

**CENTRO PAULA SOUZA  
FACULDADE DE TECNOLOGIA DE FRANCA  
“DR. THOMAZ NOVELINO”**

**TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**ALETICIANA GENEROSO MENDES**

**AUTOMAÇÃO RESIDENCIAL**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Franca - “Dr. Thomaz Novelino”, como parte dos requisitos obrigatórios para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

**Orientador:** Prof. Me. Carlos Eduardo de França Roland.

**FRANCA/SP  
2020**

## Ficha catalográfica

M538a      Mendes, Aleticiana Generoso  
              Automação residencial / Aleticiana Generoso Mendes /  
              [ s.n.], 2020

44 f.; 30 cm; il

Trabalho de Graduação (Curso Superior de Análise e  
Desenvolvimento de Sistemas) Fatec - Faculdade de  
Tecnologia “Dr. Thomaz Novelino”.

Orientador: Prof. Me. Carlos Eduardo de França Rolan

1. Automação residencial. 2.Arduino. 3. Domótica.  
4.Sensores. I. Autor. II. Título.

CDD – 005.1

# AUTOMAÇÃO RESIDENCIAL

Aleticiana Generoso Mendes<sup>1</sup>

## Resumo

Com os avanços tecnológicos surgem várias possibilidades de uso da Tecnologia Digital de Informação e Comunicação como para a automação residencial também chamada de Domótica. A automação residencial é a aplicação de sistemas de controle baseados na automação de processos para as funções encontradas no ambiente residencial, integrando seus acionamentos e tornando a casa mais inteligente, funcional e segura. Visando a praticidade, a simplicidade de operação e realizando controle remoto através de *smartphones*, *tablets*, e computadores é possível utilizar essas tecnologias. O objetivo deste trabalho é a construção de um protótipo de automação residencial, utilizando plataforma microcontrolada, que possibilita o controle de acesso remoto através de uma aplicação web para acionamento dos componentes como iluminação, eletrodomésticos, controle do portão. A motivação para a escolha do tema foi porque, como a tecnologia está ao alcance de todos de uma forma acessível, a automação residencial não é mais um recurso de segurança patrimonial para usuários da classe alta ou de filmes de ficção futurista, e neste projeto buscou-se demonstrar que técnica e economicamente é possível fazer uso de sistemas de AR compostos de hardware e software dedicados. Os resultados esperados foram alcançados e novas e melhores funcionalidades são propostas em projetos futuros.

**Palavras-chave:** Automação residencial. Arduino. Domótica. Sensores.

## Abstract

*With technological advances, there are several possibilities for using Digital Information and Communication Technology as for home automation, also called Domotics. Home automation is the application of control systems based on process automation for the functions found in the residential environment, integrating its drives and making the home more intelligent, functional and safer. Aiming at practicality, simplicity of operation and performing remote control through smartphones, tablets, and computers it is possible to use these technologies. The objective of this work is the construction of a prototype of home automation, using a microcontrolled platform, which enables remote access control through a web application to activate components such as lighting, appliances, gate control. The motivation for choosing the theme was because, as technology is accessible to everyone in an accessible way, home automation is no longer a property security resource for users of the upper class or futuristic fiction films, and in this project sought - demonstrate that it is technically and economically possible to make use of AR systems composed of dedicated hardware and software.*

---

<sup>1</sup> Graduanda em Análise e Desenvolvimento de Sistemas pela Fatec Dr Thomaz Novelino – Franca/SP. Endereço eletrônico: aleticiana@gmail.com.

*The expected results have been achieved and new and better features are proposed in future projects.*

**Key-words:** *Arduino. Domotics. Home automation.*

## **1 Introdução**

Com os avanços tecnológicos surgem várias possibilidades de uso da Tecnologia Digital de Informação e Comunicação (TDIC), dentre elas para a automação residencial também chamada de Domótica. Este termo resulta da junção da palavra do latim *Domus* que tem o significado de casa, com a do tcheco *Robota* (servo), que é conjunto das técnicas que permitem a utilização de automação de equipamentos e processos domésticos (DOMINGUES, 2013).

Por volta dos anos 80 durante a construção dos primeiros grandes edifícios, teve-se início a implantação e utilização da Domótica, e a partir daí percebeu-se a necessidade de controlar e interligar as funções prediais aplicando-se também, a partir de 2010, às residências.

A Automação Residencial (AR) trata-se da aplicação de sistemas de controle baseados na automação de processos para as funções encontradas no ambiente residencial, integrando seus acionamentos e tornando a casa mais inteligente, funcional e segura. Visando a praticidade, a simplicidade dos comandos e realizando controle remoto através de *smartphones*, *tablets*, e computadores é possível utilizar essas tecnologias estando dentro ou fora da residência pela internet, que está ao alcance de todos.

O objetivo deste trabalho é a construção de um protótipo de automação residencial utilizando plataforma microcontrolada, que possibilita o controle de acesso remoto através de uma aplicação web dos comandos para os acionamentos dos componentes como iluminação, eletrodomésticos, controle do portão, e o acionamento, por sensor de presença, de iluminação de alerta na ocorrência de algum movimento no ambiente.

A motivação para a escolha do tema foi porque, como a tecnologia está ao alcance de todos de uma forma acessível, a automação residencial não é mais um recurso de segurança patrimonial para usuários da classe alta ou de filmes de ficção futurista, e neste projeto buscou-se demonstrar que técnica e economicamente é possível fazer uso de sistemas de AR compostos de hardware e software dedicados.

A estrutura deste trabalho apresenta-se em seções iniciando por esta Introdução, seguida da apresentação da viabilidade do projeto e dos artefatos de Engenharia de Software criados na fase de análise, que apresenta os Documentos de Requisitos e os Diagramas UML. Em seguida são apresentados os Materiais utilizados no protótipo (hardware) e os Métodos de Implementação da lógica de operação (software), concluindo com as Considerações Finais.

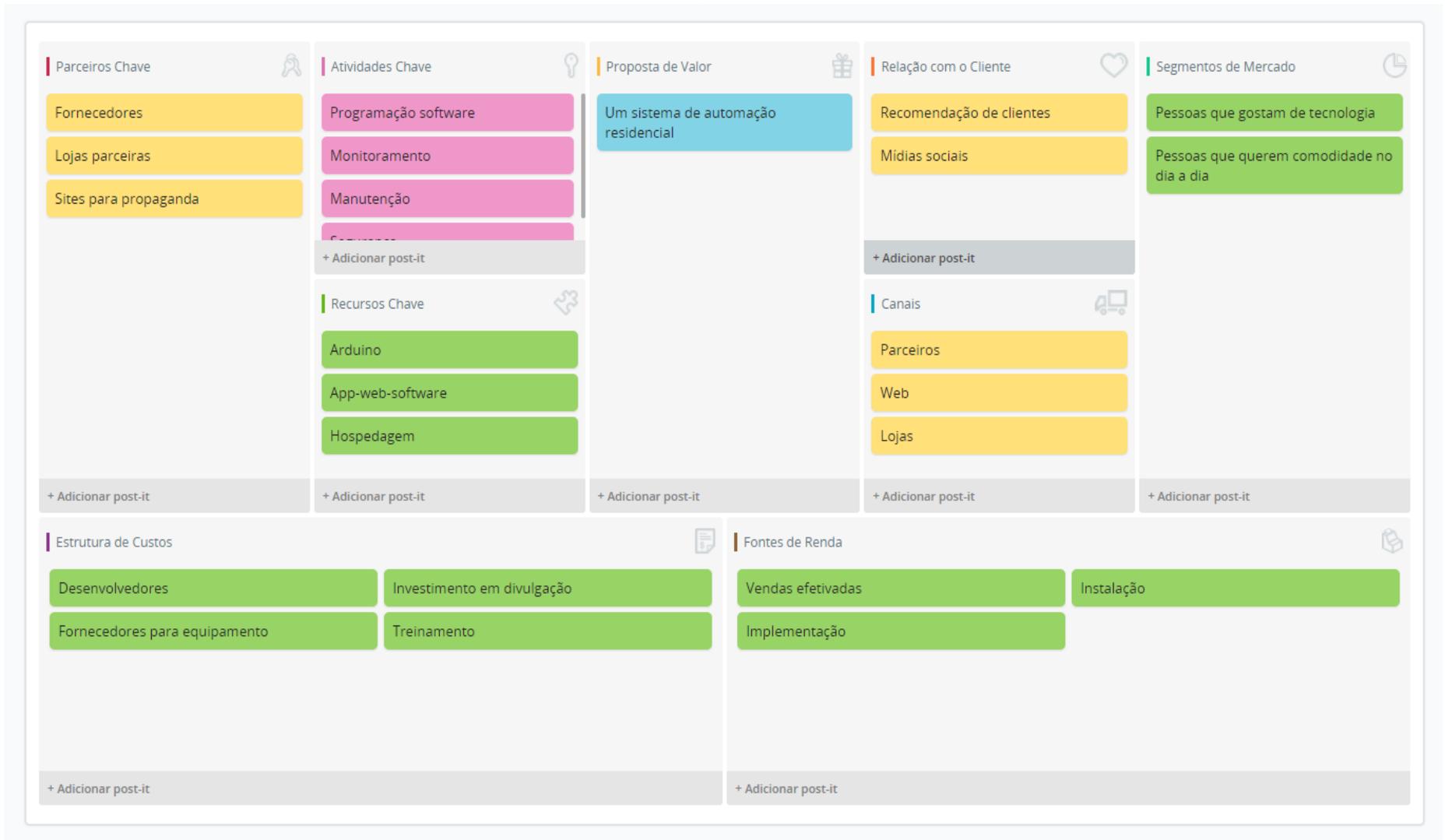
## **2 Viabilidade do projeto**

O Business Model Generation (CANVAS), é uma ferramenta de planejamento estratégico criado pelo Suíço Alexander Osterwalder. É uma metodologia flexível que permite uma linguagem simples e comum para descrever, avaliar, visualizar, e alterar os principais elementos que definem um modelo de negócio novo ou existente. É apresentado como uma tabela ilustrada dividida em nove componentes: Segmentos de Clientes; Proposta de Valor; Canais; Relações com Clientes; Receitas; Recursos Chaves; Atividades Chaves; Parcerias; e Custos.

A proposta de divisão em componentes ou blocos melhora a percepção e a compreensão de cada um destes módulos por meio da utilização de imagens, descrevendo a lógica de criação, entrega e captura de valor por parte de uma organização, ou seja, como uma empresa cria, proporciona e gera valor.

A Figura 1 apresenta o modelo CANVAS criado no estudo de viabilização do projeto.

Figura 1: CANVAS



Fonte: autor

O CANVAS define a proposta de valor que representa os produtos e serviços que criam benefício para um segmento de clientes que esclarece as características adequadas às exigências específicas de um nicho de mercado.

### **3 Levantamento de Requisitos**

Nesta sessão serão apresentados os Requisitos Funcionais e Não-Funcionais, os Fluxos do Sistema, e os Diagramas UML representando as funcionalidades e como deve se comportar o sistema.

Segundo Guedes (2009) o Levantamento de Requisitos é a primeira fase de um processo de desenvolvimento de software. Nesta etapa o engenheiro de software compreende as necessidades do usuário e o que sistema a ser desenvolvido deve resolver.

#### **3.1 Especificação dos Requisitos**

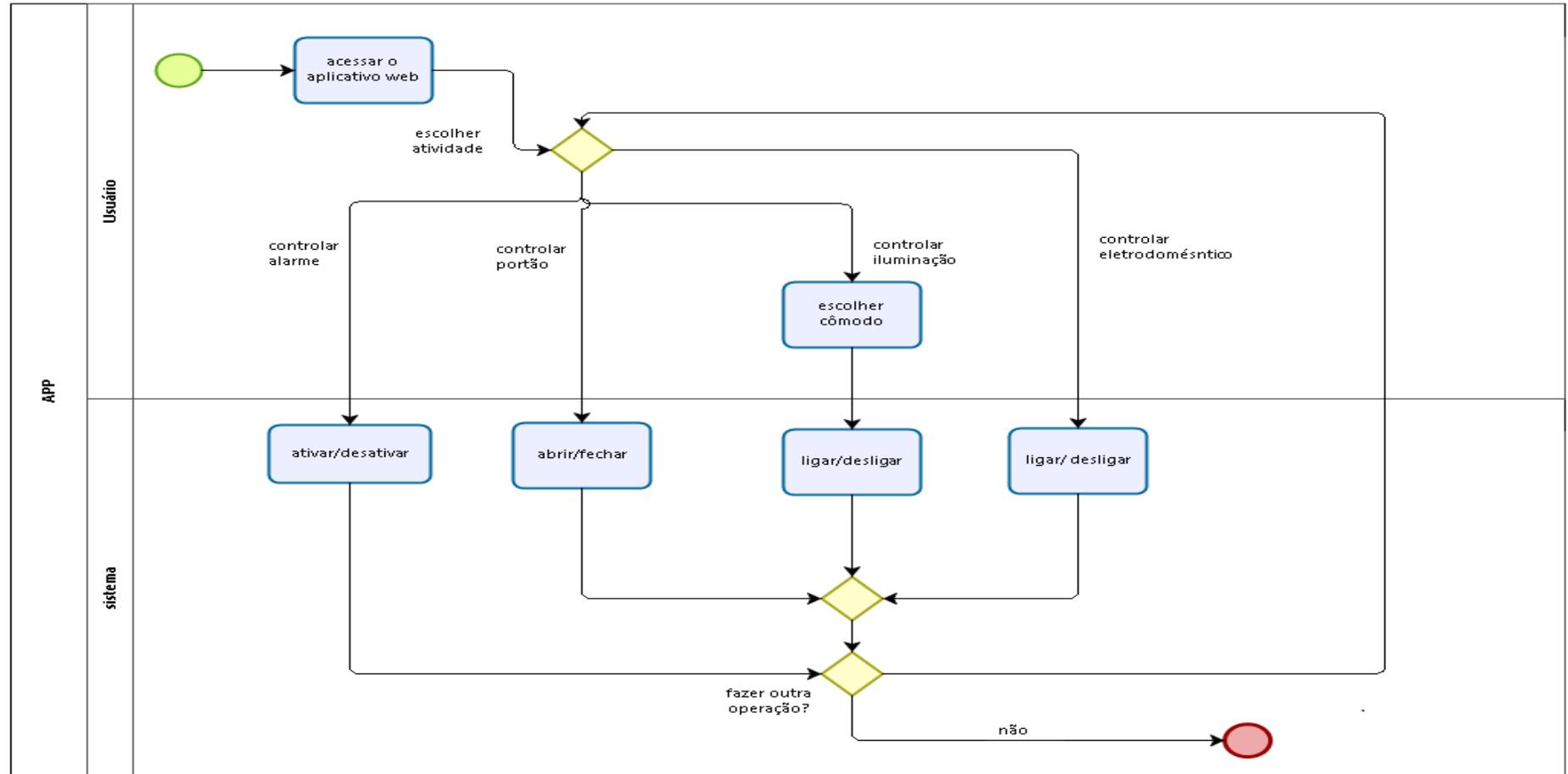
O sistema é composto de hardware e software embarcado para controle e acionamento de dispositivos de uma residência.

##### **3.1.1 BPMN**

O *Business Process Model and Notation* (BPMN) define um *Business Process Diagram* (BPD), que é uma representação como fluxograma para criar modelos gráficos de fácil interpretação dos processos de negócios. Um modelo de processo de negócios então, é uma rede de objetos gráficos, que descrevem o fluxo das atividades que possibilitam a execução e o controle de processos e de seus desempenhos dentro de uma organização (BIZAGE, 2018).

A Figura 2 exhibe o BPMN do projeto.

Figura 2: BPMN



Fonte: autor

### 3.1.2 Requisitos Funcionais

Os requisitos funcionais são aqueles que fazem parte do sistema. De acordo com Guedes (2009) os que correspondem ao que o cliente quer que o sistema realize ou seja, as funcionalidades do software. Estes são requisitos que definem o comportamento do sistema e que são visíveis pelos usuários.

Os requisitos funcionais do sistema projetado são exibidos no Quadro 1.

**Quadro 1:** Requisitos funcionais do sistema

<b>RF 01 – Iluminação</b>	Categoria: (X) Evidente	Prioridade: (X) Alta
<b>Descrição:</b> O usuário tem a opção de escolher esta funcionalidade para acender e/ou apagar luz.		
<b>RF 02 – Comodo</b>	Categoria: (X) Evidente	Prioridade: (X) Alta
<b>Descrição:</b> O usuário escolhera qual o cômodo que será usado para o controle de iluminação.		
<b>RF 03 – Portão</b>	Categoria: (X) Evidente	Prioridade: (X) Alta
<b>Descrição:</b> O usuário tem a opção de escolher se o portão será aberto ou fechado.		
<b>RF 04 – Alarme</b>	Categoria: (X) Evidente	Prioridade: (X) Alta
<b>Descrição:</b> O usuário tem a opção de escolher se o alarme será acionado ou não.		
<b>RF 05– Controlar Eletrodoméstico</b>	Categoria: (X) Evidente	Prioridade: (X) Alta
<b>Descrição:</b> O usuário tem a opção de escolher a funcionalidade de ligar ou desliga-lo.		

**Fonte:** autor

### 3.1.3 Requisitos Não Funcionais

Segundo Guedes (2009) Requisitos Não-funcionais condizem às restrições, condições, consistências, e validações que apresentam efeito sobre os Requisitos Funcionais. Encontram-se diversos tipos de requisitos não-funcionais, como usabilidade, desempenho, confiabilidade, segurança e interface.

Os Requisitos Não-funcionais do sistema estão descritos no Quadro 2.

**Quadro 2:** Requisitos não funcionais do sistema

Requisitos não funcionais			
Nome	Categoria	Obrigatoriedade	Permanência
RNF 1.1 Aplicativo Web	Interface	Obrigatório	Permanente
Restrição: O sistema deverá ser desenvolvido para web.			
RNF 1.2 Wifi	Interface	Obrigatório	Permanente
Restrição: Os comandos para as funcionalidades do sistema devem ser acionados por via wifi.			

**Fonte:** autor

### 3.1.4 Regras de Negócio

O Quadro 3 apresenta as regras de negócio do projeto.

**Quadro 3:** Regras de negócio

<b>RN 01 – Escolher Comodo</b>
<b>Descrição:</b> Usuário só poderá ter controle da iluminação se houver escolhido cômodo.
<b>RN 02 – Acesso a internet</b>
<b>Descrição:</b> Usuário só poderá controlar o aplicativo se estiver conectado à internet

**Fonte:** autor

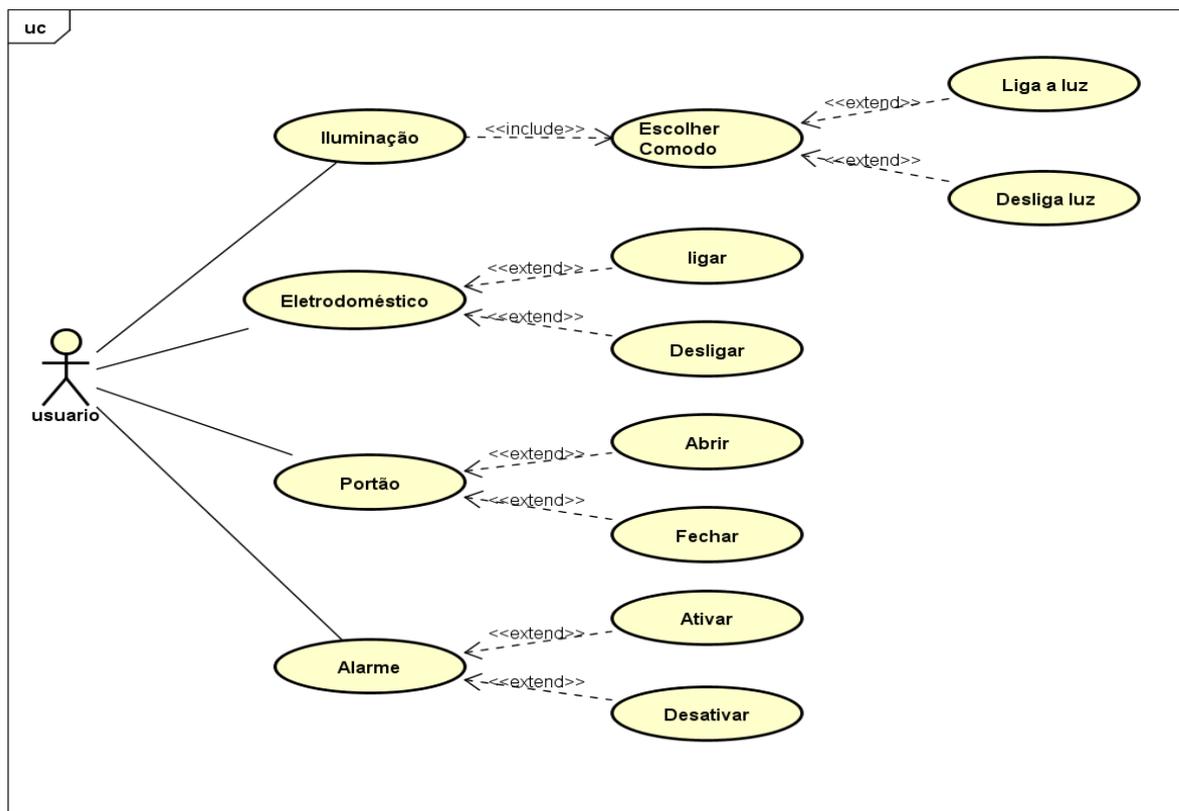
### 3.1.5 Casos de Uso

São apresentados os Casos de Uso do sistema e como se comportarão as funcionalidades de automação residencial.

O Diagrama de Casos de Uso (DCU) é um diagrama da Unified Modeling Language (UML), geralmente utilizado nas fases de Levantamento e Análise de Requisitos do sistema. Ele normalmente é consultado durante todo o processo de modelagem e serve de apoio para outros diagramas. O DCU retrata numa linguagem simples e de fácil compreensão as funcionalidades do sistema e passa uma visão geral de como o sistema irá se comportar. São representados os Atores demonstrando os papéis desempenhados pelos diversos usuários do aplicativo, os serviços e as funções do sistema (GUEDES,2009).

Na figura 3 é apresentado o caso de uso do projeto desenvolvido.

Figura 3: Diagrama Caso de Uso



powered by Astah

Fonte: autor

Os Quadros 4 a 16 apresentam as descrições dos casos de uso do projeto.

Quadro 4: Especificação do Use Case 001

Caso de Uso – Controlar Iluminação	
<b>ID</b>	UC 001
<b>Descrição</b>	Este caso de uso tem por objetivo dar a opção de o usuário controlar a iluminação do cômodo desejado.
<b>Ator Primário</b>	Usuário
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case iniciar quando o usuário deseja controlar a iluminação.</li> <li>2. Inicia UC002.</li> <li>3. Escolhe se a lâmpada será ligada ou desligada.</li> </ol>
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	UC002
<b>Extensão</b>	UC003 UC004

Fonte: autor

**Quadro 5:** Especificação do Use Case 002

<b>Caso de Uso – Escolher Comodo</b>	
<b>ID</b>	UC 002
<b>Descrição</b>	Este caso tem como finalidade escolher qual cômodo a iluminação será controlada
<b>Ator Primário</b>	Usuário
<b>Pré-condição</b>	UC 001
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário tem que escolher o cômodo.</li> <li>2. O usuário escolhe o cômodo.</li> <li>3. O sistema encerra a operação</li> </ol>
<b>Pós-condição</b>	UC003
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	UC002
<b>Extensão</b>	UC 003 UC 004

**Fonte:** autor

**Quadro 6:** Especificação do Use Case 003

<b>Caso de Uso – Acender Lâmpada</b>	
<b>ID</b>	UC 003
<b>Descrição</b>	Este caso de uso tem por objetivo acender a lâmpada.
<b>Ator Primário</b>	Usuário.
<b>Pré-condição</b>	UC 001 UC 002
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário desejar acender a lâmpada.</li> <li>2. O usuário acende a lâmpada.</li> <li>3. O sistema encerra a operação.</li> </ol>
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	Não possui

**Fonte:** autor

**Quadro 7:** Especificação do Use Case 004

<b>Caso de Uso – Apagar Lâmpada</b>	
<b>ID</b>	UC 004
<b>Descrição</b>	Este caso de uso tem por objetivo apagar a lâmpada.
<b>Ator Primário</b>	Usuário.
<b>Pré-condição</b>	UC 001 UC 002
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário desejar apagar a lâmpada.</li> <li>2. O usuário apaga a lâmpada.</li> <li>3. O sistema encerra a operação.</li> </ol>
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	Não possui

**Fonte:** autor

**Quadro 8:** Especificação do Use Case 005

<b>Caso de Uso – Controlar portão</b>	
<b>ID</b>	UC 005
<b>Descrição</b>	Este caso de uso inicia quando o usuário escolhe a opção de controlar portão
<b>Ator Primário</b>	Usuário.
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário escolhe a opção de controlar o portão.</li> <li>2. O sistema encerra a operação após a escolha.</li> </ol>
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	UC 006 UC 007

**Fonte:** autor

**Quadro 9:** Especificação do Use Case 006

<b>Caso de Uso – Abrir Portão</b>	
<b>ID</b>	UC 006
<b>Descrição</b>	Este caso o inicia quando o usuário escolhe a opção de abrir o portão.
<b>Ator Primário</b>	Usuário.
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário desejar abrir o portão.</li> <li>2. O usuário abre o portão.</li> <li>3. O sistema encerra a operação.</li> </ol>
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	Não possui

**Fonte:** autor

**Quadro 10:** Especificação do Use Case 007

<b>Caso de Uso –Fechar Portão</b>	
<b>ID</b>	UC 007
<b>Descrição</b>	Este caso de uso o usuário escolhe a opção fechar o portão
<b>Ator Primário</b>	Usuário.
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário desejar fechar o portão.</li> <li>2. O usuário fecha o portão.</li> <li>3. O sistema encerra a operação.</li> </ol>
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	Não possui

**Fonte:** autor

**Quadro 11:** Especificação do Use Case 008

<b>Caso de Uso –Controlar Eletrodoméstico</b>	
<b>ID</b>	UC 008
<b>Descrição</b>	Este caso de uso o usuário escolhe a opção ligar ou desligar o eletrodoméstico
<b>Ator Primário</b>	Usuário.
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	1. O use case inicia quando o usuário desejar controlar eletrodoméstico. 2. O sistema encerra a operação.
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	UC009 UC010

**Fonte:** autor

**Quadro 12:** Especificação do Use Case 009

<b>Caso de Uso –Ligar Eletrodoméstico</b>	
<b>ID</b>	UC 009
<b>Descrição</b>	Este caso de uso o usuário escolhe a opção ligar o eletrodoméstico.
<b>Ator Primário</b>	Usuário.
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	1. O use case inicia quando o usuário desejar ligar o a eletrodoméstico. 2. O usuário liga o eletrodoméstico. 3. O sistema encerra a operação.
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	Não possui

**Fonte:** autor

**Quadro 13:** Especificação do Use Case 010

<b>Caso de Uso –Desligar Eletrodoméstico</b>	
<b>ID</b>	UC 010
<b>Descrição</b>	Este caso de uso o usuário escolhe a opção de desligar o eletrodoméstico. .
<b>Ator Primário</b>	Usuário.
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	1. O use case inicia quando o usuário desejar desligar o eletrodoméstico. 2. O usuário desliga o eletrodoméstico. 3. O sistema encerra a operação.
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	Não possui

**Fonte:** autor

**Quadro 14:** Especificação do Use Case 011

<b>Caso de Uso – Controlar Alarme</b>	
<b>ID</b>	UC 011
<b>Descrição</b>	Este caso de uso tem por objetivo dar a opção de o usuário controlar o alarme.
<b>Ator Primário</b>	Usuário
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	1. O use case iniciar quando o usuário deseja controlar o alarme. 2. Escolhe se o alarme será ligado ou desligado.
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	UC012 UC013

**Fonte:** autor

**Quadro 15:** Especificação do Use Case 012

<b>Caso de Uso –Ativar Alarme</b>	
<b>ID</b>	UC 012
<b>Descrição</b>	Este caso de uso tem por objetivo dar a opção de o usuário ativar o alarme.
<b>Ator Primário</b>	Usuário
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case iniciar quando o usuário deseja ativar alarme.</li> <li>2. O usuário ativa o alarme.</li> <li>3. O sistema encerra a operação.</li> </ol>
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	Não possui

**Fonte:** autor

**Quadro 16:** Especificação do Use Case 013

<b>Caso de Uso –Desativar Alarme</b>	
<b>ID</b>	UC 013
<b>Descrição</b>	Este caso de uso tem por objetivo dar a opção de o usuário desativar o alarme.
<b>Ator Primário</b>	Usuário
<b>Pré-condição</b>	Não possui
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case iniciar quando o usuário deseja desativar o alarme.</li> <li>2. O usuário desativa o alarme.</li> <li>3. O sistema encerra a operação.</li> </ol>
<b>Pós-condição</b>	Não possui
<b>Cenário Alternativo</b>	*a – Em qualquer momento o usuário pode sair do sistema
<b>Inclusão</b>	Não possui
<b>Extensão</b>	Não possui

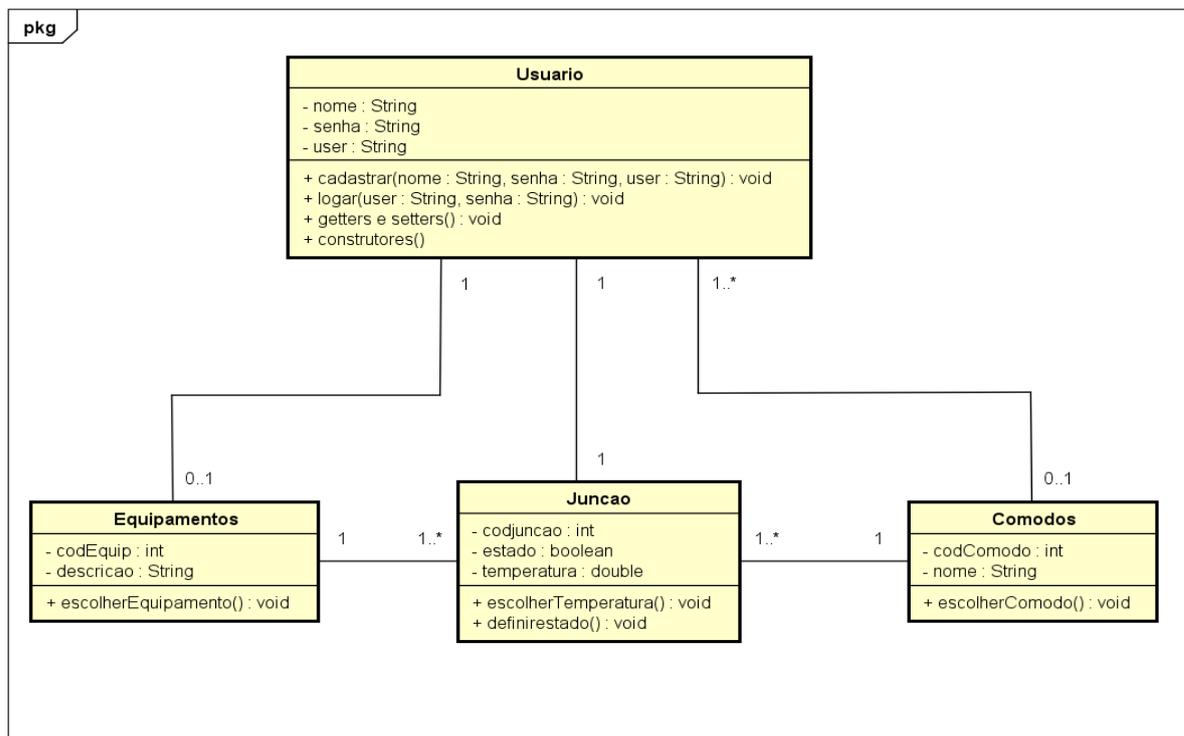
**Fonte:** autor

### 3.1.6 Diagrama de Classes

O Diagrama de Classes (DC) é o mais utilizado e o mais importante da UML. Serve de apoio para a maioria dos demais diagramas. Como o próprio nome diz,

define a estrutura das classes utilizadas pelo sistema, determinando os atributos e métodos que cada classe possui, além de estabelecer como as classes se relacionam e trocam informações (GUEDES, 2009). A Figura 4 apresenta o DC do projeto.

**Figura 4:** Diagrama de Classe



powered by Astah

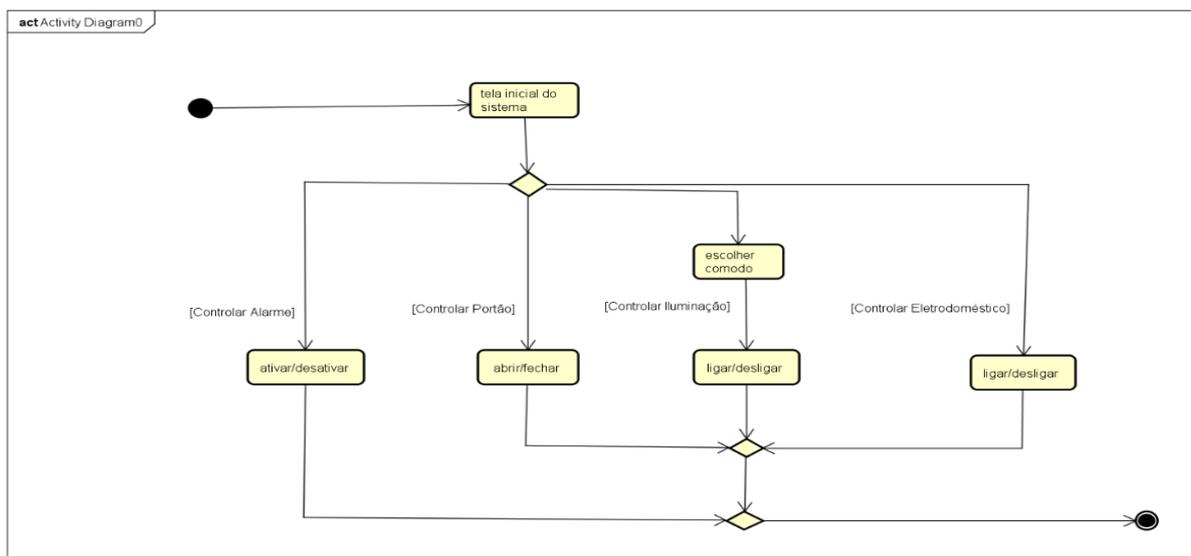
Fonte: autor

### 3.1.7 Diagrama de Atividades

O Diagrama de Atividades (DA) tem como principal foco, enfatizar a sequência e as condições de um processo, sendo o diagrama que mais se parece com o algoritmo de UML, sendo assim um dos mais detalhistas de um processo (GUEDES, 2009).

A Figura 5 exibe o DA gerado para o desenvolvimento da aplicação.

**Figura 5:** Diagrama de Atividade



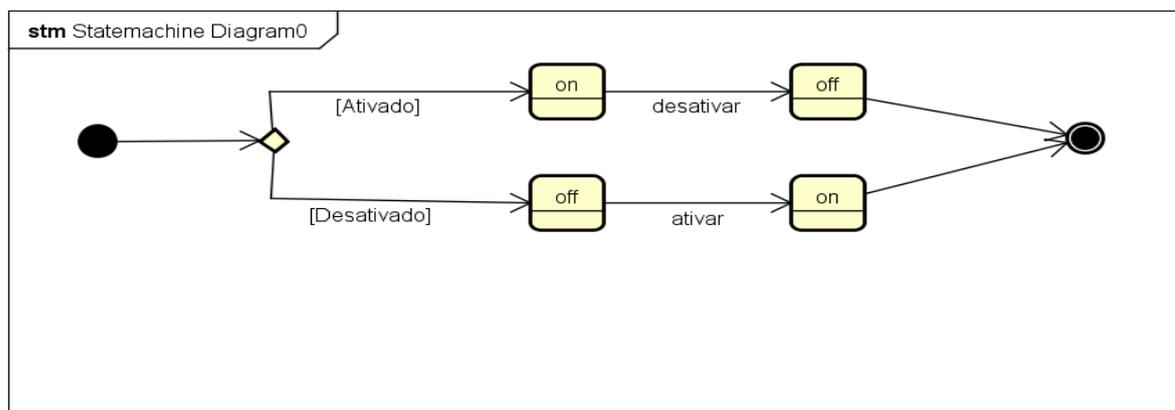
Fonte: autor

### 3.1.8 Diagrama de Estados

O Diagrama de Máquina de Estados (DME) é utilizado normalmente para acompanhar os estados por que passa uma instância de uma classe, mas pode ser utilizado para representar os estados de um caso de uso ou mesmo os estados gerais de um sistema completo (GUEDES, 2009).

A Figura 6 exibe o diagrama de estado gerado para o desenvolvimento da aplicação.

**Figura 6:** Diagrama de Atividade



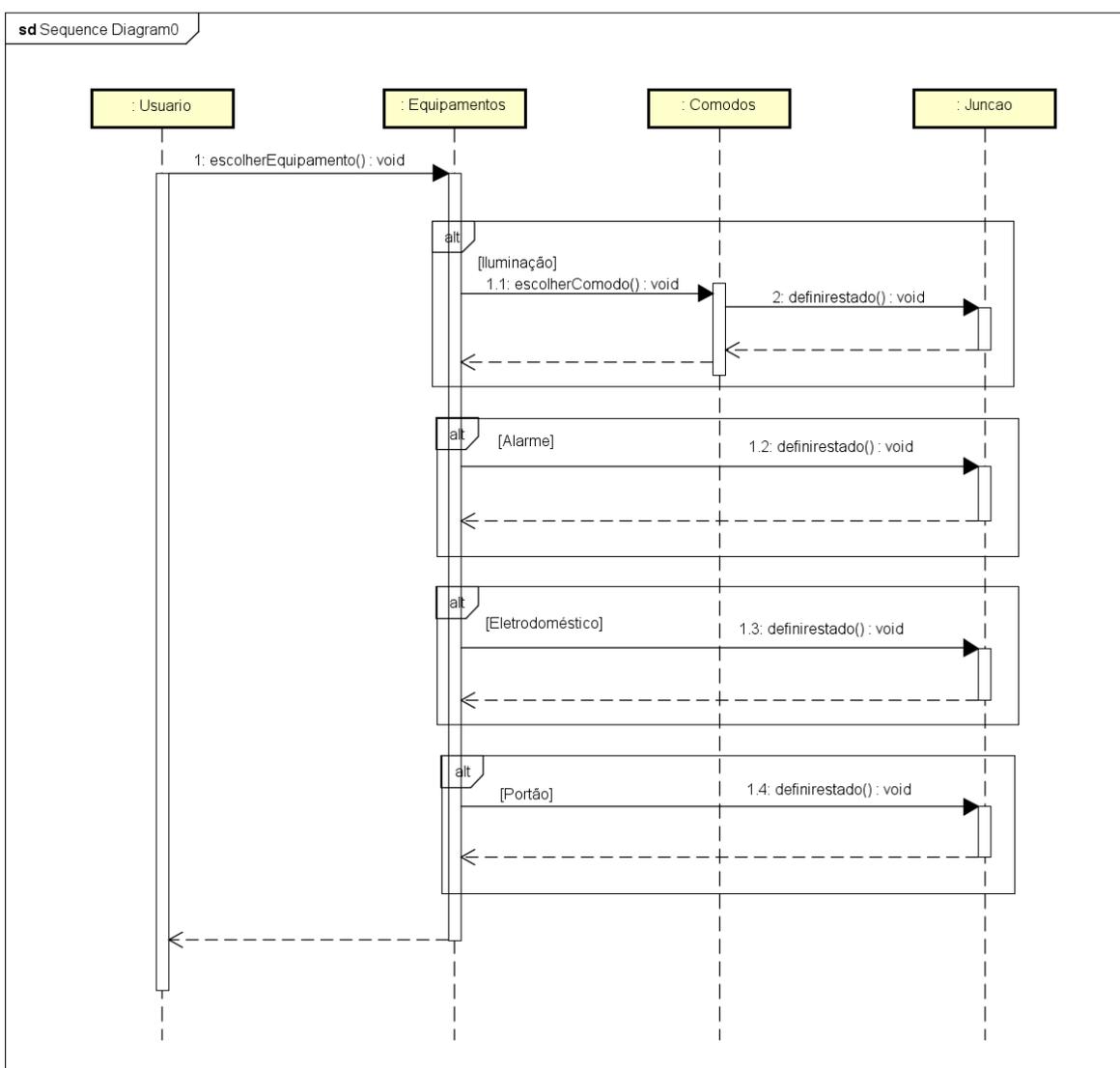
Fonte: autor

### 3.1.9 Diagrama de Sequência

O Diagrama de Sequência (DS) preocupa-se com a ordem temporal em que as mensagens são trocadas entre os objetos envolvidos em um determinado processo. Um DS costuma identificar o evento gerador do processo modelado, bem como o ator responsável por esse evento, e determina como o processo deve se desenrolar e ser concluído por meio da chamada de métodos disparados por mensagens enviadas entre os objetos (GUEDES, 2009).

A Figura 7 exibe o DS gerado para o desenvolvimento da aplicação.

**Figura 7:** Diagrama de Sequência



## 4 Materiais e Métodos ou Desenvolvimento

A seleção dos materiais utilizados no projeto é importante em todos os aspectos pois determina a confiabilidade em termos dos aspectos industriais e econômicos.

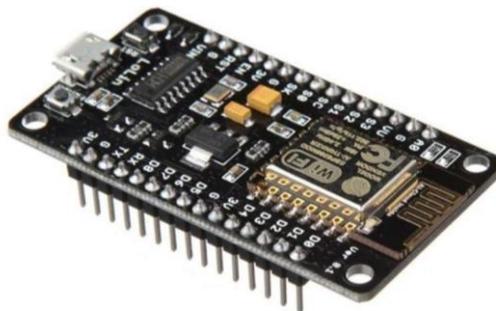
Esta sessão apresenta os materiais que foram utilizados para a construção do protótipo.

### 4.1 NodeMcu

O módulo Wifi ESP8266 NodeMCU é um microcontrolador de desenvolvimento que combina o chip ESP8266 para conexão *wifi*, uma interface *usb* e um regulador de tensão 3.3V. A programação pode ser feita usando LUA ou a IDE do Arduino, utilizando a comunicação via cabo *usb*.

A Figura 8 exibe uma imagem da placa utilizada no projeto.

**Figura 8** – Placa NodeMcu



**Fonte:** Castro (sd, *online*)

### 4.2 Led

Um LED (*Light-Emitting Diode* em português Diodo Emissor de Luz) é um dispositivo semiconductor que emite luz em resposta à passagem de corrente elétrica.

A Figura 9 mostra uma imagem do led utilizado no projeto.

**Figura 9** – Figura do Led

**Fonte:** Castro (sd, *online*)

### 4.3 Sensor Pir

O Sensor de Presença e Movimento PIR DYP-ME003 consegue detectar a presença em uma área de até 7 metros. Caso algo ou alguém se movimentar nessa área a tensão no pino de sinal é ativada.

O módulo contém internamente duas faixas com material sensível ao infravermelho e uma lente Fresnel, que quando há variação na detecção do sinal infravermelho entre essas duas faixas de material sensível, a tensão no pino de saída é acionada por um determinado tempo. A lente Fresnel tem a função de ampliar o campo de visão do sensor condensando a luz em um único ponto.

A Figura 10 apresenta a imagem do sensor PIR usado no projeto.

**Figura 10** – Figura do Sensor Pir

**Fonte:** Baú&Cia (2014)

#### 4.4 Modulo Relé

O Módulo Relé 5V permite uma integração com microcontroladores. A partir das entradas digitais pode-se controlar cargas maiores e dispositivos como motores AC ou DC, eletroímãs, solenoides e lâmpadas incandescentes.

A Figura 11 exibe a imagem do Módulo Relé utilizado no projeto.

**Figura 11** – Figura do Relé



**Fonte:** Eletrogate (sd *online*)

#### 4.5 Servo Motor

Micro Servo 9g SG90 TowerPro – é um motor que controla a posição angular de 0° a 180°. e acompanha um cabo de 3 pinos referente a alimentação/controle no projeto foi utilizado para o controle do acionamento para o portão eletrônico.

A Figura 12 apresenta a imagem do micro servo usado no projeto.

**Figura 12** – Figura do Relé



**Fonte:** Castro (sd, *online*)

## 5 Métodos

O desenvolvimento do protótipo foi elaborado a partir do diagrama de Modelagem de Processos de Negócio que foi criado utilizando-se a plataforma Bizagi Modeler.

Do Levantamento de Requisitos foram desenvolvidos os diagramas UML utilizando-se a ferramenta Astah Community.

Os circuitos eletrônicos e a lógica de controle foram testados no simulador Tinkercad, uma plataforma *online* de acesso gratuito disponibilizada pela AutoDesk.

A prototipação da interface do aplicativo *mobile* fez-se através da plataforma web Moqups.

A ferramenta escolhida para a implementação da lógica do sistema foi a IDE Arduino para a codificação do microcontrolador, e para a aplicação web que permite o controle das funções dos componentes do protótipo, a codificação foi desenvolvida com linguagem de marcação de texto HTML e CSS para estilização pensado em uma interface fácil e amigável, e o JavaScript para a interação dos usuários e o sistema.

### 5.1 Bizagi Modeler BPMN

Bizagi Modeler é ferramenta usada para desenvolver o diagrama BPMN, uma modelagem que analisa e ilustra os processos de negócio. É uma ferramenta simples com funções de arrastar e soltar. A modelagem de processos de negócios é um modelo gráfico que relata as etapas de um processo de negócios, esse modelo foi projetado para organizar a sequência dos processos, e possibilita a modelagem dos fluxos de trabalho, a documentação tem várias formas de visualização, como pdf, word, excel, e ainda a exportação do modelo. Através desta ferramenta é possível uma visão dos processos para o desenvolvimento do projeto de Automação Residencial.

A versão usada para o desenvolvimento deste projeto foi o Bizage Modeler-3.3.11.0, e a empresa de desenvolvimento da ferramenta é Bizage Limited.

### 5.2 Astah Community Diagramas da UML

Astah Community é uma ferramenta dinâmica usada para desenvolver diagramas. Ela é voltada para a modelagem de sistemas utilizando a UML. Na área de Engenharia de Software, a UML é uma linguagem de modelagem que permite representar um sistema de forma padronizada.

A ferramenta é simples na construção dos diagramas, geralmente apenas clicando, arrastando e nomeando. A ferramenta apresenta nomes dos diagramas de acordo com a UML e os diagramas podem ser exportados nos formatos PNG e JPEG.

Esta ferramenta foi utilizada no projeto de Automação Residencial para o desenvolvimento dos diagramas da UML:

Diagrama de Caso de Uso

Diagrama de Classe

Diagrama de Atividade

Diagrama de Máquina de Estado

Diagrama de Sequência

A versão usada para o desenvolvimento desse projeto foi Astah Community 7\_0\_0-846701 64bit. A empresa de desenvolvimento da ferramenta é Change Vision que descontinuou suas atualizações em 2018.

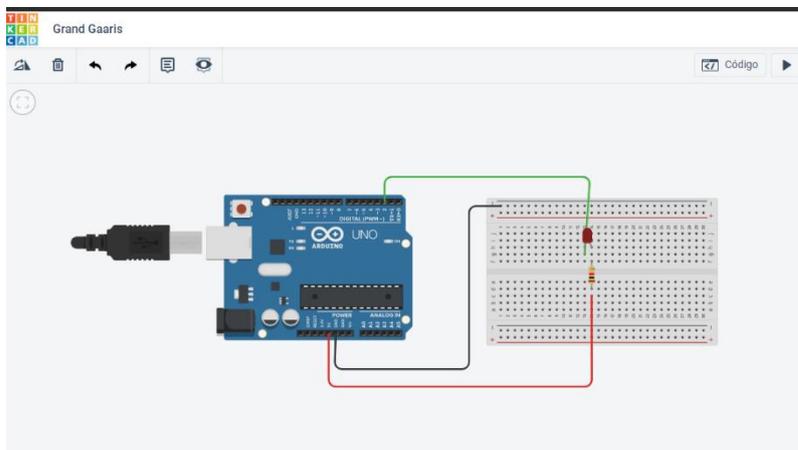
### 5.3 Tinkercad Simulador de Circuitos

O Tinkercad é uma ferramenta *online* de *design* de modelos 3D em CAD e de simulação de circuitos elétricos analógicos e digitais. Esta ferramenta é simples de utilizar e de acesso gratuito. Por isso é também uma forma de aprendizagem para programação de microcontroladores. Esta ferramenta foi de muita importância nesse projeto de Automação Residencial, pois ela permite a simulação de um projeto real, o que possibilitou o aprendizado, a implementação e os testes do hardware e do software componentes do projeto.

Testou-se nesta ferramenta o uso de componentes como leds, resistores, motor, sensor e o microcontrolador.

A figura 13 apresenta a simulação de como se comporta uma lâmpada em um ambiente real controlada por um microcontrolador.

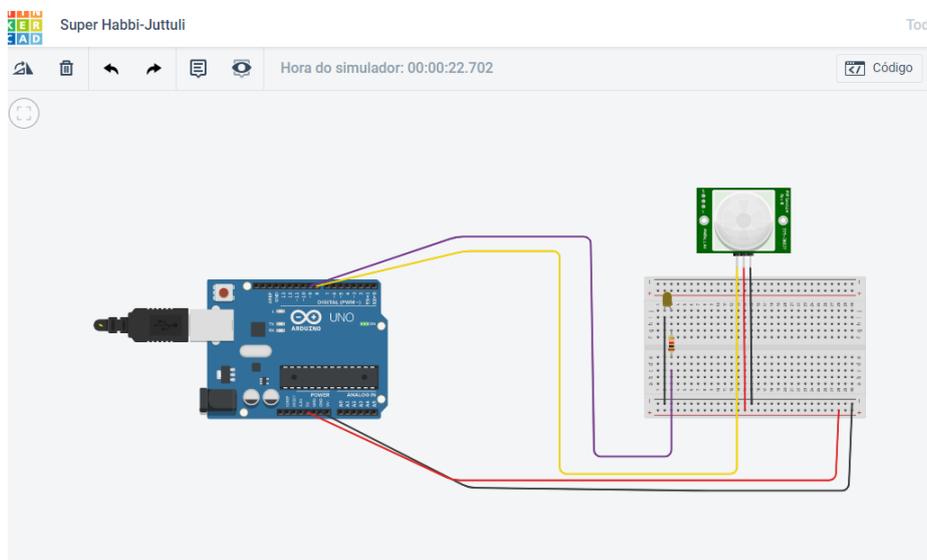
**Figura 13 - Simulação Led**



Fonte: autor

Segue a Figura 14 apresenta a simulação de como se comporta um sensor de presença. Neste projeto o PIR teve a funcionalidade de quando acionado por presença, acender o led.

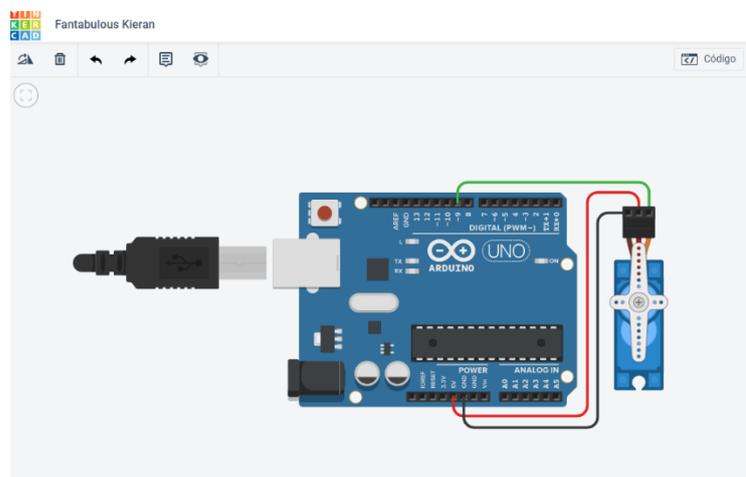
**Figura 14 - Simulação Sensor de Presença**



Fonte: autor

A Figura 15 mostra como funciona o acionamento do motor servo para simular o controle do portão da garagem.

**Figura 15 - Simulação do Motor**



**Fonte:** autor

#### 5.4 Moqups

Para prototipação de Telas usou-se o Moqups, que é um aplicativo HTML5 usado para criar *wireframes*, podendo ser definido como um esqueleto, ou uma versão primitiva da aparência da interface do aplicativo. Esta ferramenta foi utilizada no projeto de Automação Residencial para o projeto das telas da aplicação, sendo possível a visualização de como a aplicação se apresenta ao usuário. A interface foi criada de forma que apresentasse facilidade e praticidade em seu uso. A ferramenta está disponível atualmente através do site <https://moqups.com>.

## 6 DESENVOLVIMENTO

O projeto iniciou-se pelo desenvolvimento dos códigos e comandos para conexão na rede wifi da residência, conforme a Figura 16 mostra a biblioteca que deve ser incluída no *sketch* para reconhecimento dos componentes e como deve ser realizada a inserção dos dados de autenticação na rede sem fio.

**Figura 16 – Código de autenticação wifi e biblioteca**

```
AUTOMA__O_RESIDENCIAL$
1 #include <ESP8266WiFi.h>
2 #include <Servo.h>
3
4 const char* ssid = "SSID";
5 const char* password = "PASSWORD";
6 WiFiServer server(80);
7
```

Fonte: autor

Deve se realizar a configuração dos pinos, e da comunicação serial para depuração das operações, bem como implementar os códigos que realizam a conexão à rede wifi, e a lógica que inicia o servidor web conforme mostrado na Figura 17.

Figura 17 – Código de conexão dos pinos, wifi e servidor

```
AUTOMA__O_RESIDENCIAL$
10 void setup() {
11   delay(1000);
12   pinMode(13, OUTPUT);
13   pinMode(5, OUTPUT);
14   Serial.begin(9600);
15
16   Serial.println();
17   Serial.println();
18   Serial.print("Conectando com: ");
19   Serial.println(ssid);
20   WiFi.begin(ssid, password);
21   while (WiFi.status() != WL_CONNECTED) {
22     delay(500);
23     Serial.print(".");
24   }
25   Serial.println("");
26   Serial.println("WiFi conectado com sucesso");
27
28   server.begin();
29   Serial.print("Servidor startado em ");
30   Serial.print("http://");
31   Serial.print(WiFi.localIP());
32   Serial.println("/");
33 }
34
```

Fonte: autor

Nesta etapa são definidos os conjuntos de instruções que o programa percorre e repete em um laço até que sejam alcançadas as condições desejadas que são as requisições do *client* para a conexão com o wifi. Tendo o servidor retornado a confirmação de recebimento da requisição através de uma mensagem, as condições de acionamento do botão do Led (linhas 49 à 51 da Figura 18), do botão do

eletrodoméstico (linhas 52 à 54), e do motor do portão eletrônico (linhas 55 à 57) são tratadas para realizar as operações desejadas, determinados pelo acionamento dos componentes da interface do aplicativo.

**Figura 18** – Código de instruções de respostas e repetição

```
AUTOMA__O_RESIDENCIAL$
36 void loop() {
37   WiFiClient client = server.available();
38   if (!client) {
39     return;
40   }
41   Serial.print("Mensagem do cliente: ");
42   while(!client.available()) {
43     delay(1);
44   }
45   String request = client.readStringUntil('\r');
46   Serial.println(request);
47   client.flush();
48
49   if (request.indexOf("ONOFFL1") != -1) {
50     digitalWrite(13, !digitalRead(13));
51   }
52   if (request.indexOf("ONOFFL2") != -1) {
53     digitalWrite(5, !digitalRead(5));
54   }
55   if (request.indexOf("Motor") != -1) {
56     motorligar();
57   }
}
```

**Fonte:** autor

A Figura 19 apresenta a montagem e conexão dos circuitos na maquete.

**Figura 19** – Montagem dos circuitos na maquete



**Fonte:** autor

A Figura 20 mostra a declaração da variável para a funcionalidade do motor, e a Figura 21 exibe a função de rotação do motor eletrônico quando esta é executada através do botão.

**Figura 20 – Declaração de variável**

```

7 |
8 | Servo motor;
9 | int angulo = 0;
10 |

```

Fonte: autor

**Figura 21 – Código de rotação do motor**

```

AUTOMA__O_RESIDENCIAL $
69 void motorligar() {
70     motor.attach(14);
71     if(angulo == 0) {
72         for(angulo; angulo<=180; angulo++)
73             {
74                 motor.write(angulo);
75                 delay(10);
76             }
77     } else {
78         for(angulo; angulo=1; angulo--)
79             {
80                 motor.write(angulo);
81                 delay(10);
82             }
83     }
84 }

```

Fonte: autor

A Figura 22 demonstra o código da conexão com a função entre a placa e a aplicação.

**Figura 22 – Código da aplicação web**

```

86 function OnOffI(id) {
87
88     var xmlhttp = new XMLHttpRequest();
89     xmlhttp.open("GET", "http://ssid/LED=ONOFFL1");
90     xmlhttp.send();
91
92     var IL=""IL"", ID=""ID"";
93     if (id == "IL"){
94         document.getElementById('BI').className='botaoludesl';
95         var I =
96         '<div class="iluimg"></img>'+
97         '<button type="button" id="BI" class="botaoludesl" onclick="OnOffI('+ID+'");"> Apagar </button></div>';
98     }else if (id == "ID"){
99         var I =
100         '<div class="iluimg"></img>'+
101         '<button type="button" id="BI" class="botaoilu" onclick="OnOffI('+IL+'");"> Acender </button></div>';
102     }
103     document.getElementById("alarme").innerHTML = document.getElementById("alarme").innerHTML
104     document.getElementById("eletro").innerHTML = document.getElementById("eletro").innerHTML
105     document.getElementById("garagem").innerHTML = document.getElementById("garagem").innerHTML
106     document.getElementById("iluminacao").innerHTML = I
107 }

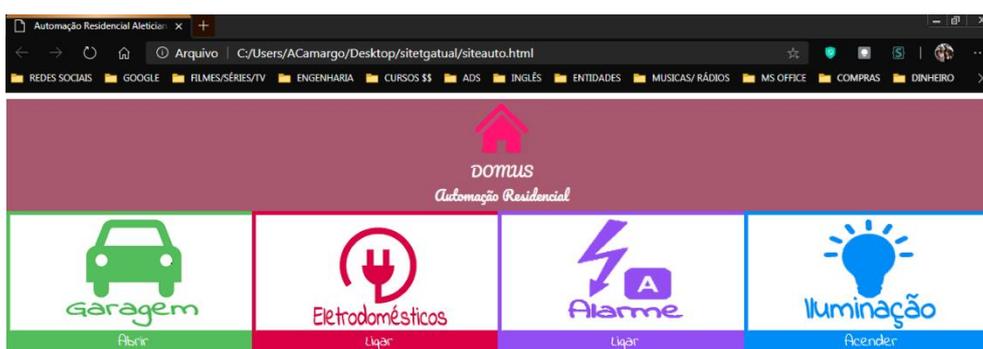
```

Fonte: autor

## 7 RESULTADOS E DISCUÇÃO

Como resultado obteve-se uma aplicação web responsiva denominada Domus Automação Residencial, que funciona de maneira prática acionando os botões através de conexão wifi e emite os comandos que serão interpretados e executados pelo controlador conforme mostrado nas figuras 23 e 24.

**Figura 23 – Aplicação web**



**Fonte:** autor

**Figura 24 – Aplicação mobile**



**Fonte:** autor

A automação residencial foi executada em maquete de placas de MDF onde foi realizada a montagem dos circuitos como mostrado na figura 25.

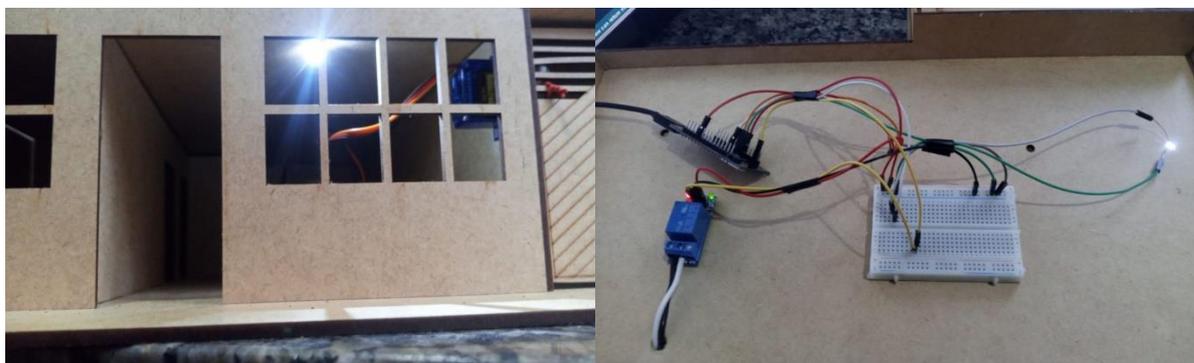
**Figura 25 – Montagem da maquete**



**Fonte:** autor

A Figura 26 demonstra o funcionamento do Led representando uma iluminação residencial.

**Figura 26 – Acionamento da iluminação**



**Fonte:** autor

A Figura 27 representa o funcionamento de um portão eletrônico movido por um servo motor.

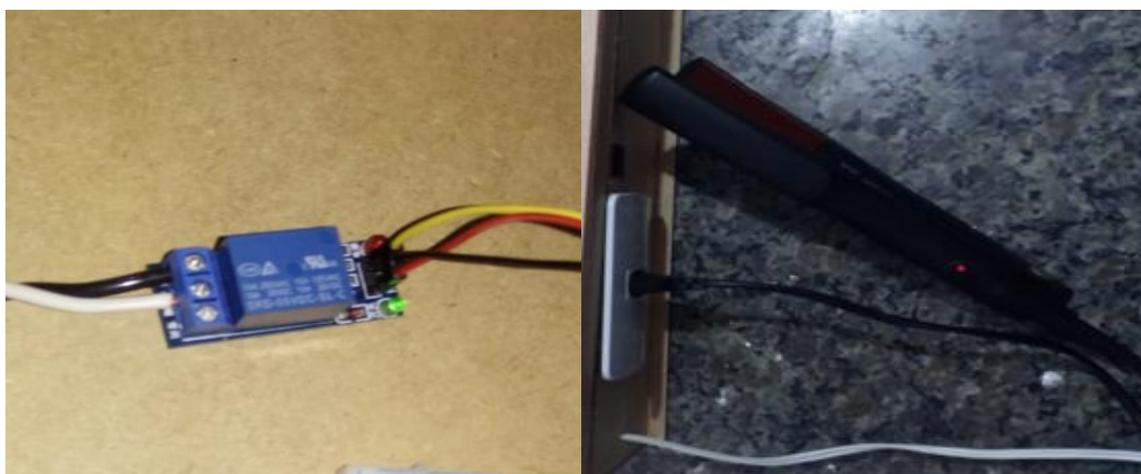
**Figura 27 – Acionamento do motor servo**



Fonte: autor

A Figura 28 representa o funcionamento de um eletrodoméstico acionado por um relé.

**Figura 28 – Acionamento do motor servo**



Fonte: autor

## 8 Considerações finais

Com os avanços tecnológicos a automação utiliza a informação através da combinação de sistemas digitais e meios de telecomunicação, assim este projeto estimula e viabiliza a implantação de sistema de automação em uma residência que permite ao usuário ligar e desligar aparelhos de determinado ambiente através do seu

*smartphone* ou computador, uma vez que podem ser controlados por meio de uma rede *wifi*.

O Projeto apresentado pode ser integrado com outros sensores como de umidade, temperatura, potenciômetro para regulação de intensidade luminosa, entre outros, resultando em uma forma de integração automatizada entre o morador e sua residência.

Durante o período de desenvolvimento o processo avançou conforme o planejado, porém com certas limitações não foi possível incluir no projeto a função de controle com o sensor de presença. Esta funcionalidade bem como outras percebidas durante o desenvolvimento do protótipo serão objeto de trabalhos futuros, como, por exemplo:

- Inserção de outros pontos de iluminação,
- Configurar a aplicação para vários usuários,
- Inserir uma forma para aprimorar a segurança do projeto,
- Inserção de sensor de umidade e temperatura,
- Potenciômetro para controle de velocidade dos equipamentos,
- Controlador display LCD para leitura,
- Sensor de distância ultrassônico hc-sr04,
- Sensor de fluxo de corrente não invasivo 100ª sct-o13.

O resultado da comunicação entre a maquete e a aplicação web foi satisfatória tendo em vista a funcionalidade e execução das funções estabelecidas.

## **AGRADECIMENTOS**

A Deus por proporcionar-me saúde, confiança, força e por ter colocado pessoas incríveis para fazerem parte de minha vida. A Fatec FRANCA, juntamente com seu corpo docente, direção e administração, pela oportunidade, confiança, apoio e proporcionar um ensino de extrema excelência. Ao meu orientador, Prof. Carlos Eduardo de França Roland, pelo apoio, confiança dedicação, paciência e amizade que me ajudou a crescer academicamente e pessoalmente sendo um exemplo e terá meu

eterno agradecimento. À minha família pela paciência, amor e incentivo nesta jornada. E a todos que direta e indiretamente fizeram parte da minha formação, o meu muito obrigado.

## REFERÊNCIAS

ARDUINO&CIA. **Como Usar um Sensor de Presença com Arduino**. 2014. Disponível em: <<https://www.arduinoocia.com.br/sensor-presenca-arduino-modulo-pir-dyp-me003/>>. Acesso em: 7 dez.2020.

BAÚ. **LED Difuso 5mm Vermelho**. sd online. Disponível em: <<https://www.baudaeletronica.com.br/led-difuso-5mm-vermelho.html>>. Acesso em 7 dez.2020b.

CASTRO. **Nodemcu Lua Wifi CH340 IoT ESP8266 Lolin Arduino**. sd online. Disponível em: <[https://www.eletronicacastro.com.br/componentes-eletronicos/16850-nodemcu-lua-wifi-ch340-iot-esp8266-lolin-arduino.html?search\\_query=esp8266&results=9](https://www.eletronicacastro.com.br/componentes-eletronicos/16850-nodemcu-lua-wifi-ch340-iot-esp8266-lolin-arduino.html?search_query=esp8266&results=9)>. Acesso em 7 dez.2020.

DOMINGUES, R. G. **A Domótica como Tendência na Habitação: aplicação em habilitações de interesse social com suporte aos idosos e incapacitados**. 2013. Disponível em: <<http://www.dissertacoes.poli.ufrj.br/dissertacoes/dissertpoli988.pdf>>. Acesso em 22 nov.2020.

ELETROGATE. **Módulo Relé 1 Canal 5V**. sd online. Disponível em: <<https://www.eletrogate.com/modulo-rele-1-canal-5v>>. Acesso em 8 dez.2020.

CASTRO. **Micro Servo Tower Pro sg90**. sd online. Disponível em: <[https://www.eletronicacastro.com.br/componentes-eletronicos/14035-arduino-micro-servo-tower-pro-sg90-000000140355.html?search\\_query=motor+servo&results=6](https://www.eletronicacastro.com.br/componentes-eletronicos/14035-arduino-micro-servo-tower-pro-sg90-000000140355.html?search_query=motor+servo&results=6)>. Acesso em 17 dez.2020.

BLOGDOJOSECINTRA. **Arduino & IoT – Nivel Avançado**.sd online. Disponível em:<<https://josecintra.com/blog/>>

## APENDICE

### APENDICE A – Código Automação Residencial da placa Nodemcu

```
#include <ESP8266WiFi.h>
#include <Servo.h>
const char* ssid = "ssid";
```

```

const char* password = "*****";
WiFiServer server(80);
Servo motor;
int angulo = 0;

void setup() {
  delay(1000);
  pinMode(13, OUTPUT);
  pinMode(5, OUTPUT);
  Serial.begin(9600);
  Serial.println();
  Serial.println();
  Serial.print("Conectando com: ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi conectado com sucesso");
  server.begin();
  Serial.print("Servidor startado em ");
  Serial.print("http://");
  Serial.print(WiFi.localIP());
  Serial.println("/");
}

void loop() {
  WiFiClient client = server.available();
  if (!client) {
    return;
  }
  Serial.print("Mensagem do cliente: ");
  while(!client.available()) {
    delay(1);
  }
  String request = client.readStringUntil('\r');
  Serial.println(request);
  client.flush();

  if (request.indexOf("ONOFFL1") != -1) {
    digitalWrite(13, !digitalRead(13));
  }
  if (request.indexOf("ONOFFL2") != -1) {
    digitalWrite(5, !digitalRead(5));
  }
  if (request.indexOf("Motor") != -1) {
    motorligar();
  }
  client.println("HTTP/1.1 200 OK");
  client.println("Content-Type: text/html");
  client.println("");
  client.println("<!DOCTYPE HTML>");
  client.println("<html>");
  client.println("<head><title AUTOMAÇÃO RESIDENCIAL</title></head>");
  client.println("<body>ESP8266 Web Server </body>");
  client.println("</html>");
  delay(1);
}

void motorligar(){

```

```

motor.attach(14);
if(angulo == 0) {
  for(angulo; angulo<=110; angulo++)
  {
    motor.write(angulo);
    delay(10);
  }
} else {
  for(angulo; angulo>=1; angulo--)
  {
    motor.write(angulo);
    delay(10);
  }
}
}
}

```

## APENDICE B – Código HTML da Automação Residencial

```

<!DOCTYPE html>
<html Lang="pt-br">
  <head>
    <title>Automação Residencial Aleticiana</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet" href="css.css">

    <script charset="UTF-
8" Language="javascript" type="text/javascript" src="index.js"></script>
  </head>
  <body >
    <div class="logo">

      <div class="logotexto">
        </img></br>
        DOMUS</br>Automação Residencial
      </div>
    </div>

    <div class="responsive">
      <section id="garagem" >
        <div class="garagemimg"></img>
        <button type="button" id="BG" class="botaogaragem" onclick="On
OffG('GL')"> Abrir </button></div>

      </section>
    </div>
    <div class="responsive">
      <section id="eletro" >

        <div class="eletroimg"></img>

```

```

        <button type="button" id="BE" class="botaoeletro" onclick="OnO
ffe('EL');"> Ligar </button></div>

    </section>
</div>
<div class="responsive">
    <section id="alarme">
        <div class="alarmeimg"></img>
        <button type="button" id="BA" class="botaoalarme" onclick="OnO
ffa('AL');"> Ligar </button></div>

    </section>
</div>
<div class="responsive">
    <section id="iluminacao">
        <div class="iluimg"></img>
        <button type="button" id="BI" class="botaoilu" onclick="OnOffI
('IL');"> Acender </button></div>

    </section>
</div>

</body>
</html>

```

## APENDICE C – Código Javascript da Automação Residencial

```

function OnOffG(id) {
    var xmlhttp = new XMLHttpRequest();
    xmlhttp.open("GET", "http://192.168.43.120/LED=Motor");
    xmlhttp.send();

    var GL="'GL'", GD="'GD'";
    if (id == "GL"){
        var G =
        '<div class="garagemimg"></img>'+
        '<button type="button" id="BG" class="botaogaragemdesl" onclick="O
nOffG('+GD+')"> Fechar </button></div>';
    }else if (id == "GD"){
        var G =
        '<div class="garagemimg"></img>'+

```

```

        '<button type="button" id="BG" class="botaogaragem" onclick="OnOffG('+GL+')";"> Abrir </button></div>';
    }
    document.getElementById("alarme").innerHTML = document.getElementById("alarme").innerHTML
    document.getElementById("garagem").innerHTML = G
    document.getElementById("eletro").innerHTML = document.getElementById("eletro").innerHTML
    document.getElementById("iluminacao").innerHTML = document.getElementById("iluminacao").innerHTML
}
function OnOffE(id) {

    var xmlhttp = new XMLHttpRequest();
    xmlhttp.open("GET", "http://192.168.43.120/LED=ONOFFL2");
    xmlhttp.send();

    var EL="'EL'", ED="'ED'";
    if (id == "EL"){
        var E =
        '<div class="eletroimg"></img>'+
        '<button type="button" id="BE" class="botaoeletoresl" onclick="OnOffE('+ED+')";"> Desligar </button></div>';
    }else if (id == "ED"){
        var E =
        '<div class="eletroimg"></img>'+
        '<button type="button" id="BE" class="botaoeleetro" onclick="OnOffE('+EL+')";"> Ligar </button></div>';
    }
    document.getElementById("alarme").innerHTML = document.getElementById("alarme").innerHTML
    document.getElementById("garagem").innerHTML = document.getElementById("garagem").innerHTML
    document.getElementById("iluminacao").innerHTML = document.getElementById("iluminacao").innerHTML
    document.getElementById("eletro").innerHTML = E
}
function OnOffA(id) {

    var xmlhttp = new XMLHttpRequest();
    xmlhttp.open("GET", "http://192.168.43.120/LED=ONOFF");
    xmlhttp.send();

    var AL="'AL'", AD="'AD'";
    if (id == "AL"){

        var A =

```

```

        '<div class="alarmeimg"></img>'+
        '<button type="button" id="BA" class="botaoalarmedesl" onclick="On
OffA('+AD+');"> Desligar </button></div>';
    }else if (id == "AD"){
        var A =
        '<div class="alarmeimg"></img>'+
        '<button type="button" id="BA" class="botaoalarme" onclick="OnOff
A('+AL+');"> Ligar </button></div>';
    }
    document.getElementById("eletro").innerHTML = document.getElement
ById("eletro").innerHTML
    document.getElementById("garagem").innerHTML = document.getElemen
tById("garagem").innerHTML
    document.getElementById("iluminacao").innerHTML = document.getEle
mentById("iluminacao").innerHTML
    document.getElementById("alarme").innerHTML = A
}
function OnOffI(id) {

    var xmlhttp = new XMLHttpRequest();
    xmlhttp.open("GET", "http://192.168.43.120/LED=ONOFFL1");
    xmlhttp.send();

    var IL="'IL'", ID="'ID'";
    if (id == "IL"){
        document.getElementById('BI').className='botaoiludesl';
        var I =
        '<div class="iluing"></img>'+
        '<button type="button" id="BI" class="botaoiludesl" onclick="OnOf
fI('+ID+');"> Apagar </button></div>';
    }else if (id == "ID"){
        var I =
        '<div class="iluing"></img>'+
        '<button type="button" id="BI" class="botaoilu" onclick="OnOffI('
+IL+');"> Acender </button></div>';
    }
    document.getElementById("alarme").innerHTML = document.getElement
ById("alarme").innerHTML
    document.getElementById("eletro").innerHTML = document.getElement
ById("eletro").innerHTML
    document.getElementById("garagem").innerHTML = document.getElemen
tById("garagem").innerHTML
    document.getElementById("iluminacao").innerHTML = I
}

```

## APENDICE D – Código CSS da Automação Residencial

```

@charset "UTF-8";
.Logo{
  position: relative;
  width: 99.4%;
  display: block;
  color: white;
  text-align: center;
  border: 5px solid #9b5b71;
  font-size: 18px;
  font-family: 'Pacifico', cursive;
  background-color: #9b5b71;}
.Logo img{
  width: 70px;
  position: relative;
  align-content: center;}

.responsive {
  padding: 0px;
  float: left;
  width: 24.86%;
  background-color: black;
}
@media only screen and (max-width: 800px) {
  .responsive {
    width: 49.99999%;
    margin: 0px;
    background-color: black;
  }
}
@media only screen and (max-width: 500px) {
  .responsive {
    width: 100%;
    background-color: black;
  }
}
.clearfix:after {
  content: "";
  display: table;
  clear: both;
}
/* garagem//////////////////////////////////// */
/
.garagemimg{
  position: relative;

```

```

width: 100%;
border: 5px solid #72bb53;
background-color: white;
align-items: center;}
.imggaragem {
display: block;
margin-left: auto;
margin-right: auto;
max-width: 100%;
height: auto;}
.botaogaragem{
position: relative;
color: white;
width: 100%;
background-color: #72bb53;
font-family: 'Indie Flower', cursive;
border: none;
font-size: 15pt;
cursor: pointer;}
.botaogaragemdesl{
position: relative;
color: white;
width: 100%;
background-color: rgb(255, 4, 4);
font-family: 'Indie Flower', cursive;
/*border: 2px solid #72bb53;*/
border: none;
font-size: 15pt;
cursor: pointer;}

/*eletro////////////////////////////////////*/
.eletroimg{
position: relative;
width: 100%;
border: 5px solid #c31c4f;
background-color: white;
align-items: center;}
.imgeletro {
display: block;
margin-left: auto;
margin-right: auto;
max-width: 100%;
height: auto;}
.botaoeletro{
position: relative;
color: white;
width: 100%;
background-color: #c31c4f;
font-family: 'Indie Flower', cursive;

```

```

border: none;
font-size: 15pt;
cursor: pointer;}
.botaoLetrodesl{
position: relative;
color: white;
width: 100%;
background-color: rgb(255, 4, 4);
border: none;
font-family: 'Indie Flower', cursive;
font-size: 15pt;
cursor: pointer;}

/* alarme ////////////////////////////////////// */
.alarmeimg{
position: relative;
width: 100%;
border: 5px solid #8b51f5;
background-color: white;
align-items: center;}
.imgalarme{
display: block;
margin-left: auto;
margin-right: auto;
max-width: 100%;
height: auto;}
.botaoalarme{
position: relative;
color: white;
width: 100%;
background-color: #8b51f5;
font-family: 'Indie Flower', cursive;
border: none;
font-size: 15pt;
cursor: pointer;}
.botaoalarmedesl{
position: relative;
color: white;
width: 100%;
background-color: rgb(255, 4, 4);
border: none;
font-family: 'Indie Flower', cursive;
font-size: 15pt;
cursor: pointer;}

/* iluminação ////////////////////////////////////// */
.iluimg{
position: relative;
width: 100%;
border: 5px solid #3d8af7;

```

```
background-color: white;
align-items: center;}
.imgilu{
display: block;
margin-left: auto;
margin-right: auto;
max-width: 100%;
height: auto;}
.botaoilu{
position: relative;
color: white;
width: 100%;
background-color: #3d8af7;
font-family: 'Indie Flower', cursive;
border: none;
font-size: 15pt;
cursor: pointer;}
.botaoiludesl{
position: relative;
color: white;
width: 100%;
background-color: rgb(255, 4, 4);
border: none;
font-family: 'Indie Flower', cursive;
font-size: 15pt;
cursor: pointer;}
/*////////////////////////////////////*/
```