como o editor de código para o desenvolvimento do Simuload foi embasada em sua popularidade, facilidade de uso específica para códigos Python e a ampla disponibilidade de extensões úteis que auxiliam no desenvolvimento do software. Essa escolha contribui para uma experiência de programação agradável e produtiva, garantindo um ambiente de trabalho adequado para o desenvolvimento do Simuload.

### *Ferramenta de Interface Visual.*

O processo de desenvolvimento do software Simuload envolveu diversas etapas, dentre as quais se destacam a escolha das ferramentas e tecnologias a serem utilizadas. Para a criação da interface gráfica, optou-se pela ferramenta PyQt 5, que oferece uma vasta gama de recursos para desenvolvimento de interfaces gráficas em Python, incluindo widgets, diálogos e estilos personalizados. A escolha da linguagem de programação Python se deu pelo seu caráter de código aberto, grande comunidade de desenvolvedores, além de sua facilidade de uso e versatilidade.

Através do software Qt Designer, é possível desenhar as telas no formato .ui, definindo todas as suas estruturas e nome das variáveis onde serão inseridos os valores de texto. Com os arquivos .ui em mãos, o PyQt 5 é responsável por transformá-los em código Python, possibilitando a integração das telas com o sistema.

*Fig. 3 Sistema de controle.*  


### *Ferramentas Matemáticas.*

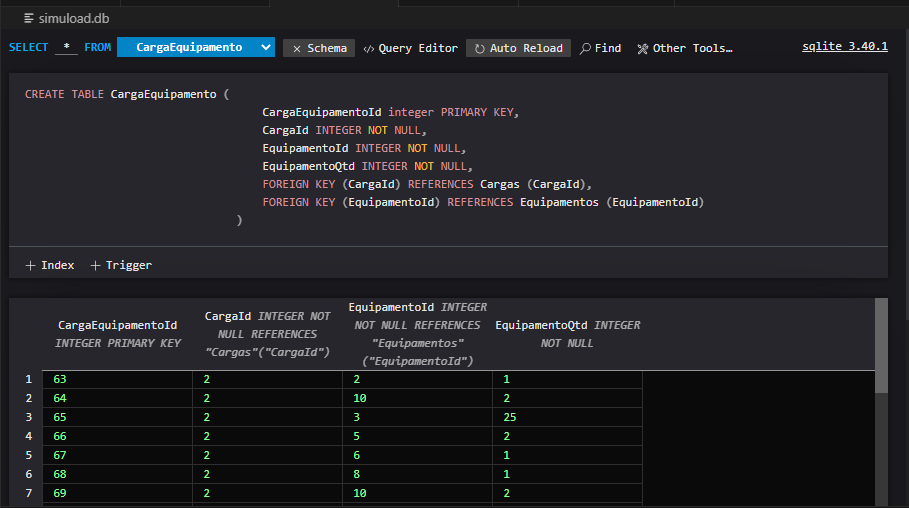
Para a simulação de gráficos, foi escolhida a biblioteca Matplotlib, que oferece uma ampla variedade de gráficos e recursos de visualização de dados em Python, além de ser fácil de usar e ter uma boa documentação. O Matplotlib foi utilizado para a criação dos gráficos de curvas de carga simulados pelo software, permitindo a visualização das informações de maneira clara e eficiente.

Os cálculos realizados pelo software foram feitos diretamente pelo Python, utilizando as bibliotecas padrão do pacote, o que permitiu uma implementação mais simples e direta. Para obter os valores distribuídos em intervalos menores de simulação, foi utilizada a biblioteca Numpy, que oferece uma variedade de recursos para processamento de dados numéricos em Python, incluindo a interpolação de dados, que permitiu a obtenção de valores intermediários a partir dos dados de entrada.

### *Manipulação de Dados.*

No âmbito das especificações técnicas, uma decisão relevante diz respeito ao armazenamento dos dados das simulações realizadas. Nesse sentido, optou-se por utilizar o banco de dados em memória SQLite, uma solução confiável e eficiente para a persistência de dados estruturados no ambiente Python. A escolha do SQLite se deve à sua facilidade de uso e à compatibilidade com diferentes sistemas operacionais, garantindo a portabilidade e a acessibilidade dos dados armazenados.

O VSCode também possui uma extensão de visualização do banco de dados SQLite, exibindo os registros de cada tabela, suas colunas e restrições, scripts de criação das tabelas e possibilidade de consultas personalizadas no banco. A interface da extensão no VSCode pode ser visualizada na figura abaixo.

*Fig. 4 Sistema de controle.*  


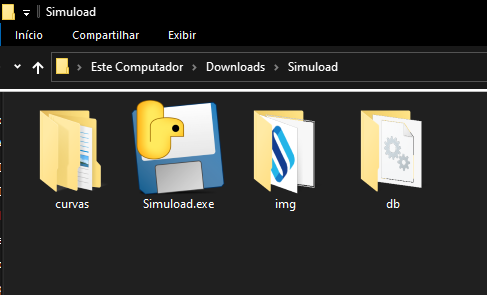
Além disso, foi necessário considerar a exportação dos dados dos gráficos em formato CSV, visando a facilitar a análise e o compartilhamento dessas informações. Para realizar essa tarefa, fez-se uso do módulo csv nativo do Python, que oferece funcionalidades específicas para manipulação desse tipo de arquivo. Adicionalmente, foram explorados os métodos utilitários do módulo os, permitindo a criação de pastas no sistema operacional para uma organização adequada dos arquivos CSV gerados.

Essas decisões técnicas foram fundamentais para garantir a integridade dos dados, a eficiência no armazenamento e a praticidade na exportação dos resultados gerados pelo Simuload. Ao adotar o SQLite como banco de dados e o módulo csv do Python para exportação de dados em formato CSV, o projeto assegura uma abordagem robusta e adaptável, facilitando o acesso e a utilização das informações geradas pela ferramenta.

### *Distribuição do Software.*

Para gerar um executável do software, foi utilizada a ferramenta PyInstaller, que permite empacotar o código-fonte e todas as suas dependências em um único arquivo executável, facilitando a distribuição do software para usuários finais. Como o PyInstaller gera os arquivos através do próprio sistema operacional utilizado, as versões de distribuição do Simuload foram geradas em máquinas com sistemas operacionais diferentes, Linux e Windows, para permitir o seu uso nos dois sistemas.

O executável é capaz de armazenar as curvas exportadas e reconhecer o arquivo do banco de dados na sua própria pasta, além das imagens de logo do software, sendo essa a estrutura dos arquivos compactados de distribuição para download. Abaixo, temos uma imagem da estrutura de pastas e arquivos da versão para o sistema operacional Windows do Simuload.

*Fig. 5 Sistema de controle.*  


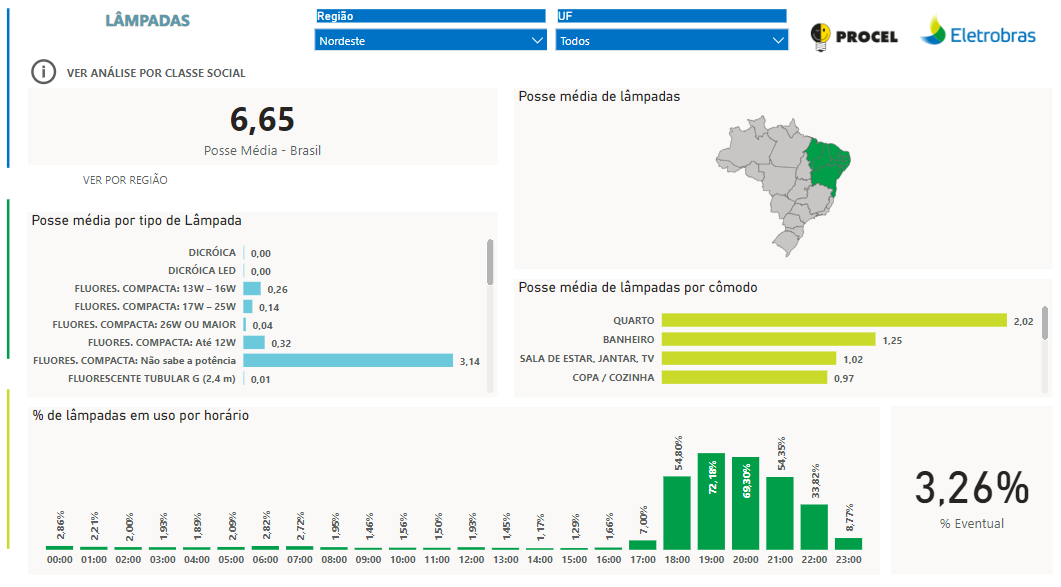
## **Mapeamento de Dados**

Para a coleta de dados utilizados na simulação, foram utilizadas informações retiradas do PPH 2019 do Procel. O Procel é um programa de governo coordenado pelo Ministério de Minas e Energia – MME e executado pela Eletrobras, cujo objetivo é promover a conservação de energia elétrica. A Pesquisa de Posse e Hábitos de Consumo de Energia (PPH) é uma pesquisa declaratória que traça um perfil da posse e hábitos de consumo de equipamentos elétricos e nos setores residencial, comercial e industrial, com o intuito de avaliar o mercado de eficiência energética nas cinco regiões do Brasil.

Os dados utilizados na simulação foram obtidos a partir da PPH, que foi realizada em diferentes regiões do país, o que permitiu a obtenção de um conjunto diversificado de dados de consumo de energia elétrica. As informações foram coletadas através de questionários que perguntavam sobre a posse e uso de diferentes equipamentos elétricos, bem como sobre os hábitos de consumo de energia elétrica em residências, estabelecimentos comerciais e industriais.

Os dados coletados foram utilizados para a definição das características dos equipamentos e cargas presentes na simulação do Simuload, tais como a potência, fator de potência e distribuição horária do uso percentual. Esses dados foram importantes para que as simulações realizadas pelo Simuload fossem o mais próximas possíveis da realidade, permitindo assim que os resultados obtidos a partir das simulações fossem mais precisos e confiáveis. Foi escolhido o filtro da região Nordeste em todos os dados mapeados para o sistema se aproximar da realidade da UFBA. O filtro por Estado escolhendo a Bahia não foi utilizado porque nem todos os equipamentos possuem os seus dados disponíveis a partir dessa opção. Na figura abaixo podemos visualizar uma das telas do PPH 2019 com o filtro da região Nordeste para o equipamento delâmpada, os dados de curva de carga e outras informações de uso que mapeadas pelo Procel.

*Fig. 6 Sistema de controle.*



## **Algoritmo de Aleatoriedade**

Dentro da metodologia adotada neste Trabalho de Conclusão de Curso (TCC), foi desenvolvido um algoritmo de aleatoriedade para representar as curvas de carga em intervalos de tempo menores. Esse algoritmo desempenha um papel essencial no aprimoramento da representação das curvas de carga no sistema Simuload, visando proporcionar uma experiência mais detalhada e realista aos usuários. Isso permite que, a partir de vetores de apenas 24 pontos nos equipamentos, o usuário tenha acesso a simulações com resolução maior sem precisar preencher centenas de pontos para isso.

O algoritmo, implementado em Python, segue uma sequência de passos para a geração de pontos aleatórios que serão adicionados aos vetores de 24 pontos, representando o consumo hora a hora de um equipamento ou a curva de carga horária de um transformador.

Inicialmente, é realizado o cálculo do número de pontos necessários para a interpolação, com base no intervalo desejado. Por exemplo, se o intervalo for de 30 minutos, serão necessários 48 pontos (24 horas divididas em intervalos de 30 minutos). Para esse caso de 48 pontos, cada subintervalo (intervalo de pontos por hora) possui 2 pontos. Em seguida, é criado um vetor chamado que contém os pontos de intervalo entre 0 e 23, representando as horas do dia.

Dentro de um loop principal, percorre-se cada ponto do vetor de interpolação. Para cada ponto, são realizadas as seguintes etapas:

1. O valor atual do vetor de consumo é atribuído a uma variável que representa a média do subintervalo atual. Esse valor é calculado a partir do vetor de consumo original, que contém os valores de consumo em intervalos de 1 hora.
2. Se o intervalo de interpolação for menor do que 1 hora (por exemplo, 30 minutos), entra-se em um loop adicional para gerar os pontos aleatórios. Caso contrário, pula-se esta etapa.
3. Uma seed única é calculada com base na soma de todos os valores do vetor de consumo original. A seed é um número fixo que, quando definido, garante que os valores aleatórios gerados sejam os mesmos para todas as simulações de uma mesma curva.
4. Um número aleatório é gerado dentro de um intervalo predefinido, representando a variação percentual permitida em relação ao valor-base. Neste caso, a variação é definida em até 40% do valor-base, positiva ou negativa.
5. Dependendo do índice do ponto no loop, o sinal do valor aleatório gerado é invertido para alternar entre valores positivos e negativos.
6. O valor aleatório gerado é adicionado ao valor médio multiplicado pela porcentagem de variação e dividido por 100. Isso resulta em um valor ajustado que representa o consumo aleatório no subintervalo.
7. Um novo valor médio é acumulado a partir dos valores gerados, pois cada valor contribui para a média total do subintervalo.
8. É calculado um fator de normalização com base na média original e na nova média acumulada. Esse fator é utilizado posteriormente para garantir que os valores não ultrapassem a média original.
9. Os valores do subintervalo são multiplicados pelo fator de normalização para ajustá-los à média original. Se algum valor exceder a média original, ele é substituído pela média e o valor excedente é adicionado ao próximo ponto, mantendo a consistência do consumo total do subintervalo. Caso seja o último ponto do subintervalo, o valor excedente é adicionado ao ponto anterior.
10. O loop continua para o próximo subintervalo até que todos os pontos do vetor de interpolação sejam processados.

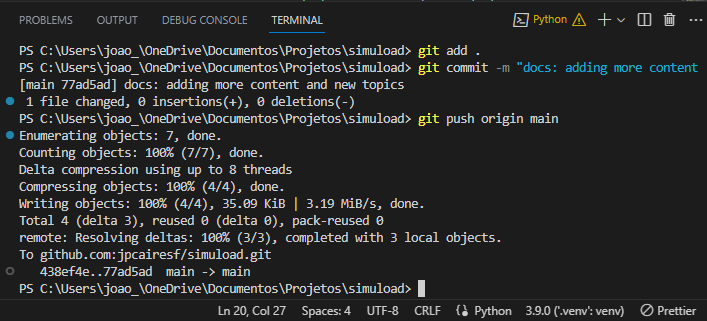
Ao final do algoritmo, é retornado um vetor contendo os pontos interpolados da curva de carga, representados em quilowatts (kW). Esse vetor é utilizado para visualizar a curva de carga com maior resolução, refletindo o consumo detalhado dos equipamentos ao longo do tempo. Em algumas simulações, tem-se a impressão de que a curva de carga com intervalos menores possuem magnitude menor que a curva simulada para o intervalo de 1 hora. Isso acontece por conta da premissa de que o valor aleatório deve ser sempre menor ou igual ao valor da curva. A implementação completa do algoritmo pode ser encontrada no apêndice.

## **Versionamento de Código**

No desenvolvimento do Simuload, foi adotado o sistema de controle de versão Git junto a plataforma de hospedagem GitHub para gerenciar o versionamento do código fonte e facilitar a disponibilização das versões de distribuição do software.

O Git é um sistema de controle de versão distribuído amplamente utilizado na indústria de desenvolvimento de software. Ele permite controlar o histórico de alterações do código fonte, rastrear e registrar as modificações realizadas ao longo do tempo. Com o Git, é possível criar ramificações para desenvolver novos recursos ou corrigir problemas sem afetar a versão principal do software. Com o Git instalado, é possível executar seus comandos no terminal da máquina utilizada para o desenvolvimento. Além disso, o VSCode possui um terminal embutido em sua plataforma que permite a execução desses comandos, como demonstrado na figura abaixo.

*Fig. 7 Submissão de alterações no código utilizando o Git através do terminal embutido no VSCode*

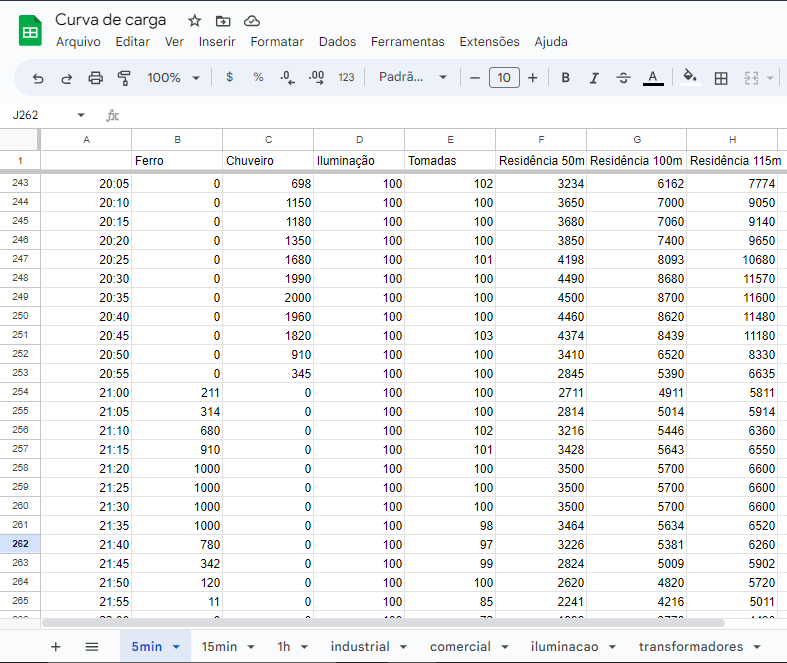
 O GitHub é uma plataforma web que oferece recursos de hospedagem de repositórios Git. Por meio do GitHub, é possível armazenar de forma segura o código fonte do Simuload em um repositório centralizado. Além disso, ele fornece uma interface amigável para visualizar as alterações feitas no código ao longo do tempo. No Apêndice B deste trabalho, são fornecidos os links para acessar o repositório do Simuload no GitHub. Lá, você encontrará o código fonte do software disponível para consulta e download. O Apêndice também inclui instruções detalhadas sobre como baixar e instalar o Simuload, bem como os requisitos do sistema e os procedimentos passo a passo.

# **Estudo de Caso**

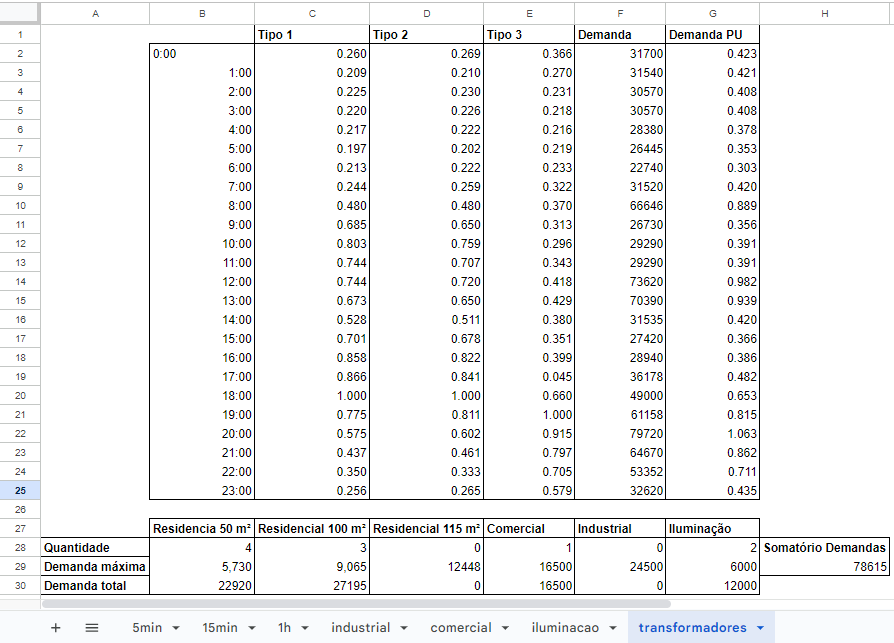
A ferramenta Simuload foi utilizada por alunos em um estudo de caso real, visando mapear uma região com suas respectivas cargas e equipamentos elétricos. Essa atividade é realizada durante a matéria – Laboratório Integrado VI, no sétimo semestre do curso de Engenharia Elétrica da UFBA. Tal atividade consiste em simular a curva de carga da região e escolher o transformador adequado para atender as condições de demanda de energia elétrica. Essa é uma atividade de suma importância na formação dos estudantes, pois a partir dela é possível ter insumos que nortearão a necessidade elétrica de um determinado local.

## **Simulando com outros Softwares**

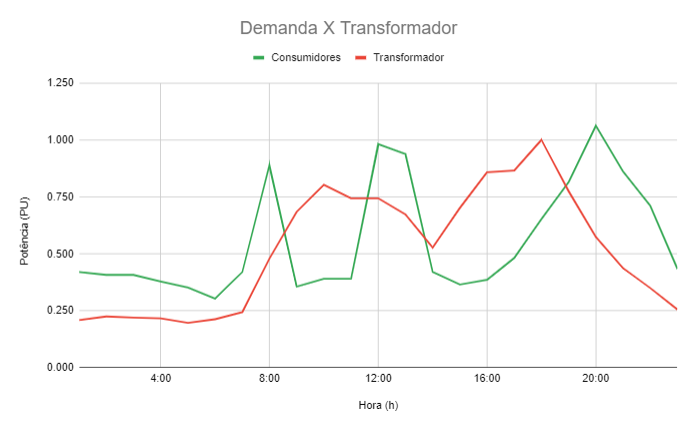
Para simular os gráficos de curvas de cargas sem o Simuload, geralmente é utilizado um software de planilhas como Microsoft Excel ou Google Sheets. Estes são os recursos comumente usados atualmente pelos estudantes durante o componente curricular Laboratório Integrado VI. No entanto, tal processo acaba sendo muito manual e trabalhoso para o preenchimento dos dados. No intervalo de 5 minutos, por exemplo, são praticamente 300 células de valores preenchidos por intervalo para diversos equipamentos.

*Fig. 8 Sistema de controle.*  


Com os dados de consumo dos equipamentos em mãos, é combinada a quantidade de cada equipamento para compor as residências e estabelecimentos, que por sua vez serão combinados para compor uma região para ser atendida pelos transformadores. O consumo dos estabelecimentos são calculados a partir desses valores e utilizados na simulação, comparando com as curvas do transformador e ajustando as quantidades para adequar o consumo conforme mostrado nas figuras abaixo.

*Fig. 9 Cálculo das curvas de carga a partir de consumidores residencial, comercial, industrial e iluminação no Google Sheets.*

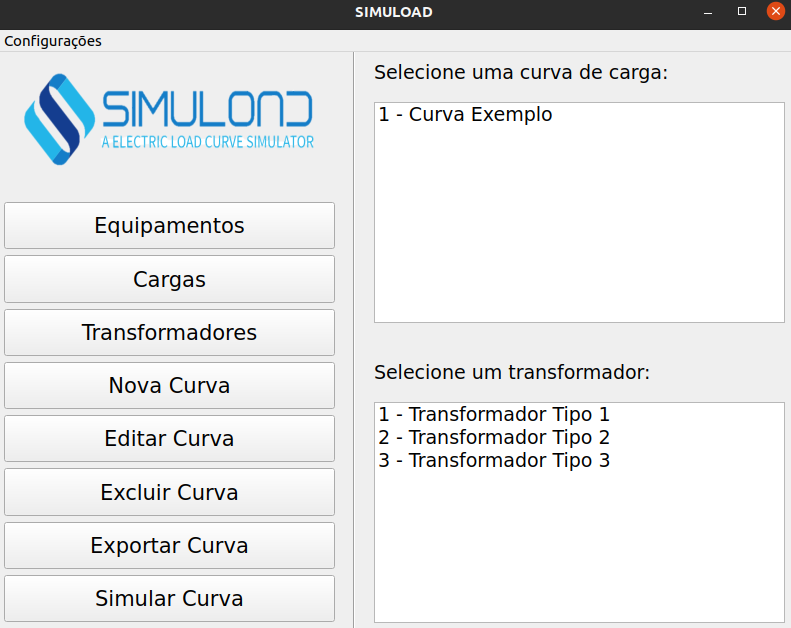
*Fig. 10 Curva de simulação da demanda dos estabelecimentos pela curva característica do transformador no Google Sheets.*



## **Aplicação do Simuload**

Realizando a simulação com o Simuload, é possível utilizar os equipamentos e cargas pré-definidos no banco de dados ou customizar novas opções. Dentre os registros pré-definidos, estão os equipamentos e suas curvas de carga horária mapeadas pelo PPH 2019 do Procel e potência mapeadas pela norma SM04.14-01.001 12ª edição da Neoenergia em seu anexo I, que pode ser encontrado no apêndice C. Na tela principal é possível visualizar os comandos de criação de componentes ou simulações de curvas.

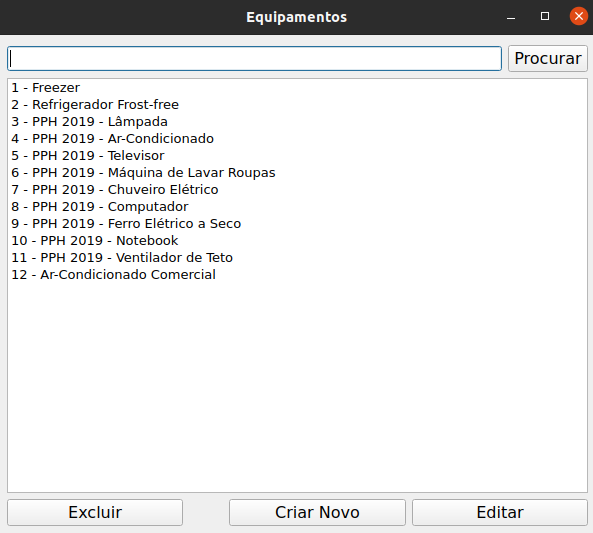
*Fig. 11 Sistema de controle.*

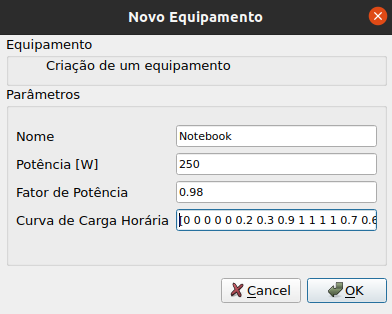


Os alunos iniciaram a atividade adicionando os equipamentos na ferramenta. Adicionar, editar ou excluir um equipamento são tarefas possíveis no menu equipamentos. No campo Curva de Carga Horária, podemos definir a distribuição de uso em 24 horas do equipamento seguindo o padrão de números entre 0 e 1 nos colchetes: [1 1 1 1 1 1 1 1 1 1 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.2 0.2 0.2 0.2].

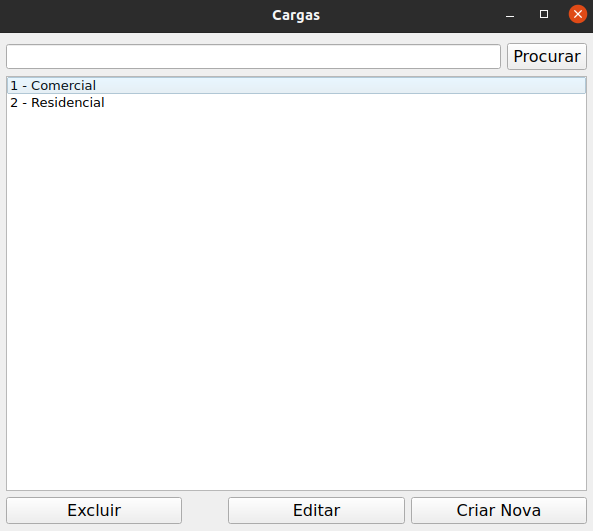
Após adicionar os equipamentos, foram definidas a potência, o fator de potência e a distribuição horária de uso de cada equipamento. No campo Potência [W] deve ser inserido um valor numérico que pode ser decimal utilizando um ponto no lugar da vírgula. O Fator de Potência também pode ser decimal mas deve variar entre 0 e 1.

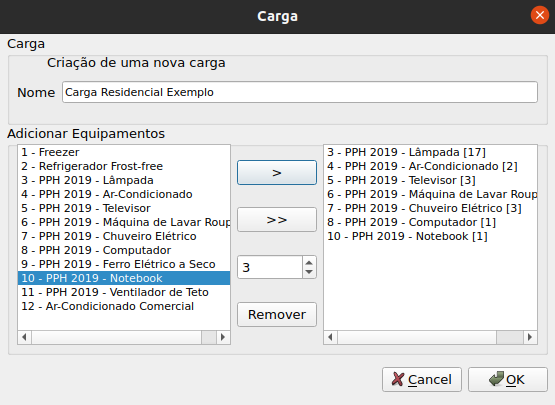
*Fig. 12 Sistema de controle.*



*Fig. 13 Sistema de controle.*  


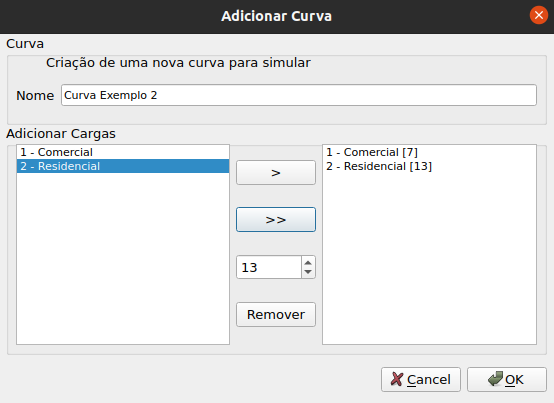
Em seguida, eles adicionaram as cargas, que poderiam ser estabelecimentos residenciais, comerciais ou até mesmo iluminação pública. No menu cargas é possível juntar diversas configurações de equipamentos para construir a carga desejada. Na tela de criação é possível adicionar equipamentos unitários (>) ou em lote (>>) selecionando o número desejado logo abaixo. Também é possível remover equipamentos selecionados à direita.

*Fig. 14 Sistema de controle.*  


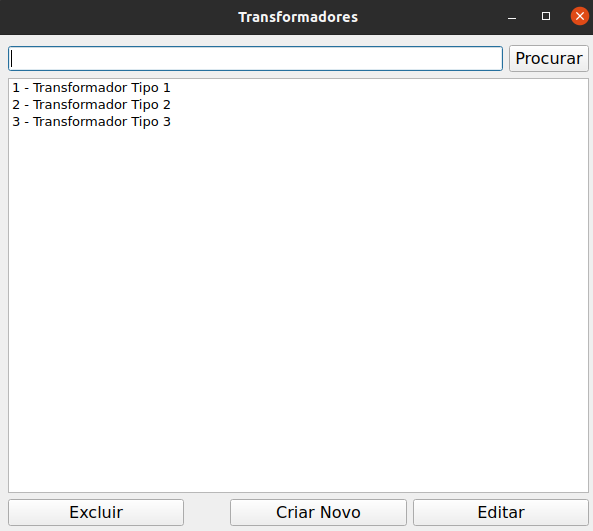
*Fig. 15 Sistema de controle.*  


A carga era alimentada pelos equipamentos adicionados anteriormente. Após a adição das cargas, os alunos criaram uma curva, que representa a distribuição temporal do consumo de energia elétrica das várias cargas. Essa curva representa o consumo de energia elétrica numa região atendida por um transformador.

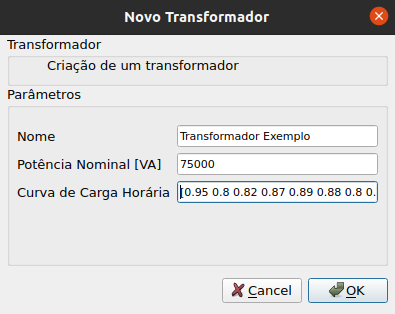
Na tela principal é possível editar as configurações para a simulação da curva e criar curvas baseadas nas cargas existentes. Da mesma forma que a janela de novas cargas, as curvas utilizam a estrutura de adição individual (>) ou em lote (>>) das cargas, sendo possível escolher a quantidade do lote adicionado pela numeração abaixo.

*Fig. 16 Sistema de controle.*  


Os transformadores característicos de 75 kVA dos tipos 1, 2 e 3, utilizados no projeto da disciplina Laboratório Integrado VI, já estão registrados por padrão no Simuload e prontos para uso nas simulações. No menu transformadores é possível criar e configurar outros transformadores com a mesma estrutura de distribuição de 24 horas dos equipamentos no campo Curva de Carga Horária: [1 1 1 1 1 1 1 1 1 1 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.2 0.2 0.2 0.2]. A Potência Nominal é representada em Volt-Ampère [VA].

*Fig. 17 Sistema de controle.*

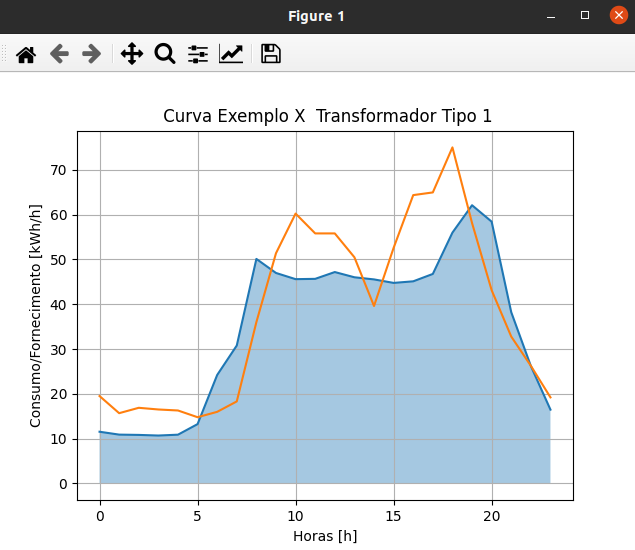
*Fig. 18 Sistema de controle.*

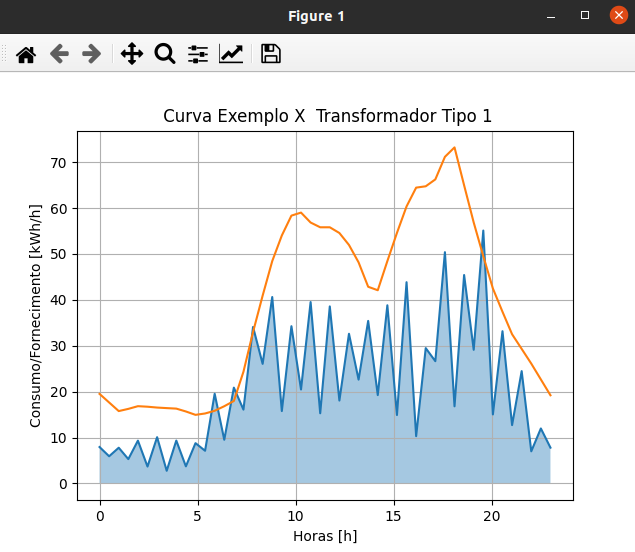


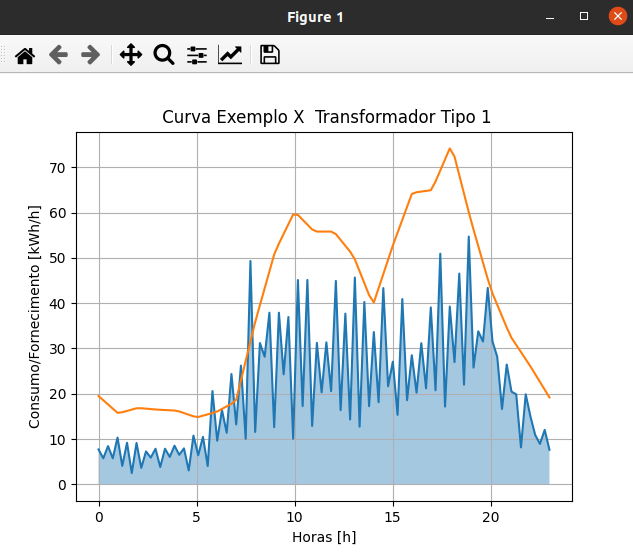
Para realizar a simulação, basta clicar na opção "Simular Curva" na janela principal. É possível selecionar uma curva de carga individual ou uma curva de carga junto com um transformador. Se a simulação ou exportação não funcionarem, é provável que algum formato esteja incorreto na Curva de Carga Horária de um equipamento ou transformador.

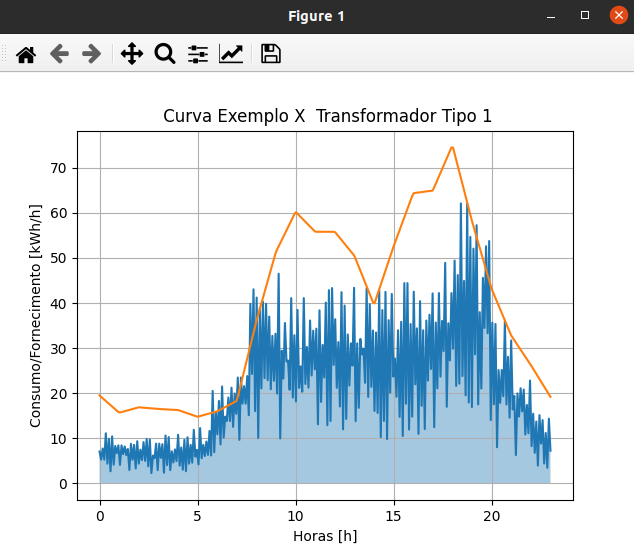
Durante a simulação, a curva de carga é representada pela linha azul, enquanto a do transformador é representada pela linha laranja. A simulação oferece várias opções utilitárias padrão fornecidas pelo pacote Matplotlib. É possível mover a curva, aplicar zoom, desfazer ações, editar bordas e espaçamentos, configurar as características da curva e salvá-la como imagem.

No gráfico, o eixo Y representa o consumo/fornecimento da curva e do transformador em Kilowatt-hora [kWh], enquanto o eixo X representa as horas [h]. Antes de realizar a simulação, é possível selecionar o intervalo desejado nas configurações. As opções disponíveis são: 1 hora, 30 minutos, 15 minutos e 5 minutos, sendo que o intervalo padrão é de 1 hora. A seguir, apresentamos as simulações realizadas com intervalo de 1 hora e, respectivamente, intervalos de 30 minutos, 15 minutos e 5 minutos. Essas simulações utilizam as curvas registradas por padrão no Simuload.

*Fig. 19 Sistema de controle.* 

*Fig. 20 Sistema de controle.*  


*Fig. 21 Sistema de controle.*  


*Fig. 22 Sistema de controle.*  


Seguindo as etapas de utilização do Simuload, os alunos conseguiram adaptar os equipamentos, cargas e curvas para atender aos critérios do projeto em questão. O objetivo é ajustar a curva de carga para que o consumo esteja entre 100% e 130% da capacidade do transformador.

Para uma manipulação e avaliação mais precisa dos dados, foi implementada a funcionalidade de "Exportar Curva". Essa opção gera um arquivo .CSV com base na curva e no transformador selecionados, considerando o intervalo escolhido nas configurações. O arquivo .CSV é salvo na pasta "curvas", localizada no mesmo diretório do Simuload. Cada simulação cria uma nova pasta com o nome da curva e do transformador utilizados. Dentro dessa pasta, há o arquivo .CSV com os dados de simulação, além do nome do arquivo que indica a data e horário em que foi exportado.

## **Resultados e Comparações**

A utilização do Simuload se mostrou uma alternativa mais prática e eficiente quando comparada com a maneira comumente utilizada pelos alunos, na qual consiste em fazer as simulações manualmente, hora a hora, inserindo valores de vários equipamentos em vários estabelecimentos e calculando tudo por planilhas. Todo esse trabalho aumenta o tempo total destinado a essa atividade e pode levar a uma maior possibilidade de erro, pois os cálculos são feitos de forma manual pelos estudantes. Com o Simuload foi possível fazer todo esse cadastramento, análise e escolha do transformador adequado de forma mais prática e sem comprometer o processo de aprendizagem proposto no componente curricular.

Ao utilizar o Simuload, os alunos puderam ajustar as cargas, equipamentos e a curva para atender às condições do transformador, de forma mais ágil e eficiente. A ferramenta se mostrou uma alternativa prática e útil para a realização da atividade de Laboratório Integrado VI, proporcionando uma experiência mais enriquecedora e produtiva para os alunos.

# **Conclusão**

Video provides a powerful way to help you prove your point. When you click Online Video, you can paste in the embed code for the video you want to add. You can also type a keyword to search online for the video that best fits your document. To make your document look professionally produced, Word provides header, footer, cover page, and text box designs that complement each other. For example, you can add a matching cover page, header, and sidebar.

Click Insert and then choose the elements you want from the different galleries. Themes and styles also help keep your document coordinated. When you click Design and choose a new Theme, the pictures, charts, and SmartArt graphics change to match your new theme. When you apply styles, your headings change to match the new theme. Save time in Word with new buttons that show up where you need them.

### Título da subseção.

Video provides a powerful way to help you prove your point. When you click Online Video, you can paste in the embed code for the video you want to add. You can also type a keyword to search online for the video that best fits your document.

To make your document look professionally produced, Word provides header, footer, cover page, and text box designs that complement each other. For example, you can add a matching cover page, header, and sidebar. Click Insert and then choose the elements you want from the different galleries.

Themes and styles also help keep your document coordinated. When you click Design and choose a new Theme, the pictures, charts, and SmartArt graphics change to match your new theme. When you apply styles, your headings change to match the new theme.

Save time in Word with new buttons that show up where you need them. To change the way a picture fits in your document, click it and a button for layout options appears next to it. When you work on a table, click where you want to add a row or a column, and then click the plus sign.

Reading is easier, too, in the new Reading view. You can collapse parts of the document and focus on the text you want. If you need to stop reading before you reach the end, Word remembers where you left off - even on another device.

Video provides a powerful way to help you prove your point. When you click Online Video, you can paste in the embed code for the video you want to add. You can also type a keyword to search online for the video that best fits your document.

To make your document look professionally produced, Word provides header, footer, cover page, and text box designs that complement each other. For example, you can add a matching cover page, header, and sidebar. Click Insert and then choose the elements you want from the different galleries.

Themes and styles also help keep your document coordinated. When you click Design and choose a new Theme, the pictures, charts, and SmartArt graphics change to match your new theme. When you apply styles, your headings change to match the new theme.

### Título da subseção.

Video provides a powerful way to help you prove your point. When you click Online Video, you can paste in the embed code for the video you want to add. You can also type a keyword to search online for the video that best fits your document. To make your document look professionally produced, Word provides header, footer, cover page, and text box designs that complement each other. For example, you can add a matching cover page, header, and sidebar. Click Insert and then choose the elements you want from the different galleries. Themes and styles also help keep your document coordinated. When you click Design and choose a new Theme, the pictures, charts, and SmartArt graphics change to match your new theme. When you apply styles, your headings change to match the new theme.

# Referências

ERICKSON, Robert W.; MAKSIMOVIC, Dragan. **Fundamentals of power electronics**. Springer Science & Business Media, 2007.

CHEN, Jingquan; MAKSIMOVIC, Dragan; ERICKSON, Robert W. Analysis and design of a low-stress buck-boost converter in universal-input PFC applications. **IEEE Transactions on Power Electronics**, v. 21, n. 2, p. 320-329, 2006.

# Apêndice A – **Código do algoritmo de aleatoriedade**

def interpolador\_curva(self, consumo\_curva, intervalo = 1):

values = int(24 \* (60/intervalo))

intervalo\_interp = np.linspace(0, 23, values)

# cria um vetor vazio com a quantidade de valores e define o subintervalo

# subintervalo por ser de 5, 15 ou 30 minutos

consumo\_curva\_interp = [None] \* values

sub\_intervalo = int(values/24)

count = 0

# fixa uma seed para que o resultado seja o mesmo em todas as simulações

seed = int(np.sum(consumo\_curva))

percentual\_var = 40

base\_avg = 100 - percentual\_var

while (count < values):

# valor no vetor de 24h é a média do sub intervalo

consumo\_curva\_interp[count] = consumo\_curva[int(count/sub\_intervalo)]

avg = consumo\_curva\_interp[count]

if(values > 24):

new\_avg = 0

for i in range(sub\_intervalo):

# setar uma seed única e gerar um valor de 0% a 40%

np.random.seed(seed + count + i)

rgen = np.random.randint(0, percentual\_var)

# comutar o sinal do valor gerado rgen

if (count + i) % 2 != 0:

rgen = rgen\*(-1)

# soma o valor gerado a partir de meia média e acumula uma nova média

consumo\_curva\_interp[count+i] = avg\*(base\_avg+rgen)\*0.01

new\_avg += consumo\_curva\_interp[count+i]

i+=1

# calcular fator do que falta para a média original

normalization = 100/base\_avg

fator = avg/(normalization\*new\_avg/sub\_intervalo)

# multiplicar todos os valores do sub intervalo pelo fator

j = 0

while (j < sub\_intervalo):

# se o valor exceder a média:

# o ponto será igual à média e salvar o valor excedente

if (consumo\_curva\_interp[count+j] \* fator) > avg:

consumo\_curva\_interp[count+j] = avg

exceed = consumo\_curva\_interp[count+j] \* fator - avg

# se ainda houver próximo valor, adicionar o excedente ao próximo

# e contar duas iterações

if (j + 1) < sub\_intervalo:

consumo\_curva\_interp[count+j+1] = consumo\_curva\_interp[count+j+1] \* fator + exceed

j += 2

# se não, adicionar ao anterior e contar uma iteração só (fim do laço)

else:

consumo\_curva\_interp[count+j-1] = consumo\_curva\_interp[count+j-1] \* fator + exceed

j += 1

# se não exceder a média, seguir a fórmula do fator

else:

consumo\_curva\_interp[count+j] = consumo\_curva\_interp[count+j] \* fator

j += 1

# a iteração acontece pra cada sub intervalo entre uma hora e outra

count += sub\_intervalo

# retorna valor em kW

return intervalo\_interp, np.array(consumo\_curva\_interp)/1000