

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTOS ACADÊMICOS DE ELETRÔNICA E INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

ÁLVARO LUIZ SORDI FILHO

GERENCIADOR DE CONSUMO ELÉTRICO RESIDENCIAL

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2015

ÁLVARO LUIZ SORDI FILHO

GERENCIADOR DE CONSUMO ELÉTRICO RESIDENCIAL

Trabalho de conclusão de curso apresentado aos departamentos acadêmicos de Eletrônica e Informática da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de “Engenheiro de Computação”

Orientador: André Schneider de Oliveira

CURITIBA

2015

RESUMO

SORDI FILHO, Álvaro Luiz. GERENCIADOR DE CONSUMO ELÉTRICO RESIDENCIAL. 73 f. Trabalho de conclusão de curso – Departamentos acadêmicos de Eletrônica e Informática Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Curitiba, 2015.

O projeto consiste em um sistema capaz de monitorar o consumo de energia em uma residência, individualizado pelo consumo em cada tomada. O sistema é capaz de ligar ou desligar a alimentação de energia em cada uma das tomadas também de forma individual. A interface do usuário com o sistema será um website, na qual o usuário poderá verificar o consumo e programar o funcionamento de cada tomada individualmente.

Palavras-chave: Automação residencial, economia de energia, economia de energia

ABSTRACT

SORDI FILHO, Álvaro Luiz. HOME ELECTRICAL CONSUMPTION MANAGER. 73 f. Trabalho de conclusão de curso – Departamentos acadêmicos de Eletrônica e Informática Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Curitiba, 2015.

The project represents a system that is capable of monitoring the energy consumption of a house, individualized by consumption at each outlet. It's also capable of turning on or off the power supply on each of the outlets also in a individual fashion. The user interface with the system is a website, in which the user is going to be able to verify the power consumption and program the working hours of each of the power outlets.

Keywords: Home automation, energy control, energy economy

LISTA DE FIGURAS

FIGURA 1	– Geração de energia no Brasil	12
FIGURA 2	– Consumo de energia no Brasil por setores	12
FIGURA 3	– Sistema proposto	14
FIGURA 4	– Exemplo de IHD.	17
FIGURA 5	– Medidor da marca iMeter	17
FIGURA 6	– Medidores MeterPlug	18
FIGURA 7	– Casos de uso.	20
FIGURA 8	– Optoacoplador	22
FIGURA 9	– NCTR por IF do 4N25	23
FIGURA 10	– Diagrama do optoacoplador 4N25 como medidor de tensão.	24
FIGURA 11	– Calibração de um medidor de corrente.	25
FIGURA 12	– Visão geral do sistema	29
FIGURA 13	– Visão geral do interfaceamento usuário-controlador	30
FIGURA 14	– Visão geral do interfaceamento medidor-tomada	30
FIGURA 15	– Diagrama em blocos dos componentes do medidor	31
FIGURA 16	– Diagrama em blocos dos componentes do controlador	32
FIGURA 17	– Diagrama em blocos do software do controlador	32
FIGURA 18	– Arquitetura funcional do módulo medidor	33
FIGURA 19	– Arquitetura funcional do controlador	33
FIGURA 20	– Etapas de uma fonte de tensão.	34
FIGURA 21	– Esquemático de uma fonte sem transformador	36
FIGURA 22	– Esquemático do módulo medidor	37
FIGURA 23	– Topologia Ponto-Multiponto (broadcast)	39
FIGURA 24	– Estrutura de um pacote API	39
FIGURA 25	– Fluxo da informação entre medidor e controlador	40
FIGURA 26	– Pacote "Requisição de Transmissão"	41
FIGURA 27	– Estrutura de uma mensagem do sistema	42
FIGURA 28	– Máquina de estados da configuração e leitura do medidor	43
FIGURA 29	– Máquina de estados do tratamento de comandos recebidos pelo medidor	44
FIGURA 30	– Máquina de estados da montagem de mensagens pelo controlador	45
FIGURA 31	– Máquina de estados do tratamento de mensagens no controlador	45
FIGURA 32	– Fluxograma da transmissão da informação em alto nível	47
FIGURA 33	– Diagrama de sequência entre página web e Servidor	48
FIGURA 34	– Diagrama de sequência entre servidor e controlador	49
FIGURA 35	– Diagrama entidade relacionamento do banco de dados	50
FIGURA 36	– Dados do teste da medição de corrente	53
FIGURA 37	– Gráfico de consumo minuto a minuto	58
FIGURA 38	– Elementos da página web referentes ao gerenciamento do sistema	59
FIGURA 39	– PCI contendo o módulo medidor	60
FIGURA 40	– Raspberry Pi conectado ao Xbee	61
FIGURA 41	– Diagrama de funcionamento geral do sistema	68
FIGURA 42	– Arquitetura funcional	69

FIGURA 43 – Máquina de Estados do firmware do medidor	70
FIGURA 44 – Máquina de Estados do firmware do controlador	71
FIGURA 45 – Diagrama de sequência da inicialização da página web.	72
FIGURA 46 – Diagrama de sequência do tratamento pelo servidor das ações do usuário.	73

LISTA DE TABELAS

TABELA 1	–	Previsão de consumo energia em 2020 no Brasil	13
TABELA 2	–	Requisitos do sistema	21
TABELA 3	–	Alocação das funções ao Hardware e Software	33
TABELA 4	–	Relacionamento entre comandos do modo API e resultados esperados .	42
TABELA 5	–	Relação entre comandos e funções do protocolo de comunicação	43
TABELA 6	–	Dados da medição de corrente	54
TABELA 7	–	Custos dos componentes do projeto	61

LISTA DE SIGLAS

TOU	<i>Time of use</i>
IHD	<i>In home displays</i>
PLC	<i>Power Line Communication</i>
LED	<i>Light Emissor Diode</i>
CTR	<i>Current Transfer Rate</i>
IF	<i>forward input</i>
AC	<i>alternating current</i>
I/O	<i>In/Out</i>
RF	Radiofrequência
Mb/s	Megabits por segundo
SMPS	<i>Switching mode power supply</i>
API	Application Programmable Interface
JSON	<i>JavaScript Object Notation</i>
DOM	<i>Document Object Model</i>
PCI	Placa de Circuito Impresso

LISTA DE SÍMBOLOS

SUMÁRIO

1 INTRODUÇÃO	11
1.1 TEMA	11
1.2 PROBLEMAS E PREMISSAS	13
1.3 OBJETIVOS	15
1.3.1 Objetivo Geral	15
1.3.2 Objetivos Específicos	15
1.4 ESTRUTURA DO DOCUMENTO	15
2 ANÁLISE DE SOLUÇÕES ATUAIS	16
2.1 <i>IN HOME DISPLAYS</i> (IHD)	16
2.2 COMPARATIVO	18
3 AVALIAÇÃO DE CONCEITOS E TÉCNICAS NECESSÁRIOS	20
3.1 REQUISITOS DO SISTEMA	20
3.2 MEDIÇÃO DE CONSUMO	21
3.2.1 Medição da tensão	22
3.2.2 Medição da corrente	25
3.3 ESCOLHA DE PROCESSADORES	27
3.4 TÉCNICAS DE COMUNICAÇÃO	28
3.5 RESULTADOS OBTIDOS	29
4 PROJETO DO SISTEMA	31
4.0.1 Visão geral	32
4.1 FONTE DE ALIMENTAÇÃO	34
4.1.1 Hardware	36
4.2 COMUNICAÇÃO	38
4.2.1 Protocolo de comunicação	38
4.2.2 Firmware	43
4.3 INTERFACE	46
4.3.1 Fluxo da informação	46
4.3.2 Website e Servidor	47
4.3.3 Armazenamento	50
5 ENSAIOS E RESULTADOS	52
5.1 MEDIÇÃO DO CONSUMO	52
5.2 COMUNICAÇÃO MEDIDOR-CONTROLADOR	54
5.3 COMUNICAÇÃO WEBSITE-CONTROLADOR	56
5.4 COMANDOS DO USUÁRIO	58
6 CONCLUSÃO	60
6.1 DESAFIOS ENCONTRADOS	62
6.2 CONTRIBUIÇÕES DO TRABALHO	64
6.3 SUGESTÕES DE TRABALHOS FUTUROS	64
REFERÊNCIAS	66
Apêndice A – DIAGRAMA DE FUNCIONAMENTO GERAL DO SISTEMA	68
Apêndice B – ARQUITETURA FUNCIONAL	69

Apêndice C – MÁQUINA DE ESTADOS DO FIRMWARE DO MEDIDOR	70
Apêndice D – MÁQUINA DE ESTADOS DO FIRMWARE DO CONTROLADOR .	71
Apêndice E – DIAGRAMA DE SEQUÊNCIA DA INICIALIZAÇÃO DA PÁGINA	
 WEB.	72
Apêndice F – DIAGRAMA DE SEQUÊNCIA DO TRATAMENTO PELO SERVI-	
 DOR DAS AÇÕES DO USUÁRIO.	73

1 INTRODUÇÃO

1.1 TEMA

A eletricidade é fundamental para a humanidade, desde o seu descobrimento no início do século XIX ela vem otimizando tarefas e possibilitando o avanço tecnológico. Atividades simples do dia a dia, como assistir televisão, navegar na internet só são possíveis graças à energia elétrica. Além disso, fábricas, supermercados e aeroportos precisam dela para funcionar. Sem energia elétrica estaríamos vivendo anos atrasados em desenvolvimento.

A eletricidade é obtida através de outros tipos de energia, por exemplo, em uma hidrelétrica a energia potencial é transformada em energia eletromotriz pelas turbinas, e então é transportada para os consumidores na forma de energia elétrica. A energia pode ser obtida por diferentes tipos de usinas, como hidrelétricas, termelétricas, nucleares e por meio de usinas solares. No Brasil as principais responsáveis pelo fornecimento de energia são as usinas hidrelétricas, seguidas pelas termelétricas, usinas de biomassa e por último nucleares como visto na figura 1.

A energia pode ser dividida de acordo com os setores da economia que a consomem. Dados de 2007 relatam que mundialmente o setor que lidera o consumo de energia é o setor industrial consumindo cerca de 37% de toda energia gerada, seguido pelo setor de transporte (20%) tanto comercial como pessoal, e o setor residencial somado ao setor comercial de uso de energia consome cerca de 16%. Os 27% restantes são perdidos na geração e transmissão de energia (ADMINISTRATION, 2007).

No Brasil em 2007 tivemos um consumo total de aproximadamente 412TWh, sendo que destes a indústria é responsável por aproximadamente 48%, seguido pelo setor residencial com 22%, e pelos setores comercial e público com 22% ((EPE), 2008). Os 8% restantes pertencem aos setores de transportes, agropecuário e perdas de geração e transporte como mostra a figura 2.

Dados ainda apontam que entre os anos de 2002 a 2007, houve um aumento de con-

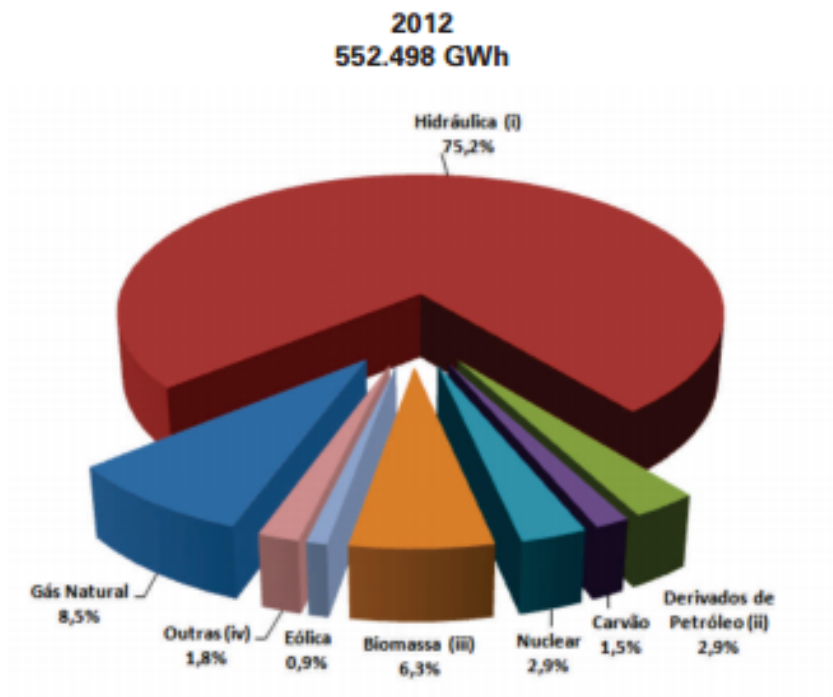


Figura 1: Geração de energia no Brasil separado por setores

Fonte: Anuário estatístico de energia elétrica 2013, página 63

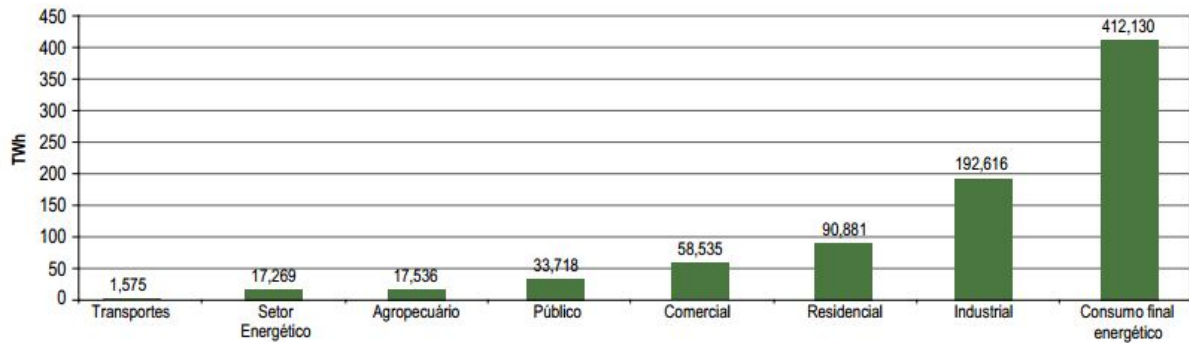


Figura 2: Consumo de energia no Brasil separado por setores

Fonte: (ELÉTRICA, 2009)

sumo de 25% pelo setor residencial, muito atribuído pelo abandono das práticas de racionamento de energia pela população. Estima-se que até o ano de 2020 o consumo de energia do Brasil deva crescer até 4,8% ao ano, sendo que o setor residencial terá um crescimento de 4,5% ao ano, e o consumo de 154kWh/mês em 2010, passe para 191kWh/mês, e que o máximo histórico de 180kWh/mês atingido em 2001 será ultrapassado por volta de 2017 ((EPE), 2011). A tabela 1 representa o consumo por setor no ano de 2010, e a previsão para 2020.

Tabela 1: Brasil - Consumo total de eletricidade por classe (TWh)

Classe	2010	2020	% ao ano
Residencial	107,2	166,9	4,5
Industrial	221,2	354,8	4,8
Comercial	69,1	123,8	6,0
Outros	59,0	84,7	3,7
Total	456,5	730,1	4,8

Fonte: ((EPE), 2011)

1.2 PROBLEMAS E PREMISSAS

Em 2001 a população brasileira na eminência de um apagão elétrico nacional, foi obrigada a adotar medidas de racionamento de energia. Dois foram os principais fatores que influenciaram no déficit de energia elétrica. O primeiro foram medidas governamentais que promoviam a privatização de empresas estatais, entre elas empresas responsáveis pela distribuição de energia. O segundo foi que em 2001, 90% da energia gerada no Brasil era proveniente de usinas hidrelétricas. Estas necessitam de chuva para manter o nível adequado de seus reservatórios para a geração de energia. Em 2001 houve uma escassez de chuva e o nível do reservatório das hidrelétricas estava baixo, conseqüentemente produzindo menos energia (VOITCH, 2009).

Como resultado o governo foi obrigado a preparar um plano de contingência, baseado no acionamento de termelétricas e no investimento em transmissão, e de novas usinas de geração de energia. Porém o que mais afetou a população foi que, a partir de 1º de Julho de 2001 cerca de 70% da população brasileira (representando o consumo mensal acima de 100kWh) deveria cortar 20% do consumo de eletricidade, caso contrário deveria pagar sobretaxa sobre o valor consumido, cujo valor era 50% ou 200% dependendo se o consumo mensal ultrapassasse 500kWh mensais. Como resultado da crise energética, segundo o Tribunal de Contas da União, houve um prejuízo de R\$ 54,2 bilhões porém trouxe conseqüências como a recuperação no setor de energia elétrica, e a criação das usinas de Santo Antônio, Jirau e Belo Monte, todas na Amazônia (VOITCH, 2009).

Atualmente o Brasil está vivenciando novamente uma crise energética, desta vez considerada a pior crise energética da história, segundo professor da Universidade Federal do Rio de Janeiro e diretor do Centro Brasileiro de Infra Estrutura, Adriano Pires (GARCIA, 2015). Entre os motivos da crise, destacam-se novamente a falta de planejamento no setor energética e a falta de chuva em âmbito nacional. Apesar do programa de bandeira tarifárias existir desde 2013, ele foi efetivado em 2015. O programa possui três tipos de bandeiras, que representam diferentes valores tarifários, separado por cores de bandeira. A bandeira verde representa que

a tarifação não terá alteração em seu valor. A amarela e vermelha representam acréscimo na tarifa, sendo de R\$1,50 a cada 100 kWh consumidos para a bandeira amarela e R\$3,00 para a vermelha (ANEEL, 2013). No mês de Janeiro de 2015, todos os subsistemas do país estiveram sob bandeira vermelha.

O problema de energia poderia ser solucionado através de duas medidas, uma geração mais eficiente e um consumo responsável. A primeira refere-se à geração de energias renováveis que causem o mínimo de impacto ambiental, como a implementação de sistemas de energia solar, eólica e a produção de biocombustível e a utilização de biomassa. A segunda diz respeito ao consumo responsável, situação em que vale a conscientização da população sobre práticas de economia de energia, e a otimização no uso de energia pelas indústrias.

Este trabalho está focado na conscientização da população, por meio da informação de como a energia utilizada em cada residência é empregada. Além de informar, o trabalho também disponibiliza formas de controlar o consumo da energia na residência, tudo isso por meio de uma interface de fácil entendimento para o usuário.

Propõe-se então, desenvolver um sistema para auxiliar o usuário no gerenciamento do consumo elétrico residencial. Este sistema é composto por medidores que, acoplados às tomadas elétricas, medem quanto está sendo consumido naquele ponto. Os valores das leituras são enviados a um controlador, responsável por disponibilizar esta informação ao usuário através de uma interface web, além de gerenciar os medidores presentes na rede. Propõe-se o uso do canal de comunicação sem fio entre medidores e controlador. A figura 3 representa, em alto nível, o sistema proposto.

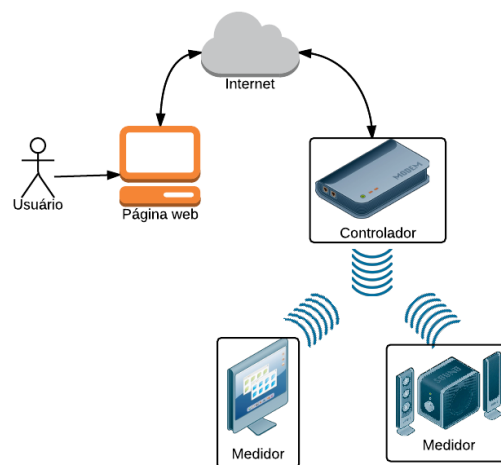


Figura 3: Sistema proposto

Fonte: Autoria própria

1.3 OBJETIVOS

1.3.1 OBJETIVO GERAL

Projetar e construir um sistema capaz de medir e gerenciar o consumo de energia elétrica residencial. Cada ponto de consumo deverá ser gerenciado de forma individual, por meio de uma interface de fácil entendimento para o usuário.

1.3.2 OBJETIVOS ESPECÍFICOS

- Analisar e comparar produtos presentes no mercado com o sistema proposto.
- Avaliar formas de medição, comunicação e disponibilização da informação.
- Projetar o sistema e seus respectivos módulos.
- Testar e validar o funcionamento geral do sistema, e individual dos seus módulos.
- Avaliar criticamente os resultados obtidos, sugerindo eventuais melhorias.

1.4 ESTRUTURA DO DOCUMENTO

O presente trabalho está estruturado em 5 capítulos, sendo que o primeiro apresenta ao leitor uma introdução ao projeto, contextualizando o tema e a proposta de trabalho. O segundo capítulo tem como objetivo analisar as soluções existentes atualmente ao problema especificado na seção 1.1. O capítulo 3 introduz as técnicas e conceitos necessários para o desenvolvimento do projeto. O projeto é dividido em submódulos, e os conceitos e técnicas necessários para cada um destes será abordado. No capítulo 4 está o projeto do sistema. Lá será mostrado como o sistema foi desenvolvido e como deve ser o seu funcionamento. Novamente o sistema será dividido em partes, cada qual com a sua função específica. O capítulo 5 objetiva por o projeto em prática, testando o funcionamento do mesmo, parte a parte e por fim a interação entre os módulos. Encerra-se com o capítulo 6 analisando se o projeto atingiu os objetivos estipulados e propondo eventuais melhorias para trabalhos futuros.

2 ANÁLISE DE SOLUÇÕES ATUAIS

2.1 *IN HOME DISPLAYS (IHD)*

Como visto no capítulo 1, o consumo de energia tende a se intensificar ao longo dos anos, não somente em nível nacional, mas também internacional. O consumo responsável no Brasil ainda é uma ideia em maturação, pois apesar de existir a tarifação por bandeiras no Brasil desde 2013, ela só foi efetivada no ano de 2015. Diferentemente da tarifação por bandeira que se aplica igualmente a todos os consumidores, em alguns países quando o consumo ultrapassa os níveis estimados, utiliza-se uma tarifação diferenciada em horários de pico, a chamada TOU (Time of use).

A medição do consumo elétrico no Brasil é feita mensalmente por meio da leitura do medidor de consumo de energia elétrica. Neste formato de medição não é possível utilizar a tarifação TOU, pois esta é uma tarifa que varia em curtos períodos de tempo, podendo ter diferentes valores em um mesmo dia. Em lugares onde a TOU é utilizada, a medição é feita por medidores chamados "Medidores de Intervalo", em que a informação é armazenada a cada meia hora e de forma totalmente eletrônica. Apesar do Brasil ainda não utilizar a tarifação TOU, com o crescimento de consumo atual de energia elétrica, é possível que muito em breve medidas semelhantes sejam adotadas.

Com o objetivo de auxiliar o usuário no gerenciamento de consumo de energia, foram desenvolvidos os IHD (In home display) que significam displays dentro de casa. Estes dispositivos são capazes de medir o consumo de energia de uma residência a qualquer momento, e disponibilizar esta informação ao usuário em tempo real através de um display. Medidores mais complexos podem além de medir o consumo de energia, traçar comparativos entre as medições atuais e as anteriores e calcular o gasto na última hora, semana ou mês considerando as diferentes tarifações da TOU (WILKENFELD, 2013). Um exemplo de IHD está na figura 4.



Figura 4: Exemplo de IHD.

Fonte: (ENERGATEINC, 2010)

Produtos disponíveis comercialmente com o objetivo de medir o consumo elétrico possuem diferentes modos de operação. O iMeter Solo (SMARTHOME, 2011) da marca INSTEON mede o consumo de energia elétrica em cada tomada da residência. Para o funcionamento dos sistemas são necessários medidores e um modem. Ambos são acoplados às tomadas elétricas da residência, enquanto os medidores realizam as medições, o modem fica responsável pelo acesso à informação coletada pelos medidores e pela disponibilização da mesma ao usuário. A comunicação entre os medidores e o modem é feita através da própria malha elétrica residencial, e a informação pode ser acessada remotamente por um computador, smartphone ou por um display de mesa da mesma marca. Além disso o produto possui mais funções que podem ser adicionadas se o sistema descrito for integrado a outros produtos de automação residencial da mesma marca. Na figura 5 é apresentado o medidor da marca iMeter.



Figura 5: Medidor da marca iMeter

Fonte: (SMARTHOME, 2011)

O MeterPlug (METERPLUG, 2013) é um exemplo de produto semelhante que possui a capacidade de medir e gerenciar o funcionamento de eletrodomésticos ligados à tomadas. Diferente de um IHD, a informação é transmitida via *bluetooth* diretamente para o aplicativo no celular, onde os dados podem ser visualizados. Outra diferença é que o produto ao invés de medir o consumo elétrico da residência, mede o consumo em cada tomada elétrica separadamente. Como função adicional, o usuário ainda pode cortar o fornecimento de energia em uma tomada a qualquer momento. Os diferentes MeterPlugs estão na figura 6.

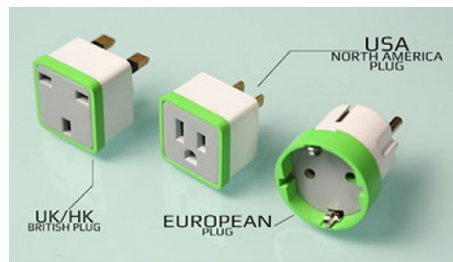


Figura 6: Medidores MeterPlug

Fonte: (METERPLUG, 2013)

2.2 COMPARATIVO

O sistema proposto busca informar ao usuário a quantidade de energia elétrica que está sendo consumida em cada uma de suas tomadas. Esta informação será disponibilizada então uma plataforma on-line, que pode ser acessada remotamente. Além de possibilitar ao usuário o corte do fornecimento de energia em uma tomada específica, ele ainda é capaz de programar o funcionamento de cada tomada para diferentes momentos do dia. Tudo isso feito através de um website.

Se comparada aos IHD's, o sistema é mais versátil pois mede o consumo em cada tomada ao invés de medir o consumo geral. O sistema, diferentemente dos IHD's, disponibiliza a informação para acesso remoto, além de fornecer funções extras.

A solução proposta faz uso de comunicação sem fio e por isto diferencia-se da marca INSTEON, que tem sua comunicação estabelecida pela instalação de cabos elétricos ao longo da residência. Esta abordagem garante que mesmo que haja uma falha na rede elétrica, a comunicação entre os dispositivos continua intacta. Em contrapartida, a comunicação via rede elétrica PLC (Power Line Communication) alcança distâncias maiores do que comunicação sem fio. Como o escopo do projeto trata de residências, considerou-se que a distância coberta pela comunicação sem fio é suficiente. O produto mais semelhante encontrado é o MeterPlug. Este

sistema não utiliza modem pois a informação é transmitida diretamente para o smartphone via *bluetooth*, e visualizada pelo aplicativo instalado. Esta abordagem possui pontos positivos e negativos. A não utilização de um modem simplifica o sistema, porém o usuário obrigatoriamente deve estar próximo ao medidor (devido ao alcance do sinal *bluetooth*). Outra vantagem do sistema proposto são as funções de gerenciamento de energia. O MeterPlug apenas informa quanta energia foi consumida, enquanto o sistema proposto oferece maior funcionalidade de controle do consumo de energia, como por exemplo a programação de faixas de horário do funcionamento de cada tomada.

3 AVALIAÇÃO DE CONCEITOS E TÉCNICAS NECESSÁRIOS

Neste capítulo serão definidos os requisitos do projeto e suas funcionalidades. Então será feito um estudo dos conceitos necessários para a medição do consumo de uma tomada elétrica. Por último, de como transmitir a informação referente ao consumo medido para o usuário.

3.1 REQUISITOS DO SISTEMA

Para a definição dos requisitos do sistema, foi primeiramente criado o diagrama de casos de uso do projeto, presente na figura 7. À partir dos casos de uso obtém-se as possíveis interações do usuário com o sistema e então definem-se quais seus requisitos.

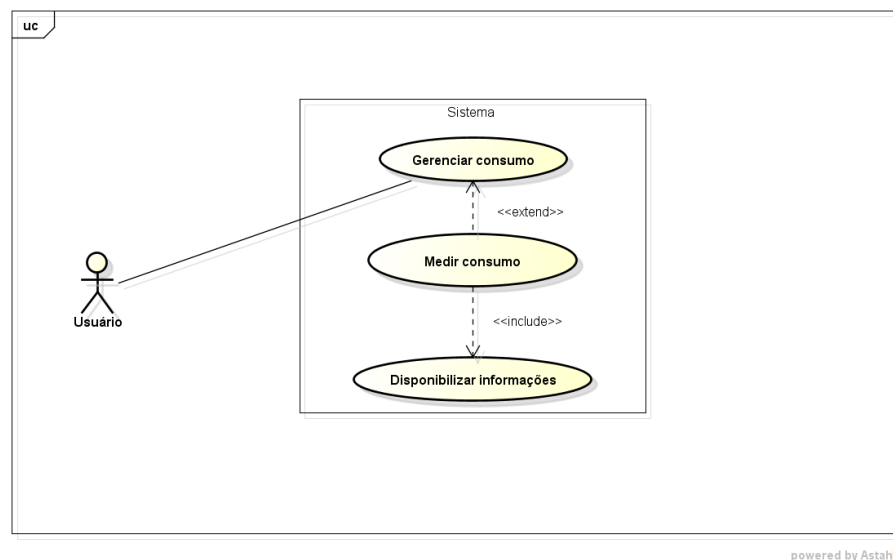


Figura 7: Casos de uso.

Fonte: autoria própria

Tendo definido os requisitos do sistema, o restante do capítulo descreve quais os conceitos e técnicas considerados necessários para atingir os requisitos. Inicialmente será dado

Tabela 2: Requisitos do sistema

Requisito	Descrição
1	O sistema deverá possuir um dispositivo (denominado medidor ou módulo medidor) para analisar o consumo de energia.
1.1	O dispositivo deve ser acoplado a uma tomada elétrica residencial, semelhante a um adaptador.
2	O sistema deve possibilitar que os dados coletados sejam acessados pelo usuário remotamente.
3	O usuário deve ser capaz de gerenciar o consumo elétrico em cada um dos medidores de forma individual.
3.1	O usuário deve ser capaz de ativar/desativar o acesso a energia em um medidor.
3.2	O usuário deve ser capaz de programar faixas de horário de funcionamento para cada medidor.
3.3	O usuário deve ser capaz de visualizar o estado e informações de consumo de cada medidor.

Fonte: autoria própria

enfoque aos módulos medidores de consumo, seguido pelas técnicas de disponibilização da informação ao usuário e finalmente uma visão geral do funcionamento sistema de forma geral e individualizada de seus componentes.

3.2 MEDIÇÃO DE CONSUMO

Para medir o consumo, é preciso primeiro entender qual a tensão da rede elétrica de uma residência. O sistema elétrico brasileiro é composto de várias etapas. Desde produção, passando por transmissão e distribuição. A parte importante para o projeto é referente a distribuição, mais especificamente quando a subestação abaixadora baixa a tensão alternada da ordem de dezenas de kV para a tensão residencial que pode ser 380/220V e 220/127V para sistemas trifásicos ou 220/110V para sistemas monofásicos com tape, ambas alternadas (NISKIER, 2014).

O consumo é calculado por kWh, que representa a quantidade de energia consumida por um aparelho em um determinado período de tempo. Por exemplo, um aparelho que tem potência de 1W, se ligado por 1 hora irá consumir 1Wh. O sistema deve então ser capaz de medir a potência da carga ligada à cada tomada residencial. Para medir o consumo de energia em termos de kWh, é preciso saber a tensão fornecida pela tomada, a corrente que está passando sobre a carga e o período de duração desta corrente, ou seja, calcular a integral da corrente em relação ao tempo, pois a tensão é constante.

3.2.1 MEDIÇÃO DA TENSÃO

Como dito, a tensão é uma das variáveis necessárias para o cálculo da potência, sendo corrente a outra. A tensão a ser detectada está na faixa entre 127 a 380V. Em Curitiba a tensão da rede elétrica residencial é geralmente 127V(uma fase e um neutro) ou 220V(duas fases).

A primeira e mais simples solução para detectar o valor da tensão é por meio de um divisor de tensão resistivo. O divisor tem o papel de reduzir a tensão de entrada para um valor que possa ser lido pela porta analógica do processador (comumente entre 0 a 5V). Apesar de reduzir a tensão para o valor desejado algumas características da rede elétrica limitam o uso do divisor resistivo. A tensão analisada será de no máximo 380V e deve ser reduzida para 5V, sendo assim a razão entre os resistores utilizados no divisor deve ser de 76 para 1. Se utilizarmos resistores de 1k e 75k a potência dissipada no resistor de 75k será de 2W aproximadamente. Isto faz com que, além do consumo de energia desnecessário, o tamanho do resistor seja grande para o projeto e portanto esta solução não pode ser utilizada. Outro problema que deve ser considerado ao utilizar um divisor resistivo é a proteção para a porta analógica do processador em caso de sobretensão.

A melhor solução encontrada ao problema baseia-se no uso de um optoacoplador ou optoisolador. O optoacoplador é um componente eletrônico que transfere energia elétrica entre dois circuitos isolados utilizando a luz (PEASE, 1991). Ele é capaz de em um lado do circuito suportar tensões da grandeza de kV enquanto do outro esta tensão pode ser reduzida para qualquer valor desejado, incluindo 5V. O esquemático de um optoacoplador está presente na figura 8.

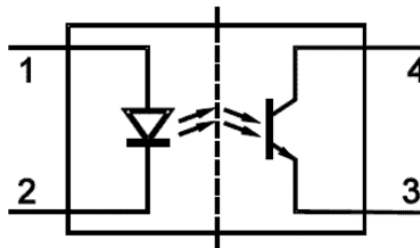


Figura 8: Optoacoplador

Fonte: (ANALOGDEVICES, 2013)

Do lado esquerdo do esquemático a tensão presente é a da rede elétrica residencial, que como visto, pode chegar até 380V enquanto do lado direito está presente a tensão medida pelo microcontrolador, e tem valor máximo de 5V. O funcionamento do optoacoplador está estrita-

mente ligado a relação entre o LED(Light Emissor Diode) do lado esquerdo e o fototransistor do lado direito. O transistor do lado direito é acionado quando o LED do lado esquerdo é ativado (existe corrente no lado esquerdo). Desta forma o optoacoplador funciona como uma chave. Quando existe tensão sobre o LED, e conseqüentemente uma corrente, o transistor é saturado e a corrente passa por ele. Quando não há tensão, o transistor (a chave) está cortado e não há corrente passando por ele. Desta forma, o optoacoplador pode ser usado como um sensor de presença ou ausência de tensão.

Para termos a capacidade de medir a tensão do lado esquerdo e não somente saber se há ou não tensão, o LED é operado em uma faixa reduzida da recomendada pelo datasheet. No caso do optoacoplador 4N25, garante-se um estado totalmente saturado da chave se a corrente que passa pelo LED for de no mínimo 10mA. Para este projeto é preciso que o transistor trabalhe como amplificador, isto é, na sua região linear, e não em estado de saturação. Com o transistor trabalhando como amplificador, a medida que a corrente sobre o LED aumentar (devido o aumento da tensão), a corrente de saída também aumenta.

De acordo com a figura 9 retirada do datasheet do 4N25 o transistor possui diferentes valores de CTR(Current Transfer Rate) de acordo com a corrente I_F (forward input) sobre o LED. Baseando-se neste gráfico, podemos calcular a corrente de saída para diferentes valores de corrente de entrada, uma vez que o valor da corrente de saída no emissor será igual a corrente do coletor multiplicada pelo fator CTR, e dessa forma, dimensionar o valor dos resistores utilizados para o projeto.

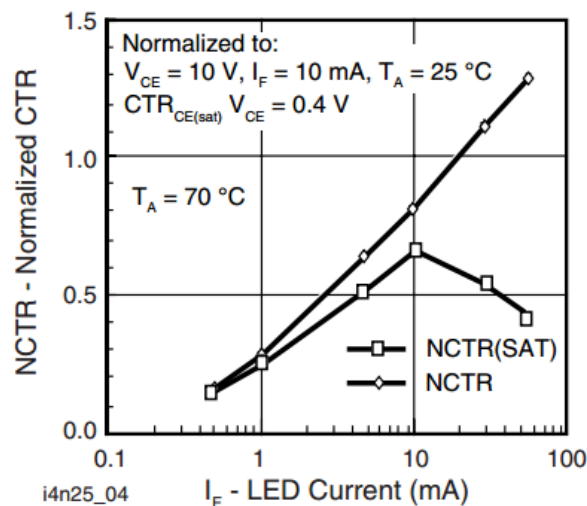


Figura 9: NCTR por I_F

Fonte: (VISHAY, 2004)

O diagrama da figura 10 representa o optoacoplador 4N25 sendo utilizado como medidor de tensão. Com o resistor de 220kΩ na entrada, obtém-se uma corrente que varia entre aproximadamente 0,5mA e 1,8mA para valores de tensão entre 127 e 380V. Experimentalmente observamos que, se uma tensão de 127V for aplicada na entrada, temos como saída uma tensão de 1,6V aproximadamente, e se a tensão de entrada for de 220V, temos 2,6V na saída. Estes valores condizem com o gráfico da figura 9, pois se aplicada uma tensão alternada de 127V na entrada, temos como corrente de pico:

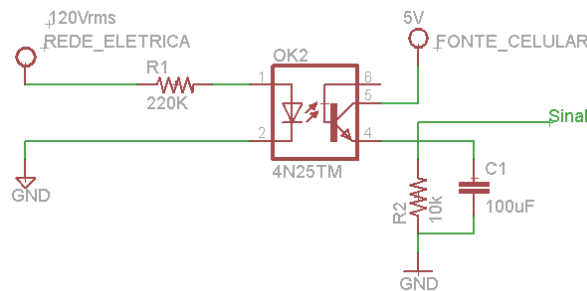


Figura 10: Optoacoplador 4N25 como medidor de tensão.

Fonte: Autoria própria

$$\frac{127 \times \sqrt{2}}{220000} = 0,817mA$$

De acordo com o gráfico, a corrente de saída será aproximadamente este valor multiplicado por 0,2, ou seja:

$$0,817mA \times 0,2 = 0,164mA$$

Esta corrente passando sobre um resistor de 10kΩ equivale a uma tensão de 1,6V, conforme medida experimentalmente. O mesmo cálculo pode ser feito para uma tensão de 220V aplicada e será obtido o valor de 2,8V.

Contudo a tensão obtida na saída é uma tensão contínua pulsante, ou seja, possui apenas os valores positivos da onda senoidal, diferente da tensão de entrada. Para filtrar esta tensão e obtermos seu valor máximo, foi utilizado um capacitor em paralelo. Este capacitor tem o papel de manter o nível de saída constante, e seu valor varia de acordo com a tensão de pico, da meia-onda retificada, aplicada sobre o mesmo.

Com este sistema, é possível diferenciar entre os possíveis valores de tensão de uma rede elétrica residencial. Os valores de tensão alternada são reduzidos e convertidos em uma

tensão contínua na faixa entre 0 e 5V, que é então lida pelo microprocessador através de um conversor A/D e interpretada.

3.2.2 MEDIÇÃO DA CORRENTE

Com o valor da tensão obtido, precisamos apenas do valor da corrente para sabermos a potência consumida pela carga. Para medir a corrente utilizada pela carga, foi utilizado um sensor de não intrusivo, modelo TA12-100. Este é um sensor de corrente não intrusivo que permite medir a corrente em uma rede AC (alternating current) apenas envolvendo o condutor elétrico, sem contato direto com a fiação elétrica. O sensor gera como saída uma corrente proporcional à corrente medida. No caso do TA12-100, ele mede até 5A e como saída fornece uma corrente de até 5mA. O sensor possui um resistor que tem o papel de transformar a corrente de saída em uma tensão, a ser lida pelo microcontrolador.

A tensão de saída será de formato senoidal, pois, a corrente que está sendo analisada é uma corrente alternada. A figura 11 representa o funcionamento de um sensor de efeito hall. Nela também estão presentes o resistor de carga e os resistores R1 e R2 utilizados para que a fase negativa da onda possa ser analisada.

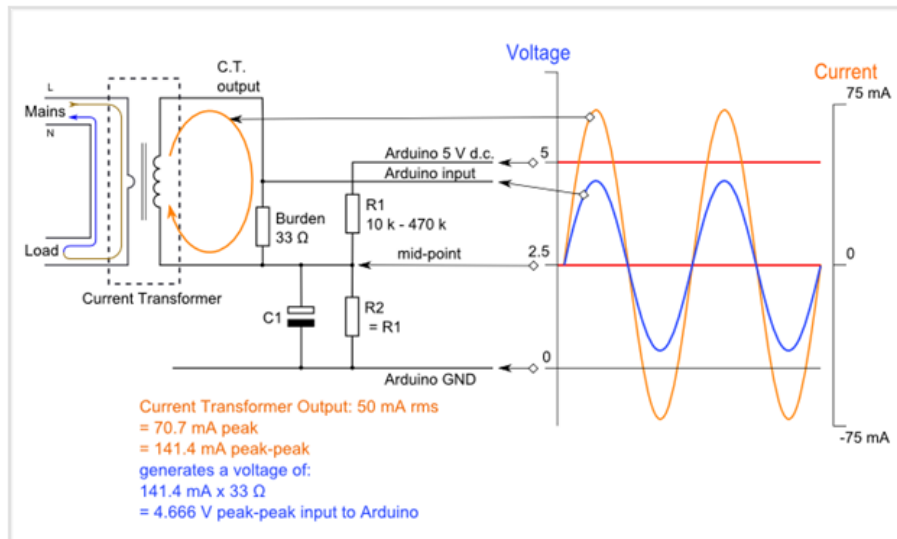


Figura 11: Calibração de um medidor de corrente.

Fonte: (OPENENERGYMONITOR, 2012)

No caso do TA12-100 ele já possui um resistor de carga embutido de 200Ω, tendo um valor de tensão em seu pino S (indicado como *mid-point* na figura 11 que é medido diretamente pelo microcontrolador. Como visto no datasheet do componente (ELECTFREAKS,

2013), o valor de saída varia entre 0V - 1V, sendo que 0 representa a ausência de corrente e 1V representa corrente de 5A. Sendo assim, a sensibilidade da leitura, devido a resolução de 10 bits do conversor A/D, é de 17mA, ou seja, haverá variação no resultado da leitura somente a cada 17mA. Não foram utilizados os resistores R1 e R2, pois, para sabermos quanta corrente está sendo utilizada precisamos apenas de uma das fases, neste caso é medida a positiva.

Como já mencionado, a tensão de saída do sensor de corrente varia entre 0 e 1V. Esta tensão será convertida para termos o valor da corrente sendo utilizada pela carga. A fórmula usada na conversão do valor da tensão lido pelo microcontrolador, para o valor da corrente medida é:

$$\textit{amplitude da corrente} = \frac{\textit{valor do sensor} \times 5 \times 1000}{1024 \times 210} \textit{mA}$$

$$\textit{correnteRMS} = \frac{\textit{amplitude da corrente}}{\sqrt{2}} \textit{mA}$$

onde:

- valor do sensor: é o valor máximo entre 1000 leituras consecutivas do conversor AD. A quantidade de leituras consecutivas pode ser diminuída, porém sugere-se que sejam feitas, no mínimo, 200 leituras consecutivas;
- 5: é o valor da razão entre a tensão de alimentação do processador e a tensão máxima lida do TA12-100;
- 1024: corresponde à resolução do conversor AD de 10 bits
- 1000: razão da conversão entre a corrente analisada pela corrente gerada do TA12-100;
- 210: valor do resistor de carga do TA12-100.
- $\sqrt{2}$: razão da conversão para corrente efetiva.

Como o consumo deve ser medido em kWh, precisamos além dos valores de tensão e corrente, o tempo que estes valores mantiveram-se os mesmos. Considerando que a tensão se manterá a mesma, a medição da corrente e conseqüentemente o consumo, devem ser atualizados periodicamente. Esta informação deve estar disponível ao usuário quando requisitada. Decidiu-se então que os módulos medidores após a medição deverão informar uma unidade central dos valores medidos, que por sua vez será responsável por disponibilizar ao usuário estes dados obtidos.

3.3 ESCOLHA DE PROCESSADORES

O controlador é o componente central do sistema. Ele é o componente que irá disponibilizar as ferramentas de gerenciamento para o usuário, além de receber informações dos medidores e enviar comandos a eles. Por isso este controlador deve ter uma capacidade de processamento suficiente para tratar múltiplas requisições e ainda ser flexível para facilitar a interação entre diferentes componente e possivelmente diferentes linguagens de programação. Além disso o seu tamanho deve ser reduzido. Entre as opções analisadas estão um computador de mesa (*desktop*), o kit ARM LPC1768, e o Raspberry PI.

O computador de mesa apesar de ser a opção mais flexível, é a mais custosa em termos financeiros, além de ocupar um grande espaço. Esta opção foi descartada, pois, as outras duas opções apresentam características mais atraentes em relação a custo e tamanho.

O kit ARM LPC1768 possui vantagens em questão de tamanho e custo se comparado à alternativa anterior. O kit pode ser programado tanto em assembly como em C. Além disso possui vários pinos de I/O(In/Out) para o interfaceamento com componentes de comunicação e sua capacidade de processamento é mais que suficiente, graças ao seu procesasdor Cortex-M3 capaz de funcionar a 100 MHz (NXP, 2014).

Semelhante o kit LPC1768, o Raspberry Pi possui um processador ARM1176JZF-S de 700MHz (BRADBURY, 2014). Além disso possui GPU VideoCore, 512MB de memória RAM, entrada HDMI e espaço para cartão de expansão SD. Como o Raspberry Pi foi desenvolvido para fins de ensino de ciências da computação, a existência de uma comunidade auxilia no desenvolvimento do projeto. O kit é extremamente flexível sendo capaz de operar com uma grande quantidade de distribuições Linux. Esta foi a solução escolhida para o projeto, pois possui características positivas das duas opções anteriores. É flexível como um computador de mesa por ser capaz de rodar uma distribuição linux, e possui pinos de I/O como o kit LPC1768 para o interfaceamento com os componentes de comunicação. Além disso, o componente já havia sido adquirido e trabalhado em projetos anteriores.

Para o módulo medidor também é necessário um processador, para receber os dados dos medidores de tensão e corrente e cumprir a função de gerenciador de consumo e de comunicação com o controlador. Para esta função foi escolhido o Arduino Pro Mini. Ele é uma placa baseada no chip ATmega168, com 14 pinos de I/O digital, e 8 analógicos. Possui velocidade de 8MHz para os modelos de 3.3V e 16MHz para os modelos de 5V. Além disso possui pinos dedicados para a comunicação serial. Pela placa na qual vem montado ser pequena (aproximadamente 1,8cm x 3,3cm) e ser facilmente programado quando utilizando a linguagem

Wiring, esta foi a opção escolhida para o projeto.

3.4 TÉCNICAS DE COMUNICAÇÃO

Conforme definido no capítulo 2, o meio de comunicação entre os medidores e o controlador é *wireless*. Sendo assim, foram estudadas 3 possibilidades: transmissor MX-FS-03V e receptor MX-05V, *bluetooth*, e o protocolo e equipamentos zigbee.

A primeira alternativa é a mais econômica em termos financeiros e possuem um alcance suficiente para os fins do projeto. MX-FS-03V e MX-05V são componentes simples de RF(Radiofrequência), e transmitem informação de forma serial entre o transmissor(MX-FS-03V) e o receptor(MX-05V). Como a comunicação do sistema proposto assemelha-se à uma topologia de rede estrela, onde uma unidade (controlador) é responsável por receber e enviar dados para os nós (medidores) , os módulos RF foram descartados, pois, apresentam dificuldades na implementação deste protocolo.

O *bluetooth* é uma solução para a comunicação utilizada por vários dispositivos. É um padrão global de comunicação sem fio que tem o alcance máximo de 100 metros quando o dispositivo é da classe 1, e a velocidade de transferência pode chegar até 24 Mb/s(Megabits por segundo) na versão 3.0 (BAKKER, 2002). A topologia de uma rede *bluetooth* é denominada *piconet*. Nesta topologia podem existir até 8 integrantes, sendo que somente um deles é o mestre e os outros escravos. Se mais de 8 elementos devem ser conectados, mais de uma *piconet* deve estar interligada, formando uma *scatternet*. Uma *scatternet* é um número de *piconets* interconectados que suporta a comunicação entre mais de 8 integrantes. Para se formarem é preciso que um membro da *piconet* (mestre ou escravo) escolha participar em outra *piconet* como escravo. Este integrante é capaz então, de transmitir informação entre as *piconets*. Sendo assim é possível utilizar o bluetooth no projeto.

A última opção é a utilização de dispositivos de radiofrequência baseados no conjunto de especificações zigbee. Esta tecnologia é comparável a tecnologia bluetooth, porém possui um menor consumo, distância de alcance equivalente à WiFi(cerca de 100 metros) e é capaz de organizar redes em diversos tipos de topologias. Possui ainda fácil configuração e a comunicação entre processador e módulo de comunicação é feita de modo serial, o que facilita na etapa de programação. Em contrapartida o custo é o mais alto dentre as opções, porém o fato de facilitar a implementação e conhecimento prévio de trabalho com módulos xbee de projetos anteriores, esta foi a opção escolhida.

Por último foi definido qual seria a interface para o usuário. Esta escolha está intima-

mente ligada a escolha do controlador, uma vez que ele é o responsável por esta tarefa. O fato do Raspberry Pi ser capaz de trabalhar com uma distribuição linux, e na busca de uma interface acessível remotamente e por diferentes plataformas, escolheu-se desenvolver um website para que o usuário possa gerenciar o sistema.

3.5 RESULTADOS OBTIDOS

Como resultado da etapa de estudos, foi possível definir todos os componentes do projeto. Além disso, devido aos componentes escolhidos, tem-se uma idéia inicial de como eles irão se relacionar. Em uma visão geral, o sistema foi definido de acordo com a figura 12.

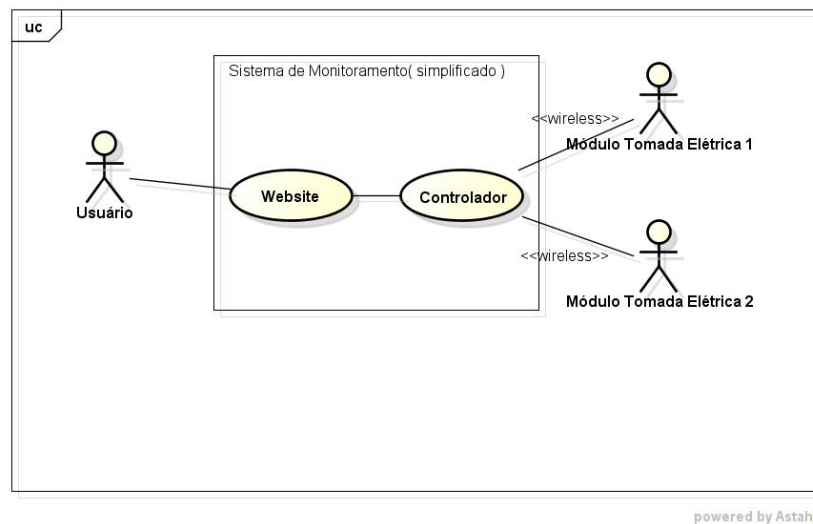


Figura 12: Visão geral do sistema

Fonte: autoria própria

Na figura 13 está representada a interação entre o usuário e o controlador, bem como os componentes principais do controlador responsáveis pelo interfaceamento usuário-medidores. Já na figura 14 está representado o interfaceamento entre o módulo medidor com a tomada elétrica e o aparelho consumidor de energia.

No Apêndice A representa o sistema em uma visão simplificada após a etapa de avaliação de componentes. Nesta figura estão presentes os componentes do projeto (figura 13 e 14) e a relação entre eles de forma simplificada. Este diagrama tem como objetivo apresentar de forma simples, o funcionamento intra e inter-componentes.

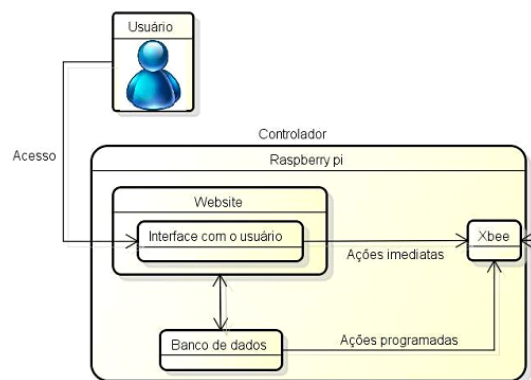


Figura 13: Visão geral do interfaceamento usuário-controlador

Fonte: autoria própria

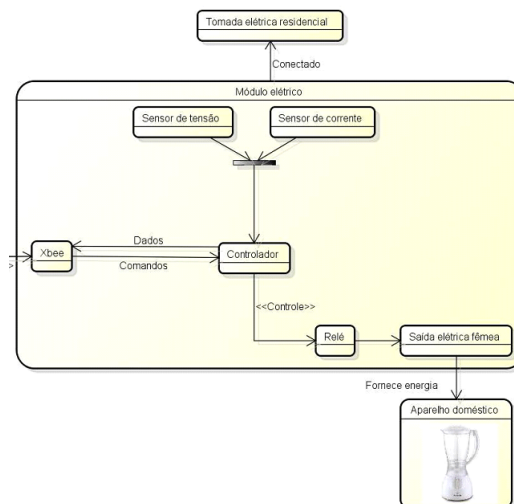


Figura 14: Visão geral do interfaceamento medidor-tomada

Fonte: autoria própria

4 PROJETO DO SISTEMA

Este capítulo será dedicado ao projeto dos módulos do sistema. Será utilizado o conhecimento descrito no capítulo 2 juntamente com o conhecimento adquirido ao longo do curso para esquematizar o funcionamento dos módulos e posteriormente do sistema.

Inicialmente será abordado o desenvolvimento do módulo medidor de consumo. Em seguida serão abordados de forma paralela o protocolo de comunicação entre os medidores e o controlador e as funcionalidades do sistema. Por último serão apresentados todos os módulos envolvidos no a interfaceamento com o usuário.

A figura 15 representa o diagrama em blocos dos componentes utilizados no medidor. Além de indicar os componentes utilizados, também esta presente a relação entre estes em alto nível.

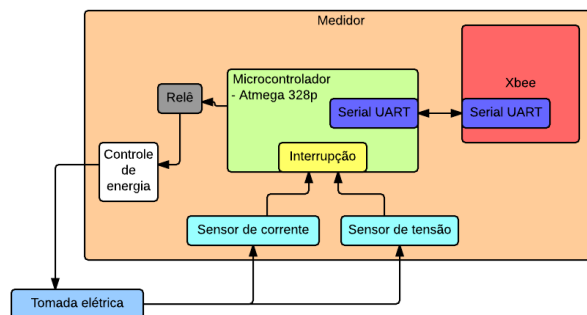


Figura 15: Diagrama em blocos dos componentes do medidor

Fonte: autoria própria

Já a figura 16 representa o diagrama em bloco dos componentes utilizados no controlador. A quantidade de componentes físicos utilizadas é bem menor se comparado ao medidor. A figura 17 representa os blocos de software que, executados pelo controlador, fornecem o suporte ao funcionamento geral do sistema.

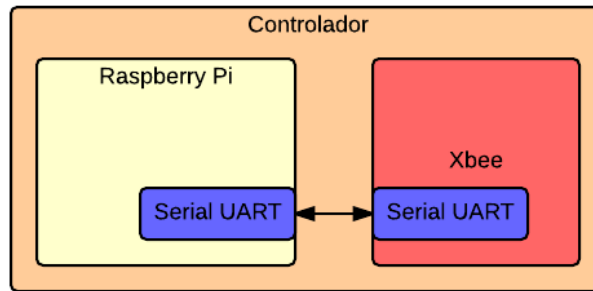


Figura 16: Diagrama em blocos dos componentes do controlador

Fonte: autoria própria

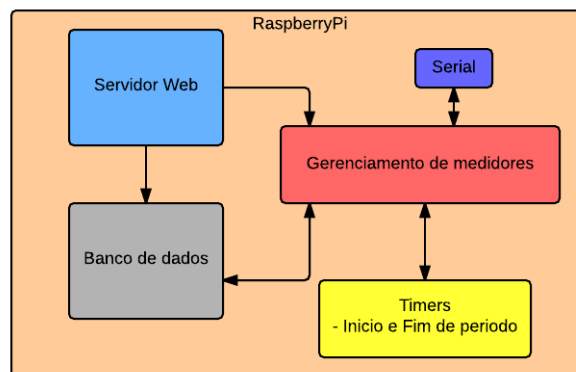


Figura 17: Diagrama em blocos do software do controlador

Fonte: autoria própria

4.0.1 VISÃO GERAL

Definidos os componentes que serão utilizados em cada parte do sistema, foi definida a arquitetura funcional do sistema. Os diagramas das figuras 18 e 19 representam, respectivamente, a arquitetura funcional do módulo medidor e do controlador. O diagrama completo está descrito no Apêndice B. As caixas representam as funções do sistema e as retas apontam para o sentido que estas funções são executadas. A relação entre as funções e os requisitos levantados será dada em seguida.

Elementos que possuem «PERIFÉRICO» como estereótipo representam hardware, e deverão fazer o interfaceamento do sistema com o usuário e com os outros periféricos. Elementos que possuem «INTERFACE» fazem referência a interface com o usuário.

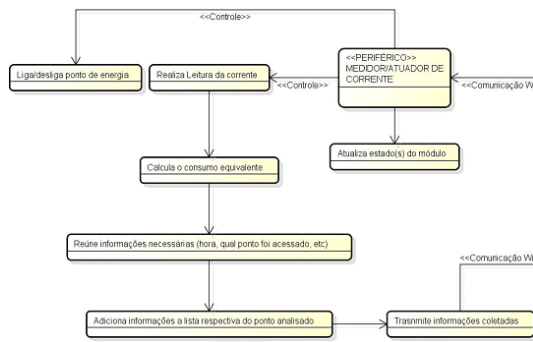


Figura 18: Arquitetura funcional do módulo medidor

Fonte: autoria própria

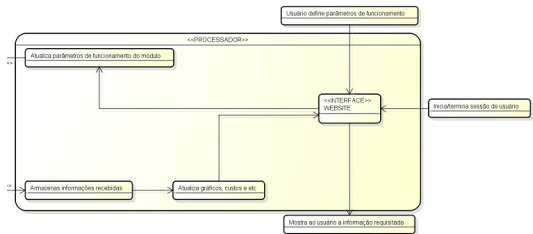


Figura 19: Arquitetura funcional do controlador

Fonte: autoria própria

A partir da arquitetura funcional, é possível alocar os requisitos descritos no capítulo 3 com os componentes presentes nos diagramas das figuras 18 e 19. Como resultado temos a tabela 3.

Tabela 3: Alocação das funções ao Hardware e Software

HW/SW	Requisito atendido	Função
HW	1, 1.1	MEDIDOR/ATUADOR DE CORRENTE
SW	2	Mostrar ao usuário pela interface os dados obtidos por meio de gráficos, etc.
SW	3	INTERFACE WEBSITE
HW/SW	3.1	Liga/desliga ponto de energia, WEBSITE
HW/SW	3.2	Usuário define parâmetros de funcionamento, WEBSITE
HW/SW	3.3	Atualiza estado(s) do módulo, WEBSITE

Fonte: autoria própria

4.1 FONTE DE ALIMENTAÇÃO

O módulo medidor é composto por vários componentes eletrônicos. Estes componentes requerem alimentação de baixa tensão em corrente contínua para operar. O sistema requer pelo menos 210mA de corrente para funcionar, uma vez que cada pino de I/O do arduino utiliza 40mA de corrente (ARDUINO, 2008) totalizando 160mA (2 pinos para a comunicação serial, um para o sinal de tensão e um para o sinal de corrente), e o Xbee requer 50mA (DIGI, 2015). Dessa forma como os componentes utilizados precisam de 5V ou 3.3V para alimentação, é necessário uma fonte de tensão DC regulada no valor de 5V que forneça pelo menos 250mA de corrente. A primeira opção foi de utilizar uma bateria 9V para a alimentação. Esta opção foi rapidamente descartada pois o usuário deveria trocar a bateria sempre que a mesma estivesse sem energia. Outro fator relevante foi a disponibilidade de uma fonte elétrica residencial, porém com uma tensão elevada e alternada. Decidiu-se por utilizar uma fonte de alimentação que transformasse a tensão da tomada elétrica em 5V DC.

Uma fonte de tensão é composta por 4 etapas. A primeira é a de redução da tensão da rede para uma tensão menor e fazer o isolamento galvânico entre a rede e a carga por medidas de segurança. O principal componente nesta etapa é o transformador. A segunda é a de retificar o sinal, ou seja, transformar o sinal alternado em um sinal pulsante unidirecional. Nesta etapa o principal componente é a ponte retificadora, normalmente composta por 4 diodos. A terceira etapa é a de filtragem, normalmente composta por capacitores, que irão transformar o sinal pulsante em uma forma mais linear que servirá de entrada para a quarta etapa, a de regulagem, que transformará este sinal filtrado em uma tensão contínua no valor desejado pela fonte. Nesta última etapa podem ser utilizados diodos zener ou reguladores de tensão comerciais (CAPELLI, 2006), cada um com sua limitação. Na figura 20 é apresentado um diagrama exemplificando as etapas descritas.

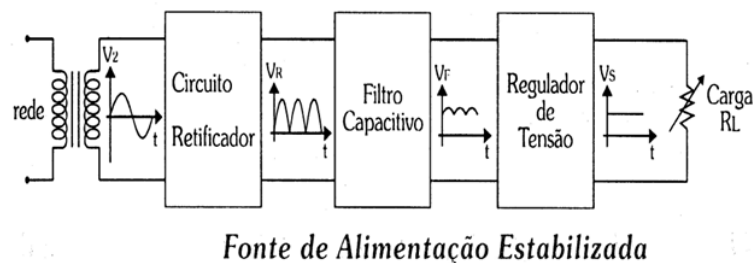


Figura 20: Etapas de uma fonte de tensão.

Fonte: (FONTES, 2006)

Fontes reguladoras de tensão que utilizam transformador são capazes de alcançar correntes altas, muito acima do necessário pelo projeto. Apesar de fornecerem corrente mais que suficiente, as fontes com transformadores ocupam um volume muito grande se comparado com o resto do circuito. Outro fator negativo destas fontes está relacionado ao regulador de tensão.

O regulador é o elemento responsável por reduzir a tensão de saída da etapa de retificação para uma tensão menor aceitável pelo sistema (no caso 5V e posteriormente 3,3V). Dependendo da corrente que passará pelo regulador e da diferença de tensão de entrada e saída do mesmo, ele pode superaquecer, o que torna o uso de um dissipador de calor obrigatório. Porém ao utilizar um dissipador de calor voltamos ao problema inicial de volume físico. Quanto maior a quantidade de calor gerada por um componente, maior deve ser o tamanho do dissipador, com o fim de manter a temperatura do componente controlada (neste caso o regulador de tensão). Sendo assim, o uso de uma fonte com transformador foi descartada.

A segunda alternativa é a de construir uma fonte sem transformador. Este tipo de fonte tem tamanho pequeno se comparado ao circuito, porém corrente reduzida. A corrente que esta fonte disponibiliza aumenta de acordo com o tamanho do capacitor usado, (capacitor C2 da figura 21) porém a corrente máxima fornecida pode chegar a 150mA. Esta opção foi descartada pois apesar do tamanho ser reduzido, a corrente fornecida não é suficiente para atingir os requisitos do projeto, uma vez que uma fonte sem transformador fornece no máximo 150mA.

A última opção seria a de desenvolver uma fonte SMPS (Switching mode power supply). A fonte SMPS é uma fonte que utiliza um transistor para fazer o chaveamento da tensão. Esta fonte pode ser encontrada em diversos equipamentos elétricos, principalmente carregadores portáteis. O chaveamento de uma fonte SMPS garante que a dissipação de energia será mínima, ou seja, a eficiência dela é maior (o que não gera calor suficiente para necessitar de um dissipador de calor). O transistor responsável por este chaveamento em totalmente cortado ou totalmente saturado é controlado (geralmente por um CI dedicado) de tal forma que quando uma carga demande mais energia (e conseqüentemente a tensão na saída caia), o controle altera a duração do estado totalmente saturado do transistor ajustando a quantidade de tensão fornecida, mantendo assim o valor de saída desejado (para este projeto 5V).

Este tipo de fonte possui todas as características desejadas pois é pequeno e fornece corrente acima do necessário (pode fornecer entre 0,5 a 1A). Porém a dificuldade de encontrar os componentes, vendidos comercialmente, para implementá-la impossibilita que ela seja desenvolvida, e por isso uma fonte comercial genérica utilizada como carregador de celular foi empregada.

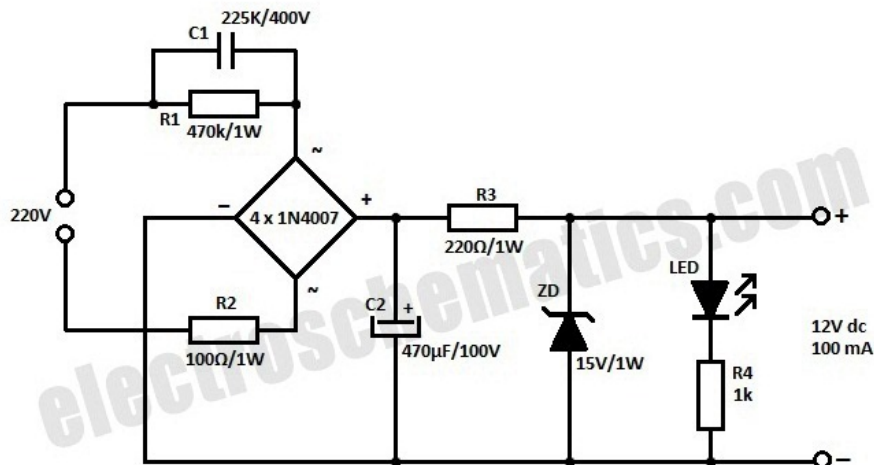


Figura 21: Esquemático de uma fonte sem transformador

Fonte: (MICROCHIP, 2009)

4.1.1 HARDWARE

Nesta seção será projetada a interação entre os componentes eletrônicos utilizados na medição do consumo e o microcontrolador responsável. Serão utilizados dois pinos analógicos e quatro pinos digitais de I/O para o projeto. O esquemático da ligação dos componentes está representado na figura 22.

Para medir a tensão foram utilizados o optoacoplador 4N25, os resistores R1 e R2 o capacitor C1, uma fonte de 5V e a entrada analógica A0 do microcontrolador. Conforme observado na figura 22, a ligação dos pinos 1 e 2 do optoacoplador é feita em paralelo à carga a ser medida pelo módulo. Isto garante que a corrente utilizada pela carga não danifique o LED do 4N25 pois pode ser muito mais alta que o suportado pelo LED, além de cumprir o papel de medir a mesma tensão utilizada pela carga. O resistor R1 limita a corrente para no máximo 1,72mA no casos onde a tensão da tomada é de 280V e no mínimo 0,5mA para tensões de 110V. Este resistor é necessário para manter a corrente dentro da faixa na qual o transístor trabalha como amplificador, na qual pode-se utilizar o optoacoplador como medidor de tensão conforme discutido no capítulo 2.

Nos pinos 5 e 4 do 4N25 são conectados a tensão de 5V e a porta A0 do microcontrolador, cujo valor lido será interpretado e processado posteriormente. O resistor R2 é utilizado como resistor de pull-down, ou seja, garante que a tensão lida será zero caso a tensão seja muito abaixo de 110V nos pinos 1 e 2. O capacitor C1 é utilizado para filtrar a tensão alternada medida. Sem este capacitor o resultado da leitura oscilaria entre 0 e valor máximo da tensão de

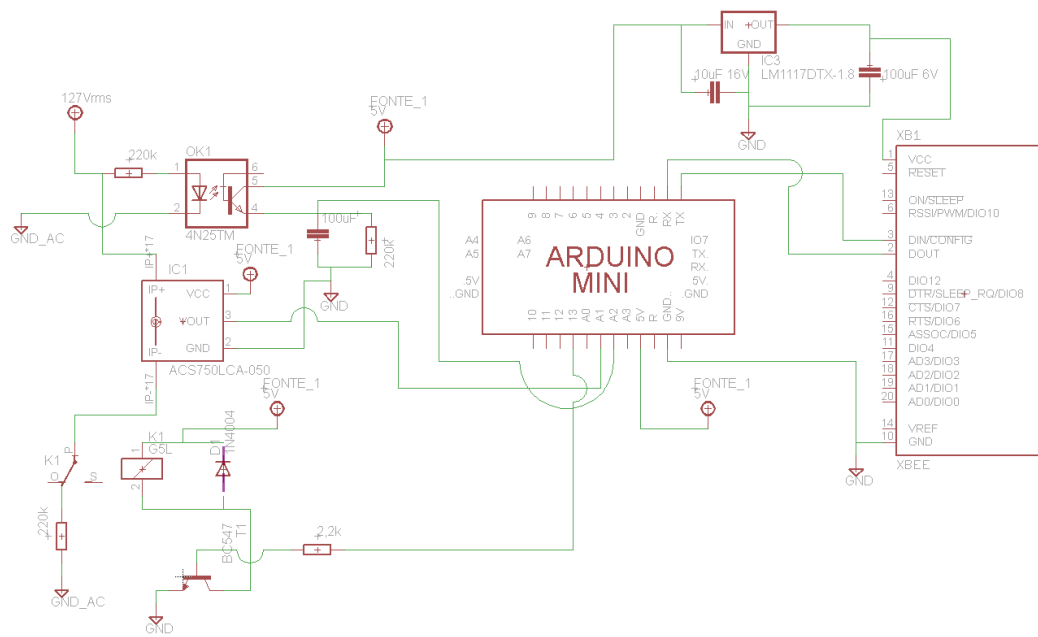


Figura 22: Esquemático do módulo medidor

Fonte: Autoria própria

entrada, representando o semiciclo positivo da onda senoidal de uma fonte de tensão alternada.

A medição da corrente é realizada pelo TA12-100. Por tratar-se de um sensor não intrusivo, o TA12-100 envolve o condutor que está ligado no Relé K1, que por sua vez está ligado em série com a carga. O pino 3, chamado de OUT, do medidor de corrente é o resultado da corrente medida. Este pino é conectado a porta analógica A1 do Arduino, e seu valor varia entre 0 e 1V. Este valor será interpretado pelo microcontrolador para calcular quanta corrente está sendo utilizada pela carga.

A função do relé é de trabalhar como chaveador da alimentação da carga. Por ser do tipo NF (normalmente fechado) a corrente circula livremente até que o pino de controle esteja em estado alto. A corrente volta a circular quando o estado do pino de controle retornar a LOW. O controle do pino de controle, e consequentemente do chaveamento da corrente é feito pelo microcontrolador, mais especificamente pelo pino 13 do Arduino.

Os pinos RX1 e TX0 do microcontrolador são utilizados para a comunicação serial. A comunicação é utilizada tanto para programar o microcontrolador, quanto para a comunicação com o módulo Xbee. Apesar de existirem 2 conjuntos de pinos RX/TX apenas um possui interface de comunicação UART dedicada, porém qualquer pino digital pode ser utilizado para comunicação serial se utilizada a biblioteca SoftwareSerial (ARDUINO, 2014). Por tratar-se

de uma comunicação serial o pino RX, ou receptor, é conectado ao pino TX(ou transmissor) do Xbee e vice-versa. Vale notar que a alimentação do Xbee é de 3.3V, diferentemente da 5V dos outros componentes.

Na figura 22 estão presentes duas fontes de tensão. A primeira é a fonte 127Vrms, que representa a rede elétrica. A segunda fonte utilizada é a fonte "FONTE_1". Ela tem valor de 5V de corrente contínua. Ela é utilizada na alimentação do Arduino, dos sensores de tensão e corrente, como chaveadora do relé K1 e como entrada do sistema regulador de tensão posicionado acima do Xbee na figura 22. Este regulador tem como papel reduzir a tensão de 5V original da fonte para 3.3V, para servir como alimentação ao Xbee.

4.2 COMUNICAÇÃO

Com o projeto do hardware responsável pela medição terminado, é iniciado o projeto da comunicação do sistema. Esta comunicação será dada em duas etapas. A primeira é interna ao sistema, na qual não há interação do usuário de forma direta. Nesta etapa estão envolvidos os medidores, o controlador e os módulos Xbee. Para esta etapa será definido um protocolo de comunicação, o qual deverá ser capaz de transmitir os dados obtidos pelo medidor além de transmitir comandos referentes às funcionalidades do sistema entre medidor e controlador. Como ambos os lados devem interagir com um módulo xbee, do lado dos medidores será utilizada a linguagem *Wiring*, e do lado do controlador será utilizada python.

A segunda etapa envolve a interface com o usuário. Nesta etapa o componente principal é o controlador, responsável por hospedar o servidor do website que por sua vez servirá como central de comando do sistema para o usuário. Para a programação do servidor será utilizada a linguagem python, mais especificamente o framework cherrypy. Ao utilizar python para desenvolver o servidor, a interação entre as duas etapas é facilitada.

4.2.1 PROTOCOLO DE COMUNICAÇÃO

Os módulos Xbee possuem dois modos de operação, o modo transparente ou AT e o modo programável pela aplicação ou API(Application Programmable Interface). Quando operando em modo AT os módulos funcionam como substituição de uma linha serial. Neste modo todo dado recebido pelo pino DIN ou RX to Xbee é encaminhado para transmissão. De forma similar quando algum dado é recebido pelo módulo, este é encaminhado para o pino DOUT ou TX. Os dados recebidos pelo módulo são armazenados no buffer, e transmitidos pelo pino quando uma das condições é satisfeita: O tempo limite entre caracteres recebidos é

extrapolado, é recebido o comando de fim de pacote ou a quantidade máxima de bytes suportada pelo buffer é atingida.

Como já discutido no capítulo 3, será utilizada a topologia estrela. Quando utiliza-se o modo AT nesta topologia, é necessário utilizar a configuração broadcast para mensagens enviadas pelo controlador. Sendo assim, todos os medidores receberão as mensagens conforme a figura 23, e isto deve ser tratado posteriormente para não prejudicar o funcionamento dos medidores que não são os destinatários da mensagem.

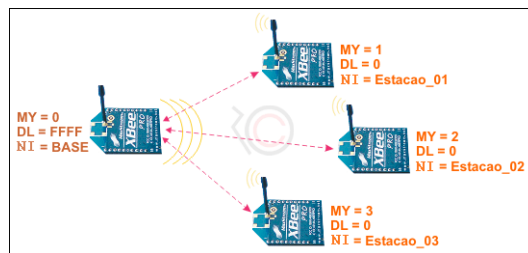


Figura 23: Topologia Ponto-Multiponto (broadcast)

Fonte: (ROGERCOM, 2008)

Ao utilizar o modo API, as capacidades de interação entre a aplicação e a rede aumentam. Isto acontece pois quando em modo API, toda a informação transmitida deve seguir um formato de pacote específico, representado na figura 24. Quando neste modo, pacotes enviados pelo módulo incluem: o campo dos dados a serem transmitidos e o campo de comando. Pacotes recebidos possuem três campos. São eles: A informação propriamente dita, a resposta ao comando do pacote enviado (se algum pacote foi enviado anteriormente), e notificações de eventos. Na figura 24 o campo cmdID representa o tipo do comando que o módulo deve executar e o campo cmdData a informação relacionada a este comando.

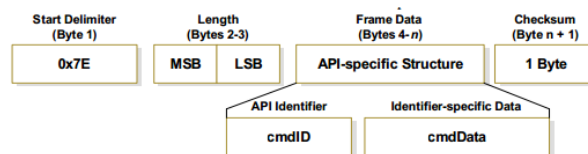


Figura 24: Estrutura de um pacote API

Fonte: Xbee Product manual, 2015

Quando comparados, o método API é mais atraente ao sistema proposto. Apesar do modo AT possuir uma interface mais simples, o modo API apresenta vantagens quando o gerenciamento de mensagens de diferentes fontes é necessário. Esta vantagem somada a capacidade

de identificar o endereço do remetente da mensagem, possibilita que o controlador não mais envie a informação em formato broadcast, mas apenas para o destinatário desejado. Além disso, o uso do modo API possibilita a identificação de falha ao enviar mensagens, tornando o canal de comunicação mais confiável.

Uma das características do sistema é que a entrada na rede dos medidores deve ser feita de forma dinâmica. Além disso o sistema deve ser capaz de suportar vários medidores simultaneamente. Por estes motivos julgou-se necessário definir a sequência de ações entre medidor e controlador para uma comunicação bem sucedida. O diagrama 25 representa esta sequência, desde o momento que um medidor informa ao controlador que está disponível até o momento que este medidor retira-se da rede. Possíveis erros de comunicação não são representados no diagrama.

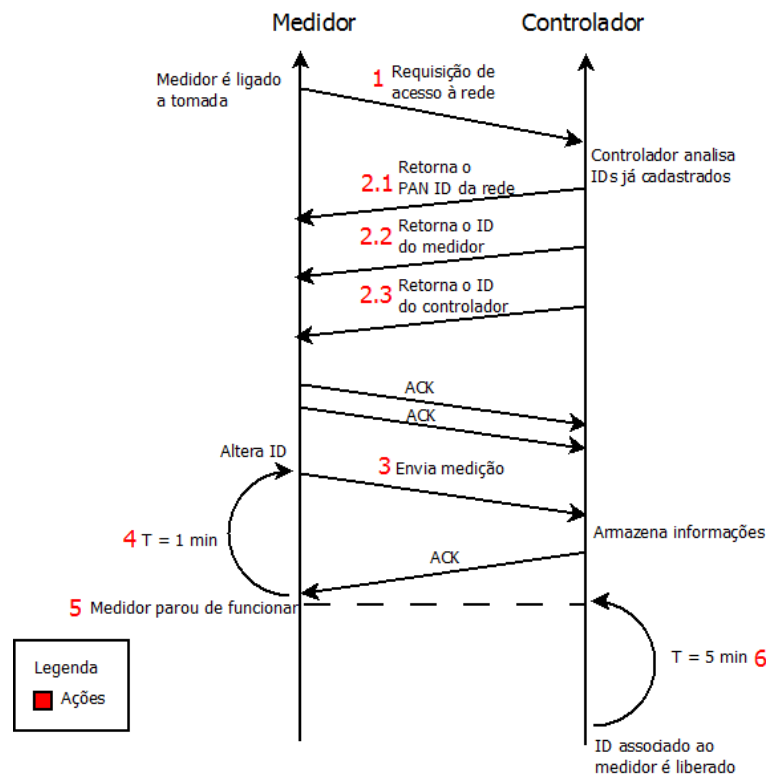


Figura 25: Fluxo da informação entre medidor e controlador

Fonte: Autoria própria

A implementação das ações representadas pelos números 1 a 4 na figura 25 é possível graças as funcionalidades disponíveis quando utilizado o modo de operação API. Ao utilizar o modo API, a configuração de um dispositivo Xbee pode ser alterada remotamente. Esta capacidade será utilizada para fazer o gerenciamento dos dispositivos conectados a rede do sistema. Pacotes que possuem o valor 0x17 no campo cmdID indicam que aquele *frame* é

do tipo "Comando AT Remoto" e tem a finalidade de alterar a configuração de um módulo presente na rede. A informação do endereço bem como qual alteração será feita está presente no campo cmdData. O destinatário ao receber um pacote do tipo "Comando AT Remoto" executa as alterações requisitadas e então envia um pacote "Resposta ao Comando AT Remoto", podendo conter um ou mais *frames* API. O módulo que recebeu o o pacote "Resposta ao Comando AT Remoto" encaminha estes frames pela saída serial, que podem então ser analisados.

Para a transmissão da informação são utilizados pacotes com comandos AT de "Requisição de Transmissão", "Status da Transmissão" e "Pacote Recebido". Para enviar dados via RF para um dispositivo, um pacote do tipo "Requisição de Transmissão" deve ser montado. Este pacote será interpretado pelo módulo xbee e enviado para o destino especificado. A informação contida no campo "RF Data" da figura 26 é formatada automaticamente em um pacote do tipo "Pacote Recebido", posteriormente interpretado pelo destinatário. Quando o destinatário receber um pacote do tipo "Requisição de Transmissão" (representado na figura 26), enviará como resposta um pacote do tipo "Status da Transmissão" indicando se a transmissão foi efetuada com sucesso ou se houve erro, e encaminhará pela UART os dados presentes no pacote "Pacote Recebido".

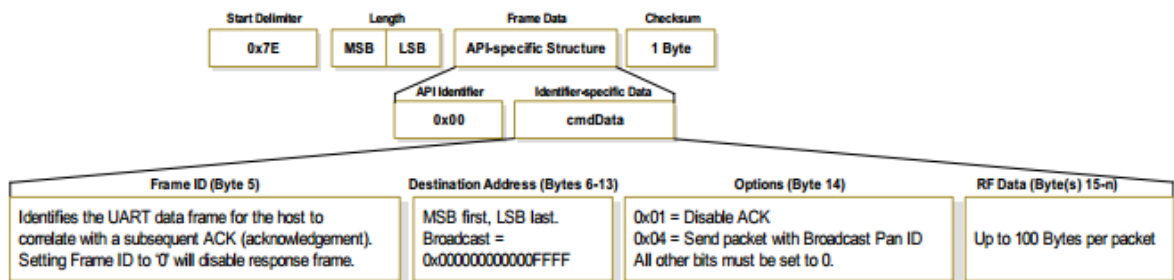


Figura 26: Pacote "Requisição de Transmissão"

Fonte: Xbee Product manual, 2015

A tabela 4 relaciona as ações enumeradas na figura 25 com o tipo de comando utilizado e os resultados esperados. Não estão representadas as respostas ACK, pois estas são internas ao funcionamento do protocolo zigbee. Como as ações 4, 5 e 6 não estão relacionadas ao protocolo de comunicação, serão explicadas a parte. A Ação 4 indica o período utilizado para a atualização do valor das medidas. Ela indica que a cada um minuto será enviada a informação do consumo do medidor. Caso o medidor apresente falha e pare de funcionar, ou seja desacoplado da tomada pelo usuário (representado por 5 na figura 25), o mesmo tem um intervalo de cinco minutos até enviar um novo valor de medição ao controlador, caso contrário será considerado inativo e portanto seu ID será considerado disponível para alocação.

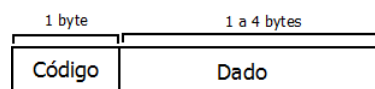
Definida a sequência de etapas para estabelecer um medidor na rede, resta definir o

Tabela 4: Relacionamento entre comandos do modo API e resultados esperados

Ação	Comando API	Interpretação	Resultado esperado
1	0x01	Envia a mensagem "!!!" para todos os PAN ID's em formato broadcast	O controlador ao receber este tipo de mensagem analisa qual o primeiro ID disponível na lista de ID's e associa ao medidor remetente da mensagem.
2.1	0x17	Atualiza a configuração do PAN ID da rede	Ao informar o PAN ID o medidor entra na rede do sistema.
2.2	0x17	Atualiza o ID do medidor	A atualização do ID do medidor é necessária pois ao receber novas mensagens deste medidor, o controlador gerenciará a informação corretamente.
2.3	0x17	Atualiza a configuração do ID do controlador	Com o ID do controlador, o medidor não enviará as informações no modo broadcast, mas sim somente para o controlador.
3	0x01	Envia dados de medição para o controlador	os dados da medição são salvos e disponibilizados no website.

Fonte: autoria própria

formato que cada tipo de dado será transmitido pelo controlador ou medidor. O medidor pode enviar dois tipos de informações: dados da medição ou estado do medidor. Diferentemente, o controlador pode requisitar o estado do medidor, ou enviar comandos, como por exemplo cessar o acesso à energia naquele medidor. Desta forma, baseado no formato de mensagem API, definiu-se dois campos para toda mensagem que transitar na rede. O primeiro campo representa a função da mensagem, e o segundo a informação relacionada. A figura 27 representa a estrutura da mensagem e a tabela 5 o código da mensagem, a função da mensagem e o respectivo dado.

**Figura 27: Estrutura de uma mensagem do sistema**

Fonte: Autoria própria

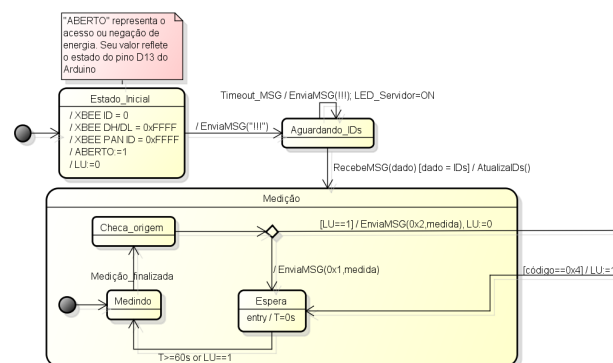
Tabela 5: Relação entre comandos e funções do protocolo de comunicação

Fonte	Código	Função	Dado
Controlador	0x1	Verificar estado do medidor	-
Controlador	0x2	Desligar acesso à energia	-
Controlador	0x3	Ligar acesso à energia	-
Controlador	0x4	Informação do consumo atual	-
Medidor	0x1	Consumo medido no último minuto	4 bytes
Medidor	0x2	Consumo atual	4 bytes
Medidor	0x3	Energia acessível no medidor	0x1 - acessível; 0x2 - não acessível

Fonte: autoria própria

4.2.2 FIRMWARE

Neste projeto existem dois firmwares. O primeiro é o do medidor e o seu papel é de ler periodicamente o valor do consumo, e quando requisitado executar comandos recebidos pelo Xbee via serial. Estes comandos já foram definidos na tabela 5. A figura 28 representa os estados de configuração inicial e leitura de consumo da máquina de estados do medidor, enquanto na figura 29 está representando o estado de análise de comandos recebidos pelo medidor. No Apêndice C está representada a máquina de estados completa do firmware do medidor.

**Figura 28: Máquina de estados da configuração e leitura do medidor**

Fonte: Autoria própria

A máquina de estados é composta por dois superestados, cada um correspondendo a cada uma das funções do firmware, e dois estados representando a inicialização do medidor. São definidas 2 variáveis para controle sendo elas "ABERTO" e "LU" (leitura única). A primeira representa o estado do relé encarregado por autorizar/negar o acesso à energia. A segunda

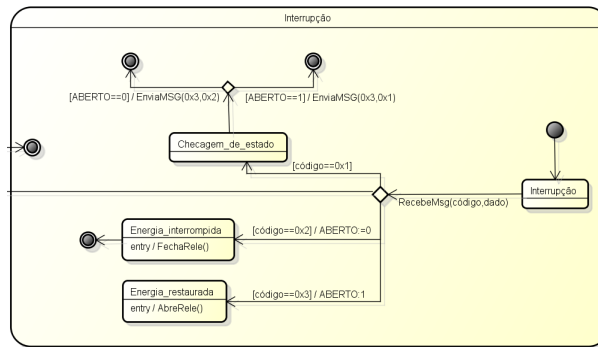


Figura 29: Máquina de estados do tratamento de comandos recebidos pelo medidor

Fonte: Autoria própria

representa se a medição a ser realizada é reposta a uma requisição feita pelo controlador, ou se é uma medição periódica. As três etapas da máquina de estados são Inicialização, Medição e Comunicação e serão explicadas a seguir.

- **Inicialização:** Neste estado as variáveis de configuração são inicializadas e o módulo Xbee é configurado para enviar informações em formato broadcast para toda as redes disponíveis. Ao receber comandos de atualização das configurações do módulo Xbee, tem início a próxima etapa.
- **Medição:** Uma vez que o medidor está inserido na rede, medições de consumo são enviadas periodicamente para o controlador ou quando uma requisição específica do controlador é recebida, representada pela transição do superestado "Interrupção" para "Medição".
- **Comunicação:** A comunicação do sistema está representada pelo superestado Interrupção e pelos métodos EnviaMSG() e RecebeMSG(). Quando uma mensagem é recebida pelo Xbee, ocorre uma interrupção de sua respectiva rotina de tratamento. Cada código presente na máquina de estados foi descrito na tabela 5.

Como já discutido, será utilizada a linguagem *python* para a programação tanto do servidor que hospedará o site quanto para o firmware responsável por manipular dados e comandos enviados pela serial. Desta forma a interação entre ambas as partes é facilitada pois a informação que transitará entre o servidor e o gerenciador da serial é facilmente acessada. Os estados de montagem de mensagens e do tratamento de mensagens no controlador estão presentes nas figuras 30 e 31 respectivamente. A relação entre estes estados, bem como mais detalhes da máquina de estados do controlador podem ser encontrados no Apêndice D.

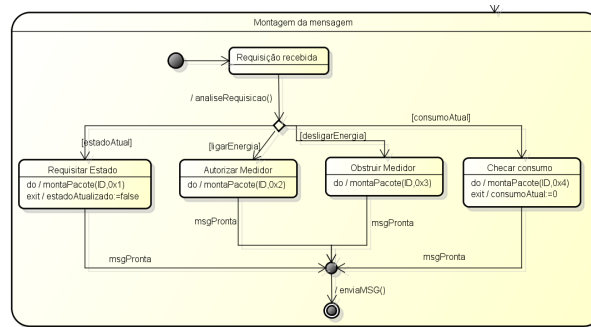


Figura 30: Máquina de estados da montagem de mensagens pelo controlador

Fonte: Autoria própria

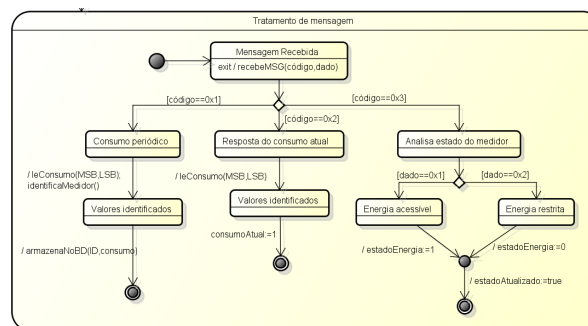


Figura 31: Máquina de estados do tratamento de mensagens no controlador

Fonte: Autoria própria

Esta máquina de estados é semelhante a do medidor, porém não interage com nenhum componente eletrônico com exceção do Xbee mas compartilha dados com o servidor. Ela é composta por 3 etapas: Inicialização do firmware, a montagem de mensagens e o tratamento de mensagens recebidas. São utilizadas 3 variáveis de controle. A explicação destas está presente no próprio diagrama. As três etapas serão descritas a seguir:

- **Inicialização:** Semelhante ao firmware do medidor, nesta etapa são inicializadas as variáveis de controle. Após a inicialização, o controlador entra em um estado IDLE, ou seja, um estado de aguardo. Ele sai deste estado quando recebe uma requisição do usuário através do servidor(website) ou quando recebe uma mensagem de algum medidor.
- **Montagem da mensagem:** Este superestado tem início quando o servidor deseja obter alguma informação de um medidor específico. Neste estado a requisição é formatada seguindo o protocolo definido na tabela 5 e a mensagem é então enviada. Ao fim do envio da mensagem, o sistema retorna ao estado IDLE.
- **Tratamento de mensagem:** Quando uma mensagem é recebida pelo controlador, uma

exceção ocorre e então inicia-se o seu tratamento. Diferentes tipos de mensagens possuem diferentes tratamentos. Semelhante ao estado descrito anteriormente, ao fim do tratamento da mensagem o sistema retorna ao estado IDLE

4.3 INTERFACE

Esta seção tem o objetivo de apresentar o planejamento de como será feita a interação com o usuário. Aqui serão definidos o fluxo da informação envolvendo o usuário, controlador e medidores. Será definido como as funções necessárias ao website serão apresentadas, e como o website irá interagir com o servidor (controlador) e usuário.

4.3.1 FLUXO DA INFORMAÇÃO

Para demonstrar como a informação transita na rede, e como esta interage com os componentes, foi gerado o fluxograma da figura 32. Neste fluxograma são definidas 5 ações do sistema, enumeradas de um a cinco, que geram o trânsito da informação. Algumas destas ações não utilizam o armazenamento do sistema pois elas são de efeito imediato. As ações são executadas pelo usuário, com exceção da atualização periódica de consumo, a qual tem origem no medidor. A seguir serão levantados pontos relevantes para cada uma das ações.

- **Autoriza/nega acesso a energia:** Autoriza ou nega o acesso a energia imediatamente do medidor desejado. Esta é uma ação que não utiliza o armazenamento do sistema.
- **Consumo em tempo real:** Ação imediata que não necessita acessar a armazenagem porém acessa o medidor e percorre o caminho completo do sistema. Quando ele realiza tal ação a informação é armazenada pelo servidor. O controlador é responsável por ativar/desativar o medidor quando o período programado tiver início/fim.
- **Adiciona/remove rotina de funcionamento:** O usuário pode alterar a qualquer momento a rotina em que um medidor está autorizado a negar/permitir o acesso a energia.
- **Verificar estado do medidor:** Verifica se o medidor está negando ou permitindo o acesso a energia. Apesar de ser uma ação imediata, a informação é acessada no armazenamento. A informação do estado do medidor é atualizada pelo controlador, conforme parâmetros definidos na figura 25.
- **Atualizações periódicas do consumo:** Periodicamente os medidores enviam ao controlador a medição realizada naquele período. O controlador analisa a informação recebida e

a armazena. A informação do consumo periódico é lida pelo servidor do armazenamento e então encaminhada ao website.

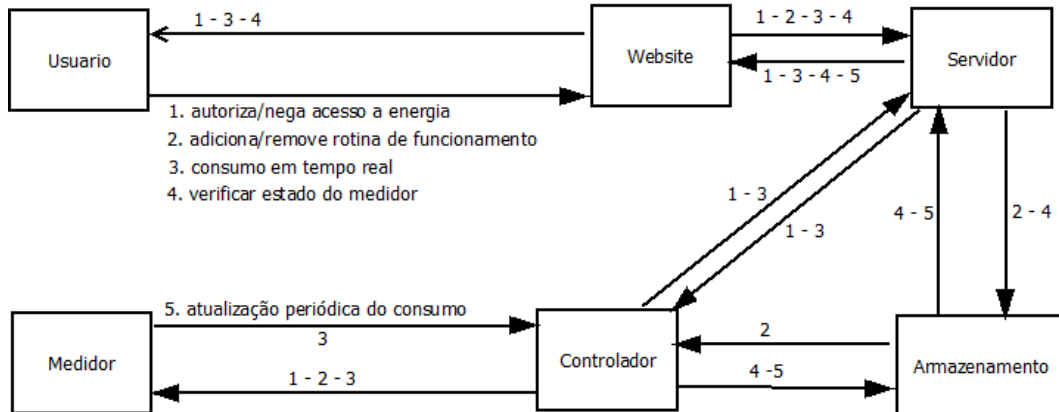


Figura 32: Fluxograma da transmissão da informação em alto nível

Fonte: Autoria própria

4.3.2 WEBSITE E SERVIDOR

Tomando como base os requisitos da tabela 2, foi traçado um esboço do website. Este será composto por duas páginas principais. A primeira página é definida como index, e será a página inicial apresentada ao usuário. Nesta página estará presente uma lista com todos os medidores disponíveis no sistema. O usuário poderá então clicar em um destes, o que o levará a segunda página.

A segunda página contém todas as informações pertinentes ao medidor escolhido. Nela serão mostrados o consumo total medido no módulo, o consumo medido por período, se o módulo está autorizando ou negando o acesso a energia, e o comportamento programado do módulo. A interação do usuário com esta página é muito maior se comparada à página inicial, pois todas as ferramentas de gerenciamento estão presentes nela.

Para definir como as funcionalidades fornecidas pelo website irão interagir com o sistema, foram definidos os diagramas de sequência entre o usuário, servidor e website. Neste diagrama foram inclusas as ações executadas pelo usuário e como estas serão tratadas pelo sistema. Nos diagramas de sequência também estão definidas as etapas tomadas desde o primeiro acesso do usuário ao website, e como a informação já disponível no sistema será transmitida e exibida para o usuário

Para a gerenciar as requisições entre servidor e website, foi utilizada uma arquitetura

RESTful (ALLAMARAJU, 2010), disponibilizada ao utilizar o framework *cherrypy* (TEAM, 2001). Este framework possibilita programar aplicações web semelhantemente a softwares orientados a objetos. Sendo assim possibilita, dentro de sua estrutura, criar classes e objetos para tratar as diferentes requisições e suprir a estrutura necessária que uma arquitetura RESTful aconselha.

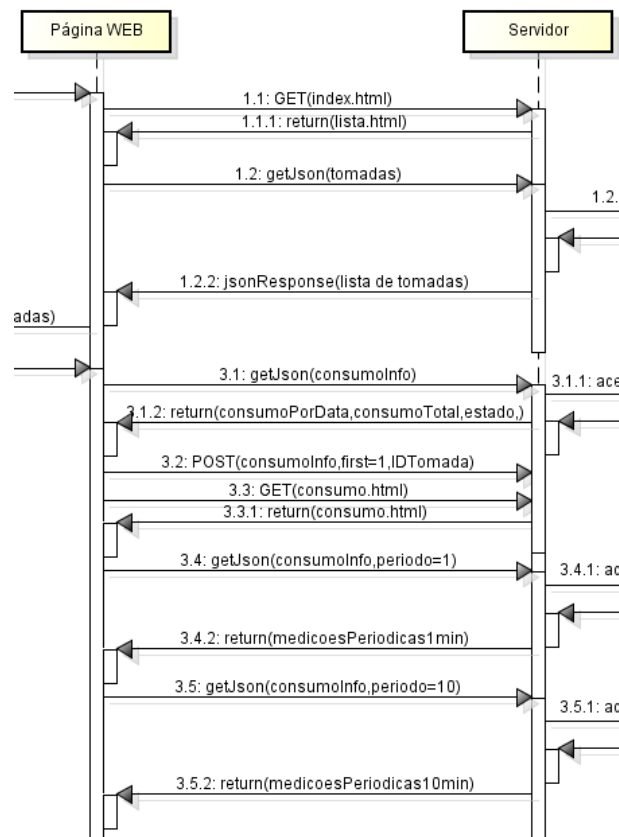


Figura 33: Diagrama de sequência entre página web e Servidor

Fonte: Autoria própria

Na figura 33 está representado o diagrama de sequência associado a troca de informação, entre a página web e o servidor, quando o usuário abre o website pela primeira vez. O mesmo diagrama também está disponível no Apêndice E, porém também contendo o usuário como ator. Considerou-se que a inicialização da página corresponde ao acesso da lista de tomadas e o acesso a mais informações de qualquer uma das tomadas presentes nesta lista. Os principais métodos são explicados a seguir:

- **GET e POST:** São a forma mais simples de métodos de requisição HTTP. GET geralmente é utilizado quando se requer uma informação e POST quando deseja-se enviar um elemento.

- **acessaBD**: Este método representa o acesso ao banco de dados, queira para obter ou atualizar algum dado. Seu parâmetro indica qual tabela será acessada.
- **get.Json e jsonResponse**: estes métodos são utilizados quando deseja-se transmitir informação em formato texto. Utiliza-se o formato JSON(JavaScript Object Notation) (CROCKFORD, 2006) pois este usa convenções, familiares a programadores, para organizar seu conteúdo. Desta forma o uso da formatação JSON possibilita que a manipulação da informação seja facilitada e otimiza o tráfego na rede.
- **atualizaDisplay**: Neste método geralmente é processada a informação recebida do servidor. É também onde o DOM(Document Object Model), ou layout da página, é atualizado com a informação recebida.

A figura 34 representa as ações entre o servidor e o controlador. Na figura do Apêndice F está representada a figura 34 bem como o diagrama de sequência de todas as ações não representadas pelo diagrama do Apêndice E. Nele são mostradas quais ações o usuário poderá exercer sobre o sistema, onde para cada uma existe uma rotina específica. Os métodos representados no diagrama do Apêndice F são os mesmos do Apêndice E.

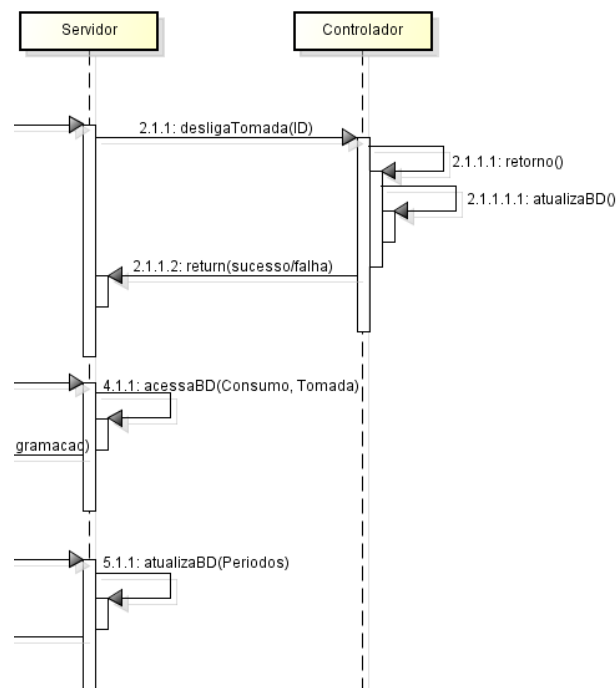


Figura 34: Diagrama de sequência entre servidor e controlador

Fonte: Autoria própria

4.3.3 ARMAZENAMENTO

Como o sistema deve gerenciar diferentes faixas de horário de funcionamento para diferentes medidores, escolheu-se por utilizar um banco de dados como ferramenta. O diagrama entidade-relacionamento do banco de dados está representado na figura 35 e cada tabela será explicada a seguir:

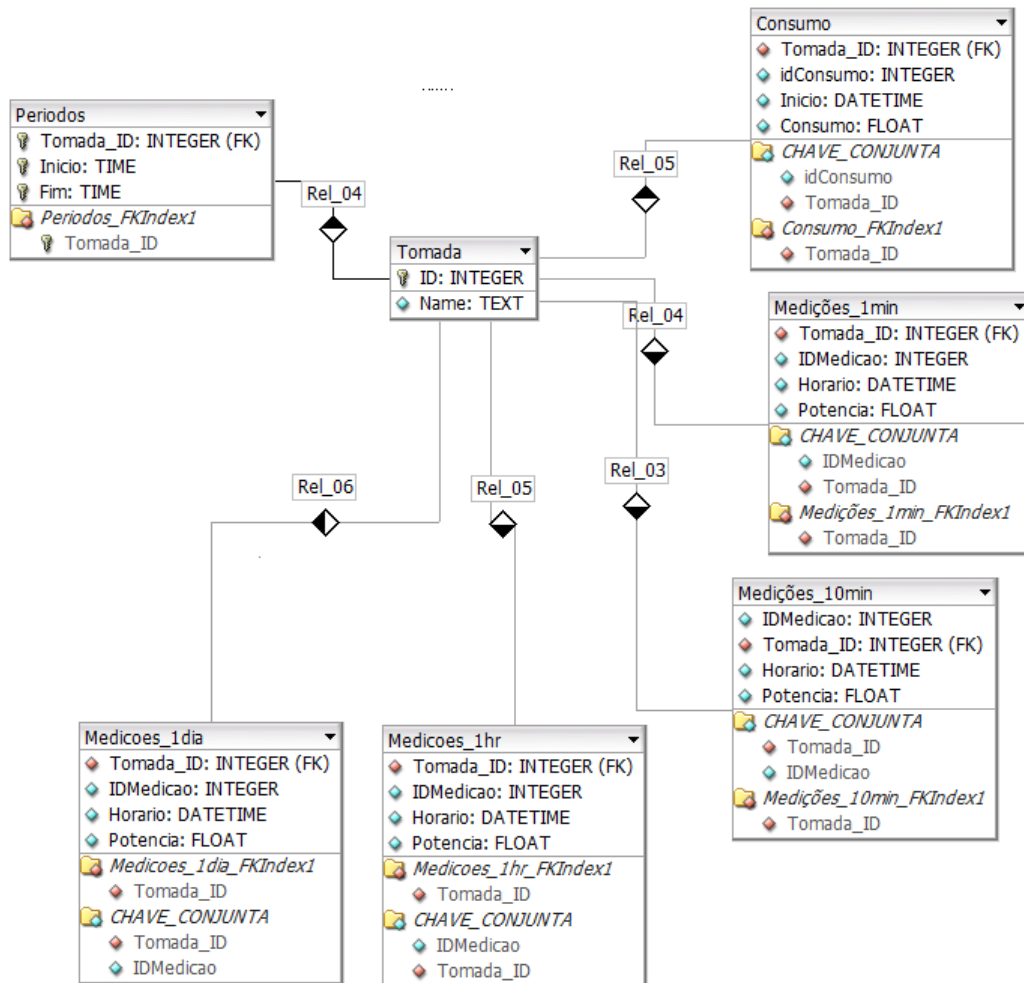


Figura 35: Diagrama entidade relacionamento do banco de dados

Fonte: Autoria própria

- **Tomada:** Tabela contendo todos os medidores presentes na rede. Esta tabela será alterada sempre que um novo medidor entrar/sair da rede.
 - ID: chave primária identificando o ID de cada medidor presente na rede.
 - Name: Nome opcional dado para um medidor.

- **Consumo:** Representa o consumo medido em cada módulo. Cada tupla da tabela representa o consumo total medido a partir de uma data.
 - Tomada_ID: chave estrangeira herdada da tabela "Tomada".
 - idConsumo: ID do consumo.
 - Inicio: A data a partir da qual começou-se a medir o consumo.
 - Consumo: O valor do consumo medido.
 - CHAVE_CONJUNTA: o "idConsumo" junto a "Tomada_ID" identificam unicamente uma tupla.

- **Medições_xmin:** Estas tabelas são utilizadas para armazenar o consumo medido periodicamente pelos módulos. A tabela com período inferior atualiza a sua sucessora (período imediatamente maior) quando a quantidade de elementos armazenados para cada medidor atingir a quantidade especificada pela sua sucessora. Por exemplo, quando houverem 10 medições para um dado medidor na tabela "Medições_1min", uma rotina de atualização soma o valor das últimas 10 medições e insere uma nova tupla na tabela "Medições_10min".
 - Tomada_ID: chave estrangeira herdada da tabela "Tomada".
 - IDMedicao: ID da medição.
 - Horario: O horário da medição.
 - Potencia: O valor do consumo medido.
 - CHAVE_CONJUNTA: o "IDMedição" junto a "Tomada_ID" identificam unicamente uma tupla.

- **Periodos:** Esta tabela armazena a rotina de operação de cada medidor. Esta rotina pode ser alterada pelo usuário por meio do website.
 - Tomada_ID: chave estrangeira herdada da tabela "Tomada".
 - Inicio: Momento no qual será autorizado o acesso a energia no medidor "Tomada_ID".
 - Fim: Momento no qual será negado o acesso a energia no medidor "Tomada_ID".

5 ENSAIOS E RESULTADOS

Este capítulo tem como objetivo realizar ensaios com o sistema, com o intuito de constatar se os requisitos estipulados conforme tabela 2 foram alcançados. Para cada um dos requisitos, serão definidos um ou mais testes. Nas próximas seções serão definidos os ensaios, qual(is) requisito(s) eles atendem e quais funcionalidades do sistema serão testadas.

5.1 MEDIÇÃO DO CONSUMO

O primeiro ensaio a ser realizado tem por objetivo averiguar a capacidade de medição de um módulo medidor. Neste ensaio apenas o módulo medidor será envolvido sem nenhuma interação com o resto do sistema.

O teste consiste em medir a quantidade de energia consumida em uma tomada. O módulo medidor será acoplado a uma tomada residencial, na qual cargas com diferentes valores de potência serão ligadas. A medição será feita durante 10 minutos, e os valores das medições serão transmitidos via serial para o computador, e lidos através de um console. Serão transmitidos ambos os valores da corrente medida, como do consumo calculado, uma vez que este depende da corrente e tensão. Ao final do teste espera-se obter uma lista de valores medidos, representando as medidas obtidas periodicamente a cada minuto. Ao longo do teste será avaliado o valor da corrente medida a cada variação de carga e comparado com o valor medido pelo amperímetro comercial fluke 322.

Com este teste pretende-se avaliar as seguintes funções:

- Medir a corrente utilizada pela carga acoplada ao medidor.
- Avaliar a capacidade do medidor de realizar medições periódicas.

Resultados:

Para calibrar o sensor e verificar a sua funcionalidade foram realizadas diferentes medidas de corrente, usando tanto o sensor TA12-100 quanto um amperímetro comercial modelo

fluke 322. Os aparelhos utilizados como carga no experimento foram 3 ventiladores, sendo 2 com potência nominal de 105W e um com potência nominal de 150W. Para alcançar corrente próxima a de 5A foi utilizado um aquecedor com potência nominal mínima de 600W, uma vez que a tensão da tomada utilizada é de aproximadamente 127V.

Iniciou-se com a menor carga possível que foi aumentada gradativamente até alcançar a carga máxima. A medida que a corrente aumentou foram medidos os valores relatados pelo sensor TA12-100 e pelo amperímetro. Estes valores estão presentes na tabela 6 e no gráfico da figura 36. Como resultado de várias iterações deste teste com diferentes valores para o resistor de carga, chegou-se ao resultado com menor erro ao utilizar o valor de 210Ω para o resistor de carga.

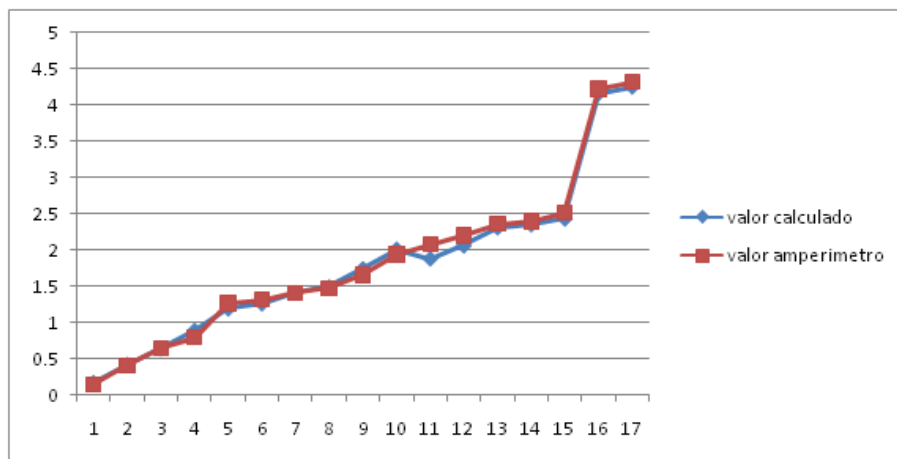


Figura 36: Dados do teste da medição de corrente

Fonte: autoria própria

Para definir a precisão do medidor, comparou-se apenas os valores medidos para corrente, uma vez que considera-se que a tensão será sempre estável. Com os valores obtidos, a precisão dos valores obtidos foram satisfatórios. Dos valores medidos, o maior erro relativo foi de 11,25%, no qual foi medida a corrente de 890mA pelo módulo e de 800mA pelo amperímetro.

Foi definido o período de um segundo entre medições de tensão e corrente. Para cada uma destas medições é calculada a potência. Ao se obterem 60 medições, é realizada uma média destas medições, e obtém-se o valor do consumo no espaço de tempo de um minuto. O valor determina o consumo em kWh se a carga que estava acoplada ao medidor durante o último minuto se mantivesse por uma hora. Posteriormente, similarmente as medições espaçadas em um minuto, será feita uma média das medições enviadas ao controlador e será obtido o consumo

Tabela 6: Dados da medição de corrente

Valor no sensor	Valor calculado (A)	Valor amperímetro (A)	Erro relativo (%)
11	0.17	0.16	6.25
27	0.42	0.41	2.43
42	0.64	0.66	3.03
58	0.89	0.8	11.25
76	1.19	1.27	6.29
80	1.26	1.32	4.54
91	1.41	1.41	0
95	1.49	1.48	.67
111	1.74	1.66	4.81
128	2.01	1.94	3.60
120	1.88	2.08	9.61
131	2.06	2.2	6.36
147	2.31	2.36	2.11
150	2.35	2.4	2.08
155	2.43	2.51	3.18
241	4.16	4.22	1.42
246	4.25	4.31	1.39

Fonte: autoria própria

em kWh relativo a uma hora de funcionamento.

5.2 COMUNICAÇÃO MEDIDOR-CONTROLADOR

Este ensaio tem o objetivo de avaliar o funcionamento da comunicação entre o medidor e o controlador. Os componentes utilizados neste teste serão um medidor, um controlador e seus respectivos módulos Xbee.

No primeiro teste a inserção do medidor na rede será testada, e o medidor será responsável por enviar periodicamente mensagens ao controlador contendo o valor de consumo medido no último minuto. Estes valores ao serem recebidos serão armazenados pelo controlador. Ao final espera-se ter recebido os valores corretamente e que os mesmos estejam prontos para serem acessados pelo servidor.

No segundo teste serão trocadas mensagens entre o controlador e medidor, representando as mensagens definidas pelo protocolo da tabela 5. Estas mensagens representam os comandos e informações trocadas entre o medidor e controlador. Para este teste os comandos serão enviados de forma manual pelo controlador, enquanto ao fim do projeto serão enviados de forma automatizada. Como resultado espera-se que o controlador execute os comandos enviados pelo controlador, e que o controlador receba corretamente as informações do medidor.

Com o primeiro teste deseja-se avaliar a comunicação inicial entre controlador e medidor, bem como a capacidade de transmitir as medições do consumo de forma automatizada e periódica. O segundo teste tem como objetivo comprovar a eficácia do protocolo de comunicação definido e confirmar a capacidade do módulo de executar os comandos de controle de energia.

Resultados: Foi utilizada a biblioteca Xbee-Arduino (RAPP, 2014) para realizar a comunicação entre o Arduino e os Xbee's sendo utilizados no modo API. A biblioteca possui métodos para criar os pacotes necessários no formato API especificado pela desenvolvedora do Xbee. Com ela é possível criar os pacotes tanto de recepção de dados quanto para envio. De forma geral, ela simplifica a montagem e desmontagem dos pacotes, com flexibilidade suficiente para manipulação dos dados contidos dentro destes pacotes.

Para o lado do controlador é possível (assim como também é possível com o Arduino) fazer o interfaceamento entre processador e Xbee através da porta serial diretamente. Porém optou-se por utilizar a biblioteca python-Xbee (FALUDI, 2010). Possui características semelhantes à utilizada pelo arduino, porém seu uso é mais transparente que a biblioteca do Arduino. Ambas as bibliotecas possibilitam o tratamento de mensagens no modo API, tanto para recebimento quanto para envio.

O primeiro teste foi dividido em duas subetapas, sendo a primeira correspondente ao *handshake* entre o controlador e medidor e a segunda o envio das medições realizadas pelo medidor. Para a primeira etapa, foi utilizada a sequência definida na figura 25. Inicialmente apaga-se qualquer configuração presente no xbee do módulo. Então o medidor fica em *loop* enviando a mensagem "!!!" em formato broadcast a cada 5 segundos. Quando recebe uma resposta com a mensagem "

"junto com o ID da rede, altera estes valores no xbee. Desta forma qualquer pacote enviado será recebido apenas por membros desta rede. Em seguida o medidor recebe um pacote com o seu novo ID. Após realizar a alteração de seu ID no xbee, o controlador poderá identificar pacotes recebidos deste medidor. O medidor está apto então a começar a enviar as medidas.

Do lado do controlador foi criada uma thread para o tratamento de mensagens recebidas pelo xbee. Enquanto uma thread fica responsável pelo tratamento das requisições do website, outra fica responsável pelas mensagens da rede interna. A primeira ação realizada pelo controlador é de limpar tanto a lista de medidores conectados quanto entradas do banco de dados. A thread responsável fica então em estado IDLE até que receba uma mensagem contendo a mensagem "///". Após recebida, ele retorna ao remetente a mensagem contendo o ID da rede. Realiza uma pesquisa para encontrar o primeiro ID de medidores disponível. O controlador

envia então a mensagem contendo o novo ID do medidor e sua thread volta ao estado IDLE.

Após o fim do processo de *handshake*, o medidor começa a enviar os dados de medição ao controlador. Estes dados são enviados a cada minuto, porém a medição é feita a cada segundo pelo medidor. Utiliza-se um timer que, a cada segundo realiza a medição e a armazena em um vetor de tamanho 60. Após ter-se passado um minuto, outro timer é ativado e calcula a média do vetor de medições, e envia esta informação ao controlador. Ambos os timers são reativados ao fim de seus períodos. O primeiro teste foi bem sucedido, a demonstração de seu resultado será feita junto ao segundo teste e o da seção 5.3.

Para o segundo teste utilizaram-se as mesmas bibliotecas do teste anterior porém a informação transmitida agora seguiu o protocolo definido na tabela 5. A maioria do campo de dados das mensagens carrega apenas um byte contendo a informação, com exceção de mensagens contendo o valor da medição. Para esta mensagens utilizou o tipo de variável *float*, e foram utilizados 4 bytes.

O tratamento das mensagens ocorre diferentemente no medidor e controlador. No controlador, como já comentado, é utilizado uma thread para o tratamento das mensagens porém no medidor o tratamento das mensagens ocorre em seu loop principal, através de *polling*. Enquanto no controlador ao receber uma mensagem a thread sai do estado IDLE e inicia o tratamento, no controlador ele fica em loop tentando ler mensagens pela serial, somente saindo deste loop quando um dos timers tem sua função de retorno chamada. Ao final, todos os comandos foram implementados com sucesso.

5.3 COMUNICAÇÃO WEBSITE-CONTROLADOR

Semelhante ao ensaio anterior, busca-se avaliar a comunicação entre os módulos, porém desta vez entre o website e o controlador. O único componente utilizado será o controlador, pois ele é responsável por hospedar o website.

O teste terá como entrada o website, no qual será simulada a navegação do usuário. Neste teste os dados mostrados no website serão previamente adicionados no banco de dados. Como resultado, espera-se que o website esteja integrado ao controlador e ao banco de dados. Quando ações que exigem atualização dos dados no website forem requisitadas pelo usuário, espera-se que seja sinalizado no website que a ação foi completada.

Com este teste as seguintes etapas serão avaliadas:

- O website possui todas as ferramentas necessárias para cumprir os requisitos do projeto.

- A informação entre website e servidor (controlador) está sendo transmitida, interpretada e armazenada quando necessário.
- A informação presente no banco de dados está sendo recuperada e transmitida ao usuário corretamente.

Resultados:

Como já comentado, para implementar o lado do servidor no controlador, foi utilizado o framework cherrypy. Para o projeto foram definidas duas páginas, sendo a primeira a página principal contendo uma lista com os medidores ativos, e a segunda a página do consumo de cada medidor. Para o lado da página web foi utilizado javascript para manipular os dados e os elementos da página. Para transmitir a informação entre página web e servidor, foi utilizado o formato JSON.

A página inicial pode estar em dois estados. O primeiro é a de uma página em branco. Este estado ocorre quando não há nenhum medidor presente na rede. O segundo estado representa uma lista com os medidores ativos.

A página do consumo foi dividida em duas colunas. A da esquerda pode possuir até quatro gráficos, todos de horário por consumo. Estes gráficos representam o consumo daquele medidor nos últimos minutos, a cada 10 minutos passados, nas últimas horas e nos últimos dias. Para cada gráfico foi definido uma quantidade máxima de valores, dependente da resolução temporal. Por exemplo, para o gráfico que representa o consumo nos últimos minutos, são mostradas as 20 medições mais recentes, enquanto que para o gráfico de horas as 12 últimas horas são mostradas. A figura 37 representa o gráfico de minutos retirado do website já em funcionamento.

É no lado direito que o usuário pode realizar as ações de gerenciamento. Neste lado estão presentes os botões para ligar ou desligar o fornecimento de energia naquele medidor, é onde o usuário pode obter quanto está sendo utilizado de energia naquela tomada e onde ele pode programar os períodos de funcionamento de cada medidor. Neste lado também está presente o botão de atualizar, que atualiza o estado do medidor e a informação dos gráficos na coluna da esquerda. A figura 38 representa como estão representados estes elementos.

Para este teste inicialmente foram utilizados valores previamente inseridos no banco de dados. Estes valores foram recuperados pelo website com sucesso. Para os elementos que requerem atualização de rotinas ou alteração do estado de cada medidor, o teste realizado mostrou resultado no prompt de comando do controlador e na informação presente no banco de dados.



Figura 37: Gráfico de consumo minuto a minuto

Fonte: autoria própria

5.4 COMANDOS DO USUÁRIO

Por ser o último teste, este será uma avaliação do sistema completo. Todos os módulos estarão envolvidos. Busca-se neste teste comprovar o funcionamento do sistema. Através do website, o usuário deverá ser capaz de gerenciar os módulos medidores. Todas as funcionalidades do gerenciamento serão testadas.

Como resultado espera-se:

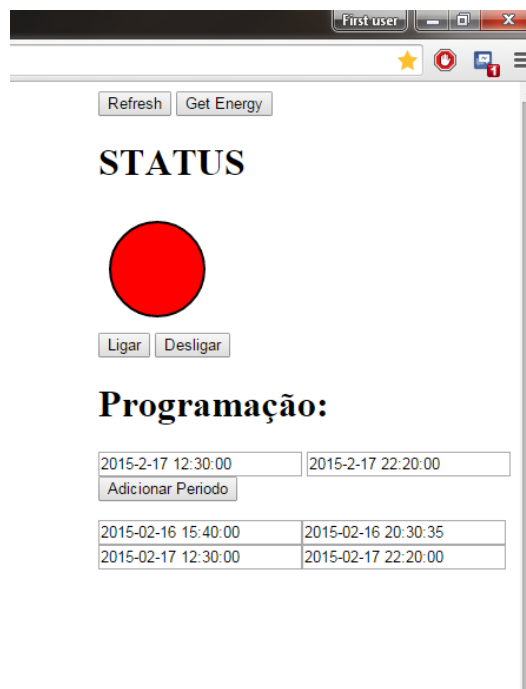


Figura 38: Elementos da página web referentes ao gerenciamento do sistema

Fonte: autoria própria

- Ler as medições de consumo dos medidores.
- Programar rotinas de funcionamento para os medidores.
- Averiguar se um medidor está autorizando ou negando o acesso à energia.
- Negar e Autorizar o acesso à energia em um medidor de forma imediata.

Resultados:

A integração dos módulos foi implementada sem grandes dificuldades, graças aos testes e preparo feito nas etapas anteriores. Uma demonstração total do sistema em funcionamento está disponível através do link

http://youtu.be/G0ryaw7 – AG4

6 CONCLUSÃO

Ao final do trabalho, os resultados encontrados foram satisfatórios em relação ao proposto inicialmente. O propósito do trabalho de mostrar ao usuário como ele utiliza a energia elétrica residencial foi atingido, juntamente com a capacidade básica de gerenciamento deste consumo. Apesar dos desafios encontrados e a possibilidade de melhorias em todas os módulos do projeto, os objetivos foram alcançados.

Ao final do trabalho, foi utilizada uma PCI(Placa de Circuito Impresso) padrão, como suporte para desenvolver o módulo medidor. O resultado final está na figura 39. Nesta figura também estão representados todos os módulos do medidor, sendo eles os sensores de tensão e corrente, o Arduino como processador, o Xbee para a comunicação, o regulador de tensão de 5V para 3.3V e o sistema para o acionamento do relê.

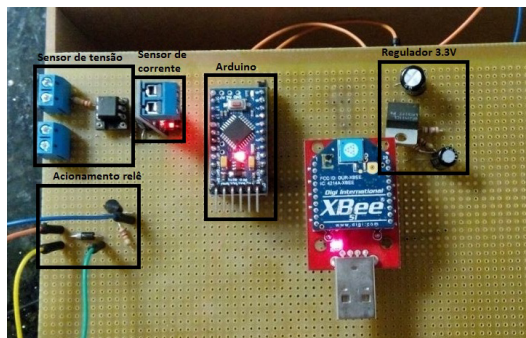


Figura 39: PCI contendo o módulo medidor

Fonte: autoria própria

A figura 40 representa o estado final do controlador do sistema, sendo composto pelo Raspberry Pi e Xbee.

A tabela 7 representa os custos e a descrição individual de cada componente.

Ao final do projeto, conclui-se que o custo do controlador do sistema é de R\$340,00, e o custo de cada módulo medidor é de R\$175,00. Ao trocar os componentes de comunicação por componentes mais simples, como o MX-FS-03V e MX-05V, pode-se diminuir o custo em



Figura 40: Raspberry Pi conectado ao Xbee

Fonte: autoria própria

Tabela 7: Custos dos componentes do projeto

Componente	Descrição	Custo (R\$) por unidade	Alternativas
Arduino	Processador do módulo de medição	15,00	lpc1768
Xbee Série 1	Utilizado para a comunicação entre medidores e controlador. Utiliza-se um para cada medidor, e um para o controlador. No projeto foram utilizados dois ao todo.	100,00	bluetooth, MX-FS-03V e MX-05V
Regulador de tensão	Utilizado para transformar a tensão de 5V para 3.3V, para então alimentar o xbee	4,00	-
Sensor de corrente	Foi utilizado o TA12-100 para medir a corrente	15,00	ACS712
Sensor de tensão	Foi utilizado o optoacoplador 4N25	6,00	4N35, divisor de tensão capacitivo
Relê	O relê e os componentes relacionados a ele, foram utilizados para o controle de acesso a energia	5,00	-
Raspberry Pi	Utilizado como controlador do sistema	240,00	computador de mesa, kit de desenvolvimento lpc1768

Fonte: autoria própria

cada medidor e no controlador em R\$80,00. O sistema mostrou-se ser mais barato que as soluções disponíveis no mercado. Enquanto cada módulo medidor do sistema desenvolvido

custa R\$175,00, os medidores da marca INSTEON custam U\$39,99, com impostos mais frete para o Brasil totaliza aproximadamente R\$320,00. Já os módulos do meterplug custam U\$49,99 e atualmente não são enviados para o Brasil.

Outro ponto importante está relacionado à eficiência energética do produto. Em estado de operação total, ou seja, quando todos os componentes estão executando alguma tarefa, o consumo do medidor atinge 200mA. Nestes casos, uma vez que o objetivo do produto é de diminuir o consumo de energia elétrica, aconselha-se que somente sejam acoplados medidores a tomadas nas quais esteja sendo utilizada, pela carga, uma corrente acima de 200mA.

6.1 DESAFIOS ENCONTRADOS

Por tratar-se de um projeto que envolve diferentes áreas do conhecimento e a integração entre elas, desafios eram esperados. Estes foram separados de acordo com os módulos do trabalho respectivos:

Módulo medidor: Os principais desafios ocorreram com os componentes que envolvem o uso de hardware. Em relação ao módulo medidor, o maior desafio foi a fonte de alimentação. Por ser essencial ao módulo, este foi o primeiro problema a ser abordado. A dificuldade encontrava-se no fato da fonte ter que seguir os requisitos do medidor. Ela deveria ser pequena, o que eliminou a possibilidade de uma fonte com transformador, fornecer ao menos 200mA, o que descartou fontes baseadas em diodos zener sem transformador. A solução encontrada foi utilizar fontes comerciais, geralmente utilizadas como carregadores portáteis.

Em seguida precisava-se medir a tensão e corrente, para poder se calcular o consumo. O problema encontrado foi como conseguir medir ambos, utilizando componentes que suportassem alta tensão (até 380V) e corrente muito mais alta que a utilizada pelo próprio sistema (até 5A). A melhor solução encontrada para medir a tensão foi o uso de um optoacoplador (no sistema foi utilizado o 4N25), desta forma além deste ser o único ponto do sistema na qual existe a presença de alta tensão, não há perdas de energia na alternativa (utilização de um divisor de tensão). Em relação a medição de corrente, a maior dificuldade foi calibrar o medidor.

No caso da medição da corrente, a precisão era um fator crítico, uma vez que para a tensão seu valor geralmente não difere significativamente em relação ao esperado. Como o valor lido é comparado sempre a 5V, caso a fonte de alimentação fornecesse mais ou menos tensão ao sensor de corrente o valor da medição seria alterado proporcionalmente, uma vez que a medida é proporcional ao valor da fonte de alimentação do sensor. Além disso podem ocorrer pequenas variações na tensão fornecida pela fonte ao longo do ciclo de funcionamento.

Para tratar estes problemas, o medidor realiza calibrações periódicas, considerando a tensão fornecida pela fonte de alimentação naquele instante de tempo. Para esta calibração, são feitas durante um segundo leituras dos valores máximos e mínimos da saída do sensor de corrente. Estes valores representam a corrente alternada da rede elétrica residencial. Com os valores máximo e mínimo, calcula-se:

$$pontozero = maximo - \frac{maximo - minimo}{2}$$

Sendo:

- máximo: Valor máximo, medido pelo Arduino, da saída do sensor de corrente, variando entre 511 e 1024.
- mínimo: Valor mínimo, medido pelo Arduino, da saída do sensor de corrente, variando entre 0 e 511.

O ponto zero representa o valor ao qual os valores medidos posteriormente serão comparados. Desta forma quando for necessário calcular a corrente, será calculada a diferença do valor máximo medido e do ponto zero. Esta diferença representa o valor da corrente calculada.

Integração: Por ser um projeto que envolve diferentes componentes, e comunicação de duas vias entre eles, esperava-se encontrar dificuldades no interfaceamento entre os módulos. A primeira dificuldade foi nas escolhas dos componentes. Nesta etapa foram considerados os seguintes fatores:

- A facilidade no acesso dos possíveis componentes;
- A experiência prévia com os componentes em trabalhos anteriores;
- O custo e tamanho(para o medidor e controlador) destes componentes.

Com os componentes em mãos, encontrou-se dificuldade principalmente na implementação do servidor (no controlador). Apesar de, ao início do projeto, ter-se conhecimento básico com a linguagem python, o framework utilizado (cherrypy) para a implementação do servidor apresenta peculiaridades até então não conhecidas. Apesar da documentação fornecida no website oficial do framework (TEAM, 2001) ter sido de grande ajuda, foi necessário o acesso a fóruns para complementar a informação.

6.2 CONTRIBUIÇÕES DO TRABALHO

Considera-se que, a maior contribuição do trabalho é a capacidade de auxiliar na boa utilização da energia elétrica. Como já descrito na seção 1.2, o racionamento de energia está cada vez mais necessário, com previsões de se agravarem ao longo dos próximos anos. Com o sistema projetado, é possível ter acesso do consumo elétrico em cada tomada que possui um módulo medidor, em qualquer lugar, provido o acesso a internet. Além disso pode-se gerenciar o fornecimento de energia de cada uma destas tomadas, especificando períodos de funcionamento de forma individual.

Como contribuições menores, podem ser citados:

- Troca de dados entre diferentes microcontroladores, baseados em diferentes linguagens de programação através de um canal de comunicação sem fio;
- A capacidade de medir tensão e corrente utilizando componentes pequenos e de baixo custo;

6.3 SUGESTÕES DE TRABALHOS FUTUROS

Apesar do sistema ter se comportado conforme o esperado, ele possui espaço para melhoras. Como melhorias ao projeto sugere-se:

- Criação da fonte de tensão para substituir as fontes comerciais utilizadas;
- Implementar o tratamento de problemas relacionados à falta de energia, uma vez que se falte energia, o sistema irá reiniciar e perderá informações prévias.
- Otimizar o uso de energia, programando o microprocessador de cada medidor para utilizar modo *sleep* quando nenhuma tarefa está sendo executada.
- Desenvolver mais elementos para ampliar para um sistema de automação residencial, como por exemplo controle de temperatura automática;
- Ampliar o website para suportar os novos módulos de automação residencial;
- Adicionar funcionalidades ao website:
 - Gerar relatórios de consumo de energia
 - Envio automático de relatórios gerados

- Melhorias no design do website
- Substituir os Xbee's por módulos de radiofrequência AM para a transmissão dos dados, diminuindo os custos do projeto.

REFERÊNCIAS

- ADMINISTRATION, U. E. I. **Internation Energy outlook 2007**. 2007. [Http://www.eia.gov/forecasts/ieo/index.cfm](http://www.eia.gov/forecasts/ieo/index.cfm). Disponível em: <<http://www.eia.gov/forecasts/ieo/index.cfm>>.
- ALLAMARAJU, S. **RESTful Web Services Cookbook: Solution for Improving Scalability and Simplicity**. [S.l.]: O'Reilly, 2010.
- ANALOGDEVICES. **Atividade 22. Optoacopladores**. 2013. Disponível em: <<http://wiki.analog.com/university/courses/electronics/electronics-lab-22>>.
- ANEEL. **Bandeira Tarifárias**. 2013. Disponível em: <<http://www.aneel.gov.br/area.cfm?idArea=758>>.
- ARDUINO. **Arduino Pro Mini**. 2008. Disponível em: <<http://arduino.cc/en/Main/ArduinoBoardProMini>>.
- ARDUINO. **Biblioteca SoftwareSerial**. 2014. Disponível em: <<http://www.arduino.cc/en/Reference/SoftwareSerial>>.
- BAKKER, D. M. G. D. M. **Bluetooth End to End**. [S.l.]: Wiley, 2002.
- BRADBURY, B. E. A. **Learning python with Raspberry Pi**. [S.l.]: John Wiley & Sons, 2014.
- CAPELLI, A. **Fontes de Alimentação**. [S.l.: s.n.], 2006.
- CROCKFORD, D. **ECMA-404 The JSON Data Interchange Standard**. 2006. Disponível em: <<http://json.org>>.
- DIGI. **Especificações Xbee**. 2015. Disponível em: <<http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/point-multipoint-rfmodules/>>.
- ELECFREAKS. **Corrente de sensor não invasivo TA12-100**. 2013. Disponível em: <http://www.electfreaks.com/wiki/index.php?title=Octopus_Non-invasive_AC_current_sensor_%28TA12-100%29_Brick>.
- ELÉTRICA, A. N. de E. **Atlas de Energia Elétrica no Brasil, Parte 1 Capítulo 2**. 2009. Disponível em: <http://www.aneel.gov.br/arquivos/pdf/atlas_par1_cap2.pdf>.
- ENERGATEINC. **In-home energy displays**. 2010. Disponível em: <<http://www.energateinc.com/our-products/home-energy-displays/>>.
- (EPE), E. de P. E. **Balanco Energético Nacional 2008: Ano base 2007: Resultados Preliminares**. 2008.
- (EPE), E. de P. E. **Projeção da demanda de energia elétrica para os próximos 10 anos (2011-2020)**. Fevereiro 2011. Disponível em: <http://www.epe.gov.br/mercado/Documents/S%C3%A9rie%20Estudos%20de%20Energia/20110222_1.p>

- FALUDI, R. **python-xbee overview**. 2010. Disponível em: <<https://code.google.com/p/python-xbee/>>.
- FONTES, M. B. A. **Reguladores de tensão**. 2006. Disponível em: <<http://www.lsi.usp.br/bariatto/fatec/aca/aula4-reguladores.pdf>>.
- GARCIA, G. **Brasil enfrenta a pior crise energética da história**. Janeiro 2015. Disponível em: <<http://noblat.oglobo.globo.com/geral/noticia/2015/01/brasil-enfrenta-pior-crise-energetica-da-historia.html>>.
- METERPLUG. **MeterPlug**. 2013. Disponível em: <<https://www.sticknfind.com/MeterPlug/>>.
- MICROCHIP. **Transformerless power supplies: Resistive and Capacitive**. 2009. Disponível em: <<http://ww1.microchip.com/downloads/en/AppNotes/00954A.pdf>>.
- NISKIER, J. **Manual de Instalações Elétricas**. Primeira edição. [S.l.]: LTC, 2014.
- NXP. **mbed LPC1768**. 2014. Disponível em: <<http://developer.mbed.org/platforms/mbed-LPC1768/>>.
- OPENENERGYMONITOR. **CT sensor - Interfacing with Arduino**. 2012. Disponível em: <<http://openenergymonitor.org/emon/buildingblocks/ct-sensors-interface>>.
- PEASE, R. A. **Troubleshooting Analog Circuits**. [S.l.]: Newnes, 1991.
- RAPP, A. **xbee-arduino Overview**. 2014. Disponível em: <<http://code.google.com/p/xbee-arduino/>>.
- ROGERCOM. **Zigbee protocol**. 2008. Disponível em: <<http://www.rogercom.com/ZigBee/ZigBeePag03.htm>>.
- SMARTHOME. **IMeter Solo**. 2011. Disponível em: <<http://www.insteon.com/imeter-solo>>.
- TEAM, T. C. **CherryPy: a minimamini python web framework**. 2001. Disponível em: <<http://www.cherrypy.org>>.
- VISHAY. **Optocoupler, Phototransistor Output, Wit Base Connection 4N25/4N26/4N27/4N28**. 2004. Disponível em: <<http://pdf.datasheetcatalog.com/datasheet/vishay/83725.pdf>>.
- VOITCH, G. **Apagão de 2001 deu prejuízo de R\$54,2 bilhões**. 2009. Disponível em: <<http://www.gazetadopovo.com.br/vida-e-cidadania/apagao-de-2001-deu-prejuizo-de-r-542-bilhoes-bzjwzmng5707utyhm0gmbdyku>>.
- WILKENFELD, G. **Smart Meters, displays and appliances**. 2013. Disponível em: <<http://www.yourhome.gov.au/energy/smart-meters-displays-and-appliances>>.

APÊNDICE A - DIAGRAMA DE FUNCIONAMENTO GERAL DO SISTEMA

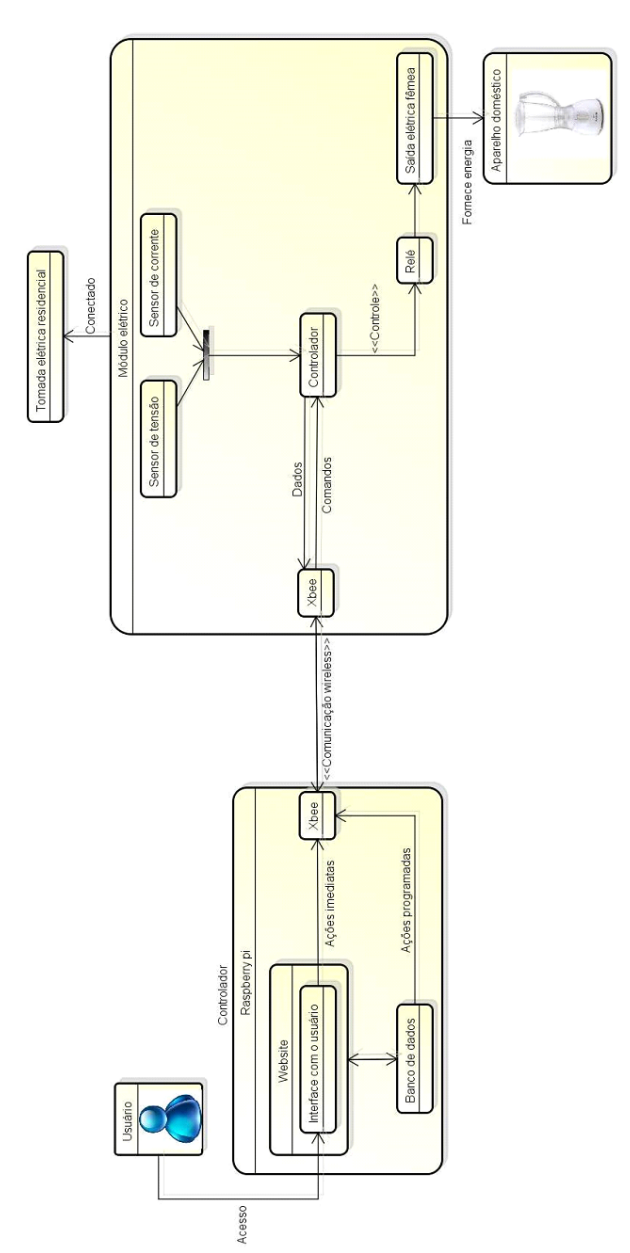


Figura 41: Diagrama de funcionamento geral do sistema

Fonte: autoria própria

APÊNDICE B – ARQUITETURA FUNCIONAL

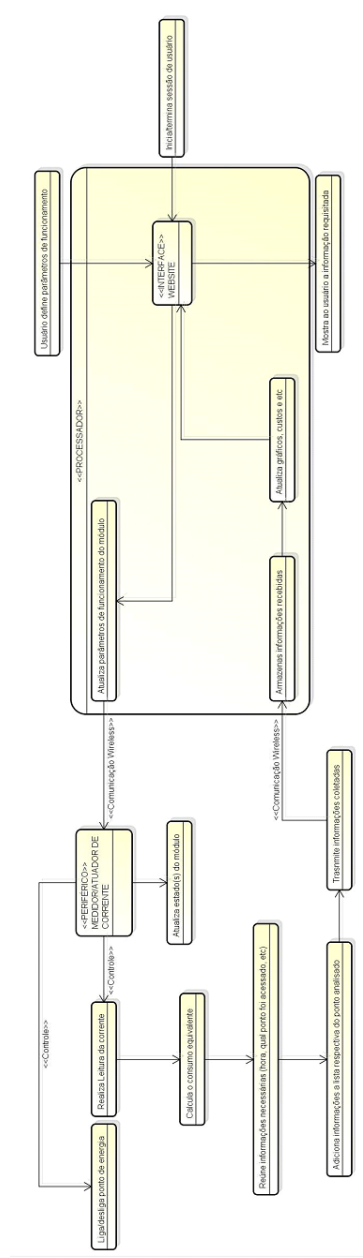


Figura 42: Arquitetura funcional

Fonte: autoria própria

APÊNDICE C – MÁQUINA DE ESTADOS DO FIRMWARE DO MEDIDOR

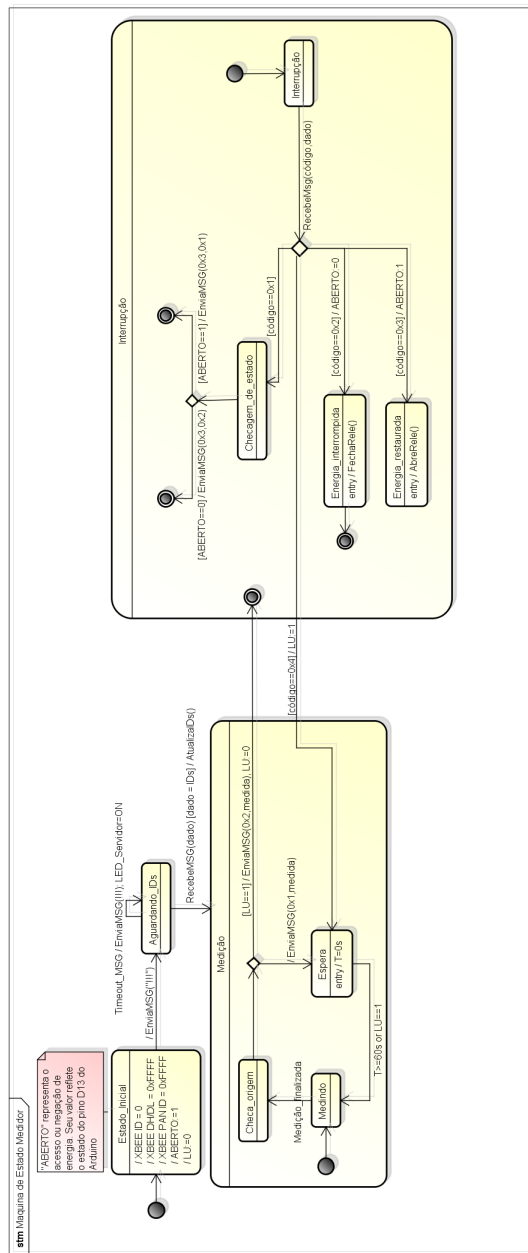


Figura 43: Máquina de Estados do firmware do medidor

Fonte: Autoria própria

APÊNDICE E – DIAGRAMA DE SEQUÊNCIA DA INICIALIZAÇÃO DA PÁGINA WEB.

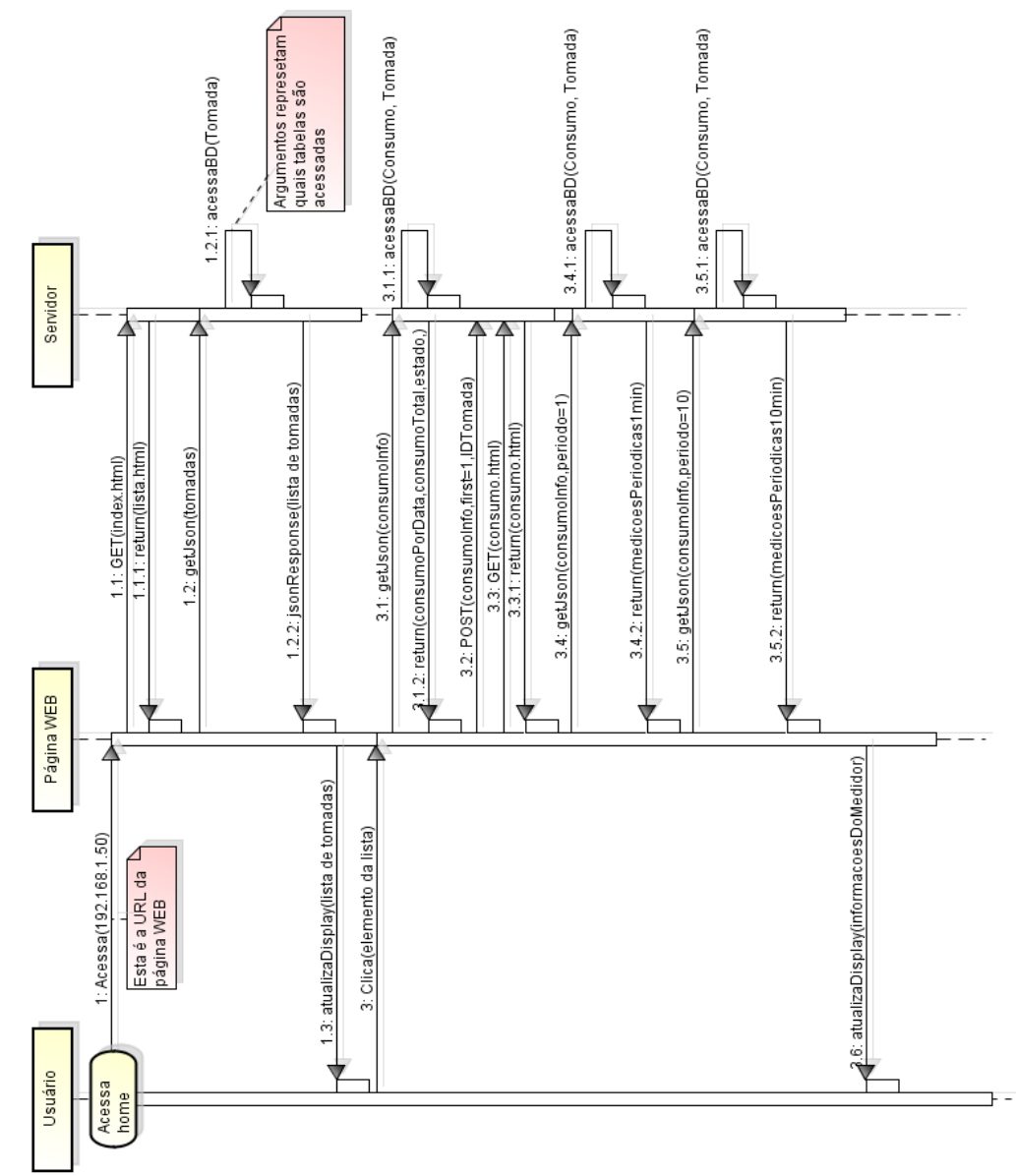


Figura 45: Diagrama de sequência da inicialização da página web.

Fonte: Autoria própria

APÊNDICE F – DIAGRAMA DE SEQUÊNCIA DO TRATAMENTO PELO SERVIDOR DAS AÇÕES DO USUÁRIO.

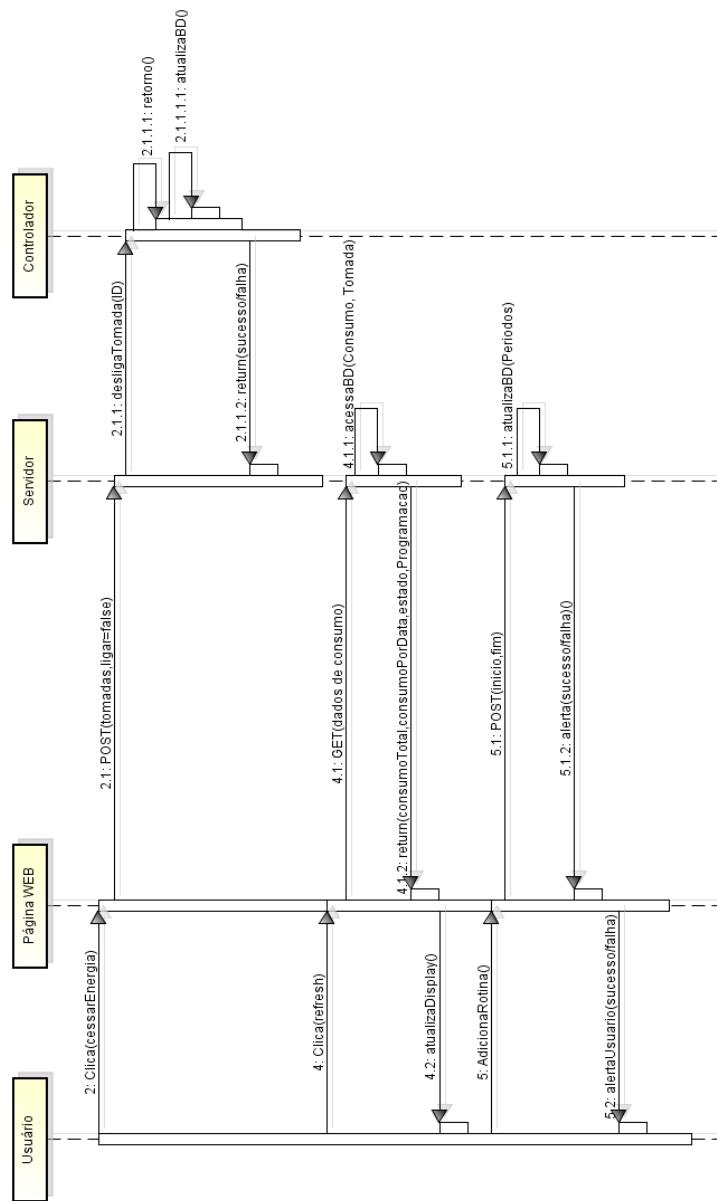


Figura 46: Diagrama de sequência do tratamento pelo servidor das ações do usuário.

Fonte: Autoria própria