

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
CAMPUS JOÃO PESSOA
COORDENAÇÃO DO CURSO TÉCNICO EM ELETRÔNICA

RAFAEL SOBRAL DE MORAIS
JOSÉ ROBERTO MESQUITA CORDEIRO DA CRUZ

AUTOMAÇÃO RESIDENCIAL COM COMANDOS DE VOZ E TALKBACK

João Pessoa-PB

2018

RAFAEL SOBRAL DE MORAIS

JOSÉ ROBERTO MESQUITA CORDEIRO DA CRUZ

AUTOMAÇÃO RESIDENCIAL COM COMANDOS DE VOZ E TALKBACK

Trabalho de Conclusão de Curso apresentado a Coordenação do Curso Técnico em Eletrônica do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba como pré-requisito para obtenção do diploma de Técnico em Eletrônica.

Orientador: Prof.Dr. Adaildo G D'Assunção Jr.

João Pessoa-PB

2018

FICHA CATALOGRÁFICA

RAFAEL SOBRAL DE MORAIS
JOSÉ ROBERTO MESQUITA CORDEIRO DA CRUZ

AUTOMAÇÃO RESIDENCIAL COM COMANDOS DE VOZ E TALKBACK

Trabalho de conclusão de curso Técnico Integrado ao Médio apresentado ao Instituto Federal da Paraíba requisito parcial para a obtenção do título de Técnico em Eletrônica.

Aprovado em: ____ de ____ de ____.

BANCA EXAMINADORA

Prof. Dr. Adaildo Gomes D'Assunção Jr. _____

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba

Prof. Dr. Emmanuel Benoit J.B. Dupouy _____

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba

Prof. Msc. Marcos Moura Bandeira _____

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba

RESUMO

Este trabalho apresenta uma solução de automação residencial simples, projetada utilizando microcontroladores e incorporando recursos de comando de voz e *talkback*, disponíveis em uma interface desenvolvida para Android, visando trazer acessibilidade ao sistema. Para isso é realizado um estudo sobre os elementos envolvidos no projeto, além da domótica e suas aplicações. Apesar da possibilidade de facilitar inúmeras tarefas, os sistemas automatizados têm baixa adesão no ambiente residencial, devido ao elevado custo que acaba por restringir o acesso. Há uma necessidade de pesquisar formas alternativas de automação para residências, e este projeto busca explorar as possibilidades utilizando microcontroladores para implementação de um sistema de automação de baixo custo.

Palavras-chave: Automação Residencial. Domótica. Microcontroladores.

ABSTRACT

This work presents a simple home automation solution, designed using microcontrollers and incorporating voice command and talkback capabilities, available in an interface developed for Android, aiming to bring accessibility to the system. For this is a study on the elements involved in the project, as well as home automation and its applications. Despite the possibility of facilitating numerous tasks, automated systems have low adherence in the residential environment, due to the high cost that ends up restricting access. There is a need to research alternative forms of home automation, and this project seeks to explore possibilities using microcontrollers to implement a low cost automation system.

Keywords: Home Automation. Domotics. Microcontrollers.

LISTA DE FIGURAS

Figura 1 - Arduino Uno R3	15
Figura 2 - WeMos D1 R2	16
Figura 3 - Plataforma AppInventor2	19
Figura 4 - Página de blocos do AppInventor	20
Figura 5 - Exemplo de Combinação de Blocos	21
Figura 6 - Tela 1 do Aplicativo	22
Figura 7 - Tela 2 do Aplicativo	23
Figura 8 - Tela de ajuda	24
Figura 9 - IDE Arduino	25
Figura 10 - Configuração da Conexão	26
Figura 11 - Página HTML	27
Figura 12 - Leitura e execução de comandos	28
Figura 13 - Circuito Utilizado para Testes	29
Figura 14 - Blocos de reconhecimento de voz	30
Figura 15 - Blocos de funcionalidade do Botão	30
Figura 16 - Blocos de Resposta ao Comando	31

LISTA DE SIGLAS

AR	Automação Residencial
CI	Circuito Integrado
HTML	HyperText Markup Language
IDE	Integrated Development Environment
IoT	Internet of Things
LDR	Light Dependent Resistor
LED	Light Emitting Diode
MCU	Microcontroller Unit
MIT	Massachusetts Institute of Technology
RAM	Random Access Memory
ROM	Read-Only Memory

SUMÁRIO

1 INTRODUÇÃO	9
1.1 - APRESENTAÇÃO	9
1.2 – OBJETIVOS	9
1.3 – METODOLOGIA.....	10
2. SISTEMAS DOMÓTICOS	11
2.1 – APLICAÇÕES.....	12
2.1.1 – <i>Controle de Luminosidade</i>	12
2.1.2 - <i>Controle de Temperatura</i>	12
2.1.3 - <i>Sistema de Segurança</i>	13
2.2 – ESTRUTURA DO SISTEMA DE AUTOMAÇÃO.....	13
3 – SISTEMA DE CONTROLE	14
3.1 – MICROCONTROLADOR	14
3.1.1 – <i>Plataforma de desenvolvimento Arduino</i>	15
3.1.2 – <i>Plataforma WeMos e ESP8266</i>	15
3.2 – DISPOSITIVOS MÓVEIS	16
3.2.1 – <i>Sistema Operacional Android</i>	17
3.3 – INTERFACE VISUAL E COMANDO DE VOZ.....	17
4 - COMUNICAÇÃO.....	18
5 – DESENVOLVIMENTO	19
5.1 – APlicativo ANDROID	19
5.1.1 - <i>Funcionalidades do Aplicativo</i>	21
5.2 – PROGRAMAÇÃO DO CONTROLADOR.....	24
5.2.1 – <i>Estabelecimento da Comunicação</i>	25
5.2.2 - <i>Software de Controle</i>	27
5.3 – TESTES DE FUNCIONAMENTO	28
6 – CONCLUSÃO	32
6.1 – PERPESCTIVAS.....	32
REFERÊNCIAS	33
APÊNDICE A – CÓDIGO DO MICROCONTROLADOR	34
APÊNDICE B – CÓDIGO DO APlicativo	36

1 INTRODUÇÃO

1.1 - APRESENTAÇÃO

A automação residencial representa o emprego de tecnologias ao ambiente doméstico (incluindo residências, condomínios, hotéis) com o objetivo de propiciar conforto, praticidade, produtividade, economia, eficiência e rentabilidade, com valorização da imagem do empreendimento de seus usuários (WORTMEYER, 2005)

Apesar das possibilidades apresentadas, no Brasil, a automação residencial não encontra grande mercado consumidor devido ao custo elevado para implementação, e a carência por empresas capacitadas para o fornecimento de soluções nessa área.

Contudo é preciso demonstrar que o avanço tecnológico e a criação de novas tecnologias possibilitam a flexibilização dos projetos de automação, que podem ser implementados de maneira simples e com baixo orçamento, tornando-se acessíveis a um público maior.

1.2 – OBJETIVOS

Este trabalho propõe como principal objetivo o desenvolvimento de um sistema de automação residencial simples e de baixo custo utilizando-se de sistemas microcontrolados e de protocolos de conexão de rede. Além disso o projeto procura incluir recursos de acessibilidade ao sistema, como a possibilidade de comandos utilizando a voz, e um recurso de *talkback*. Serão abordadas as aplicações da domótica e será realizado um estudo sobre as tecnologias utilizadas na confecção destes sistemas. Envolvendo assim, vasta gama de conhecimentos adquiridos durante o curso.

1.3 – METODOLOGIA

Este projeto visa a aplicação prática e experimental do objeto de pesquisa, utilizando conhecimento disponíveis em livros e sites como referências.

A primeira etapa do projeto é a pesquisa bibliográfica que possibilitará o desenvolvimento prático do projeto, para isso, é realizada a consulta em sites da internet e livros que detalhem o processo de implementação do sistema de automação e que forneçam informações úteis sobre cada um dos componentes utilizados.

Finalizada a pesquisa, se inicia o desenvolvimento prático do experimento, com a criação da aplicação de controle e da plataforma de comunicação com o controlador, procurando otimizar o sistema para maior interação com o usuário.

Após a montagem de todo os equipamentos e do sistema de controle, são realizados testes para verificar o funcionamento do sistema.

2. SISTEMAS DOMÓTICOS

Domótica é um novo domínio de aplicação tecnológica, tendo como objetivo básico melhorar a qualidade de vida, reduzindo o trabalho doméstico, aumentando o bem-estar e a segurança de seus habitantes e visando também uma utilização racional e planejada de diversos recursos (ANGEL, 1993). Por meio da domótica é possível ter um ambiente doméstico automatizado e integrado, programado para as necessidades do residente e podendo realizar tarefas como fechamento de portas e janelas, controle de temperatura e luminosidade, ativação de alarmes, dentre outros.

O sistema de domótica geralmente possui um controle central, concentrado em um dispositivo com grande capacidade de processamento de dados, como *smartphones* e *tablets*, que podem estar conectados a uma rede interna ou externa. O controle de um sistema residencial deve possuir uma interface intuitiva para uma melhor interação do usuário, evitando complicações no uso do sistema.

A evolução das tecnologias em geral, especialmente no segmento de IoT, possibilitou que diversos equipamentos presentes nas casas pudessem ser conectados à internet. Atualmente encontram-se disponíveis no mercado desde lâmpadas até geladeiras conectadas a rede. Esta vasta gama de dispositivos adaptados vem facilitando o desenvolvimento de ambientes automatizados, uma vez que o sistema de conexão já está presente nos equipamentos, basta conectá-los a um controle central.

Sendo assim, existe a expectativa de que os sistemas domóticos se tornem cada vez mais presentes, com suas facilidades expandidas para um número maior de pessoas, e tornando a casa futurista, apresentada em várias obras de ficção, uma realidade não muito distante.

2.1 – APLICAÇÕES

O sistema de domótica deve ser adaptado para cada situação específica, satisfazendo as necessidades de cada um dos habitantes, que podem usufruir de diferentes recursos presentes no projeto de automação. Assim a quantidade e a funcionalidade destes recursos variam de acordo com sistema instalado, visto que a rotina em uma casa onde vive apenas uma pessoa geralmente difere bastante de onde vivem seis, e diferentes pessoas tem diferentes exigências, podendo assim simplificar ou complexar o sistema.

A seguir são apresentados alguns exemplos de aplicações dos sistemas domóticos:

2.1.1 – Controle de Luminosidade

Esse sistema de controle é um dos mais impactantes do projeto de automação, pois além de contribuir para o conforto dos residentes, ele também é importante para a eficiência energética, gerando menor consumo e consequentemente reduzindo o gasto na fatura de energia elétrica.

O sistema de controle de Iluminação pode ser desenvolvido envolvendo sensores e atuadores (lâmpadas), nesse caso o indivíduo pode controlar a luminosidade de determinado ambiente interagindo diretamente com a interface de controle, usando seus recursos para ligar ou desligar lâmpadas e abrir ou fechar cortinas. Há também a possibilidade de automatizar totalmente o controle, com o usuário programando os horários em que os atuadores são acionados, ou deixando que o acionamento seja feito de acordo com os dados coletados pelos sensores.

2.1.2 - Controle de Temperatura

O usuário pode modificar a temperatura de um ambiente por meio do controle integrado de aparelhos de ar-condicionado, aquecedores ou ventiladores, que podem também serem regulados por sensores de temperatura presentes. Pela interface deve ser possível controlar o acionamento dos aparelhos e a intensidade de funcionamento. Além disso o sistema pode conter um sistema de aquecimento do piso e controle da qualidade do ar, regulando a umidade e quantidade de CO₂.

2.1.3 - Sistema de Segurança

Outra possibilidade da automação residencial é o controle centralizado dos sistemas de segurança do ambiente, por meio da interação entre alarmes, sensores de presença e câmeras, que enviam seus dados para a interface de controle do sistema de domótica, para que os moradores monitorem as câmeras e os sensores. O sistema pode ser programado para comunicar uma central de segurança no caso de invasão, e também avisar os proprietários da residência caso estes não se encontrem presentes.

2.2 – ESTRUTURA DO SISTEMA DE AUTOMAÇÃO

Ao desenvolver um sistema de automação residencial, a interação entre os equipamentos é estabelecida devido a coordenação entre setores.

O setor de controle é responsável pelo gerenciamento dos atuadores, dispositivos de controle e sensores. Muitas dessas tecnologias adotaram nós de controle com inteligência e memória embutidas, o que garante uma abordagem mais descentralizada do que a utilizada em ambientes prediais, provendo uma maior autonomia a cada um destes pontos. Cálculos e tomadas de decisões são realizados diretamente por microprocessadores instalados próximos aos sensores e atuadores, desafogando o gerenciador principal. Isso traz vários benefícios ao sistema, reduzindo tráfego na rede e evitando uma total paralização do conjunto em caso de pane. O setor de dados apresenta as redes Ethernet (e derivadas) como padrão de fato. Recentemente, as redes sem fio (Wi-Fi) têm estado em grande evidência devido à facilidade de instalação e ao custo final.

(BOLZANI, 2007, p.19).

Por fim, os sensores e atuadores são os responsáveis pela real interação com o ambiente. Os sensores têm como função captar os dados que serão enviados ao setor de controle, onde estes dados serão processados e uma decisão será tomada, resultando no envio de sinais para os atuadores instalados nos dispositivos da casa.

3 – SISTEMA DE CONTROLE

No desenvolvimento de um sistema de automação simples, de baixo custo, deve-se considerar que o sistema de controle será utilizado pelo usuário sem o conhecimento prático ou teórico da tecnologia empregada, então, uma interface de controle intuitiva, de fácil entendimento é desenvolvida para diminuir os requisitos de conhecimento para o usuário. Pode-se utilizar uma interface já difundida como os smartphones e computadores pessoais que irão se comunicar com o microcontrolador que efetuará a operação dos atuadores.

3.1 – MICROCONTROLADOR

Os microcontroladores são basicamente computadores no encapsulamento de um circuito integrado (PENIDO e TRINDADE, 2013, p.15), este componente está presente na grande maioria dos aparelhos eletrônicos disponíveis atualmente, sendo responsáveis por processar dados e tomar as decisões baseadas na sua programação.

Ao contrário de seus irmãos microprocessadores, os microcontroladores são dispositivos mais simples, com memórias RAM e ROM internas, oscilador interno de clock, I/O interno, entre outros. Tais características tornam mais simples o projeto de dispositivos inteligentes, pois os MCU's raramente precisam de CI's externos para funcionar, o que contribui para diminuição de custos e tamanho. (PEREIRA, 2001, p.18).

Devido suas características, o microcontrolador é uma alternativa viável para uso no sistema de automação residencial, sendo o componente responsável por tomar as decisões centrais, podendo ser programado adaptando-se a possíveis alterações no projeto. Além disso, o microcontrolador é um componente de custo acessível, e sua incorporação contribui para a diminuição dos custos no projeto de automação.

Existem diversos modelos e fabricantes de microcontroladores comerciais, alguns estão disponíveis em placas de prototipagem, que facilitam o desenvolvimento de projetos microcontrolados.

3.1.1 – Plataforma de desenvolvimento Arduino

A plataforma Arduino é uma placa contendo um microcontrolador AVR da Atmel (figura 1), em conjunto com circuitos de entrada e saída, além de possuir suporte para outras placas conhecidas como Shields, que expandem as funcionalidades do Arduino.

Esta plataforma utiliza o conceito de Software e Hardware Open Souce, contribuindo para o baixo custo (menos de R\$ 50,00) e para a facilidade de desenvolvimento, uma vez que a placa pode ser programada (em linguagem baseada em C/C++) pelo computador, utilizando-se apenas um cabo USB e o software disponibilizado livremente no site oficial.

Figura 1 - Arduino Uno R3

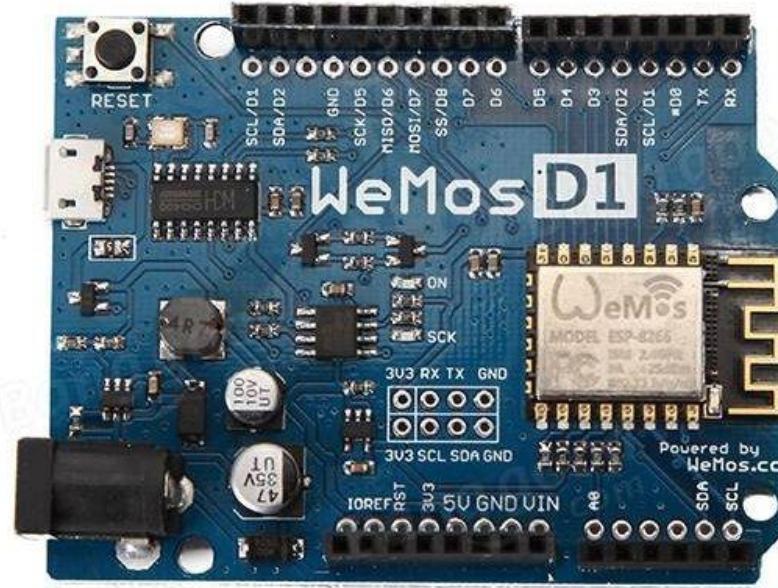


Fonte: <https://store.arduino.cc/arduino- uno-rev3>

3.1.2 – Plataforma WeMos e ESP8266

Devido ao conceito Open Souce do Arduino várias plataformas foram desenvolvidas utilizando-o como base. Dentre estas plataformas destaca-se a WeMos (Figura 2), que utiliza o layout do Arduino Uno, porém com um microcontrolador ESP8266 substituindo o Atmel. Este chip possui como bônus a capacidade de comunicação via Wi-fi, sem a necessidade de Shields para tal.

Figura 2 - WeMos D1 R2



Fonte: <http://www.instructables.com/id/P>

A plataforma WeMos foi escolhida para o desenvolvimento do projeto, por possuir as mesmas facilidades do Arduino (Podendo inclusive ser programada pela IDE Arduino) e também por possibilitar o desenvolvimento de um sistema de comunicação sem fio de forma mais prática.

3.2 – DISPOSITIVOS MÓVEIS

A cada dia, um número maior de pessoas interessa-se pela mobilidade, o fácil acesso às informações em qualquer lugar, com alcance amplo a qualquer hora, se conectando de forma fácil e rápida a outros dispositivos móveis, localizando pessoas, produtos e serviços personalizados. Estes são os fatores que impulsionam a internet móvel a se estruturar e crescer rapidamente para adaptar às modernidades e necessidades dos usuários finais, bem como das organizações. (ALCANTÂRA e VIEIRA, 2011).

Em um projeto de automação residencial, deve-se considerar que o usuário irá comandar o sistema de diferentes partes da residência, ou até mesmo estando distante dela. Para isso pode ser empregado um dispositivo móvel como plataforma de interação do usuário com o sistema. Nesse dispositivo será instalada uma interface de controle, onde o usuário entra com os comandos que o dispositivo irá enviar pela rede para serem processados e executados pelos controladores.

Atualmente existe um grande número de dispositivos móveis que carregam distintos sistemas operacionais. Dentre estes dispositivos, foi escolhido o smartphone para a realização do projeto, por ser o aparelho móvel mais comum no brasil segundo dados da Global Mobile Consumer Survey de 2017, e possuir grande capacidade de processamento, atendendo as necessidades do projeto.

3.2.1 – Sistema Operacional Android

O Android é um sistema operacional desenvolvido pela Google, baseado no kernel Linux, com foco nos dispositivos móveis sensíveis ao toque. Segundo dados da Statcounter, o Android é o sistema mais utilizado no mundo, levando em conta todos os dispositivos com acesso à internet. O Android possui muitas funcionalidades inovadoras e combina funções presentes em outros sistemas de um jeito livre e aberto. Nem os desenvolvedores nem os fabricantes de dispositivos pagam royalties ou taxas de licenciamento para desenvolver para a plataforma. (DARCEY e CONDER, 2012, p. 18) Assim o desenvolvimento de um aplicativo para esse sistema se torna mais viável.

3.3 – INTERFACE VISUAL E COMANDO DE VOZ

Os sistemas domóticos foram pensados para o maior conforto e comodidade do usuário, portanto, as plataformas de controle do sistema são desenvolvidas com este mesmo propósito. Visando a facilidade no controle do usuário sobre o sistema, é necessário estabelecer uma conexão entre estes, e para isso serve a interface de controle.

A interface deve ser intuitiva, para que qualquer pessoa consiga utilizá-la sem dificuldades, sendo assim, é recomendável utilizar como base uma plataforma já difundida, como os smartphones, que são usados no dia a dia de 87% dos brasileiros (Global Mobile Consumer Survey, 2017). Os smartphones disponibilizam milhares de aplicações, que normalmente tem interfaces parecidas, justamente para facilitar o uso. Essas aplicações são desenvolvidas pensando na tela sensível ao toque, por isso é comum encontrar botões para interação do usuário.

Recentemente, as aplicações têm se tornado mais interativas, utilizando outros recursos dos smartphones além da tela, como por exemplo as câmeras, sensores e microfones. Um dos recursos que tem sido muito explorado é o comando de voz, atualmente todos os grandes sistemas operacionais (Windows, IOS, Android) possuem as chamadas “assistentes pessoais”,

que são controladas pela voz do usuário, essas assistentes tem como função executar uma tarefa que é designada pela fala do usuário, podendo abrir aplicativos, mudar configurações, fazer pesquisas e até enviar mensagens.

Uma forma de tornar o sistema de controle de um projeto de automação residencial mais interativo é trazer o comando de voz para o sistema, desse jeito, além das funções já apresentadas, é possível executar comandos como acender lâmpadas, ligar aparelhos, ajustar a temperatura e outros. Além disso, a utilização de um aplicativo de smartphone com uma interface padrão presente na maioria dos aplicativos, aumenta a comodidade do usuário, que tem a possibilidade de controlar um ambiente por meio da fala ou pelos toques na tela do celular.

4 - COMUNICAÇÃO

Para a integração dos dispositivos que compõem o sistema de domótica, é necessário estabelecer uma comunicação entre eles, ou seja, todos os dispositivos devem estar conectados a uma rede. O uso de redes Wi-fi tem diminuído o custo dos projetos devido a facilidade de implementação, além de deixar o ambiente mais “limpo” por dispensar o uso de cabos.

O padrão IEEE 802.11 é usado para montagem de redes locais sem fio usando transmissão por ondas de rádio (RF). A taxa de transferência e o alcance dependem do padrão usado na camada física da rede (IEEE 802.11b, IEEE 802.11g, etc.), do ambiente e do tipo de antena usado. (TORRES, 2014, p. 126)

Uma rede doméstica normalmente é do tipo difusão. Este modelo de conexão possui apenas um canal de comunicação compartilhado por todas as máquinas desta rede (TANENBAUM, 2003). No desenvolvimento do projeto de AR, é estabelecida uma rede na qual a interface de controle do sistema (Smartphone) comunica-se com um roteador que direciona o pacote de dados proveniente do smartphone para o microcontrolador que irá interpretar estes dados e efetuar o controle dos atuadores.

Neste projeto, os dados são fornecidos pelo sistema de controle, que cria uma página HTML e envia o endereço para o roteador, por sua vez o microcontrolador lê o endereço fornecido e com isso toma a decisão de atuação, baseada na programação feita.

5 – DESENVOLVIMENTO

Este capítulo aborda o desenvolvimento de um sistema de automação exemplificado, com os subcapítulos demonstrando o estabelecimento da comunicação e a construção do aplicativo de controle além da programação do microcontrolador.

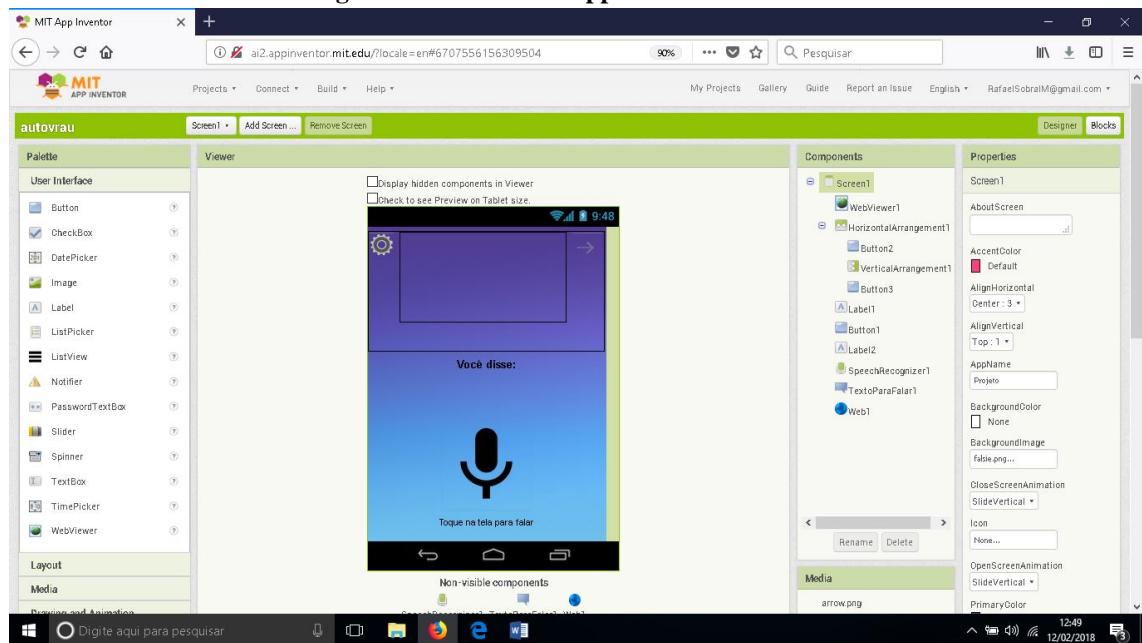
Os programas criados para este projeto estão disponíveis de forma integral nos apêndices.

5.1 – APlicativo ANDROID

Para a interação do usuário com o sistema, foi desenvolvido um aplicativo compatível com *Tablets* e *Smartphones* que executem o sistema Android. O desenvolvimento do aplicativo foi possível pelo uso da plataforma AppInventor 2 disponibilizada pelo MIT.

O AppInventor facilita o desenvolvimento de aplicativos para celular, podendo ser acessado pelo navegador de internet (Figura 3) e possuindo diversas ferramentas que adicionam várias funcionalidades ao aplicativo, além de mostrar uma prévia do layout para *Tablets* e *Smartphones*.

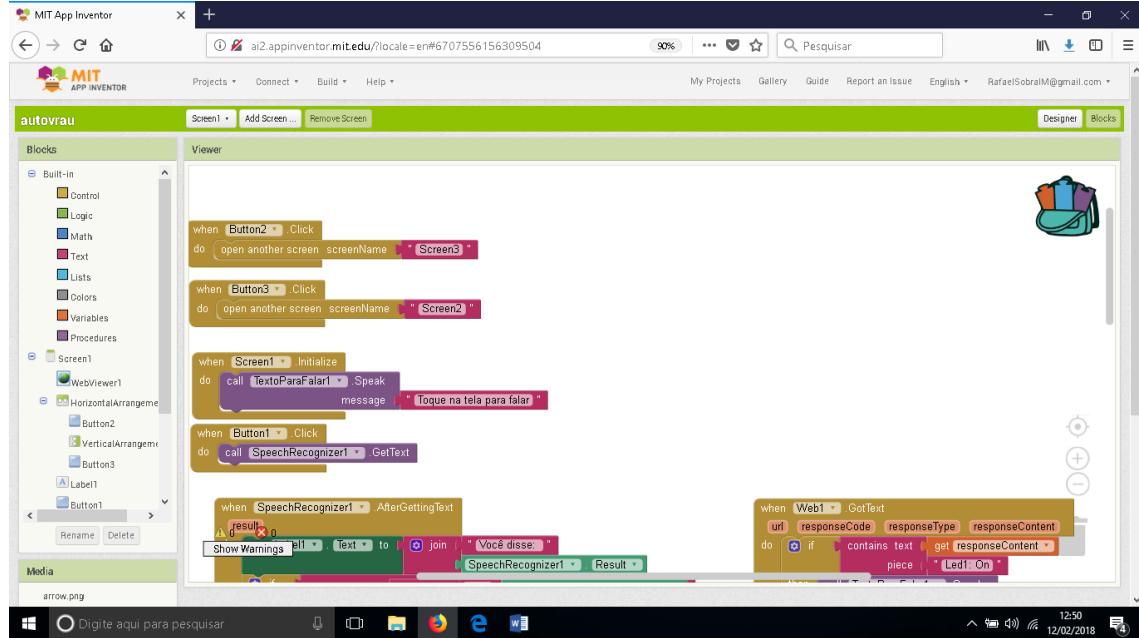
Figura 3 - Plataforma AppInventor2



Fonte: O Autor

O App inventor utiliza blocos para a programação das funções da aplicação (Figura 4), estes blocos estão ligados a interface visual e podem ser combinados para a definição das funcionalidades dos elementos visuais.

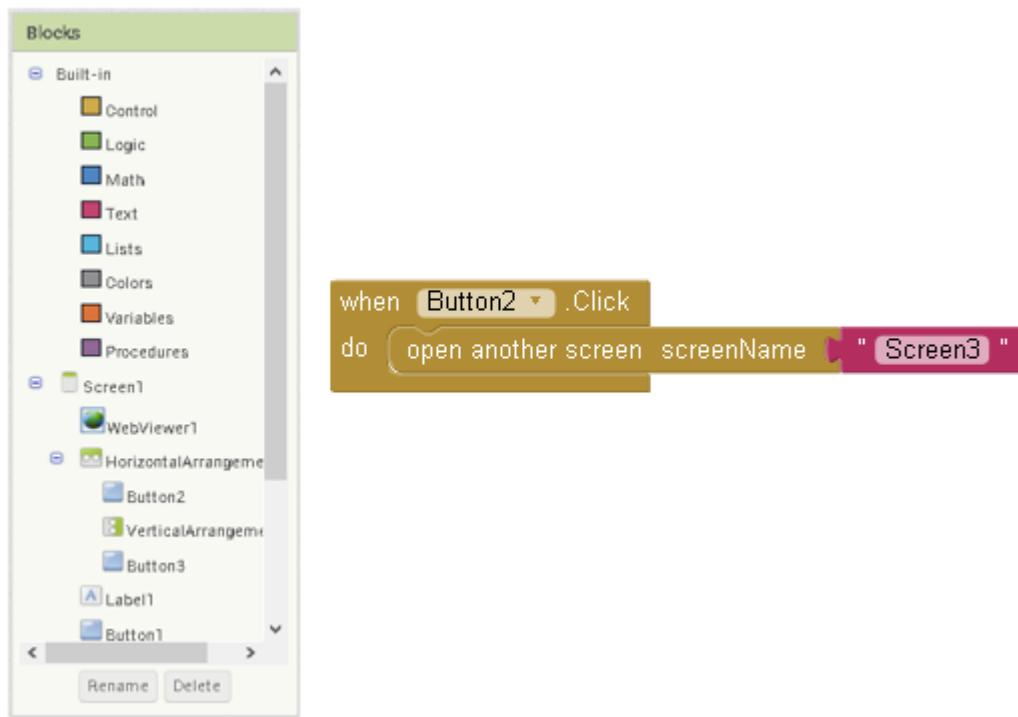
Figura 4 - Página de blocos do AppInventor



Fonte: O Autor

Na Figura 5 estão em destaque a lista com as várias possibilidades de blocos e um exemplo de combinação entre os blocos. Neste exemplo é possível observar que ao juntar 3 blocos é definida a função de um botão. Quando o Botão2 é clicado, outra tela do aplicativo vai se abrir.

Figura 5 - Exemplo de Combinação de Blocos



Fonte: O Autor

5.1.1 - Funcionalidades do Aplicativo

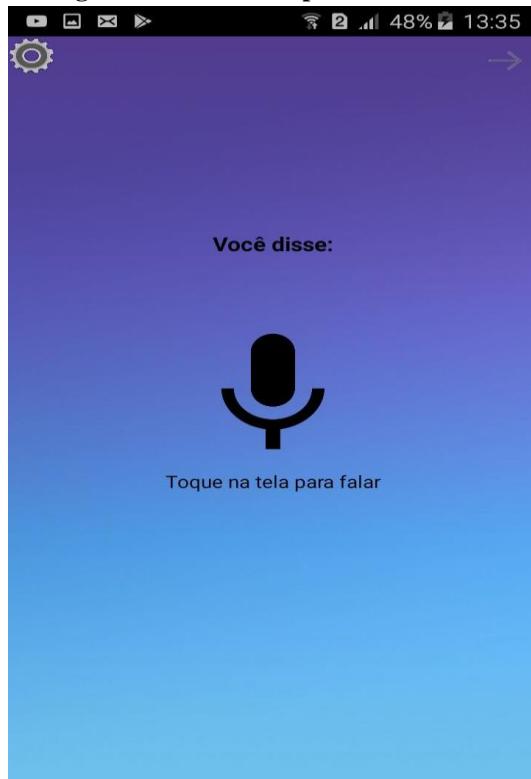
O aplicativo desenvolvida possui três telas diferentes, as duas primeiras são as telas de comando do sistema de automação, e a terceira é uma tela de ajuda contendo algumas orientações ao usuário e configurações do sistema.

A primeira tela (Figura 6) apresenta a funcionalidade do comando por voz, quando o usuário toca na imagem o celular começa a captar a voz que será analisada para verificar se corresponde a algum comando pré-definido, caso correspondente o comando será executado.

Quando o aplicativo é iniciado esta tela se abre e o celular narra uma frase instruindo o usuário a tocar na tela. A narração também é utilizada para informar o estado atualizado das lâmpadas. Estes recursos podem ser utilizados como ferramenta de acessibilidade, permitindo a interação de pessoas com problemas de visão com o sistema de automação.

Nessa tela ainda estão presentes dois outros botões, com a engrenagem direcionando para a tela de ajuda e a seta direcionando para a segunda tela.

Figura 6 - Tela 1 do Aplicativo



Fonte: O Autor

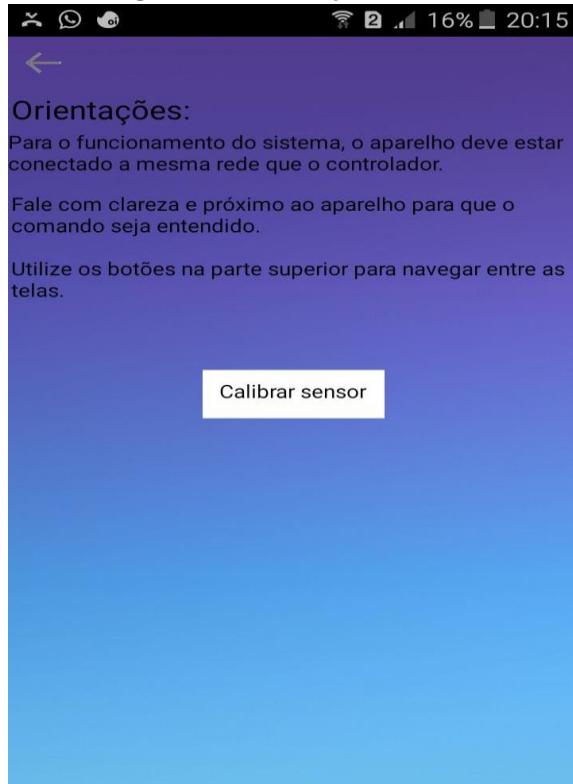
A segunda tela (Figura 7) apresenta as mesmas funcionalidades da primeira, mas com a substituição do comando de voz pelo comando por botões. Além dos botões para navegação entre telas, foram adicionados 7 botões que nesse caso controlam lâmpadas. Nessa tela o usuário tem a possibilidade de controlar cada lâmpada individualmente, verificando seu estado através das imagens de lâmpadas presentes na tela e podendo liga-las totalmente ou com metade da potência e desliga-las, ou então ativar o controle automático, no qual as lâmpadas serão acesas ou apagadas de acordo com os dados captados por um sensor de luminosidade.

Luzes foram utilizadas neste projeto, a fim de testar a funcionalidade do sistema, contudo, este projeto oferece a possibilidade de incorporar mais elementos ao sistema. Utilizando-se do mesmo princípio de controle de luminosidade, pode-se, por exemplo, controlar a climatização de um ambiente, acionando aparelhos de ar-condicionado e comandando sua ativação e intensidade.

Figura 7 - Tela 2 do Aplicativo

Fonte: O Autor

A última tela (Figura 8) é acessada pelos botões representados por um sinal de interrogação, esta é a tela de ajuda ao usuário, contendo algumas recomendações para a utilização do sistema e uma lista com todos os comandos que podem ser ditos para o controle por voz, além de um botão cuja a função é regular o sensor de luminosidade, adaptando-o para o ambiente desejado.

Figura 8 - Tela de ajuda

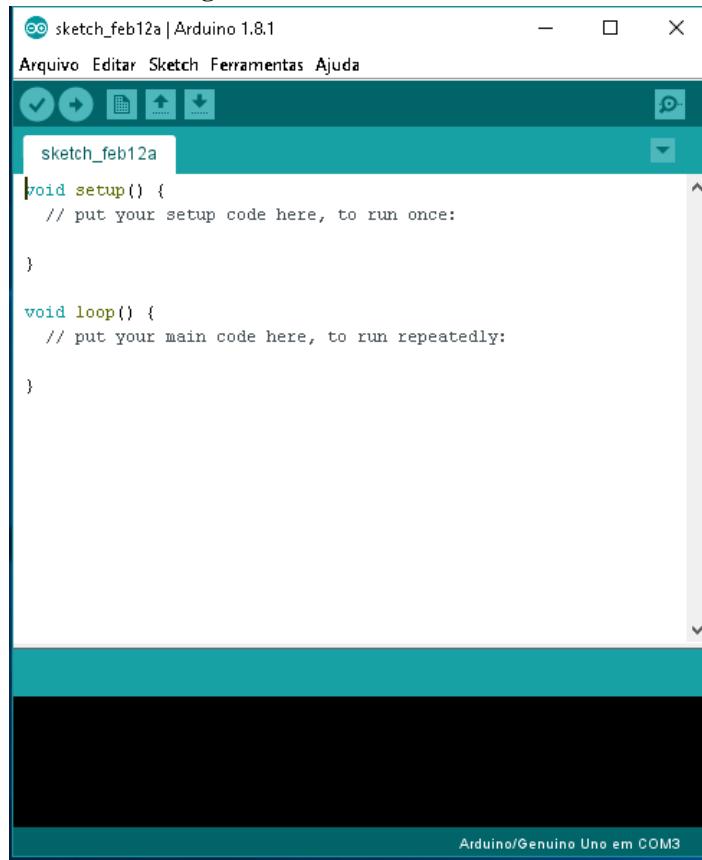
Lista de Comandos

Fonte: O Autor

5.2 – PROGRAMAÇÃO DO CONTROLADOR

Neste Projeto foi utilizado o microcontrolador ESP8266, disponível na placa de prototipagem WeMos D1 R1. A programação dessa plataforma é realizada utilizando a linguagem C/C++, e para implementação do código na placa foi usada a IDE Arduino (Figura 9).

O código desenvolvido pode ser dividido em duas partes: A primeira estabelece a conexão da placa com a rede, e a segunda parte é responsável pelo gerenciamento do sistema, interpretando os dados recebidos e executando a rotina determinada.

Figura 9 - IDE Arduino

Fonte: O Autor

5.2.1 – Estabelecimento da Comunicação

A Wemos oferece conectividade Wi-Fi nativa, ou seja, basta configurar a conexão para que a placa comece a receber e a enviar dados. Para isso basta informar o ID e a senha da rede sem fio, configurar um IP válido e iniciar o servidor. O destaque na figura 10 exemplifica estas configurações.

Figura 10 - Configuração da Conexão

```

heee | Arduino 1.8.1
Arquivo Editar Sketch Ferramentas Ajuda
heee
int ledPin3 = D5;
int valor = 0;
int ldr = A0;
boolean iniciar = false;
const char* ssid = "Net 0302";
const char* password = "125303020";

WiFiServer server(80);
IPAddress ip(192, 168, 0, 166);
IPAddress gateway(192, 168, 0, 1);
IPAddress subnet(255, 255, 255, 0);

void setup() {
  Serial.begin(115200);
  WiFi.config(ip, gateway, subnet);
  WiFi.begin(ssid, password);
  server.begin();
}

```

Arduino/Genuino Uno em COM3

Fonte: O Autor

Uma vez que a comunicação foi estabelecida, foi configurada uma página HTML para o envio e o recebimento de dados. Os comandos são definidos pelo final do endereço e o envio de dados se dá pelas informações contidas na página.

A imagem 11 demonstra a página HTML acessada por um navegador de internet. Na barra de endereço se vê o IP da página, e o conteúdo são dados fornecidos pela placa.

Figura 11 - Página HTML

Led1: Off

Led2: Off

Led3: Off

Luminosidade: 90



Fonte: O Autor

5.2.2 - Software de Controle

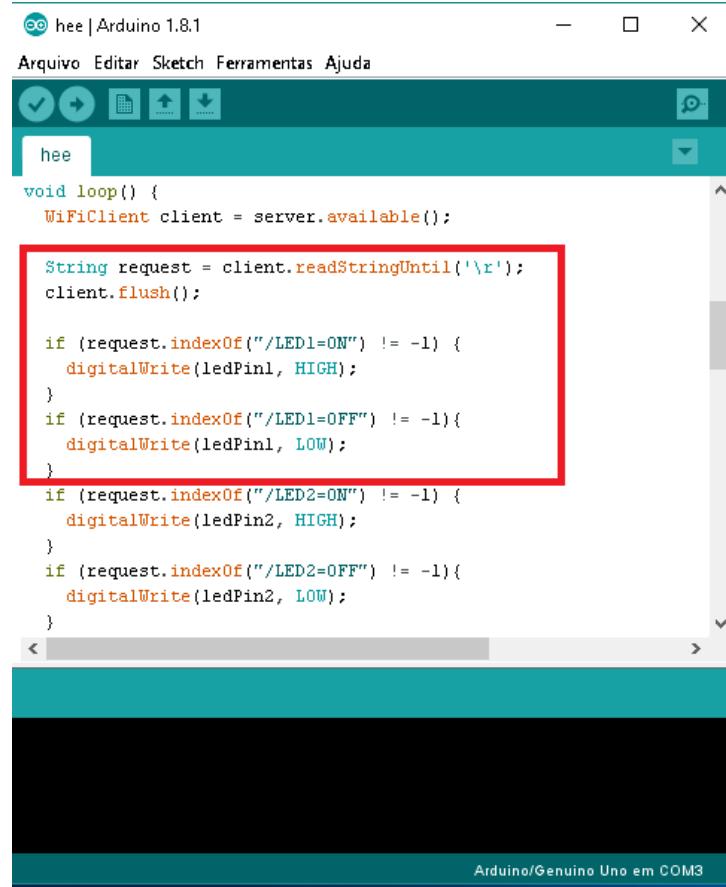
Aqui será abordada a parte do código responsável por interpretar os dados recebidos e executar os devidos comandos.

No código desenvolvido neste projeto, os comandos são definidos pelo endereço da página HTML, assim programa lê a parte final do endereço e faz a comparação para executar o comando correspondente.

O destaque na Imagem 12 mostra a parte do código em que o endereço é lido, e alguns comandos disponíveis. Nesse caso SE o final do endereço for LED1=ON então o LED1 será aceso. Se o endereço for LED1=OFF o mesmo LED será apagado. O mesmo princípio é utilizado em todos os comandos presentes no sistema, inclusive no controle automático e no controle de potência das cargas.

Caso o endereço acessado tenha o final representado pelo comando SENSOR, então a função do controle automático de luminosidade será executada. Nesse caso as luzes serão ligadas de acordo com a luminosidade do ambiente, captada por um sensor.

Figura 12 - Leitura e execução de comandos



```

heee | Arduino 1.8.1
Arquivo Editar Sketch Ferramentas Ajuda
heee
void loop() {
    WiFiClient client = server.available();

    String request = client.readStringUntil('\r');
    client.flush();

    if (request.indexOf("/LED1=ON") != -1) {
        digitalWrite(ledPin1, HIGH);
    }
    if (request.indexOf("/LED1=OFF") != -1){
        digitalWrite(ledPin1, LOW);
    }
    if (request.indexOf("/LED2=ON") != -1) {
        digitalWrite(ledPin2, HIGH);
    }
    if (request.indexOf("/LED2=OFF") != -1){
        digitalWrite(ledPin2, LOW);
    }
}

```

Arduino/Genuino Uno em COM3

Fonte: O Autor

5.3 – TESTES DE FUNCIONAMENTO

Para testar o sistema foram utilizados componentes vistos na figura 13, com os LEDs na função de atuadores, representando os equipamentos disponíveis em uma residência que podem ser controlados pelo sistema de automação. A função de controle automático dos LEDs foi executada com a utilização de um LDR, que varia sua resistência em função da luminosidade.

Figura 13 - Circuito Utilizado para Testes



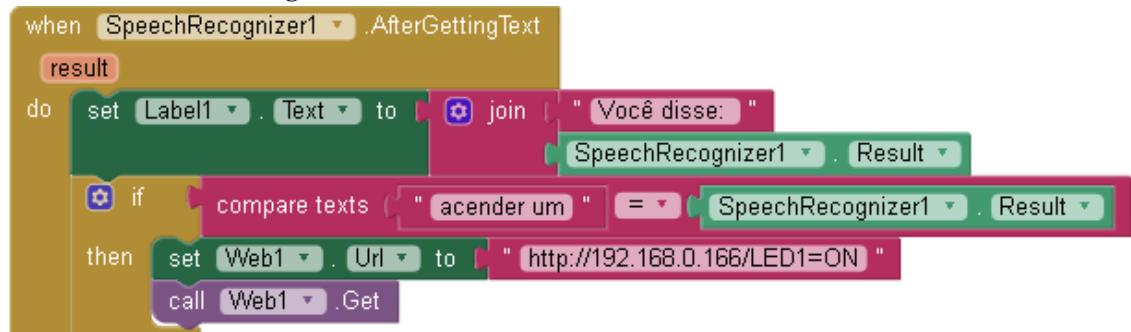
Fonte: O Autor

O processo para a execução de um comando somente é possível se os dispositivos envolvidos (celular e microcontrolador) estiverem conectados à mesma rede. Uma vez estabelecida a conexão, o usuário pode controlar o sistema pelo aplicativo, usando o comando por voz ou os botões. O princípio de funcionamento desses recursos é praticamente o mesmo.

Ao ativar o comando por voz, o aplicativo vai interpretar o que foi dito e fará algumas comparações.

A figura 14 mostra parte dos blocos responsáveis por comparar o que foi dito e enviar o comando. Nesse exemplo o programa compara o resultado da captura de voz com o texto “acender um”, caso compatível, o aplicativo irá acessar a página WEB no endereço indicado.

Figura 14 - Blocos de reconhecimento de voz



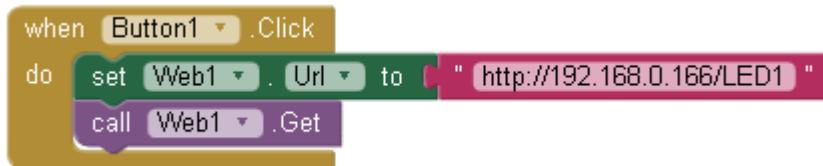
Fonte: O Autor

Na figura 15 é possível notar a semelhança entre o comando por voz e por botões, a única diferença é ao invés de comparar palavras, o programa apenas verifica se o botão é pressionado, somente nesse caso o aplicativo acessa a página HTML.

Quando o endereço é acessado, o microcontrolador compara a String no final do endereço e executa o comando definido na sua programação.

Uma vez que o comando é executado, o aplicativo deve receber um retorno confirmando o êxito na operação, para isso o controlador analisa a atuação dos LEDs, e escreve o estado atualizado na página HTML.

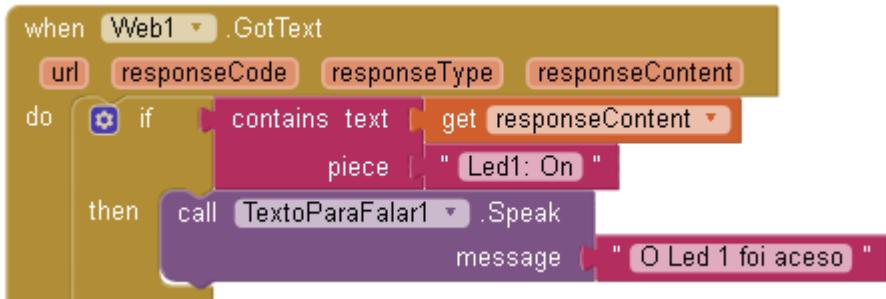
Figura 15 - Blocos de funcionalidade do Botão



Fonte: O Autor

Depois que a página é atualizada, o aplicativo verifica a página procurando pelo texto que informa o estado do LED. Os blocos na figura 16 mostram a programação para que o aplicativo narre o estado atual, este recurso de narração é utilizado apenas na primeira tela, em conjunto ao comando de voz. No caso da segunda tela, a resposta do aplicativo aos comandos por botões se dá pelas imagens que aparecem na tela, representando o estado do LED.

Figura 16 - Blocos de Resposta ao Comando



Fonte: O Autor

6 – CONCLUSÃO

O desenvolvimento de novas tecnologias tem influenciado cada vez mais no dia a dia da sociedade. A todo instante são criadas novas possibilidades para facilitar a vida das pessoas. Nesse contexto, a popularização da automação residencial se apresenta como o próximo passo no sentido de trazer conforto e segurança por meio da tecnologia. Para que este processo aconteça é necessário o estudo de maneiras alternativas de se desenvolver um sistema de AR. O estudo de algumas destas possibilidades foi um dos objetivos alcançados por este trabalho.

O projeto obteve êxito na exploração de recursos para automação residencial simples e de baixo custo. Também foi possível criar uma interface para smartphone que além de ser intuitiva para o usuário comum estando presente em uma plataforma popular, traz recursos de acessibilidade para pessoas com necessidades especiais. Estes usuários podem controlar um ambiente utilizando apenas a voz para tal, e ainda podem verificar a atuação dos componentes do sistema escutando a narração do celular.

Os objetivos propostos só foram alcançados pelo uso de conhecimentos adquiridos durante todo o curso técnico. O projeto foi estimulado por teorias aprendidas principalmente nas disciplinas que envolvem programação, estes conceitos foram postos em prática e se mostram importantes para o desenvolvimento profissional.

6.1 – PERPESCTIVAS

Este projeto pode ser adaptado para controle real de uma residência, para isso é necessário fazer algumas melhorias como:

- Substituir os LEDs por relés que atuam controlando lâmpadas
- Adicionar um circuito de Infra-Vermelho para controle de Ar-Condicionado
- Adicionar Alarmes
- Adicionar Sensores de Temperatura e Humidade do Ar
- Adaptar o Aplicativo a estas adições e desenvolvê-lo para mais plataformas.

REFERÊNCIAS

ALCANTARA, Carlos; VIEIRA, Anderson. **Tecnologia Móvel: Uma tendência, uma realidade.** Juíz de Fora. 2011

ANGEL, Patrícia Marta. **Introducción a la domótica;** Escuela Brasileño-Argentina de Informática. EBAI. 1993.

BOLZANI, C.A.M. **Desmistificando a domótica;** Sinergia. São Paulo. 2007

BOLZANI, C.A.M. **Residências Inteligentes;** Livraria da Física. São Paulo. 2004

DARCEY, Lauren; CONDER, Shane. **Desenvolvimento de Aplicativos Wireless para Android Vol. 1 – Fundamentos do Android;** Editora Ciência Moderna. Rio de Janeiro. 2012

DELOITTE. **Mobile Consumer Survey 2017.** Disponível em: <<https://www2.deloitte.com/br/pt/pages/technology-media-and-telecommunications/articles/mobile-survey.html>> Acesso em: 15 de Fevereiro de 2018

PENIDO, Édilus; TRINDADE, Ronaldo. **Microcontroladores;** Ouro Preto. 2013

PEREIRA, Fábio. **Microcontroladores PIC: Técnicas Avançadas;** Érica. São Paulo. 2001

STATCOUNTER. **Operating System Market Share Worldwide.** Disponível em: <gs.statcounter.com/os-market-share/> Acesso em: 17 de Janeiro de 2018

TANENBAUM, Andrew. S. **Redes de Computadores.** Editora Campus. 2003

TORRES, Gabriel. **Redes de Computadores;** Novaterra. Rio de Janeiro. 2014

WORTEMEYER, C.; FREITAS, F.; CARDOSO, L.; **Automação Residencial: Busca de Tecnologias visando o Conforto, a Economia, a Praticidade e a Segurança do Usuário,** II Simpósio de Excelência em Gestão e Tecnologia. SEGeT, 2005.

APÊNDICE A – CÓDIGO DO MICROCONTROLADOR

```

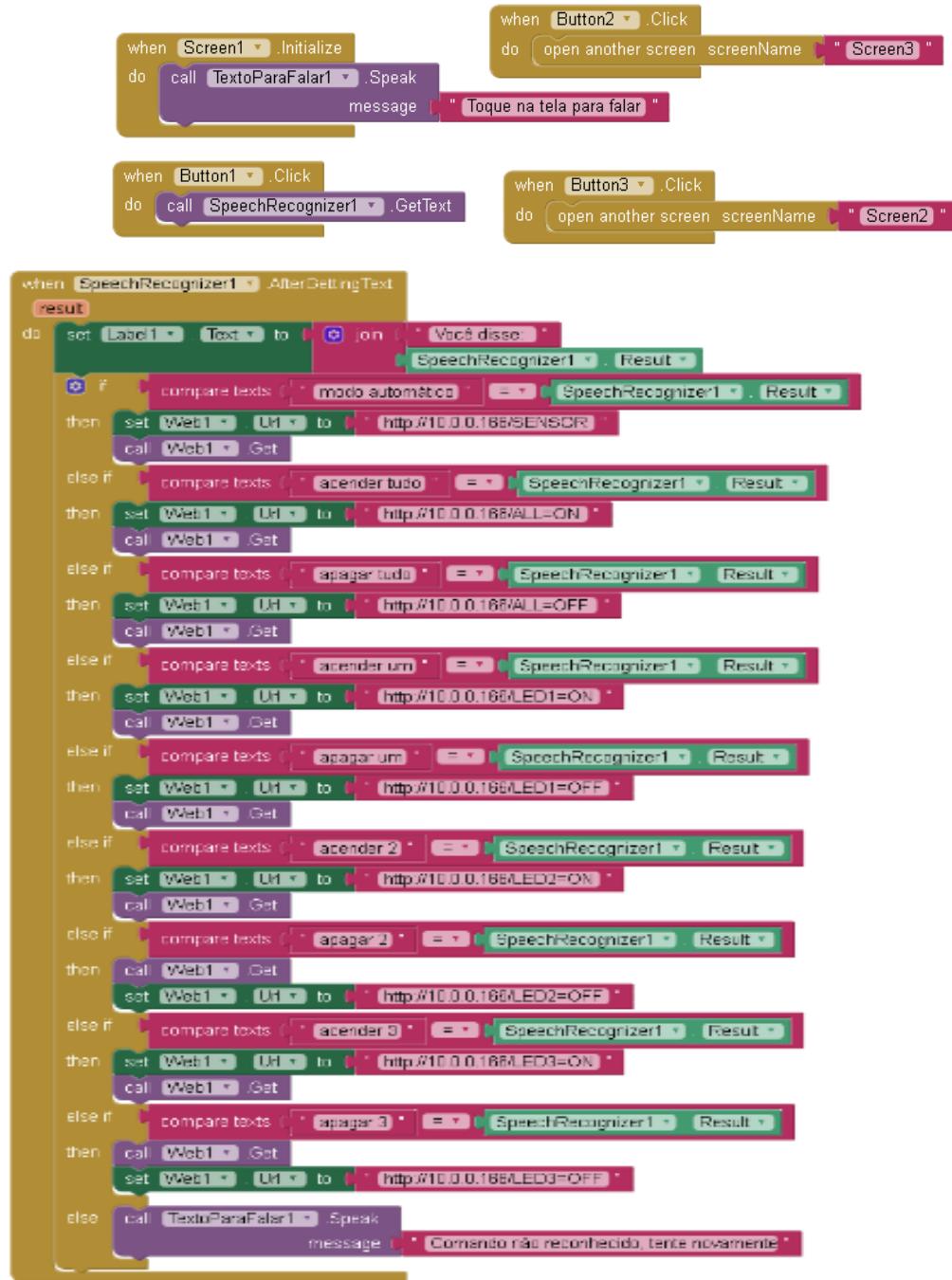
1. #include <ESP8266WiFi.h> // inclui a biblioteca ESP8266
2. int ledPin1 = D2;
3. int ledPin2 = D3; // definição dos Pinos
4. int ledPin3 = D4;
5. int ldr = A0;
6. int valor = 0; // inicializa variáveis com 0
7. int lumi = 0;
8. boolean automatico = false;
9.
10. const char* ssid = "Betoshow";
11. const char* password = "125303020";
12.
13. WiFiServer server(80);
14. IPAddress ip(10, 0, 0, 166); // configuração do IP
15. IPAddress gateway(10, 0, 0, 1); // IP do computador
16. IPAddress subnet(255, 255, 255, 0); // máscara de subrede
17.
18. void setup() {
19.   WiFi.config(ip, gateway, subnet);
20.   WiFi.begin(ssid, password); // inicializa o servidor com o IP e Mac
21.   server.begin();
22.
23.   pinMode(ledPin1, OUTPUT);
24.   pinMode(ledPin2, OUTPUT); // Configura pinos como Saída
25.   pinMode(ledPin3, OUTPUT);
26. }
27.
28. void loop() {
29.   WiFiClient client = server.available(); // Verifica se a conexão foi estabelecida
30.
31.   String request = client.readStringUntil('\r'); // Verifica o final do endereço HTTP
32.   client.flush();
33. // Configurações dos Comandos
34.   if (request.indexOf("/calibrar") != -1 ){
35.     lumi = analogRead (ldr);
36.   }
37.   if (request.indexOf("/LED1=ON") != -1) {
38.     digitalWrite(ledPin1, HIGH);
39.   }
40.   if (request.indexOf("/LED1=OFF") != -1){
41.     digitalWrite(ledPin1, LOW);
42.   }
43.   if (request.indexOf("/LED2=ON") != -1) {
44.     digitalWrite(ledPin2, HIGH);
45.   }
46.   if (request.indexOf("/LED2=OFF") != -1){
47.     digitalWrite(ledPin2, LOW);
48.   }
49.   if (request.indexOf("/LED3=ON") != -1) {
50.     digitalWrite(ledPin3, HIGH);
51.   }
52.   if (request.indexOf("/LED3=OFF") != -1){
53.     digitalWrite(ledPin3, LOW);
54.   }
55.   if (request.indexOf("/LED.1") != -1){
56.     digitalWrite(ledPin1, !digitalRead(ledPin1));
57.   }
58.   if (request.indexOf("/LED.2") != -1){
59.     digitalWrite(ledPin2, !digitalRead(ledPin2));
60.   }

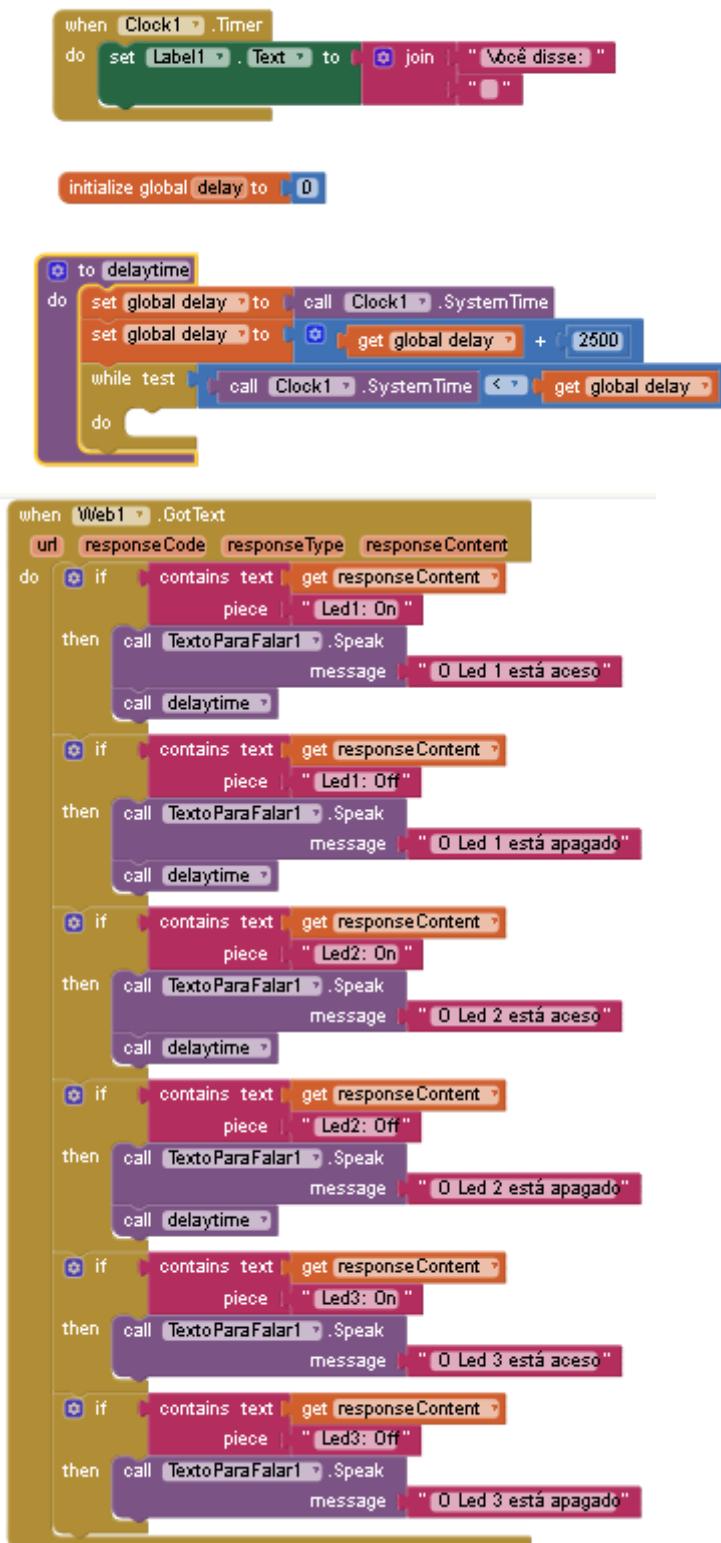
```

```
61.  if (request.indexOf("/LED.3") != -1){
62.      digitalWrite(ledPin3, !digitalRead(ledPin3));
63.  }
64.  if (request.indexOf("/LED/1") != -1){
65.      if(analogRead(ledPin1) == 0) analogWrite(ledPin1, 512);
66.      else analogWrite(ledPin1, 0);
67.  }
68.  if (request.indexOf("/LED/2") != -1){
69.      if(analogRead(ledPin2) == 0) analogWrite(ledPin2, 512);
70.      else analogWrite(ledPin2, 0);
71.  }
72.  if (request.indexOf("/LED/3") != -1){
73.      if(analogRead(ledPin3) == 0) analogWrite(ledPin3, 512);
74.      else analogWrite(ledPin3, 0);
75.  }
76.  if(request.indexOf("/ALL=OFF") != -1){
77.      digitalWrite(ledPin1,LOW);
78.      digitalWrite(ledPin2,LOW);
79.      digitalWrite(ledPin3,LOW);
80.  }
81.  if(request.indexOf("/ALL=ON") != -1){
82.      digitalWrite(ledPin1,HIGH);
83.      digitalWrite(ledPin2,HIGH);
84.      digitalWrite(ledPin3,HIGH);
85.  }
86.  if(request.indexOf("/SENSOR") != -1) automatico = !(automatico);
87.  valor = analogRead(ldr);
88.  if((valor >= lumi) && (automatico)){
89.      digitalWrite(ledPin1,LOW);
90.      digitalWrite(ledPin2,LOW);
91.      digitalWrite(ledPin3,LOW);
92.  }
93.  if((valor < lumi) && (automatico)){
94.      digitalWrite(ledPin1,HIGH);
95.      digitalWrite(ledPin2,HIGH);
96.      digitalWrite(ledPin3,HIGH);
97.  }
98. //Configurações da Página HTML
99. client.println("HTTP/1.1 200 OK");
100. client.println("Content-Type: text/html");
101. client.println(""); // do not forget this one
102. client.println("<html>");
103. client.print("Led1: ");
104.     if(digitalRead(ledPin1) == HIGH) client.print("On");
105.     else client.print("Off");
106. client.println("<br><br>");
107. client.print("Led2: ");
108.     if(digitalRead(ledPin2) == HIGH) client.print("On");
109.     else client.print("Off");
110. client.println("<br><br>");
111. client.print("Led3: ");
112.     if(digitalRead(ledPin3) == HIGH) client.print("On");
113.     else client.print("Off");
114. client.println("<br><br>");
115. client.print("Luminosidade: ");
116. client.print(valor);
117. client.print(" <meta http-
equiv=\"refresh\" content=\"10; url=http://10.0.0.166/\"> ");
118. client.println("</html>");
119. client.stop();
120. }
```

APÊNDICE B – CÓDIGO DO APLICATIVO

Blocos da Tela 1:





Blocos da Tela 2:

```

when Button8 .Click
do open another screen screenName "Screen3"

when Button4 .Click
do open another screen screenName "Screen1"

when Button11 .Click
do set Web1 .Url to "http://10.0.0.166/LED.1"
call Web1 .Get

when Button9 .Click
do set Web1 .Url to "http://10.0.0.166/LED.2"
call Web1 .Get

when Button13 .Click
do set Web1 .Url to "http://10.0.0.166/LED.3"
call Web1 .Get

when Button12 .Click
do set Web1 .Url to "http://10.0.0.166/LED.1"
call Web1 .Get

when Button10 .Click
do set Web1 .Url to "http://10.0.0.166/LED.2"
call Web1 .Get

when Button14 .Click
do set Web1 .Url to "http://10.0.0.166/LED.3"
call Web1 .Get

when Button15 .Click
do set Web1 .Url to "http://10.0.0.166/SENSOR"
call Web1 .Get

```

```

when Web1 .GotText
url responseCode responseType responseContent
do if contains text get responseContent piece "Led1: On"
then set Image3 .Visible to false
set Image5 .Visible to true

if contains text get responseContent piece "Led1: Off"
then set Image3 .Visible to true
set Image5 .Visible to false

if contains text get responseContent piece "Led2: On"
then set Image6 .Visible to true
set Image2 .Visible to false

if contains text get responseContent piece "Led2: Off"
then set Image6 .Visible to false
set Image2 .Visible to true

if contains text get responseContent piece "Led3: On"
then set Image7 .Visible to true
set Image4 .Visible to false

if contains text get responseContent piece "Led3: Off"
then set Image7 .Visible to false
set Image4 .Visible to true

```

Blocos da Tela 3:

```

when Button1 .Click
do open another screen screenName "Screen1"

when Screen3 .Initialize
do set ListPicker1 .Elements to get global lista

initialize global lista to make a list
"Comandos de Voz: "
"Acender 1"
"Apagar 1"
"Acender 2"
"Apagar 2"
"Acender 3"
"Apagar 3"
"Acender Tudo"
"Apagar Tudo"
"Sensor"
"Calibrar"

when Button2 .Click
do set Web1 .Url to "http://10.0.0.166/calibrar"
call Web1 .Get

```