

Centro Federal de Educação Tecnológica de Minas Gerais
Engenharia Mecatrônica

**Desenvolvimento de um sistema inteligente para automação
residencial - Smart House**

Elias Vieira de Sena
Rafael Caetano Paiva

Divinópolis - 2018

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
Campus DIVINÓPOLIS
GRADUAÇÃO EM ENGENHARIA MECATRÔNICA

Elias Vieira de Sena
Rafael Caetano Paiva

DESENVOLVIMENTO DE UM SISTEMA INTELIGENTE PARA
AUTOMAÇÃO RESIDENCIAL - SMART HOUSE

Divinópolis
2018

Elias Vieira de Sena
Rafael Caetano Paiva

DESENVOLVIMENTO DE UM SISTEMA INTELIGENTE PARA
AUTOMAÇÃO RESIDENCIAL - SMART HOUSE

Monografia de Trabalho de Conclusão de Curso
apresentado ao Colegiado do curso de Graduação
em Engenharia Mecatrônica como parte dos requi-
sitos exigidos para a obtenção do título de Enge-
nheiro Mecatrônico.

Áreas de integração: Eletrônica e Computação.

Orientador: Prof. Me. Daniel Moraes dos Reis

Co-orientador: Prof. Me. Marlon Henrique Tei-
xeira

Divinópolis
2018

Elias Vieira de Sena
Rafael Caetano Paiva

DESENVOLVIMENTO DE UM SISTEMA INTELIGENTE PARA
AUTOMAÇÃO RESIDENCIAL - SMART HOUSE

Monografia de Trabalho de Conclusão de Curso
apresentado ao Colegiado do curso de Graduação
em Engenharia Mecatrônica como parte dos requi-
sitos exigidos para a obtenção do título de Enge-
nheiro Mecatrônico.

Eixos de formação: Eletrônica e Computação.

Comissão Avaliadora:

Prof. Me. Daniel Morais dos Reis
CEFET-MG *Campus V*

Prof. Dr. Alisson Marques da Silva
CEFET-MG *Campus V*

Prof. Me. Alan Mendes Marotta
CEFET-MG *Campus V*

Divinópolis
2018

(Catalogação - Biblioteca Universitária – Campus Divinópolis – CEFET-MG)

S474d Sena, Elias Vieira de.

Desenvolvimento de um sistema inteligente para automação residencial *Smart House*. / Elias Vieira de Sena ; Rafael Caetano Paiva. - Divinópolis, 2018.

76f. : il.

Orientador: Prof. Me. Daniel Morais dos Reis.

Trabalho de Conclusão de Curso (graduação) – Colegiado de Graduação em Engenharia Mecatrônica do Centro Federal de Educação Tecnológica de Minas.

1. Eletrônica. 2. Computação. 3. Automação Residencial. 4. Sistema Inteligente. 5. *Wireless*. 6. Domótica. 7. Aplicação *Web*. I. Reis, Daniel Morais dos. II. Paiva, Rafael Caetano. III. Centro Federal de Educação Tecnológica de Minas Gerais. IV. Título.

CDU: 62(043)

“Que todos os nossos esforços estejam sempre focados
no desafio à impossibilidade. Todas as grandes
conquistas humanas vieram daquilo que
parecia impossível.”

Charles Chaplin

Agradecimentos

Eu, Elias, agradeço aos meus pais, Cláudio e Vandréia, por todo apoio, investimento, compreensão, amor e confiança que depositaram em mim durante todo o curso. Agradeço também meu avô, Antônio, e minhas avós, Elza e Marilene, que sempre me apoiaram e acreditaram em mim nessa longa caminhada.

Eu, Rafael, agradeço aos meus pais, Caetano e Irene, por todo apoio, investimento, compreensão, amor e confiança que depositaram em mim durante todo o curso. À minha namorada, Bárbara, agradeço pelo companheirismo, pela ajuda na execução do trabalho, por ter sorrido e sofrido ao nosso lado no decorrer do trabalho.

Nós agradecemos também aos nossos amigos de graduação que caminharam conosco ao longo de todo o curso, Pedro, Brenda, Vinícius, Paulo e Leonardo. Obrigado pela amizade, pelas risadas e, principalmente, pelo companheirismo nos momentos de sufoco que o curso nos trouxe.

Agradecemos aos nossos orientadores Daniel Moraes e Marlon Teixeira, por todo o auxílio durante a realização do trabalho, por todo o acompanhamento e pelos conhecimentos que recebemos nessa convivência, não só durante o TCC, mas em todo o curso. Por fim, agradecemos a instituição CEFET-MG, aos professores e aos funcionários por tornarem possível o sonho de nos tornarmos Engenheiros Mecatrônicos.

Resumo

Neste projeto foi desenvolvido um sistema eletrônico inteligente de comunicação para automação residencial que, através de uma central eletrônica, é capaz de realizar a comunicação entre morador e moradia. Desta forma, o projeto engloba duas das grandes áreas da engenharia mecatrônica: computação e eletrônica. Apesar da automação residencial já ser aplicada, a integração entre ambientes dificilmente acontece, visto que, na maior parte dos modelos existentes, não há comunicação entre os elementos domésticos. Entretanto, essa interação entre os dispositivos visa intensificar a experiência do usuário do sistema em sua residência. Desse modo, o Smart House foi projetado para atender a esta demanda. O sistema apresenta como elemento principal uma central eletrônica e como componentes secundários os módulos periféricos. Ambos são constituídos por sensores, atuadores, microcontroladores e circuitos eletrônicos, projetados para realizar a comunicação dentro do espaço residencial. Desta forma, os sensores fornecem dados para um monitoramento aliado ao microcontrolador, visando a tomada de decisão do usuário perante os atuadores. Além disso, o módulo central realiza a comunicação wireless com qualquer dispositivo do usuário conectado à rede e com os demais elementos da casa. Na sequência, uma Aplicação Web foi desenvolvida para promover a interação do sistema com o utilizador. A Web App (do inglês - *Web Application*) se comunica e recebe os dados provenientes da central eletrônica, via comunicação wireless. Portanto, o usuário pode, por exemplo, realizar o fechamento da janela ao ser avisado de que está chovendo em sua residência. A aplicação inicial do projeto foi realizada com iluminação e o controle da abertura e fechamento de janelas, embora o projeto esteja aberto a integração de novos elementos. Espera-se com o desenvolvimento desse projeto, contribuir para a comodidade do usuário em sua casa, elevando a praticidade envolvida na realização de tarefas cotidianas, visando em todos os aspectos, a facilidade, simplicidade e o dinamismo durante a utilização dos recursos do sistema, uma vez que esse tipo de tecnologia envolve tendências da automação.

Palavras-chave: Automação residencial; Sistema inteligente; Wireless; Aplicação Web; Domótica.

Abstract

In this project an intelligent electronic system of communication was developed for residential automation that, through an electronic headquarters, it is capable to accomplish the communication between the resident and home. This way, the project includes two of the great areas of the mecatronic engineering: computation and electronics. In spite of the residential automation already to be applied, the integration among atmospheres difficultly happens, because, in most of the existent models, there is no communication among the domestic elements. However, this interaction seeks to intensify the user's of the system experience in their residence. This way, Smart House was projected to assist their demand. This system presents as main element an electronic central and as secondary peripheral modules. These are constituted for sensors, actuators, microcontrollers and electronic circuits, projected to make the communication inside the residence. This way, the sensor just supply data for a monitoring allied to a microcontroller sending the user decision by a socket to actuators. The central module make the wireless communication with any device. An Aplicação Web was developed to promote the interaction of the system with the user. The initial application of the project was for control the illumination and windows, however the architecture is ready to be integrated to new devices. Is waited with the development of this project contribute to practicing involved in the execution of daily tasks, once this technology type involves a lot of automation tendencies.

Word-key: Residential automation; Intelligent system; Wireless; Application Web.

Sumário

Lista de Figuras	xii
Lista de Tabelas	xiii
1 Introdução	1
1.1 Definição do Problema	2
1.2 Motivação	2
1.3 Objetivos do Trabalho	4
1.3.1 Objetivo geral	4
1.3.2 Objetivos específicos	4
1.4 Organização do Documento	4
2 Fundamentos	6
2.1 Revisão de Literatura	6
2.2 Estado da Arte	7
2.3 Fundamentação Teórica	9
2.3.1 Automação Residencial	9
2.3.2 Rede de computadores	10
2.3.3 Comunicação cliente-servidor	10
2.3.4 Protocolos de rede	11
2.3.5 WiFi	14
2.3.6 Web Server	14
2.3.7 Banco de dados	15
2.3.8 HTML e CSS	17
2.3.9 Linguagem de Programação JAVA	17
2.3.10 ESP32	18
2.3.11 Sensor de Luminosidade - LDR	19
2.3.12 Módulo Dimmer	20
2.3.13 Sensor Pluviométrico YL-83	21
2.3.14 Motor de Passo	23
3 Metodologia	25
3.1 Seleção de materiais e métodos	26
3.2 Central eletrônica	31
3.2.1 Aplicação Web	31
3.2.2 Banco de dados	31

3.3	Módulos periféricos	32
3.3.1	Módulo de sensoriamento	33
3.3.2	Módulo de iluminação	34
3.3.3	Módulo de atuação da janela	37
3.4	Estruturas para comportar os módulos	39
3.4.1	Projetos	39
3.4.2	Construção	40
4	Resultados	41
4.1	Central Eletrônica	41
4.1.1	Aplicação Web	42
4.1.2	Banco de dados	46
4.2	Módulos periféricos	47
4.2.1	Módulo de sensoriamento	47
4.2.2	Módulo de iluminação	48
4.2.3	Módulo de atuação da janela	50
4.3	Comunicações	51
4.3.1	Comunicação interna da central	51
4.3.2	Comunicação entre central e módulos periféricos	51
4.3.3	Comunicação entre módulos periféricos	52
4.3.4	Comunicação do sistema com a rede externa	52
5	Considerações Finais	53
5.1	Conclusão	53
5.2	Propostas de Continuidade	54
	Referências	55
A	Complementos	1
A.1	Configuração do Raspberry	1
A.1.1	Instalação do sistema operacional	2
A.1.2	Configuração do IP estático	3
A.1.3	Instalação do Glassfish	4
A.1.4	Instalação do MariaDB	6

Lista de Figuras

1.1	Expectativa para o mercado mundial para equipamentos de casas inteligentes.	2
1.2	Potencial do mercado.	3
1.3	Potencial do mercado em números.	3
2.1	O advento das chamadas a distância.	6
2.2	Arquitetura proposta para o sistema.	9
2.3	Comunicação entre cliente e servidor.	10
2.4	Comunicação entre cliente e servidor - Solicitações e respostas.	11
2.5	Protocolo de rede.	12
2.6	Conexão TCP.	13
2.7	Interior da Camada de Rede.	13
2.8	Sistema de Banco de Dados.	16
2.9	Placa ESP32-DevKitC com controlador ESP32.	18
2.10	IDE Arduino com exemplo de programação.	19
2.11	Sensor LDR.	20
2.12	Diagrama de blocos do funcionamento do módulo dimmer.	21
2.13	Sensor Pluviométrico YL-83.	22
2.14	Modo de operação <i>Half Driver</i> para Motor de Passo.	24
3.1	Diagrama do projeto.	25
3.2	Fonte Hi Link 5V.	27
3.3	Fonte com conector micro USB.	28
3.4	Janela prototipada - Vista em perspectiva frontal.	28
3.5	Janela prototipada - Vista em perspectiva posterior.	29
3.6	Motor de Passo e Driver escolhidos.	30
3.7	Diagrama do Banco de Dados desenvolvido.	32
3.8	Circuito do módulo de sensoriamento.	34
3.9	Circuito do módulo de iluminação.	35
3.10	Circuito do módulo de dimmer.	36
3.11	Circuito do módulo de atuação de janela.	38
3.12	Vista em perspectiva com foco na parte superior da peça.	39
3.13	Vista em perspectiva da solução utilizada para encapsular os módulos.	40
4.1	Case da central do sistema.	41
4.2	Página de login.	42
4.3	Página de cadastro.	43

4.4	Página Home.	44
4.5	Página de um cômodo.	45
4.6	Página de controle geral da residência.	46
4.7	Dados de registros de uso retornados do banco de dados usando join.	47
4.8	Case do módulo de sensoriamento do sistema.	48
4.9	Case do módulo de iluminação do sistema.	50
A.1	Raspberry Pi 3 Modelo B.	2
A.2	<i>NOOBS</i> disponível para download.	3
A.3	Arquivo "rc.local" editado.	4
A.4	Arquivo "resolv.conf" editado.	4
A.5	Teste do Glassfish.	6

Lista de Tabelas

2.1	Especificações Técnicas do Sensor Pluviométrico.	23
3.1	Características do ESP32. Fonte: [1]	26
3.2	Características do Motor de Passo e Driver.	30
3.3	Orçamento da construção dos módulos projetados.	40

Introdução

A tecnologia tem evoluído a um ritmo acelerado e aliada a busca por conforto, acessibilidade e segurança, vem fazendo com que as residências possuam cada vez mais sistemas automatizados [2]. A aplicação de tecnologia no ambiente doméstico é conhecida como automação residencial, e tem sido uma das tendências de destaque mundial que tem conquistado grande relevância no setor da engenharia.

Atualmente, o emprego deste modelo de tecnologia é realizado em duas arquiteturas diferentes: (i) uma baseada na automação individual de elementos; (ii) outra na integração entre os mesmos dentro do ambiente. Embora este último esteja relacionado com um grande sucesso de mercado [3], sua complexidade faz com que o sistema individual seja mais utilizado [4].

No Brasil, existe um potencial para o fornecimento de equipamentos para 1,8 milhão de residências, estimando-se que cerca de 300 mil já os possuam [5]. Portanto, existe ainda um mercado inexplorado de pelo menos 1,5 milhão de moradias. Porém, esses dados representam apenas uma parcela da população com um poder aquisitivo mais elevado, uma vez que o interesse dos consumidores de um modo geral, gira em torno de 78% [5]. Desta forma, a implementação de sistemas integrados em residências com o uso de aplicativos e comunicação wireless, como proposto neste trabalho, busca levar comodidade ao usuário do sistema utilizando a tecnologia para facilitar e tornar mais dinâmica a realização de tarefas cotidianas.

Tendo todo quadro descrito acima em consideração, este trabalho almeja o desenvolvimento de um sistema inteligente para automação residencial, que proporcione a interação entre usuário e residência utilizando comunicação wireless e uma Aplicação Web para dispositivos eletrônicos, que possibilitam a utilização de comandos remotos. Além disso, pretende-se que o sistema desenvolvido possua uma acessibilidade maior em relação aos já existentes com este tipo arquitetura. Desta forma, este trabalho contempla duas grandes áreas da engenharia mecatrônica: computação e eletrônica.

1.1 Definição do Problema

O problema fundamental no qual se sustenta este trabalho, refere-se à carência de sistemas inteligentes para automação residencial que apresentem arquitetura integrada entre os dispositivos domésticos e que sejam acessíveis, de um modo geral, à população brasileira.

1.2 Motivação

A motivação para o desenvolvimento deste projeto se sustenta na análise do mercado atual para o tipo de sistema desenvolvido e também nos efeitos sociais de melhoria na qualidade de vida promovidos por ele.

O mercado mundial apresenta uma expectativa de evolução significativa no que diz respeito à venda de produtos relacionados a automação residencial, segundo estudo feito por [6], uma das principais fontes de informação em áreas críticas que moldam o panorama empresarial atual. A Figura 1.1 apresenta a expectativa do cenário supracitado no período entre 2015 e 2020.

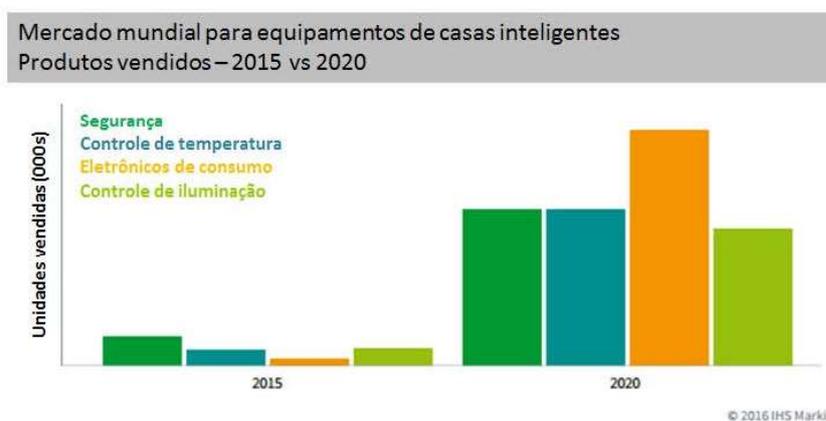


Figura 1.1: Expectativa para o mercado mundial para equipamentos de casas inteligentes.

Fonte: [6]

O mercado nacional também revela um grande potencial no ramo da automação residencial, uma vez que no país existe um número extremamente baixo de residências com automação se comparado ao total de residências, como mostra os resultados do levantamento de [5] apresentados nas Figuras 1.2 e 1.3.



Figura 1.2: Potencial do mercado.

Fonte: Adaptado de [5]

	EUA	ALE	REINO UNIDO	FRANÇA	ESPANHA	BRASIL (efetivo)	BRASIL (potencial)
TOTAL DE RESIDÊNCIAS (milhões)	134	38	27	25	17	63	63
RESIDÊNCIAS COM AUTOMAÇÃO (milhões)	24,0	6,5	4,4	4,5	2,6	0,3	1,9
RESIDÊNCIAS COM AUTOMAÇÃO (%)	18%	17%	16%	18%	15%	0,4%	3%

Figura 1.3: Potencial do mercado em números.

Fonte: Adaptado de [5]

No âmbito social, este trabalho busca através da inserção de tecnologia no ambiente residencial, levar ao usuário final:

- **Comodidade:** Uma vez que o ambiente pode ser controlado na tela de qualquer dispositivo do usuário que possua acesso à internet;
- **Praticidade e facilidade:** Visto que o ambiente realiza tomadas de decisões baseadas nas escolhas do usuário, como fechar uma janela em situações de chuva, ou acender luzes ao anoitecer;
- **Acessibilidade:** Uma vez que o projeto em questão será mais acessível ao público brasileiro, se comparado às arquiteturas integradas de [3] por exemplo.

1.3 Objetivos do Trabalho

Nestes termos, são objetivos deste projeto:

1.3.1 Objetivo geral

Desenvolver um sistema inteligente para automação residencial, que proporcione a interação entre usuário e residência, utilizando comunicação wireless e uma Aplicação Web que possibilite comandos remotos.

1.3.2 Objetivos específicos

- Projetar e construir o sistema eletrônico dos módulos periféricos;
- Selecionar o dispositivo que será utilizado como central e configurá-lo;
- Arquetetar e implantar os sistemas de sensoriamento relacionados à luminosidade e chuva;
- Projetar e construir os sistemas de abertura/fechamento da janela e de controle de luminosidade;
- Desenvolver a Aplicação Web para estabelecer a comunicação entre o usuário e o sistema;
- Implementar todo o sistema necessário para alocar a Aplicação Web na central eletrônica e hospedá-la na rede;
- Criar um banco de dados para armazenar as informações;
- Construir uma estrutura para comportar a central;
- Produzir as estruturas para comportar os módulos periféricos.

1.4 Organização do Documento

Este trabalho encontra-se particionado em cinco capítulos, sendo eles:

- **Capítulo 1:** Apresenta a definição do problema estudado, a motivação para a realização do projeto, os objetivos do trabalho e a organização deste documento.
- **Capítulo 2:** São caracterizados os conceitos fundamentais teóricos para o desenvolvimento deste trabalho.

- **Capítulo 3:** São apresentadas as etapas do trabalho desenvolvido, bem como seu andamento até o fim deste projeto.
- **Capítulo 4:** Refere-se aos resultados obtidos com base no desenvolvimento do trabalho.
- **Capítulo 5:** Corresponde às considerações finais, contando ainda com propostas para a continuidade deste trabalho.

Fundamentos

Neste capítulo serão apresentados os seguintes tópicos: revisão de literatura, estado da arte, fundamentação teórica.

2.1 Revisão de Literatura

A Revolução Industrial no século XVIII impulsionou o ramo da automação pelo mundo, baseando-se em processos de mecanização. Alguns destes processos ainda são utilizados em cadeias produtivas.

No início do século passado, os procedimentos de automação já se baseavam em dispositivos mecânicos e eletromecânicos. Desta forma, já era possível obter certas comodidades aos usuários destes sistemas, como visto na Figura 2.1 que apresenta o advento das chamadas a distância.



Figura 2.1: O advento das chamadas a distância.
Fonte: [7]

Na década de 1970, o surgimento de alguns problemas habitacionais relacionados à

qualidade dos espaços impulsionaram a utilização dos primeiros sistemas de automação a serem controlados de forma eletrônica nos EUA [7].

A década de 80 caracteriza-se pelo aparecimento dos primeiros termos voltados para a automação residencial, como *Smart House* e *Domótica*. Neste período havia uma grande tendência voltada para a economia de energia. Além disso, surgiram diversos sistemas de automação para segurança e iluminação, por exemplo. Ainda nesta época, houve a chegada da TV a cabo, de antenas via satélites e outros serviços de telecomunicações [7].

Nos anos 1990, a tendência da domótica foi a de integrar os diferentes sistemas (segurança, condicionamento ambiental, etc.) que compõem uma residência, com o objetivo de melhorar a produtividade e o conforto das pessoas. Conhecida como “Casa Sistema”, onde todos os elementos começam a se inter-relacionar tanto internamente como com o exterior [7].

Por volta de 2006, o segmento de automação crescia cerca de 20% ao ano e já era possível encontrar construtoras que ofereciam a tecnologia. A indústria de construção civil foi estimulada a incorporar soluções inovadoras de automação residencial em seus novos empreendimentos devido a evolução tecnológica dos diferentes sistemas domésticos, aliada ao crescimento explosivo da internet e aos resultados dos investimentos em infraestrutura de comunicações [5].

Em meados de 2012, a domótica já havia se modernizado bastante, principalmente devido a grande evolução da computação que permitiu uma maior facilidade de interação entre os sistemas de controle presentes nas residências e os sistemas operacionais dos *smartphones*. Além disso, com a popularização do sinal WiFi, praticamente extinguiu-se a necessidade de utilização de fios elétricos para comunicar os aparelhos e as centrais. Porém, mesmo com todo avanço tecnológico ocorrido, os sistemas mais robustos e com uma grande quantidade de recursos e funcionalidades permaneceram acessíveis apenas a uma pequena parcela da população devido ao alto custo.

2.2 Estado da Arte

Nos últimos anos, inúmeras pesquisas estão sendo realizadas a respeito da automação residencial, principalmente com relação à integração entre os dispositivos domésticos e a utilização de aplicações desenvolvidas para acessar os sistemas de automação de forma remota, através da internet.

Em [4] é possível constatar o aprimoramento de uma arquitetura para realização de automação residencial, conhecida como *House Management System* (HMS). Ela é capaz de integrar novas tecnologias de dispositivos eletrônicos de maneira simples e trabalhar com comunicações heterogêneas. Além disso, o aprimoramento realizado, permitiu que

arquitetura alcançasse um nível de abertura que a diferencia das tecnologias existentes no mercado. Embora o trabalho apresente provas em relação a robustez do software desenvolvido, os testes com dispositivos eletrônicos são superficiais e insuficientes para determinar se, na prática, o sistema é realmente aplicável, uma vez que foi realizado apenas um teste no qual se comandou duas lâmpadas. Além disso, as informações enviadas para o banco de dados não são aproveitadas.

No projeto de [8] o sistema desenvolvido é focado em uma central que reúne todas as informações relacionadas à automação do ambiente em questão. Todos os módulos do sistema comunicam-se apenas com ela através de cabos. Foi desenvolvida também, uma Aplicação Web como interface de comunicação do usuário com a central, ela permite que toda a automação seja comandada através de um dispositivo que esteja na rede local. O projeto atinge o que foi posto como objetivo, porém o próprio autor cita alguns problemas como o fato da dificuldade em estruturar as ligações, devido à grande quantidade de cabos e a ausência de possibilidade de controle através de redes externas (Internet).

No trabalho de [9] pode-se observar o desenvolvimento de um projeto de automação residencial com base em um sistema completo e sofisticado, visando um público de padrão econômico elevado. Este projeto apresenta uma ampla gama de elementos automatizados, além da presença de unidades de sensoriamento. Entretanto, o foco abordado neste trabalho define um público alvo restrito, uma vez que traz consigo dispositivos de alto valor agregado, limitando sua acessibilidade. Além disso, não houve implementação do projeto de forma física.

A pesquisa de [2] apresenta um trabalho que evidencia um breve estudo sobre a domótica e os componentes eletrônicos necessários para sua aplicação. No projeto do sistema, esboça-se um modelo barato e que possua uma interface simples que seja de fácil uso. O sistema desenvolvido apresentou-se estável durante a realização dos testes em laboratório. Contudo, ao fim do trabalho destaca-se a não utilização de sensores no ambiente, dentre eles, cita-se o sensor de temperatura. A arquitetura proposta pode ser observada na Figura 2.2.

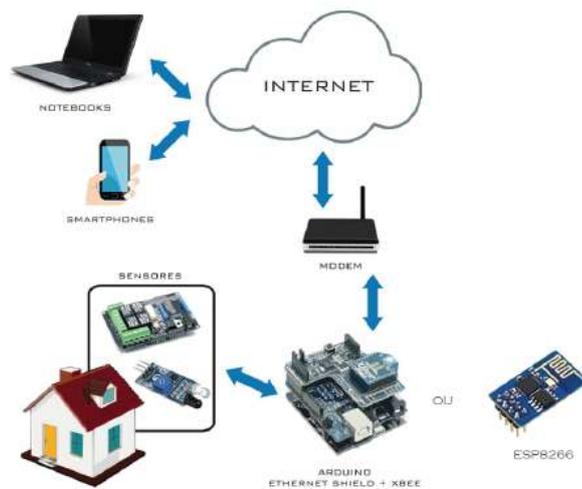


Figura 2.2: Arquitetura proposta para o sistema.
Fonte: [2]

Já em [10] é apresentado um aplicativo para *smartphone* que o torna um controle remoto universal para realizar a interação do usuário com diferentes dispositivos eletrônicos no ambiente residencial. Além disso, o sistema ainda possui mais duas camadas, sendo elas: (i) circuito eletrônico de atuação e (ii) o *middleware* que realiza a comunicação entre o controle remoto e os atuadores. O trabalho apresenta testes realizados com dois atuadores, porém, os testes são bastante simples e utilizam apenas duas lâmpadas para demonstrar o funcionamento. Além disso, o sistema desenvolvido não contempla nenhum tipo de sensoriamento do ambiente o que torna sua interação com o usuário limitada, uma vez que os dispositivos eletrônicos apenas recebem comandos de controle. Assim, o usuário não tem informações sobre o ambiente doméstico em seu aplicativo.

2.3 Fundamentação Teórica

Nesta seção serão abordados alguns conceitos necessários para a compreensão deste trabalho, os quais serão organizados na forma de tópicos.

2.3.1 Automação Residencial

Segundo [11] uma definição bastante interessante e que acrescenta também a ideia de integração de sistemas residenciais pode ser resumida pela seguinte definição: “É um processo que, usando diferentes soluções e equipamentos, possibilita ao usuário usufruir o máximo de qualidade de vida na sua habitação”.

Em [12] a automação residencial, tratada também como domótica, é abordada de forma mais ampla, como a automatização e o controle aplicados à residência. Esta automatização e controle se realizam mediante o uso de dispositivos que dispõem de capacidade para

se comunicar interativamente entre si e com capacidade de seguir as instruções de um programa previamente estabelecido pelo usuário da residência e com possibilidades de modificações conforme seus interesses.

Já [13] define a domótica como uma nova ciência com caráter multidisciplinar, agregando conceitos de Arquitetura, Engenharia, Ciência da Computação, Medicina, Sociologia e Psicologia, com a finalidade de estudar todas as necessidades do usuário diante das possibilidades oferecidas pelo mundo digital e suas interações com a residência automatizada.

2.3.2 Rede de computadores

Uma rede de computadores é formada por um conjunto de computadores interligados por um sub-sistema de comunicação. Através do qual, os computadores da rede são capazes de realizar trocas de informações e compartilhamento de recursos.

Dentre os tipos de comunicações em rede, destacam-se: (i) *uni* que caracteriza-se por permitir o fluxo de dados somente em um sentido, de acordo com [14]; (ii) *half* que consiste em uma conexão que possibilita o tráfego nos dois sentidos, porém, não de forma simultânea, segundo [14] e [15] e (iii) *full duplex* que é definida por [14] e [15] como uma conexão que permite tráfego simultâneo em ambos os sentidos.

2.3.3 Comunicação cliente-servidor

De acordo com [16], uma aplicação de rede consiste em pares de processos que enviam mensagens entre si por meio de uma rede. Para cada par de processos comunicantes atribui-se um processo cliente e um servidor. Desta forma, defini-se em [16] que o processo que inicia a comunicação é rotulado como cliente e o processo que espera ser contatado para o início do procedimento é o servidor.

Na Web, por exemplo, o browser inicia o processo de comunicação com um servidor Web, logo, o browser é definido como cliente e o servidor Web como servidor. A Figura 2.3 ilustra a comunicação entre cliente e servidor.

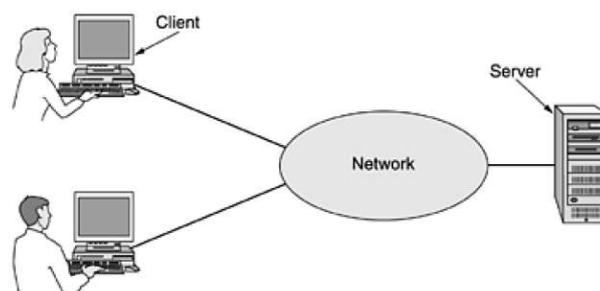


Figura 2.3: Comunicação entre cliente e servidor.

Fonte: [14]

Segundo [14], no modelo cliente/servidor ocorrem dois processos, sendo um deles na máquina cliente e outro na servidora. A comunicação toma a forma do processo cliente por meio do envio de uma mensagem pela rede ao processo servidor. A seguir, o cliente aguarda uma mensagem em resposta. Ao receber a solicitação, o servidor executa o trabalho solicitado ou realiza uma busca pelos dados e envia uma resposta ao cliente. Esse funcionamento pode ser observado na Figura 2.4.

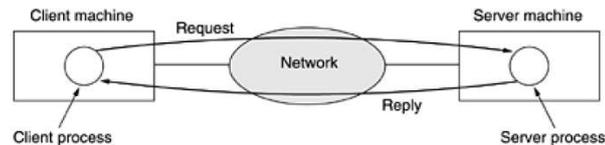


Figura 2.4: Comunicação entre cliente e servidor - Solicitações e respostas.

Fonte: [14]

2.3.4 Protocolos de rede

De acordo com [16], um protocolo de rede trata-se de uma troca de informações e realização de ações por meio de hardwares e softwares de algum equipamento, como por exemplo um computador. Todas as atividades em quaisquer redes de computadores que envolvem duas ou mais entidades remotas comunicantes são governadas por um protocolo.

Sendo assim, [16] define protocolo por: "Um protocolo define o formato e a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou no recebimento de uma mensagem ou outro evento".

A Figura 2.5 ilustra o funcionamento de um protocolo de rede.

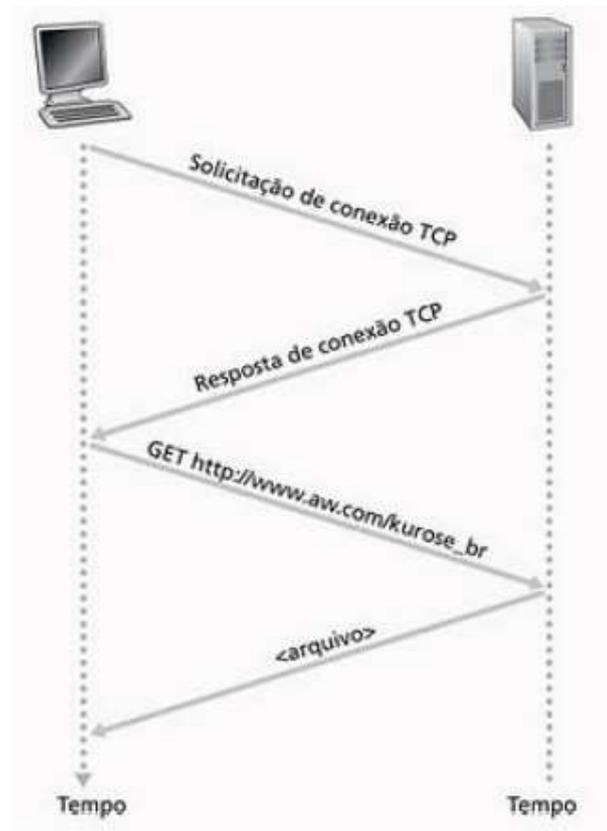


Figura 2.5: Protocolo de rede.

Fonte: [16]

TCP/IP

Segundo [17], os protocolos TCP (do inglês - *Transmission Control Protocol* - Protocolo de Controle de Transmissão) e IP (do inglês - *Internet Protocol* - Protocolo de Internet) são os responsáveis por organizar fisicamente o tráfego de dados pela internet.

O protocolo TCP é orientado para conexão, visto que antes que um processo de aplicação possa começar a enviar dados a outro, os dois devem enviar alguns segmentos preliminares entre si para definir os parâmetros de transferência de dados em questão [16].

Uma conexão TCP provê um serviço *full-duplex*, ou seja, caso haja uma conexão TCP entre o processo A em um hospedeiro e o processo B em outro hospedeiro, os dados da camada de aplicação poderão fluir ao mesmo tempo de A para B e de B para A. Além disso, a conexão TCP é sempre ponto a ponto, isto é, entre um único remetente e um único destinatário [16].

O processo que inicia uma conexão TCP é denominado processo cliente, enquanto o outro processo é chamado de servidor. Em uma conexão, na qual ocorrerá o envio de dados do cliente para o servidor, o processo cliente passa uma cadeia de dados através do *socket* (a porta do processo). Logo após, os dados chegam ao TCP que roda no

cliente, o qual direciona seus dados para o *buffer* de envio da conexão. Quando o TCP na outra extremidade recebe um pacote, os dados contidos neste são colocados no *buffer* de recepção da conexão. Feito isso, a aplicação realiza a leitura da cadeia de dados do *buffer* [16]. A Figura 2.6 ilustra o processo acima citado.

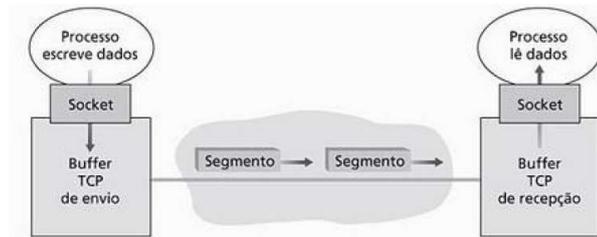


Figura 2.6: Conexão TCP.

Fonte: [16]

A internet possui dispositivos (computadores, celulares, entre outros) espalhados ao redor do mundo que disponibilizam serviços diversos a todo tempo. Para que estes dispositivos distintos, que rodam sistemas operacionais diferentes possam se comunicar, é preciso que tenham os mesmos padrões de comunicação. Neste âmbito o protocolo IP desempenha um papel fundamental, concedendo aos dispositivos presentes na rede um endereço IP globalmente único e de formato uniforme, o que possibilita que sejam identificados e, conseqüentemente, que a comunicação possa ocorrer [18].

Para entender o conceito de IP, é necessário considerar os três componentes que formam a camada de rede da internet, sendo eles: (i) o Protocolo IP; (ii) o componente de roteamento, que determina o caminho que um datagrama segue desde a origem até um destino; (iii) um dispositivo para comunicação de erros em datagramas e para atender requisições de certas informações da camada de rede. A Figura 2.7 contempla o interior da camada de rede.

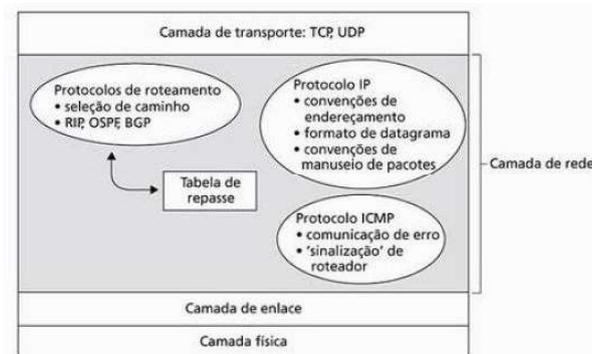


Figura 2.7: Interior da Camada de Rede.

Fonte: [16]

UDP

O protocolo de transporte UDP (*User Datagram Protocol*) é um dos protocolos originais da arquitetura da Internet [19]. Este protocolo adiciona somente a aplicação de multiplexação (usando números de porta) e *checksum* (protege os dados de adulteração de mensagem) para os dados antes de ser enviado por IP. Em comparação com o protocolo de transporte TCP, o UDP não fornece ordenação de dados, controle de fluxo ou confiabilidade e fornece dados enquadrados em datagramas, não como um fluxo contínuo como é o caso do TCP.

UDP é especialmente utilizado em aplicações que necessitam de desempenho em tempo real. Para alcançar um maior desempenho, o protocolo permite que os pacotes individuais possam ser descartados.

2.3.5 WiFi

Apesar de o termo WiFi ser uma marca registrada pela *Wi-Fi Alliance*, a expressão hoje se tornou um sinônimo para a tecnologia IEEE 802.11, que permite a conexão entre diversos dispositivos sem fio. Amplamente utilizado na atualidade, a origem do termo, diferente do que muito acreditam, não tem um significado específico. A expressão WiFi surgiu como uma alusão à expressão *High Fidelity (Hi-Fi)*, utilizada pela indústria fonográfica na década de 50.

As redes WiFi funcionam por meio de ondas de rádio. Elas são transmitidas por meio de um roteador, que recebe os sinais, decodifica e os emite a partir de uma antena. Para que um computador ou dispositivo tenha acesso a esses sinais, é preciso que ele esteja dentro um determinado raio de ação, conhecido como *hotspot*.

O raio de ação de uma rede WiFi é variável, de acordo com o roteador e a antena utilizada. Essa distância pode variar de 100 metros, em ambientes internos, a até 300 metros, em ambientes externos. O mesmo tráfego de dados ocorre no sentido oposto estabelecendo, assim, a comunicação entre os dispositivos.

2.3.6 Web Server

As pessoas utilizam a web a partir de sistemas finais, tais como: TV's, celulares, entre outros através de um software chamado navegador, isto é o sistema cliente. Ele recupera as informações em formato de documentos de texto, imagens e outros tipos de informação a partir de outros computadores, que são chamados de servidores [20]. O software servidor web atua na camada de aplicação, sendo responsável pela implementação do protocolo HTTP na porção servidora.

O servidor web tem a função de aceitar requisições de navegadores web, por meio de

uma conexão de rede. Ele decifra a requisição para determinar qual arquivo é solicitado, procura o arquivos e, se estiver disponível, o envia para o navegador web através da conexão de rede que ficou aberta [21].

Segundo [20], o servidor web pode ser entendido pela seguinte fórmula:

$$\textit{Servidor Web} = \textit{Plataforma} + \textit{Software} + \textit{Informações}$$

- **Plataforma:** É a parte que fornece a base para o servidor funcionar e está composta pelo computador (hardware), o sistema operacional e o software de rede;
- **Software:** O servidor web;
- **Informações:** A mais importante, as informações que o servidor contém. Sem elas o servidor web não teria sentido.

2.3.7 Banco de dados

Segundo [22], um banco de dados “é uma coleção de dados inter-relacionados, representando informações sobre um domínio específico”, ou seja, sempre que for possível agrupar informações que se relacionam e tratam de um mesmo assunto, posso dizer que tenho um banco de dados.

Já um Sistema Gerenciador de Base de Dados (SGBD) é uma coleção de programas que permitem aos usuários criarem e manipularem uma base de dados. Um SGBD é, assim, um sistema de software de propósito geral que facilita o processo de definir, construir e manipular bases de dados de diversas aplicações [23].

A base de dados e o software de gerenciamento da base de dados compõem o chamado Sistema de Banco de Dados que pode ser entendido como o conjunto de quatro componentes básicos: dados, hardware, software e usuários. Em [24] conceituou-se que “sistema de bancos de dados pode ser considerado como uma sala de arquivos eletrônica”. A Figura 2.8 apresenta um esquema genérico de um Sistema de Banco de Dados em sua interação com seus usuários.

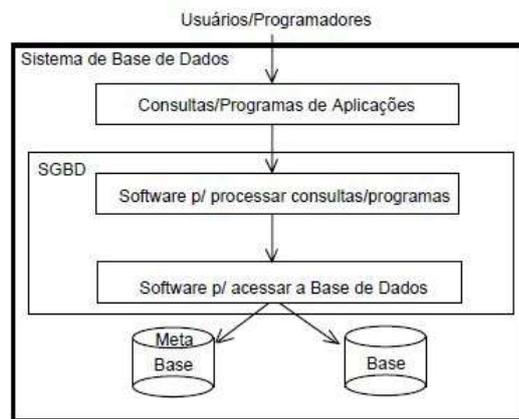


Figura 2.8: Sistema de Banco de Dados.
Fonte: [23]

Os objetivos de um sistema de banco de dados são o de isolar o usuário dos detalhes internos do banco de dados (promover a abstração de dados) e promover a independência dos dados em relação às aplicações, ou seja, tornar independente da aplicação, a estratégia de acesso e a forma de armazenamento.

Existem diversas arquiteturas para banco de dados, porém neste trabalho será utilizada apenas a cliente-servidor, portanto, somente ela será definida. A arquitetura cliente-servidor foi desenvolvida para separar ambientes de computação onde um grande número de computadores, estações de trabalho, servidores de arquivos, impressoras, servidores de banco de dados e outros equipamentos são conectados por uma rede. A ideia é definir servidores especializados, por exemplo, servidores de impressão que podem estar conectados a várias impressoras, assim, quando se desejar imprimir algo, todas as requisições de impressão são enviadas a este servidor. As máquinas clientes disponibilizam para o usuário as interfaces apropriadas para utilizar esses servidores, bem como poder de processamento para executar aplicações locais. Esta arquitetura se tornou muito popular por algumas razões: (i) a facilidade de implementação dada a clara separação das funcionalidades e dos servidores; (ii) um servidor é inteligentemente utilizado porque as tarefas mais simples são delegadas às máquinas clientes mais baratas; (iii) o usuário pode executar uma interface gráfica que lhe é familiar, ao invés de usar a interface do servidor. Desta maneira, a arquitetura cliente-servidor foi incorporada aos SGBDs comerciais [23].

Na arquitetura Cliente-Servidor, o cliente executa as tarefas do aplicativo, ou seja, fornece a interface do usuário (tela, e processamento de entrada e saída). O servidor executa as consultas no DBMS e retorna os resultados ao cliente. Apesar de ser uma arquitetura bastante popular, são necessárias soluções sofisticadas de software que possibilitem: o tratamento de transações, as confirmações de transações, desfazer transações, linguagens de consultas e gatilhos. A principal vantagem desta arquitetura é a divisão

do processamento entre dois sistemas, o que reduz o tráfego de dados na rede [23]. Neste projeto, será utilizada esta arquitetura e o banco de dados terá como função manter o cadastro dos usuários, dispositivos e todo o log de ações efetuadas.

2.3.8 HTML e CSS

O padrão HTML determina a forma pela qual um conjunto de elementos deve estar estruturado para compor um documento, assim como a definição de elos entre seus elementos e a inclusão de outras mídias no corpo de um documento. A navegação em documentos HTML se faz através das operações básicas definidas pela tecnologia de hipertextos. Entretanto, a simplicidade do padrão limita a estrutura de relacionamentos que pode ser associada aos hiperdocumentos [25].

Segundo [26], o descritor CSS (do inglês - *Curvature Scale Space*) foi proposto com o objetivo de realizar a representação de uma forma desenhada por um contorno de um objeto com base nos locais este contorno altera-se de côncavo para convexo e vice-versa. Desta forma, é possível fazer com que o contorno do objeto seja suavizado progressivamente, permitindo a construção de curvaturas totalmente convexas, auxiliando assim no design do objeto [26].

2.3.9 Linguagem de Programação JAVA

Java é uma linguagem de programação Orientada a objetos desenvolvida na década de 90 pela empresa Sun Microsystems. Diferentemente das linguagens convencionais, que são compiladas para código nativo, a linguagem Java é compilada para um *bytecode* que é executado por uma máquina virtual Java. Aplicações Java alcançam bons desempenhos se comparadas com aplicações desenvolvidas por linguagens que compilam em código nativo, proporcionando portabilidade entre vários tipos de processadores e sistemas operacionais [27].

J2EE

De acordo com [28], a plataforma J2EE é uma coleção de especificações de tecnologias relacionadas que descrevem APIs (do inglês - *Application Programming Interface*) e políticas requeridas. A plataforma J2EE apresenta um modelo de aplicação distribuída, componentes reutilizáveis, um modelo de segurança unificado, controle de transações flexível e suporte a Web Services utilizando troca de dados em XML, que baseia-se em padrões e protocolos [28].

2.3.10 ESP32

O ESP32 é uma plataforma composta por uma placa microcontrolada que você pode programar através de uma IDE (do inglês - *Integrated Development Environment*) para processar entradas e saídas entre o dispositivo e os componentes externos conectados a ele. O ESP32 é o que chamamos de plataforma de computação física ou embarcada, ou seja, um sistema que pode interagir com seu ambiente por meio de hardware e software. Uma das placas mais difundidas que utiliza o ESP32 como controlador é a ESP32-DevKitC, produzida pela Espressif (fabricante do ESP32), que possui integração com a Arduino IDE e, conseqüentemente, suporte a muitas bibliotecas compatíveis, contando com a facilidade de programação carregada pela interface do Arduino. A Figura 2.9 apresenta a placa ESP32-DevKitC, com o ESP32 como controlador.

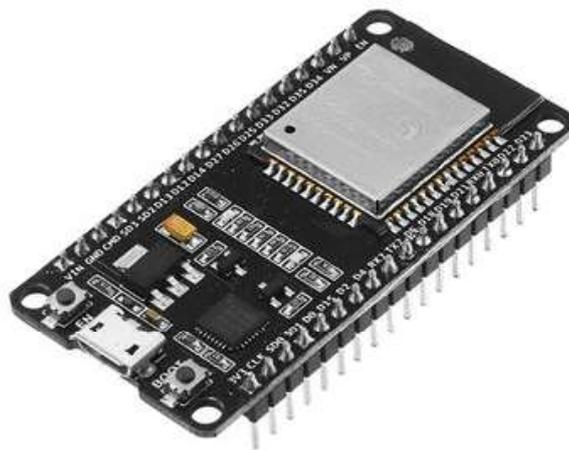


Figura 2.9: Placa ESP32-DevKitC com controlador ESP32.

Fonte: <filipecflop.com/produto/modulo-wifi-esp32-bluetooth/>. Acesso em: 08/11/2018.

O ESP32 tem WiFi integrado e 36 sinais de GPIO. Além disso, vem com Bluetooth 4.2 (BLE), mais sinais de PWM possíveis e mais canais em seu conversor Analógico-Digital. Seu processador é dual-core e vem com cerca de 500 kBytes de memória SRAM, o que permite executar programas mais complexos. Ele possui ainda mais alguns adicionais interessantes como o sensor de temperatura, que permite monitorar a temperatura do microcontrolador, a presença de interface para Sensor de Toque, o que facilita o uso do microcontrolador com teclados ou botões capacitivo, e a presença de um conversor Digital Analógico (DAC).

O ESP32 reúne um apanhado de recursos que tornam seu uso em Internet das Coisas

algo bem interessante, dado que a presença de mais periféricos permite a sua integração com mais dispositivos e componentes diversos. Dentre as interfaces de comunicação, ele possui suporte a SPI, UART e I2C, como também tem suporte a Infravermelho (IR), SDIO (para interface com cartão de memória), CAN, Ethernet, DAC, Sensor de Toque, e I2S, que é uma interface de comunicação útil para comunicar com dispositivos de áudio.

Arduino IDE

Foi utilizado o ambiente de programação denominado Arduino IDE que é baseado na linguagem de programação C. É neste ambiente que se desenvolvem os programas, chamados de *sketches*, os quais irão controlar a placa, os sensores e os atuadores. A Figura 2.10 apresenta a interface da IDE com o *sketch* do programa utilizado no módulo de atuação da janela deste trabalho. Todos os programas utilizados nos ESP32 foram desenvolvidos através da Arduino IDE.



```
JanelaX | Arduino 1.8.1
Arquivo Editar Sketch Ferramentas Ajuda
JanelaX
#include <WiFi.h>

WiFiServer sv(5038); //Cria o objeto servidor na porta 5038
WiFiClient cl; //Cria o objeto cliente.

const int SENSOR_CHUVA = 33; //Define o pino 33 como SENSOR - pino
const int SENSOR_FCA = 13; //Define o pino 13 como SENSOR - pino
const int SENSOR_FCF = 12; //Define o pino 12 como SENSOR - pino

//ENTRADAS MOTOR

int A = 2;
int B = 15;
int C = 4;
int D = 16;

//CARACTERÍSTICAS DA REDE

ESP32 Dev Module, Disabled, Default, QIO, 80MHz, 4MB (32Mb), 921600, None em COM7
```

Figura 2.10: IDE Arduino com exemplo de programação.

2.3.11 Sensor de Luminosidade - LDR

Segundo [29], o LDR (do inglês - *Light Dependent Resistor* - Resistor Dependente de Luz), apresentado na Figura 2.11, é um componente de eletrônico constituído por Sulfeto de Cádmio - CdS, substância semicondutora muito sensível a luz.



Figura 2.11: Sensor LDR.
Fonte: [30]

De acordo com [31], o sensor é constituído de dois terminais e sua resistência varia linearmente com a luz ou radiação incidente com base na equação 2.1.

$$R = CLA \quad (2.1)$$

Na equação 2.1 tem-se que:

- L: Luminosidade em lux;
- C e A: Constantes dependentes do processo de fabricação e do material constituinte.

Sendo assim, a resistência do LDR é inversamente proporcional com a energia da luz ou radiação incidente, logo, se o LDR estiver exposto a ambiente de pouca ou nenhuma luminosidade, maior será sua resistência e menor será sua condutividade, caso ele esteja exposto a um ambiente com elevada luminosidade, maior será sua condutividade e menor sua resistência.

De acordo com [31], o LDR possui como material semiconductor, uma região de valência e uma região de condução. A radiação ou luz que incide sobre o sensor favorece o deslocamento de elétrons da camada de valência para a camada de condução proporcionalmente a sua intensidade, reduzindo a resistência.

O LDR se apresenta como elemento principal do sensor de luminosidade utilizado neste trabalho e será implementado junto ao módulo de sensoriamento desenvolvido.

2.3.12 Módulo Dimmer

O Dimmer é um dispositivo criado com o intuito de proporcionar ao projetista a possibilidade de realizar o controle da quantidade de energia que é enviada para a carga.

Existem dois tipos básicos de módulos dimmers: os que possuem circuito zero crossing e os que não possuem. Este circuito é utilizado para detectar a passagem de tensão por zero e é fundamental para o funcionamento correto do módulo dimmer.

O funcionamento do módulo dimmer pode ser descrito da seguinte forma: inicialmente o sinal da rede passa pelo circuito zero crossing para ser identificado quando a tensão passa por zero. Uma vez identificado este momento e sabendo a frequência da rede, é possível determinar o ângulo de disparo desejado através de um controlador. Sempre que o controlador receber o sinal de passagem por zero, deverá aguardar um tempo determinado (que dependerá do quanto de potência se deseja entregar à carga) e então ativar uma saída que comandará um circuito de potência, responsável por entregar energia à carga. A Figura 2.12 apresenta um diagrama de blocos do circuito do módulo dimmer.

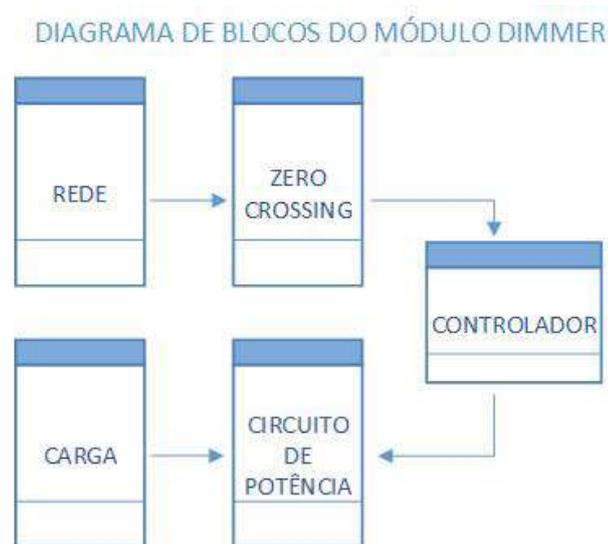


Figura 2.12: Diagrama de blocos do funcionamento do módulo dimmer.

Baseado no descrito acima, se sua rede possui uma frequência de 60Hz, ao ser retificada cada semiciclo terá uma duração de aproximadamente 8300 microssegundos. Desta forma, se deseja-se uma potência de 50% em sua carga, ao ser identificada a passagem por zero, o controlador deve aguardar 4150 microssegundos para ativar a saída. Sendo assim, apenas metade do semiciclo é conduzido para sua carga e, portanto, ela recebe apenas metade da potência.

O módulo dimmer foi utilizado neste trabalho para realizar o controle de intensidade luminosa e faz parte do módulo de iluminação desenvolvido.

2.3.13 Sensor Pluviométrico YL-83

O sensor pluviométrico ou de chuva YL-83, apresentado na Figura 2.13 pode ser usado para monitorar uma variedade de condições climáticas como gotas de chuva ou neve, com

ele você consegue ainda passar a um microcontrolador, como o ESP32, quando começa uma precipitação. A partir dessa informação você pode, por exemplo, fazer um motor funcionar para fechar uma janela, ou acionar um limpador de para-brisa. Quando o clima está seco a saída do sensor fica em estado alto e quando há gotas de chuva a saída fica em estado baixo.



Figura 2.13: Sensor Pluviométrico YL-83.

Fonte: <filipeflop.com/produto/sensor-de-chuva/>. Acesso em: 10/11/2018.

O limite entre tempo seco e chuva pode ser ajustado através do potenciômetro presente no sensor que regulará a saída digital D0. Contudo para ter uma resolução melhor é possível utilizar a saída analógica A0 e conectar a um conversor AD, como a presente no ESP32 por exemplo.

A placa do sensor de chuva é revestida em ambos os lados com um tratamento de níquel contra oxidação, melhorando assim a condutividade, desempenho e duração.

As especificações técnicas do sensor que será utilizado neste projeto são apresentadas na Tabela 2.1.

Especificações Técnicas	
Tensão de operação	3,3 - 5V
Corrente de saída	100mA
Dimensões do sensor	5x4cm
Dimensões da placa	2,1x1,4cm
Comprimento do cabo	20cm
Saída Digital	D0
Saída Analógica	A0
Sensibilidade ajustável via potenciômetro	Sim
Led indicador para tensão	Sim
Led indicador para saída digital	Sim
Comparador	LM393

Tabela 2.1: Especificações Técnicas do Sensor Pluviométrico.

Fonte: <fileplop.com/produto/sensor-de-chuva/>. Acesso em: 10/11/2018.

2.3.14 Motor de Passo

O motor de passo é um tipo de motor elétrico, isto é, ele é uma máquina que transforma energia elétrica em energia mecânica. Os motores de passo são projetados a fim de se ter um maior controle sobre a rotação do motor, uma importante característica desses motores, é sua compatibilidade com sistemas digitais [32].

Os motores de passo são comumente utilizados em sistemas de controle digital, convertendo informações digitais em uma saída mecânica, assim ele recebe comandos na forma de uma sequência de pulsos para girar ou mover um objeto por uma distância precisa, tornando-se uma interface satisfatória para computadores digitais e microcontroladores [33].

As características básicas de um motor de passo consistem em um rotor e um estator. O rotor é a parte interna e móvel do motor, é a parte que gira, já o estator é a parte externa do motor que é estática, é nesta segunda estrutura que são enroladas as bobinas, enrolamentos de fios de cobre, que criam o campo magnético girante. O conjunto de estator e rotor estão separados por um pequeno espaço vazio entre eles, para que seja possível o movimento de rotação[32].

A disposição dos enrolamentos de fios nas bobinas do motor de passo gera um efeito semelhante àquelas encontradas nos ímãs naturais. Este efeito é criado por um dispositivo que utiliza corrente elétrica para gerar um campo magnético, conhecido como eletroímã. Quando o fio é percorrido por uma corrente elétrica, a espira se submete a uma tensão, o que gerará um campo magnético na área circunvizinha a essa espira através da Lei de Biot-Savart [32].

A passagem de corrente elétrica por um condutor produz campos magnéticos nas imediações do condutor e estabelece-se um fluxo magnético no material ferromagnético

envolto pelas espiras do condutor [32].

O motor de passo é utilizado módulo de atuação desenvolvido neste trabalho para fechar e abrir a janela e precisa de um driver para realizar o acionamento de suas bobinas de modo controlado.

Os drivers para motores de passo funcionam basicamente acionando as bobinas deles através de uma sequência lógica definida. Existem três modos básicos de operação do driver:

- *Wave Driver*: Liga uma bobina por vez, com menor consumo de energia, porém, com menor torque.
- *Full Driver*: Liga duas bobinas por vez, com maior consumo de energia e maior torque.
- *Half Driver*: Alterna a ligação de uma e duas bobinas por vez, dobrando a quantidade de passos necessários para o motor girar completamente, porém, é menos veloz.

Para este trabalho foi adotado o modo *Half Driver* uma vez que o motor será utilizado para abrir e fechar uma janela, portanto, é interessante que este movimento ocorra de uma forma mais lenta. A Figura 2.14 ilustra o acionamento das bobinas para este modo de operação.

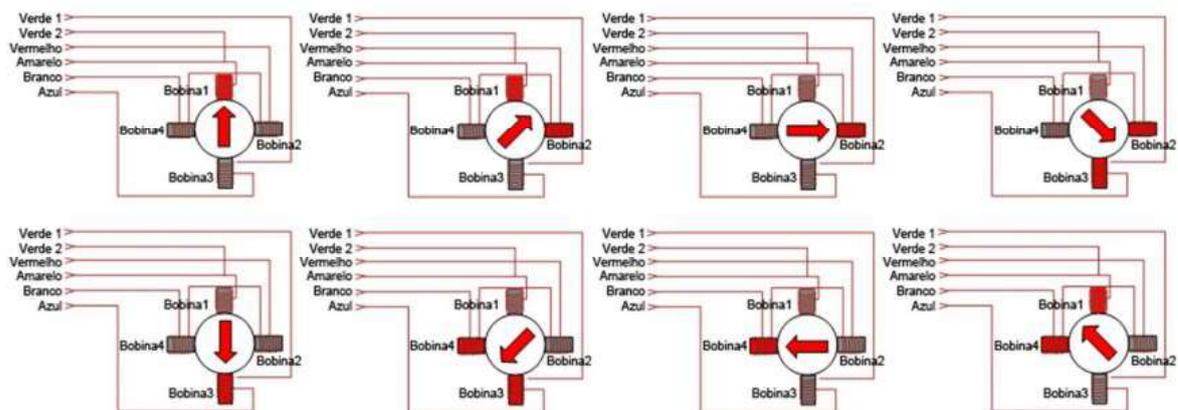


Figura 2.14: Modo de operação *Half Driver* para Motor de Passo.

Fonte: <engineersgarage.com/articles/stepper-motors>. Acesso em: 10/11/2018.

Metodologia

Esta seção apresenta o desenvolvimento do projeto com base nos objetivos propostos. A divisão em tópicos visa manter o nível de organização do trabalho.

A Figura 3.1 apresenta o diagrama geral e fundamental do projeto, o qual abrange toda a implementação do trabalho.

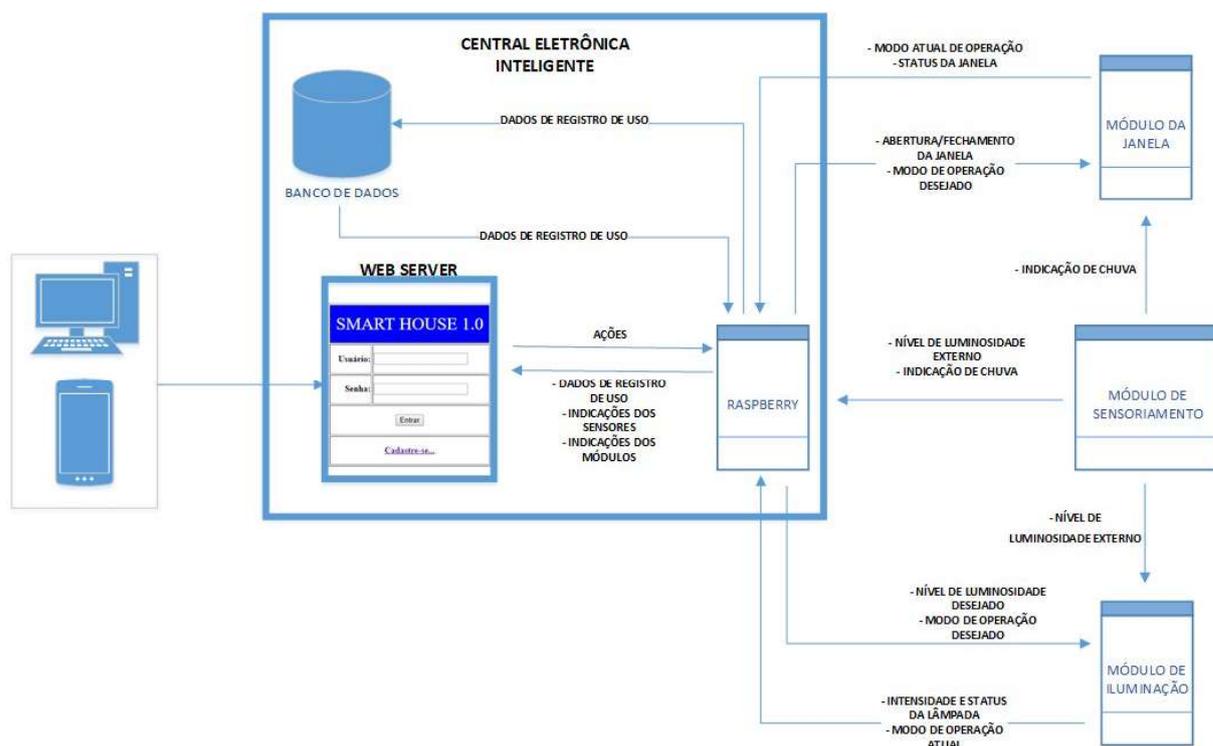


Figura 3.1: Diagrama do projeto.

Com base na Figura 3.1, observa-se que o usuário recebe as informações do sistema através da Aplicação Web, e pode enviar ações para o sistema, através da mesma, por exemplo, abrir uma janela. Ao receber o comando para executar uma ação, a central grava no banco de dados as informações de registro de uso (usuário, ação executada e registro temporal), as quais podem ser acessadas pelo usuário a qualquer momento via

Web App.

Após receber o comando da ação a ser executada na Aplicação Web, a central envia as informações necessárias ao módulo periférico responsável por realizar aquela ação. O módulo de atuação reúne então as informações enviadas pela central e pelo módulo de sensoriamento e então realiza a ação solicitada.

A seguir serão descritas separadamente as etapas de desenvolvimento do projeto.

3.1 Seleção de materiais e métodos

Para o desenvolvimento deste projeto foi necessário realizar uma seleção de materiais, visando utilizar componentes compactos que atendam aos requisitos do projeto.

Controlador - Módulo central

A escolha inicial foi efetuada em relação ao controlador da central, optando entre Arduino Uno e Raspberry Pi 3 Modelo B. Essa escolha levou mais em conta o desempenho e os recursos que cada dispositivo pode oferecer que seu custo no mercado, uma vez que a central precisa ser robusta e capaz de agrupar recursos como a Aplicação Web e o banco de dados. Desta forma, foi selecionado o Raspberry Pi 3 Modelo B.

Controlador - Módulos periféricos

Na etapa seguinte, a escolha foi feita em relação ao controlador dos módulos periféricos. Foi decidida a utilização do ESP32, que pode ser programado através da Arduino IDE. A decisão foi baseada na análise do hardware da placa.

A escolha do controlador dos módulos periféricos foi feita com base nos dados da Tabela 3.1 que apresenta as características do EPS32.

Características do ESP32	
Núcleos do Processador	2
Arquitetura	32 bits
Clock	160 MHz
WiFi	Sim
Bluetooth	Sim
RAM	512KB
Flash	16Mb
GPIO	36
Interfaces	SPI/I2C/UART/I2S/CAN
ADC	18
DAC	2

Tabela 3.1: Características do ESP32. Fonte: [1]

Sendo assim, optou-se pela escolha do ESP32 em virtude das características analisadas. Um destaque dentre as especificações estudadas fica por conta da presença da tecnologia WiFi embutida, uma vez que este item é de suma importância no trabalho.

Módulo dimmer

A seguir foi necessário escolher o dimmer a ser utilizado, tendo como opções a utilização de um módulo dimmer comercial ou a construção do mesmo. A escolha levou em conta principalmente o custo envolvido e optou-se pela construção da placa baseando-se também no fato de que o projeto utilizado para confecção da mesma resultou em sucesso em trabalhos anteriores, como em [8]. O projeto utilizado para a confecção do módulo dimmer será apresentado adiante neste trabalho.

WebServer

Para hospedar a Aplicação Web na central houve a necessidade de escolher um Web-Server. Neste quesito, o Glassfish apareceu como a melhor opção por ser gratuito.

Fonte de alimentação

Cada módulo deste trabalho é alimentado com 5V, portanto são necessárias 4 fontes para os módulos e uma outra para alimentar o motor.

Sendo assim, foi necessária a escolha entre a fonte da Figura 3.2 e uma fonte com conector micro USB (Figura 3.3).



Figura 3.2: Fonte Hi Link 5V.

Fonte: <arduoeletro.com/mini-fonte-5v-hi-link-hlk-pm01-100240vac>. Acesso em: 10/11/2018.



Figura 3.3: Fonte com conector micro USB.

Embora a fonte Hi Link favoreça a compactação do sistema como um produto, seu custo ($R\$25,00$) no momento não é considerado válido para a implementação deste trabalho, uma vez que os autores já possuem o outro modelo supracitado.

Sendo assim, para este trabalho será utilizada a fonte da Figura 3.3, visando reduzir os custos deste trabalho. Entretanto, têm-se ciência da necessidade da utilização da fonte Hi Link para a confecção de um produto.

Janela

Para aplicar o módulo de atuação da janela foi necessário a escolha da janela de correr a ser utilizada, dentre as opções constavam uma janela comercial e uma janela prototipada no SolidWorks e impressa em 3D com ABS (Figuras 3.4 e 3.5).

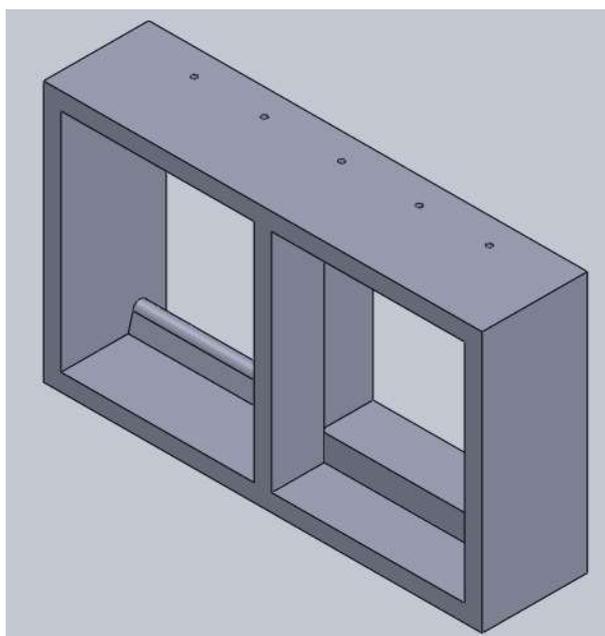


Figura 3.4: Janela prototipada - Vista em perspectiva frontal.

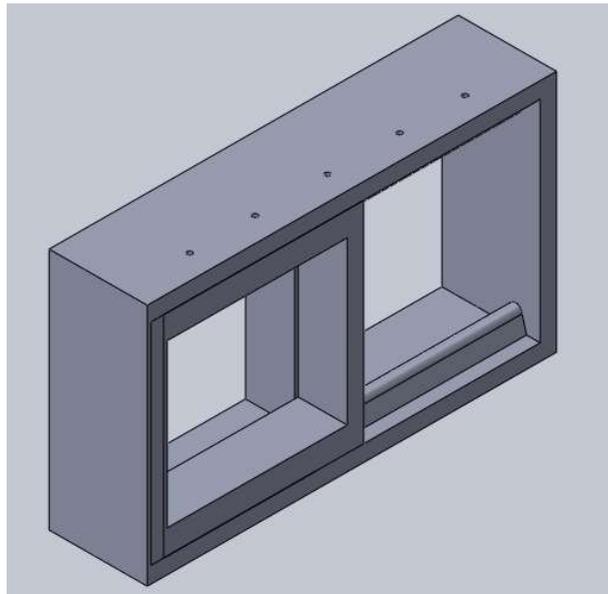


Figura 3.5: Janela prototipada - Vista em perspectiva posterior.

Sendo assim, a escolha deste elemento foi baseada no custo, na questão de deslocamento e na influência deste na escolha do motor. Desta forma, optou-se pelo componente prototipado, uma vez que ele é mais barato, menor e mais leve permitindo a utilização de um motor menos potente, conseqüentemente mais barato.

Motor e driver

Inicialmente, foi desenvolvida uma pesquisa, visando encontrar um motor que fosse capaz de realizar a movimentação da janela de correr selecionada para o trabalho. Além disso, foram considerados também aspectos como alimentação necessária e tamanho do mesmo, principalmente para reduzir ou simplificar circuitos adicionais para atender a alimentação e buscando também produzir um sistema compacto.

Feito isso, foi escolhido o Motor de Passo 28BYJ-48 com controlador (driver ULN2003) que pode ser alimentado com a mesma fonte do controlador e que conseguirá movimentar a janela escolhida. A Tabela 3.2 apresenta as características do motor e do driver.

Especificações do Motor e do Driver	
Tensão	5V
Tipo	Unipolar
Resistência	60 ohms/fase
Torque Máx	2.2 Kgf.cm
Ângulo de passo	5,625 x 1/64
Relação de redução	1/64
Extensão do fio	24 cm
Diâmetro do motor	28 mm
Dimensão do driver (CxLxA)	3,5 x 3,1 x 1,1 cm

Tabela 3.2: Características do Motor de Passo e Driver.

Fonte: <usinainfo.com.br/motores-de-passo/motor-de-passo-28byj-48-com-controlador-driver-uln2003-2535.html>. Acesso em: 10/11/2018.

A Figura 3.2 apresenta o motor e o driver escolhidos para o projeto.

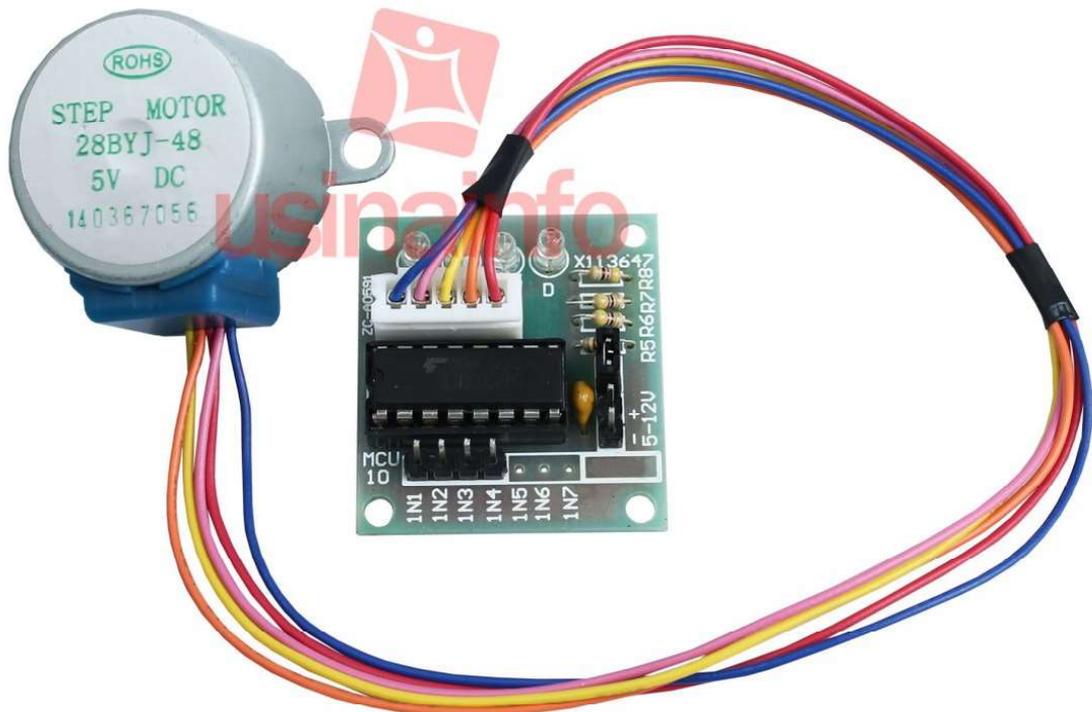


Figura 3.6: Motor de Passo e Driver escolhidos.

Fonte: <usinainfo.com.br/motores-de-passo/motor-de-passo-28byj-48-com-controlador-driver-uln2003-2535.html>. Acesso em: 10/11/2018.

3.2 Central eletrônica

Após a decisão do controlador que atua como central eletrônica foram necessárias algumas configurações para seu funcionamento.

Inicialmente, foi instalado o sistema operacional Raspbian no Raspberry utilizado. A seguir, foi realizada a configuração para estabelecer um IP estático ao dispositivo, com o intuito de favorecer sua comunicação com os módulos periféricos dentro da rede local.

O próximo passo foi a instalação do servidor que hospeda a Aplicação Web, o Glassfish. E por fim, a instalação do software MariaDB para a utilização do banco de dados na central.

Todas os passos acima citados constam detalhadamente no Apêndice I deste trabalho.

3.2.1 Aplicação Web

A Aplicação Web é uma peça fundamental do trabalho, uma vez que ela será a interface entre usuário e sistema. Portanto, buscou-se uma plataforma que fosse simples e objetiva para auxiliar o usuário a comandar sua casa.

Inicialmente foi desenvolvida a página inicial, a qual é responsável por gerenciar o login e realizar o encaminhamento para o cadastro de usuários.

Na etapa seguinte, foi implementada a página para inserir um novo usuário que trabalha com nome, senha e tipo de usuário.

A seguir, foi desenvolvida a página home que realiza a comunicação com o módulo de sensoriamento, recebendo os dados dos sensores por WiFi. Além disso, esta página redireciona para os cômodos da casa e para uma página de controle geral da residência.

Na próxima etapa, foram produzidas as páginas dos cômodos e a página de controle geral que permitem o envio de comandos para o sistema, além de fornecer, também, os dados referentes as leituras dos sensores.

O sistema permite ainda que o usuário gere um arquivo de configuração para seu módulo, para que em casos de necessidade o próprio usuário consiga implementar uma nova configuração.

Toda a aplicação foi desenvolvida utilizando o software NetBeans IDE.

3.2.2 Banco de dados

Para a modelagem do banco, adotou-se o MySQL Workbench como ferramenta de administração de banco de dados, uma vez que ele é amplamente utilizado, compatível com vários servidores e gratuito. A Figura 3.7 apresenta o diagrama do banco de dados desenvolvido.

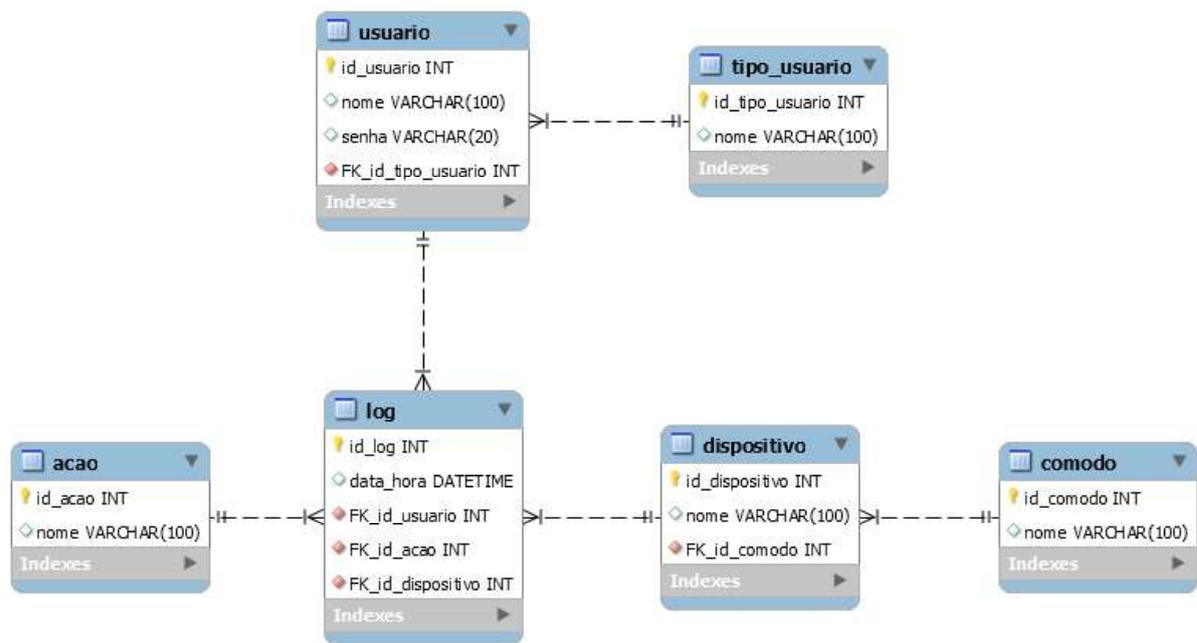


Figura 3.7: Diagrama do Banco de Dados desenvolvido.

O banco de dados do sistema apresenta as funções de backup e restore para situações adversas.

3.3 Módulos periféricos

Para o desenvolvimento dos módulos periféricos, adotou-se dois modelos distintos baseados na finalidade do módulo, portanto, podemos dividi-los em dois grupos: (i) Módulos de sensoriamento e (ii) Módulos atuadores. Ambos possuem o mesmo controlador (ESP32) que realiza a comunicação wireless.

Sendo assim, o sistema apresenta três módulos periféricos, os quais possuem algumas funções comuns no desenvolvimento de seus software, as quais foram projetadas, principalmente, porque os dispositivos permanecem ligados durante todo o tempo de funcionamento do sistema.

A seguir são expostas as funções supracitadas, e logo adiante as particularidades de cada módulo estão apresentadas em forma subseções buscando favorecer a organização do trabalho.

Proteção contra aquecimento

Esta implementação visa evitar que os módulos sejam danificados em caso de um possível aquecimento.

Este código utiliza o sensor de temperatura interno do ESP32 e, com base no dado fornecido pelo mesmo, promove reinicializações sucessivas até que sua temperatura seja

normalizada.

Proteção contra travamentos

Esta funcionalidade foi definida com o intuito de evitar que seja necessário um reset manual do código em caso de travamentos, favorecendo assim a dinâmica de todo o sistema.

Desta forma, por meio da utilização de um Watchdog Timer, caso o sistema fique travado em algum ponto por mais de 6 segundos, ele se reinicia e volta a funcionar normalmente.

Teste de conexão

Esta aplicação tem o intuito de garantir a conexão com a internet durante todo o funcionamento do módulo.

Por funcionarem em grandes intervalos de tempo e de forma contínua, sabe-se que quedas de internet podem acontecer e por vezes até reinicializações dos roteadores. Com esta implementação visou-se conferir se a conexão está estabelecida a todo instante, desta forma, em situações de queda de internet, o próprio dispositivo é capaz de retomar seu funcionamento normal, reestabelecendo a conexão.

Configuração de IP e comunicação

Todos os módulos apresentam em seu códigos uma função para configuração do IP estático, visando favorecer as condições de comunicação, uma vez que os dados enviados e recebidos pelos módulos são direcionados pelo IP e por uma porta. Sendo assim, há duas maneiras possíveis de comunicação: (i) via protocolo TCP/IP e (ii) via protocolo UDP. Desta forma, foram escolhidos os tipos de comunicação dependendo de cada aplicação.

A comunicação entre um módulo e a Aplicação Web hospedada na central é realizada via protocolo TCP. Uma vez que estes dados são os que ficam gravados no log e, além disso, são enviados apenas quando solicitados, é fundamental que eles sempre cheguem ao seu destino e o protocolo TCP é mais eficiente para esta aplicação.

Já a comunicação entre módulos é feita utilizando protocolo UDP. Uma vez que a possível perda de alguns pacotes nesse caso, não impacta a aplicação.

3.3.1 Módulo de sensoramento

A implementação deste módulo foi feita com uso dos seguinte dispositivos:

- ESP32;
- Sensor de luminosidade;

- Sensor de chuva;
- Fonte de alimentação.

Inicialmente, foi realizada a montagem do circuito da Figura 3.8.

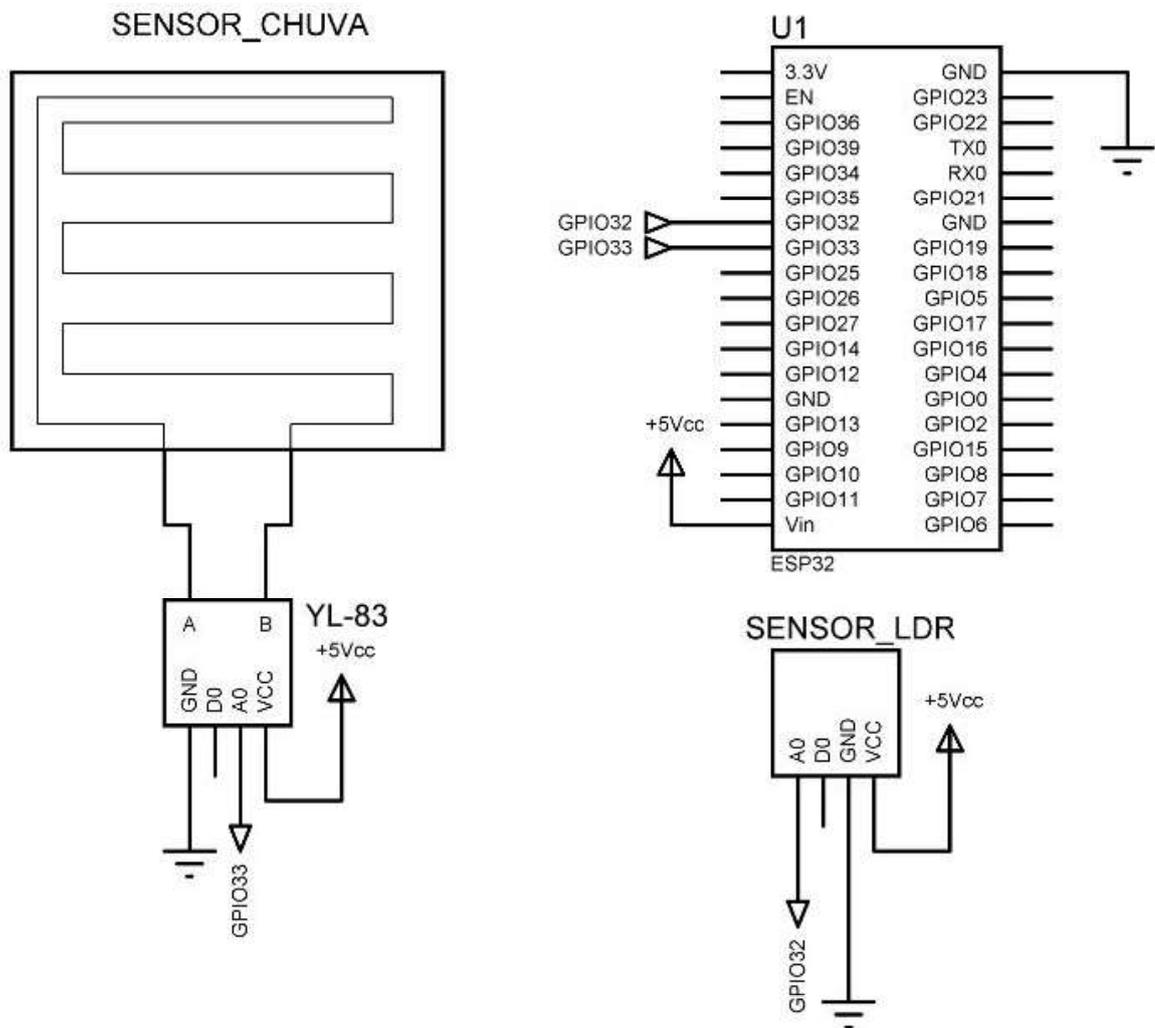


Figura 3.8: Circuito do módulo de sensoriamento.

A seguir, iniciou-se a programação do módulo, a qual consiste em ler os dados dos sensores e fornece-los a Aplicação Web quando solicitado, e aos demais módulos em tempo real.

3.3.2 Módulo de iluminação

Para realizar a implementação deste módulo foram utilizados os seguinte componentes:

- ESP32;
- Módulo dimmer.

O projeto do circuito do módulo pode ser observado na Figura 3.9.

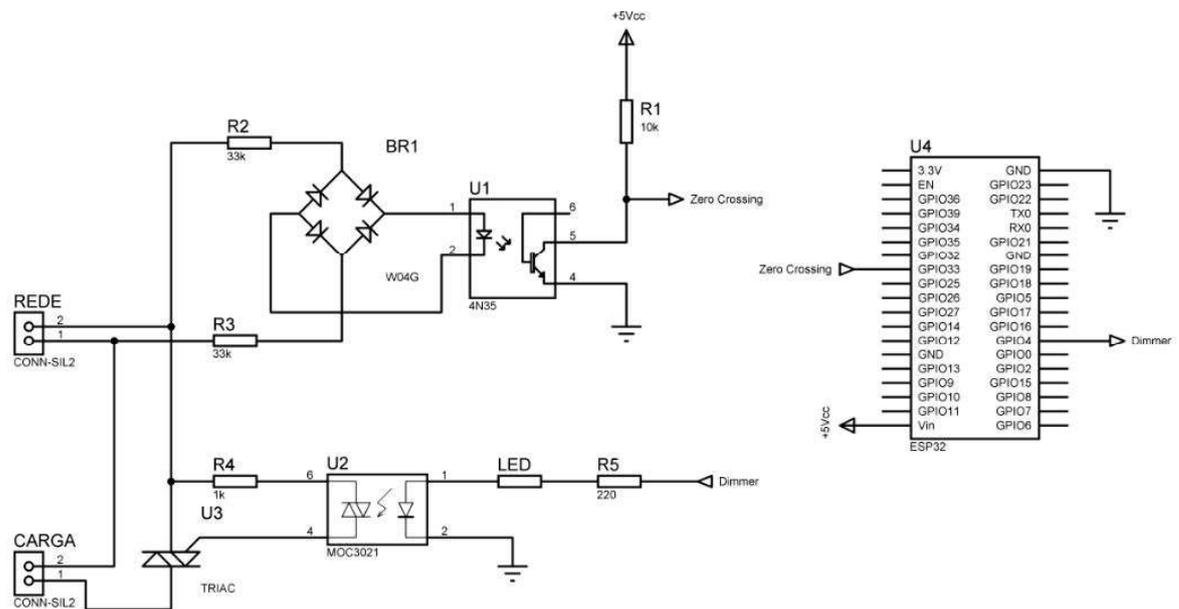


Figura 3.9: Circuito do módulo de iluminação.

Inicialmente foi implementado um código capaz de receber e enviar dados para a Aplicação Web utilizando a rede WiFi. Este programa foi feito visando receber comandos como acender, apagar, aumentar ou diminuir gradativamente o brilho, definir o modo de operação.

Feito isso, seguiu-se para a recepção dos dados provenientes do módulo de sensoria-mento, visando receber o nível de luminosidade externa. Foram desenvolvidas as partes dos códigos para ambos os módulos de modo que o de sensoria-mento enviase o nível de luminosidade para o módulo de atuação a todo tempo sem intermediação da central.

O passo seguinte tratou da implementação do módulo dimmer que foi montado com base no esquema da Figura 3.10.

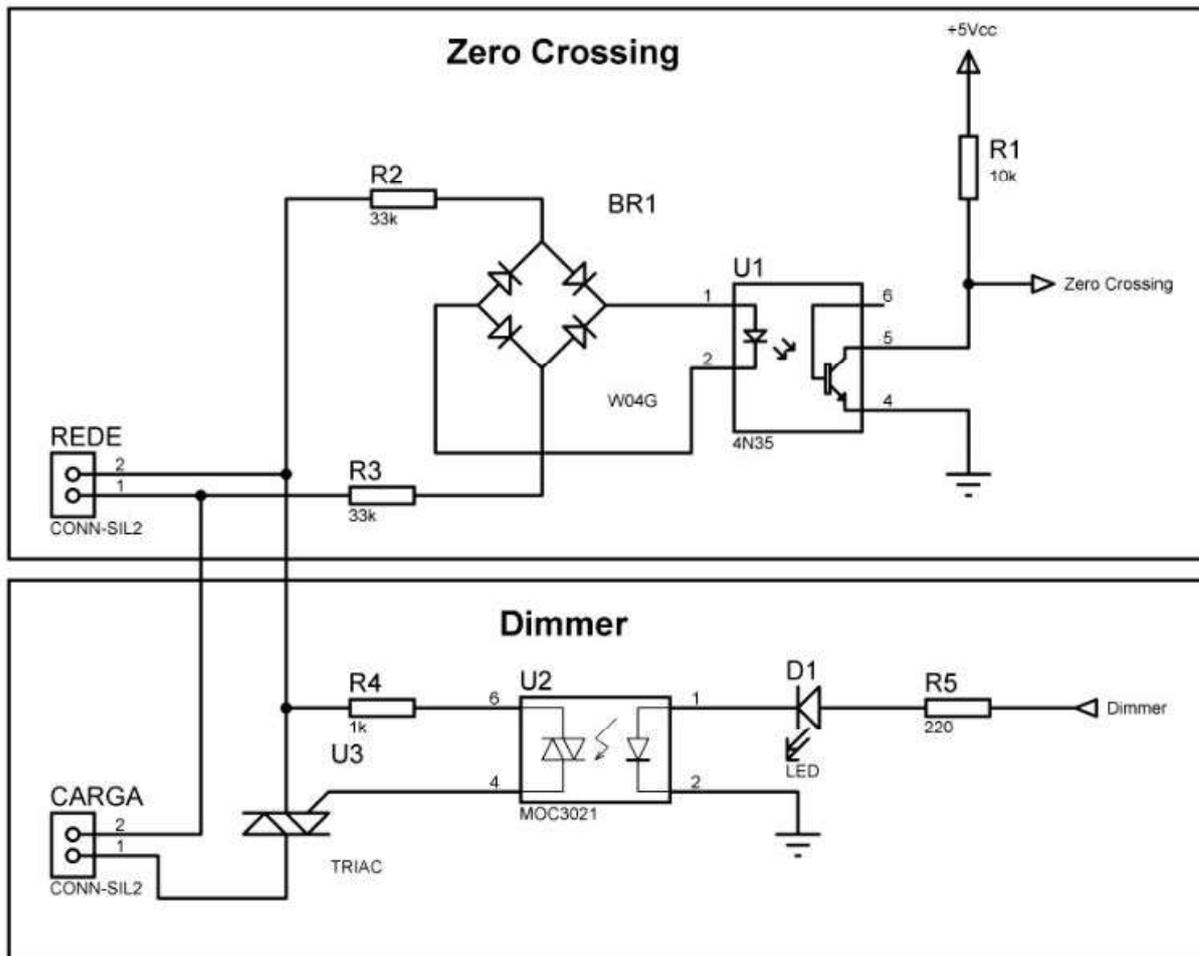


Figura 3.10: Circuito do módulo de dimmer.

O circuito pode ser dividido em duas partes: (i) O circuito zero crossing e (ii) Circuito que leva a potência a lâmpada. O funcionamento do módulo dimmer desenvolvido, pode ser descrito da seguinte forma: inicialmente a rede é retificada e passa a ficar com os dois semiciclos positivos, então com o auxílio de um fotoacoplador é possível identificar quando a tensão passa por zero. Uma vez identificado este momento e sabendo a frequência da rede, é possível determinar o ângulo de disparo desejado através de um controlador. Sempre que o controlador receber o sinal de passagem por zero, deverá aguardar um tempo determinado (que dependerá do quanto de potência se deseja entregar à carga) e então ativar uma saída. A saída do controlador é isolada da parte final do circuito através de outro fotoacoplador pois é necessário entregar potência a sua carga e o controlador utilizados possui limitações de corrente em suas saídas. O fotoacoplador trabalha em conjunto com a saída do controlador na ativação do gatilho de um TRIAC e assim é possível controlar o ângulo de disparo dele e, conseqüentemente, a potência entregue à carga.

Com os dados sendo recebidos de forma satisfatória e o módulo dimmer funcionando,

partiu-se para o tratamento dessas informações, gerando o comando a ser enviado para atuar sobre a lâmpada com base no comando recebido, e definindo os modos de operação.

A seguir, foi implementado um interruptor com retorno por mola que gera uma interrupção no código para que o usuário consiga atuar sobre o sistema sem precisar acessar a Aplicação Web.

Por fim, foram definidos os 3 modos de operação:

- Manual;
- Automático;
- Automático gradual.

O modo manual realiza a atuação apenas mediante a comandos e é definido toda vez que o comando enviado é acender, apagar, aumento ou redução gradativa manual, modo manual, ou quando o interruptor do módulo é acionado.

O modo automático atua no sistema quando o nível de luminosidade é inferior a 50%, promovendo o acionamento total da lâmpada. Ele é acionado ao pressionar o botão Automático da Aplicação Web.

O modo automático gradual age sobre o sistema gradativamente, variando a intensidade da luz conforme a variação externa medida. Este modo é acionado através do botão Automático gradual da Aplicação Web.

3.3.3 Módulo de atuação da janela

Este módulo é composto pelos componentes abaixo:

- ESP32;
- Driver do motor de passo;
- Motor de passo;
- Sensor de fim de curso A;
- Sensor de fim de curso B.

O circuito do módulo pode ser observado na Figura 3.11.

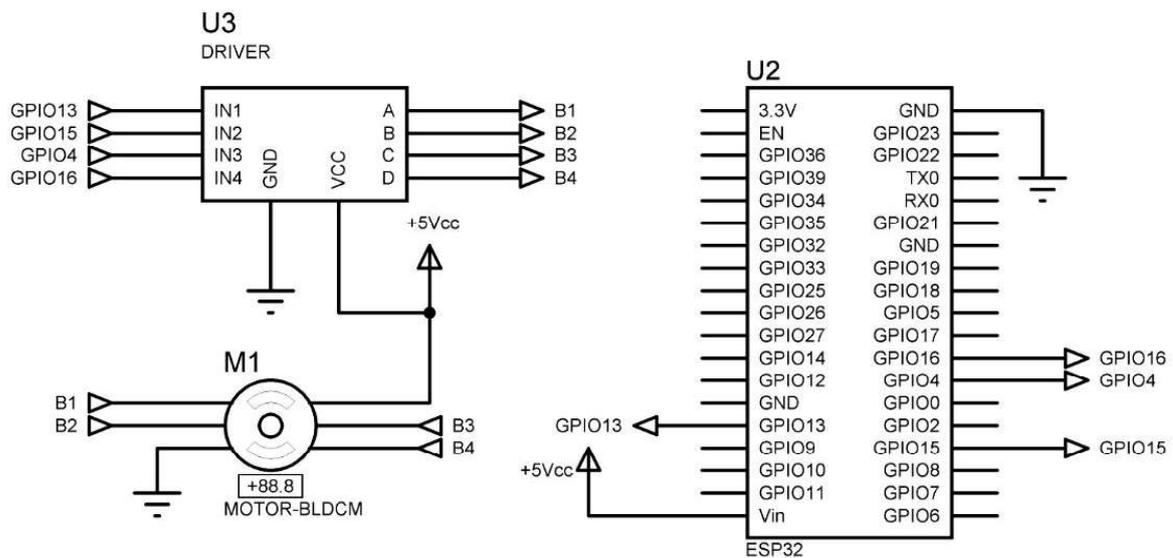


Figura 3.11: Circuito do módulo de atuação de janela.

O primeiro passo no desenvolvimento deste módulo foi a implementação do código para comunicação entre este módulo e a Aplicação Web, permitindo que ele consiga enviar e receber dados. Dentre os dados recebidos pelo sensor estão os comandos abrir, fechar e parar, e o modo de operação. Já entre os dados enviados constam o estado atual da janela (aberta, fechada ou parcialmente aberta) e o modo de operação ativado.

A seguir, foi feita a programação referente a comunicação com o módulo de sensoria-mento, por onde o de atuação da janela recebe a informação referente a chuva ou tempo seco.

O passo seguinte consistiu na lógica para a indicação dos sensores de fim de curso para indicar a abertura ou fechamento total da janela.

Por fim, foi implementado o tratamento dos dados recebidos, gerando, desta forma, o comando a ser enviado para atuação sobre a janela.

Os modos de operação deste módulo consistem em:

- Manual;
- Automático.

O modo manual atua somente mediante ao recebimento de um comando da Aplicação Web e é definido ao pressionar o botão Manual. Já o modo automático, quando ativado, inicia o fechamento da janela caso identifique que há precipitação. Ele é iniciado pelo botão Automático.

3.4 Estruturas para comportar os módulos

Visando aumentar a durabilidade dos módulos e a melhoria da estética do sistema, foram desenvolvidas 3 cases no software *SolidWorks* para comportar os 2 módulos periféricos (sensoriamento e iluminação) e a central do sistema.

O módulo de atuação da janela será acoplado abaixo da mesma, que foi apresentada nas Figuras 3.4 e 3.5.

3.4.1 Projetos

Para cada módulo foi projetada uma estrutura para comportar seus componentes. Estas estruturas foram idealizadas para serem impressas utilizando a tecnologia 3D. Porém, devido a questões de custo, optou-se por partir para outra solução, apresentada no próximo tópico. A Figura 3.12 apresenta um dos modelos projetados que seria utilizado no módulo de sensoriamento.

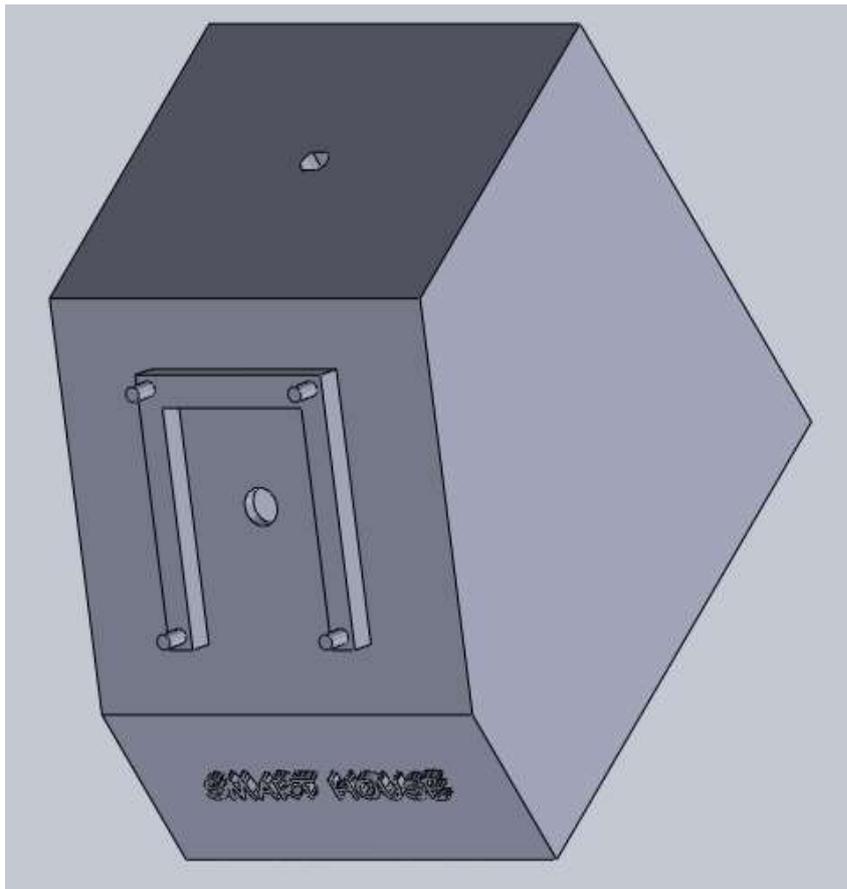


Figura 3.12: Vista em perspectiva com foco na parte superior da peça.

3.4.2 Construção

Para a construção dos módulos, foi inicialmente realizado o orçamento para efetivação da impressão 3D, porém o mesmo apresentou-se como empecilho ao desenvolvimento uma vez que os preços foram considerados exorbitantes para as peças, como pode ser visto na Tabela 3.3.

Peça	Preço
Case do módulo de sensoriamento	R\$ 80,00
Case do módulo de iluminação	R\$ 70,00
Case do módulo central	R\$ 40,00
Total	R\$ 190,00

Tabela 3.3: Orçamento da construção dos módulos projetados.

Com base no orçamento apresentado, buscou-se uma alternativa mais viável para a aplicação proposta. Dentre as opções encontradas, optou-se pela utilização de uma caixa de embutir com tampa, utilizada em instalações residenciais. A solução encontrada é apresentada na Figura 3.13.



Figura 3.13: Vista em perspectiva da solução utilizada para encapsular os módulos.

Sendo assim, foram realizadas adaptações na estrutura da Figura 3.13 para o encapsulamento dos dois módulos e da central.

Resultados

4.1 Central Eletrônica

A Central Eletrônica desenvolvida consiste em um Raspberry Pi 3 Modelo B e abrange três componentes do sistema, sendo eles: (i) Aplicação Web; (ii) Web Server e (iii) Banco de Dados. Todos estes fazem parte da central e estão alocados dentro dela e para isso, foi desenvolvida uma case capaz de comportar todos os elementos como mostra a Figura 4.1.



Figura 4.1: Case da central do sistema.

4.1.1 Aplicação Web

A Aplicação Web desenvolvida para realizar a interação entre o usuário e o sistema conta com as seguintes telas:

- Login (Figuras 4.2, ?? e ??);
- Cadastro (Figuras 4.3 a ??);
- Página inicial da casa (Figura 4.4);
- Página do comodo (Figura 4.5);
- Página de controle geral (Figura 4.6).

Para o correto funcionamento desta aplicação foram definidos dois tipos de usuário: (i) master e (ii) comum. A diferença entre os dois tipos está presente no cadastro de um novo usuário, visto que apenas um usuário master pode cadastrar novos utilizadores. Sendo assim, o primeiro cadastro do sistema, automaticamente define o usuário como master.

Ao acessar a Aplicação o usuário é direcionado a página de Login que apresenta os campos usuário e senha, bem como os botões entrar e cadastrar como apresentado na Figura 4.2.



SMART HOUSE 1.0	
Usuário:	<input type="text" value="Rafael"/>
Senha:	<input type="password" value="....."/>
<input type="button" value="Entrar"/>	
<input type="button" value="Cadastrar"/>	

Figura 4.2: Página de login.

No primeiro acesso, o usuário precisa realizar seu cadastro, portanto ao clicar em cadastrar, ele será redirecionado para a página de Cadastro de Usuários vista na Figura 4.3. Feito o primeiro cadastro, esta tela só poderá ser acessada novamente através do usuário e a senha de alguém já cadastrado como usuário master.

SMART HOUSE 1.0	
CADASTRO DE USUÁRIOS	
Usuário:	<input type="text"/>
OBS: Insira um usuário sem caracteres especiais com até 10 dígitos.	
Senha:	<input type="text"/>
OBS: Insira uma senha sem caracteres especiais com até 8 dígitos.	
Confirmar senha:	<input type="text"/>
Tipo de usuário:	<input type="radio"/> Master <input type="radio"/> Comum
<input type="button" value="Cadastrar"/>	
<input type="button" value="Voltar"/>	

Figura 4.3: Página de cadastro.

Na página de cadastro estão contidos quatro campos: (i) usuário; (ii) senha; (iii) confirmar senha e (iv) tipo de usuário. Caso ocorra algum problema com os dados são exibidas as possíveis mensagens:

- Cadastro não efetuado... Por Favor, insira um nome de usuário e uma senha...
- Cadastro não efetuado... Por Favor, insira um nome de usuário...
- Cadastro não efetuado... Por Favor, insira uma senha...
- Cadastro não efetuado... As senhas informadas são diferentes...

Se não houver erros, a página efetuará o cadastro do novo usuário no banco de dados do sistema. Após realizar o cadastro, o usuário é novamente direcionado a tela de login e ao efetuá-lo, a página confere os dados no banco e libera ou não o acesso. Em caso negativo, podem ser exibidas as seguintes mensagens:

- Usuário não cadastrado...
- Senha incorreta...

Efetuada o login com sucesso, o usuário tem a sua disposição a tela Home, a qual exibe o nome e o tipo do usuário atual, as funções que dão acesso aos cômodos e ao controle

geral da casa, bem como o logoff. Esta tela, assim como todas as outras que vem a seguir, exibe os dados da leitura dos sensores, permitindo ainda que eles possam ser atualizados a qualquer momento. Estes dados são recebidos via WiFi e transcritos nos campos como pode ser visto na Figura 4.4.

SMART HOUSE 1.0		
Usuário master: Rafael	Sair	
CÔMODOS DA CASA		
Sala	Quarto	Cozinha
Varanda	Garagem	Banheiro
Controle geral		
Leitura dos sensores externos:		
Sensor de luminosidade:	95 %	
Sensor de chuva:	Sem chuva	
Atualizar		

Figura 4.4: Página Home.

A seleção de páginas de cômodos no projeto atual é indiferente, uma vez que como há apenas um módulo de cada, todos os comandos, independente do comodo em que é enviado, seguem para o mesmo módulo, seja ele de iluminação, sensoriamento ou atuação da janela. Entretanto, existem páginas com o cabeçalho diferente visando favorecer o layout da Aplicação Web. Um aspecto interessante é que desta forma, os demais cômodos já se apresentam pré-configurados, necessitando apenas da inserção dos IPs e portas pelos quais se comunicam com os módulos.

A Figura 4.5 apresenta a página padrão dos cômodos do sistema, neste caso, representando a sala.

SMART HOUSE 1.0			
Usuário master: Rafael			Sair
SALA			
ILUMINAÇÃO			
Status:	Luz apagada	Intensidade:	0%
Acender	Apagar	+	-
Manual	Auto	Auto gradual	
Modo de operação:		Manual	
JANELA			
Status:	Janela parcialmente aberta		
Abrir		Fechar	
Parar			
Manual		Auto	
Modo de operação:		Manual	
Leitura dos sensores externos:			
Sensor de luminosidade:		94 %	
Sensor de chuva:		Sem chuva	
Atualizar			
Voltar			

Figura 4.5: Página de um cômodo.

Nesta página o usuário consegue executar comandos referentes a iluminação e a janela. Ao selecionar algum comando a Aplicação direciona o sinal para o módulo responsável via WiFi e recebe do mesmo uma resposta quanto a atuação, indicando que a lâmpada foi acesa ou não, bem como sua intensidade, ou que a janela está aberta, parcialmente aberta ou fechada. Além disso, a página recebe o modo de operação atual de cada módulo.

Há ainda, a tela para o controle geral da residência (Figura 4.6) que permite que o usuário dispare comandos a todos os módulos de uma só vez.

SMART HOUSE 1.0			
Usuário master: Rafael			Sair
CONTROLE GERAL			
Manual		Auto	
ILUMINAÇÃO			
Acender	Apagar	+	-
Manual	Auto	Auto gradual	
JANELAS			
Abrir		Fechar	
Parar			
Manual		Auto	
Leitura dos sensores externos:			
Sensor de luminosidade:		94 %	
Sensor de chuva:		Sem chuva	
Atualizar			
Voltar			

Figura 4.6: Página de controle geral da residência.

A página em questão apresenta a funcionalidade manual e automático geral, que defini estes modos para ambos os módulos de atuação ao mesmo tempo. Abaixo na tela, constam as funcionalidades gerais separadas por atuação. Sendo assim, é possível enviar comandos a todos os módulos de iluminação da casa instantaneamente, bem como aos de atuação da janela.

4.1.2 Banco de dados

O banco de dados projetado armazena as características do usuário como login, senha e tipo de usuário, porém armazena também diversas outras informações baseadas nos acontecimentos do sistema e na residência do usuário. Tudo isso, visa possibilitar ao utilizador do sistema os dados do registro de uso.

A Figura 4.7 apresenta um exemplo de alguns dados de registro de uso cadastrados no banco de dados do sistema.

Usuário	Ação	Dispositivo	Comodo	Data e hora
Rafael	Acendeu	Lâmpada	Sala	2018-11-11 17:54:39
Elias	Acendeu	Lâmpada	Quarto	2018-11-11 17:54:39
Elias	Redução gradual	Lâmpada	Quarto	2018-11-11 17:54:39
Rafael	Apagou	Lâmpada	Sala	2018-11-11 17:54:39
Rafael	Modo automático	Janela	Sala	2018-11-11 17:54:39
Elias	Modo automático gradual	Lâmpada	Quarto	2018-11-11 17:54:39
Rafael	Abrir	Janela	Sala	2018-11-11 17:54:39
Elias	Parar	Janela	Sala	2018-11-11 17:54:39
Rafael	Fechar	Janela	Sala	2018-11-11 17:54:39
Rafael	Modo automático gradual	Lâmpada	Sala	2018-11-11 17:54:39

Figura 4.7: Dados de registros de uso retornados do banco de dados usando join.

Como observado na Figura 4.7, a implementação deste recurso é de suma importância para o sistema, uma vez que permite identificar onde uma falha ocorreu, bem como que iniciou essa ação. Além disso, esta funcionalidade resguarda os desenvolvedores do mau uso do sistema, uma vez que através dela é possível indicar quem disparou cada ação.

4.2 Módulos periféricos

O modelo final generalizado dos módulos periféricos é composto por dois elementos fundamentais, sendo eles: (i) o ESP32 e (ii) módulos sensores ou atuadores, dependendo da função do módulo periférico em questão.

Basicamente, os módulos de sensoriamento são capazes de receber o sinal de um sensor e transmiti-lo através da conexão WiFi para a central. Já os módulos de atuação são capazes de receber instruções da central e mediante a elas, executar os devidos comandos dos atuadores. Eventualmente - como no caso do módulo de atuação da janela - os módulos atuadores poderão contar com um ou mais sensores que auxiliem o seu controlador na tomada de decisões.

Foram construídos três módulos periféricos nesta pesquisa e o resultado de cada um será apresentado nas próximas seções. Ressalta-se que mais módulos podem ser desenvolvidos futuramente e adicionados ao sistema criado.

4.2.1 Módulo de sensoriamento

O modelo final do módulo de sensoriamento desenvolvido pode ser dividido em três partes: (i) controlador EPS32; (ii) sensor de luminosidade e (iii) sensor de chuva.

O sensor de luminosidade entrega ao seu controlador o nível de luminosidade do ambiente, seu controlador envia este sinal para o módulo de iluminação e para a central que, por sua vez, envia o valor para Aplicação Web.

O sensor de chuva faz um processo semelhante ao de luminosidade, porém nele não se mede a intensidade de chuva, ele informa apenas se está chovendo ou não. Esse dado é enviado também para o módulo de atuação da janela e para a central que exibe o dado na Aplicação Web.

Visando encapsular o módulo de sensoriamento, desenvolveu-se uma estrutura com as dimensões comuns dos sensores de presença utilizados em aplicações residenciais, buscando uma forma de facilitar a instalação deste componente pelo usuário. A Figura 4.8 apresenta o visual final do módulo de sensoriamento.



Figura 4.8: Case do módulo de sensoriamento do sistema.

4.2.2 Módulo de iluminação

O modelo final do módulo de iluminação pode ser dividido em três partes: (i) ESP32; (ii) módulo dimmer e (iii) botão de acionamento.

O ESP32 é responsável por receber os comandos advindos da central e interpretá-los para então comandar o atuador e realizar a ação devida. Quando o usuário solicita que a

lâmpada acenda ou apague, através da Aplicação Web, o ESP32 recebe esta informação e comanda o circuito do dimmer para acender ou apagar a lâmpada em sua totalidade. Quando o usuário solicita a operação automática, o ESP32 se baseia nos dados advindos do módulo de sensoriamento e se estiver recebendo um valor de 50% de luminosidade ou inferior, ele comanda o dimmer para ativar a lâmpada, caso esteja acima de 50% a lâmpada deve ser apagada. Por fim, se o usuário solicita o modo automático gradual, o ESP32 também se baseia no nível de luminosidade fornecido pelo módulo de sensoriamento e aumenta a intensidade da lâmpada a medida em que o nível de luminosidade medido diminui. Esta função só é possível devido ao circuito dimmer, que é capaz de controlar o ângulo de disparo de um TRIAC através do comando do controlador e, conseqüentemente, controlar a potência na lâmpada.

O módulo conta ainda com um botão, caso o usuário resolva fazer o acionamento manual. Ao pressionar o botão a lâmpada entra automaticamente em modo manual e troca o seu estado atual. Portanto, se está ligada ela desliga e se está desligada ela acende.

Visando encapsular o módulo de sensoriamento, produziu-se uma estrutura com as dimensões comuns dos interruptores domésticos, buscando uma forma de facilitar a instalação deste componente pelo usuário. A Figura 4.9 apresenta o visual final do módulo de iluminação.



Figura 4.9: Case do módulo de iluminação do sistema.

4.2.3 Módulo de atuação da janela

O modelo final do módulo de atuação da janela desenvolvido pode ser dividido em duas partes: (i) o controlador ESP32; (ii) o motor de passo e seu driver e (iii) os sensores de fim de curso.

O ESP32 é responsável por receber os comandos advindos da central e interpretá-los para então comandar o atuador e realizar a ação devida. Quando o usuário solicita que a janela abra ou feche, através da Aplicação Web, o ESP32 recebe esta informação e comanda o driver para abrir ou fechar a janela em sua totalidade. Quando o usuário solicita a operação automática, o ESP32 se baseia nos dados advindos do módulo de sensoramento e se estiver recebendo sinal de chuva, ele comanda o driver para abrir a janela, caso esteja sem chuva a janela deve ser aberta.

O módulo conta ainda dois fim de curso, uma para indicar que a janela está totalmente aberta e outro para indicar que ela está totalmente fechada. Caso nenhum dos dois esteja ativo, é sinal que a janela está parcialmente aberta. O sinal dos sensores é sempre levado em conta pelo controlador, assim se a janela está abrindo e o fim de curso é ativado, o controlador desliga o motor, o mesmo ocorrer para o fechamento.

Para este módulo, foi desenvolvida não só a estrutura do módulo em si, mas também a janela na qual será aplicada a automação. Foi projetada uma janela com a movimentação baseada no modelo de pinhão e cremalheira. A janela já possui em seu design um compartimento para comportar o motor e o circuito eletrônico do módulo na sua parte inferior. Além disso, ela conta com acomodações para o encaixe dos sensores de fim de curso.

4.3 Comunicações

O sistema desenvolvido conta com uma ampla rede de comunicação, na qual praticamente todos os elementos se comunicam de forma direta ou indireta. A Aplicação Web se comunica com a central, com o banco de dados e com os módulos periféricos; os módulos se comunicam entre si e com a central e o sistema ainda se comunica com a rede externa, além da rede local.

4.3.1 Comunicação interna da central

A Central possui três componentes que precisam se comunicar e isto ocorre de maneira interna, uma vez que todos eles estão alocados dentro dela. A central em si utiliza de sua comunicação WiFi para buscar dados dos módulos periféricos. Estes dados são requeridos através da Aplicação Web via protocolo TCI/IP. Nas páginas da web, foram desenvolvidos códigos que possuem os IP's e portas de comunicação de cada módulo periférico. Ao navegar por elas, os códigos solicitam ou enviam os dados para os módulos periféricos de acordo com cada ação.

Além disso, a Aplicação Web se comunica também com o banco de dados e pode gravar, editar, apagar ou buscar dados dele. A Aplicação se comunica tanto com o servidor do banco quanto com os módulos, através de códigos desenvolvidos em javascript.

4.3.2 Comunicação entre central e módulos periféricos

Como descrito no tópico anterior, a Aplicação Web solicita ou envia dados para os módulos periféricos através do IP de cada um e de uma porta escolhida para a comunicação. Porém, para que isso funcione é necessário que o código presente no ESP32 de cada módulo esteja preparado para realizar esta comunicação também.

O processo no ESP32 para o protocolo TCP/IP ocorre um pouco diferente se comparado com a Aplicação Web, já que no controlador não são passados IP e porta como parâmetros. Portanto, após o ESP32 ser conectado à rede, ele fica aguardando a solicitação da central para só então enviar ou receber dados.

4.3.3 Comunicação entre módulos periféricos

Diferentemente da comunicação com a Aplicação Web, os módulos comunicam entre si via protocolo UDP. Isso ocorre devido ao fato de que a comunicação entre eles é praticamente constante e, portanto, não há uma preocupação caso haja perda de algum pacote.

4.3.4 Comunicação do sistema com a rede externa

A central é dotada de um servidor web, o qual por padrão já permite acesso a qualquer dispositivo dotado de navegador e a redes TCP/IP. Assim sendo, cabe ao proprietário da central garantir que ela possua acesso à internet por um IP válido, o qual pode ser fixo, dinâmico e se desejado resolvido por um servidor de nomes DNS ou DDNS.

Desta forma as configurações de redirecionamento de portas, firewalls ou afins dependerão do usuário dado que cada qual possuirá uma configuração diferente.

Neste trabalho, para hospedar a página e fornecer acesso pela rede externa será utilizado um serviço DDNS gratuito fornecido por Winco DDNS. Sendo assim, foi implementado um domínio gratuito para a apresentação e testes.

Considerações Finais

Analisando as premissa do projeto e realizando uma comparação entre a metodologia atual e a fundamentação teórica, é possível apresentar as seguintes constatações.

5.1 Conclusão

Este trabalho apresentou o desenvolvimento de um sistema inteligente para automação residencial. O sistema apresentado faz referência aos módulos de domótica existentes em residências, mas foi desenvolvido artesanalmente e para duas aplicações apenas, embora seja possível acrescentar novos módulos ao sistema.

Nessa pesquisa foi fundamentado, projetado e construído o sistema. Foram realizadas ainda a especificação de componentes, construção dos módulos e simulações.

O intuito principal do trabalho foi implementar a automação em ambientes residenciais com arquitetura integrada, tornando possível controlar os elementos presente nas residencias. O sistema desenvolvido foi aplicado para dois casos distintos, sendo eles o controle de luminosidade e o controle da abertura e fechamento de uma janela. Embora estes dois módulos tenham sido desenvolvidos e utilizados para aplicação do sistema, os mesmo não se limita a eles, deixando espaço para o desenvolvimento de novos módulos.

A implementação do projeto se deu de maneira satisfatória. A escolha do modo de comunicação (wireless) contribuiu bastante para a miniaturização dos módulos e viabilidade de instalação. A arquitetura de integração escolhida, permitiu que fossem alcançados bons resultados no que tange à quantidade de informações disponíveis ao usuário do sistema, como a indicação dos status atuais de cada componente automatizado, modo de operação vigente e os dados fornecidos pelos sensores.

Embora o grau de eficiência do sistema tenha sido satisfatório, alguns problemas foram identificados ao longo de seu desenvolvimento. Um dos mais marcantes foi a ausência de um programa java para ser executado o tempo todo dentro da central, o que tornou o sistema um pouco dependente da Aplicação Web. Outro problema foi a escolha do

circuito de controle da luminosidade, uma vez que é necessário avaliar com minúcia o circuito baseado na lampada escolhida e de um modo geral estes dados são difíceis de serem encontrados.

5.2 Propostas de Continuidade

Ao fim deste trabalho foi possível a observação de algumas lacunas, as quais podem ser preenchidas por trabalhos futuros visando a melhoria do sistema desenvolvido.

Dentre as novidades que podem ser implementadas no sistema, destacam-se:

- A implementação de novos módulos, visando aumentar ainda mais o conforto do usuário;
- O desenvolvimento de um aplicativo para smartphones, tornando o sistema mais dinâmico;
- A melhoria na interface gráfica da Aplicação Web, buscando a evolução estética do projeto;
- A implementação de uma aplicação java na central para eliminar a comunicação entre os módulos periféricos;
- A utilização de fontes internas aos módulos, visando favorecer a compactação;
- A implementação de coolers dentro dos módulos desenvolvidos para atuar quando o sensor de temperatura interno registrar valores acima dos normais.

Referências

- [1] Espressif. Datasheet esp32 - espressif, 2018. Disponível em https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf Acessado em: 20 Nov 2018.
- [2] M. V. Souza. Domótica de baixo custo usando princípios de iot. Master's thesis, Universidade Federal do Rio Grande do Norte. Faculdade de Engenharia de Software, Natal, RN, 2016.
- [3] Crestron. A experiência crestron, 2017. Disponível em http://bit.ly/a_experiencia_crestron Acessado em: 13 Mar 2018.
- [4] L. Franscisco and K. Trevisani. Hms: Uma arquitetura para automação residencial aberta independente de tecnologia de rede. *Colloquium Exactarum*, 5(1):43–56, Jan-Jun 2013.
- [5] AURESIDE. Automação residencial: demanda na construção civil, 2017. Disponível em www.aureside.org.br/noticias/automacao-residencial--demanda-na-construcao-civil Acessado em: 13 Mar 2018.
- [6] IHS Markit. Expectativa para o mercado mundial para equipamentos de casa inteligentes, 2018. Disponível em ihsmarkit.com/index.html Acessado em: 07 Jun 2018.
- [7] R. Bunemer. Domótica assistiva utilizando sistemas integrados de supervisão e controle. Master's thesis, Universidade Estadual de Campinas. Faculdade de Engenharia Mecânica, Campinas, SP, 2014.
- [8] Rayner Ribeiro de Moraes. Desenvolvimento de sistemas de domótica controlados por dispositivos móveis. Master's thesis, Centro Federal de Educação Tecnológica de Minas Gerais - CEFET-MG, Divinópolis, MG - Brasil, 2015.

-
- [9] F. G. Demétrio, F. H. C. B. Furlani, M. M. Maria, and P. M. dos Santos. Análise e implantação da domótica em edifícios residenciais de alto padrão. Master's thesis, Universidade do Vale do Paraíba, São José dos Campos, SP, 2016.
- [10] F. F. B. Gomes. Remote home: Uma plataforma para interações via dispositivos móveis em ambientes residenciais. Master's thesis, Universidade Salvador. Faculdade de Sistemas e Computação, Salvador, BA, 2016.
- [11] José Roberto Muratori and Paulo Henrique Dal Bó. *Automação Residencial - Conceitos e Aplicações*. Editora Educere, 2th edition, 2014.
- [12] CEDOM. Asociación española de domótica e inmótica - qué es domótica, 2011. "Disponível em <http://www.cedom.es/sobre-domotica/que-es-domotica#gkPageTop> Acessado em: 09 Maio 2018."
- [13] Caio Augustus Morais Bolzani. Desmistificando a domótica. *Sinergia*, 8(1):17–20, January 2007.
- [14] Andrew S. Tanenbaum. *Computer Networks*. Prentice Hall, 4th edition, 2002. Vrije Universiteit, Amsterdam, Holanda.
- [15] César Augusto Hass Loureiro, Marcelo Augusto Rauh Schmitt, André Peres, and Alex Martins de Oliveira. *Redes de computadores III - Níveis de enlace e físico*. Bookman Editora, 1th edition, 2014. Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul.
- [16] James F. Kurose and Keith W. Ross. *Redes de computadores e a internet*. Pearson, 5th edition, 2010.
- [17] David Plà Santamaría. *Localizacion de informacion especifica en la web*. Editorial de la Universidad Politécnica de Valencia, 2005.
- [18] Juvenal dos Santos Barreto. Um modelo de migração de ambiente ipv4 para ipv6 em uma rede acadêmica heterogênea. Master's thesis, Universidade de Brasília. Instituto de Ciências Exatas - Departamento de Ciência da Computação., Brasília, DF - Brasil, 2015.
- [19] Edson Kazuo Hirose. Análise da qualidade de transmissão de vídeo stream sobre udp. Master's thesis, Universidade Federal de Pernambuco - UFPE, Recife, PE, 2012.
- [20] Dustin R. Callaway. *Inside Servlets: Server-side Programming for the Java Platform*. Addison-Wesley, 2th edition, 2001.

-
- [21] Edwin Luis Choquehuanca Mamani. Um sistema servidor web distribuído com provisão de qos absoluta e relativa. Master's thesis, Instituto de Ciências Matemáticas e de Computação - ICMC, USP, São Carlos, SP - Brasil, 2010.
- [22] H. F. Korth and A. Silberschatz. *Sistema de Bancos de Dados*. Makron Books, 2th edition, 1994.
- [23] Osvaldo Kotaro Takai, Isabel Cristina Italiano, and João Eduardo Ferreira. *Introdução a Banco de Dados*. DCC-IME-USP, 1th edition, 2005.
- [24] C. J. Date. *Introdução a Sistemas de Bancos de Dados*. Editora Campus, 4th edition, 1991.
- [25] Renato Ribeiro Haber. Uma ferramenta de importação de documentos html para um ambiente de ensino. Master's thesis, Instituto de Ciências Matemáticas e de Computação - USP, São Carlos, SP, 1999.
- [26] Fernando Augusto Ferreira Lordão. Reconhecimento de formas usando modelos de compressão de dados e espaço de escalas de curvatura. Master's thesis, Universidade Federal da Paraíba - UFPB, João Pessoa, PB, 2009.
- [27] Diogo Francisco Borba Rodrigues. Manejo de irrigação via acesso remoto. Master's thesis, Universidade Federal Rural de Pernambuco, Recife, PE - Brasil, 2013.
- [28] Nívia Menezes de Oliveira. Web services e j2ee: Uma solução para integração em ambientes heterogêneos. Master's thesis, Universidade Federal de Pernambuco - UFPE, Recife, PE, 2003.
- [29] Edgard Rogério de Siqueira Vasconcelos. Detector para raios x usando sensor ldr para mensurar a energia dos fótons emitidos e testar a eficiência de aparelhos hospitalares. Master's thesis, Universidade de Brasília - Faculdade de Gama, Brasília - DF, February 2014.
- [30] Hamamatsu Photonics K.K., Hamamatsu City, Japão. *CdS photoconductive cell*, October 2001.
- [31] D. Thomazini and P. U. B. Albuquerque. *Sensores Industriais - Fundamentos e Aplicações*. Érica, São Paulo, 5th edition, 2012.
- [32] Vincent Del Toro. *Fundamentos de Máquinas Elétricas*. LTC, 1th edition, 1994.
- [33] Bruno Moreira Matos. Janela residencial automatizada. Master's thesis, Centro Universitário de Brasília - UNICEUB, Brasília, DF - Brasil, 2009.

Complementos

Neste apêndice serão apresentados procedimentos de configuração e códigos utilizados no trabalho.

A.1 Configuração do Raspberry

Nesta seção, serão apresentados todos os passos necessários para a utilização do Raspberry no projeto.

A versão do dispositivo utilizado foi o Raspberry Pi 3 Modelo B, apresentado na Figura A.1 e possui as seguintes características:

- CPU Quad Core 1.2GHz Broadcom BCM2837 64bit;
- 1 GB de RAM;
- Rede sem fio BCM43438 e Bluetooth Low Energy (BLE) a bordo;
- GPIO estendido de 40 pinos;
- Saída de vídeo HDMI;
- Saída estéreo de 4 polos e porta de vídeo composto;
- 4 portas USB 2;
- Entrada Micro USB para energização;
- Slot para cartão de memória.



Figura A.1: Raspberry Pi 3 Modelo B.

Além do Raspberry, foram necessários alguns periféricos para realizar a configuração dele, são eles:

- Cartão SD de 8GB (ou superior);
- Monitor com entrada HDMI;
- Teclado e mouse com entrada USB;
- Carregador Micro USB de celular com saída de 5V;

A.1.1 Instalação do sistema operacional

O Raspberry precisa de um sistema operacional(SO) para seu funcionamento e devido a sua arquitetura, existem alguns SO's específicos baseados em linux que podem ser utilizados de forma gratuita. Estes sistemas operacionais podem ser encontrados na aba downloads do site oficial do Raspberry <www.raspberrypi.org>.

Para este projeto foi escolhido o Raspbian como SO devido a simplicidade e facilidade de obtê-lo e instalá-lo. O site do Raspberry traz ainda uma ferramenta para auxiliar na instalação do SO chamada de *NOOBS*(ja inclui o Raspbian) como mostra a Figura A.2. Esta ferramenta ajuda no gerenciamento do download e instalação do sistema operacional e neste trabalho ela foi utilizada para a instalação do Raspbian.



Figura A.2: *NOOBS* disponível para download.

Uma vez baixado o arquivo do *NOOBS*, é preciso extrair o conteúdo para um diretório conhecido e após extraí-lo é necessário copiá-lo para um cartão de memória que precisa ser formatado antes deste processo. Para realizar a formatação recomenda-se o programa *Formatter 4* que pode ser encontrado no site <www.sdcard.org/downloads>. Após instalado, basta executar o programa, selecionar a unidade onde está o cartão de memória e clicar em "Formatar". Feito isso o cartão de memória está pronto para receber os arquivos, portanto, basta copiá-los para dentro do cartão de memória.

Após ter os arquivos no cartão de memória, faz-se necessário conectá-lo ao Raspberry. Além disso, deve-se conectar os demais periféricos ao Raspberry e ligá-lo. Ao iniciar o Raspberry aparecem três opções de instalação, deve ser selecionada a opção recomendada de instalação do Raspbian e após isso clicar em "Instalar". Aparecerá uma janela na qual deve-se clicar em "Yes" e então basta aguardar a instalação ser finalizada. Por último clica-se em "OK" para o sistema ser reiniciado e em "Finish" assim que ele religar.

A.1.2 Configuração do IP estático

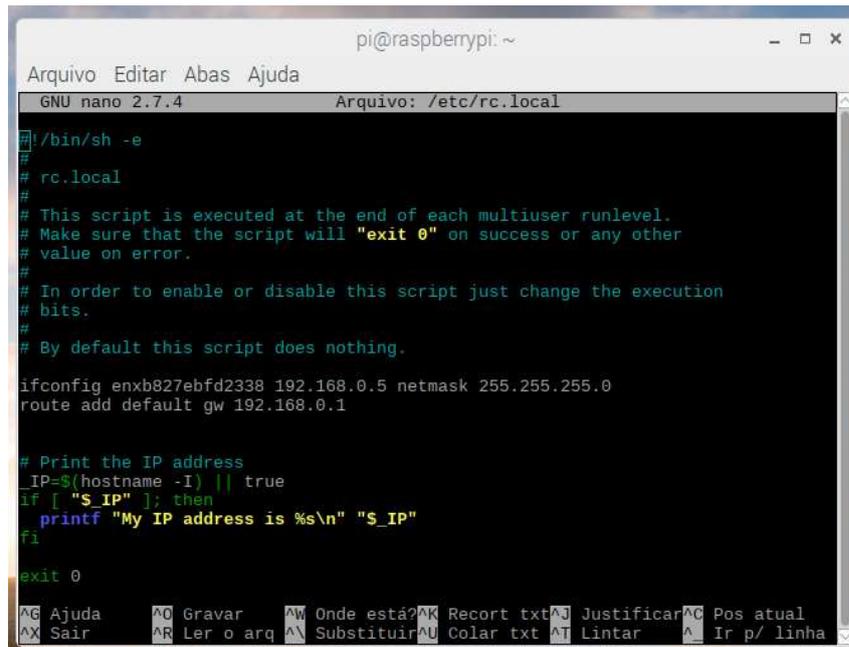
Os procedimentos a seguir devem ser realizados via LXTerminal do Raspberry.

1. Abrir arquivo "rc.local" de execução automática na inicialização.

- `sudo nano /etc/rc.local`

2. Incluir o código abaixo no arquivo para configurar o IP para 192.168.0.5, a máscara de rede para 255.255.255.0 e o gateway para 192.168.0.1. Por fim, efetue o salvamento do arquivo. A Figura A.3 apresenta o arquivo editado. Neste caso, o raspberry nomeou a placa de rede como 'enxb827ebfd2338'.

- `ifconfig enxb827ebfd2338 192.168.0.5 netmask 255.255.255.0`
- `route add default gw 192.168.0.1`



```
pi@raspberrypi: ~
Arquivo Editar Abas Ajuda
GNU nano 2.7.4 Arquivo: /etc/rc.local
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

ifconfig enx827ebfd2338 192.168.0.5 netmask 255.255.255.0
route add default gw 192.168.0.1

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
  printf "My IP address is %s\n" "$_IP"
fi
exit 0
```

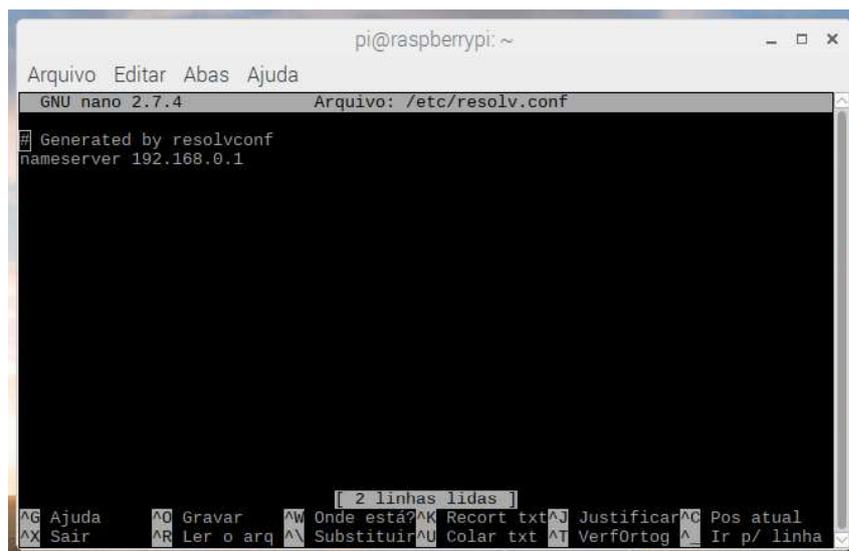
Figura A.3: Arquivo "rc.local" editado.

3. Abrir arquivo para indicar o servidor DNS.

- `sudo nano /etc/resolv.conf`

4. Inserir o código abaixo no arquivo e salvar. A figura A.4 apresenta o arquivo editado.

- `nameserver 192.168.0.1`



```
pi@raspberrypi: ~
Arquivo Editar Abas Ajuda
GNU nano 2.7.4 Arquivo: /etc/resolv.conf
# Generated by resolvconf
nameserver 192.168.0.1
```

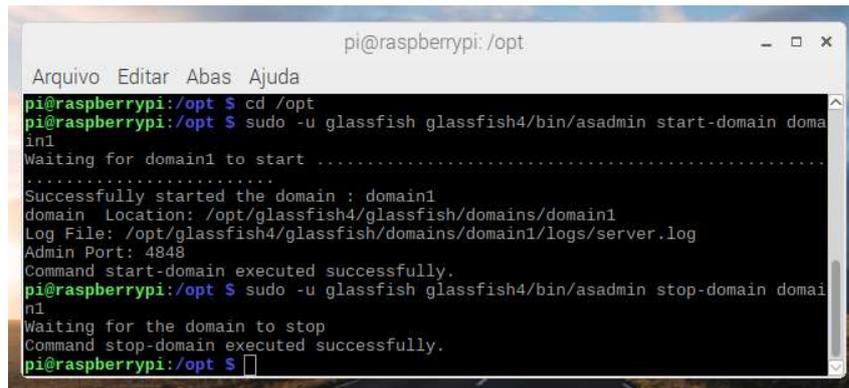
Figura A.4: Arquivo "resolv.conf" editado.

A.1.3 Instalação do Glassfish

Os procedimentos a seguir devem ser realizados via LXTerminal do Raspberry.

1. Instalar o Unzip para descompactar os arquivos de instalação.
 - `sudo apt-get install unzip`
2. Instalar o JDK.
 - `sudo apt-get install openjdk-7-jdk`
3. Baixar o Glassfish.
 - `sudo wget http://download.java.net/glassfish/4.1.1/release/glassfish-4.1.1-web.zip`
4. Alterar o diretório para "opt" que é um diretório opcional de programas.
 - `cd /opt`
5. Extrair o arquivo de instalação do Glassfish.
 - `sudo unzip /home/pi/glassfish-4.1.1-web.zip`
6. Criar um usuário para o Glassfish por conta de permissões.
 - `sudo useradd -system glassfish -d /opt/glassfish4`
7. Alterar o usuário para a pasta glassfish4.
 - `sudo chown -R glassfish glassfish4`
 - `sudo chgrp -R sudo glassfish4`
8. Conceder permissões de execução.
 - `sudo chmod -R +x glassfish4/bin`
 - `sudo chmod -R +x glassfish4/glassfish/bin`
9. Testar o servidor (iniciar).
 - `sudo -u glassfish glassfish4/bin/asadmin start-domain domain1`
10. Testar o servidor (parar).
 - `sudo -u glassfish glassfish4/bin/asadmin stop-domain domain1`

A figura A.5 mostra o resultado do teste efetuado pelos autores.



```
pi@raspberrypi: /opt
Arquivo Editar Abas Ajuda
pi@raspberrypi:/opt $ cd /opt
pi@raspberrypi:/opt $ sudo -u glassfish glassfish4/bin/asadmin start-domain domain1
Waiting for domain1 to start .....
Successfully started the domain : domain1
domain Location: /opt/glassfish4/glassfish/domains/domain1
Log File: /opt/glassfish4/glassfish/domains/domain1/logs/server.log
Admin Port: 4848
Command start-domain executed successfully.
pi@raspberrypi:/opt $ sudo -u glassfish glassfish4/bin/asadmin stop-domain domain1
Waiting for the domain to stop
Command stop-domain executed successfully.
pi@raspberrypi:/opt $
```

Figura A.5: Teste do Glassfish.

A.1.4 Instalação do MariaDB

Os procedimentos a seguir devem ser realizados LXTerminal do Raspberry.

1. Atualizar o repositório.
 - `sudo apt-get update`
2. Instalar o MariaDB.
 - `sudo apt-get install mysql-server`