



**INSTITUTO
FEDERAL**

Goiano

Campus
Trindade

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO -
CAMPUS TRINDADE**

CURSO BACHARELADO EM ENGENHARIA ELÉTRICA

RAPHAELA RAIKA DOS ANJOS FERNANDES

SÁVIO VIEIRA FERNANDES

VINÍCIUS SAMUEL ALVES SOARES

**SISTEMA DE GERENCIAMENTO DE CONSUMO ENERGÉTICO:
SOLUÇÃO AUTOMATIZADA PARA MONITORAMENTO DE CONSUMO DE ENERGIA,
GERAÇÃO DE USINAS FOTOVOLTÁICAS E CRÉDITOS ENERGÉTICOS**

TRINDADE - GO

2022

**RAPHAELA RAIKA DOS ANJOS FERNANDES
SÁVIO VIEIRA FERNANDES
VINÍCIUS SAMUEL ALVES SOARES**

**SISTEMA DE GERENCIAMENTO DE CONSUMO ENERGÉTICO:
SOLUÇÃO AUTOMATIZADA PARA MONITORAMENTO DE CONSUMO DE ENERGIA,
GERAÇÃO DE USINAS FOTOVOLTÁICAS E CRÉDITOS ENERGÉTICOS**

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Engenharia Elétrica do Instituto Federal de Educação, Ciência e Tecnologia Goiano - Campus Trindade, como parte dos requisitos para obtenção grau de Bacharel em Engenharia Elétrica.

Orientador: Esp. Marcus Vinicius Mota Pinheiro da Costa

Coorientador: Me. Geovanne Pereira Furriel

TRINDADE - GO

2022

Sistema desenvolvido pelo ICMC/USP
Dados Internacionais de Catalogação na Publicação (CIP)
Sistema Integrado de Bibliotecas - Instituto Federal Goiano

FF363s Fernandes, Raphaela Raika dos Anjos; Fernandes,
Sávio Vieira; Soares, Vinicius Samuel Alves
Sistema de Gerenciamento de Consumo Energético:
Solução automatizada para monitoramento de consumo de
energia, geração de usinas fotovoltaicas e créditos
energéticos / Raphaela Raika dos Anjos; Fernandes,
Sávio Vieira; Soares, Vinicius Samuel Alves
Fernandes; orientador Me. Marcus Vinicius Mota
Pinheiro da Costa; co-orientador Me. Geovanne
Pereira Furriel. -- Trindade, 2023.
122 p.

TCC (Graduação em Engenharia Elétrica) --
Instituto Federal Goiano, Campus Trindade, 2023.

1. Consumo de Energia. 2. Medidor de Consumo. 3.
Gestão Inteligente de Consumo Energético. 4. Gestão
de Faturas. I. Mota Pinheiro da Costa, Me. Marcus
Vinicius, orient. II. Furriel, Me. Geovanne

Responsável: Johnathan Pereira Alves Diniz - Bibliotecário-Documentalista CRB-1 nº2376

TERMO DE CIÊNCIA E DE AUTORIZAÇÃO

PARA DISPONIBILIZAR PRODUÇÕES TÉCNICO-CIENTÍFICAS

NO REPOSITÓRIO INSTITUCIONAL DO IF GOIANO

Com base no disposto na Lei Federal nº 9.610, de 19 de fevereiro de 1998, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia Goiano a disponibilizar gratuitamente o documento em formato digital no Repositório Institucional do IF Goiano (RIIF Goiano), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IF Goiano.

IDENTIFICAÇÃO DA PRODUÇÃO TÉCNICO-CIENTÍFICA

- | | |
|--|---|
| <input type="checkbox"/> Tese (doutorado) | <input type="checkbox"/> Artigo científico |
| <input type="checkbox"/> Dissertação (mestrado) | <input type="checkbox"/> Capítulo de livro |
| <input type="checkbox"/> Monografia (especialização) | <input type="checkbox"/> Livro |
| <input checked="" type="checkbox"/> TCC (graduação) | <input type="checkbox"/> Trabalho apresentado em evento |

Produto técnico e educacional - Tipo:

Nome completo do autor:

Matrícula:

Título do trabalho:

RESTRIÇÕES DE ACESSO AO DOCUMENTO

Documento confidencial: Não Sim, justifique:

Informe a data que poderá ser disponibilizado no RIIF Goiano: / /

O documento está sujeito a registro de patente? Sim Não

O documento pode vir a ser publicado como livro? Sim Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O(a) referido(a) autor(a) declara:

- Que o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
- Que obteve autorização de quaisquer materiais inclusos no documento do qual não detém os direitos de autoria, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia Goiano os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
- Que cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia Goiano.

Local

/ /

Data

DocuSigned by:

Raphaela Raika dos Anjos Fernandes

1C8824105DC5486...

DocuSigned by:

Sávio Vieira Fernandes

B597699DD18C4D3...

DocuSigned by:

Vinicius Samuel A. Soares

BB2D0F1577C949C...

Assinatura do autor e/ou detentor dos direitos autorais

Ciente e de acordo:

DocuSigned by:

[Assinatura]

B191A0522644409...

Assinatura do(a) orientador(a)



SERVIÇO PÚBLICO FEDERAL
MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA GOIANO

Ata nº 1/2023 - CE-TRI/GE-TRI/CMPTRI/IFGOIANO

ATA DE DEFESA DE TRABALHO DE CURSO

Aos **26** dias do mês de janeiro de 2023, às 19 horas e 30 minutos, reuniu-se a banca examinadora composta pelos docentes: **Marcus Vinicius Mota Pinheiro da Costa** (orientador), **Cleber Asmar Ganzaroli** (membro), **Renato Milhomem de Oliveira Filho** (membro), para examinar o Trabalho de Curso intitulado “**Sistema de gerenciamento de consumo energético: solução automatizada para monitoramento de consumo de energia, geração de usinas fotovoltaicas e créditos energéticos.**” dos estudantes **Raphaela Raika Dos Anjos Fernandes**, Matrícula nº 2018108202640049, **Sávio Vieira Fernandes**, Matrícula nº 2018108202640227 e **Vinicius Samuel Alves Soares** Matrícula nº 2018108202640243 do Curso de Bacharelado em Engenharia Elétrica do IF Goiano – Campus Trindade. A palavra foi concedida ao(a) estudante para a apresentação oral do TC, houve arguição do(a) candidato pelos membros da banca examinadora. Após tal etapa, a banca examinadora decidiu pela **APROVAÇÃO** dos estudantes. Ao final da sessão pública de defesa foi lavrada a presente ata que segue assinada pelos membros da Banca Examinadora.

Documento assinado eletronicamente por:

- Cleber Asmar Ganzaroli, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 31/01/2023 17:32:33.
- Renato Milhomem de Oliveira Filho, PROF ENS BAS TEC TECNOLOGICO-SUBSTITUTO, em 31/01/2023 15:17:32.
- Vinicius Samuel Alves Soares, 2018108202640243 - Discente, em 31/01/2023 13:37:08.
- Sávio Vieira Fernandes, 2018108202640227 - Discente, em 31/01/2023 13:05:03.
- Raphaela Raika dos Anjos Fernandes, 2018108202640049 - Discente, em 31/01/2023 12:54:28.
- Marcus Vinicius Mota Pinheiro da Costa, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 31/01/2023 12:53:08.

Este documento foi emitido pelo SUAP em 31/01/2023. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifgoiano.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 461906
Código de Autenticação: a0ffb078b8



RESUMO

O presente projeto tem por finalidade criar e implementar um sistema de gestão inteligente de e unificado do consumo de energia, considerando também o fator geração caso exista uma usina fotovoltaica (microgeração e minigeração) instalada, bem como realizar a análise do desempenho e viabilidade de um sistema implementado nas cidades de Goiânia-GO e Aparecida de Goiânia-GO durante o ano de 2022. O sistema é dividido em: montagem e teste de um dispositivo protótipo para monitoramento do consumo de energia da unidade consumidora analisada em tempo real; desempenho da usina fotovoltaica existente no local de teste; gestão de fatura e créditos da concessionária de energia; cruzamento de dados e produção de relatório automático ao usuário; criação de uma plataforma para disposição dos dados de maneira *online* ao usuário. Esta análise tem como objetivo a constatação da utilidade, viabilidade técnica e econômica e a escalabilidade do sistema para inúmeros usuários, também visa atestar o funcionamento do leitor de consumo e comparar seu desempenho com outros equipamentos do mercado.

Palavras-chave: Consumo de Energia. Medidor de Consumo. Gestão Inteligente de Consumo Energético. Gestão de Faturas.

ABSTRACT

The purpose of this project is to create and implement an intelligent and unified management system for energy consumption, also considering the generation factor if there is a photovoltaic plant (microgeneration and minigeneration) installed, as well as performing an analysis of the performance and feasibility of a system implemented in the cities of Goiânia-GO and Aparecida de Goiânia-GO during the year 2022. The system is divided into: assembly and testing of a prototype device for monitoring the energy consumption of the consumer unit analyzed in real time; performance of the existing photovoltaic plant at the test site; energy utility bill and credit management; data crossing and production of automatic report to the user; creation of a platform for online data provision to the user. This analysis aims to verify the usefulness, technical and economic viability and scalability of the system for numerous users, it also aims to attest to the operation of the consumer reader and compare its performance with other equipment on the market.

Keywords: Energy consumption. Consumption Meter. Intelligent Management of Energy Consumption. Invoice Management.

LISTA DE FIGURAS

Figura 1 – Componentes Utilizados no Protótipo	30
Figura 2 – Montagem do Protótipo	32
Figura 3 – Protótipo Vista Frontal	33
Figura 4 – Protótipo Vista das Soldas	33
Figura 5 – Consumo Diário Obtido	35
Figura 6 – Medidor de Consumo Neurio W1-HEM Trifásico	36
Figura 7 – Consumo Registrado pelo Equipamento Generac	37
Figura 8 – Valores de Consumo e Erro	38
Figura 9 – Erro de Leitura Referente a 1 Dia	39
Figura 10 – Banco de Dados SQL	40
Figura 11 – Gráfico dos Dados SQL	40
Figura 12 – Parte Superior do Relatório Mensal	41
Figura 13 – Parte Inferior do Relatório Mensal	41
Figura 14 – Página Inicial do Site	42
Figura 15 – Página de Registro de Usuário	42
Figura 16 – Página de Login Convencional	43
Figura 17 – Página de Login Administrador	43
Figura 18 – Parte Superior da Página Privada de um Usuário	44
Figura 19 – Parte Inferior da Página Privada de um Usuário	44
Figura 20 – Página de Administrador	45

LISTA DE TABELAS

Tabela 1 – Lista de Componentes	29
Tabela 2 – Consumo Diário do Período	35
Tabela 3 – Comparação dos Valores de Consumo Registrados	37
Tabela 4 – Comparação das Leituras em 24 horas	38

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Objetivos Principais e Específicos	13
1.2	Justificativa	14
2	FUNDAMENTAÇÃO TEÓRICA	15
2.1	Hardware: Dispositivo Medidor de Consumo	15
2.1.1	Sistema Elétrico de Potência	15
2.1.2	Geração Distribuída	15
2.1.3	Geração Fotovoltáica On Grid	15
2.1.4	Medição de Consumo	16
2.2	Software: API e Plataforma Online	17
2.2.1	Application Programming Interface	17
2.2.2	<i>Web Scraping</i>	18
2.2.3	Banco de Dados	18
2.2.4	<i>Hypertext Markup Language</i>	18
2.2.5	<i>PHP: Hypertext Preprocessor</i>	19
2.2.6	<i>Cascading Style Sheets</i>	19
2.2.7	<i>Bootstrap</i>	20
2.2.8	<i>Web Server</i>	20
2.2.9	<i>Web Host</i>	20
2.2.10	Estruturação para Plataforma <i>Online</i>	21
3	METODOLOGIA	23
3.1	Montagem do Hardware	23
3.1.1	Conexão do Protótipo com Banco de Dados	24
3.2	Requisito de Software	24
3.2.1	<i>Web Scraping</i> no Site da Concessionária	25
3.3	Programação da Plataforma Online	26
4	RESULTADOS E DISCUSSÃO	29
4.1	Hardware	29
4.1.1	Montagem do Protótipo	30
4.1.2	Aferição dos Sensores	32
4.1.3	Código em Arduino	33
4.1.4	Armazenamento de Dados de Leitura	34
4.1.5	Leituras Obtidas com o Protótipo Instalado	34
4.1.6	Validação das Leituras de Consumo	36

4.2	<i>Software</i>	39
4.2.1	Teste do <i>Script API</i>	39
4.2.2	Plataforma <i>Online</i>	41
5	CONCLUSÕES	46
6	CONSIDERAÇÕES FINAIS	48
	REFERÊNCIAS	50
	APÊNDICES	52
	APÊNDICE A – WEB SCREAPING: SOLARVIEW, CONCESSIONÁRIA DE ENERGIA (ENEL - GO), CONEXÃO COM O BANCO DE DADOS E EMISSÃO DE RELATÓRIO DO CLIENTE	53
	APÊNDICE B – CONEXÃO DO SITE COM BANCO DE DADOS	61
	APÊNDICE C – ESTILO PADRÃO DAS PÁGINAS	62
	APÊNDICE D – PÁGINA INICIAL DO SITE	65
	APÊNDICE E – PÁGINA DE REGISTRO DE USUÁRIO	70
	APÊNDICE F – PÁGINA DE LOGIN DE USUÁRIO	78
	APÊNDICE G – LOGOUT DE USUÁRIO	86
	APÊNDICE H – EXEMPLO DE PÁGINA PRIVADA DO USUÁRIO	87
	APÊNDICE I – PÁGINA DE CONTROLE DE USUÁRIOS - ACESSO DE ADMINISTRADOR	94
	APÊNDICE J – CÓDIGO ARDUINO	100
	APÊNDICE K – CÓDIGO EMONLIB.H	106
	APÊNDICE L – CONEXÃO DO PROTÓTIPO COM O BANCO DE DADOS	118
	APÊNDICE M – PÁGINA HTML PARA VERIFICAR CONEXÃO DO PROTÓTIPO COM O BANCO DE DADOS	120

1 INTRODUÇÃO

No Brasil grande parte dos consumidores de energia, principalmente na modalidade residencial, fazem o uso de eletricidade de maneira indiscriminada, não havendo grande preocupação em observar e desligar equipamentos quando não estão em uso. Outro hábito que reforça esse padrão é o método que as concessionárias de energia utilizam para notificar o cliente, através de uma fatura mensal que retrata somente o consumo geral do período, onde não há como o consumidor mensurar ou sequer traçar seu próprio perfil de consumo de energia ou perceber alterações significativas durante o período em questão. Temas como economia de energia elétrica, consumo inteligente e eficiência energética tem sido amplamente divulgado e discutido em várias esferas e meios de consumidores de energia, sendo assim, a possibilidade de gerar organicamente no consumidor, principalmente de pequeno e médio porte, hábitos de consumo que resultem em um comportamento cotidiano de gasto consciente pode ser bastante vantajoso para as fabricantes de sensores, medidores e *gadgets* de monitoramento de energia e, principalmente, de extrema relevância para a economia desejada pelo consumidor (MILAGAIA, 2015).

Com relação ao consumidor, que está buscando se inserir ou já integra o sistema de geração distribuída, os hábitos de controle e consumo consciente de energia são ainda mais significativos. Isso porque, o excesso de geração produzido por ele é diretamente injetado na rede e convertido em créditos energéticos que são abatidos no valor a ser pago para concessionária. Nesse caso o desperdício de energia é altamente prejudicial visto que o consumidor pode acabar perdendo mais dinheiro, pois não vai alcançar economia financeira na hora de pagar a fatura mensal e ainda vai ter o prejuízo causado pelo investimento passível de desgaste por uso da usina fotovoltaica instalada.

A utilização de Sistemas Fotovoltaicos *On Grid* tem se mostrado bem atraente ao consumidor nos últimos anos, fazendo com que o número de adesão tenha apresentado aumento crescente, principalmente com o consumidor final que abraça a ideia de gerar a sua própria energia (OLIVEIRA; MARIO; PACHECO, 2021). Usinas Fotovoltaicas podem variar seus níveis de produção por vários fatores, sendo alguns deles a irradiação solar do local, sujidade dos módulos, sombreamento, falta de manutenção ou execução de manutenção errada, decaimento de rendimento por desgaste de uso, danos causados por intempéries climáticas e outros fatores situacionais (LIMA *et al.*, 2022). Por apresentar tantas possibilidades capazes de afetar diretamente a produção da usina, possuir método ou ferramenta que seja capaz de realizar monitoramento dessa produção e exportar os dados adquiridos poderia auxiliar o sistema a alcançar melhor eficiência quando comparado ao sistema atual sem a possibilidade de feedback.

No mercado atual de geração distribuída, segundo a Resolução Normativa n.º 687 (ANEEL, 2015), existem duas configurações da usina com relação a carga: primeiro, o autoconsumo em que a carga está no mesmo local da geração, neste caso parte da energia gerada

dentro dos horários de produção é consumida instantaneamente pelo local de instalação e somente o excedente é injetado na rede, fora do horário de geração o consumo do local se dá inteiramente da concessionária de energia. Segundo, o autoconsumo remoto, em que a carga recebe somente os créditos da concessionária em forma de abatimento de fatura e saldo, isso acontece quando o local de geração e da unidade consumidora são divergentes. Apesar de ser um conceito simples, este sistema pode ser uma medida significativa para auxiliar os consumidores a estabelecer um comportamento de controle e economia de energia, principalmente em caso de usinas residenciais ou comerciais de pequeno porte que são modelos de instalação mais fáceis de gerenciar.

É válido observar, que embora as faturas emitidas por todas as concessionárias de energia do Brasil estejam submetidas a estrutura definida pelo Módulo 11 do Procedimentos de Distribuição de Energia Elétrica no Sistema Elétrico Nacional - PRODIST (ANEEL, 2017), é possível que a maneira de disposição das informações possa variar de acordo com *layout*, sistemas tarifários e impostos de acordo com cada região, o que pode facilitar ou dificultar a compreensão de informações por parte do cliente convencional. Outra barreira significativa no ato de adesão de uma usina fotovoltaica é conceito de consumo instantâneo, uma vez que é contabilizado no valor total de geração da usina porém, por ser consumido imediatamente, não chega a ser registrado no valor de energia injetada na rede da concessionária, sendo assim, somente com o uso de um equipamento que monitora o consumo mensal dentro do quadro de distribuição que este cliente conseguirá visualizar o consumo instantâneo, que será o valor total registrado pelo medidor de consumo subtraído o valor registrado pela concessionária de energia.

Com o intuito de possibilitar um método mais intuitivo para compreensão do consumo de energia, e todo o sistema de geração de energia até o momento de abatimento de créditos energéticos caso tenha uma usina fotovoltaica instalada, uma opção seria projetar um sistema capaz de realizar a aquisição de todos os dados pertinentes, interpretar e calcular dados caso necessário, registrar todas as informações de consumo e, por fim, dispor todas as informações ao usuário de uma maneira intuitiva e de fácil acesso. Sendo assim, a maneira mais viável para o sistema proposto seria estabelecer uma estrutura em duas vertentes, sendo elas: um medidor de consumo em kWh, para registrar o consumo diretamente do quadro geral de distribuição da unidade desejada; Uma plataforma *online*, composta por um site e um *bot* programado, capaz de fazer a aquisição dos valores de geração da usina fotovoltaica e as informações importantes fornecidas pela concessionária, receber os dados de consumo lidos pelo *software*, calcular o consumo instantâneo da unidade em questão e, por fim, dispor todas essas informações em uma plataforma *online* e em forma de relatório.

1.1 Objetivos Principais e Específicos

O objetivo principal deste projeto é propor e desenvolver ferramenta de gestão de sistemas de energia, contemplando a geração de usina fotovoltaica, fatura emitida pela concessionária

de energia e o consumo do cliente diretamente em seu imóvel. Os objetivos específicos deste projeto são:

- Criar Application Programming Interface (API) para efetuar a aquisição de dados de geração do sistema de monitoramento da usina fotovoltaica; buscar todos os dados relevantes no site da concessionária de energia sobre a unidade consumidora em questão; manipular e preparar os dados para o envio a um Banco de Dados.
- Comparar os dados registrados no Banco de Dados para traçar o perfil de geração, consumo e créditos do cliente;
- Verificar, caso exista uma usina fotovoltaica, se o perfil de consumo está condizente com a geração projetada, e, caso não esteja, alertar o usuário sobre a situação;
- Criar uma plataforma *online* para exibir ao cliente, de maneira transparente e simplificada, dados como valor da tarifa, energia injetada, consumo e créditos registrados na fatura de energia fornecida pela concessionária;
- Emitir automaticamente e enviar ao cliente relatório mensal com todas as informações importantes do sistema de energia gerenciado;
- Projetar e construir dispositivo protótipo capaz de registrar consumo de energia de uma unidade consumidora ao ser instalado no Quadro de Distribuição Geral, que seja capaz de registrar e disponibilizar os dados de consumo lidos de maneira instantânea;
- Instalar, testar e estudar os dados obtidos com a leitura para validação do funcionamento do sistema.

1.2 Justificativa

Uma barreira que pode ser encontrada para este usuário em potencial é o fato de que, para acessar todas as informações de seu sistema de energia completo, ele deve acessar o site da concessionária de energia, o do aparelho de monitoramento de consumo e, no caso de possuir uma usina fotovoltaica instalada, o site ou aplicativo do inversor solar que monitora a geração. Prezando pela lógica de gerar uma motivação neste usuário em potencial, ter um sistema de monitoramento de consumo que integre um sistema que seja capaz de realizar este gerenciamento é uma vantagem competitiva, técnica e até mesmo econômica. Sendo assim, uma opção é desenvolver um medidor de consumo de energia elétrica por meio da medição de tensão e corrente elétrica, utilizando uma plataforma de prototipagem eletrônica com acesso a um servidor externo para armazenar os dados lidos, possibilitando que o usuário possa acessar em tempo real o valor que está sendo consumido por ele.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Hardware: Dispositivo Medidor de Consumo

2.1.1 Sistema Elétrico de Potência

Sistema elétrico de potência é definido como rede de componentes elétricos usados para geração, transmissão e distribuição de energia elétrica. (SCHAVEMAKER; SLUIS, 2017).

Historicamente, os sistemas elétricos de potência foram desenvolvidos com base na geração centralizada com redes de transmissão e distribuição, transportando energia até os consumidores. Composto por uma parcela de combustível fóssil significativa em sua matriz, o setor de energia responde por grande parte das emissões gases poluentes, indo contra a necessidade de redução da produção de agentes agressores ao meio ambiente. (MIRANDA, 2012) Portanto, é perceptível que a redução de emissões de gases poluentes está diretamente relacionada à descarbonização do setor de energia através da procura constante de implantação de fontes de energia limpas e renováveis.

2.1.2 Geração Distribuída

Geralmente a geração distribuída (GD) é definida com base em seu tamanho e localização, mas alguns países a definem por algumas características básicas como: renovável, cogeração, não despachável ou alguma outra modalidade específica. Nas definições da literatura para geração distribuída (GD) não são consistentes, mas diferentes termos e definições são usados. Uma definição bastante consolidada e aplicada no Brasil é uma usina geradora conectada diretamente à rede de distribuição através de um medidor, geralmente bidirecional (NASCIMENTO *et al.*, 2014). Um modelo deste método de geração que tem se mostrado promissor no Brasil é a geração solar fotovoltaica que, além dos benefícios ambientais, apresenta benefícios sistêmicos energéticos e elétricos, no entanto é preciso estar ciente de que também há custos e riscos.

No Brasil, após a Resolução Normativa nº 482/2012 (ANEEL, 2012), a geração distribuída se difundiu através da micro e minigeração distribuídas de energia elétrica por meio de usinas fotovoltaicas. Com o intuito de democratizar o acesso a este modelo de geração por mais consumidor, a ANEEL adotou medidas como reduzir os custos e prazos para a conexão na rede, harmonizar o sistema de compensação de energia elétrica, possibilitar a adesão do sistema para o maior número de consumidores possível, além de melhorar as informações na fatura na resolução Normativa nº 687/2015 (ANEEL, 2015), revisando a resolução normativa nº 482/2012.

2.1.3 Geração Fotovoltaica On Grid

Entre as várias tecnologias de produção de energia renovável, a solar fotovoltaica é uma das fontes mais utilizada quando se fala em número de unidades instaladas por todo o território

nacional, uma vez que o Brasil apresenta um nível de irradiação solar satisfatório, fazendo com que as usinas sejam capazes de produzir energia durante o ano inteiro, oferece redução contínua de custos ao longo dos anos, é um sistema estável e precisa de manutenção simplificada.

Existem duas classes do sistema de energia solar, um deles é o sistema Off-Grid ou autônomo, onde a geração e o consumo de energia não se conecta à rede de distribuição, logo, não injeta energia na rede da concessionária. A outra classe, que será utilizado no presente projeto, é o sistema On Grid ou conectado à rede da concessionária realizando injeção de energia, consumindo diretamente da rede em horário que não há produção da usina e recebendo créditos energéticos (CADAVAL *et al.*, 2013).

Um ponto importante que deve ser levantado sobre uma usina fotovoltaica On Grid é sua estrutura básica que deve ser composta por uma parte para sustentar os módulos fotovoltaicos para possibilitar melhor posicionamento e inclinação, com o intuito de captar a maior quantidade de radiação solar possível; um medidor bidirecional para medir o consumo na unidade instalada e também registrar a quantidade injetada na rede elétrica; o inversor solar, que é responsável por transformar a corrente elétrica contínua para alternada, garante o fluxo regular da eletricidade e segurança, além de possibilitar a sincronia com a rede da concessionária; materiais elétricos para conexão e proteção do sistema como disjuntores, conectores e cabeamento elétrico; as células fotovoltaicas que captam a luz do sol, fazendo com que os elétrons se movimentem e produzam a corrente elétrica, vale ressaltar que um painel solar de qualidade tem vida útil de até 25 anos.

2.1.4 Medição de Consumo

Ter acesso a um sistema de medição e gerenciamento de energia possibilita ao usuário maior autonomia, ampliação de conhecimento e o controle irrestrito do seu consumo de energia, possibilitando que a redução de consumo desnecessário seja estabelecida de forma natural. Atualmente, os medidores de consumo disponíveis no mercado apresentam *softwares* e aplicativos próprios para que este usuário seja capaz de acompanhar seu consumo em tempo real, acessar seu histórico de consumo em um determinado período e até acessar estes dados em forma de gráficos. É válido ressaltar que para desencadear um comportamento natural o sistema oferecido a este usuário deve ser o mais acessível e completo possível, com o máximo de informação reunida dentro de uma plataforma única de fácil acesso.

No mercado atual existem diversos modelos de medidores de consumo com diversas características como a variação conforme a quantidade de fases do sistema, método de conexão invasivo ou não invasivo, conectividade, memória interna para gravar as leituras, além de outras características específicas que variam de acordo com o fabricante. Estes dispositivos utilizam dados de leitura da corrente e da tensão para cada fase do sistema, enquanto para o cálculo dos valores RMS destas grandezas são realizadas leituras instantâneas registradas com períodos que variam de acordo com os modelos e fabricantes (JÚNIOR; ÓPTICA; NUMÉRICA-GOMNI, 2014).

As medições de tensão podem ser feitas através do divisor resistivo ou transformador de potencial, já as medições de correntes podem ser feitas com resistor *shunt*, transformador de corrente, bobinas de Rogowski ou sensores de efeito Hall. Os sinais condicionados de tensão e corrente são enviados ao microcontrolado, com as informações sendo tratadas via *software* de forma a obter os resultados esperados. Após o tratamento, podem ser disponibilizados através do *display* na tela do produto; através da interface do usuário em algum site ou aplicativo; existem também modelos que possuem *display* e interface e outros com apenas o *display*, podem ser individuais medidores para cargas individuais com leitura independente ou podem ser instalados de forma a ter uma leitura geral ou por setor (COSTA; SERMANN; SILVA, 2016).

Uma possível dificuldade consiste no fato de que para o usuário, possa ter acesso a todas as informações de seu sistema de energia, ele deve acessar o site da concessionária de energia, o do aparelho de monitoramento de consumo e, no caso de possuir usina fotovoltaica instalada, o site ou aplicativo do inversor solar que monitora a geração. Prezando pela lógica de gerar uma motivação nesse usuário em potencial, ter um sistema de monitoramento de consumo que integre um sistema que seja capaz de realizar esse gerenciamento é uma vantagem competitiva, técnica e até mesmo econômica (MENEZES *et al.*, 2020). Sendo assim, uma opção é desenvolver um medidor de consumo de energia elétrica por meio da medição de tensão e corrente elétrica, utilizando uma plataforma de prototipagem eletrônica com acesso a um servidor externo para armazenar os dados lidos, possibilitando que o usuário possa acessar em tempo real o valor que está sendo consumido por ele.

2.2 Software: API e Plataforma Online

Para dispor os valores de geração, consumo e dados fornecidos pela concessionária de energia armazenados no banco de dados em uma plataforma para o usuário/cliente, uma solução possível é desenvolver site hospedado em servidor Web. Este site pode apresentar página de registro e login do usuário, página inicial, página particular para cada usuário com seus registros e perfil de acesso de administrador com capacidade de acessar e gerenciar todos os usuários do sistema.

Para que um site possa ser disponibilizado e acessado de maneira *online*, é preciso que todos os arquivos estejam armazenados em um servidor web, possua um domínio e banco de dados. Considerando as diversas opções disponíveis é possível obter soluções que sejam gratuitas e intuitivas.

2.2.1 Application Programming Interface

As API (*Application Programming Interface*) permitem que desenvolvedores tornem processos complexos e repetitivos, total ou parcialmente autônomos além de altamente reutilizáveis com reduzidos trechos de códigos. A API é muito flexível e pode ser usada em sistemas baseados na *web*, sistemas operacionais, sistemas de banco de dados e *hardware* de computador

e usam protocolos definidos para permitir que desenvolvedores criem, conectem e integrem aplicativos rapidamente e em escala. As APIs se comunicam por meio de conjunto de regras que definem como computadores, aplicativos ou máquinas podem se comunicar entre si (IBM, 2020). Para o projeto em questão, foi desenvolvido API composto por *scripts* na linguagem *python* para realizar o *web scraping* no site da concessionária de energia ENEL-GO e no sistema de monitoramento da geração da usina fotovoltaica que foi utilizada como teste.

2.2.2 Web Scraping

Web scraping, também conhecido como raspagem da Web, é o processo de usar “robôs” (bots) para extrair conteúdo e dados de páginas ou sites *online* e é bastante empregado em sites de comparação de preços para recuperar automaticamente preços e descrições de produtos para sites afiliados ou empresas de pesquisa de mercado para extrair dados de fóruns e mídias sociais (FARIAS; ANGELUCI; PASSARELLI, 2021). *Bots* legítimos são identificados com a organização para a qual eles raspam e obedecem ao arquivo *robot.txt* de um site, que lista as páginas que um bot tem permissão para acessar e aquelas que ele não pode. Também é a prática utilizada para construção de *bots* maliciosos que, por outro lado, rastreiam o site, independentemente do que o operador do site tenha permitido com o intuito de roubar dados protegidos por direitos autorais.

2.2.3 Banco de Dados

Um banco de dados é a coleção organizada de informações estruturadas, ou dados, geralmente armazenados eletronicamente em um sistema de computador. Os dados e o SGBD (Sistema de Gerenciamento de Banco de Dados), juntamente com os aplicativos associados a eles, são chamados de sistema de banco de dados, muitas vezes abreviado como banco de dados. Os dados podem ser facilmente acessados, gerenciados, modificados, atualizados, controlados e organizados de acordo com a estrutura de programação Structured Query Language (SQL) (ELMASRI; NAVATHE, 2010). Uma tabela é a unidade mais básica de um banco de dados e consiste em linhas e colunas de dados. As tabelas são o tipo mais comum de objetos ou estruturas de banco de dados usados para armazenar ou referenciar dados em um banco de dados relacional.

Para o sistema de gerenciamento composto por diversas informações alocadas em lugares diversos, a integração do banco de dados podem possibilitar a reunião de todas as informações necessária em único local, oferece a opção de atualização em tempo real de maneira automática ou manual e, caso esteja hospedado em um servidor *online*, pode ser consultado de qualquer lugar através de qualquer dispositivo com conexão com a *internet*.

2.2.4 Hypertext Markup Language

Hypertext Markup Language é uma linguagem de computador que renderiza a maioria das páginas da Web e aplicativos *online*. Hipertexto é um texto usado para se referir a outro texto;

A linguagem de marcação consiste em conjuntos de marcações que informam aos servidores web qual é o estilo e a estrutura de um documento. Como o HTML é amplamente usado para colocar *hiperlinks*, os usuários podem navegar facilmente e adicionar *links* entre páginas e sites relacionados (BERNERS-LEE; CONNOLLY, 1995).

O HTML é uma das linguagens de programação mais simples e indicadas para desenvolvedores *web* que buscam estruturar sites pouco complexos, com finalidade acadêmica ou de uso público gratuito. Partindo dessa recomendação, como o site a ser desenvolvido apresenta finalidade acadêmica, todas as páginas vão ser baseadas nessa linguagem, uma vez que se integra bem com as demais linguagens utilizada, possibilita a integração do *layout* responsivo desejado e oferece a função de visualização da página por meio de qualquer navegador de *internet*.

2.2.5 PHP: Hypertext Preprocessor

PHP é uma linguagem de *script* de uso geral de código aberto amplamente usada que é especialmente adequada para o desenvolvimento da *Web* e pode ser incorporada ao HTML. Essa linguagem é frequentemente utilizada para automatizar a execução de tarefas em um ambiente de tempo de execução específico como, por exemplo, usar um *script* para validar com segurança se todos os campos do formulário foram preenchidos antes de serem enviados de volta ao servidor, no qual o *script* será executado e verificará todos os campos quando o usuário enviar o formulário (DUCKETT, 2022).

Para uma plataforma com a característica de autenticação de dados de acesso através de informações como perfis particulares e senhas, considerando que a plataforma a ser acessada possui finalidade acadêmica, uma estrutura PHP é uma maneira mais indicada por sua facilidade estrutural. Uma aplicação que será utilizada é a verificação e autenticação da existência do registro que houve tentativa de *login*, autorização de acesso e encaminhamento para páginas permitidas, bem como o bloqueio de acesso em páginas que o usuário em questão não possui autorização de acesso.

2.2.6 Cascading Style Sheets

CSS significa *Cascading Style Sheets language* e é usado para estilizar elementos escritos em uma linguagem de marcação como HTML, essa relação entre HTML e CSS está fortemente marcada e perceptível, porque HTML é a base de um site e CSS é toda a estética do mesmo. Antes de usar CSS, todos os estilos tinham que ser incluídos na marcação HTML, já com as devidas atualizações da integração das *tags* CSS, é possível estilizar tudo em um arquivo diferente, criando o *design* na *Style Sheets* e depois incorporando os arquivos em cima da marcação HTML. Isso torna a marcação HTML real muito mais limpa e fácil de manter. Vale ressaltar que o CSS permite que você tenha vários estilos em uma página HTML, o que torna as possibilidades de personalização quase infinitas (LIE; BOS, 2005).

As configurações de elementos visuais estáticos da plataforma *online* devem ser desen-

volvidos através da linguagem CSS, uma vez que não necessite ações de responsividade dos itens desejados, facilita algumas definições rápidas como alteração de cores e tamanhos de elementos e textos.

2.2.7 *Bootstrap*

Bootstrap é um *framework* de desenvolvimento web gratuito e de código aberto, em outras palavras, é uma ajuda aos desenvolvedores *Web* a criar sites mais rapidamente pois não precisam se preocupar com comandos e funções básicas. O principal objetivo do *Bootstrap* é criar sites responsivos e *mobile-first*, ou seja, páginas que sejam capazes de se ajustar automaticamente de acordo com as dimensões do dispositivo utilizado pelo usuário (QUEIRÓS, 2017).

A função de responsividade do bootstrap deve ser empregada na plataforma pois realiza automaticamente a parte visual do site de acordo com as dimensões do dispositivo que está realizando o acesso, sendo assim, imagens e textos são alterados ou retirados automaticamente para não causar quebras na parte visual caso o usuário deseje acessar o site de algum dispositivo mobile.

2.2.8 *Web Server*

Em termos simples, um servidor *web* é um computador que armazena, processa e entrega arquivos de sites para navegadores da *web*. Do lado do *hardware*, um servidor *web* se conecta à *internet*, o que permite a troca de dados ou arquivos entre outros dispositivos que também estejam conectados, o *hardware* do servidor da *Web* também armazena o *software* do servidor da *Web* e o software de servidor da *Web* controla como os usuários que acessam os arquivos hospedados (CONTRIBUTORS, 2022).

A utilização do banco de dados no projeto faz com que seja necessário o uso de um servidor para acesso e comunicação com o dispositivo de monitoramento de consumo e com a plataforma *online*. Existe a opção de servidor local por meio da própria placa que comporta o microcontrolador porém, para evitar perda de dados caso ocorra danos no dispositivo físico, o indicado é o uso de servidor *online*.

2.2.9 *Web Host*

A hospedagem na *web* é um serviço *online* que torna o conteúdo de um site acessível na internet, ao adquirir um plano de hospedagem usuário aluga um espaço em um servidor físico para armazenar todos os arquivos e dados do site (RIPE, 2021). Essa aplicação vai comportar a estrutura do site, todas as páginas de acesso público ou privado, vai exibir ao usuário através do DNS de direcionamento (URL) da hospedagem.

2.2.10 Estruturação para Plataforma *Online*

Para ser possível construir uma plataforma *online*, que atenda às necessidades do projeto, é preciso que possua a função de criação de acesso por login de usuário com autenticação do mesmo, sendo assim, o primeiro passo é estabelecer a criação e conexão com um banco de dados *online*. Uma maneira é criar um *script* PHP com os dados de acesso do servidor para estabelecer a conexão com o banco de dados, após sua criação esse arquivo pode ser importado em qualquer outro *script* que precise acessar o mesmo banco de dados.

A página inicial, por se tratar de um projeto acadêmico, pode ser escrita na linguagem HTML, sua estrutura geralmente é composta por "*head*" e "*body*". O *script* geralmente é iniciado pelo "*head*", onde se coloca dados e configurações que não devem ou precisam ser exibidos na página de maneira visível ao usuário. Para o topo da página uma escolha apropriada uma barra de menu de cor sólida, com a logo à esquerda e botões à direita, esse tipo de disposição de menu facilita o direcionamento do usuário entre as páginas disponíveis no site, acessos, login e logout. O "*body*" é a estrutura que compõe a página e é visível para o usuário, comporta e exhibe imagens, textos e botões de direcionamento. Por se tratar de um projeto acadêmico, para compor toda a parte estética da página foram escolhidas imagens de domínio público

Por se tratar de um sistema de gerenciamento de dados vinculados a usuários específicos, os *logins* de usuário devem ser previamente cadastrados no sistema diretamente no banco de dados. A página de *login* pode ser criada caso o usuário queira acesso aos seus dados por múltiplos *logins*, sendo assim, ele faz o registro nessa página e solicita o vínculo entre o novo usuário com sua página privada. Para este *script*, o primeiro passo é utilizar é importar o *script* de configuração para efetuar a conexão com o banco de dados de usuários e, em seguida, verificar se o nome de usuário e senha são válidos para que o registro seja concluído. A estrutura da página pode ser feita em HTML, para o "*head*" utiliza a estrutura da página inicial caso ela seja o padrão definido a ser seguido, já para o "*body*" é preciso configurar todos os campos de entrada de textos de usuário e senha, botão de registro e *links* de redirecionamento para página de *login*.

Para realizar a autenticação de usuários cadastrados ou, em caso de cadastro ou dados inválidos, informar que o usuário não foi encontrado, a estrutura utilizada pode ser criada em PHP, fazendo conexão com o banco de dados através do arquivo de configuração. A área de login pode ser montada de maneira bem simples e padrão, com campo de entrada para usuário e senha, além do botão de envio dos dados, também é válido adicionar um link de redirecionamento para a página de registro caso não seja encontrado um usuário válido para as informações digitadas.

Uma parte muito importante de um site é dar ao usuário toda a autonomia possível para manipulação de sua página pessoal, sendo assim, é primordial criar um botão de *logout* para que a sessão possa ser encerrada automaticamente com apenas um clique.

Cada login de usuário pode ser vinculado a uma página privada que apresenta todos os dados coletados sobre a geração de sua usina, consumo de energia e créditos da concessionária

de energia e demais dados de leitura. Para encaminhar de maneira correta cada usuário para sua conta, pode ser criado um trecho de código em PHP para sincronizar os dados de usuário com sua página privada e fazer esse redirecionamento. Ao entrar na própria página pessoal, o usuário tem todos os seus dados coletados (geração, consumo e créditos) dispostos em uma tabela bem intuitiva e de fácil compreensão. Como auxílio visual, pode ser adicionado um gráfico de barras do tipo "Morris" com um conjunto de três barras referentes a cada mês registrado, sendo elas: geração na cor verde, consumo na cor vermelha e autoconsumo (valor do consumo subtraído do valor de geração) em azul.

Se tratando de um sistema de gerenciamento, é imprescindível que o administrador do sistema seja capaz de buscar dentro do banco de dados todos os registros referentes a um cliente específico, de um mês específico de vários clientes ou qualquer outra situação semelhante. Para isso, o método adotado pode ser a construção de uma tabela em uma página com PHP e HTML combinados, respeitando a configuração de estilo citada anteriormente e método de autenticação PHP.

3 METODOLOGIA

3.1 Montagem do *Hardware*

Com o intuito de criar um dispositivo protótipo que seja capaz de registrar a leitura de consumo da unidade consumidora ao qual será instalado, o primeiro procedimento que deverá ser efetuado é a listagem dos componentes principais e suas determinadas funções. Será necessário o uso de um Módulo WiFi que possa fornecer a qualquer microcontrolador acesso a uma rede WiFi e que apresente uma pré-programação que seja compatível diretamente com o Arduino. O modelo que deverá apresentar uma alta velocidade de processamento e de comunicação, (para que ocorra o envio de todas as informações sem que haja perdas por causa da quantidade de informações), protocolos de comunicação (PC) utilizado; Conversor analógico-digital (ADC) de 6 canais com interface SPI (*Serial Peripheral Interface*).

O uso desse componente será necessário para mitigar possíveis problemas nas portas analógicas de comunicação do Módulo WiFi; Sensor de corrente (SC) que apresente características não invasivas, uma vez que será instalado em um quadro de distribuição já existente e ativo, facilitando a instalação e o manuseio no sistema, com um valor de corrente suportado compatível com os níveis máximos esperados em picos do sistema para garantir maior segurança na instalação; Sensor de tensão (ST) que seja um transformador de tensão capaz de medir tensão CA até 250 volts com saída analógica, caso tensão de entrada seja alterada, a tensão de saída também mudará.

Para efetuar medição de tensão serão utilizados bornes, nos quais serão conectados cabos responsáveis por adquirir tensão; a saída do borne deverá ser conectada ao sensor de tensão utilizando o neutro comum. A saída do sensor poderá apresentar uma tensão incompatível com o sensor que será escolhido, devido a isso, uma possível solução para o problema poderá ser realizado a correção desse valor com a adição de diodos em série com cada um dos sensores.

Na leitura de corrente será utilizado um sensor de corrente não invasivo, com o intuito de evitar interrupção do funcionamento do sistema elétrico no local de instalação, facilitar a montagem e também garantir maior segurança no processo de instalação. A indicação é utilizar um sensor que apresente um pino de 3,5mm em sua saída, serão adicionados na placa ilhada 3 conectores compatíveis. Para aquisição da corrente, o sensor deverá se comportar como uma espécie de transformador, assim o fio que passa através do sensor irá operar como o primário e as saídas do sensor como um secundário, sendo um fio de referência e o outro de sinal. As portas de corrente serão conectadas nas entradas 5, 6 e 7 e as de tensão nas entradas 0, 3 e 4 do conversor Analógico-Digital. O conversor será conectado ao módulo *Wifi* através as portas 12,13,14 e 15, para alimentar o sistema será conectada uma fonte de tensão. Também poderá ser adicionado um regulador de tensão para possibilitar alcance de maior estabilidade e filtrar possíveis ruídos.

3.1.1 Conexão do Protótipo com Banco de Dados

Por se tratar de um protótipo para um projeto acadêmico, no qual não há intenção de comercialização, alguns dos componentes a serem escolhidos para integrar a montagem podem apresentar algumas limitações que não sejam prejudiciais ou críticas. Uma limitação significativa que pode ser encontrada é o envio das leituras da memória interna do equipamento para o servidor que receberá esses dados, pois a leitura instantânea poderá sofrer interrupção no momento do envio, ou seja, haverá um período indeterminado de ausência de leituras de consumo (Apêndice L).

Para acompanhar os dados lidos, uma vez que o protótipo não terá *display* para informações, será criada uma página HTML, conectada ao banco de dados, com a finalidade de exibir todas as leituras efetuadas de maneira *online*. Nessa página específica não haverá tratamento de dados ou implementação de elementos visuais, uma vez que seu propósito foi ser somente verificar andamento das leituras sem precisar abrir o servidor local do microcontrolador ou o banco de dados (Apêndice M).

3.2 Requisito de Software

Para realizar todo o procedimento de aquisição e cruzamento de dados, será preciso estruturar um *script* por meio de programação que seja capaz de executar todo o trabalho de maneira rápida e correta e, por fim, ainda seja capaz de gerar um relatório de acompanhamento com os dados trabalhados (Apêndice A). A linguagem de programação que será utilizada deverá apresentar o máximo de características necessárias possíveis, por meio de comandos ou bibliotecas, para recolher dados disponíveis em sites *web*, converter e manipular dados recolhidos, enviar os dados para um banco de dados hospedado em um web host, gerar um relatório geral em modelo de folha A4 e enviar o mesmo diretamente no email do usuário.

A linguagem utilizada será o Python, que é uma linguagem que comporta todas as características necessárias para executar todo o trabalho no mesmo *script*, facilita o trabalho de *web scraping*, possibilita a geração de PDFs personalizados com imagens e textos e também pode desempenhar atividades com emails como leitura e envio de mensagens. Para a execução dos testes necessários durante o projeto, será utilizada uma unidade consumidora trifásica com uma usina fotovoltaica instalada "junto a carga".

Para possibilitar uma gestão de dados mais precisa, na tentativa de reduzir possíveis erros de cálculo, o período de aquisição de dados deverá ser ajustado de acordo com a data de leitura de consumo feita pela concessionária de energia, presente no faturamento entregue na residência do cliente mensalmente. O modelo será estruturado para uma única unidade consumidora que receberá teste e, para definir o período de datas que serão trabalhadas, há duas maneiras: fazer média entre os dias de leitura mais frequentes, definindo assim datas fixas iniciais e finais; a segunda maneira, que será adotada no projeto, é desenvolver um trecho de código totalmente adaptável para ser ajustado automaticamente de acordo com a data de execução do script.

Nesse trecho de código as bibliotecas *time* e *datetime* deverão ser utilizadas para determinar o mês e ano atual, sendo assim, com esses dados também será possível determinar o mês anterior e, conseqüentemente, determinar as datas "início"(ano atual - mês anterior - dia da leitura do mês anterior) e "fim"(ano atual - mês atual - dia da leitura do mês atual). Com as datas de "início" e "fim" definidas, o trecho de código irá gerar um *array* que contendo todas as datas contidas no intervalo desejado. Com intervalo definido, deverá ser implementado ao script o código capaz de fazer o login automático no site de monitoramento de geração da usina fotovoltaica, seja ele diretamente no inversor solar ou com um equipamento externo instalado no sistema. Um método de *Web scraping* adequado para esse esse trabalho é uma requisição em JSON para capturar somente os dados de geração total diário em kWh.

3.2.1 *Web Scraping* no Site da Concessionária

A aquisição de dados por meio de *scraping* no site da será desenvolvida com o *Playwright*, que é uma estrutura de automação de navegador com API disponíveis em Javascript, Python e também oferece suporte à opção de drivers da *web*. Primeiramente será necessário definir um *array* com os dados de login e senha do cliente para autenticação no site da concessionária e, em seguida, a estrutura que inclui o *Playwright*. Após abrir automaticamente um navegador e redirecionamento para página de login, o *bot* aceitará os cookies da página, selecionará o formulário de login CPF/CNPJ, preencher os campos de *login* e submeter os dados. Com a autenticação do usuário feita corretamente, o *Playwright* ainda será capaz de passar por *pop-ups* (janela que aparece no navegador do usuário) de avisos do site e ir diretamente a página inicial do usuário.

Utilizando como base o site da Enel-GO, que foi desativado posteriormente ao final dos testes por motivos de venda a outra empresa, o primeiro dado a ser retirado pelo *script* será o valor da última fatura registrada pela concessionária, será utilizado o comando seletor do *Playwright page locator* para encontrar e acessar a página "conta" e, em seguida, encontrar o item "segunda via da fatura" que contém o valor desejado. Para resgatar os dados de faturamento da unidade consumidora referente ao sistema fotovoltaico conectado à rede, o script localizará a área de "minigeração e microgeração", em ordem: mês de referência de leitura, data de leitura, consumo de energia, energia injetada, crédito recebido, valor da tarifa com e sem imposto e os números registrados no medidor. Após serem localizados e recolhidos, os valores serão armazenados em uma variável para facilitar a manipulação.

Por se tratar de dados armazenados como texto (*string*), os valores que vão ser manipulados por operações matemáticas devem ser convertidos para variável de ponto flutuante (*float*), sendo elas: consumo mensal, energia injetada, crédito recebido e tarifa com imposto; para complementar os dados de consumo e fornecer maior entendimento ao cliente, também será calculado o valor de autoconsumo do sistema. Vale ressaltar que todo sistema fotovoltaico instalado junto à carga, um exemplo seria as usinas residenciais, apresenta o autoconsumo que é a energia gerada e consumida instantaneamente durante o período de geração. Esse valor de

autoconsumo pode grosseiramente ser calculado subtraindo o valor de geração registrado pela concessionária de energia do valor total de geração registrado pelo inversor solar.

Para gerar o relatório do cliente, poderá ser criado um *template* em algum aplicativo externo para ser utilizado em formato PDF para compor toda a parte visual estática, ou seja, *layout* padrão a ser utilizado para todos os relatórios gerados posteriormente. Já para a parte dinâmica, que vai incluir os dados específicos de cada cliente, a opção é escrever e gerar o gráfico e tabela diretamente com bibliotecas python. Com o intuito de facilitar o entendimento dos dados mais cruciais de todo esse processo, sendo eles a geração e o consumo registrado pela concessionária, será utilizada uma representação visual em forma de gráfico horizontal de barras paralelas através de um *array* formado por 12 parâmetros, onde cada deles significa um conjunto de dados mensal (mês referência, geração, consumo). A criação dos gráficos com a biblioteca utilizada seguirá um padrão de organização sendo o primeiro conjunto mensal do gráfico o maior valor de geração e o último conjunto o de menor valor de geração. Uma parte crucial para gerar o gráfico será definir corretamente todas as configurações de espessura das barras, cores diferentes para geração e consumo, tamanho de fonte para textos e espaçamento vertical entre as barras, também é necessário definir as configurações de visibilidade dos eixos x e y, desativação de *grid*, espaçamento das margens (topo, rodapé, direita e esquerda) e ativação da legenda do gráfico. Na tentativa de evitar quebra na configuração visual do gráfico, uma vez que sempre será projetado sobre um *template* padrão, a melhor opção é exportar esse gráfico gerado como uma imagem e, em seguida, buscar essa imagem novamente com outra função, toda a parte de inclusão de imagem e textos sobre o modelo padrão previamente definido foram adicionados com a biblioteca *reportlab*.

Para concluir a parte de texto, será preciso definir todo o posicionamento de cada dado raspado (do monitoramento de geração ou da concessionária de energia), calculado e importado do banco de dados. Toda a definição de posicionamento utilizará o comando *drawString* da biblioteca *reportlab*, definindo as coordenadas de posicionamento, sendo: horizontal, vertical, variável ou texto. Após importar a imagem do gráfico gerada no passo anterior, será necessário definir o posicionamento, dimensões da imagem e a suavização visual da sobreposição da imagem em um arquivo já existente. Para finalizar do relatório, o *script* fará a integração do arquivo *template* em PDF com a imagem do gráfico e os textos gerados pelos dados adquiridos e manipulados por todo esse sistema, resultando em um PDF final que será enviado por *email* para o cliente.

3.3 Programação da Plataforma Online

Por se tratar de uma plataforma que permitirá a criação de acesso de usuário e também fará a autenticação do mesmo, o primeiro passo é estabelecer a conexão com um banco de dados *online* através de um *script*. Este *script* poderá ser um arquivo PHP com finalidade de configuração e, após sua criação, esse arquivo poderá ser importado em qualquer outro *script*

que precise acessar o mesmo banco de dados (Apêndice B).

A definição visual e responsiva que será adotada como padrão de estilo do site será inserida dentro do "*body*" da estrutura HTML de cada página da plataforma de maneira individual, permitindo que alterações de cores e estilo possam ser realizadas de maneira isolada sem afetar outras páginas do site. Caso a intenção fosse ter esse estilo padronizado em todas as páginas, sem nenhuma alteração particular, poderia ter sido criado um arquivo CSS padrão e incluído o mesmo em todas as páginas desejadas (Apêndice C).

A página inicial será escrita na linguagem HTML, uma estrutura geralmente composta por "*head*" e "*body*", o *script* será iniciado pelo "*head*", onde se coloca dados e configurações que não devem ou precisam ser exibidos publicamente na página para o usuário. Para o topo da página a escolha será uma barra de navegação responsiva, esse modelo de disposição de menu facilita o direcionamento do usuário entre as páginas disponíveis no site, acessos, login e logout. O "*body*", estrutura que será visível na página para o usuário, será responsável por comportar e exibir imagens, textos e botões de direcionamento. Por se tratar de um projeto acadêmico, para compor toda a parte estética da página, serão utilizadas imagens de domínio público (Apêndice D).

Por se tratar de um sistema de gerenciamento com *feedback* para usuários específicos, grande parte dos *logins* de usuário serão previamente cadastrados no sistema direto no banco de dados, sendo assim, a página de registro disponibilizada caso o usuário queira acessar seus dados por múltiplos usuários. Para essa finalidade, deverá ser feito o registro de usuário e solicitar o vínculo entre o novo usuário com sua página privada. Para esse *script*, é necessário importar o *script* de configuração para efetuar a conexão com o banco de dados de usuários e, em seguida, verificar se o nome de usuário e senha são válidos para que o registro seja concluído. A estrutura da página em HTML utilizará o mesmo "*head*" da estrutura da página inicial (adotada como padrão), já para o "*body*" será configurado todos os campos de entrada de textos de usuário e senha, botão de registro e links de redirecionamento para página de login (Apêndice E).

Na página de *login*, a autenticação de usuários cadastrados ou, em caso de cadastro ou dados inválidos, informará que o usuário não foi encontrado, a estrutura utilizada será criada em PHP com conexão com o banco de dados através do arquivo configuração. A área de *login* será montada de maneira bem simples e padrão, com campo de entrada para usuário e senha além do botão de envio dos dados, também será adicionado um *link* de redirecionamento para a página de registro caso não seja encontrado um usuário válido para as informações digitadas (Apêndice F). Uma parte muito importante de um site é dar ao usuário toda a autonomia possível para manipulação de sua página pessoal, sendo assim, será primordial criar um botão de "*logout*" para que a sessão seja encerrada automaticamente com apenas um clique (Apêndice G).

Cada *login* de usuário será vinculado a uma página privada que apresentará todos os dados coletados sobre a geração de sua usina, consumo de energia e créditos da concessionária de energia e demais dados de leitura. Para encaminhar de maneira correta cada usuário para sua

conta, será desenvolvido um trecho de código em PHP para sincronizar os dados de usuário com sua página privada e fazer esse redirecionamento. Ao entrar na própria página pessoal, o usuário terá todos os seus dados coletados (geração, consumo e créditos) dispostos em uma tabela bem intuitiva e de fácil compreensão. Como auxílio visual, será adicionado um gráfico de barras do tipo "*Morris*" com um conjunto de três barras referentes a cada mês registrado, sendo elas: geração na cor verde, consumo na cor vermelha e autoconsumo (valor do consumo subtraído do valor de geração) em azul (Apêndice H).

Por se tratar de um sistema de gerenciamento, é imprescindível que o administrador do sistema seja capaz de realizar buscas dentro do banco de dados todos os registros referentes a um cliente, de um mês específico de vários clientes ou qualquer outra situação semelhante. Para isso, será realizada a construção de uma tabela em uma página com PHP e HTML combinados, assim como nos scripts anteriores, respeitando a configuração de estilo definida anteriormente e método de autenticação PHP (Apêndice I).

4 RESULTADOS E DISCUSSÃO

4.1 Hardware

Considerando todo o projeto de gerenciamento de consumo de energia do usuário realizado até esse capítulo do projeto, um ponto de suma relevância priorizado foi a possibilidade de oferecer acompanhamento mais dinâmico e acessível, diretamente do quadro de distribuição do imóvel, com sistema compacto e eficaz, sendo assim, a opção adotado foi a construção de um equipamento medidor de consumo com base em IDE Arduino ©. Após a escolha do microcontrolador a ser utilizado, toda a lista de componentes foi construída de maneira que possibilitasse a integração e comunicação com o sistema a IDE Arduino ©, sendo assim, a lista de componentes utilizados foi a seguinte:

Componente	Quantidade
Placa ilhada 5x7 mm	1 un.
Sensor de tensão ZMPT101B, 0 a 250V	3 un.
Sensor de corrente SCT-013-050, 50A	3 un.
Diodo IN4007	9 un.
Capacitor 2200uF	1 un.
Capacitor 1100uF	2 un.
Capacitor 10uF	3 un.
Resistor 6,8 k ohm	6 un.
Conector empilhável para arduino 20 vias	1 un.
Conector Jack P2 J2 estéreo áudio 3,5mm	3 un.
ESP 8266	1 un.
Fonte de 9v 1A	1 un.
LM7805	1 un.
Soquete torneado 8 vias	2 un.
Ferro de solda	
Estanho	
Mcp3208	1 un.
Borne 3 polos - KF-301 3T	1 un.
Borne 1 polo – KF-301 1T	1 un.
Conector emenda 3 polos – WAGO	1 un.
Plug fêmea 10A	1 un.
Cabo paralelo 1,5mm	3 un.
Álcool isopropílico	300ml
Cabo USB (Configuração)	1 un.
Fio para conexões	1 un.

Tabela 1 – Lista de Componentes

Os principais componentes eletrônicos para a montagem do protótipo escolhidos precisavam apresentar algumas características específicas desejadas, além de compatibilidade com o sistema do Arduino como o Esp8266 que foi escolhido por apresenta uma alta velocidade de processamento e de comunicação, ter uma estrutura compacta, *Wifi* embarcado, conta com uma arquitetura de 32 bits a um baixo custo; MCP3208 (Conversor Analógico-Digital) - o Esp8266 apresentou um problema em suas portas analógicas de comunicação, devido a isso, foi escolhido

o MCP para integrar os sensores ao módulo *Wifi*; Sensor de corrente SCT-013-050 (SC) - foi escolhido por apresentar características não invasivas ao sistema pois o mesmo se encaixa ao redor de um condutor já instalado, evitando interrupções do funcionamento do sistema e suportar um valor de corrente compatível com os níveis máximos esperados em picos; Sensor de tensão ZMPT101B, 0 A 250V (ST) - foi escolhido por possuir um isolamento galvânico, isolando as portas de leitura da saída do sensor. Por apresentarem uma arquitetura compacta, facilitam o manuseio e instalação, apresenta significativa precisão para calibração e custo atrativo.

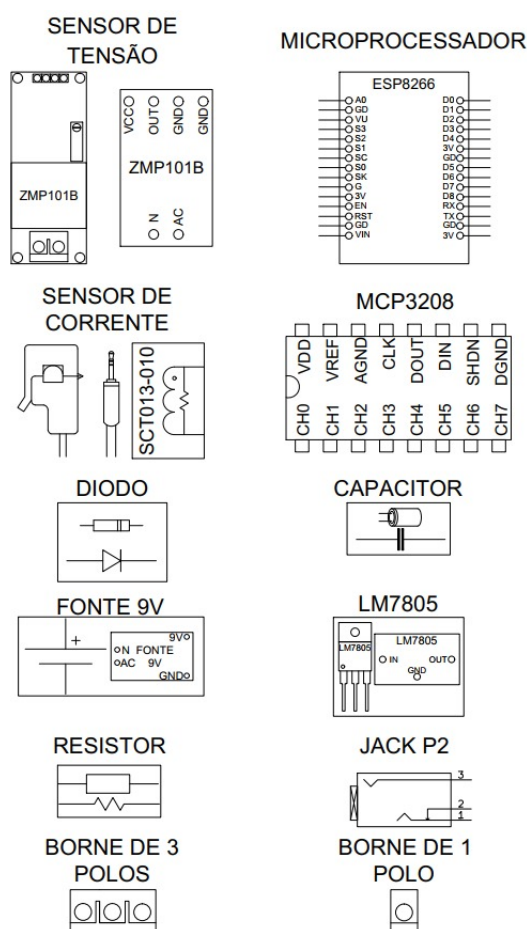


Figura 1 – Componentes Utilizados no Protótipo

4.1.1 Montagem do Protótipo

Para efetuar medição de tensão foram utilizados Bornes (um de 1 e outro de 3), nos quais foram conectados cabos responsáveis por adquirir tensão; a saída do Borne foi conectada ao sensor de tensão utilizando o neutro comum (ZMPT101B). A saída do sensor apresentou uma tensão incompatível com o sensor escolhido (MCP3208), devido a isso, foi observado através de osciloscópio que sem haver leitura de valores na entrada, a saída do osciloscópio era de 2,5V e, ao ligar em sua entrada, a tensão da tomada (no caso 220V), gerou uma senoide variando com uma tensão mínima de 2,2V. A tensão de pico máxima observada em 220V não era capaz de reproduzir um pico de 3V, ou seja, a variação de tensão pico a pico era de apenas 1V. A fim de compatibilizar a leitura de tensão (em 3,3V Máxima) e a saída teórica máxima do sensor é de

5V, foi adicionado 3 diodos em série com cada um dos sensores, cada diodo com uma queda de tensão fixa de 0,7V, ao final resultou em uma saída máxima do sensor de 2,9V (5V [que é padrão do sensor] - 2,1V [Queda de tensão]), ficando dentro do limite de leitura (no caso 3,3V).

Para leitura de corrente foi utilizado o SCT 013 por não ser invasivo, facilitando a montagem e também a segurança no processo de instalação e funcionamento, como esse sensor possui um pino de 3,5mm em sua saída, foram adicionados na placa ilhada 3 conectores compatíveis (Conector jack P2 J2 Estéreo Áudio 3,5mm). Para aquisição da corrente, o sensor se comportou como uma espécie de transformador, assim o fio que passa através do sensor operava como o primário e as saídas (p2) do sensor um secundário, sendo um fio de referência e o outro de sinal, caso a referência fosse conectada no negativo da placa resultaria em uma saída no outro terminal do sensor, porém alternada, chegando no semicírculo negativo; saindo do range de leitura do MCP do protótipo capaz de fazer a leitura apenas da parte positiva. Em busca de uma solução para este problema, foi conectado o terminal de referência a um valor intermediário utilizando um divisor de tensão com 2 resistores de 6,8 k Ω em 3,3V, apresentando um valor de 1,65V. Ligando a referência no divisor, a outra extremidade iria variar a sua leitura dentro espectro do MCP, a fim de atenuar possíveis ruídos. Também foi adicionado um capacitor de 10 μ F em paralelo com um dos resistores a placa, em específico entre o negativo e o referencial.

A ideia inicial era ligar de forma direta os sensores nas portas de leitura analógica do ESP, porém não foi possível, por se tratar de um sinal de 60Hz; para que fosse obtida uma boa reprodução do sinal, seria necessário ler diversos pontos dentro de um mesmo período a fim de uma maior precisão nos resultados, o ADC interno do ESP se mostrou totalmente incapaz de realizar a leitura por se tratar de uma entrada variável e também uma frequência de aquisição muito rápida. Na tentativa de leitura apenas com o ESP foi retornado apenas ruídos, sendo impossível retirar o ruído mesmo com filtros passa baixa e somadores operacionais, visto que esta é uma limitação do ADC interno. Para que esse problema fosse resolvido foi utilizado um ADC externo, o MCP3208, assim os sensores foram incorporados as portas do mesmo; e através do protocolo SPI o ADC teriam comunicação constante com o ESP, enviando os dados de suas portas a ele.

As portas de corrente foram conectadas nas entradas 5, 6 e 7 e as de tensão nas entradas 0, 3 e 4 do MCP, já o MCP foi conectado ao ESP através as portas 12,13,14 e 15. Para alimentar o sistema foi conectada uma fonte de 9V, como a tensão de alimentação dos sensores de tensão era de 5V e o ESP poderia ser alimentado com essa mesma tensão. O LM7805 foi utilizado para: regulagem da tensão, alcance de maior estabilidade e filtro de possíveis ruídos. Visando melhor desempenho do sistema, o circuito foi projetado da seguinte maneira:

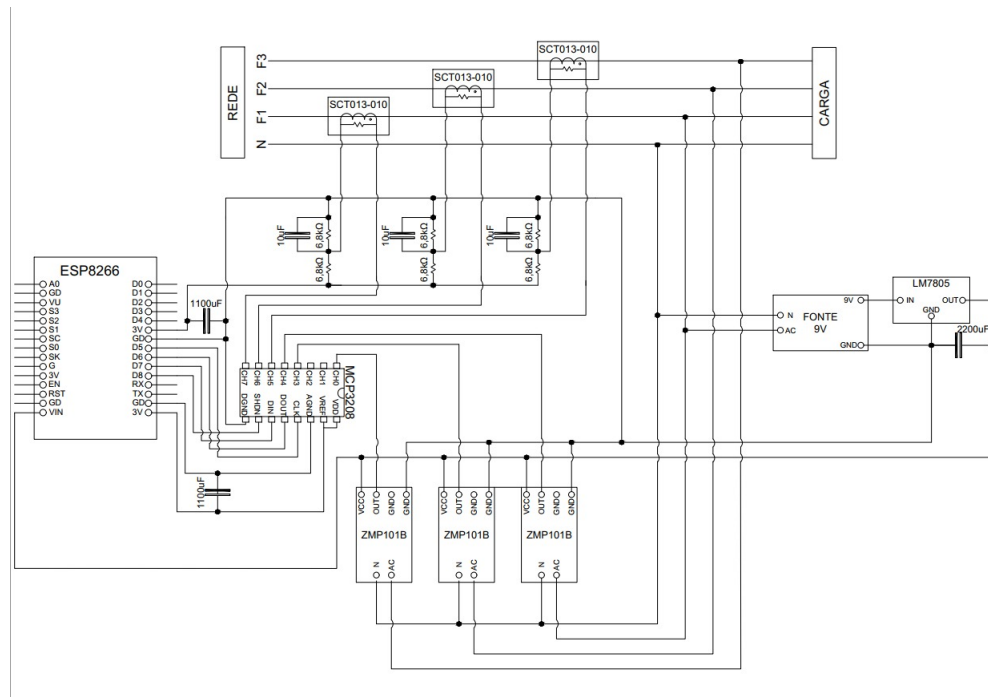


Figura 2 – Montagem do Protótipo

4.1.2 Aferição dos Sensores

Após finalização da montagem do protótipo, foi necessário calibrar a parte de leitura de tensão, para isso foi preciso entender que houve uma queda de tensão por parte do diodo acrescentado ao sistema. Para mitigar essa queda inevitável, foram colocado 3 diodos em série em cada sensor de tensão, cada diodo consumindo 0,7V do sistema e, para que esse erro na leitura fosse corrigido, houve a soma de valores equivalentes à perda na biblioteca. Como feito:

$$\frac{1024}{3,3V} * (0,7V * 3) = 651,6364$$

Esse valor foi multiplicado por uma constante pré-definida pelo sensor na biblioteca, com o seguinte valor: 234.26 . Com o intuito de alcançar uma regulagem mais precisa no sensor, foi utilizado o *trimpot* de cada um deles para permitir calibragem individual. Para ajustar os sensores foi utilizado um multímetro Minipa© ET-2082D com certificação de calibração válida que foi conectado em paralelo o sensor a ser calibrado, na rede da concessionária local, o ajuste foi realizado manualmente rotacionado o *trimpot* até a obtenção dos mesmos valores. O multímetro também foi utilizado para calibrar os sensores de corrente, o foi realizado em banco de resistores que foi conectado em serie ao leitor de corrente com o intuito de medir a corrente de uma quantia de resistores em série.

Como anteriormente, foi ajustado o valor lido pelo sensor de acordo com o multímetro de manualmente e, em seguida, após ajustar todos os sensores envolvidos, foi mudado o arranjo dos resistores para que as leituras fossem realizadas novamente pelos sensores para validar se o valor medido pelo sensor era o mesmo lido pelo multímetro.

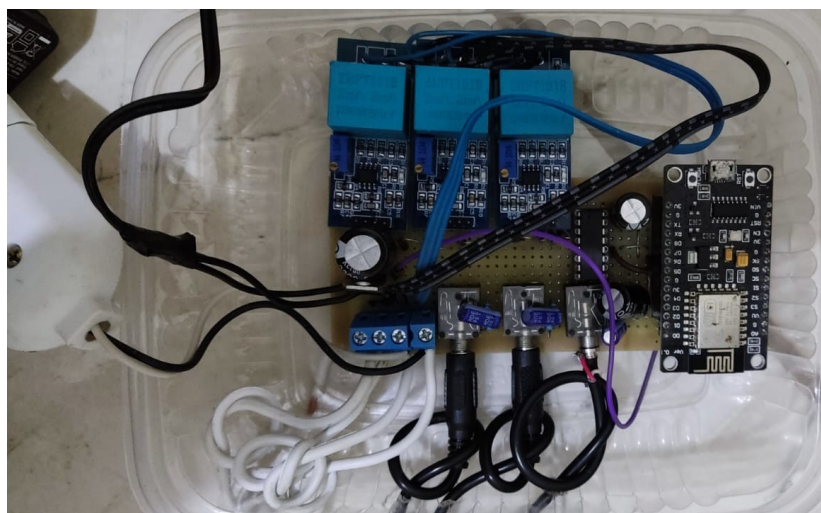


Figura 3 – Protótipo Vista Frontal

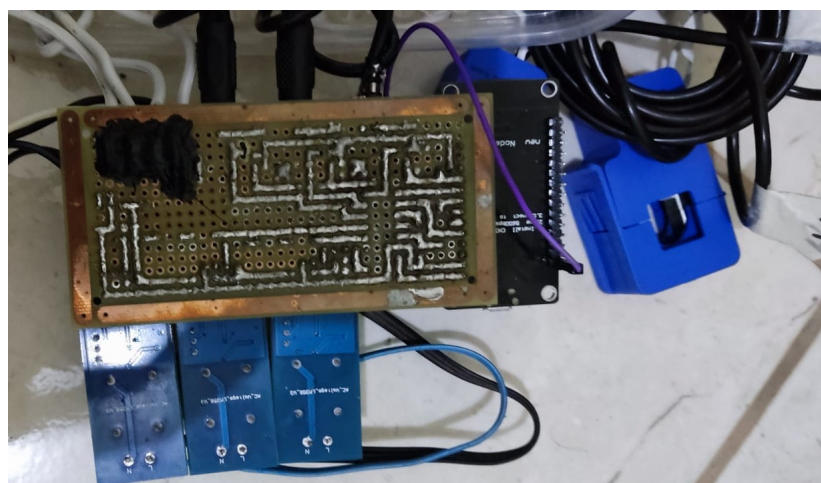


Figura 4 – Protótipo Vista das Soldas

4.1.3 Código em Arduino

Foram usadas 3 principais bibliotecas: a MCP 3008, que realiza a comunicação com o ADC externo, a *EmonLib* que realiza os cálculos de conversão da leitura dos sensores em valores de tensão e corrente, e a *ESP8266HTTPClient* que realiza a comunicação entre o protótipo e o servidor, além dessas foram utilizadas as bibliotecas *ESP8266WiFi* e *WiFiClient* responsáveis pela comunicação *Wifi*.

A biblioteca *EmonLib* foi utilizada para captar os dados analógicos dos sensores e realiza os cálculos para tensão e corrente, que se trata do processo mais delicado, contudo essa mesma biblioteca busca os pinos de leitura analógica do próprio dispositivo no qual ela é instalada, não tendo de forma nativa recursos para um ADC externo, para contornar esse empecilho foi feita a alteração da biblioteca, assim todos os pinos referenciados a leitura serial no final do ADC, outro problema enfrentado nesta mesma biblioteca foi o paralelismo de leituras visto que havia três leituras de tensão e três de corrente. Seria possível abrir várias instâncias da biblioteca e realizar a leitura RMS de uma das fases, depois a outra, mas isso implicaria em um período

grande de ausência de leitura, resultando em dados com erros de leitura significativos, o que acarretou mais uma edição da biblioteca para solucionar o problema. Essa alteração fez com que a leitura fosse feita em sequência os valores instantâneos e, em seguida, realizando os cálculos em RMS uma vez a cada 800ms aproximadamente, ou seja, dentro desse mesmo tempo seria possível armazenar muitas leituras instantâneas de ambas as fases e realizar os cálculos.

4.1.4 Armazenamento de Dados de Leitura

Por se tratar de um protótipo para um projeto acadêmico, no qual não houve intenção inicial de comercialização, alguns dos componentes que foram escolhidos para integrar essa montagem apresentaram algumas limitações que não se apresentaram prejudiciais. Uma limitação significativa encontrada foi a parte de envio das leituras da memória interna do equipamento para o servidor que receberia esses dados, o problema encontrado foi o fato de que a leitura instantânea seria interrompida no momento do envio, ou seja, haveria um período indeterminado de ausência de leituras de consumo.

Em busca de resolver o problema com a leitura dos dados, a saída escolhida foi salvar as leituras instantâneas diretamente em um banco de dados hospedado em um servidor *Web*. Para acompanhar os dados lidos foi criada uma página HTML, conectada ao Banco de Dados, com a finalidade de exibir todas as efetuadas de maneira *online*. Nessa página específica não houve tratamento de dados ou implementação de elementos visuais, uma vez que seu propósito foi somente verificar se a leitura ainda estava em andamento sem precisar abrir o servidor.

4.1.5 Leituras Obtidas com o Protótipo Instalado

Com a finalidade de testar e validar o funcionamento do medidor de consumo, o protótipo foi instalado no quadro geral de distribuição de um imóvel localizado na cidade de Goiânia-GO, com fornecimento de energia trifásico, usina fotovoltaica instalada e com perfil de consumo em dias úteis e horário comercial. Os dados foram adquiridos de maneira contínua no período de 03/11/2022 à 27/11/2022, de maneira que os sensores executavam e registravam as leituras de tensão e corrente, nas três fases do sistema, com a finalidade de calcular o consumo de maneira mais exata possível.

Para que fosse possível capturar o maior número de variações de leitura possíveis, buscando alcançar uma precisão de leitura significativa, os sensores de tensão e corrente executavam leituras nas fases a cada 1 milissegundo do dia, já o envio dos dados para o servidor *online* foi feito a cada 1 a 2 segundos, essa variação era causada de acordo com a conexão da rede de internet no local. Por ter resultado em milhares de leituras por dia, para facilitar a demonstração de dados, foi feita uma compilação dos dados para obter um valor de consumo diário. Os dados obtidos foram:

Data	Dia da Semana	Consumo do Dia
3/11	Quinta-feira	20,43
4/11	Sexta-feira	20,47
5/11	Sábado	5,90
6/11	Domingo-feira	5,20
7/11	Segunda-feira	33,21
8/11	Terça-feira	31,50
9/11	Quarta-feira	25,10
10/11	Quinta-feira	26,95
11/11	Sexta-feira	27,55
12/11	Sábado	4,70
13/11	Domingo	4,25
14/11	Segunda-feira	25,41
15/11	Terça-feira	5,60
16/11	Quarta-feira	29,43
17/11	Quinta-feira	29,83
18/11	Sexta-feira	29,02
19/11	Sábado	4,72
20/11	Domingo	4,76
21/11	Segunda-feira	34,48
22/11	Terça-feira	35,14
23/11	Quarta-feira	29,96
24/11	Quinta-feira	22,13
25/11	Sexta-feira	28,44
26/11	Sábado	4,95
27/11	Domingo	4,46

Tabela 2 – Consumo Diário do Período

Com o intuito de facilitar a visualização do consumo diário registrado os dados, dispostos na tabela anterior, foram organizados em um gráfico de barras. Vale ressaltar que o consumo de energia, no caso em questão, não apresentou oscilações muito significativas pois se tratava de um imóvel comercial com rotina de uso de equipamentos, ou seja, todos eram utilizados diariamente e de maneira contínua em dias úteis. O gráfico resultante ficou da seguinte maneira:

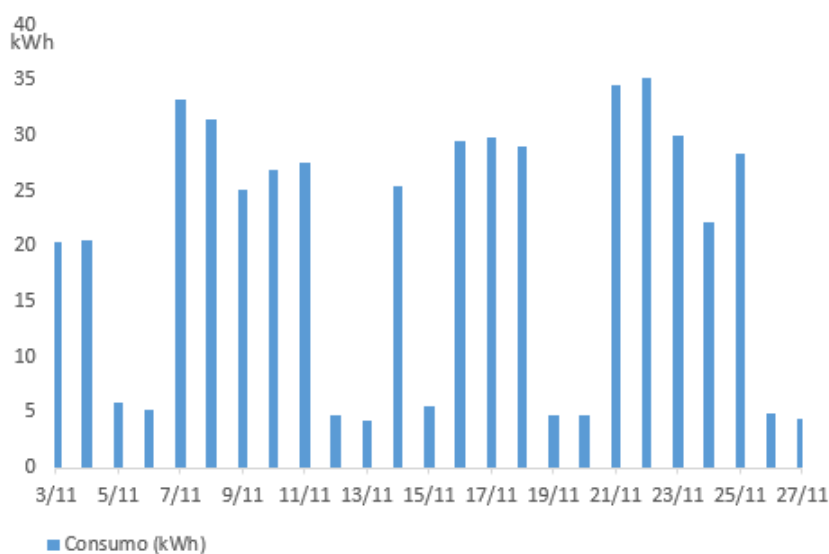


Figura 5 – Consumo Diário Obtido

4.1.6 Validação das Leituras de Consumo

Por se tratar de um protótipo com finalidade acadêmica, constituído por componentes eletrônicos de baixo custo, montagem pouco robusta e não possuir calibração e certificação pelos órgãos competentes para esse modelo de equipamento, foi encontrada uma grande necessidade de validar todas as leituras obtidas de maneira palpável, porém simples. No local em que o protótipo foi instalado já havia outro equipamento medidor de consumo, que já era certificado e veiculado no mercado atual, o qual foi mantido para fornecer dados de comparação para os fins de validação do protótipo.

O medidor de consumo que já se encontrava instalado no local era um modelo Neurio W1-HEM Trifásico da fabricante Generac®, com capacidade de leitura de consumo de energia e geração fotovoltaica, porém, foi instalado no quadro de distribuição de energia interno no qual não estava conectada a parte de geração.

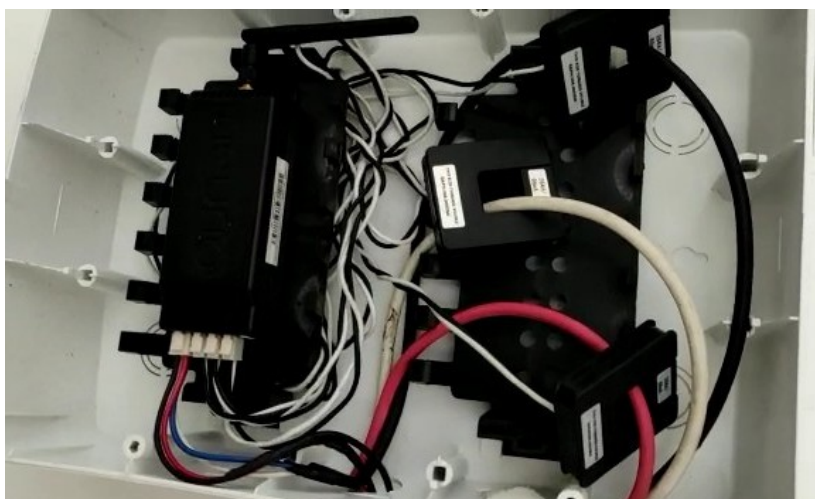


Figura 6 – Medidor de Consumo Neurio W1-HEM Trifásico

Para fins de validação, foram recolhidos os dados de consumo registrados por esse medidor pelo monitoramento *PWRview*, próprio da marca Generac®, que registrou todo o consumo de energia diariamente durante o mesmo período que o medidor protótipo, ambos conectados ao mesmo quadro e mesmas fases. Os dados de consumo registrados pelo medidor Generac resultaram no seguinte gráfico:

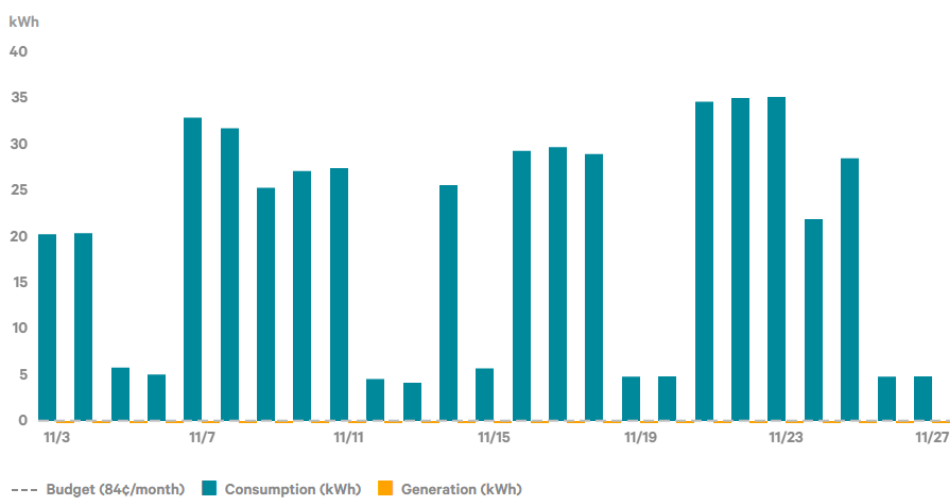


Figura 7 – Consumo Registrado pelo Equipamento Generac

Fazendo a comparação entre os valores de consumo diário registrados, pelo medidor protótipo e o medidor Generac, foi possível constatar a existência de erros na leitura do protótipo durante o período de teste.

Data	Protótipo	Neurio	Erro
3/11	20,43030843	20,25	0,18
4/11	20,47398046	20,38	0,09
5/11	5,905646888	5,78	0,13
6/11	5,200282894	5,04	0,16
7/11	33,21387389	32,91	0,30
8/11	31,50336378	31,76	0,26
9/11	25,10131957	25,31	0,21
10/11	26,95868436	27,12	0,16
11/11	27,55228406	27,44	0,11
12/11	4,703028023	4,55	0,15
13/11	4,250211712	4,14	0,11
14/11	25,41181615	25,60	0,19
15/11	5,607232505	5,71	0,10
16/11	29,43336626	29,31	0,12
17/11	29,83819716	29,71	0,13
18/11	29,02716135	28,95	0,08
19/11	4,724865529	4,82	0,10
20/11	4,765259515	4,84	0,07
21/11	34,48609239	34,64	0,15
22/11	35,14765974	35,04	0,11
23/11	29,96263895	35,17	5,21
24/11	22,13183154	21,90	0,23
25/11	28,4485979	28,51	0,06
26/11	4,95445383	4,80	0,15
27/11	4,467245203	4,85	0,38

Tabela 3 – Comparação dos Valores de Consumo Registrados

Com o intuito de mensurar o erro, de maneira visual, foi criado um gráfico representando a comparação entre o consumo lido por ambos os medidores de consumo. O erro encontrado em cada leitura, de acordo com os dados, se manteve inferior a 0,5kWh de maneira bem constante, exceto no dia 23 em que o erro foi de 5,21kWh.

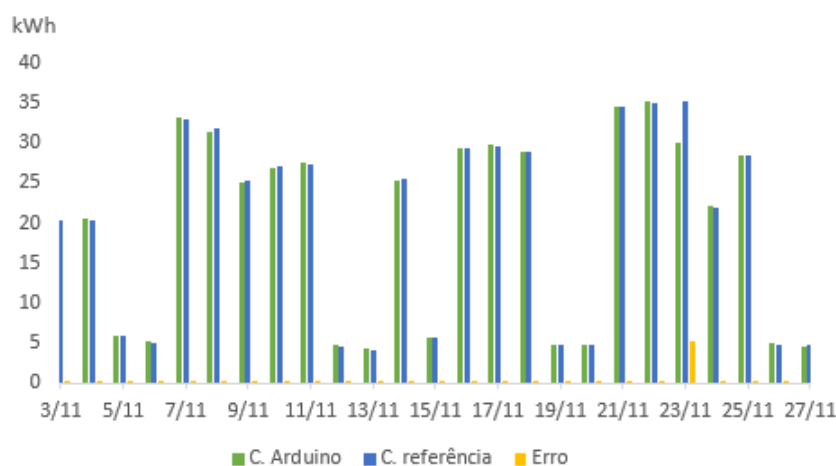


Figura 8 – Valores de Consumo e Erro

Para ser possível estabelecer uma validação mais precisa entre os equipamentos, o medidor de consumo Neurio© e o protótipo, foi feita uma análise comparativa de uma leitura a cada hora em um período de 24 horas referente ao dia 04/11/2022. O resultado obtido foi:

Hora	Protótipo	Neurio	Erro
12	0,35	0,35	0
1	0,35	0,33	0,02
2	0,35	0,33	0,02
3	0,36	0,33	0,03
4	0,36	0,33	0,03
5	0,36	0,37	0,01
6	0,35	0,27	0,08
7	0,36	0,25	0,11
8	1,54	1,46	0,08
9	1,54	1,63	0,09
10	1,54	1,62	0,08
11	1,54	1,66	0,12
12	1,55	1,79	0,24
13	1,55	2,08	0,53
14	1,54	1,99	0,45
15	1,55	1,98	0,43
16	1,54	1,82	0,28
17	1,54	0,16	1,38
18	0,45	0,21	0,24
19	0,35	0,43	0,08
20	0,35	0,24	0,11
21	0,35	0,24	0,11
22	0,35	0,28	0,07
23	0,35	0,23	0,12

Tabela 4 – Comparação das Leituras em 24 horas

Ao final da comparação entre as leituras foi possível afirmar que, apesar de ter sido calibrado manualmente através do método de tentativa e erro, o protótipo se mostrou mais eficiente e assertivo que o esperado.

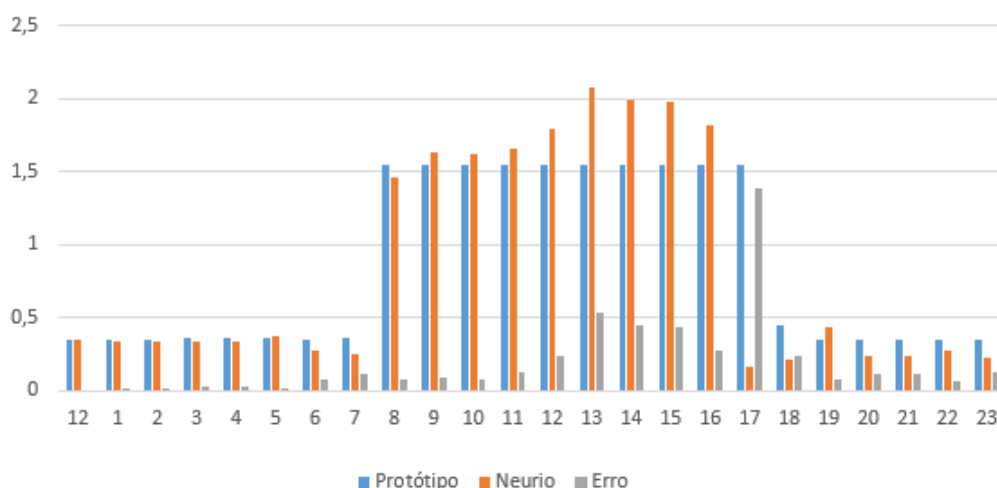


Figura 9 – Erro de Leitura Referente a 1 Dia

4.2 Software

Assim como proposto na metodologia, foi desenvolvido um API com a linguagem de programação Python capaz de recolher os dados de consumo de energia do imóvel bem como a geração fotovoltaica atual e histórico de geração da unidade consumidora em questão. Vale ressaltar que os testes do API foram realizados em uma unidade consumidora diferente do teste do protótipo de monitoramento de consumo, uma vez que o local de instalação do protótipo não possuía histórico de consumo devido a recente de troca de titularidade, já para a validação do funcionamento do *software* era desejável um histórico de consumo de energia e geração fotovoltaica superior a quatro meses.

4.2.1 Teste do *Script API*

A unidade consumidora que serviu de base para a validação do API possuía um histórico de consumo de energia elevado, com fornecimento monofásico e geração fotovoltaica com potência projetada de 17,085kWp . Com a aquisição de dados por meio de *scraping*, feito no site da concessionária de energia Enel-GO e no monitoramento de geração fotovoltaica *Solarview*®, baseado na lógica de seletores do *Playwright*, foi possível testar todos os aspectos relevantes para um gerenciamento de consumo completo. Após a execução do *scrit*, os dados foram recolhidos e tratados assim como descrito no capítulo de metodologia e, após serem atribuídos a variáveis e estruturado em uma tabela *dataframe*, as informações se tornaram adequados para envio ao banco de dados de um servidor *web*. Após conexão com o servidor, os dados foram alocados através do *phpMyAdmin* da seguinte maneira:

	id_cliente	Mes Referencia	Geracao Mensal	Valor Fatura	Leitura	Energia Inj.	Energia Consum.	Credito Compens.	Tarifa	Tarifa com Imposto	Medidor	Autoconsumo	Saldo de Credito	Economia
<input type="checkbox"/>	1	Abr/2022	2158.8	1252.23	27/04/2022	1164	2325	1164	0.93796000	0.63689	58596	994.8	0	633.58
<input type="checkbox"/>	5	Ago/2022	2594.6	93.19	25/08/2022	1791	1383	1283	0.80661000	0.63689	64096	803.6	0	835.34
<input type="checkbox"/>	4	Jul/2022	2843.5	91.53	27/07/2022	1867	1084	984	0.79652000	0.63689	62713	976.5	0	1184.3
<input type="checkbox"/>	3	Jun/2022	1799.2	396.53	25/06/2022	1023	1425	1023	0.95326000	0.63689	61629	776.2	0	494.35
<input type="checkbox"/>	2	Mai/2022	1964	596.67	26/05/2022	987	1608	987	0.93414000	0.63689	60204	977	0	622.24
<input type="checkbox"/>	8	Nov/2022	3341.9	97.66	25/11/2022	2355	1469	1369	0.85452	0.67099	68670	986.9	0	843.33
<input type="checkbox"/>	7	Out/2022	3528.9	90.46	27/10/2022	2759	1461	1361	0.8043	0.6434	67201	769.9	0	1094.65
<input type="checkbox"/>	6	Set/2022	2915.9	95.91	26/09/2022	2019	1644	1544	0.80717000	0.63689	65740	896.9	0	873.75

Figura 10 – Banco de Dados SQL

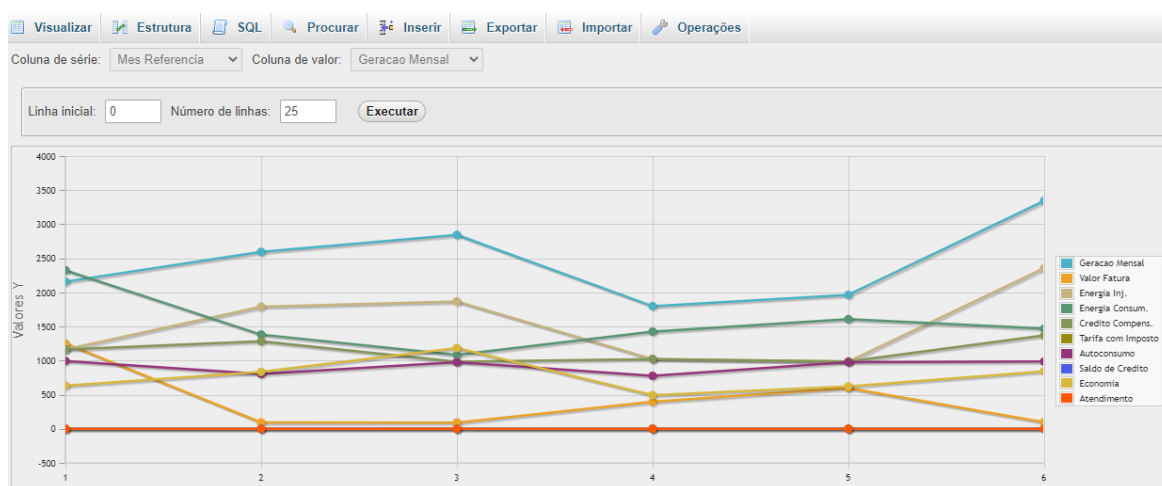


Figura 11 – Gráfico dos Dados SQL

Para gerar o relatório do usuário foi criado um *template*, no Canva®, para ser utilizado em formato PDF para compor toda a parte visual estática padrão a ser utilizado para todos os relatórios gerados posteriormente. Na parte superior do relatório foram adicionados elementos como a logo do projeto, uma ilustração simples representando os dados de geração da usina e sua distribuição, também foram incluídos dados como consumo e autoconsumo.

Já para a parte dinâmica, que inclui os dados específicos de cada usuário, a opção foi utilizar a biblioteca *Reportlab* para importar os dados armazenados no Banco de Dados em forma de variável e incorporar as mesmas em forma de texto, além de ajustar o posicionamento de cada informação de acordo com o *layout* padrão.

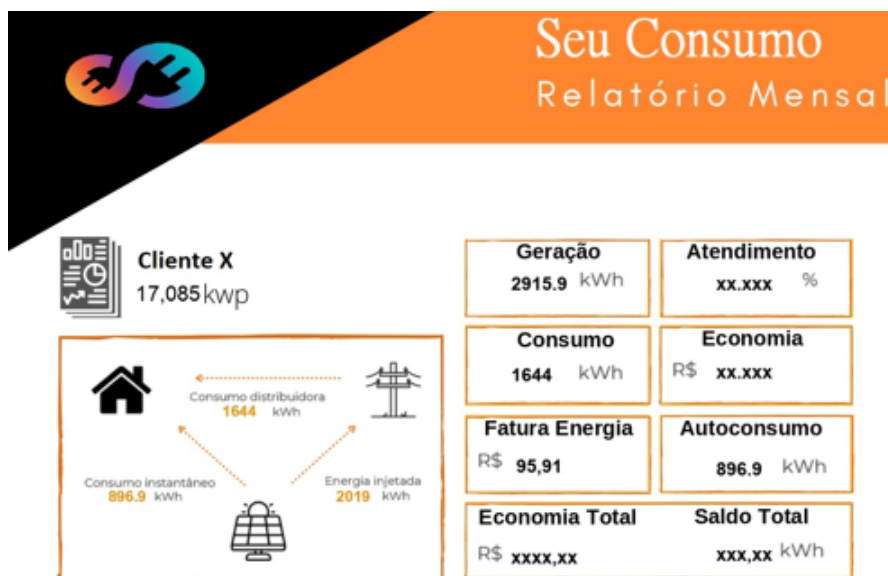


Figura 12 – Parte Superior do Relatório Mensal

A criação dos gráficos com a biblioteca utilizada seguiu um padrão de organização sendo o primeiro conjunto mensal do gráfico o maior valor de geração e o último conjunto o de menor valor de geração. Para complementar a parte inferior do relatório, também foi adicionada uma tabela com o histórico de energia consumida e valores de energia injetada e créditos energéticos do mês em questão e os onze meses anteriores.

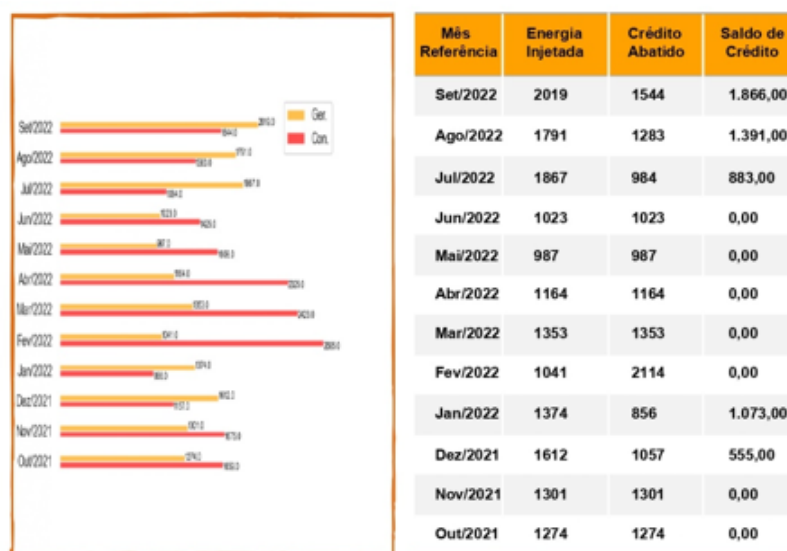


Figura 13 – Parte Inferior do Relatório Mensal

4.2.2 Plataforma Online

A página inicial idealizada e escrita da maneira mais agradável, intuitiva e de fácil entendimento possível para guiar o usuário até o local de acesso ou ação desejada. Por se tratar de um projeto acadêmico, toda a parte estética da página foi estruturada com imagens de domínio

público, além de não apresentar excesso de informações ou complexidade estrutural, ficando da seguinte maneira:

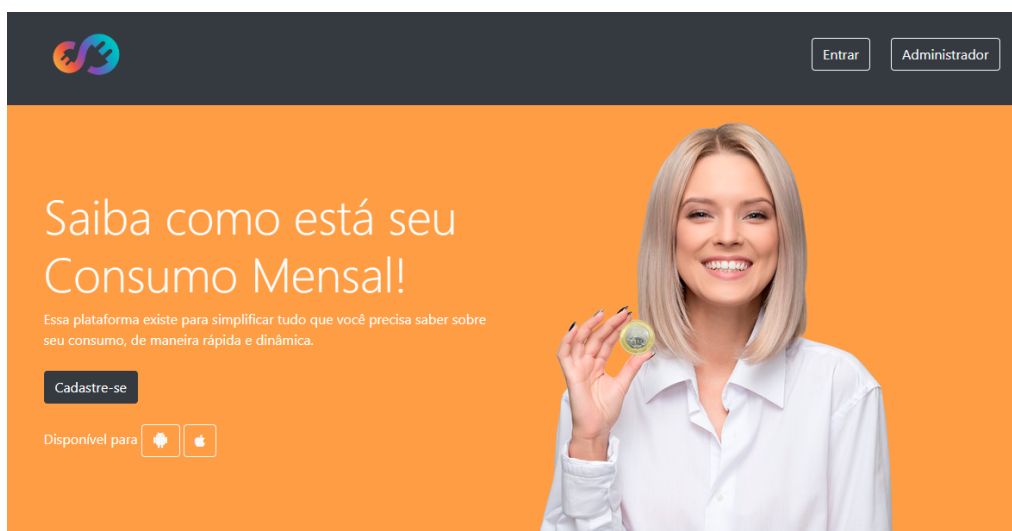


Figura 14 – Página Inicial do Site

Por se tratar de um sistema de gerenciamento de consumo no qual o usuário não participou da construção e somente aderiu ao método, é possível afirmar que seu registro já esteja previamente cadastrados no sistema, diretamente no banco de dados. Esta página foi criada caso o usuário precise de acesso aos seus dados por múltiplos logins, sendo assim, essa página serve para gerar o novo registro.

Figura 15 – Página de Registro de Usuário

A página de login seguiu a mesma identidade visual e estrutural da página de registro, alterando apenas os links de redirecionamento, dados solicitados e nível de autenticação de usuário. Para essa plataforma, foram criadas duas páginas de login, sendo elas: *login* convencional, qualquer usuário que possuir registro será autenticado e enviado para sua página particular; o *login* de administrador do sistema, no qual somente um *login* e senha específicos pode acessar páginas de gerenciamento do sistema.

Figura 16 – Página de Login Convencional

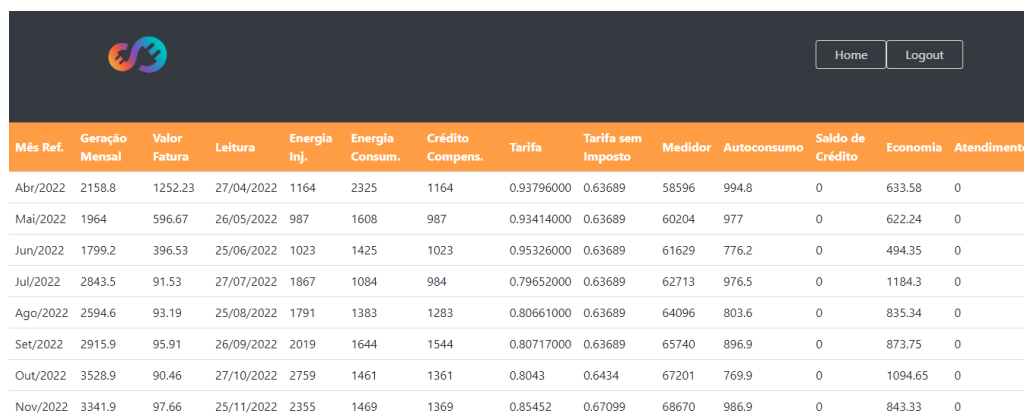
Figura 17 – Página de Login Administrador

Quando se deseja a implantação e adesão de um sistema autônomo, no qual o objetivo principal é oferecer ao indivíduo maior liberdade e acessibilidade de maneira ampla e sem auxílio de terceiros, uma parte muito importante é disponibilizar as ferramentas necessárias, sendo assim, foi primordial criar um botão de "logout" no canto superior da página particular do usuário, para que o mesmo fosse capaz de encerrar a sessão a qualquer momento com apenas um clique.

Cada login de usuário foi vinculado a uma página privada, que criada para comportar todos os dados coletados sobre o consumo de energia, a geração de sua usina e créditos da concessionária de energia (Enel-GO) e demais dados de leitura. Para que o encaminhamento de cada usuário para sua página privada fosse feito de maneira correta, foi criado um trecho de código em PHP para sincronizar os dados de usuário com sua página privada e fazendo esse redirecionamento.

Ao entrar na própria página pessoal, o usuário tem todos os seus dados coletados (geração, consumo e créditos) dispostos em uma tabela bem intuitiva e de fácil compreensão.

Como auxílio visual, foi adicionado um gráfico de barras do tipo "Morris" com um conjunto de três barras referentes a cada mês registrado, sendo elas: geração na cor verde, consumo na cor vermelha e autoconsumo (valor do consumo subtraído do valor de geração) em azul.



Mês Ref.	Geração Mensal	Valor Fatura	Leitura	Energia Inj.	Energia Consum.	Crédito Compens.	Tarifa	Tarifa sem Imposto	Medidor	Autoconsumo	Saldo de Crédito	Economia	Atendimento
Abr/2022	2158.8	1252.23	27/04/2022	1164	2325	1164	0.93796000	0.63689	58596	994.8	0	633.58	0
Mai/2022	1964	596.67	26/05/2022	987	1608	987	0.93414000	0.63689	60204	977	0	622.24	0
Jun/2022	1799.2	396.53	25/06/2022	1023	1425	1023	0.95326000	0.63689	61629	776.2	0	494.35	0
Jul/2022	2843.5	91.53	27/07/2022	1867	1084	984	0.79652000	0.63689	62713	976.5	0	1184.3	0
Ago/2022	2594.6	93.19	25/08/2022	1791	1383	1283	0.80661000	0.63689	64096	803.6	0	835.34	0
Set/2022	2915.9	95.91	26/09/2022	2019	1644	1544	0.80717000	0.63689	65740	896.9	0	873.75	0
Out/2022	3528.9	90.46	27/10/2022	2759	1461	1361	0.8043	0.6434	67201	769.9	0	1094.65	0
Nov/2022	3341.9	97.66	25/11/2022	2355	1469	1369	0.85452	0.67099	68670	986.9	0	843.33	0

Figura 18 – Parte Superior da Página Privada de um Usuário

Gráfico de Geração, Consumo e Autoconsumo

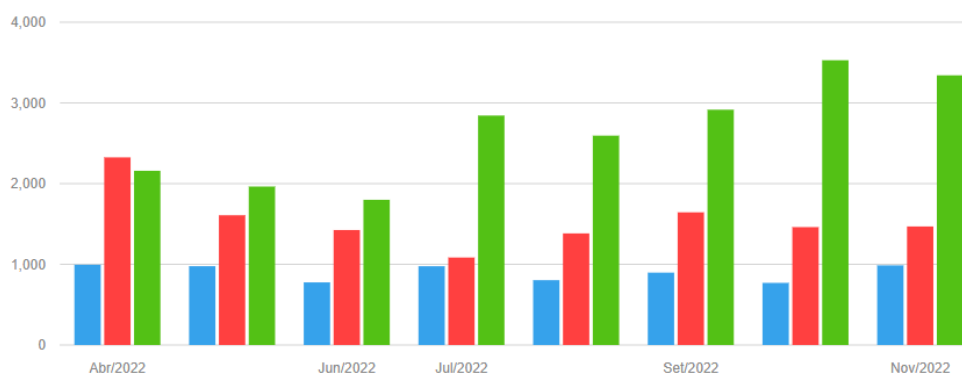
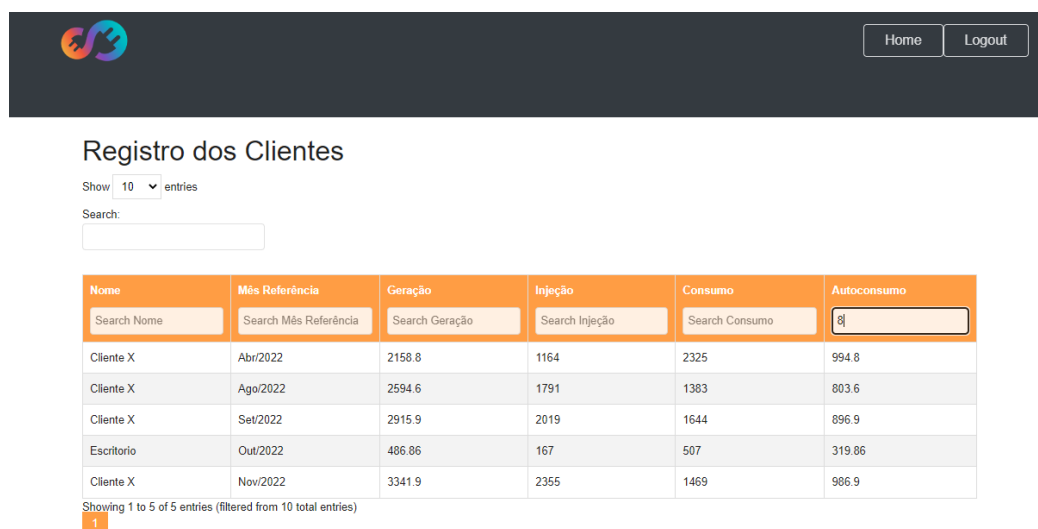


Figura 19 – Parte Inferior da Página Privada de um Usuário

Se tratando de um sistema de gerenciamento, era imprescindível que o administrador do sistema fosse capaz acessar dentro do Banco de Dados todos os registros referentes a um cliente, de um mês específico de vários clientes ou qualquer outra situação semelhante.

Para isso, o método adotado foi a construção de uma tabela em uma página com PHP e HTML combinados assim como nos *scripts* anteriores, respeitando a configuração de estilo citada anteriormente e método de autenticação PHP. Por ser tratar de uma página que pode comportar um volume de informações grande, formado por diversos clientes e datas referência, a estrutura da tabela foi criada com campos de pesquisa em seu topo para realizar filtragem de dados, podendo ser utilizados todos eles simultaneamente.



Registro dos Clientes

Show 10 entries

Search:

Nome	Mês Referência	Geração	Injeção	Consumo	Autoconsumo
Search Nome	Search Mês Referência	Search Geração	Search Injeção	Search Consumo	8
Cliente X	Abr/2022	2158.8	1164	2325	994.8
Cliente X	Ago/2022	2594.6	1791	1383	803.6
Cliente X	Set/2022	2915.9	2019	1644	896.9
Escritorio	Out/2022	486.86	167	507	319.86
Cliente X	Nov/2022	3341.9	2355	1469	906.9

Showing 1 to 5 of 5 entries (filtered from 10 total entries)

Figura 20 – Página de Administrador

5 CONCLUSÕES

No contexto deste trabalho foi proposta e desenvolvida uma ferramenta de gestão de sistemas de energia, com ênfase total no controle de consumo de energia, contemplando também aspectos da fatura emitida pela concessionária de energia e a geração de uma usina fotovoltaica instalada. A meta era criar uma ferramenta completa para efetuar todo o trâmite necessário da maneira mais automática possível, apresentando suas duas vertentes: *hardware*, que seria o protótipo do dispositivo de leitura de consumo; *software*, que representaria toda a parte de aquisição, manipulação e exibição de todo o sistema de maneira fácil e acessível.

O software se comportou parcialmente como o esperado, mas, após adaptações e ajustes de programação por meio de bibliotecas alternativas do *Python*, realizou corretamente a varredura e manipulação de dados, gerou o relatório do cliente e todos os envios de dados foram feitos para o banco de dados da maneira desejada. A plataforma *online* correspondeu todas as expectativas de funcionalidade e estética, mesmo se tratando de um projeto acadêmico, apresentou um *layout* responsivo como esperado, páginas amigáveis com fácil compreensão e um design limpo e agradável. É importante ressaltar que, por ter sido feito teste somente com duas unidades consumidoras diferentes, a programação de segurança e direcionamento do usuário para página privada não foi realizada em larga escala e sim somente para os casos de teste.

Sobre o *hardware*, houve algumas limitações técnicas por parte dos componentes utilizados na montagem, bem como a necessidade de algumas adaptações físicas e de programação para operar da maneira mais eficiente possível. Para utilização em Grupo B1 (residencial) de consumo de energia, o protótipo se mostrou aplicável e correspondeu ao esperado, apresentou fácil instalação, manuseio e principalmente facilidade no acompanhamento dos dados lidos em tempo real. Felizmente todas as limitações técnicas encontradas só seriam um empecilho em caso de comercialização, sendo assim, seria necessário apresentar uma melhor velocidade de processamento de dados e memória mais ampla para guardar os registros diários e enviar as informações de consumo total somente uma vez ao dia. Em um sistema com vários usuários, a redução do volume de dados de leitura diário de milhares de registros para apenas um faria com que fosse possível realizar o processamento de dados com maior velocidade no banco de dados, também seria mais fácil para o usuário acompanhar seu próprio consumo em tempo real com um volume menor de informação. Outro ponto importante a ser levantado é que o protótipo, ao ser comparado com um medidor de funcionalidade semelhante disponível no mercado, registrou leituras com erro inferior a 0,5kWh diário mesmo após ter sido calibrado manualmente e não possuir certificação dos órgãos competentes.

Após mapear todos os pontos negativos e positivos da execução, é possível afirmar que todas as expectativas quanto ao sistema foram atendidas. O comportamento do sistema pôde ser observado em duas situações: *hardware* e *software* atuando separadamente e, posteriormente,

todo o sistema operante de maneira integral para realizar o processo por completo. Sendo assim, é possível afirmar que o projeto desenvolvido atingiu os objetivos propostos inicialmente neste trabalho.

6 Considerações Finais

No que se refere a usinas fotovoltaicas, conforme o tipo de configuração geração e carga, há maneiras diferentes de lidar com a compensação de créditos, vale ressaltar que até o momento do projeto em questão a relação de créditos em vigor era de recebimento de 1kW de crédito energético a cada 1kW injetado na rede da concessionária de energia. Com a Lei 14300/2022, que entrará em vigor em 6 de janeiro de 2023, será preciso que haja uma injeção maior na rede para alcançar o abatimento de créditos equivalente ao da relação atual, afetando muito os consumidores de maneira geral, mas principalmente o modelo de autoconsumo remoto, pois toda sua compensação de créditos no consumo e saldo vem inteiramente da energia injetada na concessionária. Para o modelo de autoconsumo remotos, principalmente após a Lei 14300/2022 entrar em vigor, utilizar métodos que possa monitorar seu consumo pode ser uma forma de auxiliar esse consumidor a estabelecer um perfil de consumo estável e econômico, principalmente em casos que se tem mais de uma unidade consumidora, fazendo uso do sistema de rateio de crédito entre elas.

Já concessionária apresenta ao cliente o consumo de forma mensal, bem como valores de geração e créditos, nesse caso o consumidor não consegue mensurar seu gasto até receber a próxima fatura de energia. Ter um meio de monitoramento de consumo em tempo real poderia auxiliar na projeção de valores a serem pagos na próxima fatura bem como possibilitar uma redução no consumo caso seja percebido um gasto maior que o esperado. Já a fatura emitida pela concessionária poderia ser utilizada para outros fins, como coletar os dados importantes em relação ao sistema de energia fotovoltaica e a “convencional”, manipulá-los e armazená-los em um local de fácil consulta, possibilitando que esse consumidor tenha uma visão geral e total autonomia para gerir seu sistema de energia da maneira que o atender melhor. Desta forma o cliente poderia se sentir mais seguro e confiante na efetividade de seu sistema *on grid* pois poderia monitorar por completo geração total, consumo e até mesmo compreender seu autoconsumo com o auxílio do dispositivo de leitura de consumo instalado em sua residência.

Sobre o protótipo, por se tratar de um sinal de 60Hz, foi encontrada uma falha de projeto em que os sensores não possibilitavam conectividade direta com as portas de leitura analógica do módulo *Wifi*. Para alcançar uma boa reprodução do sinal, foi necessário realizar leituras de diversos pontos dentro de um mesmo período com o intuito de possibilitar maior precisão nos resultados. O conversor analógico-digital (ADC) interno do módulo *Wifi* apresentou incapacidade de executar a leitura de uma entrada variável e frequência de aquisição muito rápida, por esse motivo foi adicionado um ADC externo para incorporar os sensores as portas do mesmo. Visando um projeto futuro, a opção mais coerente para reparar os erros encontrados durante a construção e testes do protótipo medidor de consumo é a substituição do sistema embarcado utilizado por uma versão mais profissional e mercadológica comumente encontrada em dispositivos

semelhantes disponíveis no mercado, buscar sensores com características físicas e configurações mais assertivas bem como desenvolver um datasheet para o equipamento. Uma pretensão, caso o presente projeto tenha uma continuidade será ajustar e, se for coerente e possível, certificar e lançar o dispositivo no mercado

REFERÊNCIAS

- ANEEL, A. N. de E. E. RESOLUÇÃO NORMATIVA Nº 482, DE 17 DE ABRIL DE 2012. <http://www2.aneel.gov.br/cedoc/ren2012482.pdf>, 2012. Citado na página 15.
- ANEEL, A. N. de E. E. RESOLUÇÃO NORMATIVA Nº 687, DE 24 DE NOVEMBRO DE 2015. <https://www2.aneel.gov.br/cedoc/ren2015687.pdf>, 2015. Citado 2 vezes nas páginas 12 e 15.
- ANEEL, A. N. de E. E. Módulo 11 – Fatura de Energia Elétrica e Informações Suplementares. Procedimentos de distribuição de energia elétrica no sistema elétrico nacional – prodist. <http://www.cerpro.com.br/publico/arquivos/prodist/Modulo11.pdf>, 2017. Citado na página 13.
- BERNERS-LEE, T.; CONNOLLY, D. **Hypertext markup language-2.0**. [S.l.], 1995. Citado na página 19.
- CADAVAL, E. R. *et al.* **Grid-Connected Photovoltaic Generation Plants: Components and Operation**. Volume 7. <https://ieeexplore.ieee.org/abstract/document/6603396/authorsauthors>, 2013. Citado na página 16.
- CONTRIBUTORS, M. **What is a web server?** <https://developer.mozilla.org>, 2022. Citado na página 20.
- COSTA, A. P. d.; SERMANN, F. C.; SILVA, G. G. d. **Desenvolvimento de um protótipo para medição de energia elétrica**. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2016. Citado na página 17.
- DUCKETT, J. **PHP & MySQL: Server-Side Web Development**. 1ª edição. ed. [S.l.], 2022. Citado na página 19.
- ELMASRI, R.; NAVATHE, S. B. **Sistemas de Banco de Dados**. 6ª edição. ed. [S.l.], 2010. Citado na página 18.
- FARIAS, M. T. de; ANGELUCI, A. C. B.; PASSARELLI, B. Web scraping and data science in applied research in communication: a study on online reviews. **Revista Observatório**, v. 7, n. 3, p. a1en–a1en, 2021. Citado na página 18.
- IBM, I. B. M. C. **Application Programming Interface (API)**. <https://www.ibm.com/cloud/learn/api>, 2020. Citado na página 18.
- JÚNIOR, J. M. C.; ÓPTICA, G. de; NUMÉRICA-GOMNI, M. Um medidor de energia elétrica integrado em redes de comunicações. **Monografia (Dissertação de Mestrado)—Universidade Estadual de Campinas, Limeira**, v. 19, 2014. Citado na página 16.
- LIE, H. W.; BOS, B. **Cascading style sheets: Designing for the web, Portable Documents**. [S.l.]: Addison-Wesley Professional, 2005. Citado na página 19.
- LIMA, A. *et al.* Otimização de sistemas fotovoltaicos de microgeração distribuída. 2022. Citado na página 12.
- MENEZES, W. R. d. *et al.* Medidores inteligentes e comunicação de dados em redes inteligentes. Universidade Federal de Campina Grande, 2020. Citado na página 17.

MILAGAIA, M. R. **Melhoria da eficiência energética através de mecanismos de feedback: caso do consumo de energia elétrica nas habitações**. Tese (Doutorado), 2015. Citado na página 12.

MIRANDA, M. M. d. **Fator de emissão de gases de efeito estufa da geração de energia elétrica no Brasil: implicações da aplicação da Avaliação do Ciclo de Vida**. Tese (Doutorado) — Universidade de São Paulo, 2012. Citado na página 15.

NASCIMENTO, J. P. *et al.* Um algoritmo de proteção adaptativa para sistemas de distribuição com inserção de geração distribuída. Universidade Federal de Campina Grande, 2014. Citado na página 15.

OLIVEIRA, A. M. de; MARIO, M. C.; PACHECO, M. T. T. Fontes renováveis de energia elétrica: evolução da oferta de energia fotovoltaica no brasil até 2050. **Brazilian Applied Science Review**, v. 5, n. 1, p. 257–272, 2021. Citado na página 12.

QUEIRÓS, R. **Criação Rápida de Sites Responsivos com o Bootstrap**. 1ª edição. ed. [S.l.], 2017. Citado na página 20.

RIPE, A. **How Exactly Does Website Hosting Work?** <https://www.ripemedia.com/how-exactly-does-website-hosting-work/>, 2021. Citado na página 20.

SCHAVEMAKER, P.; SLUIS, L. V. der. **Electrical power system essentials**. 2ª edição. ed. [S.l.], 2017. Citado na página 15.

Apêndices

APÊNDICE A – Web Scraeping: SolarView, Concessionária de Energia (Enel - GO), Conexão com o Banco de Dados e Emissão de Relatório do Cliente

```
1 #Testando array de intervalo de datas
2
3 import datetime
4 import time
5
6 today = datetime.date.today()
7
8 from datetime import date, timedelta
9
10 #Para mes atual
11 #data_inicio = date(today.year, today.month-1, 25)
12 #data_fim = date(today.year, today.month, 26)
13
14 delta = data_fim - data_inicio
15
16 lista_datas = []
17 for i in range(delta.days + 1):
18     day = data_inicio + timedelta(days=i)
19     lista_datas.append(day)
20
21 from datetime import datetime
22
23 lista_datas_str = [datetime.strftime(dt, format="%Y-%m-%d") for dt in
24     lista_datas]
25
26 #print(lista_datas_str)
27
28 #Define os dados de login
29 parametros=["email usuario","senha usuario", "email","senha", "enel", "
30     usuario"]
31
32 Usuario = parametros[0]
33 Senha = parametros[1]
34 Usuarioenel = parametros[2]
35 Senhaenel = parametros[3]
36 search = parametros[4]
37 cliente = parametros[5]
38
39 #Chama o playwright
40 with sync_playwright() as p:
```

```
40     #Abre o navegador , faz o login na Enel e passa por eventuais
        notifica es
41     browser=p.chromium.launch(headless=False) #False abre o navegador ,
        True esconde
42     page=browser.new_page()
43     page.goto('https://www.enel.com.br/pt/login.html')
44     page.locator('text=Aceitar tudo').click()
45     page.locator('#localizatorbuttons > div > a:nth-child(3)').click()
46     page.wait_for_timeout (2500)
47     #page.locator('#main > section > div > form > div > div.help-
        tabs__nav.login-tabs > ul > li:nth-
        child(2) > a
        ').click()
48     #page.wait_for_timeout (1000)
49     #page.locator('input[type=text]:nth-child(3)').click()
50     page.fill("#username", Usuario)
51     page.fill("#privatepwd", Senha)
52     page.wait_for_timeout (1000)
53     page.locator('#personas > div.login-btn > button').click()
54     page.wait_for_timeout (1000)
55     page.locator('text=LEMBRAR MAIS TARDE').click()
56     page.wait_for_timeout (3000)
57     page.locator('#popupRegistrationLgpdButtonNo ').click()
58     page.wait_for_timeout (3000)
59     page.locator('#popupLgpdButtonInformationNo ').click()
60     page.locator('#main > div.pvtArea-wrapper > div.pvtArea-container >
        div.pvtArea-menu.pvtArea-column > nav > ul > li:nth-child(2) > a
        ').click()
61     page.wait_for_timeout (5000)
62 #Entrar na parte de segunda via da fatura
63     page.locator('xpath=//*[@id="action-item-button-second-billing-way
        "]').click()
64     valor_fatura = page.locator('#tableSegundaViaDeFactura > tbody > tr:
        nth-child(1) > td:nth-child(7)')
65     vfatura = valor_fatura.evaluate('el => el.outerText')
66     v_fatura= vfatura[2:]
67
68     print("Valor da Fatura:",vfatura)
69 #Entra na area de contas, resgata os dados de minigeracao e microgeracao
70     page.locator('xpath=//*[@id="action-item-button-energy-
        microgeneration_action_item_1062603347"]').click()
71     page.wait_for_timeout (5000)
72
73     mes12 = page.locator('#table-energy-
        microgeneration_action_item_1062603347 > tbody > tr:nth-child(1)
        > td:nth-child(1)')
74     m12= mes12.evaluate('el => el.outerText')
```

```
75
76     leitura12 = page.locator('#table-energy-
           microgeneration_action_item_1062603347 > tbody > tr:nth-
               child(1) > td:nth-child(2)')
77     l12= leitura12.evaluate('el => el.outerText')
78
79
80     consumo12 = page.locator('#table-energy-
           microgeneration_action_item_1062603347 > tbody > tr:nth-
               child(1) > td:nth-child(3)')
81     con12= consumo12.evaluate('el => el.outerText')
82
83     injetada12 = page.locator('#table-energy-
           microgeneration_action_item_1062603347 > tbody > tr:nth-
               child(1) > td:nth-child(4)')
84     inj12= injetada12.evaluate('el => el.outerText')
85
86     cred_recebido12 = page.locator('#table-energy-
           microgeneration_action_item_1062603347 > tbody >
               tr:nth-child(1) > td:nth-child(5)')
87     cred12= cred_recebido12.evaluate('el => el.outerText')
88
89     tarifa_sem_imposto12 = page.locator('#table-energy-
           microgeneration_action_item_1062603347 > tbody >
               tr:nth-child(1) > td:nth-child(8)')
90     tsi12= tarifa_sem_imposto12.evaluate('el => el.outerText')
91
92     tarifa_imposto12 = page.locator('#table-energy-
           microgeneration_action_item_1062603347 > tbody >
               tr:nth-child(1) > td:nth-child(9)')
93     ti12= tarifa_imposto12.evaluate('el => el.outerText')
94
95     medidor12 = page.locator('#table-energy-
           microgeneration_action_item_1062603347 > tbody > tr:nth-
               child(1) > td:nth-child(11)')
96     med12= medidor12.evaluate('el => el.outerText')
97     #Delay para executar o proximo script
98     time.sleep(20)
99
100    #Organizando os dados coletados
101    con_12 = float(con12)
102
103    inj_12 = float(inj12)
104
105    cred_12 = float(cred12)
106
107    ti_12 = float(ti12)
108
```

```
109     page.wait_for_timeout (3000)
110     browser.close()
111     autoconsumo = (g_perodo - inj_12)
112     a_consumo = round(autoconsumo, 2)
113     auto_consumo = str(a_consumo)
114
115     print("Autoconsumo:", auto_consumo)
116
117
118     import numpy as np
119     import seaborn as sns
120     sns.set(style="whitegrid") #deixa o grafico um pouco mais bonito
121     import matplotlib.pyplot as plt #carrega a biblioteca que contem os
        recursos de plotagem
122     from operator import itemgetter #utiliza o procedimento de classificacao
123
124     D = [(m1, inj_1, con_1), (m2, inj_2, con_2), (m3, inj_3, con_3), (m4, inj_4, con_4)
        , (m5, inj_5, con_5), (m6, inj_6, con_6), (m7, inj_7, con_7), (m8, inj_8, con_8)
        , (m9, inj_9, con_9), (m10, inj_10, con_10), (m11, inj_11, con_11), (m12, inj_12
        , con_12)] #recebe os dados para montar o grafico
125
126     lang = [x[0] for x in D] #cria uma lista a partir da primeira dimensao
        de dados
127     use = [x[1] for x in D]
128     use2 = [x[2] for x in D]
129
130     ind = np.arange(len(lang))
131     width=0.2
132
133     ax = plt.subplot(111)
134     ax.barh(ind, use, width, align='center', alpha=0.7, color='orange',
        label='Ger.>') #um gr fico de barras horizontais barh
135     ax.barh(ind - width, use2, width, align='center', alpha=0.7, color='red
        ', label='Con.>')
136     ax.set(yticks=ind - width/2, yticklabels=lang, ylim=[2*width - 1.5, len(
        lang)])
137
138     for i, v in enumerate(use):
139         ax.text(v+0.15, i-0.05, str(v), color='black', fontsize=7) #0s
        valores 0,15 e 0,05 foram definidos ap s tentativa e erro
140     for i, v in enumerate(use2):
141         ax.text(v+0.15, i-0.4, str(v), color='black', fontsize=7) #0 valor
        0,4 foi definidos ap s tentativa e erro
142
143     ax = plt.gca ()
144
145     #oculta o eixo x
146     ax.get_xaxis().set_visible( False )
```



```
147
148 #oculta eixo do eixo y
149 #ax.get_yaxis (). set_visible ( False )
150
151 plt.grid(b = None)
152 right_side = ax.spines["right"]
153 right_side.set_visible(False)
154 left_side = ax.spines["left"]
155 left_side.set_visible(False)
156 top_side = ax.spines["top"]
157 top_side.set_visible(False)
158 bottom_side = ax.spines["bottom"]
159 bottom_side.set_visible(False)
160 plt.legend()
161 plt.savefig("graphics.png", format="png") #especificando o tipo de
    arquivo explicitamente
162
163 from reportlab.pdfgen import canvas
164
165 def add_image():
166
167     from PyPDF2 import PdfFileWriter, PdfFileReader
168     import io
169
170     in_pdf_file = r'C:\Desktop\modelo_relatorio.pdf' #importa o modelo
        padrao do relat rio
171     out_pdf_file = r'C:\Desktop\Relatorio do Cliente.pdf' #define o
        documento final que ser gerado
172     img_file = r'C:\Desktop\graphics.png' #importa o gr fico gerado
        previamente em formato png
173
174 packet = io.BytesIO()
175     can = canvas.Canvas(packet)
176     can.setFont('Helvetica-Bold', 12)
177     can.drawString(335, 661, geracao_perodo) #gera o
178     can.drawString(335, 601, con12) #consumo
179     can.drawString(335, 544, v_fatura) #fatura energia
180     can.drawString(335, 481, "xxxx,xx") #economia total
181
182     can.drawString(470, 661, "xx.xxx") #atendimento
183     can.drawString(470, 603, "xx.xxx") #economia
184     can.drawString(470, 542, auto_consumo) #autoconsumo
185     can.drawString(470, 483, "xxx,xx") #gera o 3
186
187     can.setFont('Helvetica-Bold', 16)
188     can.drawString(88, 671, cliente) #nome do cliente
189     can.setFont('Helvetica', 14)
190     can.drawString(88, 654, "17,085") #kwp
```

```
191
192     can.setFillColorRGB(242/255,140/255,0)
193     can.setFont('Helvetica-Bold', 10)
194     can.drawString(145, 578, con12) #imagem consumo distribuidora
195     can.drawString(70, 522, auto_consumo) #imagem autoconsumo
196     can.drawString(220, 522, inj12) #imagem energia injetada
197
198     can.setFillColorRGB(255,255,255)
199     can.setFont('Helvetica', 12)
200     can.drawString(100, 30, "xxx,xx") #co2
201     can.drawString(310, 30, "xxx,xx") #arvores
202
203     can.setFillColorRGB(0,0,0)
204     can.setFont('Helvetica-Bold', 10) #tabela
205     can.drawString(322, 440, "M s")
206     can.drawString(308, 428, "Refer ncia")
207     can.drawString(378, 440, "Energia")
208     can.drawString(378, 428, "Injetada")
209     can.drawString(445, 440, "Cr dito")
210     can.drawString(445, 428, "Abatido")
211     can.drawString(506, 440, "Saldo de")
212     can.drawString(509, 428, "Cr dito")
213
214     can.setFillColorRGB(0,0,0)
215     can.setFont('Helvetica-Bold', 10) #dados da tabela
216     can.drawString(318, 400, m12)
217     can.drawString(318, 373, m11)
218     can.drawString(318, 345, m10)
219     can.drawString(318, 319, m9)
220     can.drawString(318, 294, m8)
221     can.drawString(318, 269, m7)
222     can.drawString(318, 243, m6)
223     can.drawString(318, 217, m5)
224     can.drawString(318, 190, m4)
225     can.drawString(318, 163, m3)
226     can.drawString(318, 137, m2)
227     can.drawString(318, 111, m1)
228
229     can.drawString(383, 400, inj12) #energia injetada
230     can.drawString(383, 373, inj11)
231     can.drawString(383, 345, inj10)
232     can.drawString(383, 319, inj9)
233     can.drawString(383, 294, inj8)
234     can.drawString(383, 269, inj7)
235     can.drawString(383, 243, inj6)
236     can.drawString(383, 217, inj5)
237     can.drawString(383, 190, inj4)
238     can.drawString(383, 163, inj3)
```

```
239     can.drawString(383, 137, inj2)
240     can.drawString(383, 111, inj1)
241
242     can.drawString(447, 400, cred12) #credito abatido
243     can.drawString(447, 373, cred11)
244     can.drawString(447, 345, cred10)
245     can.drawString(447, 319, cred9)
246     can.drawString(447, 294, cred8)
247     can.drawString(447, 269, cred7)
248     can.drawString(447, 243, cred6)
249     can.drawString(447, 217, cred5)
250     can.drawString(447, 190, cred4)
251     can.drawString(447, 163, cred3)
252     can.drawString(447, 137, cred2)
253     can.drawString(447, 111, cred1)
254
255     can.drawString(511, 400, "1.866,00") #saldo
256     can.drawString(511, 373, "1.391,00")
257     can.drawString(511, 345, "883,00")
258     can.drawString(511, 319, "0,00")
259     can.drawString(511, 294, "0,00")
260     can.drawString(511, 269, "0,00")
261     can.drawString(511, 243, "0,00")
262     can.drawString(511, 217, "0,00")
263     can.drawString(511, 190, "1.073,00")
264     can.drawString(511, 163, "555,00")
265     can.drawString(511, 137, "0,00")
266     can.drawString(511, 111, "0,00")
267
268
269     x_start = 42
270     y_start = 105
271     can.drawImage(img_file, x_start, y_start, width=235, height=340,
272                  mask='auto')
272     can.showPage()
273     can.showPage()
274     can.showPage()
275     can.save()
276
277 #mover para o inicio do buffer StringIO
278     packet.seek(0)
279
280     new_pdf = PdfFileReader(packet)
281
282     #1 o PDF existente
283     existing_pdf = PdfFileReader(open(in_pdf_file, "rb"))
284     output = PdfFileWriter()
285
```

```
286         for i in range(len(existing_pdf.pages)):
287             page = existing_pdf.getPage(i)
288             page.mergePage(new_pdf.getPage(i))
289             output.addPage(page)
290
291         outputStream = open(out_pdf_file, "wb")
292         output.write(outputStream)
293         outputStream.close()
294
295
296 add_image()
```

APÊNDICE B – Conexão do Site com Banco de Dados

```
1 <?php
2 /* Define os dados de acesso */
3 define('DB_SERVER', "usuario");
4 define('DB_USERNAME', "root");
5 define('DB_PASSWORD', "");
6 define('DB_NAME', "clientes_mvp");
7
8 /* Faz a conexao com o Banco de Dados */
9 $link = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD, DB_NAME);
10
11 //Verifica o status da conexao
12 if($link === false){
13     die("ERROR: Could not connect. " . mysqli_connect_error());
14 }
15 ?>
```

APÊNDICE C – Estilo Padrão das Páginas

```
1 <style>
2 body {
3   font-family: Arial, Helvetica, sans-serif;
4 }
5
6 * {
7   box-sizing: border-box;
8 }
9
10 /* style the container */
11 .container {
12   position: relative;
13   border-radius: 5px;
14   /* background-color: #f2f2f2;*/
15   padding: 20px 0 30px 0;
16 }
17
18 /* style inputs and link buttons */
19 input,
20 .btn {
21   width: 100%;
22   padding: 12px;
23   border: none;
24   border-radius: 4px;
25   margin: 5px 0;
26   opacity: 0.85;
27   display: inline-block;
28   font-size: 17px;
29   line-height: 20px;
30   text-decoration: none; /* remove underline from anchors */
31 }
32
33 input:hover,
34 .btn:hover {
35   opacity: 1;
36 }
37
38 /* add appropriate colors to fb, twitter and google buttons */
39 .fb {
40   background-color: #3B5998;
41   color: white;
42 }
43
44 .twitter {
```

```
45     background-color: #55ACEE;
46     color: white;
47 }
48
49 .google {
50     background-color: #dd4b39;
51     color: white;
52 }
53
54 /* style the submit button */
55 input[type=submit] {
56     background-color: #ff9d44;
57     color: white;
58     cursor: pointer;
59 }
60
61 input[type=submit]:hover {
62     background-color: #000000;
63 }
64
65 /* Two-column layout */
66 .col {
67     float: left;
68     width: 50%;
69     margin: auto;
70     padding: 0 50px;
71     margin-top: 6px;
72 }
73
74 /* Clear floats after the columns */
75 .row:after {
76     content: "";
77     display: table;
78     clear: both;
79 }
80
81 /* vertical line */
82 .vl {
83     position: absolute;
84     left: 50%;
85     transform: translate(-50%);
86     border: 2px solid #ddd;
87     height: 175px;
88 }
89
90 /* text inside the vertical line */
91 .vl-innertext {
92     position: absolute;
```

```
93     top: 50%;
94     transform: translate(-50%, -50%);
95     background-color: #f1f1f1;
96     border: 1px solid #ccc;
97     border-radius: 50%;
98     padding: 8px 10px;
99 }
100
101 /* hide some text on medium and large screens */
102 .hide-md-lg {
103     display: none;
104 }
105
106 /* bottom container */
107 .bottom-container {
108     text-align: center;
109     background-color: rgb(0, 0, 0);
110     border-radius: 0px 0px 4px 4px;
111 }
112
113 /* Responsive layout - when the screen is less than 650px wide, make the
114     two columns stack on top of each other instead of next to each other
115     */
114 @media screen and (max-width: 650px) {
115     .col {
116         width: 100%;
117         margin-top: 0;
118     }
119     /* hide the vertical line */
120     .vl {
121         display: none;
122     }
123     /* show the hidden text on small screens */
124     .hide-md-lg {
125         display: block;
126         text-align: center;
127     }
128 }
129 </style>
```

APÊNDICE D – Página Inicial do Site

```
1
2 <!DOCTYPE html>
3 <html lang="pt-br">
4   <head>
5     <!-- Meta tags Obrigat rias -->
6     <meta charset="utf-8">
7     <meta name="viewport" content="width=device-width, initial-scale=1,
8       shrink-to-fit=no">
9
10    <!-- Bootstrap CSS -->
11    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/
12      bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/
13      SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPM0"
14      crossorigin="anonymous">
15
16    <!-- Font Awesome -->
17    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5
18      .3.1/css/all.css" integrity="sha384-
19      mzrmE5qonljUremFsqc01SB46JvROS7bZs3IO2EmfFsd15uHvIt+Y8vEf7N7fWAU"
20      crossorigin="anonymous">
21
22    <!-- Estilo customizado -->
23    <link rel="stylesheet" type="text/css" href="css/estilo.css">
24
25  </head>
26
27  <body>
28
29    <header><!-- inicio Cabecalho -->
30      <nav class="navbar navbar-expand-sm navbar-light bg-dark">
31        <div class="container">
32
33          <a href="#" class="navbar-brand">
34            
35          </a>
36
37          <button class="navbar-toggler" data-toggle="collapse" data-
38            target="#nav-principal">
39            <span class="navbar-toggler-icon"></span>
40          </button>
41
42          <div class="collapse navbar-collapse" id="nav-principal">
43            <ul class="navbar-nav ml-auto">
44              <li class="nav-item">
```

```
37         <a href="" class="nav-link">Home</a>
38     </li>
39     <li class="nav-item">
40         <a href="" class="nav-link">Recursos</a>
41     </li>
42     <li class="nav-item">
43         <a href="" class="nav-link">Benefícios</a>
44     </li>
45     <li class="nav-item">
46         <a href="" class="nav-link">Preços</a>
47     </li>
48     <li class="nav-item">
49         <a href="login.php" class="btn btn-outline-light ml-4">
50             Entrar</a>
51     </li>
52 </ul>
53 </div>
54 </nav>
55 </header><!--/fim Cabecalho -->
56
57 <section id="home"><!-- Início se o home -->
58     <div class="container">
59         <div class="row">
60             <div class="col-md-6 d-flex"><!-- Textos da seção -->
61                 <div class="align-self-center">
62                     <h1 class="display-4">Saiba como está sua Usina
63                         Fotovoltáica!</h1>
64                 <p>
65                     Essa plataforma existe para simplificar tudo que você
66                         precisa saber da sua usina, de maneira rápida e
67                         dinâmica.
68                 </p>
69
70                 <form class="mt-4 mb-4">
71                     <a href="register.php" class="btn btn-dark">Cadastre-se
72                         </a>
73                 </form>
74
75                 <p>Disponível para
76                     <a href="" class="btn btn-outline-light">
77                         <i class="fab fa-android fa-lg"></i>
78                     </a>
79                     <a href="" class="btn btn-outline-light">
80                         <i class="fab fa-apple"></i>
81                     </a>
```

```
80         </p>
81
82     </div>
83 </div><!--/fim textos da seção -->
84 <div class="col-md-6 d-none d-md-block">
85     
86 </div>
87 </div>
88 </div>
89 </section><!--/fim seção home -->
90
91 <section class="caixa"><!--/Início seção saiba -->
92     <div class="container">
93         <div class="row">
94             <div class="col-md-6 d-flex">
95                 <div class="align-self-center">
96                     <h2>Saiba quanto você economiza mensalmente</h2>
97                     <p>
98                         Com esse sistema de gerenciamento, você acompanha toda
99                         a sua produção de energia mensal sem complicação.
100                     </p>
101                     <a href="" class="btn btn-dark">Veja mais</a>
102                 </div>
103             </div>
104             <div class="col-md-6">
105                 
106             </div>
107         </div>
108     </section><!--/FIM seção saiba -->
109
110 <section class="caixa"><!--/Início seção juro -->
111     <div class="container">
112         <div class="row">
113             <div class="col-md-6">
114                 
115             </div>
116             <div class="col-md-6 d-flex">
117                 <div class="align-self-center">
118                     <h2>Informe o a qualquer hora em seus dispositivos</h2>
119                     <p>
120                         Acesse suas informações de qualquer dispositivo com
121                         acesso a internet, a qualquer hora do dia.
122                     </p>
123                     <a href="" class="btn btn-dark">Veja mais</a>
124                 </div>
125             </div>
126         </div>
127     </div>
```

```
126     </div>
127 </section><!--/FIM se o juro -->
128
129 <section class="caixa"><!--/Início se o recurso -->
130   <div class="container">
131     <div class="row">
132       <div class="col-md-4">
133         
134         <h4>Facil de usar</h4>
135         <p>
136           Basta fazer seu cadastro ou login e j ter acesso a
137             todas as informa es dispon veis.
138         </p>
139       </div>
140       <div class="col-md-4">
141         
142         <h4>Economize seu tempo</h4>
143         <p>
144           Os seus dados de gera o , faturas e saldo de cr ditos
145             com a concession ria de energia em um s lugar!
146         </p>
147       </div>
148       <div class="col-md-4">
149         
150         <h4>Suporte amigo</h4>
151         <p>
152           D vidas? Perguntas? Nosso suporte super legal ajuda voc
153             !
154         </p>
155       </div>
156     </div>
157   </div>
158 </section><!--/FIM se o recurso -->
159
160 <footer class="mt-4 mb-4">
161   <div class="container">
162     <div class="row">
163       <div class="col-md-8">
164         <p>
165           <a href=""></a>
166           <a href=""></a>
167           <a href=""></a>
168           <a href=""></a>
169         </p>
170       </div>
171     </div>
172     <div class="col-md-4 d-flex justify-content-end">
173       <a href="" class="btn btn-outline-dark">
174         <i class="fab fa-facebook"></i>
```

```
171         </a>
172         <a href="" class="btn btn-outline-dark ml-2">
173             <i class="fab fa-twitter"></i>
174         </a>
175         <a href="" class="btn btn-outline-dark ml-2">
176             <i class="fab fa-instagram"></i>
177         </a>
178         <a href="" class="btn btn-outline-dark ml-2">
179             <i class="fab fa-youtube"></i>
180         </a>
181     </div>
182 </div>
183 </div>
184 </footer>
```

APÊNDICE E – Página de Registro de Usuário

```
1     <?php
2 // Include config file
3 require_once "config.php";
4
5 // Define variables and initialize with empty values
6 $username = $password = $confirm_password = "";
7 $username_err = $password_err = $confirm_password_err = "";
8
9 // Processing form data when form is submitted
10 if($_SERVER["REQUEST_METHOD"] == "POST"){
11
12     // Validate username
13     if(empty(trim($_POST["username"]))){
14         $username_err = "Please enter a username.";
15     } elseif(!preg_match('/^[a-zA-Z0-9_]+$/', trim($_POST["username"])))
16     {
17         $username_err = "Username can only contain letters, numbers, and
18         underscores.";
19     } else{
20         // Prepare a select statement
21         $sql = "SELECT id FROM users WHERE username = ?";
22
23         if($stmt = mysqli_prepare($link, $sql)){
24             // Bind variables to the prepared statement as parameters
25             mysqli_stmt_bind_param($stmt, "s", $param_username);
26
27             // Set parameters
28             $param_username = trim($_POST["username"]);
29
30             // Attempt to execute the prepared statement
31             if(mysqli_stmt_execute($stmt)){
32                 /* store result */
33                 mysqli_stmt_store_result($stmt);
34
35                 if(mysqli_stmt_num_rows($stmt) == 1){
36                     $username_err = "This username is already taken.";
37                 } else{
38                     $username = trim($_POST["username"]);
39                 }
40             } else{
41                 echo "Oops! Something went wrong. Please try again later
42                 .";
43             }
44         }
45     }
46 }
```

```
42         // Close statement
43         mysqli_stmt_close($stmt);
44     }
45 }
46
47 // Validate password
48 if(empty(trim($_POST["password"]))) {
49     $password_err = "Please enter a password.";
50 } elseif(strlen(trim($_POST["password"])) < 6) {
51     $password_err = "Password must have atleast 6 characters.";
52 } else {
53     $password = trim($_POST["password"]);
54 }
55
56 // Validate confirm password
57 if(empty(trim($_POST["confirm_password"]))) {
58     $confirm_password_err = "Please confirm password.";
59 } else {
60     $confirm_password = trim($_POST["confirm_password"]);
61     if(empty($password_err) && ($password != $confirm_password)) {
62         $confirm_password_err = "Password did not match.";
63     }
64 }
65
66 // Check input errors before inserting in database
67 if(empty($username_err) && empty($password_err) && empty(
68     $confirm_password_err)) {
69     // Prepare an insert statement
70     $sql = "INSERT INTO users (username, password) VALUES (?, ?)";
71
72     if($stmt = mysqli_prepare($link, $sql)) {
73         // Bind variables to the prepared statement as parameters
74         mysqli_stmt_bind_param($stmt, "ss", $param_username,
75             $param_password);
76
77         // Set parameters
78         $param_username = $username;
79         $param_password = password_hash($password, PASSWORD_DEFAULT)
80             ; // Creates a password hash
81
82         // Attempt to execute the prepared statement
83         if(mysqli_stmt_execute($stmt)) {
84             // Redirect to login page
85             header("location: login.php");
86         } else {
87             echo "Oops! Something went wrong. Please try again later
88                 .";
89         }
90     }
91 }
```

```
86         }
87
88         // Close statement
89         mysqli_stmt_close($stmt);
90     }
91 }
92
93 // Close connection
94 mysqli_close($link);
95 }
96 ?>
97
98 <!DOCTYPE html>
99 <html lang="en">
100 <head>
101 <head>
102 <meta name="viewport" content="width=device-width, initial-scale=1">
103 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font
    -awesome/4.7.0/css/font-awesome.min.css">
104 <style>
105 body {
106     font-family: Arial, Helvetica, sans-serif;
107 }
108
109 * {
110     box-sizing: border-box;
111 }
112
113 /* style the container */
114 .container {
115     position: relative;
116     border-radius: 5px;
117     /* background-color: #f2f2f2;*/
118     padding: 20px 0 30px 0;
119 }
120
121 /* style inputs and link buttons */
122 input,
123 .btn {
124     width: 100%;
125     padding: 12px;
126     border: none;
127     border-radius: 4px;
128     margin: 5px 0;
129     opacity: 0.85;
130     display: inline-block;
131     font-size: 17px;
132     line-height: 20px;
```



```
133     text-decoration: none; /* remove underline from anchors */
134 }
135
136 input:hover,
137 .btn:hover {
138     opacity: 1;
139 }
140
141 /* add appropriate colors to fb, twitter and google buttons */
142 .fb {
143     background-color: #3B5998;
144     color: white;
145 }
146
147 .twitter {
148     background-color: #55ACEE;
149     color: white;
150 }
151
152 .google {
153     background-color: #dd4b39;
154     color: white;
155 }
156
157 /* style the submit button */
158 input[type=submit] {
159     background-color: #ff9d44;
160     color: white;
161     cursor: pointer;
162 }
163
164 input[type=submit]:hover {
165     background-color: #000000;
166 }
167
168 /* Two-column layout */
169 .col {
170     float: left;
171     width: 50%;
172     margin: auto;
173     padding: 0 50px;
174     margin-top: 6px;
175 }
176
177 /* Clear floats after the columns */
178 .row:after {
179     content: "";
180     display: table;
```

```
181     clear: both;
182 }
183
184 /* vertical line */
185 .vl {
186     position: absolute;
187     left: 50%;
188     transform: translate(-50%);
189     border: 2px solid #ddd;
190     height: 175px;
191 }
192
193 /* text inside the vertical line */
194 .vl-innertext {
195     position: absolute;
196     top: 50%;
197     transform: translate(-50%, -50%);
198     background-color: #f1f1f1;
199     border: 1px solid #ccc;
200     border-radius: 50%;
201     padding: 8px 10px;
202 }
203
204 /* hide some text on medium and large screens */
205 .hide-md-lg {
206     display: none;
207 }
208
209 /* bottom container */
210 .bottom-container {
211     text-align: center;
212     background-color: rgb(0, 0, 0);
213     border-radius: 0px 0px 4px 4px;
214 }
215
216 /* Responsive layout - when the screen is less than 650px wide, make the
      two columns stack on top of each other instead of next to each other
      */
217 @media screen and (max-width: 650px) {
218     .col {
219         width: 100%;
220         margin-top: 0;
221     }
222     /* hide the vertical line */
223     .vl {
224         display: none;
225     }
226     /* show the hidden text on small screens */
```

```
227     .hide-md-lg {
228         display: block;
229         text-align: center;
230     }
231 }
232 </style>
233
234     <!-- Meta tags Obrigatórias -->
235     <meta charset="utf-8">
236     <meta name="viewport" content="width=device-width, initial-scale=1,
237         shrink-to-fit=no">
238
239     <!-- Bootstrap CSS -->
240     <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/
241         bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/
242         SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkF0JwJ8ERdknLPM0"
243         crossorigin="anonymous">
244
245     <!-- Font Awesome -->
246     <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5
247         .3.1/css/all.css" integrity="sha384-
248         mZrmE5qonljUremFsqc01SB46JvROS7bZs3IO2EmfFsd15uHvIt+Y8vEf7N7fWAU"
249         crossorigin="anonymous">
250
251     <!-- Estilo customizado -->
252     <link rel="stylesheet" type="text/css" href="css/estilo.css">
253
254 <header><!-- inicio Cabecalho -->
255     <nav class="navbar navbar-expand-sm navbar-light bg-dark">
256         <div class="container">
257
258             <a href="#" class="navbar-brand">
259                 
260             </a>
261
262             <button class="navbar-toggler" data-toggle="collapse" data-
263                 target="#nav-principal">
264                 <span class="navbar-toggler-icon"></span>
265             </button>
266
267             <div class="collapse navbar-collapse" id="nav-principal">
268                 <ul class="navbar-nav ml-auto">
269                     <li class="nav-item">
270                         <a href="" class="nav-link">Home</a>
271                     </li>
272                     <li class="nav-item">
273                         <a href="" class="nav-link">Recursos</a>
274                     </li>
```

```
267         <li class="nav-item">
268             <a href="" class="nav-link">Benefícios </a>
269         </li>
270         <li class="nav-item">
271             <a href="" class="nav-link">Preços </a>
272         </li>
273         <li class="nav-item">
274             <a href="index.php" class="btn btn-outline-light ml-4">
                Home</a>
275         </li>
276     </ul>
277 </div>
278
279 </div>
280 </nav>
281 </header><!--/fim Cabecalho -->
282 </head>
283 </head>
284 <body>
285     <div class="wrapper">
286         <h2>Crie sua conta</h2>
287
288         <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);
                ?>" method="post">
289             <div class="form-group">
290                 <label>Usuário </label>
291                 <input type="text" name="username" class="form-control
                    <?php echo (!empty($username_err)) ? 'is-invalid' :
                    '' ; ?>" value="<?php echo $username; ?>">
292                 <span class="invalid-feedback"><?php echo $username_err;
                    ?></span>
293             </div>
294             <div class="form-group">
295                 <label>Senha</label>
296                 <input type="password" name="password" class="form-
                    control <?php echo (!empty($password_err)) ? 'is-
                    invalid' : '' ; ?>" value="<?php echo $password; ?>">
297                 <span class="invalid-feedback"><?php echo $password_err;
                    ?></span>
298             </div>
299             <div class="form-group">
300                 <label>Confirme sua senha</label>
301                 <input type="password" name="confirm_password" class="
                    form-control <?php echo (!empty($confirm_password_err
                    )) ? 'is-invalid' : '' ; ?>" value="<?php echo
                    $confirm_password; ?>">
302                 <span class="invalid-feedback"><?php echo
                    $confirm_password_err; ?></span>
```

```
303         </div>
304         <div class="form-group">
305             <input type="submit" class="btn btn-primary" value="
306                 Registrar">
307         </div>
308         <p>J    possui uma conta? <a href="login.php">Fazer Login</a
309             >.</p>
310     </form>
311 </div>
312 </body>
313 </html>
```

APÊNDICE F – Página de Login de Usuário

```
1 <?php
2 // Initialize the session
3 session_start();
4
5 // Check if the user is already logged in, if yes then redirect him to
   welcome page
6 if(isset($_SESSION["loggedin"]) && $_SESSION["loggedin"] === true){
7     header("location: welcome.php");
8     exit;
9 }
10
11 // Include config file
12 require_once "config.php";
13
14 // Define variables and initialize with empty values
15 $username = $password = "";
16 $username_err = $password_err = $login_err = "";
17
18 // Processing form data when form is submitted
19 if($_SERVER["REQUEST_METHOD"] == "POST"){
20
21     // Check if username is empty
22     if(empty(trim($_POST["username"]))){
23         $username_err = "Please enter username.";
24     } else{
25         $username = trim($_POST["username"]);
26     }
27
28     // Check if password is empty
29     if(empty(trim($_POST["password"]))){
30         $password_err = "Please enter your password.";
31     } else{
32         $password = trim($_POST["password"]);
33     }
34
35     // Validate credentials
36     if(empty($username_err) && empty($password_err)){
37         // Prepare a select statement
38         $sql = "SELECT id, username, password FROM users WHERE username
   = ?";
39
40         if($stmt = mysqli_prepare($link, $sql)){
41             // Bind variables to the prepared statement as parameters
42             mysqli_stmt_bind_param($stmt, "s", $param_username);
```

```
43
44     // Set parameters
45     $param_username = $username;
46
47     // Attempt to execute the prepared statement
48     if(mysqli_stmt_execute($stmt)){
49         // Store result
50         mysqli_stmt_store_result($stmt);
51
52         // Check if username exists, if yes then verify password
53         if(mysqli_stmt_num_rows($stmt) == 1){
54             // Bind result variables
55             mysqli_stmt_bind_result($stmt, $id, $username,
56                                     $hashed_password);
57             if(mysqli_stmt_fetch($stmt)){
58                 if(password_verify($password, $hashed_password))
59                 {
60                     // Password is correct, so start a new
61                     session
62                     session_start();
63
64                     // Store data in session variables
65                     $_SESSION["loggedin"] = true;
66                     $_SESSION["id"] = $id;
67                     $_SESSION["username"] = $username;
68
69                     // Redirect user to welcome page
70                     header("location: welcome.php");
71                 } else{
72                     // Password is not valid, display a generic
73                     error message
74                     $login_err = "Invalid username or password
75                     .";
76                 }
77             }
78         } else{
79             // Username doesn't exist, display a generic error
80             message
81             $login_err = "Invalid username or password.";
82         }
83     } else{
84         echo "Oops! Something went wrong. Please try again later
85         .";
86     }
87
88     // Close statement
89     mysqli_stmt_close($stmt);
90 }
```

```
84     }
85
86     // Close connection
87     mysqli_close($link);
88 }
89 ?>
90
91 <!DOCTYPE html>
92 <html lang="en">
93 <head>
94 <meta name="viewport" content="width=device-width, initial-scale=1">
95 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font
    -awesome/4.7.0/css/font-awesome.min.css">
96 <style>
97 body {
98     font-family: Arial, Helvetica, sans-serif;
99 }
100
101 * {
102     box-sizing: border-box;
103 }
104
105 /* style the container */
106 .container {
107     position: relative;
108     border-radius: 5px;
109     /* background-color: #f2f2f2;*/
110     padding: 20px 0 30px 0;
111 }
112
113 /* style inputs and link buttons */
114 input,
115 .btn {
116     width: 100%;
117     padding: 12px;
118     border: none;
119     border-radius: 4px;
120     margin: 5px 0;
121     opacity: 0.85;
122     display: inline-block;
123     font-size: 17px;
124     line-height: 20px;
125     text-decoration: none; /* remove underline from anchors */
126 }
127
128 input:hover,
129 .btn:hover {
130     opacity: 1;
```



```
131 }
132
133 /* add appropriate colors to fb, twitter and google buttons */
134 .fb {
135     background-color: #3B5998;
136     color: white;
137 }
138
139 .twitter {
140     background-color: #55ACEE;
141     color: white;
142 }
143
144 .google {
145     background-color: #dd4b39;
146     color: white;
147 }
148
149 /* style the submit button */
150 input[type=submit] {
151     background-color: #ff9d44;
152     color: white;
153     cursor: pointer;
154 }
155
156 input[type=submit]:hover {
157     background-color: #000000;
158 }
159
160 /* Two-column layout */
161 .col {
162     float: left;
163     width: 50%;
164     margin: auto;
165     padding: 0 50px;
166     margin-top: 6px;
167 }
168
169 /* Clear floats after the columns */
170 .row:after {
171     content: "";
172     display: table;
173     clear: both;
174 }
175
176 /* vertical line */
177 .vl {
178     position: absolute;
```

```
179     left: 50%;
180     transform: translate(-50%);
181     border: 2px solid #ddd;
182     height: 175px;
183 }
184
185 /* text inside the vertical line */
186 .vl-innertext {
187     position: absolute;
188     top: 50%;
189     transform: translate(-50%, -50%);
190     background-color: #f1f1f1;
191     border: 1px solid #ccc;
192     border-radius: 50%;
193     padding: 8px 10px;
194 }
195
196 /* hide some text on medium and large screens */
197 .hide-md-lg {
198     display: none;
199 }
200
201 /* bottom container */
202 .bottom-container {
203     text-align: center;
204     background-color: rgb(0, 0, 0);
205     border-radius: 0px 0px 4px 4px;
206 }
207
208 /* Responsive layout - when the screen is less than 650px wide, make the
      two columns stack on top of each other instead of next to each other
      */
209 @media screen and (max-width: 650px) {
210     .col {
211         width: 100%;
212         margin-top: 0;
213     }
214     /* hide the vertical line */
215     .vl {
216         display: none;
217     }
218     /* show the hidden text on small screens */
219     .hide-md-lg {
220         display: block;
221         text-align: center;
222     }
223 }
224 </style>
```

```
225
226     <!-- Meta tags Obrigat rias -->
227     <meta charset="utf-8">
228     <meta name="viewport" content="width=device-width, initial-scale=1,
        shrink-to-fit=no">
229
230     <!-- Bootstrap CSS -->
231     <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/
        bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/
        SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPM0"
        crossorigin="anonymous">
232
233     <!-- Font Awesome -->
234     <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5
        .3.1/css/all.css" integrity="sha384-
        mzrmE5qonljUremFsqc01SB46JvROS7bZs3IO2EmfFsd15uHvIt+Y8vEf7N7fWAU"
        crossorigin="anonymous">
235
236     <!-- Estilo customizado -->
237     <link rel="stylesheet" type="text/css" href="css/estilo.css">
238
239 <header><!-- inicio Cabecalho -->
240     <nav class="navbar navbar-expand-sm navbar-light bg-dark">
241         <div class="container">
242
243             <a href="#" class="navbar-brand">
244                 
245             </a>
246
247             <button class="navbar-toggler" data-toggle="collapse" data-
                target="#nav-principal">
248                 <span class="navbar-toggler-icon"></span>
249             </button>
250
251             <div class="collapse navbar-collapse" id="nav-principal">
252                 <ul class="navbar-nav ml-auto">
253                     <li class="nav-item">
254                         <a href="" class="nav-link">Home </a>
255                     </li>
256                     <li class="nav-item">
257                         <a href="" class="nav-link">Recursos </a>
258                     </li>
259                     <li class="nav-item">
260                         <a href="" class="nav-link">Benef cios </a>
261                     </li>
262                     <li class="nav-item">
263                         <a href="" class="nav-link">Pre os </a>
264                     </li>
```

```
265         <li class="nav-item">
266             <a href="index.php" class="btn btn-outline-light ml-4">
                Home</a>
267         </li>
268     </ul>
269 </div>
270
271 </div>
272 </nav>
273 </header><!--/fim Cabecalho -->
274 </head>
275 <body>
276     <div class="wrapper">
277         <h2>Login</h2>
278
279
280         <?php
281         if(!empty($login_err)){
282             echo '<div class="alert alert-danger">' . $login_err . '</
                div>';
283         }
284         ?>
285
286         <form action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);
                ?>" method="post">
287             <div class="form-group">
288                 <label>Usu rio </label>
289                 <input type="text" name="username" class="form-control
                <?php echo (!empty($username_err)) ? 'is-invalid' :
                '' ; ?>" value="<?php echo $username ; ?>">
290                 <span class="invalid-feedback"><?php echo $username_err;
                ?></span>
291             </div>
292             <div class="form-group">
293                 <label>Senha</label>
294                 <input type="password" name="password" class="form-
                control <?php echo (!empty($password_err)) ? 'is-
                invalid' : '' ; ?>">
295                 <span class="invalid-feedback"><?php echo $password_err;
                ?></span>
296             </div>
297             <div class="form-group">
298                 <input type="submit" class="btn btn-primary" value="
                Login">
299             </div>
300             <p>N o tem uma conta? <a href="register.php">Registre-se</a
                >.</p>
301         </form>
```

```
302     </div>  
303 </body>  
304 </html>
```

APÊNDICE G – Logout de Usuário

```
1 <?php
2 // Initialize the session
3 session_start();
4
5 // Unset all of the session variables
6 $_SESSION = array();
7
8 // Destroy the session.
9 session_destroy();
10
11 // Redirect to login page
12 header("location: login.php");
13 exit;
14 ?>
```

APÊNDICE H – Exemplo de Página Privada do Usuário

```
1 <?php
2 // Initialize the session
3 session_start();
4
5 // Check if the user is logged in, if not then redirect him to login
  page
6 if(!isset($_SESSION["cliente_fulano"]) || $_SESSION["cliente_fulano"]
  !== true){
7     header("location: login.php");
8     exit;
9 }
10 ?>
11
12 <!-- PHP code to establish connection with the localserver -->
13 <?php
14
15 // Username is root
16 $user = "root";
17 $password = "";
18
19 // Database name is geeksforgeeks
20 $database = "nome_banco_de_dados";
21
22 // Server is localhost with
23 // port number 3306
24 $servername="localhost";
25 $mysqli = new mysqli($servername, $user,
26                     $password, $database);
27
28 // Checking for connections
29 if ($mysqli->connect_error) {
30     die('Connect Error (' .
31     $mysqli->connect_errno . ') '.
32     $mysqli->connect_error);
33 }
34
35 // SQL query to select data from database
36 $sql = "SELECT * FROM cliente_gustavo_torres ORDER BY Medidor";
37 $result = $mysqli->query($sql);
38 $mysqli->close();
39 ?>
40
41
42 <!-- HTML code to display data in tabular format -->
```

```
43 <!DOCTYPE html>
44 <html lang="en">
45
46 <head>
47 <style>
48 table {
49     border-collapse: collapse;
50     width: 100%;
51 }
52
53 th, td {
54     padding: 8px;
55     text-align: left;
56     border-bottom: 1px solid #ddd;
57 }
58
59 th {
60     background-color: #ff9d44;
61     color: white;
62 }
63
64 tr:hover {background-color: #ff9d44;}
65
66 body {
67     font-family: Arial, Helvetica, sans-serif;
68 }
69
70 * {
71     box-sizing: border-box;
72 }
73
74 /* style the container */
75 .container {
76     position: relative;
77     border-radius: 5px;
78     /* background-color: #f2f2f2;*/
79     padding: 20px 0 30px 0;
80 }
81
82 /* style inputs and link buttons */
83 input,
84 .btn {
85     width: 100%;
86     padding: 12px;
87     border: none;
88     border-radius: 4px;
89     margin: 5px 0;
90     opacity: 0.85;
```



```
91     display: inline-block;
92     font-size: 17px;
93     line-height: 20px;
94     text-decoration: none; /* remove underline from anchors */
95 }
96
97 input:hover,
98 .btn:hover {
99     opacity: 1;
100 }
101
102 /* add appropriate colors to fb, twitter and google buttons */
103 .fb {
104     background-color: #3B5998;
105     color: white;
106 }
107
108 .twitter {
109     background-color: #55ACEE;
110     color: white;
111 }
112
113 .google {
114     background-color: #dd4b39;
115     color: white;
116 }
117
118 /* style the submit button */
119 input[type=submit] {
120     background-color: #ff9d44;
121     color: white;
122     cursor: pointer;
123 }
124
125 input[type=submit]:hover {
126     background-color: #000000;
127 }
128
129 /* Two-column layout */
130 .col {
131     float: left;
132     width: 50%;
133     margin: auto;
134     padding: 0 50px;
135     margin-top: 6px;
136 }
137
138 /* Clear floats after the columns */
```

```
139 .row:after {
140     content: "";
141     display: table;
142     clear: both;
143 }
144
145 /* vertical line */
146 .vl {
147     position: absolute;
148     left: 50%;
149     transform: translate(-50%);
150     border: 2px solid #ddd;
151     height: 175px;
152 }
153
154 /* text inside the vertical line */
155 .vl-innertext {
156     position: absolute;
157     top: 50%;
158     transform: translate(-50%, -50%);
159     background-color: #f1f1f1;
160     border: 1px solid #ccc;
161     border-radius: 50%;
162     padding: 8px 10px;
163 }
164
165 /* hide some text on medium and large screens */
166 .hide-md-lg {
167     display: none;
168 }
169
170 /* bottom container */
171 .bottom-container {
172     text-align: center;
173     background-color: rgb(0, 0, 0);
174     border-radius: 0px 0px 4px 4px;
175 }
176
177 /* Responsive layout - when the screen is less than 650px wide, make the
178     two columns stack on top of each other instead of next to each other
179     */
180 @media screen and (max-width: 650px) {
181     .col {
182         width: 100%;
183         margin-top: 0;
184     }
185     /* hide the vertical line */
186     .vl {
```

```
185     display: none;
186   }
187   /* show the hidden text on small screens */
188   .hide-md-lg {
189     display: block;
190     text-align: center;
191   }
192 }
193 </style>
194 <!-- Meta tags Obrigat rias -->
195 <meta charset="utf-8">
196     <meta name="viewport" content="width=device-width, initial-scale=1,
197         shrink-to-fit=no">
198
199     <!-- Bootstrap CSS -->
200     <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/
201         bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/
202         SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPM0"
203         crossorigin="anonymous">
204
205     <!-- Font Awesome -->
206     <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5
207         .3.1/css/all.css" integrity="sha384-
208         mzrmE5qonljUremFsqc01SB46JvROS7bZs3IO2EmfFsd15uHvIt+Y8vEf7N7fWAU"
209         crossorigin="anonymous">
210
211     <!-- Estilo customizado -->
212     <link rel="stylesheet" type="text/css" href="css/estilo.css">
213
214 <header><!-- inicio Cabecalho -->
215     <nav class="navbar navbar-expand-sm navbar-light bg-dark">
216         <div class="container">
217
218             <a href="#" class="navbar-brand">
219                 
220             </a>
221
222             <button class="navbar-toggler" data-toggle="collapse" data-
223                 target="#nav-principal">
224                 <span class="navbar-toggler-icon"></span>
225             </button>
226
227             <div class="collapse navbar-collapse" id="nav-principal">
228                 <ul class="navbar-nav ml-auto">
229                     <li class="nav-item">
230                         <a href="" class="nav-link">Home</a>
231                     </li>
232                     <li class="nav-item">
```



```
272         <!-- FETCHING DATA FROM EACH
273             ROW OF EVERY COLUMN -->
274         <td><?php echo $rows['Mes Referencia
                '];?></td>
275         <td><?php echo $rows['Geracao Mensal
                '];?></td>
276         <td><?php echo $rows['Valor Fatura
                '];?></td>
277         <td><?php echo $rows['Leitura'];?></td>
278         <td><?php echo $rows['Energia Inj
                .'];?></td>
279         <td><?php echo $rows['Energia Consum
                .'];?></td>
280         <td><?php echo $rows['Credito Compens.'];?></td>
281         <td><?php echo $rows['Tarifa'];?></td>
282         <td><?php echo $rows['Tarifa com Imposto'];?></td>
283         <td><?php echo $rows['Medidor'];?></td>
284         <td><?php echo $rows['Autoconsumo'];?></td>
285         <td><?php echo $rows['Saldo de Credito'];?></td>
286         <td><?php echo $rows['Economia'];?></td>
287         <td><?php echo $rows['Atendimento'];?></
                td>
288     </tr>
289     <?php
290         }
291     ?>
292 </table>
293 </section>
294 </body>
295
296 </html>
```

APÊNDICE I – Página de Controle de Usuários - Acesso de Administrador

```
1 <!DOCTYPE html>
2 <head>
3   <style>
4   table {
5     border-collapse: collapse;
6     width: 100%;
7   }
8
9   th, td {
10    padding: 8px;
11    text-align: left;
12    border-bottom: 1px solid #ddd;
13  }
14
15  th {
16    background-color: #ff9d44;
17    color: white;
18  }
19
20  tr:hover {background-color: #ff9d44;}
21
22  body {
23    font-family: Arial, Helvetica, sans-serif;
24  }
25
26  * {
27    box-sizing: border-box;
28  }
29
30  /* style the container */
31  .container {
32    position: relative;
33    border-radius: 5px;
34    /* background-color: #f2f2f2;*/
35    padding: 20px 0 30px 0;
36  }
37
38  /* style inputs and link buttons */
39  input,
40  .btn {
41    width: 100%;
42    padding: 12px;
43    border: none;
44    border-radius: 4px;
```

```
45     margin: 5px 0;
46     opacity: 0.85;
47     display: inline-block;
48     font-size: 17px;
49     line-height: 20px;
50     text-decoration: none; /* remove underline from anchors */
51 }
52
53 input:hover,
54 .btn:hover {
55     opacity: 1;
56 }
57
58 /* add appropriate colors to fb, twitter and google buttons */
59 .fb {
60     background-color: #3B5998;
61     color: white;
62 }
63
64 .twitter {
65     background-color: #55ACEE;
66     color: white;
67 }
68
69 .google {
70     background-color: #dd4b39;
71     color: white;
72 }
73
74 /* style the submit button */
75 input[type=submit] {
76     background-color: #ff9d44;
77     color: white;
78     cursor: pointer;
79 }
80
81 input[type=submit]:hover {
82     background-color: #000000;
83 }
84
85 /* Two-column layout */
86 .col {
87     float: left;
88     width: 50%;
89     margin: auto;
90     padding: 0 50px;
91     margin-top: 6px;
92 }
```

```
93
94 /* Clear floats after the columns */
95 .row:after {
96     content: "";
97     display: table;
98     clear: both;
99 }
100
101 /* vertical line */
102 .vl {
103     position: absolute;
104     left: 50%;
105     transform: translate(-50%);
106     border: 2px solid #ddd;
107     height: 175px;
108 }
109
110 /* text inside the vertical line */
111 .vl-innertext {
112     position: absolute;
113     top: 50%;
114     transform: translate(-50%, -50%);
115     background-color: #f1f1f1;
116     border: 1px solid #ccc;
117     border-radius: 50%;
118     padding: 8px 10px;
119 }
120
121 /* hide some text on medium and large screens */
122 .hide-md-lg {
123     display: none;
124 }
125
126 /* bottom container */
127 .bottom-container {
128     text-align: center;
129     background-color: rgb(0, 0, 0);
130     border-radius: 0px 0px 4px 4px;
131 }
132
133 /* Responsive layout - when the screen is less than 650px wide, make the
134     two columns stack on top of each other instead of next to each other
135     */
134 @media screen and (max-width: 650px) {
135     .col {
136         width: 100%;
137         margin-top: 0;
138     }
```



```
139  /* hide the vertical line */
140  .vl {
141      display: none;
142  }
143  /* show the hidden text on small screens */
144  .hide-md-lg {
145      display: block;
146      text-align: center;
147  }
148  }
149  </style>
150  <!-- Meta tags Obrigat rias -->
151  <meta charset="utf-8">
152      <meta name="viewport" content="width=device-width, initial-scale=1,
153          shrink-to-fit=no">
154
155      <!-- Bootstrap CSS -->
156      <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/
157          bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/
158          SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPM0"
159          crossorigin="anonymous">
160
161      <!-- Font Awesome -->
162      <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5
163          .3.1/css/all.css" integrity="sha384-
164          mzrmE5qonljUremFsqc01SB46JvROS7bZs3IO2EmfFsd15uHvIt+Y8vEf7N7fWAU"
165          crossorigin="anonymous">
166
167      <!-- Estilo customizado -->
168      <link rel="stylesheet" type="text/css" href="css/estilo.css">
169
170  <header><!-- inicio Cabecalho -->
171      <nav class="navbar navbar-expand-sm navbar-light bg-dark">
172          <div class="container">
173
174              <a href="#" class="navbar-brand">
175                  
176              </a>
177
178              <button class="navbar-toggler" data-toggle="collapse" data-
179                  target="#nav-principal">
180                  <span class="navbar-toggler-icon"></span>
181              </button>
182
183              <div class="collapse navbar-collapse" id="nav-principal">
184                  <ul class="navbar-nav ml-auto">
185                      <li class="nav-item">
186                          <a href="" class="nav-link">Home</a>
```

```
179         </li>
180         <li class="nav-item">
181             <a href="" class="nav-link">Recursos</a>
182         </li>
183         <li class="nav-item">
184             <a href="" class="nav-link">Benefícios</a>
185         </li>
186         <li class="nav-item">
187             <a href="" class="nav-link">Preços</a>
188         </li>
189         <li class="nav-item">
190             <a href="index.php" class="btn btn-outline-light ml-4">
191                 Home</a>
192         </li>
193     </ul>
194 </div>
195 </div>
196 </nav>
197 </header><!--/fim Cabecalho -->
198
199     <script src="vendor/jquery/jquery-3.2.1.min.js"></script>
200
201     <link rel="stylesheet" href="vendor/DataTables/jquery.dataTables.
202         min.css">
203     <script src="vendor/DataTables/jquery.dataTables.min.js" type="text/
204         javascript"></script>
205
206     <link href="style.css" rel="stylesheet" type="text/css" />
207
208     <title>Registro dos Clientes</title>
209     <script>
210         $(document).ready(function ()
211         {
212             $('#tbl-contact thead th').each(function () {
213                 var title = $(this).text();
214                 $(this).html(title+' <input type="text" class="col-
215                     search-input" placeholder="Search ' + title + '" />')
216                 ;
217             });
218
219             var table = $('#tbl-contact').DataTable({
220                 "scrollX": true,
221                 "pagingType": "numbers",
222                 "processing": true,
223                 "serverSide": true,
224                 "ajax": "server.php",
225                 order: [[2, 'asc']],
```

```
222         columnDefs: [{
223             targets: "_all",
224             orderable: false
225         }]
226     });
227
228     table.columns().every(function () {
229         var table = this;
230         $('input', this.header()).on('keyup change', function ()
231         {
232             if (table.search() !== this.value) {
233                 table.search(this.value).draw();
234             }
235         });
236     });
237
238 </script>
239 </head>
240
241 <body>
242     <div class="datatable-container">
243         <h2>Registro dos Clientes</h2>
244         <table name="tbl-contact" id="tbl-contact" class="display"
245             cellspacing="0" width="100%">
246             <thead>
247                 <tr>
248
249                     <th>Nome</th>
250                     <th>M s Refer ncia</th>
251                     <th>Gera o</th>
252                     <th>Inje o</th>
253                     <th>Consumo</th>
254                     <th>Autoconsumo</th>
255
256                 </tr>
257             </thead>
258
259         </table>
260     </div>
261 </body>
262 </html>
```

APÊNDICE J – Código Arduino

```

1
2 #include <ESP8266WiFi.h>
3
4 #include <ESP8266HTTPClient.h>
5
6 #include <WiFiClient.h>
7
8 BIBLIOTECAS DO WIFI
9
10 -----
11
12 #include "EmonLib.h"
13
14 BIBLIOTECA QUE LE A TENS O E CORRENTE
15
16 -----
17
18 const char* ssid = "nome_da_rede";
19
20 const char* password = "senha";
21
22 NOME E SENHA DO WIFI
23
24 -----
25
26 const char* serverName = "localhost";
27
28 String apiKeyValue = "";
29
30 String sensorLocation = "local_de_teste";
31
32 #define SEALEVELPRESSURE_HPA (1013.25)
33
34 CONFIGURA ES DA INTEGRA O COM O SERVIDOR
35
36 -----
37
38 EnergyMonitor emon1;
39
40 COLOCA UM NOME MAIS FACIL PARA CHAMAR O MONITOR DE ENERGIA (EMON1)
41
42 -----
43
44 void setup()

```

```
45
46 {
47
48 -----
49
50 Serial.begin(500000);
51
52 INICIA A COMUNICACÃO SERIAL (BUTEI A VELOCIDADE MAIS ALTA PARA NÃO
    TRAVAR O CÓDIGO
53
54 -----
55
56 WiFi.begin(ssid, password);
57
58 Serial.println("Connecting");
59
60 while(WiFi.status() != WL_CONNECTED) {
61
62   delay(500);
63
64   Serial.print(".");
65
66 }
67
68 LOOP PRA CONECTAR NO WIFI
69
70 -----
71
72 Serial.println("");
73
74 Serial.print("Connected to WiFi network with IP Address: ");
75
76 Serial.println(WiFi.localIP());
77
78 PRINTA QUAL O IP QUE CONECTOU E FALA QUE DEU CERTO
79
80 -----
81
82 emon1.voltage1(0, 740, 0); // Voltage: input pin, calibration,
    phase_shift
83
84 emon1.voltage2(3, 740, 0); // Voltage: input pin, calibration,
    phase_shift
85
86 emon1.voltage3(4, 740, 0); // Voltage: input pin, calibration,
    phase_shift
87
```

```
88 SETA AS POSTAS DE LEITURA DE TENS O E CALIBRA O (IGUAIS PQ A
    CALIBRA O FOI NO POTENCIOMETRO NA PLACA)
89
90 -----
91
92 emon1.current1(5, 49.82); // Current: input pin, calibration.
93
94 emon1.current2(6, 49.80); // Current: input pin, calibration.
95
96 emon1.current3(7, 49.84); // Current: input pin, calibration.
97
98 }
99
100 SETA OS PINOS DE LEITURA DE CORRENTE E CALIBRA O
101
102 -----
103
104 void loop()
105 {
106 {
107
108 emon1.calcVI(20,800); // Calculate all. No.of half wavelengths (
    crossings), time-out
109
110 CHAMA A BIBLIOTECA E MANDA ELA CALCULAR OS VALORES DE TENS O E CORRENTE
111
112 -----
113
114 if(millis()>15000){
115
116 MANDA IGNORAR OS 15 SEGUNDOS INICIAIS QUANDO LIGAR O ARDUINO
117
118 -----
119
120 if(WiFi.status()== WL_CONNECTED){
121
122 WiFiClient client;
123
124 HTTPClient http;
125
126 // Your Domain name with URL path or IP address with path
127
128 http.begin(client, serverName);
129
130 // Specify content-type header
131
132 http.addHeader("Content-Type", "application/x-www-form-urlencoded");
133
```

```
134 // Prepare your HTTP POST request data
135
136 String httpRequestData = "api_key=" + apiKeyValue
137
138 + "&location=" + sensorLocation + "&tensao1=" + emon1.Vrms1
139
140 + "&tensao2=" + emon1.Vrms2 + "&tensao3=" + emon1.Vrms3
141
142 + "&corrente1=" + emon1.Irms1 + "&corrente2=" + emon1.Irms1
143
144 + "&corrente3=" + emon1.Irms1 + "&potencia1=" + emon1.realPower1
145
146 + "&potencia2=" + emon1.realPower2 + "&potencia3=" + emon1.realPower3 +
147
148 + "&consumo=" + emon1.consumo + "&intervalo=" + emon1.intervalo + ";
149
150 Serial.print("httpRequestData: ");
151
152 Serial.println(httpRequestData);
153
154 int httpResponseCode = http.POST(httpRequestData);
155
156 if (httpResponseCode > 0) {
157
158   Serial.print("HTTP Response code: ");
159
160   Serial.println(httpResponseCode);
161
162 }
163
164 else {
165
166   Serial.print("Error code: ");
167
168   Serial.println(httpResponseCode);
169
170 }
171
172 // Free resources
173
174 http.end();
175
176 }
177
178 else {
179
180   Serial.println("WiFi Disconnected");
181
```

```
182 }
183
184 CONECTA NO SERVIDOR, MANDA OS VALORES CALCULADOS E DESCONECTA
185
186 -----
187
188 Serial.println("");
189
190 Serial.print("Fase1: ");
191
192 Serial.print(emon1.Vrms1);
193
194 Serial.print("/");
195
196 Serial.print(emon1.Irms1);
197
198 Serial.print("/");
199
200 Serial.print(emon1.realPower1);
201
202 Serial.println("");
203
204 Serial.print("Fase2: ");
205
206 Serial.print(emon1.Vrms2);
207
208 Serial.print("/");
209
210 Serial.print(emon1.Irms2);
211
212 Serial.print("/");
213
214 Serial.print(emon1.realPower2);
215
216 Serial.println("");
217
218 Serial.print("Fase3: ");
219
220 Serial.print(emon1.Vrms3);
221
222 Serial.print("/");
223
224 Serial.print(emon1.Irms3);
225
226 Serial.print("/");
227
228 Serial.print(emon1.realPower3);
229
```

```
230 Serial.println("");
231
232 Serial.print("Consumo: ");
233
234 Serial.print(emon1.consumo);
235
236 Serial.print("/");
237
238 Serial.print("Intervalo: ");
239
240 Serial.print(emon1.intervalo);
241
242 Serial.println("");
243
244 IMPRIME OS VALORES ENCONTRADOS NA PORTA SERIAL
245
246 }}
```

APÊNDICE K – Código EmonLib.h

```

1  // #include "WProgram.h" un-comment for use on older versions of Arduino
   IDE
2
3  #include "EmonLib.h"
4
5  #if defined(ARDUINO) && ARDUINO >= 100
6
7  #include "Arduino.h"
8
9  #else
10
11 #include "WProgram.h"
12
13 #endif
14
15 #include <MCP3008.h> //(ADICIONA A BIBLIOTECA DO MCP)
16
17 MCP3008 adc(14,13,12,15); //(PINOS DO MCP CONECTADOS NO ARDUINO)
18
19 float offsetI1 = ADC_COUNTS>>1;
20
21 float offsetV1 = ADC_COUNTS>>1;
22
23 float offsetI2 = ADC_COUNTS>>1;
24
25 float offsetV2 = ADC_COUNTS>>1;
26
27 float offsetI3 = ADC_COUNTS>>1;
28
29 float offsetV3 = ADC_COUNTS>>1;
30
31 //-----
32
33 // Sets the pins to be used for voltage and current sensors
34
35 //-----
36
37 void EnergyMonitor::voltage1(unsigned int _inPinV1, double _VCAL1,
   double _PHASECAL1)
38
39 {
40
41 inPinV1 = _inPinV1;
42

```

```
43 VCAL1 = _VCAL1;
44
45 PHASECAL1 = _PHASECAL1;
46
47 }
48
49 void EnergyMonitor::voltage2(unsigned int _inPinV2, double _VCAL2,
    double _PHASECAL2)
50
51 {
52
53 inPinV2 = _inPinV2;
54
55 VCAL2 = _VCAL2;
56
57 PHASECAL2 = _PHASECAL2;
58
59 }
60
61 void EnergyMonitor::voltage3(unsigned int _inPinV3, double _VCAL3,
    double _PHASECAL3)
62
63 {
64
65 inPinV3 = _inPinV3;
66
67 VCAL3 = _VCAL3;
68
69 PHASECAL3 = _PHASECAL3;
70
71 }
72
73 //SETA OS PINOS DE TENS 0 //-----
74
75 void EnergyMonitor::current1(unsigned int _inPinI1, double _ICAL1)
76
77 {
78
79 inPinI1 = _inPinI1;
80
81 ICAL1 = _ICAL1;
82
83 }
84
85 void EnergyMonitor::current2(unsigned int _inPinI2, double _ICAL2)
86
87 {
88
```

```
89 inPinI2 = _inPinI2;
90
91 ICAL2 = _ICAL2;
92
93 }
94
95 void EnergyMonitor::current3(unsigned int _inPinI3, double _ICAL3)
96
97 {
98
99 inPinI3 = _inPinI3;
100
101 ICAL3 = _ICAL3;
102
103 }
104
105 //SETA OS PINOS DE CORRENTE
106
107 //-----
108
109 // emon_calc procedure
110
111 // Calculates realPower, apparentPower, powerFactor, Vrms, Irms, kWh
    increment
112
113 // From a sample window of the mains AC voltage and current.
114
115 // The Sample window length is defined by the number of half wavelengths
    or crossings we choose to measure.
116
117 //-----
118
119 void EnergyMonitor::calcVI(unsigned int crossings, unsigned int timeout)
120
121 {
122
123 //FUNÇÃO QUE FAZ A LEITURA INSTANTANEA NOS PINOS E CALCULA A TENSÃO E
    CORRENTE RMS
124
125 #if defined emonTxV3
126
127 int SupplyVoltage=3300;
128
129 #else
130
131 int SupplyVoltage = readVcc();
132
133 #endif
```

```
134
135 unsigned int crossCount = 0; //Used to measure number of times threshold
    is crossed.
136
137 unsigned int numberOfSamples = 0; //This is now incremented
138
139 //-----
140
141 // 1) Waits for the waveform to be close to 'zero' (mid-scale adc) part
    in sin curve.
142
143 //-----
144
145 unsigned long start = millis(); //millis()-start makes sure it doesnt
    get stuck in the loop if there is an error.
146
147 while(1) //the while loop...
148
149 {
150
151 startV1 = adc.readADC(inPinV1)+ 527.5151; //using the voltage waveform
152
153 startV2 = adc.readADC(inPinV2)+ 527.5151;
154
155 startV3 = adc.readADC(inPinV3)+ 527.5151;
156
157 if ((startV1 < (ADC_COUNTS*0.55)) && (startV1 > (ADC_COUNTS*0.45)))
    break; //check its within range
158
159 if ((startV2 < (ADC_COUNTS*0.55)) && (startV2 > (ADC_COUNTS*0.45)))
    break;
160
161 if ((startV3 < (ADC_COUNTS*0.55)) && (startV3 > (ADC_COUNTS*0.45)))
    break;
162
163 if ((millis()-start)>timeout) break;
164
165 }
166
167 //-----
168
169 // 2) Main measurement loop
170
171 //-----
172
173 start = millis();
174
175 while ((crossCount < crossings) && ((millis()-start)<timeout))
```

```
176
177 {
178
179 numberOfSamples++; //Count number of times looped.
180
181 lastFilteredV1 = filteredV1; //Used for delay/phase compensation
182
183 lastFilteredV2 = filteredV2;
184
185 lastFilteredV3 = filteredV3;
186
187 //-----
188
189 // A) Read in raw voltage and current samples
190
191 //-----
192
193 sampleV1 = adc.readADC(inPinV1)+ 527.5151; //Read in raw voltage signal
194
195 sampleI1 = adc.readADC(inPinI1); //Read in raw current signal
196
197 sampleV2 = adc.readADC(inPinV2)+ 527.5151;
198
199 sampleI2 = adc.readADC(inPinI2);
200
201 sampleV3 = adc.readADC(inPinV3)+ 527.5151;
202
203 sampleI3 = adc.readADC(inPinI3);
204
205 //-----
206
207 // B) Apply digital low pass filters to extract the 2.5 V or 1.65 V dc
      offset,
208
209 // then subtract this - signal is now centred on 0 counts.
210
211 //-----
212
213 offsetV1 = offsetV1 + ((sampleV1-offsetV1)/1024);
214
215 filteredV1 = sampleV1 - offsetV1;
216
217 offsetV2 = offsetV2 + ((sampleV2-offsetV2)/1024);
218
219 filteredV2 = sampleV2 - offsetV2;
220
221 offsetV3 = offsetV3 + ((sampleV3-offsetV3)/1024);
222
```

```
223 filteredV3 = sampleV3 - offsetV3;
224
225 //Serial.println(offsetI);
226
227 offsetI1 = offsetI1 + ((sampleI1-offsetI1)/1024);
228
229 filteredI1 = sampleI1 - offsetI1;
230
231 offsetI2 = offsetI2 + ((sampleI2-offsetI2)/1024);
232
233 filteredI2 = sampleI2 - offsetI2;
234
235 offsetI3 = offsetI3 + ((sampleI3-offsetI3)/1024);
236
237 filteredI3 = sampleI3 - offsetI3;
238
239 //-----
240
241 // C) Root-mean-square method voltage
242
243 //-----
244
245 sqV1= filteredV1 * filteredV1; //(1) square voltage values
246
247 sumV1 += sqV1; //(2) sum
248
249 sqV2= filteredV2 * filteredV2;
250
251 sumV2 += sqV2;
252
253 sqV3= filteredV3 * filteredV3;
254
255 sumV3 += sqV3;
256
257 //-----
258
259 // D) Root-mean-square method current
260
261 //-----
262
263 sqI1 = filteredI1 * filteredI1; //(1) square current values
264
265 sumI1 += sqI1; //(2) sum
266
267 sqI2 = filteredI2 * filteredI2; //(1) square current values
268
269 sumI2 += sqI2;
270
```

```
271 sqI3 = filteredI3 * filteredI3; //(1) square current values
272
273 sumI3 += sqI3;
274
275 //-----
276
277 // E) Phase calibration
278
279 //-----
280
281 phaseShiftedV1 = lastFilteredV1 + PHASECAL1 * (filteredV1 -
      lastFilteredV1);
282
283 phaseShiftedV2 = lastFilteredV2 + PHASECAL2 * (filteredV2 -
      lastFilteredV2);
284
285 phaseShiftedV3 = lastFilteredV3 + PHASECAL3 * (filteredV3 -
      lastFilteredV3);
286
287 //-----
288
289 // F) Instantaneous power calc
290
291 //-----
292
293 instP1 = phaseShiftedV1 * filteredI1; //Instantaneous Power
294
295 sumP1 +=instP1; //Sum
296
297 instP2 = phaseShiftedV2 * filteredI2; //Instantaneous Power
298
299 sumP2 +=instP2; //Sum
300
301 instP3 = phaseShiftedV3 * filteredI3; //Instantaneous Power
302
303 sumP3 +=instP3; //Sum
304
305 //-----
306
307 // G) Find the number of times the voltage has crossed the initial
      voltage
308
309 // - every 2 crosses we will have sampled 1 wavelength
310
311 // - so this method allows us to sample an integer number of half
      wavelengths which increases accuracy
312
313 //-----
```



```
314
315 lastVCross = checkVCross;
316
317 if (sampleV1 > startV1) checkVCross = true;
318
319 else checkVCross = false;
320
321 if (numberOfSamples==1) lastVCross = checkVCross;
322
323 if (lastVCross != checkVCross) crossCount++;
324
325 }
326
327 //-----
328
329 // 3) Post loop calculations
330
331 //-----
332
333 //Calculation of the root of the mean of the voltage and current squared
    (rms)
334
335 //Calibration coefficients applied.
336
337 double V_RATIO1 = VCAL1 *((SupplyVoltage/1000.0) / (ADC_COUNTS));
338
339 double V_RATIO2 = VCAL2 *((SupplyVoltage/1000.0) / (ADC_COUNTS));
340
341 double V_RATIO3 = VCAL3 *((SupplyVoltage/1000.0) / (ADC_COUNTS));
342
343 Vrms1 = V_RATIO1 * sqrt(sumV1 / numberOfSamples);
344
345 Vrms2 = V_RATIO2 * sqrt(sumV2 / numberOfSamples);
346
347 Vrms3 = V_RATIO3 * sqrt(sumV3 / numberOfSamples);
348
349 double I_RATIO1 = ICAL1 *((SupplyVoltage/1000.0) / (ADC_COUNTS));
350
351 double I_RATIO2 = ICAL2 *((SupplyVoltage/1000.0) / (ADC_COUNTS));
352
353 double I_RATIO3 = ICAL3 *((SupplyVoltage/1000.0) / (ADC_COUNTS));
354
355 Irms1 = I_RATIO1 * sqrt(sumI1 / numberOfSamples);
356
357 Irms2 = I_RATIO2 * sqrt(sumI2 / numberOfSamples);
358
359 Irms3 = I_RATIO3 * sqrt(sumI3 / numberOfSamples);
360
```

```
361 //Calculation power values
362
363 realPower1 = Vrms1 * Irms1;
364
365 realPower2 = Vrms2 * Irms2;
366
367 realPower3 = Vrms3 * Irms3;
368
369 intervalo= (millis()-start);
370
371 consumo= (realPower1+realPower2+realPower3)*intervalo;
372
373 //Reset accumulators
374
375 sumV1 = 0;
376
377 sumV2 = 0;
378
379 sumV3 = 0;
380
381 sumI1 = 0;
382
383 sumI2 = 0;
384
385 sumI3 = 0;
386
387 sumP1 = 0;
388
389 sumP2 = 0;
390
391 sumP3 = 0;
392
393 //-----
394
395 }
396
397 double EnergyMonitor::calcIrms1(unsigned int Number_of_Samples)
398 {
399 {
400
401 #if defined emonTxV3
402
403 int SupplyVoltage=3300;
404
405 #else
406
407 int SupplyVoltage = readVcc();
408
```

```
409 #endif
410
411 for (unsigned int n = 0; n < Number_of_Samples; n++)
412 {
413 {
414 sampleI1 = adc.readADC(inPinI1);
415
416 // Digital low pass filter extracts the 2.5 V or 1.65 V dc offset,
417 // then subtract this - signal is now centered on 0 counts.
418
419 offsetI1 = (offsetI1 + (sampleI1 - offsetI1) / 1024);
420
421 filteredI1 = sampleI1 - offsetI1;
422
423 // Root-mean-square method current
424 // 1) square current values
425
426 sqI1 = filteredI1 * filteredI1;
427
428 // 2) sum
429
430 sumI1 += sqI1;
431
432 }
433
434 double I_RATIO1 = ICAL1 * ((SupplyVoltage / 1000.0) / (ADC_COUNTS));
435
436 Irms1 = I_RATIO1 * sqrt(sumI1 / Number_of_Samples);
437
438 //Reset accumulators
439
440 sumI1 = 0;
441
442 //-----
443
444 return Irms1;
445
446 }
447
448 long EnergyMonitor::readVcc() {
449
450 long result;
451
452 // FUN 0 RESPONSÁVEL POR FAZER A LEITURA
453
454
```

```
457 //not used on emonTx V3 - as Vcc is always 3.3V - eliminates bandgap
      error and need for calibration http://harizanov.com/2013/09/thoughts-
      on-avr-adc-accuracy/
458
459 #if defined(__AVR_ATmega168__) || defined(__AVR_ATmega328__) || defined
      (__AVR_ATmega328P__)
460
461 ADMUX = _BV(REFS0) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1);
462
463 #elif defined(__AVR_ATmega644__) || defined(__AVR_ATmega644P__) ||
      defined(__AVR_ATmega1284__) || defined(__AVR_ATmega1284P__)
464
465 ADMUX = _BV(REFS0) | _BV(MUX4) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1);
466
467 #elif defined(__AVR_ATmega32U4__) || defined(__AVR_ATmega1280__) ||
      defined(__AVR_ATmega2560__) || defined(__AVR_AT90USB1286__)
468
469 ADMUX = _BV(REFS0) | _BV(MUX4) | _BV(MUX3) | _BV(MUX2) | _BV(MUX1);
470
471 ADCSRB &= ~_BV(MUX5); // Without this the function always returns -1 on
      the ATmega2560 http://openenergymonitor.org/emon/node/2253#comment
      -11432
472
473 #elif defined (__AVR_ATtiny24__) || defined(__AVR_ATtiny44__) || defined
      (__AVR_ATtiny84__)
474
475 ADMUX = _BV(MUX5) | _BV(MUX0);
476
477 #elif defined (__AVR_ATtiny25__) || defined(__AVR_ATtiny45__) || defined
      (__AVR_ATtiny85__)
478
479 ADMUX = _BV(MUX3) | _BV(MUX2);
480
481 #endif
482
483 #if defined(__AVR__)
484
485 delay(2); // Wait for Vref to settle
486
487 ADCSRA |= _BV(ADSC); // Convert
488
489 while (bit_is_set(ADCSRA,ADSC));
490
491 result = ADCL;
492
493 result |= ADCH<<8;
494
```

```
495 result = READVCC_CALIBRATION_CONST / result; //1100mV*1024 ADC steps
      http://openenergymonitor.org/emon/node/1186
496
497 return result;
498
499 #elif defined(__arm__)
500
501 return (3300); //Arduino Due
502
503 #else
504
505 return (3300); //Guess that other un-supported architectures will be
      running a 3.3V!
506
507 #endif
508
509 }
```

APÊNDICE L – Conexão do Protótipo com o Banco de Dados

```

1 <?php
2
3 $servername = "localhost";
4
5 // REPLACE with your Database name
6 $dbname = "banco_de_dados";
7 // REPLACE with Database user
8 $username = "root";
9 // REPLACE with Database user password
10 $password = "";
11
12 // Keep this API Key value to be compatible with the ESP32 code provided
    in the project page.
13 // If you change this value, the ESP32 sketch needs to match
14 $api_key_value = "FzIwMP0uo0aimJ";
15
16 $api_key= $location = $tensao1 = $tensao2 = $tensao3 = $corrente1 =
    $corrente2 = $corrente3 = $potencial1 = $potencia2 = $potencia3 =
    $consumo = $intervalo = "";
17
18 if ($_SERVER["REQUEST_METHOD"] == "POST") {
19     $api_key = test_input($_POST["api_key"]);
20     if($api_key == $api_key_value) {
21         $location = test_input($_POST["location"]);
22         $tensao1 = test_input($_POST["tensao1"]);
23         $tensao2 = test_input($_POST["tensao2"]);
24         $tensao3 = test_input($_POST["tensao3"]);
25         $corrente1 = test_input($_POST["corrente1"]);
26         $corrente2 = test_input($_POST["corrente2"]);
27         $corrente3 = test_input($_POST["corrente3"]);
28         $potencial1 = test_input($_POST["potencia1"]);
29         $potencia2 = test_input($_POST["potencia2"]);
30         $potencia3 = test_input($_POST["potencia3"]);
31         $consumo = test_input($_POST["consumo"]);
32         $intervalo = test_input($_POST["intervalo"]);
33
34         // Create connection
35         $conn = new mysqli($servername, $username, $password, $dbname);
36         // Check connection
37         if ($conn->connect_error) {
38             die("Connection failed: " . $conn->connect_error);
39         }
40

```

```
41     $sql = "INSERT INTO SensorData (location, tensao1, tensao2,
         tensao3, corrente1,corrente2, corrente3, potencial,potencia2,
         potencia3,consumo, intervalo)
42     VALUES ('" . $location . "', '" . $tensao1 . "', '" . $tensao2 .
         "', '" . $tensao3 . "', '" . $corrente1 . "', '" .
         $corrente2 . "', '" . $corrente3 . "', '" . $potencial . "',
         '" . $potencia2 . "', '" . $potencia3 . "', '" . $consumo .
         "', '" . $intervalo . "')";
43
44     if ($conn->query($sql) === TRUE) {
45         echo "New record created successfully";
46     }
47     else {
48         echo "Error: " . $sql . "<br>" . $conn->error;
49     }
50
51     $conn->close();
52 }
53 else {
54     echo "Wrong API Key provided.";
55 }
56
57 }
58 else {
59     echo "No data posted with HTTP POST.";
60 }
61
62 function test_input($data) {
63     $data = trim($data);
64     $data = stripslashes($data);
65     $data = htmlspecialchars($data);
66     return $data;
67 }
```

APÊNDICE M – Página HTML para Verificar Conexão do Protótipo com o Banco de Dados

```

1 <!DOCTYPE html>
2 <html><body>
3 <?php
4
5 $servername = "localhost";
6
7 // REPLACE with your Database name
8 $dbname = "banco_de_dados";
9 // REPLACE with Database user
10 $username = "root";
11 // REPLACE with Database user password
12 $password = "";
13
14 // Create connection
15 $conn = new mysqli($servername, $username, $password, $dbname);
16 // Check connection
17 if ($conn->connect_error) {
18     die("Connection failed: " . $conn->connect_error);
19 }
20
21 $sql = "SELECT id, location, tensao1, tensao2, tensao3, corrente1,
22         corrente2, corrente3, potencia1, potencia2, potencia3, consumo,
23         intervalo, reading_time FROM SensorData ORDER BY id DESC";
24
25 echo '<table cellpadding="5" cellspacing="5">
26     <tr>
27         <td>ID</td>
28         <td>Location</td>
29         <td>Tensao 1(V)</td>
30         <td>Tensao 2(V)</td>
31         <td>Tensao 3(V)</td>
32         <td>Corrente 1(A)</td>
33         <td>Corrente 2(A)</td>
34         <td>Corrente 3(A)</td>
35         <td>Potencia 1(W)</td>
36         <td>Potencia 2(W)</td>
37         <td>Potencia 3(W)</td>
38         <td>Consumo(W.ms)</td>
39         <td>Intervalo(ms)</td>
40         <td>Timestamp</td>
41     </tr>';

```



```
41 if ($result = $conn->query($sql)) {
42     while ($row = $result->fetch_assoc()) {
43         $row_id = $row["id"];
44         $row_location = $row["location"];
45         $row_tensao1 = $row["tensao1"];
46         $row_tensao2 = $row["tensao2"];
47         $row_tensao3 = $row["tensao3"];
48         $row_corrente1 = $row["corrente1"];
49         $row_corrente2 = $row["corrente2"];
50         $row_corrente3 = $row["corrente3"];
51         $row_potencia1 = $row["potencia1"];
52         $row_potencia2 = $row["potencia2"];
53         $row_potencia3 = $row["potencia3"];
54         $row_consumo = $row["consumo"];
55         $row_intervalo = $row["intervalo"];
56         $row_reading_time = $row["reading_time"];
57         // Uncomment to set timezone to - 1 hour (you can change 1 to
           any number)
58         // $row_reading_time = date("Y-m-d H:i:s", strtotime("
           $row_reading_time - 1 hours"));
59
60         // Uncomment to set timezone to + 4 hours (you can change 4 to
           any number)
61         // $row_reading_time = date("Y-m-d H:i:s", strtotime("
           $row_reading_time + 4 hours"));
62
63         echo '<tr>
64             <td>' . $row_id . '</td>
65             <td>' . $row_location . '</td>
66             <td>' . $row_tensao1 . '</td>
67             <td>' . $row_tensao2 . '</td>
68             <td>' . $row_tensao3 . '</td>
69             <td>' . $row_corrente1 . '</td>
70             <td>' . $row_corrente2 . '</td>
71             <td>' . $row_corrente3 . '</td>
72             <td>' . $row_potencia1 . '</td>
73             <td>' . $row_potencia2 . '</td>
74             <td>' . $row_potencia3 . '</td>
75             <td>' . $row_consumo . '</td>
76             <td>' . $row_intervalo . '</td>
77             <td>' . $row_reading_time . '</td>
78         </tr>';
79     }
80     $result->free();
81 }
82
83 $conn->close();
84 ?>
```

```
85 </table>  
86 </body>  
87 </html>
```
